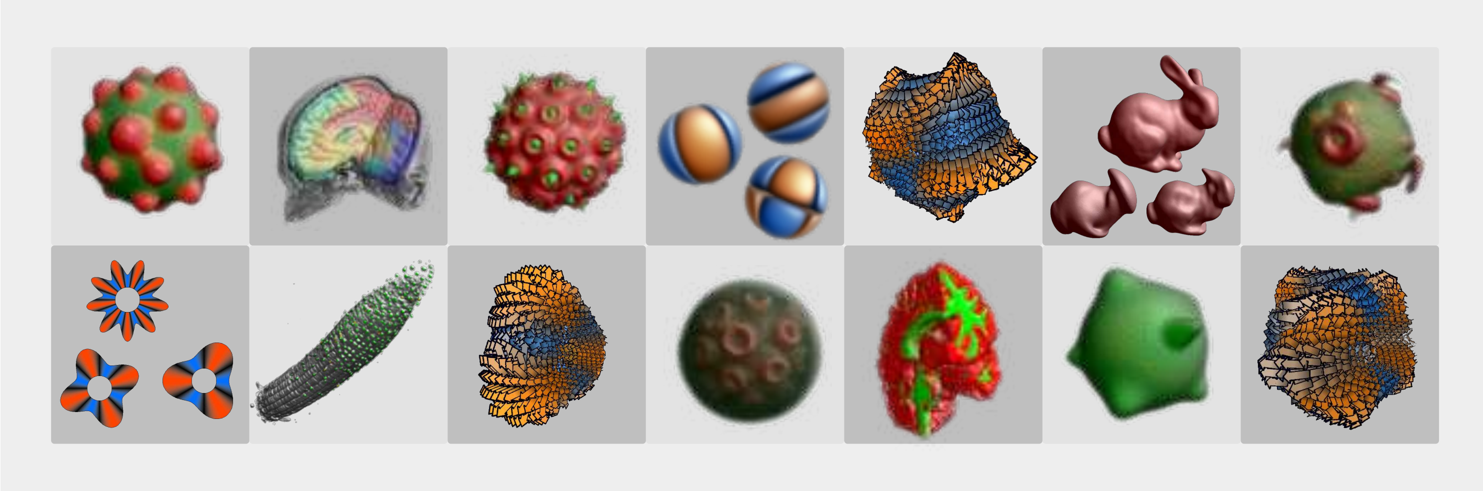
Spherical Tensor Algebra for Biomedical Image Analysis



Henrik Skibbe

2013

Dissertation zur Erlangung des Doktorgrades der Technischen
Fakultät der Albert-Ludwigs-Universität Freiburg im Breisgau

Dissertation

zur Erlangung des Doktorgrades der Technischen Fakultät der
Albert-Ludwigs-Universität Freiburg im Breisgau

Spherical Tensor Algebra for Biomedical Image Analysis

Henrik Skibbe

2013

Albert-Ludwigs-Universität
Freiburg im Breisgau

Dekan: Prof. Dr. Y. Manoli

Prüfungskommission: Prof. Dr. M. Teschner (Vorsitzender)
Prof. Dr. H. Burkhardt (Betreuer)
Prof. Dr. T. Brox (Gutachter)
Junior-Prof. Dr. O. Ronneberger (Beisitz)

Datum der Disputation: 18. Oktober 2013

Danksagung

Zuallererst möchte ich meinem Doktorvater Herrn Prof. Hans Burkhardt für das Vertrauen und die großartige Zusammenarbeit und Unterstützung danken ohne die ich niemals die Gelegenheit gehabt hätte diese Arbeit zu schreiben. Die Jahre an ihrem Lehrstuhl waren eine der lehrreichsten und spannendsten Jahre meines Lebens.

Besonderem Dank gilt auch Herrn Prof. Thomas Brox. Zum einen natürlich für die Zeit sich ausgiebig mit meiner Arbeit zu beschäftigen, aber vor allem auch für die fruchtbare Zusammenarbeit und das Vertrauen in der gemeinsamen Zeit am Lehrstuhl.

Danken möchte ich auch Herrn Junior-Prof. Olaf Ronneberger für die großartige Möglichkeit in einer interdisziplinären Arbeitsgruppe zu forschen und Erfahrungen zu sammeln. Die Arbeiten von Herrn Ronneberger waren in vielerlei Hinsicht eine Inspiration für meine eigenen Arbeiten.

Besonderer Dank geht auch an Herrn Prof. Matthias Teschner, der sich für meine Disputation Zeit genommen hat.

Ein sehr großes Dankeschön geht an meinen Kollegen und Freund Marco. Die unzähligen Diskussionen (mit und ohne Cocktail) haben maßgeblich zum Gelingen dieser Arbeit beigetragen. Ich hoffe, dass wir auch in Zukunft Gelegenheiten finden werden weiter zusammenarbeiten zu können.

Auch ein großes Dankeschön für die großartige Zeit und die Zusammenarbeit an meine ehemaligen Kollegen vom Lehrstuhl für Mustererkennung und Bildverarbeitung. Ein Danke an Thorsten, Janis, Margret, Maja, Stefan, Robert, Kun, Benjamin und Cynthia. Ein Danke mit Plus an Alexandra, die mir während der Diplomarbeit schon die Gelegenheit gab mich mit dem Thema *Features* zu beschäftigen.

Ein weiters Dankeschön geht auch an meine ehemalige Arbeitsgruppe von der Medizin Physik des Universitätsklinikums Freiburg Valerij, Elias und Matthias.

Natürlich wäre all dies nicht ohne die Unterstützung meiner Familie möglich gewesen, die mich nicht nur als Schüler ertragen, sondern auch in meinem Studium und darüber hinaus großartig unterstützt hat. Mein ganz besonderer Dank gilt meiner Mutter Ursula, meiner lieben Schwester **Kerstin**, Michael, meiner Emi und Lucas.

Einen besonderen Dank auch an meine Freunde Rudolf in Matsuyama, Matthias und Kourosh für die Unterstützung, die wertvollen Diskussionen und vor allem die gute Zeit.

ZUSAMMENFASSUNG

In dieser Arbeit stellen wir neue Algorithmen für die Berechnung von diskriminativen Merkmalen von volumetrischen Bildern vor. Die vorgeschlagenen Techniken vereinfachen die Klassifikation und die Auswertung von volumetrischen biologischen und medizinischen (biomedizinischen) Bildern. Die vorgestellten Algorithmen sind kovariant bezüglich der Euklidischen Bewegungsgruppe.

Wir behandeln das schwierige Problem der automatisierten Annotation von volumetrischen biomedizinischen Bildern. Eine Annotation, gefolgt von einer Auswertung von Bildern, ist oft ein unverzichtbarer Bestandteil einer statistischen Analyse eines wissenschaftlichen Experiments, sei es eine Gruppenstudie in einem medizinischen Experiment, oder die Auswertung von tausenden, unterschiedlich behandelten Zellen in einem biologischen Experiment. Eine Annotation von biomedizinischen Bildern wird oft manuell durchgeführt. Dies ist nicht nur sehr zeitintensiv, sondern auch sehr fehleranfällig.

Ein volumetrisches Bild erlaubt die Analyse einer medizinischen oder biologischen Probe in ihrer vollkommenen 3D Repräsentation. Jedoch basieren viele existierende Algorithmen auf 2D Bildern, sodass es wegen der zusätzlichen dritten Dimension einen Bedarf an geeigneten Algorithmen gibt. Viele Techniken können leicht von 2D auf 3D erweitert werden. Zum Beispiel ist es einfach isotropisches Glätten, die Berechnung der kartesischen Ableitungen via zentralen Differenzenquotienten oder die Fourier Transformation von 2D auf 3D zu erweitern. Dazu ist nur die Betrachtung einer weiteren Dimension nötig, am eigentlichen Algorithmus ändert sich nichts. Komplett anders verhält es sich für die Rotation. Während eine Rotation in der Bildebene durch einen Winkel beschrieben wird, benötigen wir um eine 3D Rotation zu beschreiben drei Winkel.

Rotationen spielen insbesondere dann eine Rolle, wenn das Ergebnis einer Annotation von der Orientierung des dargestellten Objektes abhängt. Wir betrachten eine Abbildung die ein Bild auf ein annotiertes Bild abbildet als einen Filter. Die Filter die wir vorstellen bestehen aus zwei Komponenten: (1) Im ersten Schritt werden Bildmerkmale aus lokalen Bildausschnitten gewonnen. Als Ergebnis erhalten wir ein neues Merkmalsbild, in welchem jedes Bildelement die lokalen Bildeigenschaften eines Bildpunktes im Eingangsbild beschreibt. (2) Im zweiten Schritt verwenden wir bekannte Techniken des maschinellen Lernens um basierend auf den Bildmerkmalen eine punktweise Annotation des Bildes zu erhalten. Die Extraktion von Bildmerkmalen kann als ein Vorverarbeitungsschritt verstanden werden, der Bilder in eine für den Computer leichter zu bearbeitende und auszuwertende Darstellung überführt.

Wir beschäftigen uns in dieser Arbeit hauptsächlich mir der Merkmalsgewinnung unter Berücksichtigung der 3D Rotation. Wir stellen die theoretischen und praktischen Grundlagen vor um Filter basierend auf der spherischen Tensor Algebra, ein mathematisches Grundgerüst welches den Umgang mit 3D Rotationen erleichtert, zu entwickeln. Dank der spherischen Tensor Algebra können wir die wesentlichen Informationen von Bildern in einer rotationsinvarianten Weise darstellen. Dies

bedeutet, dass alle rotierten Versionen eines Bildes oder eines Bildausschnitts, auf das selbe Merkmal abgebildet werden. Dies wiederum vereinfacht die automatische Klassifikation. Basierend auf der spherischen Tensor Algebra stellen wir neue Filter vor, die bezüglich der Qualität der Ergebnisse und der Laufzeit zu signifikant besseren Ergebnissen führen als existierende Techniken wie 3D SIFT oder 3D SURF.

ABSTRACT

In this dissertation we introduce new algorithms for extracting discriminative features from volumetric images. The proposed techniques facilitate the classification and evaluation of volumetric biological and medical (biomedical) images in an $SE(3)$ covariant manner, where $SE(3)$ is the 3D Euclidean motion group.

We address the challenging problem of the automated annotation of volumetric biomedical images. The annotation and evaluation of biomedical images is often an indispensable step for the statistical analysis of scientific experiments, including group studies in medical experiments, or the evaluation of thousands of cells under different treatments in biological experiments. The annotation of biomedical images is often done in a manual manner, but manual annotation is a very time consuming, tedious and error-prone task.

A big advantage of a volumetric image representation is the fact that it allows analyzing a biological or medical specimen in its full 3D appearance. However, many existing algorithms rely on 2D images so that with the third dimension coming with the volumetric image representation, there come new demands for the design of appropriate algorithms. Many algorithms can be extended from 2D to 3D in a straight forward manner. For instance, it is easy to extend filters like isotropic smoothing filters, the computation of Cartesian derivatives via central differences or the Fourier transform from 2D to 3D by simply considering an additional dimension. However, when rotations are playing a role, the situation becomes completely different. While a rotation in 2D images can be described by only one rotation angle, we have three degrees of freedom describing a full 3D rotation.

The consideration of rotations becomes important whenever the orientation of objects influences the annotation results. In this thesis we regard a mapping of images to image annotations as a filter. The proposed kinds of filters are actually the result of concatenation of two filters: (1) a feature extraction step which maps local image patches to intrinsic signatures. The result is a feature image, where each element is a signature representing the characteristics of a voxel surrounding in the input image. (2) A machine learning component utilizes these signatures to solve the annotation task in a voxel-by-voxel manner. That is, the feature extraction step transforms images or image patches into representations that are easier for the computer to process, evaluate and classify.

We emphasize on the feature extraction step with a particular focus on the 3D rotation. We introduce the theoretical and practical basis to form filters based on spherical tensor algebra, a mathematical framework which facilitates handling of 3D rotation. Thanks to spherical tensor algebra we are capable to extract intrinsic information from local image patches in a rotation invariant manner. That is, all rotated version of an image patch are mapped to the same intrinsic signature, which in turn, facilitates the classification for the machine learning part. We introduce novel filters based on tensor algebra performing significantly better than existing approaches, among them filters based on 3D SIFT and 3D SURF features, regarding quality of the results and computation performance.

Contents

1 INTRODUCTION

Volumetric images are, opposed to classical planar images, images of objects in three physical dimensions. In Fig. 1.1 on the following page we show a common representation of volumetric images. Particularly in the life-sciences, volumetric images have gained in importance during the last decades. Therefore, there is a great interest in having techniques for automatically analyzing such images. However, volumetric images are coming with new challenges that did not occur for classical 2D images. Hence not all algorithms that perform well for 2D images work for their volumetric counterparts. The first part of this introduction draws the differences between classical 2D images, like photographs, and volumetric, biological and medical (biomedical) images. We further introduce the process of pattern recognition with a particular focus on image analysis and we concentrate on the challenges occurring when addressing the problem of detecting and classifying objects and structures in volumetric images. In this context, we introduce the terminology of covariance and invariance both playing a major role in this theses. The introduction ends with an overview of related work.

In this thesis we introduce novel techniques for the computer aided annotation of biomedical images. The provided solutions are based on mathematical principles known in the fields of quantum mechanics and theoretical chemistry. The main contributions are new algorithms for the extraction of intrinsic features from patterns within volumetric images. For this, we utilize spherical tensor algebra to extract features that undergo well defined, predictable transformations under full 3D image translations and rotations. In combination with existing machine learning techniques, the proposed features show a remarkable performance in several object detection and classification tasks, outperforming state-of-the-art features like 3D SIFT and 3D SURF.

1.1. Volumetric Images

In our modern digital world, the technique of acquiring, storing and processing 2D images is available for almost everyone. This is mainly because of user friendly digital cameras and camcorders, cheap computers, the omnipresence of digital storages as well as the existence of many easy-to-use image processing programs. These techniques are accessible to all of us without requiring being an expert in the field of digital image processing and image analysis. For instance, modern digital cameras not only automatically choose the best settings regarding illumination, they also can make use of face detection algorithms to automatically choose the best settings for taking a portrait picture. Another example is the usage of image post-processing techniques. Almost any state-of-the-art image processing tool can do a gamma correction, a histogram equalization, implements a technique for sharpening or blurring an image or changing the saturation of the image's colors. The user is not directly faced with the underlying techniques: he only needs to adjust some parameters via sliders and efficient implementations ensure that the user sees the result in

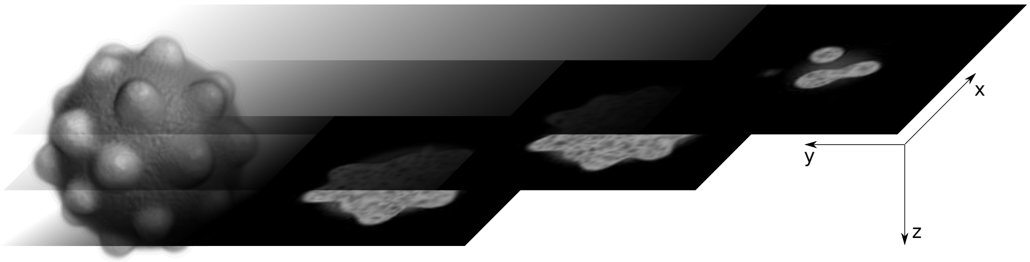


Figure 1.1.: Volumetric images are often acquired in a slice-by-slice manner. Therefore, they are often called a z-stack.

real-time. Hence all steps beginning with the image acquisition, to the printing of the final image can be done in an intuitive manner. However, digital cameras and camcorders are mimicking the human eye and the human perception of our world. The resulting image is nothing else than a projection of the real world onto a flat piece of paper. What we see is always the light reflected from the objects surfaces and captured by the camera lens.

In contrast to conventional 2D images, there exist techniques like computer tomography (CT), magnetic resonance imagining (MRI) or confocal laser scanning microscopy (CLSM) that allow us to study organisms in-vivo, which means, they allow capturing images within living organisms. For that reason, these techniques are of particular importance in biological and medical science. Moreover, these techniques allow acquiring volumetric images. Volumetric images are often a stitched assembly of several conventional 2D images that have been recorded in different depths. MRI, CLSM or CT techniques are usually taking 2D images from a plane lying within a biological sample that is in focus. For instance, in case of the CLSM, a laser beam that is steered by mirrors is used to illuminate tiny volumetric fractions within a biological sample. The detected light originating from such an illuminated focal point represents one pixel within the final image. The focus point can be moved within an xy-plane, thus the laser illuminates the specimen in a point-by-point manner. A 2D image is captured pixel-by-pixel (image x-direction) and line-by-line (image y-direction). In order to acquire a volumetric image the focus plane is slightly moved perpendicularly to the xy-plane (z-direction) by mechanically moving the specimen. Since volumetric images are typically recorded in a slice-by-slice manner volumetric images are sometimes called a z-stack; see Fig. 1.1. The specimen is often coarser sampled in z-direction than in the focal plane. Hence the voxel size of volumetric images is not necessarily cubic. A big advantage of the CLSM technique, for example, is that specific parts of a biological sample can be marked with fluorescent molecules so that light is emitted with a specific wave length. With suitable filters the light of different kinds of markers can be captured separately, allowing for selectively acquiring images of specific parts of a biological sample. An often occurring problem is that depending on the optical density of the sample, a certain amount of light is absorbed. This might lead to a decreasing intensity of a recorded volumetric image in the z-direction. Moreover, light is usually spread in an anisotropic way, that is, the acquired image is blurred in an anisotropic manner. Then, the acquired image is the result from a convolution of the “true image” of the biological sample with an (anisotropic) point spread function (PSF), which sometimes makes an initial deconvolution indispensable (Shaw and Rawlins 1991).

Along with the advent of these new acquisition techniques, there is an upcoming need for new image processing and image analysis techniques particularly designed to cope with these special kinds of volumetric data. MRI, CLSM or CT offer us new ways for studying and analyzing living organisms in their natural constellation, but at the same time, many parts of the analysis and the evaluation of biomedical images are still done in a manual or semi-automated manner. For instance, biologists or pathologists are often facing problems such as counting the occurrence of specific cells in microscopical images of cell cultures or within tissue sections. This often includes the classification of cells with respect to cell types and cell state. Considering the fact that volumetric images of biological samples might contain thousands of cells it turns out that scientists are often wasting hours on such tedious tasks. Further tasks occurring in a medical setting as well as in a biological scenario include the co-localization of specific structures or landmarks within images of different subjects. For instance, a clinical group study which is comparing the connectivity of the neuronal fibers within brains of different subjects is only possible if corresponding points and areas within the brain of different subjects can be found reliably. Another biologically related example is the co-localization and comparison of protein patterns within Zebrafish embryos (Ronneberger et al. 2012).

In this thesis, we propose new techniques for the detection and classification of structures, landmarks and objects within volumetric images. The whole procedure mimics a manual evaluation and annotation: given a volumetric image, we obtain the position and classification of objects within the image in terms of a label image of the same extent as the input image. In case of cell detection and classification the voxels of the resulting label image would contain either a label that denotes the background, or if a voxel coincides with a cell’s center, a label representing the cell type. If we are only interested in the number of cells of a specific cell type, we just run over the label image in a voxel-by-voxel manner and count the number of the corresponding occurring labels. If the position is also of interest, then we can easily automatically grab the cells positions from the label positions.

1.2. Detecting Patterns in Images

This section briefly introduces the concept of pattern recognition and the role of features. For further readings we refer to textbooks covering the fundamentals in pattern recognition and machine learning. The process of assigning labels to patterns is called *pattern recognition* and belongs to the research field of machine learning. The patterns are usually acquired from the real world via an analog image acquisition technique and are then transformed into a discrete signal that allows further digital processing steps. Such a discrete representation carrying intrinsic information about a real world object is called a *pattern*. The pattern may undergo some further initial processing steps. In case of a microscopical image, it might be a deconvolution (deblurring) or gamma correction. The cascade of such initial manipulations is called *pre-processing*.

The assignment of a label to the pattern can be e.g. categorical, like {"car", "pedestrian", "unknown"} or just numbers {1,2,3,⋯,N}. We say each label represents a *class* and the patterns are *instances* of a class. In this case, the space of all possible labels is discrete and usually finite and the assignment is called a *classification*. If the space of labels is continuous, then it is called a *regression*.

In our scenario, the patterns are images or local image patches within a large image. We illustrate the process in Fig. 1.2 on the facing page: (a) a volumetric image has been acquired via a microscope and (b) has been represented in terms of a three dimensional discrete voxel grid. We are interested in locally searching in the volumetric image for specific cells in a voxel-by-voxel manner. (c) Therefore, our patterns are small image fractions called *local patterns*. We skip (d) for the moment and continue with (e): the aim in this example is the assignment of labels to the local patterns that are either {cell type 1, cell type 2, background}. A *background* class is often used for collecting all patterns not belonging to one of the classes that are of interest. (f) Since this is done in a voxel-by-voxel manner we obtain the classification results in terms of a *label image*. Note that similar to the outcome of a manual annotation the label image (f) is the result of the automated annotation of all local input patterns (c) and thus the annotation of the input image (b).

We denote the mapping of patterns to class labels by the function $\Gamma : P \rightarrow L$. The function Γ is called the *classifier*. The space P of all patterns is called the *pattern space*. The elements of P are usually high dimensional vectors which we denote by bold letters $\mathbf{p} \in P$. The space L of all labels is called *output space*. An assignment $(\mathbf{p}, l) \in P \times L$ that we know that is correct is called a *ground truth*. The function Γ depends on a *model* that is created during a *learning* step. The creation of a model is commonly done in a data driven way, which is the machine learning part of the pattern recognition process. Pattern recognition can be dived into two categories depending on the way a model is obtained: *supervised learning* and *unsupervised learning* (It is worth mentioning that there exists a mixture of both called *semi-supervised learning*).

Supervised learning is usually performed by providing labeled training data. A set of training data is called a *training set*. A training set consists of (often manually) labeled pairs $\{(\mathbf{p}_1, l_1), (\mathbf{p}_2, l_2), \cdots, (\mathbf{p}_N, l_N)\}$, $(\mathbf{p}, l) \in P \times L$ and machine learning techniques are used to generate a model that performs as well as possible on the training set. Since the ground truth is given, the performance of a model can always be evaluated and optimized. Popular classifiers are decision trees like random forests (Breiman 2001), naive Bayes classifier (John and Langley 1995), neuronal networks (Rosenblatt 1958) or support vector machines (Cortes and Vapnik 1995).

In unsupervised learning, the training set is unlabeled. The aim is to find hidden structures within the training set that allow for grouping the patterns into classes. A classifier would then map new patterns to one of theses classes. Since no ground truth is provided a model can hardly be evaluated, which makes an evaluation of the model difficult. Popular techniques for unsupervised learning are k-means clustering (Seber 1984; Spath 1985) or hierarchical clustering methods (Jain and Dubes 1988; Zahn 1971).

Learning a model is always a trade-off between perfectly matching the training set and *generalization* to new, "unseen" data. The effect of perfectly matching the training set, but not being capable to generalize to new patterns is called *over-fitting*. This

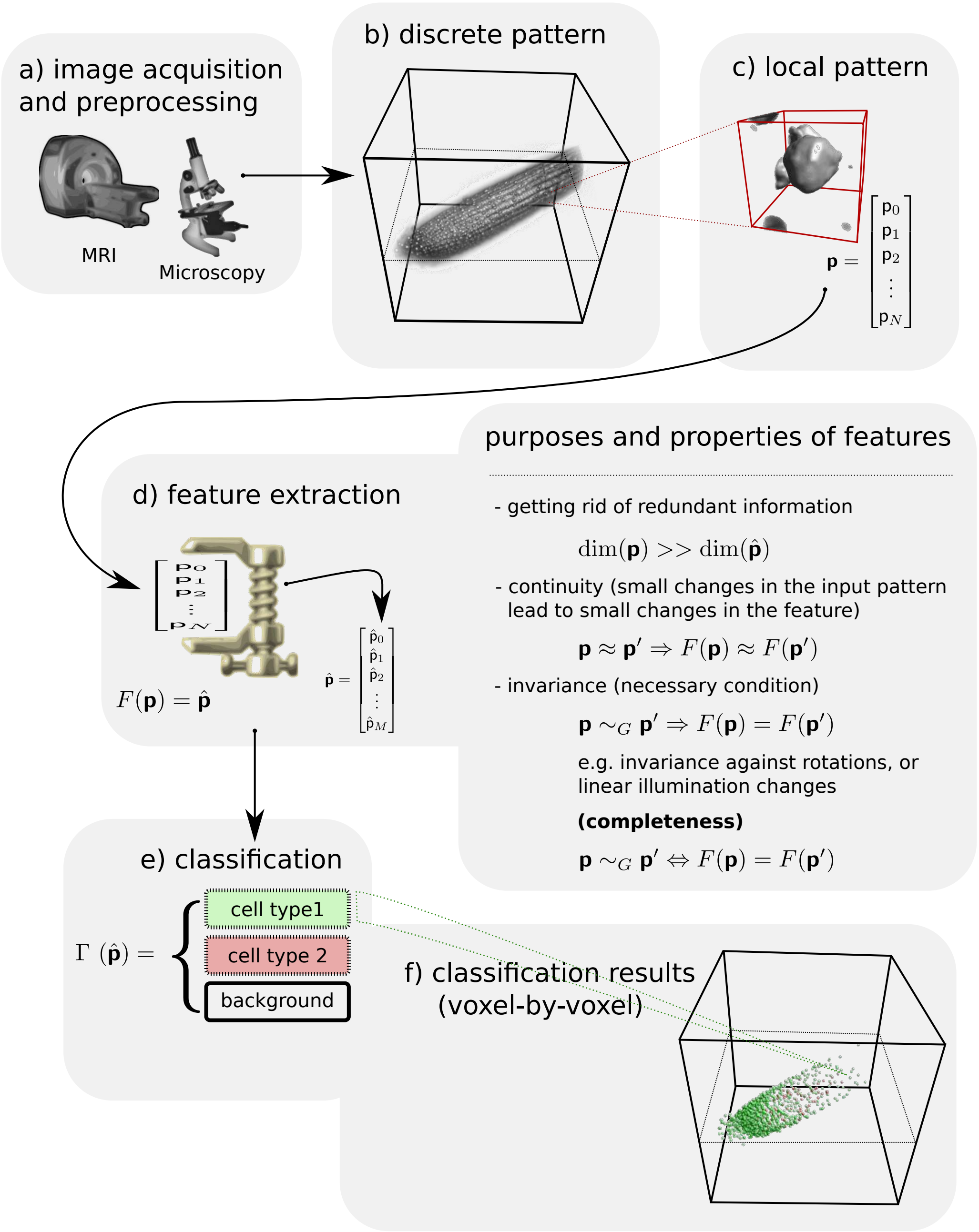


Figure 1.2.: Classification of volumetric images in a voxel-by-voxel manner.

often occurs when the dimension of the pattern space is larger than the number of instances in the training set. However, in many scenarios we have prior knowledge about which kind of patterns occur and which features are important for a successful classification. Due to this fact, it is often reasonable to utilize this knowledge in order to reduce the dimension of the pattern space by extracting intrinsic features from patterns so that the dimension of a pattern is reduced to the dimension of a feature. Features can be regarded as highly discriminative signatures of patterns. Consequently, training and classification becomes "easier" for the classifier as regards to number of training samples and computational effort.

This is achieved by introducing an intermediate feature extraction step $F : P \rightarrow \mathbb{C}^n$ that maps patterns to intrinsic feature vectors of a *feature space* $\mathbb{C}^n$; see Fig. 1.2 on the previous page (d). Hence the classification of patterns becomes a concatenated function $\Gamma \circ F : P \rightarrow L$ that first maps patterns $\mathsf{p} \in P$ to features $\mathsf{v} = F(\mathsf{p}), \mathsf{v} \in \mathbb{C}^n$ and then the features to labels of the output space L, where $l = \Gamma(F(\mathsf{p})) = \Gamma(\mathsf{v})$. While the creation of the model for Γ stays a machine learning part that builds upon a training set, the design of F is usually done manually.

1.2.1. Covariant and Invariant Pattern Recognition

Particularly in case of time-dependent or position-dependent patterns, we often expect that a classification or regression is time- or position-dependent, too. This means, the output of the classifier transforms according to the input pattern. For the definitions below we use the same notation as Reisert (2008).

For example, consider the computation of the optical flow of two consecutive frames in a movie (Horn and Schunck 1981; Lucas, Kanade, et al. 1981). The computation of an optical flow is a regression problem. We expect that if the image content in both frames is translated, the resulting optical flow field is translated, too. We say the function Γ is translation *covariant* (also called equivariant).

Definition 1.2.1 (Covariance (or Equivariance))**.** *Given a function $f : X \rightarrow Y$ with an input space X and an output space Y. Given a transformation group G whose elements we denote by $\mathfrak{g}$. Let $\boldsymbol{\tau}_{\mathfrak{g}}^X$ and $\boldsymbol{\tau}_{\mathfrak{g}}^Y$ representing the group action of $\mathfrak{g}$ on X and Y, respectively. We say a function f is covariant to G, if*

$$\forall \mathfrak{g} \in G : \; f(\boldsymbol{\tau}_{\mathfrak{g}}^X \mathsf{p}) = \boldsymbol{\tau}_{\mathfrak{g}}^Y f(\mathsf{p}) \quad . \tag{1.1}$$

Let us further consider an example where we aim at classifying cells. Assume that images each showing a single, centered cell, should be classified into different cell types. The classification result must not depend on the cell orientation, thus a model for the classifier has been created accordingly. In this case, the output of the mapping Γ does not depend on the rotation acting on the input space. We say the function Γ is rotation *invariant*.

Definition 1.2.2 (Invariance)**.** *Given a function $f : X \rightarrow Y$ that is covariant to G. We say f is invariant to G, if $\boldsymbol{\tau}_{\mathfrak{g}}^Y$ is the identity function $\boldsymbol{\mathcal{I}}^Y$ in Y. That is,*

$$\forall \mathfrak{g} \in G : \; f(\boldsymbol{\tau}_{\mathfrak{g}}^X \mathsf{p}) = \boldsymbol{\tau}_{\mathfrak{g}}^Y f(\mathsf{p}) = \boldsymbol{\mathcal{I}}^Y f(\mathsf{p}) = f(\mathsf{p}) \quad . \tag{1.2}$$

Hence invariance is a special case of covariance, see Fig. 1.3. Note that we can make any classifier invariant to a transformation group by making the feature extraction F invariant. The transformation is then just hidden for the classifier Γ.

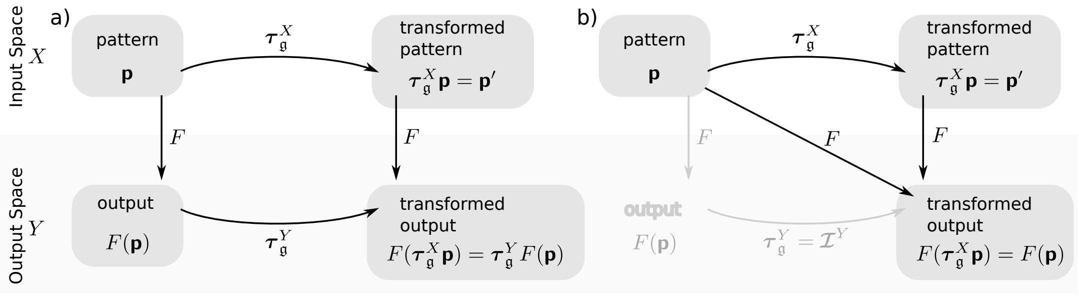


Figure 1.3.: a) Covariance, and b) Invariance as a special case of covariance.

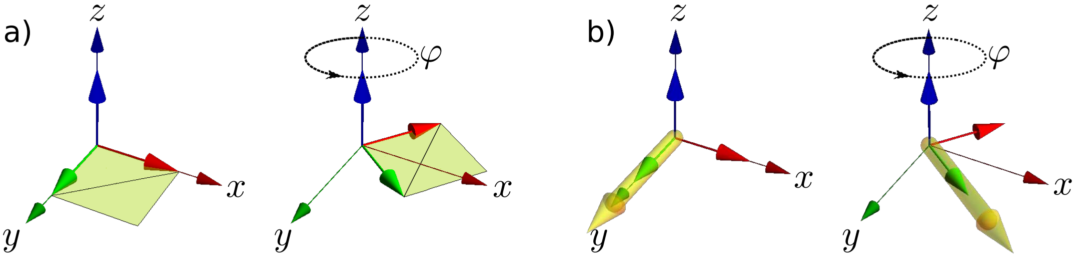


Figure 1.4.: Rotation in 2D is fully describes by one angle φ. This includes whole objects as well as gradients. The rotation angles φ is ranging from 0 to 2π.

1.3. Object Detection in Volumetric Images

At a first glance, a volumetric image is just a 2D image with an additional third dimension. However, regarding detection and classification, there are several important differences regarding the design of features between biomedical volumetric images on the one hand and "classical" 2D pictures and photographs on the other hand.

A big topic in the computer vision community is the detection of objects or persons within photographs. For instance, algorithms for the camera based detection and classification of pedestrians are of particular importance for the development of new car assistant systems, as well as for the surveillance of public areas such as airports or train stations; see e.g. the pedestrian detection benchmark of Dollár et al. (2009) for a survey on this topic or the survey of Geronimo et al. (2010). One issue is scale. It is hard to tell from a pure picture how tall a person is. Without involving nearby objects into our judgment, it is nearly impossible to conclude whether a person is short or just far away. Therefore, scale space theory gained in importance (Lindeberg 1993; Lowe 2004). A much bigger issue is occlusion. Occlusions irrevocably remove information that might be indispensable for detecting or classifying an object. In order to cope with partial occlusions objects are often described in terms of the appearance and constellation of local and canonical image patches. Such canonical image patches are just small fractions of the image representing e.g. an edge or corner or a certain type of texture. If it happens that parts of an object are occluded, then it is still likely to identify an object by the remaining image patches; (Leibe et al. 2004; Mikolajczyk et al. 2006). A minor issue is rotation within the image plane. A rotation in the image plane is just a linear transformation depending on one rotation angle; see Fig. 1.4 on the preceding page. Therefore, a rotation invariant feature representation can be obtained easily, and such representations facilitated the detection of structures or objects within photos and images in a variety of applications (Khotanzad and Hong 1990; Lowe 2004; Morel and Yu 2009; Sheng and Shen 1994). However, considering photographs of the real world, the rotation of objects in the image plane is by far not sufficient to model a real world rotation of objects. Just consider the scenario, where a person is turning around thus a photo is taken first from their front and then from their backside. Obviously, there is no rotation that turns the first picture into the second one. This is true for almost any kind of objects, such as cars or animals. Accordingly, the rotation in the image plane became less important in many state-of-the-art detection frameworks and algorithms went over to completely ignore the rotation by just treating a car's front and a car's side as two different kinds of representations of a car (Bourdev and Malik 2009; Felzenszwalb et al. 2010). Only for specific applications like the analysis of aerial views the consideration of a planar rotation is highly beneficial. Applications are including the detection of aircrafts, houses or cars (Heitz and Koller 2008; Noronha and Nevatia 1997; Schmidt and Roth 2012; Vedaldi et al. 2011).

This is contrarily to biomedical images. For biomedical images we usually know the exact size of a voxel. Therefore, we exactly know the size of structures and objects within an image. Hence scale is often a rich source of information telling us something about cell growth or allowing us to distinguish between different species. Occlusions occur always with respect to a certain point of view. Having volumetric images the problem of occlusion does not exist in a way that it exist in photographs. However, we can sometimes observe an effect of occlusions in form of absorption where objects are absorbing light so that objects "behind" are less illuminated. This is often the case for microscopical images. Moreover, in contrast to 2D images, a 3D rotation in the real world corresponds to the same 3D rotation acting on the object within the image. This is also true for 3D translations. Another major point is the amount of memory necessary for representing a volumetric image. It is noticeably larger than for a 2D image. Consider the following example: a 128×128 images contains 16,384 pixels. Assuming double precision we would need 0.125 megabyte for such an image. A volumetric image of size $128\times128\times128$ contains already 2,097,152 voxels which corresponds to 16 megabyte. For 256^3 we already have 128 megabyte in contrast to 256^2 where we only need 0.5 megabyte. Note that 0.5 megabyte can be completely stored in the second level cache of modern CPU's which is an additional plus for 2D images. Hence algorithms like the fast Fourier transform which are extremely fast on 2D images are noticeably slower when working with volumetric images.

1.3.1. $SE(3)$ Covariant Object Detection in Volumetric Images

The special Euclidean motion group is denoted by $SE(n)$ where n denotes the dimension. The group $SE(n)$ is a semidirect product of the n-dimensional translation group $T(n)$ and the n-dimensional rotation group $SO(n)$. Hence the elements of the Euclidean motion group are rotations and translations in Euclidean space. For $n = 2$

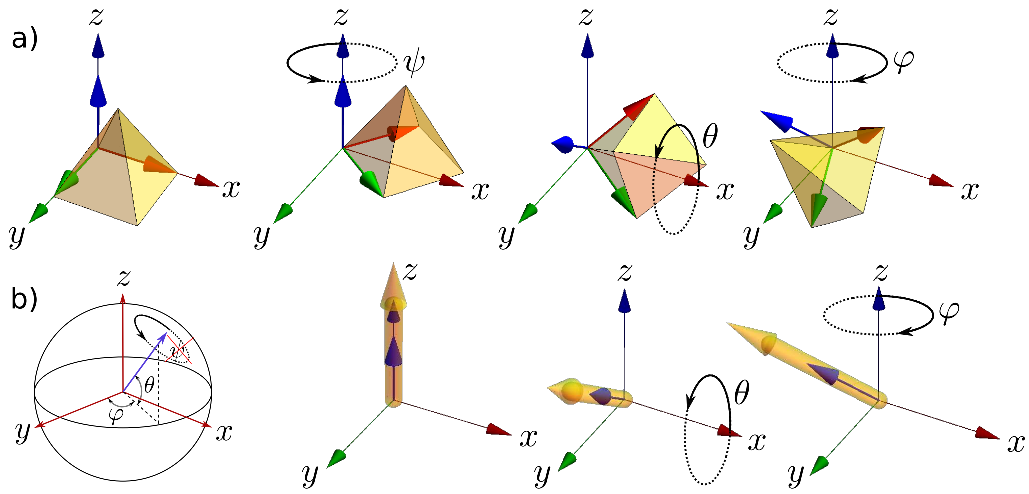


Figure 1.5.: a) Full rotation in 3D in terms of Euler angles (z,x,z - convention), and b) Rotation of gradients. The rotation angles ψ,φ are ranging from 0 to 2π, θ is ranging from 0 to π.

there are three degrees of freedom: Euclidean motion in the image plane consists of a rotation, parametrized by one angle φ (see Fig. 1.4 on page 19) and the translation in x and y direction, respectively. In case of volumetric images ($n = 3$), there are six degrees of freedom: a rotation with respect to the three Euler angles ψ,θ,φ, see Fig. 1.5 and a translation into the x, y and z direction, respectively.

In our scenario, the biomedical images are represented by scalar valued functions $I : \mathbb{R}^3 \to \mathbb{C}$, mapping from the Cartesian coordinates $\mathbb{R}^3$ to the complex numbers (we consider volumetric images as complex valued despite the fact that most of them are real valued). Hence our pattern space P is the space of scalar valued functions defined on $\mathbb{R}^3$. We define Euclidean motion acting on images in terms of a couple of an image translation $\boldsymbol{\tau}^P_{\mathbf{t}}$ and an image rotation $\boldsymbol{\tau}^P_{\theta}$. With $\mathbf{t} \in \mathbb{R}^3$ we denote a translation vector. The angle θ is a placeholder for the three Euler rotation angles. We define the image translation by

$$(\boldsymbol{\tau}^P_{\mathbf{t}} I)(\mathbf{r}) := I(\mathbf{r}-\mathbf{t}) \quad , \tag{1.3}$$

and the rotation around the origin by

$$(\boldsymbol{\tau}^P_{\theta} I)(\mathbf{r}) := I(\mathbf{U}^T_{\theta}\mathbf{r}) \quad . \tag{1.4}$$

We denote image coordinates by $\mathbf{r} \in \mathbb{R}^3$. With $\mathbf{U}_\theta \in \mathbb{R}^{3\times 3}$, $\det \mathbf{U}_\theta = 1$ we denote an ordinary 3D rotation matrix (remember, that the inverse representation of a rotation is just the transposed matrix). Hence full Euclidean motion acting on an image is defined by

$$(\boldsymbol{\tau}^P_{\mathbf{t}} \boldsymbol{\tau}^P_{\theta} I)(\mathbf{r}) = I(\mathbf{U}^T_{\theta}(\mathbf{r}-\mathbf{t})) \quad . \tag{1.5}$$

Note that $\boldsymbol{\tau}^P_{\mathbf{t}} \boldsymbol{\tau}^P_{\theta} I = \boldsymbol{\tau}^P_{\theta} \boldsymbol{\tau}^P_{(\mathbf{U}^T_{\theta}\mathbf{t})} I$, that is, the order of rotation and translation can be exchanged (identity can be shown by using the substitution rules of image translations and rotations).

While for scalar valued images the Euclidean motion can be defined purely by a coordinate transform, vector- or even tensor valued images (like a gradient field or a Hessian matrix field) undergo further transformations under rotations: a scalar valued element of an ordinary image just changes its position under motion. But a vector (or tensor) of a vector field transforms with respect to both, coordinate and value. Let $\mathbf{f} : \mathbb{R}^3 \rightarrow \mathbb{C}^n$, $\mathbf{f} \in X$ be a vector valued function, where X is the space of complex vector valued functions defined on $\mathbb{R}^3$. They transform under rotations according to

$$(\boldsymbol{\tau}_\theta^X \mathbf{f})(\mathbf{r}) := \boldsymbol{\tau}_\theta^{\mathbb{C}^n} \mathbf{f}(\mathbf{U}_\theta^T \mathbf{r}) \quad , \tag{1.6}$$

where $\boldsymbol{\tau}_\theta^{\mathbb{C}^n}$ is a value transformation. For example, a gradient field $\mathbf{f} : \mathbb{R}^3 \rightarrow \mathbb{C}^3$ transforms according to $(\boldsymbol{\tau}_\theta^X \mathbf{f})(\mathbf{r}) = \mathbf{U}_\theta \mathbf{f}(\mathbf{U}_\theta^T \mathbf{r})$. The inner rotation matrix is acting on the coordinates, while the outer rotation matrix is rotating the vectors (the values of $\mathbf{f}$) themselves. Equation (1.6) can be regarded as a generalization of Eq. (1.4) to vector valued images, where for scalar valued images the value transformation is the identity function. Another example is an image whose single values do not *mix* under rotations or whose values are attributes which are not changing under rotations. This can be e.g. an images with several color channels. Each color component transforms itself like a scalar valued image. In such a case, the value transformation is the identity transformation, i.e. $(\boldsymbol{\tau}_\theta^X \mathbf{f})(\mathbf{r}) = \boldsymbol{\mathcal{I}}^{\mathbb{C}^n} \mathbf{f}(\mathbf{U}_\theta^T \mathbf{r})$.

The translation of vector valued functions is identical to scalar valued functions namely $(\boldsymbol{\tau}_\mathbf{t}^X \mathbf{f})(\mathbf{r}) = \mathbf{f}(\mathbf{r} - \mathbf{t})$.

1.3.1.1. $SE(3)$ Covariant Filters

Labeling a biomedical image can be regarded as a detection problem: we search for specific structures, landmarks or objects within images and assign labels to them. In general, we don't have prior knowledge about their number, location or orientation.

Detecting objects or structures in volumetric images is about as easy as finding a needle in the haystack. Since we neither know the position, nor the orientation of objects within an image, there is no alternative but to search for the objects in a point-by-point manner. Such a detection task is often a quite unbalanced problem: only a minor part of an image contains the objects that we are looking for. The largest part of an image is typically considered as background. Therefore, there is a great demand for efficient algorithms.

The proposed framework can be regarded as an "intelligent saliency detector". We are searching structures in images by focusing the search on a local image region: a window is sliding over the image in a voxel-by-voxel manner and for each region lying within the window an intrinsic feature vector is extracted. The whole process works like a filter: the input is a volumetric image $I \in P$, $I : \mathbb{R}^3 \rightarrow \mathbb{C}$, the output an image $\mathbf{f} : \mathbb{R}^3 \rightarrow V$ containing features extracted from local image regions in a voxel-by-voxel manner. We call the output image a *feature image*. Such a feature image can be regarded as a generalized "saliency map". The feature image can then be processed for further analysis, e.g. each voxel can be evaluated via a classifier for the absence or presence of certain structures or whole objects. In that way, we obtain an automated annotation of the input image.

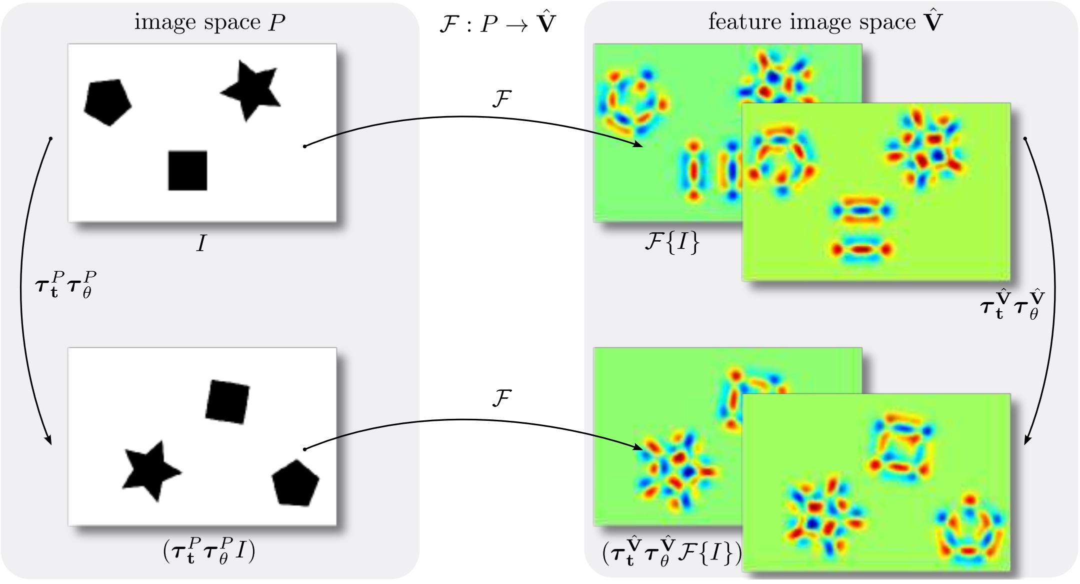


Figure 1.6.: A 2D example: $\mathcal{F} : P \to \hat{\mathbf{V}}$ is an $SE(2)$ covariant filter mapping images $I \in P$ to feature images $\mathcal{F}\{I\} \in \hat{\mathbf{V}}$ (the filter response). The elements of $\mathcal{F}\{I\}$ are locally extracted (here complex valued) features (above the real and imaginary components are shown). Since $\mathcal{F}$ is covariant to Euclidean motion, it holds that $\mathcal{F}\{\boldsymbol{\tau}_{\mathbf{t}}^{P}\boldsymbol{\tau}_{\theta}^{P}I\} = \boldsymbol{\tau}_{\mathbf{t}}^{\hat{\mathbf{V}}}\boldsymbol{\tau}_{\theta}^{\hat{\mathbf{V}}}\mathcal{F}\{I\}$, where $\boldsymbol{\tau}_{\mathbf{t}}^{P}\boldsymbol{\tau}_{\theta}^{P}$ are translation and rotation in image space and $\boldsymbol{\tau}_{\mathbf{t}}^{\hat{\mathbf{V}}}\boldsymbol{\tau}_{\theta}^{\hat{\mathbf{V}}}$ translation and rotation in feature image space, respectively. Note that rotation in feature image space transforms both coordinates and values. The values (here real and imaginary components) “mix” under rotations.

In our case, features are complex valued vectors, thus the feature space is $V = \mathbb{C}^n$. Features are extracted from regions in focus via a function $F : P \to \mathbb{C}^n$, mapping images $I \in P$ to features in a feature space $\mathbb{C}^n$. Possible features are the image gradient, the gradient magnitude or the Hessian matrix at the image point that is in focus. Alternatively, sophisticated features like local jets (Koenderink and Doorn 1987) build upon an assemble of local image derivatives are an option. Further examples are descriptors representing the local gradient orientation distribution like the structure tensor (Förstner 1986; Harris and Stephens 1988), SIFT (Lowe 2004) descriptors or HOG (Dalal and Triggs 2005) descriptors. A common property of these features is that they extract a vector valued feature from an image in a rotation covariant manner, that is

$$F(\boldsymbol{\tau}_{\theta}^{P}I) = \boldsymbol{\tau}_{\theta}^{\mathbb{C}^n}F(I) \quad , \tag{1.7}$$

i.e. for a rotation in image space exists a corresponding transformation in feature space. We call the output of F a *local* feature because it has been extracted from a certain location in the image and it is used for evaluating the content of an image at a certain position.

Now suppose we aim at using the local feature extraction F in a sliding window scenario. The whole process can be regarded as filter mapping images to feature valued saliency images: the input is a biomedical image, the output is a feature image. The feature image contains for all voxels the features extracted from within the sliding window. For this, the local feature extraction F is used for extracting intrinsic features from the regions in focus of a sliding window in a voxel-by-voxel manner. So technically, we center each voxel $\mathbf{r} \in \mathbb{R}^3$ (which we consider as current center of the window) with respect to the origin and extract a feature from the centered image using F. What we obtain is a feature representing the characteristics of the image around $\mathbf{r}$ (because we previously centered the image with respect to $\mathbf{r}$). The extracted feature is then stored back at position $\mathbf{r}$ for further evaluation of the window content. By continuing in a voxel-by-voxel manner we obtain one feature per voxel. That is, a new feature valued image, where each voxel from the input image has its corresponding feature in the output image. Such a feature image might then be evaluated with help of a classifier in a voxel-by-voxel manner with the purpose of detecting objects or structures.

We denote the space of vector valued feature images $\mathbf{f} : \mathbb{R}^3 \to \mathbb{C}^n$ by a bold $\hat{\mathbf{V}}$. A filter mapping biomedical images $I \in P$ to vector valued feature images $\mathbf{f} : \mathbb{R}^3 \to \mathbb{C}^n$ is denoted by $\mathcal{F} : P \to \hat{\mathbf{V}}$ ($\mathcal{F}$: images $\to$ feature images, which is different from the local feature extraction $F : P \to \mathbb{C}^n$ mapping images $\to$ features). The filter is induced by a local feature extraction F, which we use for extracting features in a voxel-by-voxel way:

$$\mathcal{F}\{I\}(\mathbf{r}) := F(\boldsymbol{\tau}^P_{(-\mathbf{r})} I) \quad . \tag{1.8}$$

That is, we create a new feature valued image $\mathcal{F}\{I\}$ (the filter response) whose elements are the locally extracted features from I. Such a filter is covariant to Euclidean motion, i.e. it holds that

$$\mathcal{F}\{\boldsymbol{\tau}^P_{\mathbf{t}} \boldsymbol{\tau}^P_{\theta} I\} = \boldsymbol{\tau}^{\hat{\mathbf{V}}}_{\mathbf{t}} \boldsymbol{\tau}^{\hat{\mathbf{V}}}_{\theta} \mathcal{F}\{I\} \quad , \tag{1.9}$$

where

$$(\boldsymbol{\tau}^{\hat{\mathbf{V}}}_{\mathbf{t}} \boldsymbol{\tau}^{\hat{\mathbf{V}}}_{\theta} \mathcal{F}\{I\})(\mathbf{r}) = \boldsymbol{\tau}^{\mathbb{C}^n}_{\theta} \mathcal{F}\{I\}(\mathbf{U}^T_{\theta}(\mathbf{r} - \mathbf{t})) \tag{1.10}$$

is just the $SE(3)$ covariant transformation of vector valued images. That is, any 3D rotation covariant local feature extraction F induces an $SE(3)$ covariant filter $\mathcal{F}$; a proof can be found in section B.1 on page 195. An $SE(3)$ covariant filter is illustrated in Fig. 1.6 on the preceding page.

On the other hand, any $SE(3)$ covariant filter $\mathcal{F} : P \to \hat{\mathbf{V}}$ induces a rotation covariant local feature extraction $F : P \to \mathbb{C}^n$ with

$$F(I) := \mathcal{F}\{I\}(\mathbf{0}) \;\Leftrightarrow\; \mathcal{F}\{I\}(\mathbf{r}) := F(\boldsymbol{\tau}^P_{(-\mathbf{r})} I) \quad , \tag{1.11}$$

that is, $\mathcal{F}$ can be fully recovered given F. A proof is given in section B.1 on page 195.

Hence it is only a matter of interpretation: we can either consider feature extraction as a local problem, where we extract (and process) local features that are covariant to $SO(3)$ in a voxel-by-voxel manner, or we consider the feature extraction as an $SE(3)$ covariant filter mapping images to $SE(3)$ covariant feature images. Both interpretations are equivalent.

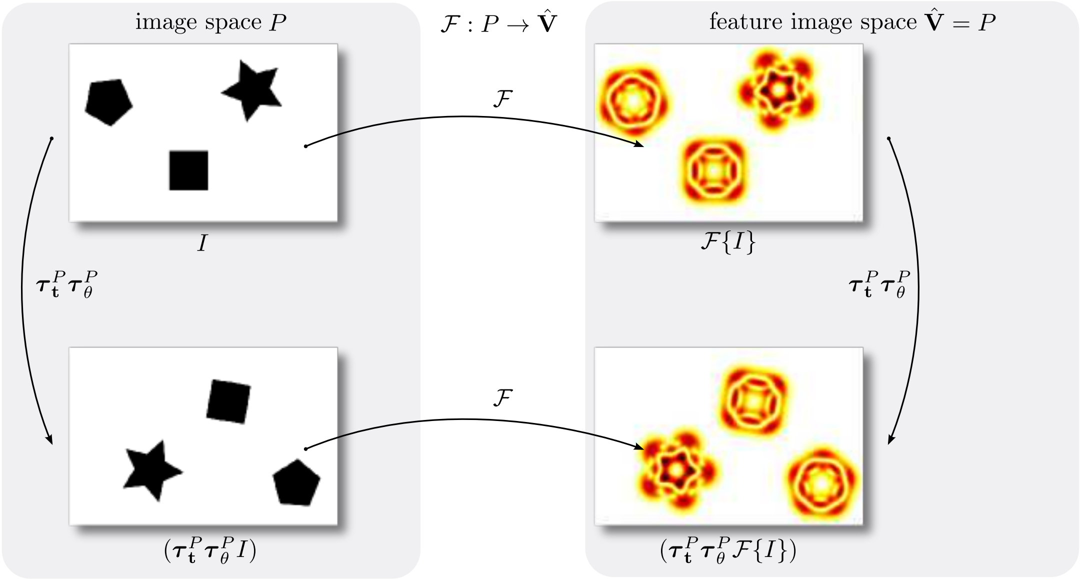


Figure 1.7.: A 2D example: $\mathcal{F} : P \to \hat{\mathbf{V}}$ is an $SE(2)$ covariant filter mapping images $I \in P$ to feature images $\mathcal{F}\{I\} \in \hat{\mathbf{V}}$ (the filter response). Here the output of $\mathcal{F}$ transforms in the same way as scalar valued images. The features are locally rotation invariant. Actually, the values of the filter response coincides with the magnitudes of the complex valued filter response of the example shown in Fig. 1.6 on page 23.

1.3.1.2. Local $SO(3)$ Invariant Features

In many applications it is beneficial to make local features invariant to local rotations, that is,

$$F(\boldsymbol{\tau}_{\theta}^{P}I) = \boldsymbol{\mathcal{I}}^{\mathbb{C}^n}F(I) = F(I) \quad . \tag{1.12}$$

The advantage for our applications is evident: cells or landmarks in biomedical images might occur in different orientations. Thanks to invariance to local rotations, a classifier $\Gamma : \mathbb{C}^n \to L$ becomes automatically capable to correctly classify cells despite their local orientation. A classifier Γ “inherits” the invariance from F, since $\Gamma(F(\boldsymbol{\tau}_{\theta}^{P}I)) = \Gamma(F(I))$. This facilitates both training and classification.

In this scenario, F induces an $SE(3)$ covariant filter whose output transforms similar to scalar valued images. The transformation of F is solely described by a coordinate transform, namely

$$(\boldsymbol{\tau}_{\mathbf{t}}^{\hat{\mathbf{V}}}\boldsymbol{\tau}_{\theta}^{\hat{\mathbf{V}}}\mathcal{F}\{I\})(\mathbf{r}) = \boldsymbol{\mathcal{I}}^{\mathbb{C}^n}\mathcal{F}\{I\}(\mathbf{U}_{\theta}^{T}(\mathbf{r}-\mathbf{t})) = \mathcal{F}\{I\}(\mathbf{U}_{\theta}^{T}(\mathbf{r}-\mathbf{t})) \quad . \tag{1.13}$$

Hence the different components in the output-space do not *mix* under rotations. More precisely, each component of the vector valued output of $\mathcal{F}$ transforms like a scalar valued image; see Eq. (1.5). It is further worth mentioning, that any feature that is covariant to local rotations can be turned into an invariant feature via group integration techniques. This is exactly the way we build the features proposed in this thesis, (1) by building a filter on covariant local features (2) and making the features invariant to local rotations. Such an $SE(3)$ covariant filter is illustrated in Fig. 1.7.

1.3.1.3. $SE(3)$ Covariant Detection in Feature Images

A label image (an automatically generated image annotation) is the outcome of an $SE(3)$ covariant mapping that maps images $I \in X$ to label images $g : \mathbb{R}^3 \to L$, where L is the set of possible classification results. We denote the set of all label images $\hat{\mathbf{L}}$. We can turn any $SE(3)$ covariant filter into such a mapping. This is done by concatenating a local feature extraction $F : P \to \mathbb{C}^n$ and a classifier $\Gamma : \mathbb{C}^n \to L$, forming a "new" $SE(3)$ covariant filter that incorporates the classification, namely

$$\mathcal{F}^L\{I\}(\mathbf{r}) := \Gamma(F(\boldsymbol{\tau}^P_{(-\mathbf{r})}I)) \quad , \tag{1.14}$$

where $\mathcal{F}^L : P \to \hat{\mathbf{L}}$.

An issue we must tackle in practice is the large number of negative examples within an image. Typically, only a few number of voxels count as detections. However, in practice, features corresponding to nearby voxels within an image are usually very similar to each other. Consequently, it is very likely that the same class label is assigned to nearby voxels, although we have prior knowledge that only one among them can be a correct detection. A solution for achieving a precise detection is combining a classification with a *confidence* value, reflecting the confidence that a classification is correct. The correct detection is found by locating local maxima within *confidence* values of neighboring voxels and count only them as "true" detections.

Most classifier are providing, in addition to a discrete classification result, a simple mathematical model that reflects the confidence that a classification is correct. In case of a simple nearest neighbor classifier, the number of nearest neighbors counting to the class that builds the majority can be regarded as a level of confidence. In case of a support vector machine, Platt's probabilities would be an option (Lin et al. 2007; Platt 1999). For random forests, the number of trees building the majority would be an alternative.

Example 1.3.1 (LoG Saliency Detector)**.** *A rather simple example is a classical saliency detector searching for blob-like structures based on the Laplace of the Gauss function (LoG detector). The Laplace of a Gaussian function is a blob like function so that filtering an image with such a function results in a suitable blob detector. In this scenario, the local feature extraction is the projection of the image onto the LoG function. Let* $I : \mathbb{R}^3 \to \mathbb{R}$ *be an image. Then the projection on the LoG is*

$$F(I) := \int_{\mathbb{R}^3} I(\mathbf{r})(\triangle e^{\frac{-x^2}{2t}})(\mathbf{r})d\mathbf{r}, \tag{1.15}$$

where $t \in \mathbb{R}_{>0}$ *defines the size of the blob. The Laplace of an isotropic Gauss function is again an isotropic function. This ensures that* F *is invariant to local rotations. Therefore,* F *induces a filter that transforms like a scalar valued image; see Eq.* (1.13)*. The filter output is an image representing the blob-lifelikeness for the whole image in a voxel-by-voxel manner. For the detection of blob like structures, local maxima within the feature image are detected (the feature image provides the confidence values) and depending on the magnitude, classified into blob/non-blob structures via thresholding.*

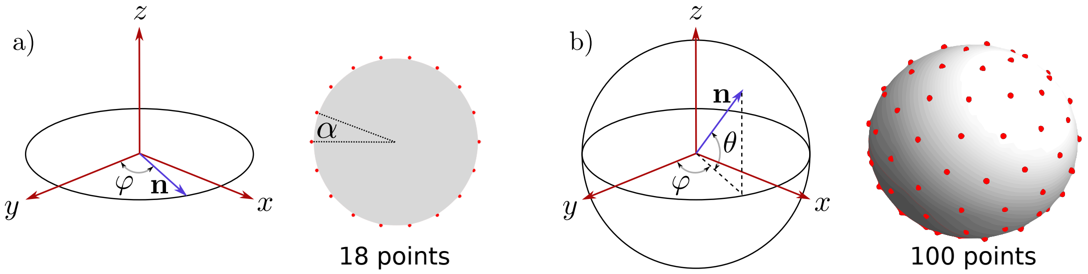


Figure 1.8.: Why avoiding sampling the 3D rotation space? This is mainly for two reasons: (1) in contrast to the 2D case, equidistant sampling of the orientation space is a nontrivial task. Consider the rotation of a gradient: Discretization of such a rotation is equivalent to distributing points on a sphere in an equidistant manner. This cannot be computed explicitly for many point configurations. (2) The number of necessary points required for sampling the sphere is significantly larger than for the 2D case (For $\alpha = 20°$ we have 18 points on a circle, and 100 points on the sphere). If we would consider all three Euler angles, then this would result in $100 \times 18 = 1800$ points! This often hinders a fast realization in 3D.

We have seen that we can quite easily create an $SE(3)$ covariant filter via locally extracted features. However, designing discriminative features which are locally invariant to 3D rotations is challenging. We master this challenge by making use of spherical tensor algebra, an appropriate tool for dealing with full Euclidean motion:

- Tensor operations allow for extracting intrinsic features from local image patches in an $SE(3)$ covariant manner.
- It is mathematically proven that regarding full, continuous rotations in 3D, spherical tensor representations are the most sparse and therefore most computational and memory efficient representations for 3D patterns.
- Local features based on spherical tensor representations can *always* analytically be turned into features which are invariant to local rotations. This can be guaranteed for all voxels.

1.4. Related Works

This work is inspired by the work of Ronneberger et al. (2005), and the work of Reisert and Burkhardt (2009a), both aiming at detecting local structures or objects in volumetric images in an $SE(3)$ covariant manner.

The proposed framework for $SE(3)$ covariant detection and classification of objects and structures within images can be regarded as an “intelligent saliency detector“. However, while locally rotation invariant salient point detectors purely rely on thresholding very simple features, like the determinate or trace of the structure tensor (Förstner 1986; Harris and Stephens 1988) or the Hessian matrix (Lowe 2004), the proposed, locally rotation invariant filters are much more discriminative. They are suited for detecting complex structures or whole objects in volumetric images. In the following we focus on related work on the extraction of features that are invariant to (local) rotations, because they build the key ingredient for our covariant filters. For a survey of existing "classical" saliency detectors we refer to Tuytelaars and Mikolajczyk (2008).

Designing discriminative features that are invariant to local 3D rotations is challenging for volumetric images. Hence related work on this specific topic is manageable and we decided that is worth mentioning some specific works on invariants for 2D rotations, too. Following Burkhardt and Siggelkow (2001), the systematic generation of invariants can be divided into three kinds of methods: normalization methods, invariance via group integration, and differential approaches. As far as we know, the latter one has not been used so far in the context of 3D rotation invariant features for volumetric images. The reason might be the fact that in practice, the resulting differential equations are often hard to solve. We will address the details regarding the generation of invariants in a chapter about 3D rotation invariants in this thesis. A forth kind of rotation invariant detection is rather based on an invariant classifier than on a rotation invariant features extraction. Creating a suited model for such a classifier requires a training set including rotated versions of patterns which are covering the high dimensional 3D orientation space. This might be a reason why such methods have not been explored for 3D rotations so far. It is further worth mentioning that template matching, including steerable filter and tensor voting approaches, is also some kind of rotation covariant detection in volumetric images. However, the aim of pattern matching algorithms is to find exact matches of existing patterns within images, while our aim is pattern recognition: designing features providing, in combination with a trainable classifier, reasonable feedback for all possible kinds of patterns within an image. Pattern matching goes beyond the scope of this thesis, but it is worth mentioning that the proposed mathematical frameworks can also be used in a pattern matching context. Related work on spherical tensors in this context is still rare. The interested reader is referred to the work of Reisert (2007); Reisert and Burkhardt (2008b).

1.4.1. 3D Rotation Invariance via Pose Normalization

The probably most intuitive methods for obtaining rotation invariant features are methods based on pose-normalization. This means that in case of rotations, local patterns are first rotated into a canonical representation before intrinsic features are extracted. The benefits are obvious: once the pattern is turned into its canonical orientation, rotation invariance is not required for the particular feature extraction process itself. However, determining the orientation parameters for turning a pattern into it's canonical orientation is often done in a heuristic manner. In such cases, rotation invariance cannot be guaranteed. A quite popular approach of this family of invariants is SIFT (Lowe 2004). It is fair to say that SIFT has revolutionized object detection in 2D, because SIFT and similar techniques are widely used in various state-of-the-art computer vision and pattern recognition algorithms. SIFT features are representing the local gradient orientation distribution in a rotation and scale invariant manner. Pose orientation normalization in SIFT is done with respect to the local gradient main direction. The computation of SIFT includes a pre-selection of salient points within the image, which most likely allow for a unique pose-normalization. That is, extracting a unique gradient main direction. SIFT and similar approaches like PCA-SIFT (Ke and Sukthankar 2004), ASIFT (Morel and Yu 2009), SURF (Bay et al. 2006) or DAISY (Tola et al. 2008) have been used in a broad variety of applications. Applications are ranging from face authentication (Bicego et al. 2006), object recognition, classification and retrieval (Lampert et al. 2008; Nister and Stewenius 2006), images stitching (Lin et al. 2011), state-of-the-art optical flow algorithms (Liu et al. 2011a) to action recognition (Scovanner et al. 2007) in the 2D+time domain and much else.

The wide range of applications of 2D SIFT-like descriptors is opposed to 3D SIFT-like descriptors, particularly with focus on applications involving biomedical images. The works on 3D extensions of SIFT are manageable and focused on solving point matching and registration problems. Cheung and Hamarneh (2007a); Cheung and Hamarneh (2007b) have proposed the n-D Sift descriptor which has been used for the co-registration of volumetric medical 3D and 3D+time images. This includes the 3D-SIFT descriptor of Allaire et al. (2008). The proposed 3D-SIFT descriptors have also been used for the registration of volumetric spectral OCT[1] images of the retina (Niemeijer et al. 2009) or the co-registration of 3D+time CT-scans of lung vessel trees (Osada et al. 2008; Paganelli et al. 2012). Recent applications on object recognition include the scanning of volumetric CT scans of bags in airports for security reasons (Flitton et al. 2010) and a voting based classification of objects in volumetric images (Knopp et al. 2010). Particualarly work on object recognition, detection and classification seems to be very rare in this field. Moreover, recent work by Liu et al. (2012) and Skibbe et al. (2012) have shown strong evidence that 3D-SIFT is less suited for such tasks than descriptors where rotation invariance is obtained via group integration techniques. This is mainly the case because of two facts: (1) pose-normalization highly relies on dominate structures within patterns and is also often strongly influenced by noise. Therefore, a unique pose-normalization cannot be guaranteed for each voxel in the image, particularly when considering the fact that for 3D pose-normalization two dominant perpendicular directions must be found. This is contrary to 2D, where it is sufficient and thus much more likely to find one dominant direction, compare Fig 1.5 on page 21 to Fig 1.4 on page 19. (2) Therefore, a pre-classification via interest point detection is used for the determination of stable point candidates. However, a pre-classification of voxels via simple salient points may lead to the loss of indispensable information so that structures or objects are just hidden from further detection or classification steps.

1.4.2. 3D Rotation Invariance via Group Integration

Group integration techniques are based on the idea of integrating over all possible transformation parameters so that a function becomes invariant to that specific kind of transformation. Via group integration we obtain the best approximation of a pattern in a subspace that is invariant to a specific transformation (Schulz-Mirbach 1994); a proof can be found in Reisert (2008, p. 29), that is the orthogonal projection of the pattern onto the invariant subspace. The idea of invariants via group integra-

---

[1] Optical Coherence Tomography

tion goes back to the work of Hurwitz (1897). In general, any feature that transforms with respect to a finite or at least compact local linear transformation group, can be turned into a group-invariant feature via group integration. For instance, the rotation group is a compact linear group. Therefore, we can turn any rotation variant feature into an invariant feature by integrating over all possible rotated version of that feature. In contrast to pose-normalization, group integration always guarantees rotation invariance.

The straight forward way to obtain rotation invariant features is to explicitly sample the orientation space. However, an explicit integration over the whole rotation group via a sparse sampling of the orientation space is far to computational expensive, see Fig. 1.8 on page 27, particularly when computing features in a point-by-point manner. Therefore, explicit sampling of the 3D rotation space is rarely used, see e.g. the work of Schael and Siggelkow (2000) or Ronneberger et al. (2002), where group integration techniques are used to form invariants for the rotation invariant recognition of microscopic images of airborne-pollen.

However, it is known that for any compact linear group there exists an *irreducible* group representation in terms of group specific Fourier basis functions allowing for a systematic computation of invariants in an analytical way. Moreover, it is shown that for these kinds of group invariants (if the expansion coefficients are regular matrices), completeness can be achieved via the Fourier bi-spectrum (Kakarala 1992). Completeness, as we will discuss later in detail in the chapter about invariants, is a very strong restriction that guarantees separability of all patterns up to the transformation.

For instance, for the cyclic translation group, which is a compact group, the irreducible representations are the well known Cartesian Fourier basis functions. Moreover, for discrete n-dimensional signals represented on a discrete Cartesian grid, there exists the Fast Fourier Transform (FFT, Cooley and Tukey (1965)), which transforms signals into their irreducible counterparts in reasonable time. Translation invariants can then easily be obtained via the power- or bi-spectrum.

The biggest challenges within this group integration frameworks are: (1) group integration is a linear operation. Linear operations are very “simple” operations and we risk loosing a lot of information during this process. Therefore, it is very important to create non-linear covariant features before group integration. This is achieved by combining the group integration with kernel techniques (Haasdonk et al. 2004; Haasdonk et al. 2005; Schulz-Mirbach 1995a). This is in opposition to the pose-normalization technique, where features are first turned into a canonical view before (nonlinear) features are extracted. (2) Group integration often requires much more mathematical (and at a first glance computational) effort for creating an invariant feature extraction process. However, thanks to irreducible representations of local linear compact groups, group integration is often possible in reasonable computation time.

The irreducible representations of the rotation group allow for analytically performing a group integration over the continuous space of the rotation parameters. Rotation in the 2D image plane is just a cyclic 1D translation. Therefore, the irreducible representations of 2D rotations are the 1D Fourier basis functions, in this context often called circular harmonics. Any 2D pattern can be represented in terms of polar coordinates (φ, r), where φ is an angle depending parameter and r the distance to the center. That is, ordinary Cartesian Fourier analysis including the 1D Fourier transform can be used within these settings for rotation analysis. For instance, Hsu et al. (1982) realized a fast angular correlation of 2D images based on a 1D FFT using a circular harmonic representation.

There is a broad variety of rotation invariant features for 2D patterns based on the irreducible representation of the 2D rotation. To mention some, Khotanzad and Hong (1990) proposed rotation invariant image descriptors based on the power spectrum of a polar representation of Zernike moments. Sheng and Shen (1994) endorsed Fourier-Mellin moments for the rotation invariant recognition of 2D patterns. The radial parts of Fourier-Mellin moments are polynomials in the radius r^n of degree n. Yap et al. (2010) proposed the polar harmonics for the rotation invariant image representation. Polar harmonics are having a wavy pattern in radial direction. The radial functions are complex valued waves whose frequency can, depending on the application, either be increased in radial direction, which gives the function a Chirplet like character (Mann and Haykin 1991) or be decreased; see also the work on polar harmonics of Hoang (2011); Hoang and Tabbone (2011). A survey on moments and the creation of invariants in 2D can be found in Flusser et al. (2009). Recently, invariants build upon the irreducible representations of the 2D rotation group have been used for the generic detection of structures within microscopic recordings of airborne pollen and fungal spores (Reisert and Burkhardt 2008c; Skibbe and Reisert 2012a) or the detection of motorbikes in images (Liu et al. 2012). In the recent work of Schmidt and Roth (2012), features are not directly represented in terms of circular harmonics. However, a rotation covariant feature representation on a polar grid is transformed into irreducible representations of 2D translations via a Fourier transformation. Compared to 3D rotation invariant features based on pose-normalization, the group integration technique lead to a much broader range of applications in 3D.

Contrarily to 2D, for 3D rotations, the relation to the Cartesian Fourier basis is not obvious. But, there exists an angular dependent expansion of the 3D Cartesian Fourier basis in terms of spherical coordinates (θ, φ, r). That is, instead of a Cartesian parameter vector (x, y, z) we have an angular dependent component (θ, φ) and a distance to the center r. The angular dependent Fourier expansion is a decomposition of the plane wave into angular dependent spherical harmonic functions representing the plane-wave in angular direction (θ, φ). The radial component r is captured by the one-dimensional so-called spherical Bessel functions. Similar to the ordinary Cartesian Fourier basis, spherical harmonics in combination with spherical Bessel functions decompose a function into basic angular and radial frequency patterns (the spherical harmonic functions are the ψ-angle independent irreducible representations of the 3D rotation group). Due to this fact, while in the Cartesian scenario the power spectrum of the Fourier transform of a function is translation invariant, the power spectrum of a spherical harmonic expansion is rotation invariant.

Any volumetric image $I : \mathbb{R}^3 \rightarrow \mathbb{R}$ can be defined in terms of spherical coordinates $I(\theta, \varphi, r)$, that is, instead of a Cartesian parameter vector $I(x, y, z)$ we have an angular dependent component (θ, φ) and a distance to the center r. Hence the angular parametrization of a volumetric image is independent from the Euler angle ψ. Consequently, the spherical harmonics in combination with a 1D function capturing the signal in radial direction form a suitable basis for representing volumetric images with the aim of extracting rotation invariant features. This property of spherical harmonics has been exploited in some recent approaches, which differ mainly in the way the images are represented in radial direction. The spherical representation of the Fourier basis itself is used in Wang et al. (2009) to represent single 3D recordings of airborne-pollen in a rotation invariant manner. An extension to a dense computation of Gabor-like image features has been proposed by Skibbe et al. (2012). Fourier-like functions have also been used in Skibbe et al. (2009a) for describing and classifying volumetric objects. Mak et al. (2008) advocate Zernike polynomials (see also Canterakis (1999)) as suitable radial functions and build rotation invariants for the description and comparison of molecular shapes; see also the works of Novotni and Klein (2004); Venkatraman et al. (2009). In Kazhdan et al. (2003) and Fehr (2009); Fehr (2010); Fehr and Burkhardt (2006); Fehr and Burkhardt (2007); Ronneberger et al. (2005); Skibbe et al. (2009b), a theoretically infinite set of nested spheres, each expanded in spherical harmonics, is used for the detection and/or classification of volumetric biological images. Ma et al. (2012) used spherical harmonic based invariants for the identification of alpha-helices in protein secondary structures. A detailed survey regarding the state-of-the-art of the dense, fast computation of invariants based on such pattern representations can be found in the dissertation of Fehr (2009). Representing the radial component in terms of Laguerre polynomials leads to the spherical representation of the generic neighborhood operators presented in Koenderink and Doorn (1992). Particularly for the latter representation, efficient algorithms allow for densely computing features in a voxel-by-voxel manner in almost linear time (Reisert and Burkhardt 2009a; Skibbe et al. 2012) and have shown remarkable performance in several recent applications including the detection of mitotic cancer cells in microscopic images (Schlachter et al. 2010), the detection of landmarks within images of zebra fish embryos with the aim of the co-registration of protein patterns (Liu et al. 2012; Ronneberger et al. 2012), or the detection of landmarks and the classification of tissue within diffusion MRI data of human brains (Skibbe and Reisert 2011; Skibbe and Reisert 2012b; Skibbe et al. 2011a). Rotation invariant features have also been used by Schnell et al. (2009) for the tissue classification of diffusion MRI images. However, they only fit the angular dependent measurement to spherical harmonics without implementing a radial function. Furthermore, recent works cope with spherical harmonic representations of SIFT like features thus group integration techniques can be used to obtain rotation invariants from gradient orientation distributions. A gradient orientation binning with respect to surface normals of volumetric objects has been used to form invariants for the classification of pollen (Reisert and Burkhardt 2006; Ronneberger 2007; Ronneberger et al. 2007). Furthermore, HOG-like features have been proposed for the detection and classification of objects and landmarks within microscopic images (Liu et al. 2011b; Skibbe et al. 2011b).

1.5. Structure of this Thesis

This thesis is divided into seven chapters, including the introduction and the conclusion. To ensure better readability we shifted some examples, lengthly definitions and proofs, as well as lengthly auxiliary calculations and some figures and plots to the appendix. The main chapters are organized as follows:

Chapter 2 (Harmonic Analysis in 3D) This chapter introduces the mathematical notations, which we use during this thesis. Furthermore, it covers the basic concepts of group representation theory with focus on the 3D rotation group. Furthermore, a survey on spherical tensor algebra is provided. Details regarding a fast tensorial harmonic transformation have been presented at a conference (Skibbe et al. 2009b).

Chapter 3 (Rotation Invariant Features) In this chapter we show how to transform spherical tensor features into locally rotation invariant feature representations via group integration techniques. One specific topic is the creation of phase preserving invariants based on the bi-spectrum (Kakarala 1992). We show that a straight forward computation of the bi-spectrum leads to linear dependencies and we show how to solve this issue leading to an up to three times smaller feature representations and hence a three times faster feature computation. The authors related publications are manifold, covering several applications where rotation invariant features are used for object detection and classification (Skibbe and Reisert 2011; Skibbe and Reisert 2012b; Skibbe et al. 2009a; Skibbe et al. 2009b; Skibbe et al. 2010; Skibbe et al. 2011a; Skibbe et al. 2011b; Skibbe et al. 2012; Skibbe et al. April, 2011).

Chapter 4 (Spherical Tensor Derivatives) This chapter covers the theory and implementation details of spherical tensor derivatives. We put the focus on image processing. Tensor derivatives are playing an indispensable role in extracting rotation covariant features from local voxel neighborhoods. The authors most related publications are Skibbe et al. (2010); Skibbe et al. (2012); Skibbe et al. (April, 2011).

Chapter 5 ($SE(3)$ Covariant Filters) This chapter covers the major contributions of this thesis, including the theoretical background and the implementation details. We introduce three $SE(3)$ covariant feature extraction techniques building the essential covariant filters for mapping volumetric images to covariant feature images based on spherical harmonic transformations:

Section 5.1 (An $SE(3)$ Covariant Gauss-Laguerre Transform) In this section we introduce a computational efficient dense spherical harmonic transformation. The transformation is realized in a voxel-by-voxel manner in almost linear time (apart from one initial convolution). The resulting features are a spherical counterpart of the popular local jet-features (Koenderink and Doorn 1987). We further show that the provided basis functions are optimal for representing local smooth processes. Furthermore, the advocated approach is extended from scalar valued images to higher order tensor valued images. This chapter is based on previous publications of the author (Skibbe and Reisert 2011; Skibbe and Reisert 2012b; Skibbe et al. 2011a; Skibbe et al. 2012).

Section 5.2 (An $SE(3)$ Covariant Spherical Gabor Transform) Gabor wavelet are widely known and used in various applications. We provide a spherical harmonic based Gabor function and show how to efficiently design an

$SE(3)$ covariant Gabor filter in a voxel-by-voxel manner via tensor derivatives. Preliminary results on this topic have been published in Skibbe et al. (2010); Skibbe et al. (2012).

Section 5.3 (SHOG - Spherical Histograms of Oriented Gradients) SHOG, the spherical histogram of oriented gradients, is a tensor representation of SIFT- and HOG-like histograms of oriented gradients. We show how to compute them efficiently via tensor products and show their strong relation to the popular structure tensor. Preliminary results have been published in Skibbe and Reisert (2012a); Skibbe et al. (2011b).

Chapter 6 (Experiments) We conducted three exhaustive experiments on large data sets consisting of synthetically generated and real microscopic data. We compare the performance of the proposed techniques with state-of-the-art approaches like 3D SIFT in a scenario where we aim at $SE(3)$ covariant object and landmark detection.

The appendices are organized as follows:

Appendix A (Biomedical Applications) We introduce two applications: cell detection and cell classification in biological images and a voxel-by-voxel parcellation and landmark detection in diffusion MRI images of human brains. The results have been presented at several conferences and have been published in two journal papers (Skibbe and Reisert 2011; Skibbe and Reisert 2012b; Skibbe and Reisert 2013; Skibbe et al. 2010; Skibbe et al. 2011a; Skibbe et al. 2012).

Appendix B (Auxiliary Calculations) We shifted some lengthly auxiliary calculations to here.

Appendix C (Functions and Formulas) Here we list all relevant definitions of functions and formulas used in this thesis.

Appendix D (Harmonic Filters) A brief introduction on trainable harmonic filters (Reisert and Burkhardt 2009a).

Appendix E (Experiments: Plots and Figures) To ensure better readability, we shifted several figures and plots from the experiment chapter to the end of this thesis.

LIST OF PUBLICATIONS OF THE AUTHOR

Liu, K., H. Skibbe, T. Schmidt, T. Blein, K. Palme, and O. Ronneberger (Aug. 2011b). „3D Rotation-Invariant Description from Tensor Operation on Spherical HOG Field.“ In: *Proc. of the BMVC*. Ed. by Manuel Trucco Jesse Hoey Stephen McKenna and Jianguo Zhang. BMVA. Dundee, UK (cit. on pp. 32, 83, 115, 138, 166, 179).

Reisert, M. and H Skibbe (2012). „Left-Invariant Diffusion on the Motion Group in terms of the Irreducible Representations of SO(3).“ In: *arXiv:submit/0423757 [math.AP]* (cit. on pp. 51, 61, 143).

Ronneberger, O., K. Liu, M. Rath, D. Rueß, T. Mueller, H. Skibbe, B. Drayer, T. Schmidt, A. Filippi, R. Nitschke, T. Brox, H. Burkhardt, and W. Driever (2012). „ViBE-Z: a framework for 3D virtual colocalization analysis in zebrafish larval brains.“ In: *Nature Methods* 9.7, pp. 735–742 (cit. on pp. 15, 32, 90, 226).

Skibbe, H. and M. Reisert (2011). „Dense Rotation Invariant Brain Pyramids for Automated Human Brain Parcellation.“ In: *Proc. of the Informatik 2011, Workshop on Emerging Technologies for Medical Diagnosis and Therapy*. Berlin, Germany (cit. on pp. 32–34, 110, 115, 124, 184, 185).

– (2012a). „Circular Fourier-HOG features for rotation invariant object detection in biomedical images.“ In: *Proc. of the ISBI*. IEEE. Barcelona, Spain, pp. 450–453 (cit. on pp. 31, 34, 138, 225).

– (2012b). „Detection of Unique Point Landmarks in HARDI Images of the Human Brain.“ In: *Proc. of the Workshop on Computational Diffusion MRI (CDMRI’12), part of the MICCAI 2012*. Nice, France (cit. on pp. 32–34, 110, 115, 184, 226, 228).

– (2013). „Rotation Covariant Image Processing for Biomedical Applications.“ In: *Computational and Mathematical Methods in Medicine, Special Issue on Mathematical Methods in Biomedical Imaging* 2013 (cit. on pp. 34, 62, 100, 157, 184, 226).

Skibbe, H., M. Reisert, Q. Wang, O. Ronneberger, and H. Burkhardt (2009a). „Fast Computation of 3D Spherical Fourier Harmonic Descriptors - A Complete Orthonormal Basis for a Rotational Invariant Representation of Three-Dimensional Objects.“ In: *Proc. of the 3DIM, part of the ICCV*. IEEE. Kyoto, Japan (cit. on pp. 32, 33, 61).

Skibbe, H., M. Reisert, O. Ronneberger, and H. Burkhardt (2009b). „Increasing the Dimension of Creativity in Rotation Invariant Feature Design Using 3D Tensorial Harmonics.“ In: *Proc. of the DAGM*. Jena, Germany: LNCS, Springer, pp. 141–150 (cit. on pp. 32, 33, 75, 77, 79, 83, 200).

Skibbe, H., M. Reisert, T. Schmidt, K. Palme, O. Ronneberger, and H. Burkhardt (2010). „3D Object Detection using a Fast Voxel-Wise Local Spherical Fourier Tensor Transformation.“ In: *Proc. of the DAGM*. Darmstadt, Germany: LNCS, Springer, pp. 412–421 (cit. on pp. 33, 34, 83, 130, 133, 181, 184).

Skibbe, H., M. Reisert, and H. Burkhardt (2011a). „Gaussian Neighborhood Descriptors for Brain Segmentation.“ In: *Proc. of the MVA*. IAPR. Nara, Japan (cit. on pp. 32–34, 110, 115, 124, 184, 185).

– (2011b). „SHOG - Spherical HOG Descriptors for Rotation Invariant 3D Object Detection.“ In: *Proc. of the DAGM*. Frankfurt, Germany: LNCS, Springer, pp. 142–151 (cit. on pp. 32–34, 138, 154, 160, 166, 179, 226).

Skibbe, H., M. Reisert, T. Schmidt, T. Brox, O. Ronneberger, and H. Burkhardt (2012). „Fast Rotation Invariant 3D Feature Computation utilizing Efficient Local Neighborhood Operators.“ In: *IEEE Trans. Pattern Anal. Mach. Intell.* (Cit. on pp. 29, 32–34, 100, 110, 130, 150, 157, 171, 181, 184).

Skibbe, H., M. Reisert, O. Ronneberger, and H. Burkhardt (April, 2011). „Spherical Bessel Filter for 3D Object Detection.“ In: *Proc. of the ISBI*. IEEE. Chicago,Illinois, USA (cit. on pp. 33, 130, 160, 226).

2 Harmonic Analysis in 3D

2.1. Preliminaries

We write vectors $\mathbf{v} \in \mathbb{C}^n$ and matrices $\mathbf{M} \in \mathbb{C}^{n \times m}$ in bold letters. Matrices are always written in capital letters. Single components are denoted by indexes v_i so that $\mathbf{v} = (v_1, \cdots, v_n)^T$. Accordingly, we have two indexes for single matrix elements, that is, for $\mathbf{M} \in \mathbb{C}^{m \times n}$: $M_{i,j} \in \mathbb{C}$, $1 \leq i \leq m, 1 \leq j \leq n$. For three dimensional vectors we sometimes use the notation $\mathbf{v} = (v_x, v_y, v_z)^T$, or just $\mathbf{x} = (x, y, z)^T$, to emphasize the relation to the Cartesian coordinate system in 3D. If additional lower indexes are associated with a vector or matrix (e.g. $\mathbf{v}_m \in \mathbb{C}^n$) we denote single components by using square brackets, e.g. $\mathbf{v}_m = ([\mathbf{v}_m]_1, \cdots, [\mathbf{v}_m]_n)^T$. For single vectors we use the letters $\mathbf{u}, \mathbf{v}, \mathbf{w}$. If the vector denotes a cordinate vector we use $\mathbf{r}, \mathbf{t}, \mathbf{x}$. Scalar valued functions are written in small letters, e.g. $f : \mathbb{R}^n \to \mathbb{C}$. If f is a volumetric image we often write $I : \mathbb{R}^3 \to \mathbb{C}$ to facilitate interpretation. Vector valued functions are written in bold small letters, e.g. $\mathbf{f} : \mathbb{R}^n \to \mathbb{C}^\ell$. For vector valued functions we use a vector representation, thus $\mathbf{f}$ can be written as vector $\mathbf{f} = (f_1, \cdots, f_\ell)^T$, where $f_i : \mathbb{R}^n \to \mathbb{C}$. Moreover, if additional lower indexes are associated with a function (e.g. $\mathbf{f}_m : \mathbb{R}^n \to \mathbb{C}^\ell$), then we write $\mathbf{f} = ([\mathbf{f}_m]_1, \cdots, [\mathbf{f}_m]_\ell)^T$. If the elements of a vector valued function can be regarded as some kind of expansion coefficients or covariant features, then we use the letters $\mathbf{a}, \mathbf{b}, \mathbf{c}$ or $\mathbf{d}$. We denote the complex conjugate of a vector, matrix or a function by $\overline{\mathbf{v}}$, $\overline{\mathbf{M}}$, $\overline{f}$ and $\overline{\mathbf{f}}$, respectively. The transpose of a vector or matrix is written as $\mathbf{v}^T$ and $\mathbf{M}^T$. $\mathbf{M}^*$ is the adjoint of the matrix $\mathbf{M}$ with $\mathbf{M}^* = \overline{\mathbf{M}}^T$. We denote the imaginary unit by i. Let $a = b + ic$, $b, c \in \mathbb{R}$ be a complex number. Then $\mathfrak{Re}(a) = b$ and $\mathfrak{Im}(a) = c$. Depending on the context we express the Cartesian coordinate

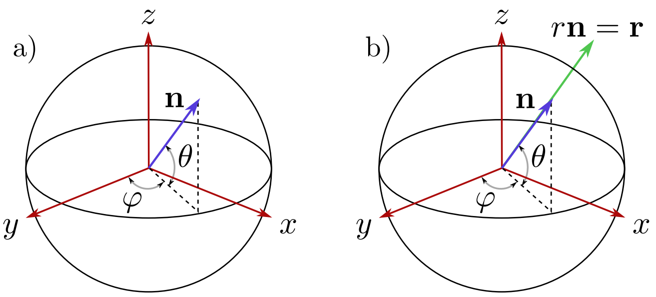


Figure 2.1.: Spherical and Cartesian representation of points in a) S_2 and b) $\mathbb{R}^3$

vector $\mathbf{r} = (x,y,z)^T$, $\mathbf{r} \in \mathbb{R}^3$ in spherical coordinates (θ,φ,r), where $\theta = arcos(z/||r||)$, $\varphi = atan2(y,x)$ and $r = ||\mathbf{r}||$; see Fig. 2.1. We represent points on the 2-sphere (S_2, the sphere embedded in $\mathbb{R}^3$ with radius 1) in terms of the tuple $(\theta,\varphi) \in S_2$. or, equivalently use the unit vector $\mathbf{n} \in \mathbb{R}^3$, $||\mathbf{n}|| = 1$, $\mathbf{n} = (n_x,n_y,n_z)^T$, where $\theta = arcos(n_z)$, $\varphi = atan2(n_y,n_x)$.

$L_2(S,\mu)$ is the Hilbert space of the quadratic integrable functions $f : S \to \mathbb{C}$ defined on the domain S with measure $\mu : S \to \mathbb{R}_{\geq 0}$. If we restrict on $f : S \to \mathbb{R}$, then we write $L_2(S,\mathbb{R},\mu)$. Square integrable means that the integral of the square of the absolute value of a function $f \in L_2(S,\mu)$ is finite, that is

$$\int_S |f(x)|^2 d\mu(x) < \infty \quad . \tag{2.1}$$

This is a very important property, because it guarantees that the inner product between images is finite.

Remember that a Hilbert space H is an inner product space with the following properties: there exists a Hermitian form that is called the inner product. We denote the inner product by $\langle\cdot,\cdot\rangle : H \times H \to \mathbb{C}$. The inner product induces the norm $||\cdot|| := \sqrt{\langle\cdot,\cdot\rangle}$. Moreover, the distance between two points in this space is defined by the norm, where for $x,y \in H : d(x,y) = ||x - y||$. Furthermore, a Hilbert space is complete, that is, all Cauchy sequence converge with respect to the norm of H; see section C.2 in the appendix for details concerning Hilbert spaces. Further information about vector spaces and Hilbert spaces can be found in almost any book covering the basics in analysis.

Let $f,g \in L_2(S,\mu)$, then the inner product is defined by

$$\langle f,g\rangle_\mu := \int_S f(x)\overline{g(x)}d\mu(x) \, . \tag{2.2}$$

We use the subscript μ as a reminder indicating that we must take special care of the measure (in such cases the measurement differs from a constant). In this work it only occurs when working with polynomials which are orthogonal with respect to a Gaussian function. In all other scenarios we shortly write $L_2(S)$ and neglect the subscript writing $\langle f,g\rangle$ instead of $\langle f,g\rangle_\mu$. This inner product induces the norm and the distance measure on $L_2(S,\mu)$.

Example 2.1.1 ($L_2(\mathbf{R}^3)$, the Space of Volumetric Images)**.** *This is the basic Hilbert space we use to represent volumetric images. We often use the letter I for denoting images, that is* $I \in L_2(\mathbf{R}^3)$. *The inner product is the standard inner product of two images, where for* $f : \mathbb{R}^3 \to \mathbb{C}$ *and* $g : \mathbb{R}^3 \to \mathbb{C}$

$$\langle f,g\rangle = \int_{\mathbb{R}^3} f(\mathbf{r})\overline{g(\mathbf{r})}d\mathbf{r} \, . \tag{2.3}$$

We equivalently make use of spherical coordinates using the measure $\mu(\theta,\varphi,r) = \sin(\theta)d\theta d\varphi r^2 dr$. *In this case the inner product becomes*

$$\langle f,g\rangle = \int_{\theta=0}^{\pi}\int_{\varphi=0}^{2\pi}\int_{r=0}^{\infty} f(\theta,\varphi,r)\overline{g(\theta,\varphi,r)}\sin(\theta)d\theta d\varphi r^2 dr \, . \tag{2.4}$$

Note that we freely switch between both representations. The norm and the distance are then defined in terms of this inner product.

Example 2.1.2 ($L_2(S_2)$, the Space of Functions on the Sphere)**.** *Let* $f : S_2 \rightarrow \mathbb{C}$ *and* $g : S_2 \rightarrow \mathbb{C}$ *be two square integrable functions on the sphere (*S_2 *denotes the unit-sphere embedded in* $\mathbb{R}^3$*). The inner product is then*

$$\begin{aligned}\langle f,g\rangle &= \int_{S_2} f(\mathbf{n})\overline{g(\mathbf{n})}d\mathbf{n} \qquad ,(\textit{where } \mathbf{n}\in\mathbb{R}^3, \|\mathbf{n}\|=1)\\ &= \int_{\theta=0}^{\pi}\int_{\varphi=0}^{2\pi} f(\theta,\varphi)\overline{g(\theta,\varphi)}\sin(\theta)d\theta d\varphi\ . \end{aligned} \tag{2.5}$$

We use both representations in this thesis. The norm and the distance are again defined in terms of this inner product.

Definition 2.1.3 (Translation of n-dimensional Functions)**.** *We denote the translation group by* $T(n)$*, where* $T(n) = SE(n)/SO(n)$ *is the Euclidean motion group of order n without rotations. We parametrize a translation by a translation vector* $\mathbf{t} = (t_1,\cdots,t_n)^T \in \mathbb{R}^n$*, where the* t_i *are representing the translation into direction i. We denote the elements of* $T(n)$ *by* $\mathfrak{h}_\mathbf{t}$*. Given a function* $f \in L_2(\mathbb{R}^n)$*. We define the action of* $\mathfrak{h}_\mathbf{t}$ *on an image in terms of the coordinate transform*

$$(\mathfrak{h}_\mathbf{t} f)(\mathbf{r}) := f(\mathbf{r}-\mathbf{t}) \quad . \tag{2.6}$$

See the example 2.1.14 on page 43 for further details.

Definition 2.1.4 (The Fourier Transformation)**.** *Let* $f \in L_2(\mathbb{R}^n)$*. We denote by* $\mathcal{FT} : L_2(\mathbb{R}^n) \rightarrow L_2(\mathbb{R}^n)$ *the n-dimensional Fourier transform acting on images* $f : \mathbb{R}^n \rightarrow \mathbb{C}$*. We further denote by* $\widetilde{f} = \mathcal{FT}\{f\}$ *the Fourier representation of* f*. The Fourier transform is an invertible mapping that maps functions into the space spanned by the orthogonal Fourier basis functions* $\{\omega(\mathbf{k})\}$*,* $\omega(\mathbf{r},\mathbf{k}) := e^{i\mathbf{k}^T\mathbf{r}}$*. The functions* $\omega(\mathbf{k}) : \mathbb{R}^n \rightarrow \mathbb{C}$ *define a plane wave. We call the vector* $\mathbf{k} \in \mathbb{R}^n$ *the wave vector, where* $\mathbf{k} = (k_1,\cdots,k_n)^T$*. The wave vector defines the frequency* $k = \|\mathbf{k}\|$ *and the direction* $\mathbf{n} = \frac{\mathbf{k}}{\|\mathbf{k}\|}$ *of the plane wave in* $\mathbf{R}^n$*. The vector* $\mathbf{r} \in \mathbb{R}^n$ *is the position (or time) vector, which is a position dependent parameter. The forward transformation* $\mathcal{FT}$ *is defined in a frequency by frequency manner:*

$$\mathcal{FT}\{f\}(\mathbf{k}) := (2\pi)^{-n/2}\langle f,\omega(\mathbf{k})\rangle = (2\pi)^{-n/2}\int_{\mathbb{R}^n} f(\mathbf{r})e^{-i\mathbf{k}^T\mathbf{r}}d\mathbf{r} = \widetilde{f}(\mathbf{k}) \quad . \tag{2.7}$$

The inverse is defined by

$$\mathcal{FT}^{-1}\{\widetilde{f}\}(\mathbf{r}) := (2\pi)^{-n/2}\langle \widetilde{f},\overline{\omega(\mathbf{r})}\rangle = (2\pi)^{-n/2}\int_{\mathbb{R}^n} f(\mathbf{k})e^{i\mathbf{k}^T\mathbf{r}}d\mathbf{k} = f(\mathbf{r}) \quad . \tag{2.8}$$

The Fourier transform is unitary and thus preserves the energy (Parseval's theorem, $\langle f,g\rangle = \langle \widetilde{f},\widetilde{g}\rangle$ *with the conclusion that* $\|f\|^2 = \|\widetilde{f}\|^2$*, see e.g. Rudin (2006)).*

Definition 2.1.5 (The Convolution). *The convolution is a commutative, linear operation. Let* $f, g \in L_2(\mathbb{R}^n)$. *Then the convolution is defined by*

$$\begin{aligned}(f * g)(\mathbf{x}) &:= \int_{\mathbb{R}^n} f(\mathbf{r}) g(\mathbf{x} - \mathbf{r}) d\mathbf{r} \\ &= \int_{\mathbb{R}^n} f(\mathbf{x} - \mathbf{r}) g(\mathbf{r}) d\mathbf{r}\,. \end{aligned} \tag{2.9}$$

The convolution has several important properties. Let $h \in L_2(\mathbb{R}^n), \alpha \in \mathbb{C}$. *Then*

- $(f * g) = (g * f)$ *(commutativity)*
- $(f * (g * h)) = ((g * g) * h)$ *(associativity)*
  $\alpha(f * g) = (\alpha f) * g = f * (\alpha g)$
- $(f * (g + h)) = (f * g) + (f * h)$ *(distributivity)*
- $(f * \delta^n_{\mathbf{0}}) = f$, *where* $\delta^n_{\mathbf{x}} : \mathbb{R}^n \to \mathbb{R}$ *is the Dirac delta function* $\delta^n_{\mathbf{x}}(\mathbf{r}) = \delta(x_1 - r_1) \cdots \delta(x_n - r_n)$ *and* $\int_{\mathbb{R}^n} \delta^n_{\mathbf{x}}(\mathbf{r}) d\mathbf{r} = 1$. *(multiplicative identity)*
- $\overline{(f * g)} = (\overline{f} * \overline{g})$ *(complex conjugation)*
- $\frac{\partial}{\partial x_i}(f * g) = (\frac{\partial f}{\partial x_i} * g) = (f * \frac{\partial g}{\partial x_i})$ *(differentiation)*
- $\mathcal{FT}\{f * g\} = \mathcal{FT}\{f\}\mathcal{FT}\{g\}$, *(the convolution theorem)*
- $\tau_{\mathbf{x}}(f * g) = ((\tau_{\mathbf{x}} f) * g) = (f * (\tau_{\mathbf{x}} g))$, *(translation invariance)*

The Fourier domain and the spatial/time domain are connected in the following sense: operations that act globally in spatial domain are simple pointwise operations in Fourier domain and vise versa. Switching between these two worlds often offers the only way to cope with the proposed techniques in a convenient manner. Particularly the convolution theorem plays an important role in practice. The convolution is a complex operation. Thanks to efficient algorithms realizing the Fourier transform for discrete images in reasonable time (The Fast Fourier Transform, shortly FFT, Cooley and Tukey (1965)), convolution is practical for large images (see particularly the implementation of Frigo and Johnson (2005)). Fourier correspondences used during this thesis are summarized in the following.

Corollary 2.1.6 (Properties of the Fourier Transform in 3D). *We denote a pair of spatial representation and Fourier representation by* $\circ\!\!-\!\!\!-\!\!\bullet$. *Hence for a function* f *and its Fourier representation* $\widetilde{f} = \mathcal{FT}\{f\}$ *we write* $f \circ\!\!-\!\!\!-\!\!\bullet \widetilde{f}$. *The following correspondences are used during this work:*

$$\begin{array}{rcll} f & \circ\!\!-\!\!\!-\!\!\bullet & \widetilde{f} & \textit{(Fourier Trafo)} \\ w(k') & \circ\!\!-\!\!\!-\!\!\bullet & (2\pi)^{3/2}\delta(k' - k) & \textit{(Fourier Trafo)} \\ \frac{\partial}{\partial x_i} f & \circ\!\!-\!\!\!-\!\!\bullet & ik_i \widetilde{f} & \textit{(Differentiation)} \\ \Delta f & \circ\!\!-\!\!\!-\!\!\bullet & -k^2 \widetilde{f} & \textit{(Laplace operator)} \\ (f * g) & \circ\!\!-\!\!\!-\!\!\bullet & \widetilde{f}\widetilde{g} & \textit{(Convolution)} \end{array}$$

That is, partial differentiation corresponds to a point-by-point multiplication in Fourier domain with the frequency vector projected onto the direction of differentiation. The Laplace operation depends on the magnitude of the frequency, and the convolution is simply a point-by-point multiplication.

2.1.1. Groups and Group Representations

In this section we introduce the concept of groups and group representations. A more detailed introduction can be found e.g. in the lecture notes of Miller (1991).

Definition 2.1.7 (Group). *We call a set of objects* $X = \{\mathfrak{g}_0, \mathfrak{g}_1, \mathfrak{g}_2, \mathfrak{g}_3, \cdots\}$ *together with a binary operation* $\cdot$ *a* ***group*** $G = (X, \cdot)$, *if the binary operation (also called the group-multiplication) fulfills the following requirements:*

1. ***Closure*** $\forall \mathfrak{g}_i, \mathfrak{g}_j \in G : (\mathfrak{g}_i \cdot \mathfrak{g}_j) \in G$ *(or shortly* $\mathfrak{g}_i \mathfrak{g}_j \in G$*)*
2. ***Associativity*** $\forall \mathfrak{g}_i, \mathfrak{g}_j, \mathfrak{g}_k \in G : (\mathfrak{g}_i \cdot \mathfrak{g}_j) \cdot \mathfrak{g}_k = \mathfrak{g}_i \cdot (\mathfrak{g}_j \cdot \mathfrak{g}_k)$ *(or shortly* $(\mathfrak{g}_i \mathfrak{g}_j)\mathfrak{g}_k = \mathfrak{g}_i(\mathfrak{g}_j \mathfrak{g}_k)$*)*
3. ***Identity Element*** *There exists the (unique) identity element* $\mathfrak{e}$, *such that for all* $\mathfrak{g} \in G$ *the equation* $\mathfrak{g} \cdot \mathfrak{e} = \mathfrak{e} \cdot \mathfrak{g} = \mathfrak{g}$ *holds.*
4. ***Inverse element*** $\forall \mathfrak{g} \in G, \exists \mathfrak{g}^{-1} \in G : \mathfrak{g} \cdot \mathfrak{g}^{-1} = \mathfrak{g}^{-1} \cdot \mathfrak{g} = \mathfrak{e}$

Note that it is not necessarily true that $\forall \mathfrak{g}_i, \mathfrak{g}_j \in G : \mathfrak{g}_i \mathfrak{g}_j = \mathfrak{g}_j \mathfrak{g}_i$ ($\mathfrak{g}_i$ and $\mathfrak{g}_j$ commute). In the case where $\mathfrak{g}_i$ and $\mathfrak{g}_j$ commute the group is called an *abelian* group.

Definition 2.1.8 (Subgroup). *A subset* $Y \subseteq X$ *of a group* $G = (X, \cdot)$ *that forms itself a group* H *under the group multiplication of* G *such that* $H = (Y, \cdot)$ *is called a* ***subgroup*** *of* G. *The (trivial) subgroups* $(\mathfrak{e}, \cdot)$ *and* $(X, \cdot)$ *are called* ***improper*** *subgroups, all other subgroups are called* ***proper*** *subgroups.*

Definition 2.1.9 (Group Homomorphism). *Given two groups* $G = (X, \cdot)$ *and* $H = (Y, *)$. *A* ***group homomorphism*** *from* G *to* H *is a function* $\boldsymbol{\tau} : G \to H$ *such that*

$$\forall \mathfrak{g}_i, \mathfrak{g}_j \in G : \underbrace{\boldsymbol{\tau}(\mathfrak{g}_i \cdot \mathfrak{g}_j)}_{\text{mult. in } G} = \underbrace{\boldsymbol{\tau}(\mathfrak{g}_i) * \boldsymbol{\tau}(\mathfrak{g}_j)}_{\text{mult. in } H} \quad . \tag{2.10}$$

If the homomorphism is bijective, it is called a group isomorphism.

The groups playing a key role in this theses are listed below.

- With $GL(V)$ we denote the **general linear** group of invertible linear maps acting on a vector space V.
- With $GL(\mathbb{R}^n)$ we denote the **real general linear** group of the non-singular (invertible) $n \times n$ matrices acting on the vector space $\mathbb{R}^n$ (the representations of invertible linear functions acting on the finite dimensional vector space $\mathbb{R}^n$).
- $GL(\mathbb{C}^n)$ is the **complex general linear** group of the non-singular (invertible) $n \times n$ matrices acting on the vector space $\mathbb{C}^n$.

- The group $SO(n)$ represents the **special orthogonal** groups of dimension n, where

  $$SO(n) := \{A \in GL(\mathbb{R}^n) : AA^T = \mathcal{I} \wedge \det A = 1\}. \tag{2.11}$$

  These groups are known as rotation groups, where particularly **the 3D rotation group** $SO(3)$ plays a major role in our work.

- The complex valued counterparts of $SO(n)$ are the **special unitary** groups

  $$SU(n) := \{A \in GL(\mathbb{C}^n) : AA^* = \mathcal{I} \wedge \det A = 1\} . \tag{2.12}$$

Note that $GL(\mathbb{R}^n)$ is a subgroup of $GL(\mathbb{C}^n)$, $SU(n)$ a subgroup of $GL(\mathbb{C}^n)$ and $SO(n)$ is a subgroup of $SU(n)$.

Definition 2.1.10 (The Haar[1] Integral on $SO(n)$ and $SU(n)$). *For every compact linear Lie group G (like $SO(n)$ and $SU(n)$ (see Def. C.2.6 on page 213) there exists an (left and right) invariant integral such that for every integrable complex function $f \in L_2(G)$ and for every $\mathfrak{h} \in G$,*

$$\int_G f(\mathfrak{h}\mathfrak{g})d\mathfrak{g} = \int_G f(\mathfrak{g}\mathfrak{h})d\mathfrak{g} = \int_G f(\mathfrak{g})d\mathfrak{g} \quad . \tag{2.13}$$

This integral converges for any bounded function; see Nachbin (1965).

A proof of definition 2.1.10 can be found in the lecture notes by Miller (1991), chapter 3.3 on page 42.

Definition 2.1.11 (Group Representation). *A representation of a group G is a homomorphism $\boldsymbol{\tau} : G \to GL(V)$ into the general linear group of invertible mappings acting on a vector space V.*

The dimension of the representation is the dimension of the corresponding vector space, thus $\dim\boldsymbol{\tau} = \dim V$. In the following scenarios we almost exclusively deal with representations of finite dimension, that is, mappings acting on the finite-dimensional vector space $\mathbb{C}^n$ or $\mathbb{R}^n$. Hence we consider a representation as finite dimensional unless it is explicitly stated contrarily.

Definition 2.1.12 (Finite Dimensional Group Representation). *An n-dimensional **matrix representation** is a homomorphism $\boldsymbol{\tau} : G \to GL(\mathbb{C}^n)$ that maps elements of the group G into elements of the general linear group $GL(\mathbb{C}^n)$ (the group of the non-singular (invertible) $n \times n$ matrices) acting on the **representation space** $\mathbb{C}^n$. (In case of $\boldsymbol{\tau} : G \to GL(\mathbb{R}^n)$ we call it a **real matrix representation**). Since $\boldsymbol{\tau}$ is a group homomorphism (Def. 2.1.9 in the appendix) we can conclude that for $\mathfrak{g}_1, \mathfrak{g}_2 \in G$:*

$$\boldsymbol{\tau}(\mathfrak{g}_1)\boldsymbol{\tau}(\mathfrak{g}_2) = \boldsymbol{\tau}(\mathfrak{g}_1\mathfrak{g}_2), \qquad \boldsymbol{\tau}({\mathfrak{g}_1}^{-1}) = \boldsymbol{\tau}(\mathfrak{g}_1)^{-1}, \qquad \textit{and} \qquad \boldsymbol{\tau}(\mathfrak{e}) = \mathcal{I} \quad . \tag{2.14}$$

---

[1] based on theoretical work of Alfréd Haar, a Hungarian mathematician (1885-1933)

We shortly write $\boldsymbol{\tau}_{\mathfrak{g}}$ instead of $\boldsymbol{\tau}(\mathfrak{g})$. We further denote the matrix elements of $\boldsymbol{\tau}_{\mathfrak{g}}$ by $[\boldsymbol{\tau}_{\mathfrak{g}}]_{mn}$.

Suppose there is a matrix representation $\boldsymbol{\tau}$ of a group G mapping to matrices acting on the vector space $\mathbb{C}^n$. Then each change of the basis of $\mathbb{C}^n$ induces a new (we call it an equivalent) matrix representation $\boldsymbol{\tau}' : G \to GL(\mathbb{C}^n)$: let the rows of $\mathbf{B} \in GL(\mathbb{C}^n)$ be a basis of $\mathbb{C}^n$ ($\mathbf{B} = [\mathbf{e}_0, \cdots, \mathbf{e}_n]^T$, $\mathbf{e}_i \in \mathbb{C}^n$ with $\langle \mathbf{e}_i, \mathbf{e}_j \rangle = \delta_{i,j}$). Suppose a basis change is given by a matrix $\mathbf{S} \in GL(\mathbb{C}^n)$ such that the new basis is given by $\mathbf{B}' = \mathbf{SB}$. Then we can identify equivalent transformations $\boldsymbol{\tau}'_{\mathfrak{g}}$ by

$$\mathbf{S}\boldsymbol{\tau}_{\mathfrak{g}}\mathbf{B} = \underbrace{\mathbf{S}\boldsymbol{\tau}_{\mathfrak{g}}\mathbf{S}^{-1}}_{=\boldsymbol{\tau}'_{\mathfrak{g}}}\underbrace{\mathbf{SB}}_{=\mathbf{B}'} = \boldsymbol{\tau}'_{\mathfrak{g}}\mathbf{B}' \quad . \tag{2.15}$$

Definition 2.1.13. *Two n-dimensional (real/complex) matrix representations are* ***equivalent*** *($\boldsymbol{\tau} \approx \boldsymbol{\tau}'$) if there exists a matrix $\mathbf{S} \in GL(\mathbb{C}^n)$ with the result that*

$$\boldsymbol{\tau}'_{\mathfrak{g}} = \mathbf{S}\boldsymbol{\tau}_{\mathfrak{g}}\mathbf{S}^{-1} \quad . \tag{2.16}$$

Each $\boldsymbol{\tau}$ induces an equivalence class $\{\boldsymbol{\tau}' | \boldsymbol{\tau} \approx \boldsymbol{\tau}'\}$.

Example 2.1.14 (The 3D Translation Group $T(3)$)**.** *We denote the 3D translation group by $T(3)$. The 3D translation group is the 3D Euclidean motion group without rotations, that is $T(3) = SE(3)/SO(3)$. A translation acting on 3D images $I \in L_2(\mathbb{R}^3)$ is commonly parametrized by a 3D coordinate vector $\mathbf{t} = (t_x, t_y, t_z)^T$ that defines the Cartesian translation in x,y and z direction. The natural representations of the 3D translation group are the matrices*

$$\overset{\leftrightarrows}{T}_{\mathbf{t}} = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} , \tag{2.17}$$

acting on the vector space $\mathbb{R}^3$. Note that we use homogeneous coordinates here. The identity element is $\overset{\leftrightarrows}{T}_{\mathbf{0}}$ and the inverse of $\overset{\leftrightarrows}{T}_{\mathbf{t}}$ is just $\overset{\leftrightarrows}{T}_{(-\mathbf{t})}$. During this thesis we always associate the elements of the 3D translation group acting on images with a translation vector $\mathbf{t} = (t_x, t_y, t_z)^T$. The group elements $\mathfrak{h}_{\mathbf{t}} \in T(3)$ are acting on images in the following way (the following definitions are equivalent) :

$$\begin{aligned} (\mathfrak{h}_{\mathbf{t}} I)(\mathbf{x}) &= (I * \delta_{\mathbf{t}})(\mathbf{x}) \\ &= I((\overset{\leftrightarrows}{T}_{\mathbf{t}})^{-1}\mathbf{x}) \\ &= I(\mathbf{x} - \mathbf{t}) \quad . \end{aligned} \tag{2.18}$$

The identity is $\mathfrak{h}_{\mathbf{0}}$ and the inverse of $\mathfrak{h}_{\mathbf{t}}$ is $\mathfrak{h}_{(-\mathbf{t})}$.

Furthermore, in case of a compact group G, such as $SO(n)$ and $SU(n)$ (see Def. C.2.6 on page 213), there always exists for each n-dimensional matrix representations $\boldsymbol{\tau}$ an equivalent **unitary** matrix representation $\boldsymbol{\tau}'$ with $\boldsymbol{\tau}' \approx \boldsymbol{\tau}$, such that

$$\forall \mathfrak{g} \in G : \boldsymbol{\tau}'_{\mathfrak{g}}(\boldsymbol{\tau}'_{\mathfrak{g}})^* = \mathcal{I} \quad . \tag{2.19}$$

(A proof can be found in (Miller 1991), page 16, below Theorem 3.1). As a consequence it is sufficient to consider only the class of unitary representations $\boldsymbol{\tau} : G \rightarrow SU(n)$ (and their real valued counterparts $\boldsymbol{\tau} : G \rightarrow SO(n)$) as representatives of their equivalence classes.

2.1.1.1. Irreducible Representations

In the previous section we have mentioned that for compact groups like $SO(n)$ and $SU(n)$ it is sufficient to consider only the unitary matrix representations. This is a very powerful property. In the following we introduce a certain family of (unitary) matrix representations, the irreducible representations. We obtain irreducible representations by decomposing a unitary matrix representation into a direct sum of lower dimensional matrix representations i.e. an equivalent matrix representation having block diagonal form. Irreducible means, that there is no further decomposition possible (maximum sparse block diagonal matrix).

Before considering irreducibility more closely, we first must define invariant subspaces.

Definition 2.1.15 (Invariant Subspace)**.** *Given an n-dimensional matrix representation* $\boldsymbol{\tau} : G \rightarrow GL(\mathbf{C}^n)$*. We call a subspace* $U \subset \mathbb{C}^n$ ***invariant*** *under* $\boldsymbol{\tau}$ *if* $\forall \mathbf{u} \in U, \mathfrak{g} \in G : \boldsymbol{\tau}_{\mathfrak{g}} \mathbf{u} \in U$.

Definition 2.1.16 (Irreducible Subspaces)**.** *An n-dimensional matrix representation* $\boldsymbol{\tau} : G \rightarrow GL(\mathbb{C}^n)$ *is called* ***reducible****, if there exists a non-empty (proper) subspace* $U \subset \mathbb{C}^n$ *which is invariant under* $\boldsymbol{\tau}$*. Otherwise, we call* $\boldsymbol{\tau}$ ***irreducible****.*

If $\boldsymbol{\tau}$ is a unitary n-dimensional matrix representation $\boldsymbol{\tau} : G \rightarrow GL(\mathbb{C}^n)$, then we can decompose $\boldsymbol{\tau}$ into irreducible representations $\boldsymbol{\tau}^{\ell=\{0,1,\cdots\}}$ in an (almost) unique manner as explained in the following.

Let W be a subspace of an inner product space V. The orthogonal complement $W^\perp$ of W is the subspace of V, where all vectors are orthogonal to every vectors in W: $W^\perp := \{\mathbf{v} \in V \mid \forall \mathbf{w} \in W : \langle \mathbf{w}, \mathbf{v} \rangle = 0\}$. As a consequence we can uniquely write each element $\mathbf{v} \in V$ as sum of two vectors of $\mathbf{w} \in W$ and $\mathbf{w}' \in W^\perp$, namely $\mathbf{v} = \mathbf{w} + \mathbf{w}'$. We say V is the direct sum of W and $W^\perp$, denoted by $V = W \oplus W^\perp$.

Theorem 2.1.17. *Let* $\boldsymbol{\tau} : G \rightarrow GL(\mathbb{C}^n)$ *be a reducible representation of G. Let W be a proper invariant subspace of* $\mathbb{C}^n$ *under* $\boldsymbol{\tau}$*. Then* $W^\perp$ *is also a proper invariant subspace under* $\boldsymbol{\tau}$ *and we say,* $\boldsymbol{\tau}^W$ *and* $\boldsymbol{\tau}^{W^\perp}$ *are the unitary* ***restrictions*** *of* $\boldsymbol{\tau}$ *on W and* $W^\perp$*, respectively. According to* $\mathbb{C}^n = W \oplus W^\perp$ *we write* $\boldsymbol{\tau} = \boldsymbol{\tau}^W \oplus \boldsymbol{\tau}^{W^\perp}$ *(*$\boldsymbol{\tau}$ *is the direct sum of* $\boldsymbol{\tau}$ *and* $\boldsymbol{\tau}^W$*). (A proof can be found in (Miller 1991), page 18, below Theorem 3.2)*

Using theorem 2.1.17 we can decompose a unitary n-dimensional reducible matrix representation $\boldsymbol{\tau} : G \rightarrow GL(\mathbb{C}^n)$ into irreducible representations $\boldsymbol{\tau}^{\ell=\{0,1,\cdots\}}$ in the following way:

Algorithm 1 n-dimensional matrix representation → irreducible representations

```
**input:** $\boldsymbol{\tau} : G \to GL(\mathbb{C}^n)$
$V := \mathbb{C}^n$
$\ell := 0$
**while** $V \neq \emptyset$ **do**
    let $W_\ell \subset V$ be the smallest, proper subspace of $V$, which is invariant under $\boldsymbol{\tau}$
    $\boldsymbol{\tau}^{(\ell)} = \boldsymbol{\tau}^{W_\ell}$
    $V := W_\ell^\perp$
    $\ell := \ell + 1$
**end while**
**return** $\boldsymbol{\tau}^{(n)} : G \to GL(W_n)$, $n = \{0, \cdots, (\ell - 1)\}$ with $\boldsymbol{\tau} = \bigoplus_n \boldsymbol{\tau}^{(n)}$ and $V = \bigoplus_n W_n$
```

We denote by $\bigoplus_{n=0}^N$ the direct sum

$$V = \bigoplus_{n=0}^{N} W_n = W_0 \oplus \cdots \oplus W_N \quad . \tag{2.20}$$

The subspaces W_ℓ are not uniquely determined. But the decomposition is "almost unique": for two decompositions $\boldsymbol{\tau} = \bigoplus_{n=0}^N \boldsymbol{\tau}^{(n)}$ and $\boldsymbol{\tau} = \bigoplus_{m=0}^M \boldsymbol{\tau}'^{(m)}$ we end up with the same number of irreducible representations, hence $M = N$. Furthermore, we can find a bijective mapping $\pi : \{0, \cdots N\} \to \{0, \cdots M\}$ thus $\forall n \in \{0, \cdots N\} : \boldsymbol{\tau}^{(n)} \approx \boldsymbol{\tau}'^{(\pi(n))}$. Accordingly the equivalence classes of irreducible representations are unique. We can **uniquely** index the equivalence classes in an increasing order and choose one representative out of each. As a result we can consider the decomposition

$$\boldsymbol{\tau} = \bigoplus_{\ell} \boldsymbol{\tau}^{\ell}, \quad \boldsymbol{\tau}^{\ell} \text{ is irreducible,} \tag{2.21}$$

as unique with respect to the index ℓ.

A very important fact is, that two unitary irreducible representations $\boldsymbol{\tau}^\ell : G \to GL(\mathbb{C}^{n_\ell})$ and $\boldsymbol{\tau}^{\ell'} : G \to GL(\mathbb{C}^{n_{\ell'}})$ are orthogonal according to

$$\int_G [\boldsymbol{\tau}_{\mathfrak{g}}^{\ell}]_{mn} \overline{[\boldsymbol{\tau}_{\mathfrak{g}}^{\ell'}]_{m'n'}} d\mathfrak{g} = \frac{1}{\eta_\ell} \delta_{\ell,\ell'} \delta_{m,m'} \delta_{n,n'} \, , \tag{2.22}$$

where η_ℓ is called the group weight. The representations $\boldsymbol{\tau}_{\mathfrak{g}}^{\ell}$ form an ONB (orthonormal basis) for square-integrable functions on G. This is known as the Peter-Weyl Theorem (see also Gaal (1973) and Naĭmark (1959)).

Theorem 2.1.18. *(Peter-Weyl Theorem (Peter and Weyl 1927)). If G is a compact linear Lie group (like $SU(n)$ and $SO(n)$), then the countably infinite set of matrix elements $\{\sqrt{\eta_\ell}[\boldsymbol{\tau}_{\mathfrak{g}}^{\ell}]_{mn}\}$ of the irreducible representations $\boldsymbol{\tau}^\ell : G \to GL(\mathbb{C}^{n_\ell})$ is an ONB for $L_2(G)$.*

As a result we can represent each $f \in L_2(G)$ in terms of $\{\sqrt{\eta_\ell}[\boldsymbol{\tau}_{\mathfrak{g}}^{\ell}]_{mn}\}$ by

$$f(\mathfrak{g}) = \sum_{\ell=0}^{\infty} \sqrt{\eta_\ell} \sum_{m=0}^{n_\ell} \sum_{n=0}^{n_\ell} A_{mn}^{\ell} [\boldsymbol{\tau}_{\mathfrak{g}}^{\ell}]_{mn} \quad . \tag{2.23}$$

The expansion coefficients (Also known as the "**generalized Fourier coefficients**") are computed by orthogonal projection of f into the space spanned by the orthonormal basis $\{\sqrt{\eta_\ell}[\boldsymbol{\tau}^\ell_{\mathfrak{g}}]_{mn}\}$:

$$\begin{aligned} A^\ell_{mn} &= \langle f, \sqrt{\eta_\ell}[\boldsymbol{\tau}^\ell_{\mathfrak{g}}]_{mn}\rangle \\ &= \sqrt{\eta_\ell}\int_G f(\mathfrak{g})\overline{[\boldsymbol{\tau}^\ell_{\mathfrak{g}}]_{mn}}d\mathfrak{g} \quad . \end{aligned} \tag{2.24}$$

If the group weight fully weights the expansion according to

$$f(\mathfrak{g}) = \sum_{\ell=0}^{\infty}\eta_\ell\sum_{m=0}^{n_\ell}\sum_{n=0}^{n_\ell}A^\ell_{mn}[\boldsymbol{\tau}^\ell_{\mathfrak{g}}]_{mn} \quad \text{and} \tag{2.25}$$

$$A^\ell_{mn} = \langle f, [\boldsymbol{\tau}^\ell_{\mathfrak{g}}]_{mn}\rangle \quad , \tag{2.26}$$

then we call the basis functions "**Schmidt semi-normalized**".

Example 2.1.19 (The 2D Rotation Group $SO(2)$). *The natural representations of the 2D rotation group are the real valued rotation matrices* $\mathbf{U}_{\mathfrak{g}} \in SO(2)$*, which are parametrized according to*

$$\mathbf{U}(\varphi) := \begin{pmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{pmatrix}, \tag{2.27}$$

with the angle $\varphi \in [0, 2\pi)$*; see Fig. 1.4 on page 19. The 2D rotation group is isomorph to the one dimensional unitary group* $U(1) := \{z \in \mathbb{C} \mid |z| = 1\}$ *which has a natural parameterization in terms of the angle* φ *via*

$$U(1) := \{e^{i\varphi} \mid \varphi \in [0, 2\pi)\} \quad , \tag{2.28}$$

that is, points on the units circle; see Fig. 1.8 on page 27 for an illustration of the 2D rotation. The irreducible representations $\boldsymbol{\tau}^\ell_{\mathfrak{g}}$ *of* $SO(2)$ *are the periodic Fourier basis functions* $Y^\ell_{\mathfrak{g}}$ *defined on the unit circle. They are defined by*

$$Y^\ell(\varphi) := e^{i\ell\varphi} \, . \tag{2.29}$$

With $\ell \in \mathbb{N}$ *we denote the angular frequency. Note that they coincide with the irreducible representations of the cyclic 1D translation group (up to scale). Hence the irreducible representations of the 2D rotation groups are just the ordinary periodic Cartesian Fourier basis functions. In the context of rotations, they are also known as circular harmonics. In Fig. 2.2 on page 48 we show two kinds of equivalent representations: (1) in the upper row in terms of functions on the circle and (2) in terms of a periodic 1D signal. According to the Peter-Weyl theorem, they form an ONB for functions on the unit circle, with* $\langle Y^{\ell_1}, Y^{\ell_2}\rangle = 2\pi\delta_{\ell_1,\ell_2}$*. Hence for any square integrable function on the circle* f *exists the Fourier expansion*

$$f(\varphi) = \frac{1}{\sqrt{2\pi}}\sum_{\ell=-\infty}^{\infty} a^\ell Y^\ell(\varphi) \quad , \tag{2.30}$$

where $a^\ell = \langle f, \frac{1}{\sqrt{2\pi}} Y^\ell \rangle$ *are the Fourier coefficients. Let* $\mathfrak{g}_\theta \in SO(2)$ *be a group element rotating the function* f *by the angle* $\theta \in (0, 2\pi]$. *The group action is defined by* $(\mathfrak{g}_\theta f)(\varphi) := f(\varphi - \theta)$. *It turns out that if* f *is represented in terms of the Fourier expansion, the group action (the rotation) is just a multiplication with the corresponding irreducible representation:*

$$
\begin{aligned}
a'^\ell &= \langle \mathfrak{g}_\theta f, \frac{1}{\sqrt{2\pi}} Y^\ell \rangle = \tfrac{1}{\sqrt{2\pi}} \int f(\varphi - \theta) \overline{Y^\ell(\varphi)} d\varphi = \tfrac{1}{\sqrt{2\pi}} \int f(\varphi) \overline{Y^\ell(\varphi + \theta)} d\varphi = \\
&= \overline{Y^\ell(\theta)} \tfrac{1}{\sqrt{2\pi}} \int f(\varphi) \overline{Y^\ell(\varphi)} d\varphi = \overline{Y^\ell(\theta)} \langle f, \tfrac{1}{\sqrt{2\pi}} Y^\ell \rangle = \overline{Y^\ell(\theta)} a^\ell \,, \qquad (2.31)
\end{aligned}
$$

where the $a'^\ell \in \mathbb{C}$ *are the new rotated expansion coefficients. Hence a rotation in Fourier domain acts on the expansion coefficients in a coefficient by coefficient manner in terms of irreducible representations themselves. In the following we shortly write* $Y_{\mathfrak{g}}^\ell$ *to indicate the irreducible representation of an* $\mathfrak{g} \in S_2$, *thus*

$$
\underbrace{(\mathfrak{g} f)(\varphi)}_{rotation} = \tfrac{1}{\sqrt{2\pi}} \sum_{\ell=-\infty}^{\infty} \underbrace{\overline{Y_{\mathfrak{g}}^\ell} a^\ell}_{=\mathbf{a}'^\ell} Y^\ell(\varphi) = \tfrac{1}{\sqrt{2\pi}} \sum_{\ell=-\infty}^{\infty} a'^\ell Y^\ell(\varphi) \quad . \qquad (2.32)
$$

Consequently, a rotation has the same representation as the basis functions. Such a representation has several benefits, where the following is of particular importance for us: rotation invariants can be formed analytically via simple products of expansion coefficients. An example is the angular power-spectrum $F_2(f, \ell) := \|a^\ell\|^2 = \langle a^\ell, a^\ell \rangle$. *Let* $\{a^\ell\}$ *and* $\{a'^\ell\}$ *be the expansion coefficients of* f *and its rotated version* $f' = (\mathfrak{g} f)$, *respectively. The angular power spectrum does not change under rotations:*

$$
\begin{aligned}
F_2(\mathfrak{g} f, \ell) = \|a'^\ell\|^2 = \langle a'^\ell, a'^\ell \rangle &= \langle \overline{Y_{\mathfrak{g}}^\ell} a^\ell, \overline{Y_{\mathfrak{g}}^\ell} a^\ell \rangle \\
&= \langle \underbrace{Y_{\mathfrak{g}}^\ell \overline{Y_{\mathfrak{g}}^\ell}}_{=1} a^\ell, a^\ell \rangle = \langle a^\ell, a^\ell \rangle = \|a^\ell\|^2 = F_2(f, \ell) \,. \qquad (2.33)
\end{aligned}
$$

The angular power spectrum is rather a weak invariant. The reason is the fact that the coefficients corresponding to different frequencies ℓ *are considered separately and thus phase information is lost. Using higher order spectra overcomes this restriction. In our work we use the 3D counterpart of the 2D angular bi-spectrum* $F_3(f, \ell_1, \ell_2) := \langle a^{\ell_{1+2}}, a^{\ell_1} a^{\ell_2} \rangle$ *to solve these ambiguities. For the 2D example above holds*

$$
\begin{aligned}
F_3(\mathfrak{g} f, \ell_1, \ell_2, \ell_3) = \langle a'^{\ell_{1+2}}, a'^{\ell_1} a'^{\ell_2} \rangle &= \langle \overline{Y_{\mathfrak{g}}^{\ell_{1+2}}} a^{\ell_{1+2}}, \overline{Y_{\mathfrak{g}}^{\ell_1}} a^{\ell_1} \overline{Y_{\mathfrak{g}}^{\ell_2}} a^{\ell_2} \rangle = \langle Y_{\mathfrak{g}}^{\ell_1} Y_{\mathfrak{g}}^{\ell_2} \overline{Y_{\mathfrak{g}}^{\ell_{1+2}}} a^{\ell_{1+2}}, a^{\ell_1} a^{\ell_2} \rangle \\
&= \langle Y_{\mathfrak{g}}^{\ell_{1+2}} \overline{Y_{\mathfrak{g}}^{\ell_{1+2}}} a^{\ell_{1+2}}, a^{\ell_1} a^{\ell_2} \rangle = \langle a^{\ell_{1+2}}, a^{\ell_1} a^{\ell_2} \rangle = F_3(f, \ell_1, \ell_2, \ell_3) \,. \qquad (2.34)
\end{aligned}
$$

The bi-spectrum describes the interrelation of the coefficients in an unique manner: the bi-spectrum defined on $SO(2)$ *yields complete features (Kakarala 1992). See also Fig. 3.3 on page 92.*

In the following we introduce the 3D rotation group. The 3D rotation group shows several similarities to the 2D rotation group. However, because of the two additional

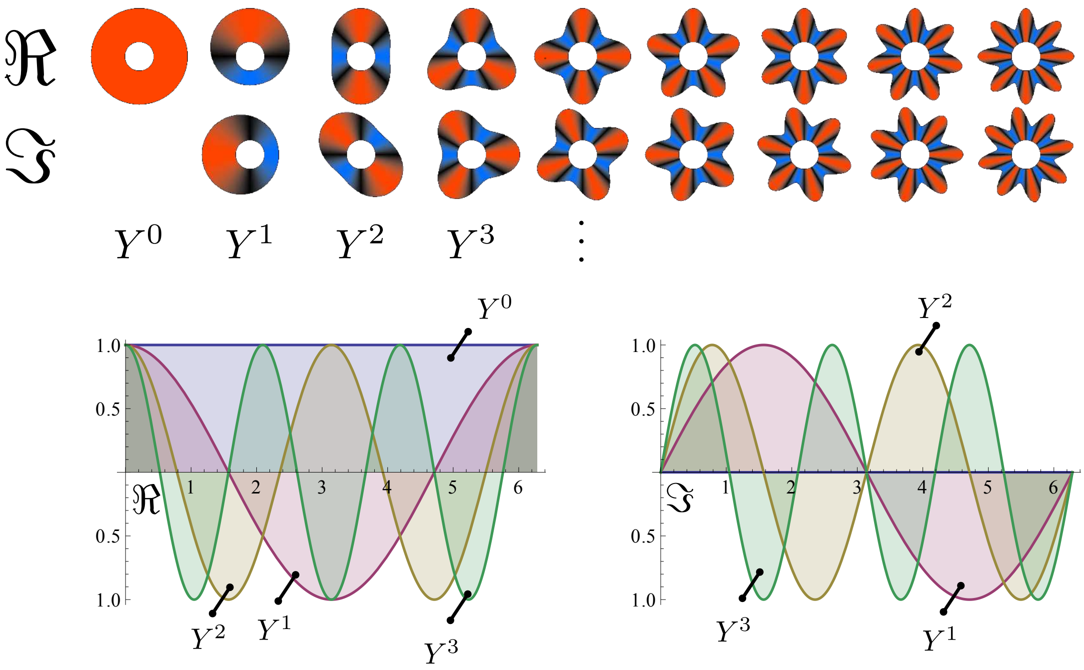


Figure 2.2.: Circular Harmonics: The Irreducible Representations of $SO(2)$. Top: representation as function on the unit circle. Bottom: representation as periodic 1D function.

rotation angles, algorithms on $SO(3)$ tend to have a higher computational complexity. We will show how to represent volumetric images in terms of the irreducible representations of the 3D rotation group with the goal to make the computation of local rotation invariant features, such as the power- and the bi-spectrum, computationally efficient.

2.2. The 3D Rotation Group SO(3)

2.2.1. Rotations in 3D: The Reducible Representations of $SO(3)$

In this section we introduce the 3D rotation and how it transforms 3D images represented in Cartesian coordinates. The 3D rotation group is denoted by $SO(3)$. The ordinary rotation matrices $\mathbf{U}_\mathfrak{g} \in SO(3)$ are the natural 3-dimensional matrix representations $\boldsymbol{\tau} : SO(3) \to SO(3)$ of the 3D rotation group $SO(3)$ (see Def. 2.1.12 on page 42 regarding finite dimensional matrix representations). They are defined by

$$\boldsymbol{\tau}(\mathfrak{g}) := \mathbf{U}_\mathfrak{g} \,. \tag{2.35}$$

Note that Eq. (2.35) is a group homomorphism into $SO(3)$ itself.

We parametrize the rotation matrices by the three Euler angles φ, θ and ψ in the following way: $\mathbf{U}(\varphi,\theta,\psi) = \mathbf{U}_z(\varphi)\mathbf{U}_x(\theta)\mathbf{U}_z(\psi)$, where

$$\mathbf{U}_z(\varphi) := \begin{pmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{U}_y(\varphi) := \begin{pmatrix} \cos\varphi & 0 & -\sin\varphi \\ 0 & 1 & 0 \\ \sin\varphi & 0 & \cos\varphi \end{pmatrix}, \quad \mathbf{U}_x(\varphi) := \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & -\sin\varphi \\ 0 & \sin\varphi & \cos\varphi \end{pmatrix}$$

are the rotation matrices, rotating around the z,y and x-axis, respectively. See Fig. 1.5 on page 21 for an example ($\mathbf{U}_y$ is not used here).

A rotation acting on a scalar valued image just moves intensity values from their current position to a new position defined by the rotation. The rotation does not alter the intensity values themselves. Thus a rotation of a scalar valued image can be purely defined by a coordinate transform. Given a volumetric image $I : \mathbb{R}^3 \to \mathbb{R}$, then a rotation of I around the origin is defined by

$$I'(\mathbf{x}) = I(\mathbf{g}(\mathbf{x}))\,, \tag{2.36}$$

where $I' : \mathbb{R}^3 \to \mathbb{R}$ is the rotated image. The mapping

$$\mathbf{g}(\mathbf{x}) = \mathbf{U}_{\mathfrak{g}}^T \mathbf{x}, \tag{2.37}$$

$\mathbf{g} : \mathbb{R}^3 \to \mathbb{R}^3$ is a linear, unitary coordinate transform, where $\mathbf{U}_{\mathfrak{g}} \in SO(3)$ is an ordinary rotation matrix. We say a group element $\mathfrak{g} \in SO(3)$ of the 3D rotation group is acting on an image I by a coordinate transform. This is denoted by

$$(\mathfrak{g}I)(\mathbf{x}) := I(\mathbf{g}(\mathbf{x})) = I(\mathbf{U}_{\mathfrak{g}}^T \mathbf{x}) \quad . \tag{2.38}$$

While for scalar valued images the rotation can be purely described in terms of a coordinate transform, in a vector or even tensor valued field the values undergo a transformation, too. An example is a gradient field which is the result of differentiation of an image in a point-by-point manner. A rotation of the image induces a rotation of the gradient field. While the coordinate transform is the same for both the image and the gradient field, the gradients themselves are rotating according to the rotation acting on the field, too. A similar transformation behavior is shown by a Hessian matrix field, which we obtain by pointwise differentiation of the gradient field.

Scalar valued images as well as all fields derived via pointwise differentiation belong to the family of Cartesian tensor fields. Their elements are called Cartesian tensors. Cartesian tensor fields show a predictable and mathematically well defined value and coordinate transformation under rotation. Therefore, tensors are an adequate tool for the design of $SE(3)$ covariant filters.

Definition 2.2.1 (Cartesian Tensor Fields). *Let* $\mathbf{T}^j : \mathbb{R}^3 \to \mathbb{C}^{3^j}$, $j \geq 0$ *be a vector valued function. We call* $\mathbf{T}^j$ *a Cartesian tensor field of rank* $j \geq 0$, *iff for all* $\mathfrak{g} \in SO(3)$

$$(\mathfrak{g}\mathbf{T}^j)(\mathbf{x}) := \mathbf{U}_{\mathfrak{g}}^{3^j} \mathbf{T}^j(\mathbf{U}_{\mathfrak{g}}^T \mathbf{x}) \quad , \tag{2.39}$$

where $\mathbf{U}_{\mathfrak{g}}^{3^n} \in SO(3^n)$ *are orthogonal rotation matrices transforming the values of a field under rotation. The index* j *is called the tensor field rank. The elements* $\mathbf{T}^j(\mathbf{x}) \in \mathbb{C}^{3^j}$ *are called Cartesian tensors.*

We can identify the elements of $\mathbf{U}_{\mathfrak{g}}^{3^n} \in SO(3^n)$ by rewriting the Cartesian tensor rotation, see Eq. (2.39), in terms of the ordinary rotation matrices $\mathbf{U}_{\mathfrak{g}} \in SO(3)$. We index the elements of the field $\mathbf{T}^j$ in a multidimensional manner: $\underset{i_k \in \{0,1,2\}}{T^j_{i_0,\cdots,i_j}}(\mathbf{x}) \in \mathbb{C}$

thus we can write the rotation as

$$(\mathfrak{g}T^j_{i_0,\cdots,i_j})(\mathbf{x}) = \underbrace{\sum_{\substack{n_0,\cdots,n_j\\ n_k\in\{0,1,2\}}} U_{\mathfrak{g}_{(i_0,n_0)}}\cdots U_{\mathfrak{g}_{(i_j,n_j)}}}_{\text{value transformation}} T^j_{n_0,\cdots,n_j}(\mathbf{U}_{\mathfrak{g}}^T\mathbf{x}) \quad , \tag{2.40}$$

see section B.2.1 on page 195 for a derivation.

It is worth mentioning that tensor fields of order 0 are simply ordinary scalar valued images thus the rotation simplifies to a pure coordinate transform. For tensors of order 1 (like image gradients) and 2 (like image Hessian matrices) we can rewrite Eq. (2.40) in terms of the following matrix products with the ordinary rotation matrix $\mathbf{U}_{\mathfrak{g}} \in SO(3)$, leading to widely known representations

$$\text{(scalar valued)} \qquad (\mathfrak{g}\mathbf{T}^0)(\mathbf{x}) = \mathbf{T}^0(\mathbf{U}_{\mathfrak{g}}^T\mathbf{x}) \quad , \tag{2.41}$$

$$\text{(vector valued)} \qquad (\mathfrak{g}\mathbf{T}^1)(\mathbf{x}) = \mathbf{U}_{\mathfrak{g}}\mathbf{T}^1(\mathbf{U}_{\mathfrak{g}}^T\mathbf{x}) \quad \text{and} \tag{2.42}$$

$$\text{(matrix valued)} \qquad (\mathfrak{g}\mathbf{T}^2)(\mathbf{x}) = \mathbf{U}_{\mathfrak{g}}\mathbf{T}^2(\mathbf{U}_{\mathfrak{g}}^T\mathbf{x})\mathbf{U}_{\mathfrak{g}}{}^T \quad . \tag{2.43}$$

Translation on tensor fields is defined as for vector valued functions. That is, a translation $\mathfrak{h}_{\mathbf{t}} \in T(3)$ is acting on a tensor field by

$$(\mathfrak{h}_{\mathbf{t}}\mathbf{T}^j)(\mathbf{r}) := \mathbf{T}^j(\mathbf{r}-\mathbf{t}) \quad . \tag{2.44}$$

Example 2.2.2. *[A gradient image is a tensor field of order 1] Let* $\nabla I = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})^T I$ *be the gradient image of* I*. Then the group action of an element* $\mathfrak{g} \in SO(3)$ *on* ∇I *is*

$$(\mathfrak{g}(\nabla I))(\mathbf{x}) = \mathbf{U}_{\mathfrak{g}}(\nabla I)(\mathbf{U}_{\mathfrak{g}}^T\mathbf{x}) \quad . \tag{2.45}$$

This is a direct conclusion of Eq. (B.7) *in the appendix and by setting* $\mathbf{T}^1 = (\nabla I)$.

Example 2.2.3 (Moments of order j are Cartesian tensors of order j)**.** *Image moments are widely used as shape descriptors for the purpose of object detection in images in various applications (Flusser and Suk 2006; Hu 1962; Li 1992; Reddi 1981; Reiss 1991). Let* $\boldsymbol{\mathcal{M}}^j\{I\} : \mathbb{R}^3 \to \mathbb{R}^{3^n}$ *be the jth order moment field of an image* $I : \mathbb{R}^3 \to \mathbb{R}$*. Its* 3^n *components are defined by*

$$\mathcal{M}^j_{\substack{i_0,\cdots,i_j\\ i_k\in\{0,1,2\}}}\{I\}(\mathbf{x}) := \int_{\mathbb{R}^3} I(\mathbf{r}+\mathbf{x}) r_{i_0}\cdots r_{i_j} d\mathbf{r} \, . \tag{2.46}$$

If a $\mathfrak{g} \in SO(3)$ *is acting on* $\boldsymbol{\mathcal{M}}^j$*, then*

$$\begin{aligned}
(\mathfrak{g}\mathcal{M}^j_{\substack{i_0,\cdots,i_j\\ i_k\in\{0,1,2\}}}\{I\})(\mathbf{x}) &= \mathcal{M}^j_{\substack{i_0,\cdots,i_j\\ i_k\in\{0,1,2\}}}\{\mathfrak{g}I\}(\mathbf{x}) = \int_{\mathbb{R}^3} (\mathfrak{g}I)(\mathbf{r}+\mathbf{x}) r_{i_0}\cdots r_{i_j} d\mathbf{r} \\
&= \int_{\mathbb{R}^3} I(\mathbf{U}_{\mathfrak{g}}^T(\mathbf{r}+\mathbf{x})) r_{i_0}\cdots r_{i_j} d\mathbf{r} = \int_{\mathbb{R}^3} I(\mathbf{U}_{\mathfrak{g}}^T\mathbf{r}+\mathbf{U}_{\mathfrak{g}}^T\mathbf{x}) r_{i_0}\cdots r_{i_j} d\mathbf{r} \quad (\mathbf{r}' = \mathbf{U}_{\mathfrak{g}}^T\mathbf{r}) \\
&= \sum_{\substack{n_0,\cdots,n_j\\ n_k\in\{0,1,2\}}} U_{\mathfrak{g}_{(i_0,n_0)}}\cdots U_{\mathfrak{g}_{(i_j,n_j)}} \int_{\mathbb{R}^3} r'_{n_0}\cdots r'_{n_j} I(\mathbf{r}'+\mathbf{U}_{\mathfrak{g}}^T\mathbf{x}) d\mathbf{r}' \\
&= \sum_{\substack{n_0,\cdots,n_j\\ n_k\in\{0,1,2\}}} U_{\mathfrak{g}_{(i_0,n_0)}}\cdots U_{\mathfrak{g}_{(i_j,n_j)}} \mathcal{M}^j_{n_0,\cdots,n_j}\{I\}(\mathbf{U}_{\mathfrak{g}}^T\mathbf{x}) \quad .
\end{aligned} \tag{2.47}$$

Therefore, $\boldsymbol{\mathcal{M}}^j\{I\}$ *is a Cartesian tensor field of order j.*

Example 2.2.4. *[The 3D structure Tensor is a Cartesian tensors of order 2] The structure tensor represents the distribution of gradient directions within a voxel's neighborhood defined by a (usually isotropic) window function* $w : \mathbb{R}^3 \to \mathbb{R}$*. Such a window function is often an isotropic Gaussian, with* $w(r) = e^{\frac{-r^2}{2\sigma^2}}$ *(We only consider isotropic smoothing here. For anisotropic smoothing see e.g. the papers by Brox et al. (2006); Reisert and Skibbe (2012)). The structure tensor is widely used for low-level feature analysis in images; see particularly the early works of Bigun and Granlund (1987); Förstner (1986); Harris and Stephens (1988). The structure tensor is highly related to our proposed SHOG features which are introduced in section 5.3 on page 138. The interested reader might directly jump to section 5.3.1.3 on page 148, where the relation of SHOG to the structure tensor is discussed.*

Let $\boldsymbol{\mathcal{S}}\{I\} : \mathbb{R}^3 \to \mathbb{R}^9$ *be the the structure tensor field of an image I. Its nine components are defined by*

$$\mathcal{S}_{n,m}\{I\}(\mathbf{x}) := \int_{\mathbb{R}^3} (\tfrac{\partial}{\partial x_n} I)(\mathbf{x}-\mathbf{r})(\tfrac{\partial}{\partial x_m} I)(\mathbf{x}-\mathbf{r})w(r)d\mathbf{r} \quad , \tag{2.48}$$

that is, all Gaussian smoothed second order monomials formed from the first order derivatives of I. If a rotation $\mathfrak{g} \in SO(3)$ *is acting on* $\boldsymbol{\mathcal{S}}$*, then (utilizing Eq.* (B.5)*)*

$$\begin{aligned}
(\mathfrak{g}\mathcal{S}_{n,m}\{I\})(\mathbf{x}) &= (\mathcal{S}_{n,m}\{\mathfrak{g}I\})(\mathbf{x}) = \int_{\mathbb{R}^3} (\tfrac{\partial}{\partial x_n}(\mathfrak{g}I))(\mathbf{r}-\mathbf{x})(\tfrac{\partial}{\partial x_m}(\mathfrak{g}I))(\mathbf{r}-\mathbf{x})w(r)d\mathbf{r} \\
&= \sum_{\substack{n_0,n_1 \\ n_k \in \{0,1,2\}}} U_{\mathfrak{g}_{(n,n_0)}} U_{\mathfrak{g}_{(m,n_1)}} \int_{\mathbb{R}^3} (\tfrac{\partial}{\partial x_{n_0}} I)(\mathbf{U}_\mathfrak{g}^T\mathbf{x} - \mathbf{U}_\mathfrak{g}^T\mathbf{r})(\tfrac{\partial}{\partial x_{n_1}} I)(\mathbf{U}_\mathfrak{g}^T\mathbf{x} - \mathbf{U}_\mathfrak{g}^T\mathbf{r})w(r)d\mathbf{r} \\
&\text{(since } w \text{ is isotropic, the rotation has no effect on } w\text{)} \\
&= \sum_{\substack{n_0,n_1 \\ n_k \in \{0,1,2\}}} U_{\mathfrak{g}_{(n,n_0)}} U_{\mathfrak{g}_{(m,n_1)}} \int_{\mathbb{R}^3} (\tfrac{\partial}{\partial x_{n_0}} I)(\mathbf{U}_\mathfrak{g}^T\mathbf{x} - \mathbf{r})(\tfrac{\partial}{\partial x_{n_1}} I)(\mathbf{U}_\mathfrak{g}^T\mathbf{x} - \mathbf{r})w(r)d\mathbf{r} \\
&= \sum_{\substack{n_0,n_1 \\ n_k \in \{0,1,2\}}} U_{\mathfrak{g}_{(n,n_0)}} U_{\mathfrak{g}_{(m,n_1)}} \mathcal{S}_{n_0,m_1}\{I\}(\mathbf{U}_\mathfrak{g}^T\mathbf{x}) \quad .
\end{aligned} \tag{2.49}$$

Therefore, $\boldsymbol{\mathcal{S}}\{I\}$ *is a Cartesian tensor field of order* 2*.*

2.2.1.1. Sparse Representations of Cartesian Tensors

We can always represent a Cartesian tensor rotation in terms of a matrix-vector product $(\mathfrak{g}\mathbf{T}^j)(\mathbf{x}) = \mathbf{U}_\mathfrak{g}^{3^n}\mathbf{T}^j(\mathbf{U}_\mathfrak{g}^T\mathbf{x})$; see Eq. (2.39). It is obvious that a rotation of a higher order Cartesian tensor is computationally expensive, particularly when rotating a whole tensor field. Let $\mathbf{U}_\mathfrak{g}^{(3^n)} \in SO(3^n)$ be a matrix representation of a rotation acting on Cartesian tensors of order n. Since the rotation group is a compact group, there must exist irreducible representations and in fact, they are not the rotation matrices $\mathbf{U}_\mathfrak{g}^{(3^n)}$ themselves. Consequently, for each matrix representation $\mathbf{U}_\mathfrak{g}^{(3^n)} \in SO(3^n)$ exists an equivalent representation that can be turned into a maximum sparse block diagonal matrix via an unitary transformation. With such a transformation comes a decomposition of the corresponding Cartesian tensor space into irreducible subspaces.

We exemplarily consider a second order tensor field because it eases the usage of ordinary matrix calculus. The elements of a second order tensor field $\mathbf{T}^2$ are reducible with respect to rotation. In order to use common matrix calculus we consider the elements of $\mathbf{T}^2$ w.l.o.g as tensors in matrix form. In this case

$$\mathbf{T}^2 = \begin{pmatrix} T^2_{00} & T^2_{01} & T^2_{02} \\ T^2_{10} & T^2_{11} & T^2_{12} \\ T^2_{20} & T^2_{21} & T^2_{22} \end{pmatrix} \quad . \tag{2.50}$$

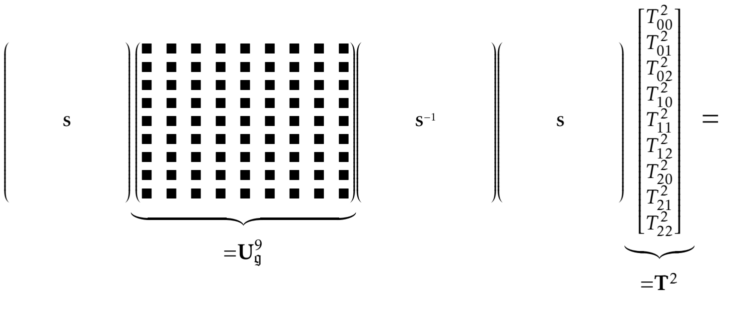
Any Cartesian second order tensor can be uniquely decomposed into three irreducible components:

$$\mathbf{T}^2 = \alpha \mathbf{I}_{3\times 3} + \mathbf{T}^2_{\text{anti}} + \mathbf{T}^2_{\text{sym}} \quad , \tag{2.51}$$

where $\mathbf{T}^2_{\text{anti}}(\mathbf{x})$ is an antisymmetric matrix, $\mathbf{T}^2_{\text{sym}}(\mathbf{x})$ a traceless ($\text{trace}\mathbf{T}^2_{\text{sym}}(\mathbf{x}) = 0$) symmetric matrix and $\alpha(\mathbf{x}) \in \mathbb{C}$ the trace of $\mathbf{T}^2(\mathbf{x})$; see e.g. the book chapter by Reisert and Burkhardt (2009b) for further details. For instance, a Hessian matrix of an image is a second order tensor which is a symmetric matrix, thus the antisymmetric part would be $\mathbf{0} \in \mathbb{C}^{3\times 3}$.

Consider the 9×9 rotation matrices $\mathbf{U}^9_\mathfrak{g} \in SO(9)$ acting on the Cartesian tensor fields of order 2. There exists, together with a unitary transformation $\mathbf{S} \in SU(9)$, a transformation into a maximum sparse block-diagonal rotation matrix consisting of irreducible unitary matrices $\mathbf{D}^\ell_\mathfrak{g} \in SU(2\ell+1)$, $\ell = \{0,1,2\}$ representing the rotations acting on the trace α, the antisymmetric part $\mathbf{T}^2_{\text{anti}}$ and the symmetric part $\mathbf{T}^2_{\text{sym}}$, respectively. This means, the transform $\mathbf{S}$ divides the Cartesian 2nd order tensor space (the vector space $\mathbb{C}^9$) into subspaces $\mathbb{C}^{2\ell+1}$, $\ell = \{0,1,2\}$ which are invariant under rotation (we consider here matrix multiplications as operations acting on the tensor field in an element-by-element manner):

$$\mathbf{S}\mathbf{U}^9_\mathfrak{g}\mathbf{S}^{-1}\mathbf{S}\mathbf{T}^2 = \tag{2.52}$$



$$
= \underbrace{\left[\underbrace{\begin{pmatrix} \blacksquare & 0 \\ 0 & \end{pmatrix}}_{=\mathbf{D}^0_{\mathfrak{g}} \in SU(1)} + \underbrace{\begin{pmatrix} \begin{matrix} \blacksquare & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \blacksquare \end{matrix} & 0 \\ 0 & \end{pmatrix}}_{=\mathbf{D}^1_{\mathfrak{g}} \in SU(3)} + \underbrace{\begin{pmatrix} & 0 \\ 0 & \begin{matrix} \blacksquare & \blacksquare & \blacksquare & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \blacksquare & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \blacksquare & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \blacksquare & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \blacksquare & \blacksquare & \blacksquare \end{matrix} \end{pmatrix}}_{=\mathbf{D}^2_{\mathfrak{g}} \in SU(5)} \right]}_{\substack{=\mathbf{S}\mathbf{U}^9_{\mathfrak{g}}\mathbf{S}^{-1}\text{,sparse block-diagonal matrix representing the} \\ \text{rotation acting on } \alpha\text{, acting on the antisymmetric part } \mathbf{T}^2_{\text{anti}} \\ \text{and on the symmetric part } \mathbf{T}^2_{\text{sym}}}}
\underbrace{\begin{bmatrix} \frac{-(T_{00}+T_{11}+T_{22})}{\sqrt{3}} \\ \frac{1}{2}(T_{20}-T_{02}+i(T_{21}-T_{12})) \\ \frac{i}{\sqrt{2}}(T_{10}-T_{01}) \\ \frac{1}{2}(T_{20}-T_{02}-i(T_{21}-T_{12})) \\ \frac{1}{2}(T_{00}-T_{11}+i(T_{01}+T_{10})) \\ \frac{1}{2}((T_{02}+T_{20})+i(T_{12}+T_{21})) \\ \frac{-1}{\sqrt{6}}(T_{00}+T_{11}-2T_{22}) \\ \frac{1}{2}(-(T_{02}+T_{20})+i(T_{12}+T_{21})) \\ \frac{1}{2}(T_{00}-T_{11}-i(T_{01}+T_{10})) \end{bmatrix}}_{\substack{\mathbf{T}^{2\prime}=\mathbf{S}\mathbf{T}^2 \\ \mathbf{T}^{2\prime}\in\mathbb{C}^9}}
= \begin{bmatrix} \mathbf{a}^0 \\ \mathbf{a}^1 \\ \mathbf{a}^2 \end{bmatrix} ,
\tag{2.53}
$$

where $\mathbf{a}^0 : \mathbb{R}^3 \to \mathbb{C}$, $\mathbf{a}^1 : \mathbb{R}^3 \to \mathbb{C}^3$ and $\mathbf{a}^2 : \mathbb{R}^3 \to \mathbb{C}^5$ are the new tensor fields representing the trace, antisymmetric part and the symmetric part of $\mathbf{T}^2$; see section 2.2.3.7 on page 78 for the exact mapping between Cartesian tensors and their irreducible counterpart. Hence the vector $\mathbf{T}^{2'}$ can be written as a direct sum of these vectors, thus $\mathbf{T}^{2'} = \mathbf{a}^0 \oplus \mathbf{a}^1 \oplus \mathbf{a}^2$. Such a decomposition exists for all Cartesian tensors (and tensor fields). It is worth mentioning that tensor fields of rank 0 (images) and 1 (vectors) are already irreducible. Consequently, regarding computational complexity and memory storage there exist equivalent irreducible matrix representations that are much better suited for representing 3D rotations than the Cartesian rotation matrices $\mathbf{U}^{3^n}_{\mathfrak{g}}$: the rotations acting on invariant, irreducible subspaces.

We will see in the next section that the irreducible representations $\mathbf{D}^{\ell}_{\mathfrak{g}}$ of $SO(3)$ are the so-called Wigner D-matrices. Similar to ordinary rotation matrices acting on Cartesian tensors, the Wigner D-matrices are acting on vector spaces whose elements are called spherical tensors. Spherical tensors will be introduced later in section 2.2.3 on page 61.

2.2.2. The Irreducible Representations of $SO(3)$

Square-integrable functions on $SO(3)$ have the form $f : SO(3) \to \mathbb{C}$. Following the Euler angle convention we have the parameterization $f(\varphi, \theta, \psi)$. The irreducible unitary representations of $SO(3)$ are the so-called Wigner D-Matrices (Wigner 1931) which we denote by $\boldsymbol{\tau}^{\ell}_{\mathfrak{g}} = \mathbf{D}^{\ell}_{\mathfrak{g}} \in SU(2\ell+1)$, where

$$
D^{\ell}_{mn}(\varphi, \theta, \psi) := e^{-im\varphi} d^{\ell}_{mn}(\theta) e^{-in\psi}, \qquad -\ell \leq m, n \leq \ell \quad . \tag{2.54}
$$

With $\mathbf{d}^{\ell} \in \mathbb{R}^{(2\ell+1)\times(2\ell+1)}$ we denote the small Wigner d-matrices (see Eq. C.39 in the appendix for an explicit expression). As stated in the Peter-Weyl theorem 2.1.18, the set $\{\frac{\sqrt{2\ell+1}}{\sqrt{8\pi}} D^{\ell}_{mn}\}$ is an ONB of $L_2(SO(3))$ ($8\pi^2$ is the volume of $SO(3)$, see Eq. (C.3) in the appendix).

We prefer w.l.o.g the Schmidt semi-normalized representation of the basis functions. In this case each function $f \in L_2(SO(3))$ can be expanded in the following manner:

$$f(\varphi,\theta,\psi) = \sum_{\ell=0}^{\infty} \tfrac{(2\ell+1)}{8\pi^2} \sum_{m,n=-\ell}^{\ell} A^{\ell}_{m,n} D^{\ell}_{mn}(\varphi,\theta,\psi) \quad . \tag{2.55}$$

We obtain the expansion coefficients $\mathbf{A}^{\ell} \in \mathbb{C}^{(2\ell+1)\times(2\ell+1)}$ via orthogonal projection onto the unnormalized basis function according to Eq. (2.26).

2.2.2.1. Spherical Harmonic Functions

For our applications, functions that are independent from the angle ψ i.e. $f(\theta,\varphi) = f'(\varphi,\theta,\psi)$, are of particular interest. We use such functions to represent the angular part of volumetric images in terms of spherical coordinates (θ,φ,r). Functions depending only on (θ,φ) are functions living on the 2-sphere. The 2-sphere is denoted by S_2. For functions on the 2-sphere, the expansion coefficients $A^{\ell}_{mn} = \langle f, [\mathbf{D}^{\ell}_{\mathfrak{g}}]_{mn}\rangle$ become 0 for all $[\mathbf{D}^{\ell}_{\mathfrak{g}}]_{mn}$ where $n \neq 0$. Consequently,

Theorem 2.2.5. *The set $\{[\mathbf{D}^{\ell}_{\mathfrak{g}}]_{m0}\}$ builds an orthogonal basis for functions $f \in L_2(S_2)$.*

Proof: $\{[\mathbf{D}^{\ell}_{\mathfrak{g}}]_{m0}\}$ is an orthogonal basis for $f \in L_2(S_2)$. It is sufficient to show that the expansion coefficients associated with a Wigner D-matrices with $n = 0$ are equal to 0. Let $f \in L_2(S_2)$ be a function on the 2-sphere, then

$$\begin{aligned}
A^{\ell}_{mn} &= \langle f, [\mathbf{D}^{\ell}_{\mathfrak{g}}]_{mn}\rangle \\
&= \int_{SO(3)} f(\mathfrak{g})\overline{[\mathbf{D}^{\ell}_{\mathfrak{g}}]_{mn}} d\mathfrak{g} \\
&= \int_{\varphi=0}^{2\pi}\int_{\theta=0}^{\pi}\int_{\psi=0}^{2\pi} f(\mathfrak{g})\overline{D^{\ell}_{mn}(\varphi,\theta,\psi)}\sin(\theta)d\psi d\theta d\varphi \\
&= \int_{0}^{2\pi}\int_{0}^{\pi}\int_{0}^{2\pi} f(\mathfrak{g})e^{im\varphi}d^{\ell}_{mn}(\theta)e^{in\psi}\sin(\theta)d\psi d\theta d\varphi \\
&= \int_{0}^{2\pi}\int_{0}^{\pi} f(\theta,\varphi)e^{im\varphi}d^{\ell}_{mn}(\theta)\sin(\theta)\underbrace{\int_{0}^{2\pi} e^{in\psi}d\psi}_{=2\pi\delta_{n,0}} d\theta d\varphi \\
&= \delta_{n,0}\, 2\pi \int_{0}^{2\pi}\int_{0}^{\pi} f(\theta,\varphi)e^{im\varphi}d^{\ell}_{mn}(\theta)\sin(\theta)d\theta d\varphi \\
&= \underline{\underline{\delta_{n,0}}}2\pi \int_{0}^{2\pi}\int_{0}^{\pi} f(\theta,\varphi)\; \underbrace{\overline{D^{\ell}_{m0}(\varphi,\theta,0)}}_{\text{according to (2.54)}}\; \sin(\theta)d\theta d\varphi \quad .
\end{aligned} \tag{2.56}$$

□

The explicit expression of the basis functions $[\mathbf{D}^{\ell}_{\mathfrak{g}}]_{m0}$ is

$$\begin{aligned}
D^{\ell}_{m0}(\varphi,\theta,0) &= \sqrt{\frac{(\ell-m)!}{(\ell+m)!}} P^{m}_{\ell}(\cos(\theta))e^{-im\varphi} \\
&= \overline{Y^{\ell}_{m}(\theta,\varphi)} \quad ,
\end{aligned} \tag{2.57}$$

where the functions $P_\ell^m : [-1,1] \to \mathbb{R}$ are the orthogonal associated Legendre polynomials. An explicit definition of P_ℓ^m can be found in Def. C.3.2 in the appendix.

Definition 2.2.6 (Spherical Harmonics)**.** *The functions* $Y_m^\ell : \mathbb{R}^3 \to \mathbb{C}$ *are widely known as* ***spherical harmonics*** *(we use the Schmidt semi-normalized spherical harmonics), where*

$$Y_m^\ell(\theta,\varphi) = \sqrt{\frac{(\ell-m)!}{(\ell+m)!}} P_\ell^m(\cos(\theta)) e^{im\varphi} \quad . \tag{2.58}$$

Apart from the fact that spherical harmonics are functions on the sphere, we consider them w.l.o.g as functions defined on $\mathbb{R}^3$*. Thus* $Y_m^\ell(\mathbf{r}) = Y_m^\ell(\mathbf{n}) = Y_m^\ell(\theta,\varphi)$*, where* $\mathbf{n} = \frac{\mathbf{r}}{\|\mathbf{r}\|}$*. That is, the value only changes in angular direction and is constant in radial direction.*

Note, that we arrange spherical harmonic functions to vector valued functions $\mathbf{Y}^\ell = (Y_{-\ell}^\ell, \cdots, Y_\ell^\ell)^T$, $\mathbf{Y}^\ell : \mathbb{R}^3 \to \mathbb{C}^{2\ell+1}$. We say $\mathbf{Y}^\ell$ is a spherical harmonic of order ℓ. Then, spherical harmonics coincides with the center column of the Wigner D-matrix $\mathbf{D}^\ell(\varphi,\theta,0)$ and a rotation in irreducible representations becomes a matrix-vector multiplication (we come back to this point immediately below).

The first order spherical harmonic is of particular importance for our applications, because based on it all remaining spherical harmonic functions can be computed in a recursive manner. We discuss details concerning this property later in section 2.2.3 on page 61, where we introduce the spherical tensor algebra. The representation of the first order spherical harmonic in Cartesian coordinates is

$$\mathbf{Y}^1(\mathbf{x}) = \frac{\left(\frac{1}{\sqrt{2}}(x-iy), z, -\frac{1}{\sqrt{2}}(x+iy)\right)^T}{\|\mathbf{x}\|} \quad ; \tag{2.59}$$

see (C.10) in the appendix.

Since all unitary matrix representations are orthogonal (Eq. (2.22), the spherical harmonics as part of the Wigner D-Matrices are orthogonal, too:

$$\begin{aligned}
\langle Y_m^\ell, Y_{m'}^{\ell'} \rangle &= \int_{S_2} Y_m^\ell(\mathfrak{g}) \overline{Y_{m'}^{\ell'}(\mathfrak{g})} d\mathfrak{g} \\
&= \int_{\theta=0}^{\pi} \int_{\varphi=0}^{2\pi} Y_m^\ell(\theta,\varphi) \overline{Y_{m'}^{\ell'}(\theta,\varphi)} \sin(\theta) d\theta d\varphi \\
&= \frac{4\pi}{(2\ell+1)} \delta_{\ell,\ell'} \delta_{m,m'} \quad .
\end{aligned} \tag{2.60}$$

Definition 2.2.7. *[Rotation of Spherical Harmonics] The group action of an element* $\mathfrak{g} \in SO(3)$*, acting on the spherical harmonic functions, is defined by*

$$(\mathfrak{g}\mathbf{Y}^\ell)(\mathbf{n}) := \mathbf{D}_\mathfrak{g}^\ell \mathbf{Y}^\ell(\mathbf{U}_\mathfrak{g}^T \mathbf{n}) = \mathbf{Y}^\ell(\mathbf{n}) \quad ; \tag{2.61}$$

that is, the spherical harmonics are ***fix*** *with respect to rotation (We use the notation of Reisert and Burkhardt (2009b)).*

As a result, we have the following identities:

$$\begin{aligned} & \mathbf{D}^{\ell}_{\mathfrak{g}}\mathbf{Y}^{\ell}(\mathbf{n}) = \mathbf{Y}^{\ell}(\mathbf{U}_{\mathfrak{g}}\mathbf{n}) \\ \Leftrightarrow \quad & \mathbf{Y}^{\ell}(\mathbf{n}) = (\mathbf{D}^{\ell}_{\mathfrak{g}})^{*}\mathbf{Y}^{\ell}(\mathbf{U}_{\mathfrak{g}}\mathbf{n}) \\ \Leftrightarrow \quad & \mathbf{Y}^{\ell}(\mathbf{n}) = \mathbf{D}^{\ell}_{\mathfrak{g}}\mathbf{Y}^{\ell}(\mathbf{U}_{\mathfrak{g}}{}^{T}\mathbf{n}) \, . \end{aligned} \tag{2.62}$$

The spherical harmonics are spanning $L_2(S_2)$. Hence there exists a representation of any function $f \in L_2(S_2)$ in terms of spherical harmonics, which is

$$\begin{aligned} f(\mathbf{n}) &= \sum_{\ell=0}^{\infty} \tfrac{2\ell+1}{4\pi} \sum_{m=-\ell}^{m=\ell} v^{\ell}_{m} \overline{Y^{\ell}_{m}(\mathbf{n})} \\ &= \sum_{\ell} \tfrac{2\ell+1}{4\pi} (\mathbf{v}^{\ell})^{T} \overline{\mathbf{Y}^{\ell}(\mathbf{n})} \quad , \end{aligned} \tag{2.63}$$

where $\mathbf{v}^{\ell} \in \mathbb{C}^{2\ell+1}$ are vector valued expansion coefficients. We obtain the expansion coefficients via projection on all $(2\ell+1)$ spherical harmonic functions associated with an angular frequency ℓ in a component-by-component manner:

$$\mathbf{v}^{\ell} = \langle f, \overline{\mathbf{Y}^{\ell}} \rangle = \int_{S_2} f(\mathbf{n})\mathbf{Y}^{\ell}(\mathbf{n}) d\mathbf{n} \quad . \tag{2.64}$$

Theorem 2.2.8 (The Spherical Harmonic Projection is Covariant to 3D Rotations)**.** *Let $f \in L_2(S_2)$. Let further $\mathfrak{g} \in SO(3)$ be a rotation acting on f thus $f'(\mathbf{n}) := (\mathfrak{g}f)(\mathbf{n}) = f(\mathbf{U}^{T}_{\mathfrak{g}}\mathbf{n})$ is the rotated function. The projection $\langle f, \overline{Y^{\ell}_{m}} \rangle$ is covariant to 3D rotations in the sense that*

$$\langle (\mathfrak{g}f), \overline{\mathbf{Y}^{\ell}} \rangle = \mathbf{D}^{\ell}_{\mathfrak{g}} \langle f, \overline{\mathbf{Y}^{\ell}} \rangle \quad , \tag{2.65}$$

where $\mathbf{D}^{\ell}_{\mathfrak{g}} \in SU(2\ell+1)$ are Wigner D-matrices.

Proof: The Spherical Harmonic Projection is Covariant to 3D Rotations.

$$\begin{aligned} \langle \mathfrak{g}f, \overline{\mathbf{Y}^{\ell}} \rangle &= \int_{\mathbf{n} \in S_2} \underbrace{f(\mathbf{U}^{T}_{\mathfrak{g}}\mathbf{n})}_{\substack{SO(3) \text{ group action} \\ \text{on spherical func.}}} \mathbf{Y}^{\ell}(\mathbf{n}) d\mathbf{n} \\ &= \int_{\mathbf{n} \in S_2} f(\mathbf{n})\mathbf{Y}^{\ell}(\mathbf{U}_{\mathfrak{g}}\mathbf{n}) d\mathbf{n} \\ &= \int_{\mathbf{n} \in S_2} f(\mathbf{n})\mathbf{D}^{\ell}_{\mathfrak{g}}\mathbf{Y}^{\ell}(\mathbf{n}) d\mathbf{n} \\ &= \mathbf{D}^{\ell}_{\mathfrak{g}} \int_{\mathbf{n} \in S_2} f(\mathbf{n})\mathbf{Y}^{\ell}(\mathbf{n}) d\mathbf{n} = \underbrace{\mathbf{D}^{\ell}_{\mathfrak{g}} \langle f, \overline{\mathbf{Y}^{\ell}} \rangle}_{\substack{SO(3) \text{ group action} \\ \text{on projection.}}} \quad . \end{aligned} \tag{2.66}$$

□

A direct consequence of Thm. 2.2.8 is the following corollary.

Corollary 2.2.9 (Rotation of a function in the harmonic domain). *Let* $\mathbf{v}^\ell$ *be the expansion coefficients of* $f \in L_2(S_2)$. *Let further* $\mathfrak{g} \in SO(3)$ *be a rotation acting on* f *thus* $f'(\mathbf{n}) := (\mathfrak{g}f)(\mathbf{n}) = f(\mathbf{U}_\mathfrak{g}^T\mathbf{n})$ *is the rotated function. Then the corresponding rotation of* f *in frequency domain is a rotation of the coefficients* $\mathbf{v}^\ell$ *by the Wigner D-matrices, with*

$$(\mathfrak{g}f)(\mathbf{n}) = \sum_\ell \tfrac{2\ell+1}{4\pi} \left(\mathbf{D}^\ell_\mathfrak{g}\mathbf{v}^\ell\right)^T \overline{\mathbf{Y}^\ell(\mathbf{n})} \,. \tag{2.67}$$

(Because $\langle(\mathfrak{g}f), \overline{\mathbf{Y}^\ell}\rangle = \mathbf{D}^\ell_\mathfrak{g}\langle f, \overline{\mathbf{Y}^\ell}\rangle = \mathbf{D}^\ell_\mathfrak{g}\mathbf{v}^\ell$ *)*

Proposition 2.2.10 (Parity of Spherical Harmonics). *The spherical harmonics have a well defined parity regarding reflections about the origin:*

$$\mathbf{Y}^\ell(-\mathbf{n}) = (-1)^\ell \mathbf{Y}^\ell(\mathbf{n}) \,. \tag{2.68}$$

Proof: Parity of Spherical Harmonics. Using the parity of the Legendre polynomials (Eq. (C.9) in the appendix) we obtain

$$\begin{aligned} \mathbf{Y}^\ell(-\mathbf{n}) &= \mathbf{Y}^\ell(\theta - \pi, \varphi + \pi) = \sqrt{\tfrac{(\ell-m)!}{(\ell+m)!}} P^m_\ell(\cos(\theta - \pi)) e^{im(\varphi+\pi)} \\ &= \sqrt{\tfrac{(\ell-m)!}{(\ell+m)!}} (-1)^{\ell-m} P^m_\ell(\cos(\theta))(-1)^\ell e^{im\varphi} = (-1)^\ell \mathbf{Y}^\ell(\theta, \varphi) = (-1)^\ell \mathbf{Y}^\ell(\mathbf{n}) \,. \end{aligned} \tag{2.69}$$

□

Proposition 2.2.11 (Symmetry of Spherical Harmonics). *The spherical harmonics show a certain symmetry which is*

$$Y^\ell_m(\mathbf{n}) = (-1)^m \overline{Y^\ell_{-m}(\mathbf{n})} \quad . \tag{2.70}$$

Proof: Symmetry of Spherical Harmonics. Using the definition of the Legendre polynomials for negative m (Eq. (C.7) in the appendix) we get

$$\begin{aligned} (-1)^m \overline{Y^\ell_{-m}(\mathbf{n})} &= (-1)^m \sqrt{\tfrac{(\ell+m)!}{(\ell-m)!}} P^{(-m)}_\ell(\cos(\theta)) \overline{e^{-im\varphi}} \\ &= (-1)^m \sqrt{\tfrac{(\ell+m)!}{(\ell-m)!}} (-1)^m \frac{(l-m)!}{(l+m)!} P^m_\ell(\cos(\theta)) e^{im\varphi} = Y^\ell_m(\mathbf{n}) \,. \end{aligned} \tag{2.71}$$

□

2.2.2.2. An $SE(3)$ Covariant Spherical Harmonic Transformation

Any $SO(3)$ covariant local feature extraction F induces an $SE(3)$ covariant filter $\mathcal{F}$ (see Eq. (1.8) on page 24). We have seen that Cartesian tensor fields offer a predictable transformation under rotation and translation thus tensors are often used for local feature representation building the ingredients of filters for e.g. corner, edge or blob detection. However, the rotation of Cartesian tensors is computationally expensive, thus higher order Cartesian tensors are rarely used; particularly in 3D. We have seen that irreducible representations of the 3D rotations offer an alternative to classical 3D rotations guaranteeing the most sparsest representation of rotations. Spherical harmonic basis functions can be used to connect volumetric

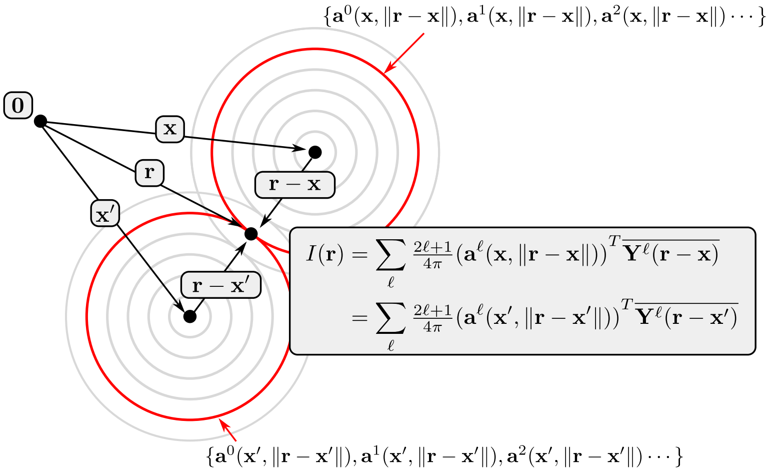


Figure 2.3.: "The whole image in a point": The idea behind the dense spherical harmonic expansion is that the whole image can be reconstructed from the coefficients at each voxel. In practice, we restrict the expansion to a small voxel-neighborhood thus only local image patches are sensed by the dense harmonic transformation.

images with the irreducible representations of $SO(3)$ with the aim of building "intelligent saliency maps" that rotate efficiently in a predictable manner.

A solution that allows for using the spherical harmonics framework for volumetric images is based on extracting local features via projection onto spherical harmonic functions. The spherical harmonics themselves build a basis for functions on the sphere. Assume a parameterization of images in spherical coordinates. We introduce an additional radial component that covers the radial part of the image. Together with the spherical harmonics covering the angular part we obtain new functions covering the whole 3D space. A straight forward approach for covering the radial part is based on nested spheres, each separately represented in terms of spherical harmonics as proposed in the paper by Kazhdan et al. (2003).

The spherical harmonic functions $\{\sqrt{\frac{(2\ell+1)}{4\pi}} Y^\ell_m\}$ build an ONB of $L_2(S_2)$ (see Eq. (2.60) and Eq. (2.22)). Considering a parameterization of a volumetric image in spherical coordinates $I(\theta, \varphi, r)$ instead of Cartesian $I(x, y, z)$, the spherical harmonics build a suitable basis for covering the angular part. Any image $I \in L_2(\mathbb{R}^3)$ can be projected on spherical harmonics via an ordinary inner product for 3D functions via

$$\langle I, \overline{\mathbf{Y}^\ell} \delta_r \rangle = \int_{\mathbf{r}' \in \mathbb{R}^3} I(\mathbf{r}') \mathbf{Y}^\ell(\mathbf{r}') \underbrace{\delta(\|\mathbf{r}'\| - r)}_{\substack{\text{selects a spherical} \\ \text{fraction of the image}}} d\mathbf{r}' \quad , \tag{2.72}$$

which is a projection on spherical harmonics in a component-by-component manner (for $m = -\ell, \cdots, \ell$). With ℓ we denote the spherical harmonic order and $r \in \mathbb{R}_{\geq 0}$ is the radius of the spherical fraction of the image that is projected onto a spherical harmonic function. Note, that we assume that the Dirac function has been normalized by $1/r^2$, thus $\int_{\mathbf{r}' \in \mathbb{R}^3} \delta_r(\|\mathbf{r}'\|) d\mathbf{r}' = \int_{\mathbf{r}' \in \mathbb{R}^3} \delta(\|\mathbf{r}'\| - r) d\mathbf{r}' = 4\pi$ is just the area on the unit sphere.

We regard a projection onto $\mathbf{Y}^\ell$ as local feature extraction $F : L_2(\mathbb{R}^3) \to \mathbb{C}^{(2\ell+1)}$, with

$$F(I,\ell,r) := \langle I, \overline{\mathbf{Y}^\ell} \delta_r \rangle \quad . \tag{2.73}$$

The features are complex valued vectors which are spherical harmonic expansion coefficients. This local feature extraction is covariant to 3D rotation.

Theorem 2.2.12 ($F(I,\ell,r)$ is covariant to $SO(3)$)**.** *F is covariant in the sense that F transforms via matrix multiplication with the Wigner D-matrices according to*

$$F(\mathfrak{g}I,\ell,r) = \mathbf{D}^\ell_{\mathfrak{g}} F(I,\ell,r) \quad . \tag{2.74}$$

(This theorem is a consequence of Thm. 2.2.8. A proof can be found on page 197)

Since any $SO(3)$ covariant local feature extraction induces an $SE(3)$ covariant filter, we use $F(I,\ell,r)$ to form an $SE(3)$ covariant filter (see (1.8) on page 24). We denote this filter by $\mathcal{SH}^\ell$ (Spherical Harmonic Filter). The filter $\mathcal{SH}^\ell$ is mapping images $I \in L_2(\mathbb{R}^3)$ to complex vector valued feature images. We denote the feature images by $\mathbf{a}^\ell(r) : \mathbb{R}^3 \to \mathbb{C}^{(2\ell+1)}$. The filter is defined by

$$\begin{aligned} \mathcal{SH}^\ell\{I,r\}(\mathbf{x}) &:= F(\mathfrak{t}_{(-\mathbf{x})}I,\ell,r) \\ &= \langle \mathfrak{t}_{(-\mathbf{x})}I, \overline{\mathbf{Y}^\ell} \delta_r \rangle \quad . \end{aligned} \tag{2.75}$$

With $\mathfrak{t}_{\mathbf{x}} \in T(3)$ we denote a translation acting on volumetric images, see example 2.1.14 on page 43. The feature images

$$\mathbf{a}^\ell(r) = \mathcal{SH}^\ell\{I,r\} \tag{2.76}$$

are the result of projection of the image onto spherical harmonics in a voxel-by-voxel manner. Therefore, the elements of $\mathbf{a}^\ell(r)$ are local expansion coefficients of an image I. We call $\mathbf{a}^\ell(\mathbf{x},r) \in \mathbb{C}^{2\ell+1}$ a spherical harmonic expansion coefficient of I with respect to a center of expansion $\mathbf{x} \in \mathbb{R}^3$ and a distance to the center of expansion $r \in \mathbb{R}_{\geq 0}$.

An image $I \in L_2(\mathbb{R}^3)$ can be recovered with respect to any center of expansion $\mathbf{x} \in \mathbb{R}^3$ via

$$\begin{aligned} I(\mathbf{r}) = I(\theta,\varphi,r) &= \sum_{\ell=0}^{\infty} \tfrac{2\ell+1}{4\pi} \sum_{m=-\ell}^{m=\ell} a^\ell_m(\mathbf{x}, \|\mathbf{r}-\mathbf{x}\|) \overline{Y^\ell_m(\mathbf{r}-\mathbf{x})} \\ &= \sum_{\ell} \tfrac{2\ell+1}{4\pi} \left(\mathbf{a}^\ell(\mathbf{x}, \|\mathbf{r}-\mathbf{x}\|)\right)^T \overline{\mathbf{Y}^\ell(\mathbf{r}-\mathbf{x})} \quad ; \end{aligned} \tag{2.77}$$

see Fig. 2.3 on the preceding page.

Since F is rotation covariant, the spherical harmonic transformation $\mathcal{SH}$ must be covariant to Euclidean motion; see Eq. (1.9) on page 24.

Theorem 2.2.13 (The Spherical Harmonic Transformation $\mathcal{SH}^\ell$ is $SE(3)$ Covariant)**.** *The spherical harmonic transform is $SE(3)$ covariant. Let $\mathfrak{g} \in SO(3)$ and $\mathfrak{t}_{\mathbf{t}} \in T(3)$). Then*

$$\mathcal{SH}^\ell\{\mathfrak{t}_{\mathbf{t}}\mathfrak{g}I,r\}(\mathbf{x}) = \mathbf{D}^\ell_{\mathfrak{g}} \mathcal{SH}^\ell\{I,r\}(\mathbf{U}^T_{\mathfrak{g}}(\mathbf{x}-\mathbf{t})) \quad . \tag{2.78}$$

(A proof can be found on page 197)

Accordingly we define the action of an element of the Euclidean motion group on the filter output.

Definition 2.2.14 (Group Action on Filter Output)**.** *The filter output is an image of spherical harmonic expansion coefficients* $\mathbf{a}^\ell(r) = \mathcal{SH}^\ell\{I,r\}$. *We define the group action on the coefficient image according to theorem 2.2.13. Let* $\mathfrak{g} \in SO(3)$ *and a translation* $\mathfrak{h}_{\mathbf{t}} \in T(3)$. *Then the coefficient image transforms according to*

$$
\begin{aligned}
(\mathfrak{h}_{\mathbf{t}}\mathfrak{g}\mathbf{a}^\ell)(\mathbf{x},r) &= \mathbf{D}^\ell_{\mathfrak{g}}\mathbf{a}^\ell(\mathbf{U}^T_{\mathfrak{g}}(\mathbf{x}-\mathbf{t}),r)\,, && (2.79)\\
(\mathfrak{g}\mathbf{a}^\ell)(\mathbf{x},r) &= \mathbf{D}^\ell_{\mathfrak{g}}\mathbf{a}^\ell(\mathbf{U}^T_{\mathfrak{g}}\mathbf{x},r) \text{ (rot. only).} && (2.80)
\end{aligned}
$$

Corollary 2.2.15. *[Rotation of a function in the harmonic domain] Let* $\mathbf{a}^\ell(\mathbf{x},r)$ *be the expansion coefficients of* $I : \mathbb{R}^3 \to \mathbb{C}$ *and* $\mathbf{x} \in \mathbb{R}^3$ *the center of expansion. Let further* $\mathfrak{g} \in SO(3)$ *be a rotation acting on* I *thus* $I'(\mathbf{r}) = (\mathfrak{g}I)(\mathbf{r}) := I(\mathbf{U}^T_{\mathfrak{g}}\mathbf{r})$ *is the rotated function. Then the corresponding rotation of* I *in frequency domain is just a rotation of the coefficients* $\mathbf{a}^\ell(\mathbf{x},r)$ *by the D-Wigner matrices, with*

$$
\begin{aligned}
(\mathfrak{g}I)(\mathbf{r}) &= \sum_\ell \tfrac{2\ell+1}{4\pi}\left(\mathbf{D}^\ell_{\mathfrak{g}}\mathbf{a}^\ell(\mathbf{U}^T_{\mathfrak{g}}\mathbf{x},\|\mathbf{r}-\mathbf{x}\|)\right)^T\overline{\mathbf{Y}^\ell(\mathbf{r}-\mathbf{x})}\\
&= \sum_\ell \tfrac{2\ell+1}{4\pi}\left(\mathbf{a}'^\ell(\mathbf{U}^T_{\mathfrak{g}}\mathbf{x},\|\mathbf{r}-\mathbf{x}\|)\right)^T\overline{\mathbf{Y}^\ell(\mathbf{r}-\mathbf{x})} \quad , && (2.81)
\end{aligned}
$$

where $\mathbf{a}'^\ell(\mathbf{x},r)$ *are the expansion coefficients of the rotated function.*

An $SE(3)$ **Covariant Spherical Harmonic Transformation**. The filter $\mathcal{SH}$ can be regarded as dense spherical harmonic transform, computing spherical harmonics expansion coefficients in a voxel-by-voxel manner. Such a transformation has been introduced by Ronneberger et al. (2005): a straight forward solution for realizing such a linear filter uses the convolution to make the projection computationally efficient. The basis functions are sliding over the image, similar to a filter and the expansion coefficients are computed in a voxel-by-voxel manner. The spherical harmonic transformation in Eq. (2.76) written as convolution is

$$
\begin{aligned}
\mathbf{a}^\ell_m(\mathbf{x},r) = \mathcal{SH}^\ell_m\{I,r\}(\mathbf{x}) &= \int_{\mathbf{r}'\in\mathbb{R}^3} I(\mathbf{r}'+\mathbf{x})Y^\ell_m(\mathbf{r}')\delta(\|\mathbf{r}'\|-r)d\mathbf{r}'\\
&= \int_{\mathbf{r}'\in\mathbb{R}^3} I(\mathbf{r}')Y^\ell_m(\mathbf{r}'-\mathbf{x})\delta(\|\mathbf{r}'-\mathbf{x}\|-r)d\mathbf{r}'\\
&= (-1)^\ell \int_{\mathbf{r}'\in\mathbb{R}^3} I(\mathbf{r}')Y^\ell_m(\mathbf{x}-\mathbf{r}')\delta(\|\mathbf{x}-\mathbf{r}'\|-r)d\mathbf{r}' \quad \text{(using SH-parity Eq. (2.68))}\\
&= (-1)^\ell(I * Y^\ell_m\delta_r)(\mathbf{x}) \quad . && (2.82)
\end{aligned}
$$

In practice, the convolution can be realized using a fast Fourier transform like the FFTW (Frigo and Johnson 2005), thus the coefficients can be computed in $O(N\log N)$, where N is the number of the voxels. We shortly write

$$
\begin{aligned}
\mathbf{a}^\ell(r) &= \mathcal{SH}^\ell\{I,r\}\\
&= (-1)^\ell(I * \mathbf{Y}^\ell\delta_r). && (2.83)
\end{aligned}
$$

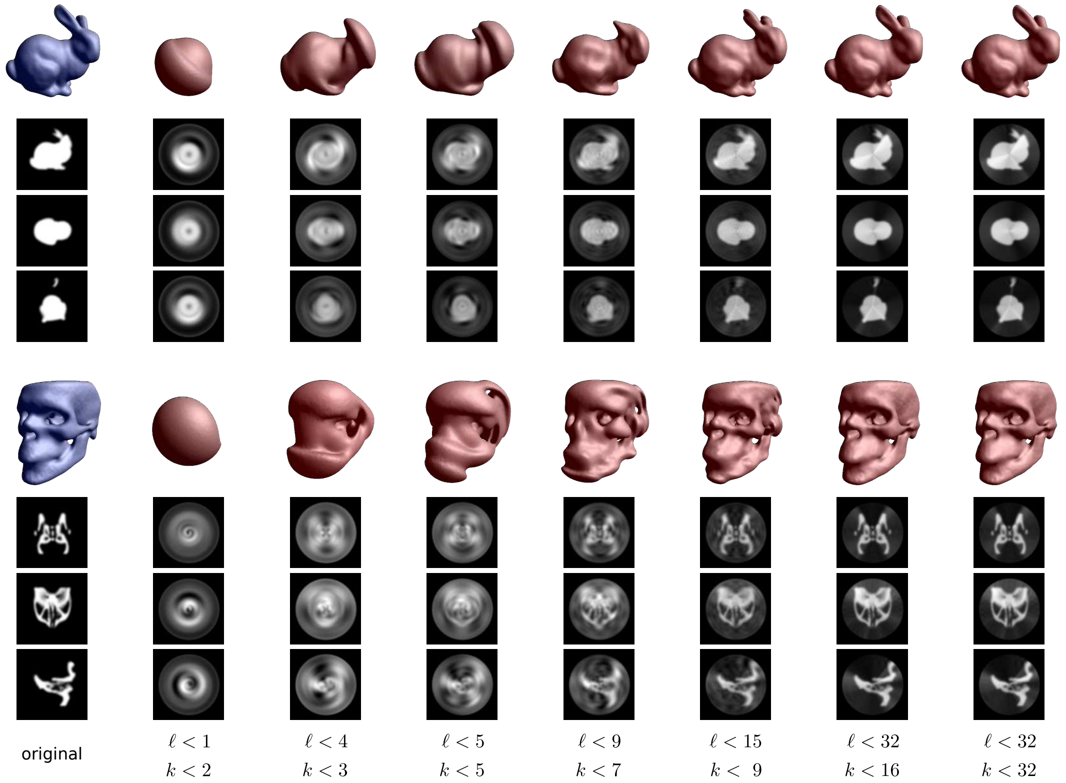


Figure 2.4.: Continuous radial functions are often better suited for image representation than nested spheres. In the continuous case, a small number of basis functions is usually sufficient to represent the important characteristics of an image in a global manner. Here a Fourier like radial profile is used, where $k \in \mathbb{N}$ denotes the frequency of the radial basis functions (Skibbe et al. 2009a).

It is worth mentioning that using nested spheres might be the most intuitive extension of the spherical harmonic approach to volumetric images, but there exist alternatives much better suited for representation (see e.g. Fig. 2.4) and efficient computation. There exist basis functions comprising orthogonal radial basis functions having a certain differential relationship. For instance, in section 5.1 and 5.2 we introduce *true* 3D functions inheriting all the properties of the spherical harmonics while allowing an efficient computation of the expansion coefficients via finite differences. Another example computes the local gradient orientation distribution via spherical harmonics instead of directly using the image's intensity values. We introduce such a filter in section 5.3.

2.2.3. Spherical Tensor Algebra

In this section we introduce the spherical tensor algebra utilizing the notations and definitions proposed by Reisert and Burkhardt (2009b). The available literature covering this topic is mainly related to physics and theoretical chemistry. To suggest some resources worth reading: Brown and Carrington (2003); Reisert and Skibbe

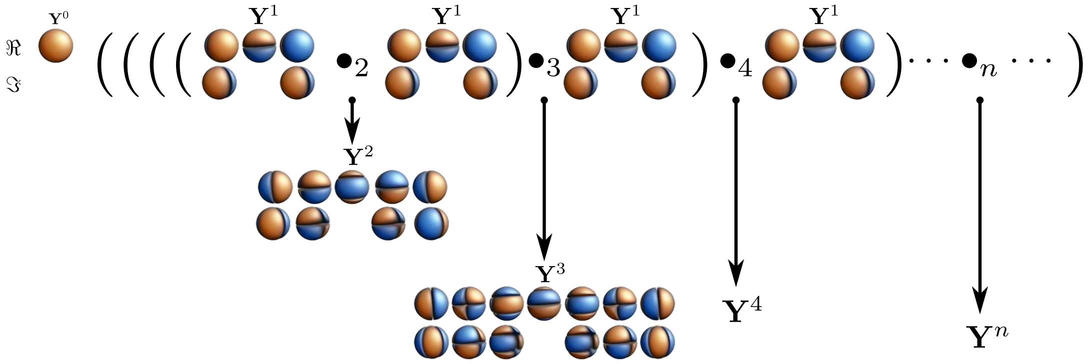


Figure 2.5.: Key property of $\bullet_\ell$ and $\mathbf{Y}$: higher order spherical harmonics $\mathbf{Y}^{\ell+1}$ can be obtained by coupling $\mathbf{Y}^\ell$ with $\mathbf{Y}^1$.

(2012); Skibbe and Reisert (2013); Tuszynski (1990) and the lecture notes of Wormer („Angular Momentum Theory“).

Definition 2.2.16 (Spherical Tensor Fields)**.** *We call a vector valued function* $\mathbf{f}^J : \mathbb{R}^3 \rightarrow \mathbb{C}^{2J+1}$*,* $\mathbf{f}^J \in \mathcal{T}_J$ *a spherical tensor field of rank* $J \geq 0$*, iff for all* $\mathfrak{g} \in SO(3)$

$$(\mathfrak{g}\mathbf{f}^J)(\mathbf{x}) := \mathbf{D}^J_{\mathfrak{g}}\mathbf{f}^J(\mathbf{U}^T_{\mathfrak{g}}\mathbf{x}) \quad . \tag{2.84}$$

We denote by $\mathcal{T}_J$ *the space of all spherical tensor fields of rank* J *(compare with Cartesian tensor fields, Thm. 2.2.1 on page 49. Here the rotation matrices are the irreducible representations of* $SO(3)$*).*

The vectors $\mathbf{f}^J(\mathbf{x}) \in \mathbb{C}^{(2J+1)}$ of a spherical tensor field $\mathbf{f}^J \in \mathcal{T}_J$ are called *spherical tensors* of order J. With $f^J_m : \mathbb{R}^3 \rightarrow \mathbb{C}$ we denote the single components of $\mathbf{f}^J$ such that $\mathbf{f}^J = (f^J_{-J},, \cdots, f^J_m, \cdots, f^J_J)^T$.

The following kinds of spherical tensor fields build the connection to volumetric images and therefore play an important role in this work:

- **Any scalar valued volumetric image is a tensor field of order 0** ($I \in \mathcal{T}_0$).
  This is the trivial case where for all $\mathfrak{g} \in SO(3) : \mathbf{D}^0_{\mathfrak{g}} = 1$:

  $$(\mathfrak{g}I)(\mathbf{x}) = I(\mathbf{U}^T_{\mathfrak{g}}\mathbf{x}) = \mathbf{D}^0_{\mathfrak{g}}I(\mathbf{U}^T_{\mathfrak{g}}\mathbf{x})\,, \tag{2.85}$$

  see Eq. (2.38) on page 49.

- $\mathbf{Y}^\ell$ **is a spherical tensor field of order** ℓ**.**
  The definition of the group action of rotations acting on spherical harmonics coincides with the definition for spherical tensor fields. The group action is defined by

  $$(\mathfrak{g}\mathbf{Y}^\ell)(\mathbf{x}) = \mathbf{D}^\ell_{\mathfrak{g}}\mathbf{Y}^\ell(\mathbf{U}^T_{\mathfrak{g}}\mathbf{x})\,, \tag{2.86}$$

  see Def. 2.2.7 on page 55. Hence the most natural spherical tensor fields rotating according to Eq. (2.84) are the spherical harmonic functions $\mathbf{Y}^\ell : \mathbb{R}^3 \rightarrow \mathbb{C}^{(2\ell+1)}$ themselves (Spherical Harmonics: Def. 2.2.6 on page 55).

- **The filter $\mathcal{SH}$ is mapping 3D images to spherical tensor fields**.
  That is, the dense field of spherical harmonic expansion coefficients $\mathbf{a}^\ell(r) : \mathbb{R}^3 \to \mathbb{C}^{2\ell+1}$ of an image $I \in L_2(\mathbb{R}^3)$ is a spherical tensor field of order ℓ. According to theorem 2.2.13, the expansion coefficients transform as
  $$(\mathfrak{g}\mathbf{a}^\ell)(\mathbf{x}, r) = \mathbf{D}^\ell_\mathfrak{g}\mathbf{a}^\ell(\mathbf{U}^T_\mathfrak{g}(\mathbf{x}), r); \tag{2.87}$$
  see Eq. (2.80) on page 60. That is, the filter $\mathcal{SH}$ is connecting 3D images to spherical tensor fields: any volumetric image $I \in L_2(\mathbb{R}^3)$ can be represented in terms of spherical tensor fields by locally expanding an image in terms of a spherical harmonic basis. We can write
  $$\mathcal{SH}^\ell : L_2(\mathbb{R}^3) \to \mathcal{T}_\ell \quad . \tag{2.88}$$

Spherical tensor fields are sharing some very basic properties with ordinary vector valued images.

Corollary 2.2.17 ($SE(3)$ Covariance of Spherical Tensor Fields)**.** *The translation of spherical tensor fields is defined as for ordinary vector valued images, thus Euclidean motion acts as follows*

$$(\mathfrak{h}_\mathbf{t}\mathfrak{g}\mathbf{f}^J)(\mathbf{r}) = (\mathfrak{g}\mathbf{f}^J)(\mathbf{r}-\mathbf{t}) = \mathbf{D}^\ell_\mathfrak{g}\mathbf{f}^J(\mathbf{U}^T_\mathfrak{g}(\mathbf{r}-\mathbf{t})) \quad . \tag{2.89}$$

We denote the Fourier representation of a spherical tensor field $\mathbf{f}^J \in \mathcal{T}_J$ by $\widetilde{\mathbf{f}^J} \in \mathcal{T}_J$. We obtain the Fourier representation of a spherical tensor field by transforming the field in a component by component way, where

$$\widetilde{f^J_m} = \mathcal{FT}\{f^J_m\} \quad . \tag{2.90}$$

We shortly write $\widetilde{\mathbf{f}^J} = \mathcal{FT}\{\mathbf{f}^J\}$.

2.2.3.1. Coupling of Spherical Tensors

The coupling of two Cartesian tensors is a common operation. For instance, gradient vectors can be coupled using the inner product leading to a tensor of rank 0 (a scalar), a cross product leads to a tensor of rank 1 (a vector) and an outer product as it occurs when computing the structure tensor creates a Cartesian tensor of order 2.

In spherical tensor algebra there exists a systematic way for coupling spherical tensors. Consider the inverse Clebsch-Gordan series[2] connecting spherical harmonics of different ranks. It is defined by

$$\sum_{m_3=m_1+m_2} \langle \ell_1 m_1, \ell_2 m_2 | \ell_3 m_3 \rangle Y^{\ell_1}_{m_1}(\mathbf{n}) Y^{\ell_2}_{m_2}(\mathbf{n}) = \langle \ell_1 0, \ell_2 0 | \ell_3 0 \rangle Y^{\ell_3}_{m_3}(\mathbf{n}). \tag{2.91}$$

The scalar valued coefficients $\langle \ell_1 m_1, \ell_2 m_2 | \ell m \rangle \in \mathbb{R}$ are weighting factors known as Clebsch-Gordan coefficients and are zero if $m_1 + m_2 \neq m$ (see section C.3.9 in the

---

[2] Wolfram function site, `http://functions.wolfram.com/05.10.23.0008.01`

appendix). The inverse Clebsch-Gordan series tells us how to couple spherical harmonics of order ℓ_1 and ℓ_2 to form a spherical harmonic of order ℓ_3. Thanks to the fix-point property of spherical harmonics ($(\mathfrak{g}\mathbf{Y}^\ell)(\mathbf{x}) = \mathbf{D}^\ell_\mathfrak{g}\mathbf{Y}^\ell(\mathbf{U}^T_\mathfrak{g}\mathbf{x})$, see Eq. (2.62)) we can conclude that the series respects rotations in the sense that

$$\begin{aligned}
&\langle \ell_1 0, \ell_2 0 \,|\, \ell_3 0\rangle Y^{\ell_3}_{m_3}(\mathbf{U}_\mathfrak{g}\mathbf{n}) = \\
&= \sum_{m_3 = m_1 + m_2} \langle \ell_1 m_1, \ell_2 m_2 \,|\, \ell_3 m_3\rangle Y^{\ell_1}_{m_1}(\mathbf{U}_\mathfrak{g}\mathbf{n}) Y^{\ell_2}_{m_2}(\mathbf{U}_\mathfrak{g}\mathbf{n}) \\
&= \sum_{m_3 = m_1 + m_2} \langle \ell_1 m_1, \ell_2 m_2 \,|\, \ell_3 m_3\rangle [\mathbf{D}^{\ell_1}_\mathfrak{g}\mathbf{Y}^{\ell_1}(\mathbf{n})]_{m_1} [\mathbf{D}^{\ell_2}_\mathfrak{g}\mathbf{Y}^{\ell_2}(\mathbf{n})]_{m_2} \\
&= \langle \ell_1 0, \ell_2 0 \,|\, \ell_3 0\rangle [\mathbf{D}^{\ell_1}_\mathfrak{g}\mathbf{Y}^{\ell_3}(\mathbf{n})]_{m_3} \,.
\end{aligned} \tag{2.92}$$

Moreover, since any spherical tensor rotates with respect to the Wigner D-matrices in the same way, we can build the following definitions upon Eq. (2.91).

Definition 2.2.18 (Tensor Products). *For the tensor rank $\ell \in \mathbb{N}$ we define the bilinear forms $\circ_\ell : \mathbb{C}^{2\ell_1+1} \times \mathbb{C}^{2\ell_2+1} \to \mathbb{C}^{2\ell+1}$ for coupling two spherical tensors of rank $\ell_1 \geq 0$ and $\ell_2 \geq 0$ to create a new tensor of rank ℓ. The products are defined by*

$$[(\mathbf{v} \circ_\ell \mathbf{w})]_m := \sum_{\substack{m= \\ m_1+m_2}} \langle \ell_1 m_1, \ell_2 m_2 \,|\, \ell m\rangle v_{m_1} w_{m_2} \,, \tag{2.93}$$

where $[(\mathbf{v} \circ_\ell \mathbf{w})]_m$ denotes the m-th component of the spherical tensor $(\mathbf{v} \circ_\ell \mathbf{w})$. These tensor products only exist for those triples out of $\ell_1, \ell_2, \ell \in \mathbb{N}$ that satisfy the triangle inequality $|\ell_1 - \ell_2| \leq \ell \leq \ell_1 + \ell_2$.

Two tensors are coupled in such a way that the rotation acting on the two tensors is preserved. This means, if a rotation $\mathfrak{g} \in SO(3)$ is acting on the two tensors $\mathbf{v} \in \mathbb{C}^{2\ell_1+1}$ of rank ℓ_1 and $\mathbf{w} \in \mathbb{C}^{2\ell_2+1}$ of rank ℓ_2, the rotation is also acting on the resulting tensor $(\mathbf{v} \circ_\ell \mathbf{w}) \in \mathbb{C}^{2\ell+1}$:

$$(\mathbf{D}^{\ell_1}_\mathfrak{g}\mathbf{v} \circ_\ell \mathbf{D}^{\ell_2}_\mathfrak{g}\mathbf{w}) = \mathbf{D}^{\ell}_\mathfrak{g}(\mathbf{v} \circ_\ell \mathbf{w}) \quad . \tag{2.94}$$

The tensor product obeys the following symmetry:

$$(\mathbf{v} \circ_\ell \mathbf{w}) = (-1)^{\ell_1+\ell_2-\ell}(\mathbf{w} \circ_\ell \mathbf{v}) \tag{2.95}$$

(According to the symmetries of Clebsch Gordan coefficients, Eq. C.28 in the appendix). Hence if ℓ is odd, then $(\mathbf{v} \circ_\ell \mathbf{v}) = \mathbf{0}$.

Definition 2.2.19 (Normalized Tensor Products). *The normalized tensor Products $\bullet_\ell : \mathbb{C}^{2\ell_1+1} \times \mathbb{C}^{2\ell_2+1} \to \mathbb{C}^{2\ell+1}$ exist for tensors of rank $\ell_1, \ell_2, \ell \in \mathbb{N}$ satisfying the triangle inequality $|\ell_1 - \ell_2| \leq \ell \leq \ell_1 + \ell_2$ and where additionally the sum $\ell + \ell_1 + \ell_2$ is even. The normalized spherical tensor products are defined by*

$$(\mathbf{v} \bullet_\ell \mathbf{w}) := \frac{(\mathbf{v} \circ_\ell \mathbf{w})}{\langle \ell_1 0, \ell_2 0 \,|\, \ell 0\rangle} \quad . \tag{2.96}$$

The spherical tensor products can be used to couple tensor fields of different rank by coupling in a point-by-point manner.

Proposition 2.2.20 (Coupling of Tensor Fields). *Let* $\mathbf{f}^{J_1} \in \mathcal{T}_{J_1}$ *and* $\mathbf{f}^{J_2} \in \mathcal{T}_{J_2}$ *be two spherical tensor fields of rank* J_1 *and* J_2*, respectively. Let further* J *be chosen according to the triangle inequality* $|J_1 - J_2| \leq J \leq J_1 + J_2$*, then*

$$\mathbf{f}^{J}(\mathbf{r}) = \mathbf{f}^{J_1}(\mathbf{r}) \circ_J \mathbf{f}^{J_1}(\mathbf{r}), \tag{2.97}$$

where $\mathbf{f}^J \in \mathcal{T}_J$*, i.e.* $\mathbf{f}^J$ *is a spherical tensor field of rank* J*. Accordingly, we can couple tensor fields with respect to the normalized tensor product* $\bullet_J$*.*

It can easily be verified that for a translation $\mathfrak{h}_{\mathbf{t}} \in T(3)$, the following holds:

$$(\mathfrak{h}_{\mathbf{t}} \mathbf{f}^{J_1}) \circ_\ell (\mathfrak{h}_{\mathbf{t}} \mathbf{f}^{J_2}) = \mathfrak{h}_{\mathbf{t}} (\mathbf{f}^{J_1} \circ_\ell \mathbf{f}^{J_2}) \quad . \tag{2.98}$$

Definition 2.2.21 (Tensor Convolution). *Let* $\mathbf{f}^{J_1} \in \mathcal{T}_{J_1}$ *and* $\mathbf{f}^{J_2} \in \mathcal{T}_{J_2}$ *be two spherical tensor fields of rank* J_1 *and* J_2*, respectively. Let further* J *be chosen according to the triangle inequality* $|J_1 - J_2| \leq J \leq J_1 + J_2$*, then we call*

$$\begin{aligned}(\mathbf{f}^{J_1} \widetilde{\circ}_J \mathbf{f}^{J_2})(\mathbf{x}) &:= \int_{\mathbf{r} \in \mathbb{R}^3} \mathbf{f}^{J_1}(\mathbf{r}) \circ_J \mathbf{f}^{J_2}(\mathbf{x} - \mathbf{r}) d\mathbf{r} \\ &= \int_{\mathbf{r}} \mathbf{f}^{J_1}(\mathbf{x} - \mathbf{r}) \circ_J \mathbf{f}^{J_2}(\mathbf{r}) d\mathbf{r} \end{aligned} \tag{2.99}$$

the spherical tensor convolution.

The spherical tensor convolution behaves similar to the ordinary convolution for scalar valued images, where

$$\mathbf{f}^{J_1} \widetilde{\circ}_\ell (\mathfrak{h}_{\mathbf{t}} \mathbf{f}^{J_2}) = (\mathfrak{h}_{\mathbf{t}} \mathbf{f}^{J_1}) \widetilde{\circ}_\ell \mathbf{f}^{J_2} = \mathfrak{h}_{\mathbf{t}} (\mathbf{f}^{J_1} \widetilde{\circ}_\ell \mathbf{f}^{J_2}) \quad . \tag{2.100}$$

This can easily be verified by using Eq. (2.89). Furthermore, similar to the ordinary scalar valued convolution we have the following convolution theorem.

Proposition 2.2.22 (Convolution Theorem). *The convolution theorem known for scalar valued functions can be transfered to the tensor convolution. In case of the tensor convolution we have*

$$(\mathbf{f}^{J_1} \widetilde{\circ}_J \mathbf{f}^{J_2})(\mathbf{x}) = \mathcal{F}\mathcal{T}^{-1}\{(\widetilde{\mathbf{f}}^{J_1} \circ_J \widetilde{\mathbf{f}}^{J_2})\}(\mathbf{x}) \, . \tag{2.101}$$

That is, a tensor convolution in frequency domain is a point-wise tensor product.

Proof: Convolution Theorem.

$$\begin{aligned}
[(\mathbf{f}^{J_1} \widetilde{\circ}_J \mathbf{f}^{J_2})]_m(\mathbf{x}) &= [\int_{\mathbf{r}} \mathbf{f}^{J_1}(\mathbf{x}-\mathbf{r}) \circ_J \mathbf{f}^{J_2}(\mathbf{r}) d\mathbf{r}]_m \\
&= \int_{\mathbf{r}} \sum_{\substack{m= \\ m_1+m_2}} \langle \ell_1 m_1, \ell_2 m_2 \,|\, \ell m \rangle f^{J_1}_{m_1}(\mathbf{x}-\mathbf{r}) f^{J_2}_{m_2}(\mathbf{r}) d\mathbf{r} \\
&= \sum_{\substack{m= \\ m_1+m_2}} \langle \ell_1 m_1, \ell_2 m_2 \,|\, \ell m \rangle \int_{\mathbf{r}} f^{J_1}_{m_1}(\mathbf{x}-\mathbf{r}) f^{J_2}_{m_2}(\mathbf{r}) d\mathbf{r} \\
&= \sum_{\substack{m= \\ m_1+m_2}} \langle \ell_1 m_1, \ell_2 m_2 \,|\, \ell m \rangle (f^{J_1}_{m_1} * f^{J_2}_{m_2})(\mathbf{x}) \\
&= \sum_{\substack{m= \\ m_1+m_2}} \langle \ell_1 m_1, \ell_2 m_2 \,|\, \ell m \rangle \mathcal{FT}^{-1}(\widetilde{f}^{J_1}_{m_1} \cdot \widetilde{f}^{J_2}_{m_2})(\mathbf{x}) \\
&= \mathcal{FT}^{-1} \left\{ \sum_{\substack{m= \\ m_1+m_2}} \langle \ell_1 m_1, \ell_2 m_2 \,|\, \ell m \rangle \widetilde{f}^{J_1}_{m_1} \widetilde{f}^{J_2}_{m_2} \right\} (\mathbf{x}) \\
&= [\mathcal{FT}^{-1}\{(\widetilde{\mathbf{f}}^{J_1} \circ_J \widetilde{\mathbf{f}}^{J_2})\}]_m(\mathbf{x}) \quad . \qquad (2.102)
\end{aligned}$$

□

Similar to complex conjugation of complex numbers, there exists a similar operations that acts on spherical tensors. One characteristic of such an operation is that it is its own inverse, that is, applying it twice leads back to the function's input. Such a mapping is called an involution. We call the involution of tensors the *tensor conjugation*. The tensor conjugation will play an important role regarding memory and computation efficient implementation of the proposed algorithms. Similar to the complex conjugation, which can be used to split the complex numbers into a real and and imaginary space, the tensor conjugation induces a real and an imaginary tensor space. In the last section of this chapter we will talk about the details and will see that when working with ordinary, scalar valued biomedical images, we automatically work on either this real or imaginary tensor space. Thanks to such a representation we can save both memory and CPU time.

Definition 2.2.23 (The Spherical Tensor Conjugation). *For all spherical tensors* $\mathbf{v}^\ell \in \mathbb{C}^{2\ell+1}$ *exits a counterpart* $\left(\mathbf{v}^\ell\right)^\ddagger \in \mathbb{C}^{2\ell+1}$ *with*

$$\left(v^\ell_m\right)^\ddagger := (-1)^m \overline{v^\ell_{-m}} \quad . \qquad (2.103)$$

Theorem 2.2.24 (Properties of the Spherical Tensor Conjugation). *The spherical tensor conjugation obeys the following equations regarding inversion:*

$$\left(\left(\mathbf{v}^\ell\right)^\ddagger\right)^\ddagger = \mathbf{v}^\ell \quad , \qquad (2.104)$$

and rotation:

$$\left(\mathbf{D}^\ell_{\mathfrak{g}} \mathbf{v}^\ell\right)^\ddagger = \mathbf{D}^\ell_{\mathfrak{g}} \left(\mathbf{v}^\ell\right)^\ddagger \quad . \qquad (2.105)$$

Consequently, $\mathbf{v}^\ell \in \mathbb{C}^{2\ell+1}$ *and* $\left(\mathbf{v}^\ell\right)^\ddagger \in \mathbb{C}^{2\ell+1}$ *have the same rotation behavior.*

Proof: Properties of the Spherical Tensor Conjugation. We first show Eq. (2.104) by two times applying the definition of the tensor conjugation:

$$\left(\left(v^{\ell}_{m}\right)^{\ddagger}\right)^{\ddagger} = \left((-1)^{m}\overline{v^{\ell}_{-m}}\right)^{\ddagger} = (-1)^{2m}\overline{\overline{v^{\ell}_{m}}} = v^{\ell}_{m} \quad . \tag{2.106}$$

For Eq. (2.105) we utilize the parity of the Wigner D-matrices (Eq (C.40) in the appendix) and consider the m-th component of the tensor. Then we have

$$\begin{aligned}\left(\sum_{n}[\mathbf{D}^{\ell}_{\mathfrak{g}}]_{m,n}v^{\ell}_{n}\right)^{\ddagger} &= (-1)^{m}\overline{\sum_{n}[\mathbf{D}^{\ell}_{\mathfrak{g}}]_{(-m),n}v^{\ell}_{n}} = (-1)^{m}\sum_{n}\overline{[\mathbf{D}^{\ell}_{\mathfrak{g}}]_{(-m),n}v^{\ell}_{n}}\\ &= (-1)^{m}\sum_{n}(-1)^{m-n}[\mathbf{D}^{\ell}_{\mathfrak{g}}]_{m,(-n)}\overline{v^{\ell}_{n}} = \sum_{n}[\mathbf{D}^{\ell}_{\mathfrak{g}}]_{m,n}(-1)^{n}\overline{v^{\ell}_{-n}} = \sum_{n}[\mathbf{D}^{\ell}_{\mathfrak{g}}]_{m,n}\left(v^{\ell}_{n}\right)^{\ddagger} \,.\end{aligned} \tag{2.107}$$

□

Corollary 2.2.25 (Tensor Products)**.** *Given two spherical tensors* $\mathbf{v}^{\ell_1} \in \mathbb{C}^{2\ell+1}$ *and* $\mathbf{w}^{\ell_2} \in \mathbb{C}^{2\ell+2}$*, we have the following identity*

$$(\left(\mathbf{v}^{\ell_1}\right)^{\ddagger} \circ_{\ell} \left(\mathbf{w}^{\ell_2}\right)^{\ddagger}) = (-1)^{(\ell_1+\ell_2+\ell)}\left(\mathbf{v}^{\ell_1} \circ_{\ell} \mathbf{w}^{\ell_2}\right)^{\ddagger} \quad . \tag{2.108}$$

Proof: Tensor Products.

$$\begin{aligned}[(\left(\mathbf{v}^{\ell_1}\right)^{\ddagger} \circ_{\ell} \left(\mathbf{w}^{\ell_2}\right)^{\ddagger})]_{m} &= \sum_{m_1,m_2}\left(v^{\ell_1}_{m_1}\right)^{\ddagger}\left(w^{\ell_2}_{m_2}\right)^{\ddagger}\langle \ell_1 m_1, \ell_2 m_2 \,|\, \ell m\rangle\\ &= \sum_{m_1,m_2}(-1)^{m_1}\overline{v^{\ell_1}_{-m_1}}(-1)^{m_2}\overline{w^{\ell_2}_{-m_2}}\langle \ell_1 m_1, \ell_2 m_2 \,|\, \ell m\rangle\\ &= (-1)^{m}\overline{\sum_{m_1,m_2} v^{\ell_1}_{-m_1}w^{\ell_2}_{-m_2}\langle \ell_1 m_1, \ell_2 m_2 \,|\, \ell m\rangle}\\ &= (-1)^{m}(-1)^{(\ell_1+\ell_2+\ell)}\overline{\sum_{m_1,m_2} v^{\ell_1}_{m_1}w^{\ell_2}_{m_2}\langle \ell_1 m_1, \ell_2 m_2 \,|\, \ell(-m)\rangle}\\ &= (-1)^{(\ell_1+\ell_2+\ell)}\left([\mathbf{v}^{\ell_1} \circ_{\ell} \mathbf{w}^{\ell_2}]_{m}\right)^{\ddagger} \quad .\end{aligned} \tag{2.109}$$

□

Corollary 2.2.26 (Tensor Inner Product)**.** *Given two spherical tensors* $\mathbf{v}^{\ell}, \mathbf{w}^{\ell} \in \mathbb{C}^{2\ell+1}$*. then we have the following identity*

$$(\mathbf{v}^{\ell} \bullet_{0} \mathbf{w}^{\ell}) = \langle \mathbf{v}^{\ell}, \left(\mathbf{w}^{\ell}\right)^{\ddagger}\rangle \quad , \tag{2.110}$$

where $\langle \cdot, \cdot \rangle$ *denotes the standard inner product of vectors, where* $\langle \mathbf{v}^{\ell}, \mathbf{w}^{\ell}\rangle = (\mathbf{v}^{\ell})^{T}\overline{\mathbf{w}}$*.*

Proof: Tensor Inner Product.

$$(\mathbf{v}^{\ell} \bullet_{0} \mathbf{w}^{\ell}) = \sum_{m}(-1)^{m}v^{\ell}_{m}w^{\ell}_{(-m)} = \sum_{m}v^{\ell}_{m}\overline{\left(w^{\ell}_{m}\right)^{\ddagger}} = \langle \mathbf{v}^{\ell}, \left(\mathbf{w}^{\ell}\right)^{\ddagger}\rangle \,. \tag{2.111}$$

□

2.2.3.2. The Angular Power- and Bi-Spectrum

The angular power spectrum is the Fourier representation of the auto-correlation of a functions on $SO(3)$ (see section B.2.3.1 on page 198), where the correlation parameters are the three Euler angles. The angular power spectrum represents a signals power with respect to a given angular frequency ℓ. The angular cross-spectrum is the Fourier representation of the angular cross-correlation, which is the correlation between two signals with respect to the three Euler angles. Therefore, it gives us information about the similarity of two signals with respect to a given angular frequency ℓ. In the following we consider the special case for functions on the sphere.

Theorem 2.2.27 (Angular Power- and Cross-spectrum)**.** *Given two functions* $f^1, f^2 \in L_2(S_2)$*. Then the angular cross- (if* $f^1 = f^2$ *power-) spectrum is defined by*

$$c^{\ell}(f_1, f_2) = \sum_m v^{\ell}_m \overline{w^{\ell}_m} = \langle \mathbf{v}^{\ell}, \mathbf{w}^{\ell} \rangle \,, \tag{2.112}$$

where $\mathbf{v}^{\ell}$ *and* $\mathbf{w}^{\ell}$ *are the spherical harmonic expansion coefficients of* f^1 *and* f^2*, respectively.*

The power spectrum completely looses any phase information of the signal. Hence in many applications it does not yield sufficient information for successfully analyzing and comparing signals. Therefore, in application where the phase information contains indispensable information, higher order spectra like the bi-spectrum are required (Hasselmann and MacDonald 1963; Kim and Powers 1978). The angular bi-spectrum is the Fourier representation of the angular triple-correlation; see Eq. (B.15) in the appendix. For our purposes it is important that the angular bi-spectrum preserves phase information so that it represents significantly more details than the power-spectrum. Note that for functions on $SO(3)$ whose coefficients are regular matrices, the angular bi-spectrum allows for completely recovering the function up to a rotation (**complete** rotation invariants in 3D); see particularly theorem 3.2.4 in Kakarala (1992), and for further readings Kakarala (1993); Kakarala (2012); Kondor (2008).

Theorem 2.2.28 (Angular Bi-Spectrum)**.** *Given a function* $f \in L_2(S_2)$ *with its spherical harmonic expansion coefficients* $\mathbf{v}^{\ell}$*. Then the angular bi-spectrum* $c^{\ell_1,\ell_2}_m : L_2(S_2) \to \mathbb{C}$ *is defined by*

$$c^{\ell_1,\ell_2}_m(f) = \sum_{|\ell_1-\ell_2| \leq \ell \leq \ell_1+\ell_2} \langle \ell_1 m, \ell_2(-m) | \ell 0 \rangle \langle \mathbf{v}^{\ell}, (\mathbf{v}^{\ell_1} \circ_{\ell} \mathbf{v}^{\ell_2}) \rangle \,. \tag{2.113}$$

The derivation of theorem 2.2.27 and 2.2.28 can be found on page 198. For spherical functions, both the angular bi-spectrum as well as the angular power- or cross-spectrum are rotation invariant. Therefore, both are widely used to form rotation invariants utilized for detecting volumetric objects and structures in a rotation invariant manner; see chapter 3 on page 83 for further details regarding invariants.

2.2.3.3. Properties and Coupling Rules for Tensor Products

For spherical harmonics there exists the following relationship:

Proposition 2.2.29 (Coupling Spherical Harmonics)**.** *The normalized spherical tensor product can be used for deriving spherical harmonics of order ℓ by coupling two spherical harmonics of order ℓ_1 and ℓ_2 with*

$$\mathbf{Y}^{\ell} = \mathbf{Y}^{\ell_1} \bullet_{\ell} \mathbf{Y}^{\ell_2}. \tag{2.114}$$

(Concluded from the Clebsch Gordan series Eq. (2.91)*).*

In Fig. 2.5 on page 62 we illustrate how higher order spherical harmonics can be computed recursively.

Forming a new tensor of a specific rank based on coupling two spherical tensors is (despite the trivial change of order) a unique operation. However, in case of coupling more than two tensors, there often are many ways to form a new tensor of given rank. Particularly the scenario, where we couple three tensors, plays a key role in the 3D rotation invariant theory and thus becomes particularly important for our applications. The triple product is used for computing the bi-spectrum for functions on the sphere which is known to provide powerful rotation invariant features.

One big issue that has not been addressed so far is the fact that by combining three tensors in all possible ways we risk extracting information from images in a redundant manner. More precisely, assume we aim at creating new tensors of order J based on three given tensors. There are usually a lot of options to combine the three tensors via tensor products and obtaining a new tensor of rank J. But it turns out that more than two thirds of them can be created via linear combination of only one third of the possible products. That is, two third of all possible products form a set of linearly dependent tensors yielding no further information about the underlying image structure. This is, regarding computation time and memory usage, a scenario that is worth avoiding. In the following, we derive the rules which guide us to set of tensors that guarantees that no information has been missed, but lead to an up to 3 times smaller number of products.

Theorem 2.2.30 (Coupling Three Spherical Tensors)**.** *We have the following identity when coupling three spherical tensors $\mathbf{u}^{\ell_1} \in \mathbb{C}^{2\ell_1+1}$, $\mathbf{v}^{\ell_2} \in \mathbb{C}^{2\ell_2+1}$ and $\mathbf{w}^{\ell_3} \in \mathbb{C}^{2\ell_3+1}$ to form a tensor of rank J based on an intermediate rank $|\ell_1 - \ell_2| \leq L_{12} \leq \ell_1 + \ell_2$:*

$$\begin{aligned} &((\mathbf{u}^{\ell_1} \circ_{L_{12}} \mathbf{v}^{\ell_2}) \circ_J \mathbf{w}^{\ell_3}) = \\ &\sum_{L_{23}} (\mathbf{u}^{\ell_1} \circ_J (\mathbf{v}^{\ell_2} \circ_{L_{23}} \mathbf{w}^{\ell_3})) \sqrt{(2L_{12}+1)(2L_{23}+1)} (-1)^{\ell_1+\ell_2+\ell_3+J} \begin{Bmatrix} \ell_2 & \ell_1 & L_{12} \\ J & \ell_3 & L_{23} \end{Bmatrix} . \end{aligned} \tag{2.115}$$

With $\begin{Bmatrix} \ell_1 & \ell_2 & \ell_4 \\ J & \ell_3 & \ell_5 \end{Bmatrix} \in \mathbb{R}$ we denote the Wigner 6j-symbol (see section C.3.10 in the appendix) They are the weighting factors playing a role when coupling three spherical tensors. Theorem 2.2.30 tells us something about symmetries that exist when coupling three spherical tensors. By exchanging the coupling order of the three tensors we see that each of the following sets of tensors, $\{(\mathbf{u}^{\ell_1} \circ_J (\mathbf{v}^{\ell_2} \circ_{L_{23}} \mathbf{w}^{\ell_3}))\}_{\forall L_{23}}$, $\{(\mathbf{w}^{\ell_3} \circ_J (\mathbf{u}^{\ell_1} \circ_{L_{12}} \mathbf{v}^{\ell_2}))\}_{\forall L_{12}}$ and $\{(\mathbf{v}^{\ell_2} \circ_J (\mathbf{u}^{\ell_1} \circ_{L_{13}} \mathbf{w}^{\ell_3}))\}_{\forall L_{13}}$, can be formed via linear combination of tensors of only one of the remaining sets. That is, they are mutually linearly dependent. This fact is illustrated in Fig. 2.6. That is, regarding the computation of linearly independent features it is sufficient (and essential) to compute only one

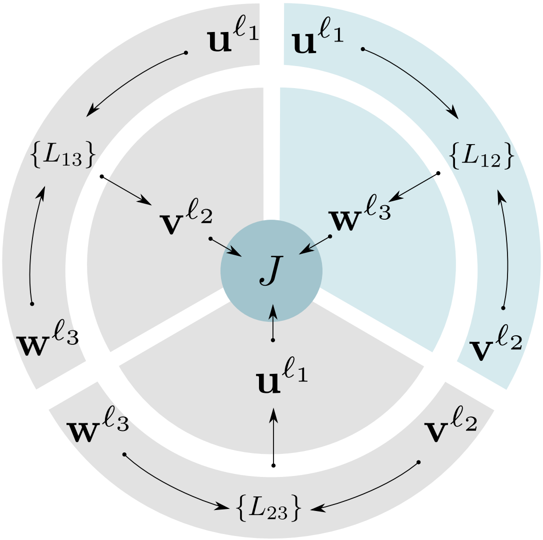


Figure 2.6.: (Coupling Three Spherical Tensors) The products defined in each third of this circle are spanning the tensor space of the products of the remaining two thirds. This means they are mutually linear dependent according to theorem 2.2.30.

set of features out of those three linearly dependent sets. We use this property for computing an linearly independent set of bi-spectrum features in our applications.

Proof. In the following we derive Eq. (2.115). According to Wormer („Angular Momentum Theory“), page 17, Eq. (90) there exists the recoupling rule

$$\begin{aligned}
&\sum_{M_{12}} \langle \ell_1 m_1, \ell_2 m_2 | L_{12} M_{12} \rangle \langle L_{12} M_{12}, \ell_3 m_3 | JM \rangle \\
&= \sum_{L_{23}, M_{23}} \sqrt{(2L_{12}+1)(2L_{23}+1)} W(\ell_1 \ell_2 J \ell_3, L_{12} L_{23}) \\
&\quad \times \langle \ell_1 m_1, L_{23} M_{23} | JM \rangle \langle \ell_2 m_2, \ell_3 m_3 | L_{23} M_{23} \rangle \qquad ,
\end{aligned} \tag{2.116}$$

where $W(\ell_1\ell_2 J\ell_3, L_{12}L_{23}) \in \mathbb{R}$ is a Racah W-coefficient (Racah 1942). Moreover, the following relation to the Wigner 6j-symbols is known (see e.g. Wormer („Angular Momentum Theory“), page 17, Eq. (93) and Eq. (94))

$$\begin{aligned}
\begin{Bmatrix} \ell_3 & L_{12} & J \\ \ell_1 & L_{23} & \ell_2 \end{Bmatrix} &= (-1)^{\ell_1+\ell_2+\ell_3+J} W(\ell_1 \ell_2 J \ell_3, L_{12} L_{23}) \\
&= \begin{Bmatrix} \ell_2 & \ell_1 & L_{12} \\ J & \ell_3 & L_{23} \end{Bmatrix} \text{(using Eq. (C.35) in the appendix)} \qquad .
\end{aligned} \tag{2.117}$$

By writing out the tensor product of three spherical tensors and by substituting Eq.

(2.117) into Eq. (2.116), we can derive the equation in theorem 2.2.30, namely

$$\begin{aligned}
&[((\mathbf{u}^{\ell_1} \circ_{L_{12}} \mathbf{v}^{\ell_2}) \circ_J \mathbf{w}^{\ell_3})]_M = \\
&= \sum_{M=M_{12}+m_3} \langle L_{12}M_{12}, \ell_3 m_3 \,|\, JM\rangle \langle \ell_1 m_1, \ell_2 m_2 \,|\, L_{12}M_{12}\rangle u^{\ell_1}_{m_1} v^{\ell_2}_{m_2} w^{\ell_3}_{m_3} \\
&= \sum_{m_3} \sum_{M_{12}} \langle L_{12}M_{12}, \ell_3 m_3 \,|\, JM\rangle \langle \ell_1 m_1, \ell_2 m_2 \,|\, L_{12}M_{12}\rangle u^{\ell_1}_{m_1} v^{\ell_2}_{m_2} w^{\ell_3}_{m_3} \\
&= \sum_{m_3} \sum_{L_{23}M_{23}} \sum_{L_{23}} \sqrt{(2L_{12}+1)(2L_{23}+1)}(-1)^{\ell_1+\ell_2+\ell_3+J} \\
&\quad \times \begin{Bmatrix} \ell_2 & \ell_1 & L_{12} \\ J & \ell_3 & L_{23} \end{Bmatrix} \langle \ell_1 m_1, L_{23}M_{23} \,|\, JM\rangle \langle \ell_2 m_2, \ell_3 m_3 \,|\, L_{23}M_{23}\rangle u^{\ell_1}_{m_1} v^{\ell_2}_{m_2} w^{\ell_3}_{m_3} \\
&= \sum_{L_{23}} \sum_{m_3} \sum_{M_{23}} \langle \ell_1 m_1, L_{23}M_{23} \,|\, JM\rangle \langle \ell_2 m_2, \ell_3 m_3 \,|\, L_{23}M_{23}\rangle u^{\ell_1}_{m_1} v^{\ell_2}_{m_2} w^{\ell_3}_{m_3} \\
&\quad \times \sqrt{(2L_{12}+1)(2L_{23}+1)}(-1)^{\ell_1+\ell_2+\ell_3+J} \begin{Bmatrix} \ell_2 & \ell_1 & L_{12} \\ J & \ell_3 & L_{23} \end{Bmatrix} \\
&= \sum_{L_{23}} [(\mathbf{u}^{\ell_1} \circ_J (\mathbf{v}^{\ell_2} \circ_{L_{23}} \mathbf{w}^{\ell_3}))]_M \sqrt{(2L_{12}+1)(2L_{23}+1)}(-1)^{\ell_1+\ell_2+\ell_3+J} \begin{Bmatrix} \ell_2 & \ell_1 & L_{12} \\ J & \ell_3 & L_{23} \end{Bmatrix} \; .
\end{aligned} \tag{2.118}$$

□

A conclusion that we can draw from the coupling rule in theorem 2.2.30 is that the spherical tensor products show some kind of associativity.

Corollary 2.2.31 (Associativity of Tensor Products). *For the triple tensor products there exist three fundamental coupling rules. These rules are:*

(Upper Bound) *If $\ell_1, \ell_2, \ell_3 \in \mathbb{N}$, then*

$$\begin{aligned}
&((\mathbf{u}^{\ell_1} \circ_{(\ell_1+\ell_2)} \mathbf{v}^{\ell_2}) \circ_{(\ell_1+\ell_2+\ell_3)} \mathbf{w}^{\ell_3}) \\
=&((\mathbf{u}^{\ell_1} \circ_{(\ell_1+\ell_3)} \mathbf{w}^{\ell_3}) \circ_{(\ell_1+\ell_2+\ell_3)} \mathbf{v}^{\ell_2}) \\
=&((\mathbf{v}^{\ell_2} \circ_{(\ell_2+\ell_3)} \mathbf{w}^{\ell_3}) \circ_{(\ell_1+\ell_2+\ell_3)} \mathbf{u}^{\ell_1}) \; .
\end{aligned} \tag{2.119}$$

(Lower Bound) *If additionally $(\ell_3 - \ell_2 - \ell_1) \geq 0$, then*

$$\begin{aligned}
&((\mathbf{w}^{\ell_3} \circ_{(\ell_3-\ell_1)} \mathbf{u}^{\ell_1}) \circ_{(\ell_3-\ell_2-\ell_1)} \mathbf{v}^{\ell_2}) \\
=&((\mathbf{w}^{\ell_3} \circ_{(\ell_3-\ell_2)} \mathbf{v}^{\ell_2}) \circ_{(\ell_3-\ell_2-\ell_1)} \mathbf{u}^{\ell_1}) \\
=&((\mathbf{u}^{\ell_1} \circ_{(\ell_1+\ell_2)} \mathbf{v}^{\ell_2}) \circ_{(\ell_3-\ell_2-\ell_1)} \mathbf{w}^{\ell_3}) \; .
\end{aligned} \tag{2.120}$$

(Scalar) *If $|\ell_1 - \ell_2| \leq \ell_3 \leq \ell_1 + \ell_2$, then*

$$\begin{aligned}
&((\mathbf{u}^{\ell_1} \circ_{\ell_3} \mathbf{v}^{\ell_2}) \circ_0 \mathbf{w}^{\ell_3}) \\
=&((\mathbf{w}^{\ell_3} \circ_{\ell_2} \mathbf{u}^{\ell_1}) \circ_0 \mathbf{v}^{\ell_2}) \\
=&((\mathbf{v}^{\ell_2} \circ_{\ell_1} \mathbf{w}^{\ell_3}) \circ_0 \mathbf{u}^{\ell_1}) \; .
\end{aligned} \tag{2.121}$$

Note, that these cases are only the special cases in Eq. (2.115), where the sum has only one addend.

Proof. We show all three equalities using the recoupling rule Eq. (2.115). In the first scenario we have

$$\begin{aligned}
&((\mathbf{u}^{\ell_1} \circ_{(\ell_1+\ell_2)} \mathbf{v}^{\ell_2}) \circ_{(\ell_1+\ell_2+\ell_3)} \mathbf{w}^{\ell_3}) = (\mathbf{u}^{\ell_1} \circ_{(\ell_1+\ell_2+\ell_3)} (\mathbf{v}^{\ell_2} \circ_{(\ell_2+\ell_3)} \mathbf{w}^{\ell_3}))\\
&\underbrace{\times \sqrt{(2(\ell_1+\ell_2)+1)(2(\ell_2+\ell_3)+1)} \begin{Bmatrix} \ell_2 & \ell_1 & (\ell_1+\ell_2) \\ (\ell_1+\ell_2+\ell_3) & \ell_3 & (\ell_2+\ell_3) \end{Bmatrix}}_{=1 \quad \text{(According to Eq. (C.36) in the appendix)}}\\
&\overset{\text{(using Eq. (C.28) in the appendix)}}{=} (\mathbf{v}^{\ell_2} \circ_{(\ell_1+\ell_2+\ell_3)} (\mathbf{u}^{\ell_1} \circ_{\ell_1+\ell_3} \mathbf{w}^{\ell_3})) \quad ,
\end{aligned} \tag{2.122}$$

and in the second scenario

$$\begin{aligned}
&((\mathbf{u}^{\ell_1} \circ_{(\ell_1+\ell_2)} \mathbf{v}^{\ell_2}) \circ_{(\ell_3-\ell_2-\ell_1)} \mathbf{w}^{\ell_3}) = (\mathbf{u}^{\ell_1} \circ_{(\ell_3-\ell_2-\ell_1)} (\mathbf{v}^{\ell_2} \circ_{(\ell_3-\ell_2)} \mathbf{w}^{\ell_3}))\\
&\underbrace{\times \sqrt{(2(\ell_1+\ell_2)+1)(2(\ell_3-\ell_2)+1)} \begin{Bmatrix} \ell_2 & \ell_1 & (\ell_1+\ell_2) \\ (\ell_3-\ell_2-\ell_1) & \ell_3 & (\ell_3-\ell_2) \end{Bmatrix}}_{=1 \quad \text{(According to Eq. (C.38) in the appendix)}} \quad .
\end{aligned} \tag{2.123}$$

Similarly, the third case can be shown to be:

$$\begin{aligned}
&((\mathbf{u}^{\ell_1} \circ_{\ell_3} \mathbf{v}^{\ell_2}) \circ_0 \mathbf{w}^{\ell_3}) = (\mathbf{u}^{\ell_1} \circ_0 (\mathbf{v}^{\ell_2} \circ_{\ell_1} \mathbf{w}^{\ell_3}))\\
&\underbrace{\times (-1)^{\ell_1+\ell_2+\ell_3} \sqrt{(2\ell_1+1)(2\ell_3+1)} \begin{Bmatrix} \ell_2 & \ell_1 & \ell_3 \\ 0 & \ell_3 & \ell_1 \end{Bmatrix}}_{=1 \quad \text{(According to eq. (C.37))}}
\end{aligned} \tag{2.124}$$

□

2.2.3.4. Tensorial Harmonic Functions

Spherical harmonic functions are building an orthogonal basis for functions on the sphere. We have further seen how to utilize spherical harmonics to build an $SE(3)$ covariant filter for volumetric images. Similar to spherical harmonic functions, whose values on the sphere are scalars, there exists an extension to higher order tensor valued basis functions with similar properties. With such basis functions it is possible to generalize the spherical harmonic framework from simple scalar valued volumetric images to higher order spherical tensor fields. The corresponding basis functions are called tensorial harmonics (Reisert and Burkhardt 2008b).

Tensorial harmonics are orthogonal functions providing a complete orthogonal basis for representing spherical tensor fields. Tensorial harmonics can be regarded as an extension of spherical harmonics to tensor fields of higher rank. Any spherical tensor field can be expanded in the following manner: given a spherical tensor field $\mathbf{f}^J \in \mathcal{T}_J$, then

$$\mathbf{f}^J(\mathbf{r}) = \sum_{\ell=0}^{\infty} \sum_{j=-J}^{j=J} N_{\ell,j} \mathbf{v}_j^{\ell}(r) \circ_J \mathbf{Y}^{\ell}(\mathbf{r}) \quad , \tag{2.125}$$

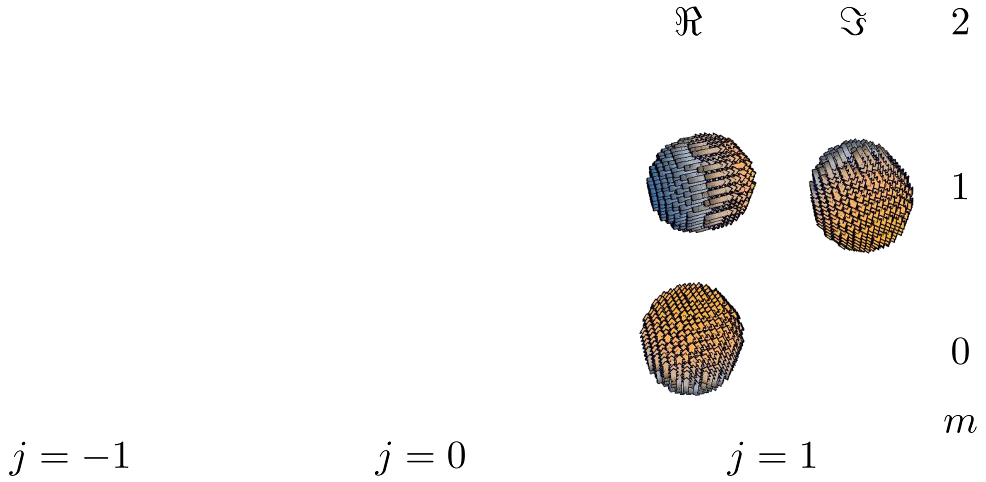


Figure 2.7.: Vectorial harmonics $\mathbf{Z}_1^{01m}$ for $m = 0, 1$ (negative components are not shown because of the internal symmetry). The vectorial harmonic associated with the spherical harmonic of order 0 consists of vectors aligned with respect to the Cartesian coordinate axes. Similar to the spherical harmonic of order 0 representing the mean value within an image, the first vectorial harmonic represents the mean direction within the vector field.

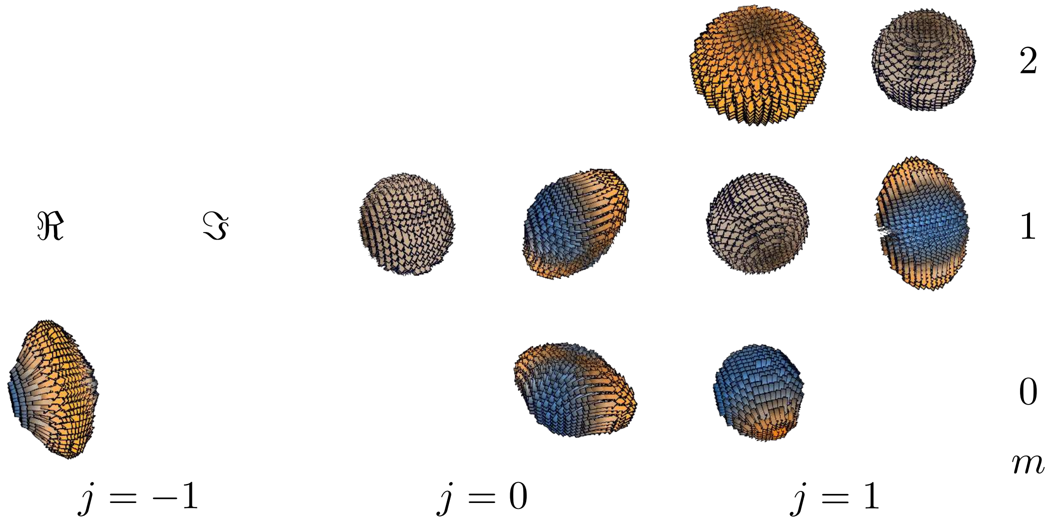


Figure 2.8.: Vectorial harmonics $\mathbf{Z}_1^{1jm}$ for $j = -1, 0, 1$ and $m = 0, \cdots, (j + 1)$ (negative components are not shown because of the internal symmetry).

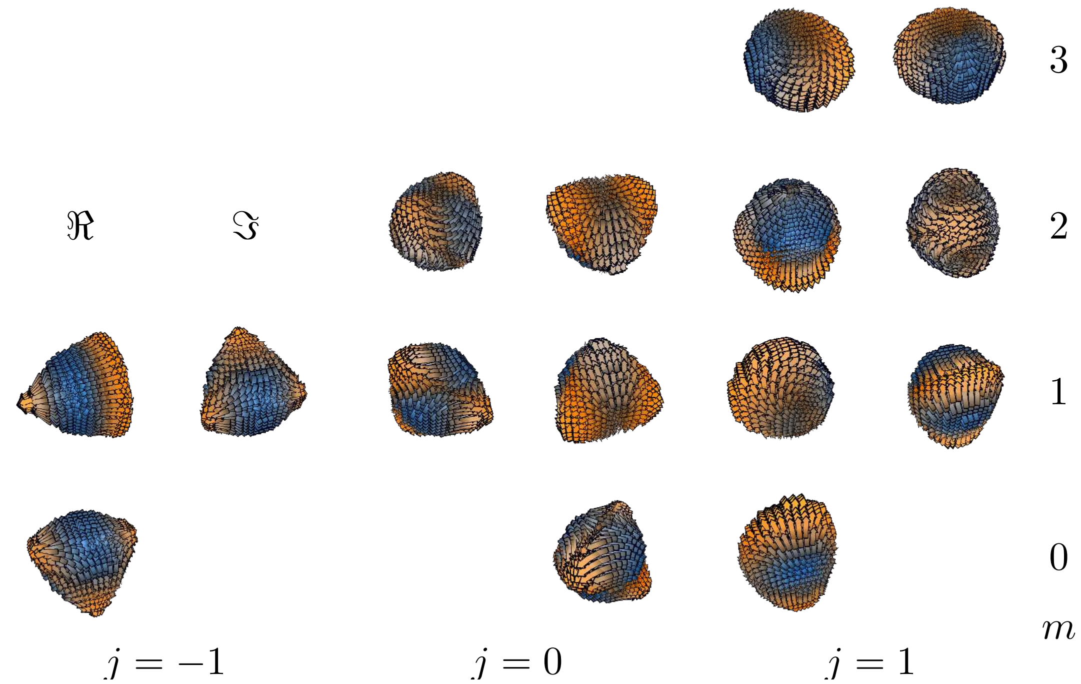


Figure 2.9.: Vectorial harmonics $\mathbf{Z}_1^{2jm}$ for $j=-1,0,1$ and $m=0,\cdots,(j+2)$ (negative components are not shown because of the internal symmetry).

where $\mathbf{v}_j^\ell(r) \in \mathbb{C}^{2(\ell+j)+1}$ are spherical tensor valued expansion coefficients, for the proof we refer to Reisert and Burkhardt (2008b).

Under rotations, the expansion coefficients transform similar to ordinary spherical harmonic expansion coefficients. Assume a rotation $\mathfrak{g} \in SO(3)$ is acting on the tensor field $\mathbf{f}^J$, then

$$\begin{aligned}(\mathfrak{g}\mathbf{f})(\mathbf{r}) = (\mathbf{D}_\mathfrak{g}^J\mathbf{f})(\mathbf{U}_\mathfrak{g}^T\mathbf{r}) &= \sum_{\ell=0}^{\infty}\sum_{j=-J}^{j=J} N_{\ell,j}\mathbf{D}_\mathfrak{g}^{\ell+j}\mathbf{v}_j^\ell(r) \circ_J \mathbf{D}_\mathfrak{g}^{\ell+j}\mathbf{v}_j^\ell\mathbf{Y}^\ell(\mathbf{U}_\mathfrak{g}^T\mathbf{r}) \\ &= \sum_{\ell=0}^{\infty}\sum_{j=-J}^{j=J} N_{\ell,j}\mathbf{D}_\mathfrak{g}^{\ell+j}\mathbf{v}_j^\ell(r) \circ_J \underbrace{\mathbf{Y}^\ell(\mathbf{r})}_{\text{fixpoint}} \quad ; \qquad (2.126)\end{aligned}$$

see the book chapter by Reisert and Burkhardt (2009b) for further details. That is, a rotation is just a transformation of the expansion coefficients via a Wigner D-matrix multiplication.

By rewriting the expansion Eq. (2.125) we identify the tensorial harmonic functions $\mathbf{Z}_J^{\ell jm}(\mathbf{r}) : \mathbb{R}^3 \to \mathbb{C}^{(2J+1)}$.

$$\begin{aligned}
\mathbf{f}^J(\mathbf{r}) &= \sum_{\ell=0}^{\infty} \sum_{j=-J}^{j=J} N_{\ell,j} \mathbf{v}_j^{\ell}(r) \circ_J \mathbf{Y}^{\ell}(\mathbf{r}) \qquad (2.127)\\
&= \sum_{\ell=0}^{\infty} \sum_{j=-J}^{j=J} N_{\ell,j} \sum_{m=-(\ell+j)}^{m=(\ell+j)} v_{jm}^{\ell}(r) \underbrace{\mathbf{e}_m^{\ell+j} \circ_J \mathbf{Y}^{\ell}(\mathbf{r})}_{:=\overline{\mathbf{Z}_J^{\ell jm}(\mathbf{r})}} = \sum_{\ell=0}^{\infty} \sum_{j=-J}^{j=J} N_{\ell,j} \sum_{m=-(\ell+j)}^{m=(\ell+j)} v_{jm}^{\ell}(r) \overline{\mathbf{Z}_J^{\ell jm}(\mathbf{r})} \,.
\end{aligned}$$

With $\mathbf{e}_m^{\ell+j} \in \mathbb{C}^{(2\ell+1)}$ we denote the standard basis, where $[\mathbf{e}_m^{\ell+j}]_n = \delta_{m,n}$. Hence the tensorial harmonic $\mathbf{Z}_J^{\ell jm} : \mathbb{R}^3 \to \mathbb{C}^{2J+1}$ is defined by

$$\mathbf{Z}_J^{\ell jm}(\mathbf{r}) := \mathbf{e}_m^{\ell+j} \circ_J \overline{\mathbf{Y}^{\ell}(\mathbf{r})} \quad . \qquad (2.128)$$

The single vector components are

$$Z_{JM}^{\ell jm}(\mathbf{r}) = \overline{Y_{(M-m)}^{\ell}} \langle (\ell+j)m, \ell(M-m) | JM \rangle \quad . \qquad (2.129)$$

The tensorial harmonics with rank J are building an orthogonal basis for spherical tensor fields of order J. The index $\ell \in \mathbb{N}$ is the corresponding spherical harmonic order, $j \in \mathbb{N}$, $j \in \{-J, \ldots, J\}$ the tensorial harmonic associated with a certain ℓ and the index $m \in \{-(\ell+j), \ldots, (\ell+j)\}$ denotes the mth tensorial harmonic associated with ℓ, j. Similar to spherical harmonics, the index m induces a group of functions representing a certain kind of local pattern, that is $\{\mathbf{Z}_J^{\ell j(-(\ell+j))}, \cdots, \mathbf{Z}_J^{\ell j(\ell+j)}\}$. However, a vector representation like for spherical harmonics is not possible (apart from $J = 0$) due to the vector valued image of tensorial harmonics.

For $J = 0$ the tensorial harmonics are identical (up to a factor) to the spherical harmonics, because

$$\mathbf{Z}_0^{\ell 0m}(\mathbf{n}) = \mathbf{e}_m^{\ell} \circ_0 \overline{\mathbf{Y}^{\ell}(\mathbf{n})} = (-1)^{\ell-m} \tfrac{1}{\sqrt{2\ell+1}} \overline{Y_{(-m)}^{\ell}(\mathbf{n})} = (-1)^{\ell} \tfrac{1}{\sqrt{2\ell+1}} Y_m^{\ell}(\mathbf{n}) \quad . \qquad (2.130)$$

The tensorial harmonics corresponding to $J = 1$ are widely known as vectorial harmonics or vector spherical harmonics (Fehr et al. 2009; Morse and Feshbach 1953; Skibbe et al. 2009b). Some vectorial harmonic functions are illustrated in Figs. 2.7,2.8,2.9. The tensorial harmonics are orthogonal functions on the tensor valued 2-sphere in the following sense:

$$\begin{aligned}
\langle \mathbf{Z}_J^{\ell jm}, \mathbf{Z}_J^{\ell' j' m'} \rangle &= \sum_{M=-J}^{J} \int_{S_2} Z_{JM}^{\ell jm}(\mathbf{n}) \overline{Z_{JM}^{\ell' j' m'}(\mathbf{n})} d\mathbf{n} \\
&= \frac{4\pi}{N_{\ell,j}} \delta_{\ell,\ell'} \delta_{j,j'} \delta_{m,m'} \,, \qquad (2.131)
\end{aligned}$$

where $N_{\ell,j} = \frac{(2\ell+1)(2(\ell+j)+1)}{2J+1}$ is a normalization factor.

Proposition 2.2.32 (Symmetry of Tensorial Harmonics). *The tensorial harmonics show a certain symmetry with respect to their order m and tensor component M which is*

$$Z_{JM}^{\ell jm}(\mathbf{n}) = (-1)^{(J+j+M+m)} \overline{Z_{J(-M)}^{\ell j(-m)}(\mathbf{n})} \quad . \qquad (2.132)$$

Proof: Tensorial Harmonic Symmetry. The symmetry is shown using the symmetry of spherical harmonics and Clebsch Gordan coefficients (Eq. (2.70) and Eq. (C.28) in the appendix):

$$\begin{aligned} Z^{\ell jm}_{JM}(\mathbf{n}) &= \overline{Y^{\ell}_{(M-m)}(\mathbf{n})}\langle(\ell+j)m,\ell(M-m)|JM\rangle \\ &= (-1)^{J+j}(-1)^{(M-m)}\overline{\overline{Y^{\ell}_{-(M-m)}(\mathbf{n})}}\langle(\ell+j)(-m),\ell(-(M-m))|J(-M)\rangle \\ &= (-1)^{J+j}(-1)^{(M-m)}\overline{Z^{\ell j(-m)}_{J(-M)}(\mathbf{n})}\,. \end{aligned} \tag{2.133}$$

□

The expansion coefficients are computed via orthogonal projection onto the tensorial harmonic basis functions. The inner product is the standard vector inner product so that the projection is done for all $m = -(\ell+j),\cdots,(\ell+j)$ components in a component-by-component manner with

$$v^{\ell}_{jm} := \langle \mathbf{f}^J, \mathbf{Z}^{\ell jm}_J \delta_r\rangle = \sum_{M=-J}^{J}\int_{\mathbb{R}^3} f^J_M(\mathbf{r}')\overline{Z^{\ell jm}_{JM}(\mathbf{r}')}\delta(\|\mathbf{r}'\|-r)d\mathbf{r}' \quad . \tag{2.134}$$

Under rotations, the expansion coefficients of a tensorial harmonic expansion undergo the same predictable transformation as spherical harmonic expansion coefficients. It is nearby to extend the spherical harmonic filter via tensorial harmonics to accept arbitrarily ranked spherical tensor fields as input:

2.2.3.5. An $SE(3)$ Covariant Tensorial Harmonic Transformation

Based on the tensorial harmonics, considered as a generalization of spherical harmonics, we extend the spherical harmonic transform $\mathcal{SH}: L_2(\mathbb{R}^3) \to \mathcal{T}_\ell$ in such a way that it accepts tensor valued images as input. We simply replace the spherical harmonics with tensorial harmonics and consider the case for scalar valued images as the standard case (because we mostly cope with scalar valued images).

We extend the local feature extraction $F(\ell,r): \mathcal{T}_0 \to \mathbb{C}^{2\ell+1}$ via tensorial harmonics to $F(\ell,j,r): \mathcal{T}_J \to \mathbb{C}^{2(\ell+j)+1}$, mapping spherical tensor fields to spherical tensor valued features. Similar to the spherical harmonic features, the features are expansion coefficients resulting from projecting a tensor field onto a family of tensorial harmonics $\{\mathbf{Z}^{\ell j(-(\ell+j))}_J,\cdots,\mathbf{Z}^{\ell j(\ell+j)}_J\}$ via

$$[F(\mathbf{f}^J,\ell,j,r)]_m := \langle \mathbf{f}^J, \overline{\mathbf{Z}^{\ell jm}_J}\delta_r\rangle \quad , \tag{2.135}$$

where $m = -(\ell+j),\cdots,(\ell+j)$. It holds that $F(\mathfrak{g}\mathbf{f}^J,\ell,j,r)$ is covariant to rotations in the sense that for any $\mathfrak{g} \in SO(3)$,

$$F(\mathfrak{g}\mathbf{f}^J,\ell,j,r) = \mathbf{D}^{(\ell+j)}_{\mathfrak{g}} F(\mathbf{f}^J,\ell,j,r) \quad . \tag{2.136}$$

The local feature extraction $F(\mathbf{f}^J,\ell,j,r)$ induces a new $SE(3)$ covariant filter $\mathcal{SH}^{\ell j}: \mathcal{T}_J \to \mathcal{T}_{(\ell+j)}$, where

$$\mathcal{SH}^{\ell j}\{\mathbf{f}^J,r\}(\mathbf{x}) := F(\mathfrak{t}_{(-\mathbf{x})}\mathbf{f}^J,\ell,j,r)\,. \tag{2.137}$$

The images

$$\mathbf{a}_j^\ell(r) := \mathcal{SH}^{\ell j}\{\mathbf{f}^J, r\}\,, \tag{2.138}$$

$\mathbf{a}_j^\ell(r) \in \mathcal{T}_{(\ell+j)}$ are images of local tensorial harmonic expansion coefficients.

Since efficient algorithms based on the convolution (see section 2.2.2.2 on page 57) and differentiation (see chapter 5 on page 109) exist for the "classical" spherical harmonic transform, it is an important fact that the tensor filter $\mathcal{SH}^{\ell j} : \mathcal{T}_J \to \mathcal{T}_{(\ell+j)}$ can be computed via its scalar valued counterpart $\mathcal{SH}^\ell : L_2(\mathbb{R}^3) \to \mathcal{T}_\ell$. In Skibbe et al. (2009b) we have shown how to compute the tensorial expansion coefficients in two steps in terms of an ordinary spherical harmonic transform: first, the expansion coefficients $\mathbf{a}_j^\ell(\mathbf{x}, r) \in \mathbb{C}^{2(\ell+j)+1}$ are computed by projecting a tensor field with its components $\mathbf{f}^J = \{f_{-J}^J, \ldots, f_M^J, \cdots, f_J^J\}$ into the space spanned by spherical harmonic functions in a component-by-component manner. Then, a weighted superposition forms the coefficients. This leads us to the following theorem.

Theorem 2.2.33 (Tensorial Harmonic Transformation)**.** *The tensorial harmonic transformation can be defined in terms of the spherical harmonic transformation, with*

$$\mathcal{SH}_m^{\ell j}\{\mathbf{f}^J, r\} = \sum_{M=-J}^{J} \sum_{n=-\ell}^{\ell} \underbrace{\mathcal{SH}_n^\ell\{f_M^J, r\}}_{\textit{standard SH trafo}} \; \underbrace{\langle (\ell+j)m, \ell n | JM \rangle}_{\textit{weights}}\,. \tag{2.139}$$

A proof can be found in section B.2.3.2 on page 200. An alternative suited for large images is based on tensor derivatives and will be discussed later in section 5.1.3.1 on page 125.

2.2.3.6. Solid Harmonic Functions

The (Schmidt semi-normalized) solid harmonics are widely known in physics as solutions of the Laplace equation in spherical coordinates (see e.g. Byerly (2003)). Let $\mathbf{R}_n^\ell \in \mathcal{T}_{\ell-n}$ be a spherical tensor field, whose $(2(\ell-n)+1)$ components $[\mathbf{R}_n^\ell]_m$ are defined by

$$[\mathbf{R}_n^\ell]_m(\mathbf{r}) := r^{\ell+n} Y_m^{\ell-n}(\mathbf{r}) \quad , \tag{2.140}$$

comprising a radial polynomial $r^{\ell+n} : \mathbb{R} \to \mathbb{R}$ with $\ell, n \in \mathbb{N}_0$ and Schmidt semi-normalized spherical harmonic functions $Y_m^{\ell-n}$, $m \in \mathbb{Z}, |m| \leq \ell - n$. We shortly write $\mathbf{R}^\ell$ instead of $\mathbf{R}_0^\ell$. The functions $\mathbf{R}^\ell$ are called solid harmonic functions.

Similar to spherical harmonics (See proposition (2.2.29) for the spherical harmonic case), solid harmonics can be coupled to form solid harmonics of higher or lower orders. They are coupled by multiplying their radial components, where

$$(\mathbf{R}_n^\ell \bullet_{(\ell+1)} \mathbf{R}^1) \quad = (\mathbf{Y}^{\ell-n} \bullet_{(\ell+1)} \mathbf{Y}^1) r^{\ell+n} r \quad = \mathbf{Y}^{(\ell+1)-n} r^{(\ell+1)+n} \quad = \mathbf{R}_n^{\ell+1} \quad \text{and} \tag{2.141}$$

$$(\mathbf{R}_n^\ell \bullet_{(\ell-1)} \mathbf{R}^1) \quad = (\mathbf{Y}^{\ell-n} \bullet_{(\ell-1)} \mathbf{Y}^1) r^{\ell+n} r \quad = \mathbf{Y}^{\ell-(n+1)} r^{\ell+(n+1)} \quad = \mathbf{R}_{n+1}^\ell \quad . \tag{2.142}$$

Therefore, we can compute $\mathbf{R}_n^\ell$ by successively coupling $\mathbf{R}^1$ in a recursive manner, where

$$\mathbf{R}_n^\ell = (\mathbf{R}^1 \bullet_{\ell-n} (\mathbf{R}^1 \bullet_{\ell-2} \cdots (\mathbf{R}^1 \bullet_{\ell-1} \underbrace{(\mathbf{R}^1 \bullet_\ell \cdots (\mathbf{R}^1 \bullet_2 (\mathbf{R}^1 \bullet_1 \mathbf{R}^1))))))}_{=\mathbf{R}^\ell}\,. \tag{2.143}$$

2.2.3.7. Connection Between Cartesian and Spherical Tensors

Since the rotations acting on Cartesian tensors are reducible matrix representations of $SO(3)$, there always exists a decomposition of Cartesian tensors into spherical counterparts: see end of section 2.2 on page 48. In this section we exemplarily show the connection between first and second order Cartesian tensors and their spherical counterparts (a tensor filed of order 0, a scalar valued image, is a Cartesian tensor field as well as a spherical tensor field). Details regarding this decomposition can be found in Reisert and Burkhardt (2009b). For a general discussion on this topic see the books of Rose (1995) or Brink and Satchler (1993).

The solid harmonics (see section 2.2.3.6 on the previous page for their Def.) induce a natural, unitary coordinate transform $\mathbf{S} \in SU(3)$, connecting Cartesian coordinates with spherical coordinates, where

$$\mathbf{S}\mathbf{x} = \mathbf{R}^1(\mathbf{x}) = \frac{1}{\sqrt{2}}\Big((x - iy), \sqrt{2}z, -(x+iy)\Big)^T \quad . \tag{2.144}$$

Solving this equations leads to

$$\mathbf{S} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -i & 0 \\ 0 & 0 & \sqrt{2} \\ -1 & -i & 0 \end{pmatrix} \quad . \tag{2.145}$$

Let $\mathbf{T}^1$ and $\mathbf{T}^2$ be two Cartesian tensor fields of order 1 and 2, respectively (see Eq. (2.40) on page 50). The field $\mathbf{T}^1$ is already irreducible. Therefore, $\mathbf{T}^1$ can directly be transformed into its spherical counterpart by the coordinate transform $\mathbf{S}$, acting on $\mathbf{T}^1$ in the following manner:

$$\mathbf{a}^1(\mathbf{x}) = \mathbf{S}\mathbf{T}^1(\mathbf{x}) \quad . \tag{2.146}$$

The second order tensor $\mathbf{T}^2$ is reducible. In order to use common matrix calculus we consider $\mathbf{T}^2$ w.l.o.g as tensor in matrix form. In this case

$$\mathbf{T}^2(\mathbf{x}) = \begin{pmatrix} T^2_{00} & T^2_{01} & T^2_{02} \\ T^2_{10} & T^2_{11} & T^2_{12} \\ T^2_{20} & T^2_{21} & T^2_{22} \end{pmatrix} \quad . \tag{2.147}$$

There exists a unique decomposition of $\mathbf{T}^2$ into three irreducible components:

$$\mathbf{T}^2 = \alpha \mathbf{I}_{3\times 3} + \mathbf{T}^2_{\text{anti}} + \mathbf{T}^2_{\text{sym}} \quad , \tag{2.148}$$

where $\mathbf{T}^2_{\text{anti}}$ is an antisymmetric matrix, $\mathbf{T}^2_{\text{sym}}$ a traceless ($\text{trace}\mathbf{T}^2_{\text{sym}} = 0$) symmetric matrix and $\alpha \in \mathbb{R}$ the trace. We get the corresponding three spherical tensors by: (1) applying the coordinate transform $\mathbf{S}$. We obtain

$$\mathbf{T'}^2(\mathbf{x}) = \mathbf{S}\mathbf{T}^2(\mathbf{x})\mathbf{S}^* \quad . \tag{2.149}$$

(2) We then obtain the spherical tensor fields $\mathbf{v}^\ell \in \mathcal{T}_\ell$, $\ell = 0, 1, 2$, corresponding to α, $\mathbf{T}^2_{\text{anti}}$ and $\mathbf{T}^2_{\text{sym}}$, according to the following spherical decomposition

$$v^\ell_m(\mathbf{x}) = \sum_{m=m_1+m_2} (-1)^{m_1} \langle 1m_1, 1m_2 | \ell m \rangle T'^2_{(1-m_1),(1+m_2)}(\mathbf{x}) \quad . \tag{2.150}$$

The inverse of this transformation is given by

$$T'^{2}_{(1+m_1),(1+m_2)}(\mathbf{x}) = \sum_{\ell=0}^{2} (-1)^{m_1} \langle 1(-m_1), 1m_2 \,|\, \ell m \rangle v^{\ell}_{m}(\mathbf{x}) \quad . \tag{2.151}$$

The explicit representations of $\mathbf{v}^{\ell}, \ell = 0, 1, 2$ are (see Skibbe et al. (2009b))

$$\mathbf{v}^0 = -\frac{\left(T^2_{00}+T^2_{11}+T^2_{22}\right)}{\sqrt{3}} \quad , \tag{2.152}$$

$$\mathbf{v}^1 = \begin{pmatrix} \frac{1}{2}\left(T^2_{20} - T^2_{02} + i(T^2_{21} - T^2_{12})\right) \\ \frac{i}{\sqrt{2}}\left(T^2_{10} - T^2_{01}\right) \\ \frac{1}{2}\left(T^2_{20} - T^2_{02} - i(T^2_{21} - T^2_{12})\right) \end{pmatrix} \quad \text{and} \tag{2.153}$$

$$\mathbf{v}^2 = \begin{pmatrix} \frac{1}{2}\left(T^2_{00} - T^2_{11} + i(T^2_{01} + T^2_{10})\right) \\ \frac{1}{2}\left((T^2_{02} + T^2_{20}) + i(T^2_{12} + T^2_{21})\right) \\ \frac{-1}{\sqrt{6}}\left(T^2_{00} + T^2_{11} - 2T^2_{22}\right) \\ \frac{1}{2}\left(-(T^2_{02} + T^2_{20}) + i(T^2_{12} + T^2_{21})\right) \\ \frac{1}{2}\left(T^2_{00} - T^2_{11} - i(T^2_{01} + T^2_{10})\right) \end{pmatrix} \quad . \tag{2.154}$$

2.2.3.8. Implementation: Benefiting from Real Valued Images

The space of complex numbers can be uniquely decomposed into a real part and an imaginary part which each forms a closed subspace under weighted superposition with real numbers. That is, if $v \in \mathbb{C}$ is a complex number, then $\mathfrak{Re}(v) = \frac{(v+\overline{v})}{2}$, $\mathfrak{Re}(v) \in \mathbb{R}$ is the real part of v and $\mathfrak{Im}(v) = \frac{(v-\overline{v})}{2}$, $\mathfrak{Im}(v) \in \mathbb{R}$ the imaginary part. Consequently, if our signal is either real or complex valued, it is sufficient to consider only one of these subspaces unless we restrict multiplications on real numbers. From a computational point of view, such a consideration is very beneficial: (1) we avoid complex multiplications which are computationally expensive and (2) we only need half the size of memory as for *true* complex numbers. The remaining parts of the complex signal would be zero anyway.

For spherical tensors there exists a similar consideration in terms of two complementary real and imaginary spherical tensor spaces. Despite the fact that they are complex valued, they behave similar to the real and imaginary space of complex numbers so that we can consider both as real valued vector spaces. This has some direct consequences for implementation: (1) the real and imaginary spherical tensor space needs each only half the amount of memory for storing the data. The same data structure can be used for both. (2) During this thesis we only work on real valued volumetric biomedical images whose elements are already part of a real spherical tensor space. (3) All spherical tensor operations we use during this thesis are mapping real or imaginary spherical tensors to real or imaginary spherical tensors. Therefore, we only have to track whether we are currently in the real or the imaginary space to never risk falling back into the *full* spherical tensor space. (4) Since we only need to store about half the amount of tensor components, all operations are almost twice as fast than when using a *full* spherical tensor representation.

Definition 2.2.34 (Real and Imaginary Tensor Space). *The tensor conjugation induces a unique decomposition of the spherical tensor space* $\mathbb{C}^{2\ell+1}$ *into two vector spaces* $\mathbb{V}_\ell, \mathbb{V}^i{}_\ell \subset \mathbb{C}^{2\ell+1}$. *Let* $\mathbf{v}^\ell \in \mathbb{C}^{2\ell+1}$, *then*

$$\mathbf{v}^\ell = \underbrace{\frac{(\mathbf{v}^\ell + \left(\mathbf{v}^\ell\right)^\ddagger)}{2}}_{\in \mathbb{V}_\ell} \oplus \underbrace{\frac{(\mathbf{v}^\ell - \left(\mathbf{v}^\ell\right)^\ddagger)}{2}}_{\in \mathbb{V}^i{}_\ell} . \tag{2.155}$$

Despite the fact that these vector spaces are complex valued we treat them as real valued vector spaces, because they have closer under weighted superposition for the real numbers. With this assumption, the spherical tensor space $\mathbb{C}^{2\ell+1}$ *is a direct sum of these two subspaces, that is* $\mathbb{C}^{2\ell+1} = \mathbb{V}_\ell \oplus \mathbb{V}^i{}_\ell$. *For the sake of consistency to standard complex numbers we call the vector space* $\mathbb{V}_\ell \subset \mathbb{C}^{2\ell+1}$ *the real spherical tensor space and* $\mathbb{V}^i{}_\ell \subset \mathbb{C}^{2\ell+1}$ *the imaginary spherical tensor space, i.e. we can always represent an* $\mathbf{v}^\ell \in \mathbb{V}^i{}_\ell$ *in terms of an* $i\mathbf{w}^\ell$, *where* $\mathbf{w}^\ell \in \mathbb{V}_\ell$ *(and verse visa). For further readings we suggest Brink and Satchler (1993); Reisert and Burkhardt (2009b); Rose (1995).*

Corollary 2.2.35 (Closed under Tensor Conjugation). *The tensor conjugation behaves similar to complex conjugation. If* $\mathbf{v}^\ell \in \mathbb{V}_\ell$ *and* $\mathbf{w}^\ell \in \mathbb{V}^i{}_\ell$ *then the tensor conjugation is closed:*

$$\begin{aligned} \left(\mathbf{v}^\ell\right)^\ddagger &= \mathbf{v}^\ell \quad \text{and} && \text{(real tensor space)} \\ \left(\mathbf{w}^\ell\right)^\ddagger &= -\mathbf{w}^\ell \quad . && \text{(imaginary tensor space)} \end{aligned} \tag{2.156}$$

(This can easily be shown by evaluating $\left((v^\ell_m + \left(v^\ell_m\right)^\ddagger)/2\right)^\ddagger$*)*

Theorem 2.2.36 (Basis of $\mathbb{V}_\ell$ and $\mathbb{V}^i{}_\ell$)). *Let the standard basis of the spherical tensor space* $\mathbb{C}^{2\ell+1}$ *be denoted by* $\{\mathbf{e}^\ell_m\}_{m=-\ell,\cdots,\ell}$, *where* $[\mathbf{e}^\ell_m]_n := \delta_{m,n}$. *Then a basis of* $\mathbb{V}_\ell$ *is* $\{\mathfrak{e}^\ell_m\}_{m=-\ell,\cdots,\ell}$, *where*

$$\mathfrak{e}^\ell_m = \mathbf{e}^\ell_m \tfrac{(1+i)}{2} + \left(\mathbf{e}^\ell_m \tfrac{(1+i)}{2}\right)^\ddagger \quad . \tag{2.157}$$

The basis of $\mathbb{V}^i{}_\ell$ *is* $\{\mathbf{e}^\ell_m \frac{(1+i)}{2} - \left(\mathbf{e}^\ell_m \frac{(1+i)}{2}\right)^\ddagger\}$.

Proof: Basis of $\mathbb{V}_\ell$. We can easily identify a basis of $\mathbb{V}^\ell$ and $\mathbb{V}^{i^\ell}$, respectively, by setting $\mathbf{v}^\ell = \mathbf{e}^\ell_m(1+i)$ thus Eq. (2.155) gives us

$$\begin{aligned} \mathbf{e}^\ell_m(1+i) &= \tfrac{(\mathbf{e}^\ell_m(1+i) + \left(\mathbf{e}^\ell_m(1+i)\right)^\ddagger)}{2} \oplus \tfrac{(\mathbf{e}^\ell_m(1+i) - \left(\mathbf{e}^\ell_m(1+i)\right)^\ddagger)}{2} \\ &= \tfrac{(\mathbf{e}^\ell_m(1+i) + (-1)^m \mathbf{e}^\ell_{-m}(1-i))}{2} \oplus \tfrac{(\mathbf{e}^\ell_m(1+i) - (-1)^m \mathbf{e}^\ell_{-m}(1-i))}{2} \\ &= \underbrace{\tfrac{(\mathbf{e}^\ell_m(1+i) + (-1)^m \mathbf{e}^\ell_{-m}(1-i))}{2}}_{=\mathfrak{e}^\ell_m} \oplus \tfrac{(\mathbf{e}^\ell_m(1+i) + (-1)^{m+1} \mathbf{e}^\ell_{-m}(1-i))}{2} \quad . \end{aligned} \tag{2.158}$$

□

Corollary 2.2.37 (Symmetry). *Let* $\mathbf{v}^\ell \in \mathbb{V}_\ell$ *and* $\mathbf{w}^\ell \in \mathbb{V}^i{}_\ell$. *The tensors* $\mathbf{v}^\ell$ *and* $\mathbf{w}^\ell$ *have the following symmetries*

$$v^\ell_m = (-1)^m \overline{v^\ell_{-m}} \quad \textit{and} \qquad \textit{(real tensor space)} \tag{2.159}$$

$$w^\ell_m = (-1)^{m+1} \overline{w^\ell_{-m}} \quad . \qquad \textit{(imaginary tensor space)} \tag{2.160}$$

This is a direct conclusion from the tensor conjugation property.

Corollary 2.2.38 (Closer under Tensor Products). *Tensor products of tensors of the space* $\mathbb{V}_\ell$ *and/or* $\mathbb{V}^i{}_\ell$ *are always resulting in tensors of* $\mathbb{V}_\ell$ *and* $\mathbb{V}^i{}_\ell$. *Moreover,* $\mathbb{V}_\ell$ *has closer under even products and* $\mathbb{V}^i{}_\ell$ *has closer under odd products.*

Let $\mathbf{v}^\ell_1 \in \mathbb{V}_{\ell_1}$ *and* $\mathbf{v}^\ell_2 \in \mathbb{V}_{\ell_2}$ *be both real spherical tensors, or* $\mathbf{v}^\ell_1 \in \mathbb{V}^i{}_{\ell_1}$ *and* $\mathbf{v}^\ell_2 \in \mathbb{V}^i{}_{\ell_2}$ *be both imaginary spherical tensors. Then*

$$\ell_1 + \ell_2 + \ell \text{ is even} \quad \Rightarrow \quad (\mathbf{v}^{\ell_1} \circ_\ell \mathbf{v}^{\ell_2}) \in \mathbb{V}_\ell \tag{2.161}$$

$$\ell_1 + \ell_2 + \ell \text{ is odd} \quad \Rightarrow \quad (\mathbf{v}^{\ell_1} \circ_\ell \mathbf{v}^{\ell_2}) \in \mathbb{V}^i{}_\ell \quad . \tag{2.162}$$

(A proof can be found in Reisert and Burkhardt (2009b)) The second scenario is the case where we mix tensors $\mathbf{v}^\ell_1 \in \mathbb{V}_{\ell_1}$ *and* $\mathbf{w}^\ell_2 \in \mathbb{V}^i{}_{\ell_2}$ *of both spaces. Then*

$$\ell_1 + \ell_2 + \ell \text{ is even} \quad \Rightarrow \quad (\mathbf{v}^{\ell_1} \circ_\ell \mathbf{w}^{\ell_2}) \in \mathbb{V}^i{}_\ell \tag{2.163}$$

$$\ell_1 + \ell_2 + \ell \text{ is odd} \quad \Rightarrow \quad (\mathbf{v}^{\ell_1} \circ_\ell \mathbf{w}^{\ell_2}) \in \mathbb{V}_\ell \quad . \tag{2.164}$$

Proof: Tensor Products: Mixed Case. Let $\mathbf{v}^\ell_1 \in \mathbb{V}_{\ell_1}$ and $\mathbf{w}^\ell_2 \in \mathbb{V}^i{}_{\ell_2}$. Then

$$\begin{aligned}
[(\mathbf{v}^{\ell_1} \circ_\ell \mathbf{w}^{\ell_2})]_m &= \sum_{m_1,m_2} v^{\ell_1}_{m_1} w^{\ell_2}_{m_2} \langle \ell_1 m_1, \ell_2 m_2 | \ell m \rangle \\
&= \sum_{m_1,m_2} (-1)^{m_1} \overline{v^{\ell_1}_{-m_1}} (-1)^{m_2+1} \overline{w^{\ell_2}_{-m_2}} \langle \ell_1 m_1, \ell_2 m_2 | \ell m \rangle \\
&= (-1)^{m+1} \overline{\sum_{m_1,m_2} v^{\ell_1}_{-m_1} w^{\ell_2}_{-m_2} \langle \ell_1 m_1, \ell_2 m_2 | \ell m \rangle} \\
&= (-1)^{m+1} (-1)^{(\ell_1+\ell_2+\ell)} \overline{\sum_{m_1,m_2} v^{\ell_1}_{-m_1} w^{\ell_2}_{-m_2} \langle \ell_1 (-m_1), \ell_2 (-m_2) | \ell (-m) \rangle} \\
&= (-1)^{m+1} (-1)^{(\ell_1+\ell_2+\ell)} \overline{\sum_{m_1,m_2} v^{\ell_1}_{m_1} w^{\ell_2}_{m_2} \langle \ell_1 m_1, \ell_2 m_2 | \ell (-m) \rangle} \\
&= (-1)^{m} (-1)^{(\ell_1+\ell_2+\ell+1)} \overline{[(\mathbf{v}^{\ell_1} \circ_\ell \mathbf{w}^{\ell_1})]_{(-m)}} \quad .
\end{aligned} \tag{2.165}$$

□

Corollary 2.2.39 (Tensor Inner Product). *Let* $\mathbf{v}^\ell_1 \in \mathbb{V}_{\ell_1}$ *and* $\mathbf{v}^\ell_2 \in \mathbb{V}_{\ell_2}$. *Then the inner product coincides with the normalized tensor product.*

$$(\mathbf{v}^\ell \bullet_0 \mathbf{w}^\ell) = \langle \mathbf{v}^\ell, \left(\mathbf{w}^\ell\right)^\ddagger \rangle = \langle \mathbf{v}^\ell, \mathbf{w}^\ell \rangle \quad . \tag{2.166}$$

(using Cor. 2.2.26 on page 67 and Cor. 2.2.35)

Corollary 2.2.40 (Spherical Harmonics are in $\mathbb{V}_\ell$)**.** *The spherical harmonics exist in the spherical tensor space $\mathbb{V}_\ell$. That is*

$$\mathbf{Y}^\ell(\mathbf{n}) \in \mathbb{V}_\ell \quad . \tag{2.167}$$

This is a direct conclusion of the symmetry of spherical harmonics; see Eq. (2.70). They obey the symmetry of $\mathbb{V}_\ell$: $Y^\ell_m(\mathbf{n}) = (-1)^m \overline{Y(\mathbf{n})^\ell_{-m}}$; see Eq. 2.159.

Corollary 2.2.41 (Biomedical Images are in $\mathbb{V}_0$)**.** *Let $I \in L_2(\mathbb{R}^3, \mathbb{R})$. A direct consequence of Eq. 2.159 is that $\mathbb{V}_0 = \mathbb{R}$ and therefore $I \in L_2(\mathbb{R}^3, \mathbb{V}_0)$.*

Corollary 2.2.42 (Spherical Harmonic Expansion Coefficients)**.** *Let $I \in L_2(\mathbb{R}^3, \mathbb{V}_0)$ be a biomedical image. The spherical harmonic expansion coefficients compute as*

$$\mathbf{a}^\ell(\mathbf{x}, r) = \langle \mathfrak{h}_{(-\mathbf{x})} I, \overline{\mathbf{Y}^\ell} \delta_r \rangle \quad ; \tag{2.168}$$

see Eq. (2.73) 59. Since in this case the inner product coincides with the normalized zero order tensor product (Cor . 2.2.39 on the preceding page) and the tensor product is even, the expansion coefficients are part of $\mathbb{V}_\ell$, that is, $\mathbf{a}^\ell(\mathbf{x}, r) \in \mathbb{V}_\ell$.

Corollary 2.2.43 (Tensorial Harmonic Expansion Coefficients)**.** *Let $\mathbf{f}^J \in \mathcal{T}_J$ be a spherical tensor field of order J. Let further be $\mathbf{f}^J : \mathbb{R}^3 \to \mathbb{V}_J$, that is, the image of $\mathbf{f}^J$ is in the real spherical tensor space. Then, for the tensorial harmonic expansion coefficients we have*

$$(J + j) \text{ is even} \qquad \Rightarrow \qquad \mathbf{a}^\ell_j(\mathbf{x}, r) \in \mathbb{V}_{(\ell+j)} \quad \text{and} \tag{2.169}$$

$$(J + j) \text{ is odd} \qquad \Rightarrow \qquad \mathbf{a}^\ell_j(\mathbf{x}, r) \in \mathbb{V}^i{}_{(\ell+j)} \quad . \tag{2.170}$$

That is, the tensorial harmonic expansion coefficients may contain both real and imaginary expansion coefficients.

Proof: Tensorial Harmonic Expansion Coefficients. Plugging the symmetric counterpart of the tensorial harmonics (Eq. (2.132)) into the tensorial harmonic projection (Eq. (B.18)) and using the interrelation of the real spherical tensor space (Eq. (2.159)) shows that

$$\begin{aligned}
a^\ell_{jm}(\mathbf{x}, r) &= \langle \mathfrak{h}_{(-\mathbf{x})} \mathbf{f}^J, \overline{\mathbf{Z}^{\ell jm}_J} \delta_r \rangle \\
&= \sum_{M=-J}^{J} \int_{\mathbb{R}^3} f^J_M(\mathbf{r}' + \mathbf{x}) \overline{Z^{\ell jm}_{JM}(\mathbf{r}')} \delta(\|\mathbf{r}'\| - r) d\mathbf{r}' \\
&= (-1)^{(J+j+m)} \sum_{M=-J}^{J} \int_{\mathbb{R}^3} (-1)^M \overline{f^J_{(-M)}(\mathbf{r}' + \mathbf{x})} (-1)^M \overline{\overline{Z^{\ell j(-m)}_{J(-M)}(\mathbf{r}')}} \delta(\|\mathbf{r}'\| - r) d\mathbf{r}' \\
&= (-1)^{(J+j+m)} \overline{\sum_{M=-J}^{J} \int_{\mathbb{R}^3} f^J_M(\mathbf{r}' + \mathbf{x}) \overline{Z^{\ell j(-m)}_{JM}(\mathbf{r}')} \delta(\|\mathbf{r}'\| - r) d\mathbf{r}'} \\
&= (-1)^{(J+j+m)} \overline{\langle \mathfrak{h}_{(-\mathbf{x})} \mathbf{f}^J, \overline{\mathbf{Z}^{\ell j(-m)}_J} \delta_r \rangle} = (-1)^{(J+j)+m} \overline{a^\ell_{j(-m)}(\mathbf{x}, r)} \, .
\end{aligned} \tag{2.171}$$

Therefore, depending on $J+j$, the coefficients fulfill either the symmetry condition of the real or the imaginary spherical tensor space (see Eq. (2.159) and Eq. (2.160)). □

3 Rotation Invariant Features

An $SE(3)$ covariant spherical harmonic transformation maps images to spherical tensor fields. The elements of such a field are expansion coefficients of local image patches. As a consequence of translation and/or rotation of the input image, the elements of the output field are transforming in a mathematically well defined and predictable manner. However, the features are not locally rotation invariant, that is, the single tensor elements "mix" under rotations. Our aim is to locally achieve rotation *invariance* thus there is no need for a classifier to cope with this issue. So we now introduce tools for systematically turning the covariant features into locally rotation invariant features via a technique called group integration.

The rotation invariant features that we propose here are obtained via an established group integration framework proposed by Schulz-Mirbach (1995a); see also Schulz-Mirbach (1996). This framework has explicitly been used in a variety of applications, ranging from 2D image analysis (Burkhardt and Siggelkow 2001; Schael 2002; Schulz-Mirbach 1995b), to 3D image analysis (Ronneberger et al. 2002; Schael and Siggelkow 2000). Recent works on 3D feature extraction are incorporating spherical harmonics into this framework, which allows an analytical computation of 3D rotation invariants. This includes i.e. the works of Reisert and Burkhardt (2006); Ronneberger (2007); Ronneberger et al. (2005). In this chapter we will see that regarding 3D rotations and spherical harmonics, the angular power-, cross- and bi-spectra can also be regarded as an invariant resulting from group integration. Hence the works of Fehr (2010); Fehr and Burkhardt (2006); Kazhdan et al. (2003); Liu et al. (2011b); Schnell et al. (2009); Skibbe et al. (2009b); Skibbe et al. (2010) can also be classed within this framework.

Before discussing the creation of invariant features, we need some preliminary definitions. Invariance is always associated with a transformation group G. Remember, a function is covariant to a transformation group G, if the output space transforms according to the input space (Def. 1.2.1 on page 18). We say a function is invariant to G, if the transformation in the output space is the identity function (Def. 1.2.2 on page 18). In this case, the output of a function is independent of the transformation acting on the input pattern. We call such an output an invariant feature of the input pattern, or shortly an invariant. The following definitions and notations are kept similar to those in previous work on this topic, see i.e. the dissertations of Fehr (2009); Reisert (2008); Ronneberger (2007); Schulz-Mirbach (1995a); Siggelkow (2002).

Given a pattern space P and a transformation group G. We call two patterns $\mathsf{p}, \mathsf{p}' \in P$ equivalent ($\mathsf{p} \sim_G \mathsf{p}'$), if there exists a transformation $\mathfrak{g} \in G$ together with a representation $\boldsymbol{\tau}^P$ mapping the first pattern into the second one, and verse visa. That is,

$$\mathsf{p} \sim_G \mathsf{p}' \quad \Leftrightarrow \quad \exists \mathfrak{g} \in G : \mathsf{p}' = \boldsymbol{\tau}^P_{\mathfrak{g}} \mathsf{p} \quad . \tag{3.1}$$

The definition of invariance can then be written in terms of equivalence.

Definition 3.0.44 (Invariance). *A function $F : P \to V$ is invariant to G, if $\forall \mathsf{p}, \mathsf{p}' \in P$:*

$$\mathsf{p} \sim_G \mathsf{p}' \quad \Rightarrow \quad F(\mathsf{p}') = F(\mathsf{p}) \quad . \tag{3.2}$$

Each pattern $\mathsf{p} \in P$ induces an equivalence class called an *orbit*.

Definition 3.0.45 (Orbits). *Each pattern p in a pattern space P induces an equivalece class called the orbit of p:*

$$\mathcal{O}^G[\mathsf{p}] = \{\mathsf{p}' \mid \mathsf{p} \sim_G \mathsf{p}'\} \, . \tag{3.3}$$

If $F : P \to V$ is invariant to G, then $\forall \mathsf{p}' \in \mathcal{O}^G[\mathsf{p}] : F(\mathsf{p}') = F(\mathsf{p})$. That is, all patterns map to the same point in the feature space V. Similar to orbits, we define the set of all patterns mapping into the same point in the feature space by $O^F[\mathsf{p}]$, where

$$\mathcal{O}^F[\mathsf{p}] = \{\mathsf{p}' \mid F(\mathsf{p}) = F(\mathsf{p}')\} \, . \tag{3.4}$$

A conclusion from Eq. (3.2) is that $\mathcal{O}^F[\mathsf{p}] \supseteq \mathcal{O}^G[\mathsf{p}]$ (and consequently $|\mathcal{O}^F[\mathsf{p}]| \geq |\mathcal{O}^G[\mathsf{p}]|$) , so, there might exist patterns that are not equivalent but map to the same feature. This is illustrated in Fig. 3.1 a). If each orbit can be uniquely identified by a feature, then F is called complete.

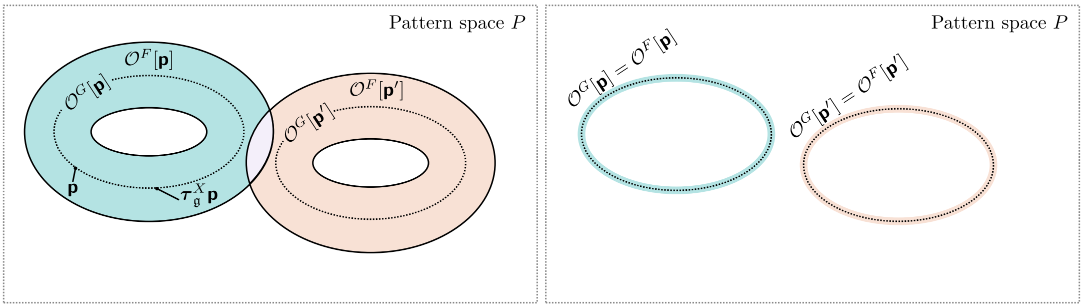
Definition 3.0.46 (Complete). *An invariant mapping F is called complete, if*

$$\forall \mathsf{p}, \mathsf{p}' \in P : \; \mathsf{p} \sim_G \mathsf{p}' \quad \Leftrightarrow \quad F(\mathsf{p}') = F(\mathsf{p}) \quad . \tag{3.5}$$

Consequently, if F is complete, then $\mathcal{O}^F[\mathsf{p}]$ coincides with $\mathcal{O}^G[\mathsf{p}]$, thus $|\mathcal{O}^F[\mathsf{p}]| = |\mathcal{O}^G[\mathsf{p}]|$; see Fig. 3.1 b). Completeness is a very strict property that guarantees a unique, distinctive feature for each orbit. However, completeness is in general hard to achieve. In practice, it is often sufficient to achieve completeness for a subset of likely patterns, that is, patterns that we expect to appear within a specific kind of problem. We say, the patterns are *separable* with respect to F (we can distinguish the likely occurring patterns with F).

3.1. Invariance via Group Integration

According to Burkhardt and Siggelkow (2001), the generation of invariants can be divided into three kinds of approaches: normalization, invariance via group integration and differentiation. The latter one is rarely used. Moreover, we are not aware of 3D rotation invariants for volumetric images that have been created via differentiation. The reason might be the fact that the differentiation framework requires the formulation and solution of differential equations depending on the group's parameters which are often hard to solve. Contrarily, there exist a (small) variety of 3D



a) Incomplete b) Completeness

Figure 3.1.: A function F that is invariant to a transformation G maps patterns to features (here patterns mapping to the same feature are within an ellipsoid: $\mathcal{O}^F[\mathbf{p}]$ blue(left ellipsoid) and $\mathcal{O}^F[\mathbf{p}']$ red(right ellipsoid), the two orbits $\mathcal{O}^G[\mathbf{p}]$ and $\mathcal{O}^G[\mathbf{p}']$ are depicted as dotted ellipsoids). b) F is complete, if we can conclude the orbit from the feature (here the color) and verse visa.

rotation invariants that are explicitly, or implicitly making use of the normalization or group integration frameworks. The idea of normalization is to bring a pattern into a distinctive pose before extracting features. That is, the creation of invariant features requires the normalization of patterns of an orbit to one specific representative, see Fig. 3.2 a). Normalization often depends on dominant features within the pattern (see. e.g. SIFT (Lowe 2004), where the local gradient main direction is used) and thus is often sensitive to noise, or only works at specific key-points. For our features, we recommend the group integration technique for turning the proposed spherical harmonic features into rotation invariants. See the Related Works section on page 27 for a survey on related works on these topics with focus on 3D rotations and volumetric images.

The idea of group integration means that we average over all patterns within an orbit. The resulting integral is invariant to the transformation spanning the orbit: given two patterns $\mathbf{p}, \mathbf{p}' \in P$. Let G be a compact group and $\boldsymbol{\tau}^P$ be a group representation of G acting on the pattern space P. If $\exists \mathfrak{g} \in G : \mathbf{p} = \boldsymbol{\tau}^P_{\mathfrak{g}}\mathbf{p}'$, then $\mathbf{p} \sim \mathbf{p}'$. We conclude that $\mathcal{O}^G[\mathbf{p}] = \mathcal{O}^G[\mathbf{p}']$. Consequently, integrating over these orbits would lead in both cases to the same result. Hence the integral

$$F(\mathbf{p}) := \frac{1}{|\mathcal{O}^G[\mathbf{p}]|} \int_{\mathcal{O}^G[\mathbf{p}]} \mathbf{p}' d\mathbf{p}' \overset{\text{(Orbits, Eq. (3.3))}}{=} \frac{1}{|G|} \int_G \boldsymbol{\tau}^P_{\mathfrak{g}} \mathbf{p} d\mathfrak{g} \tag{3.6}$$

is invariant to G, see Def. 2.1.10 on page 42. That is, averaging over all patterns within an orbit; see Fig. 3.2 b). An invariant obtained via Eq. (3.6) is often neither complete nor separable, because averaging is a simple linear operation. Schulz-Mirbach (1995a) proposed to overcome this limitation by introducing a (typically nonlinear) mapping within this framework that first extracts intrinsic features from patterns before integrating over the transformation group. Such a mapping is called a *kernel*.

Let $\mathcal{K}(\xi) : P \to \mathbb{C}$ be a kernel that extracts intrinsic features from a pattern. If G is a compact group and $\boldsymbol{\tau}^P$ a group representation of G, acting on the pattern space P,

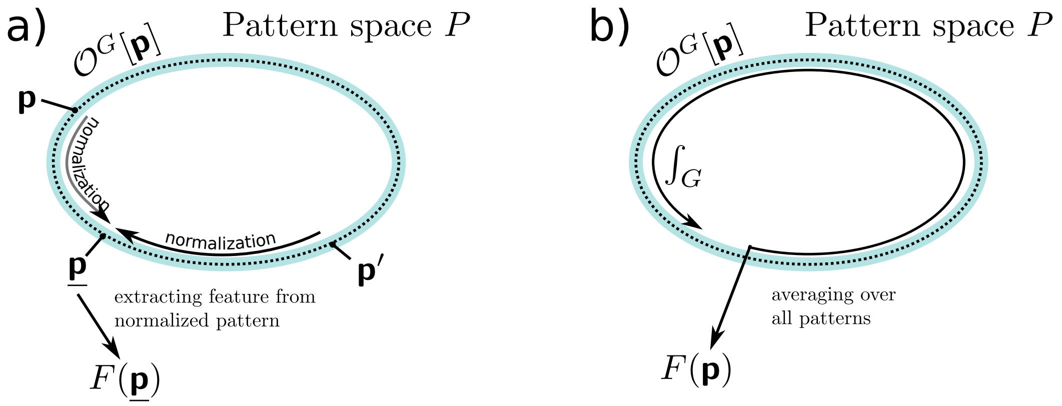


Figure 3.2.: a) Normalization: patterns are normalized to one specific representative of the orbit before extracting the feature. b) Group integration: averaging over all patterns in the orbit.

then the integral

$$F(\mathbf{p},\xi) := \frac{1}{|G|}\int_G \mathcal{K}[\boldsymbol{\tau}^P_{\mathfrak{g}}\mathbf{p}](\xi)\, d\mathfrak{g} \tag{3.7}$$

is invariant to G. ξ is a place-holder denoting additional meta-parameters of the kernel function.

Our aim is to create $SE(3)$ covariant filters whose output values are locally rotation invariant. Such a filter is induced by a feature extraction which is rotation invariant; see Eq. (1.12) in the introduction. That is, fulfilling for all images $I \in L(\mathbb{R}^3)$ and for all $\mathfrak{g} \in SO(3)$ the equation $F(\mathfrak{g}I) = F(I)$. In the following we consider a specific kind of rotation invariant features based on spherical harmonic functions. We derive the features via group integration techniques. Since we always can represent volumetric images in terms of spherical harmonic functions (in combination with a radial function), we focus on the creation of invariants for functions on the sphere. The results can then easily transfered to the "full" volumetric images. Moreover, it is worth mentioning that replacing the spherical harmonics with tensorial harmonics allows deriving the same kind of invariants for higher order spherical tensor fields.

3.1.1. $SO(3)$ Invariants For Functions on the Sphere

Any function $f \in L(S_2)$ can be represented as a linear combination of the orthogonal spherical harmonic basis functions. We derive the expansion coefficients via orthogonal projection of the spherical functions onto this basis. We denote the expansion coefficients by

$$\mathbf{v}^\ell := \langle f, \overline{\mathbf{Y}^\ell}\rangle, \tag{3.8}$$

where $\mathbf{v}^\ell \in \mathbb{C}^{2\ell+1}$ is a spherical tensor. The projection is a component-by-component projection in the sense that

$$v^\ell_m := \langle f, \overline{Y^\ell_m}\rangle \tag{3.9}$$

for all $m = -\ell \cdots \ell$. We have seen that the coefficients are obeying a predictable rotation behavior via multiplication with a Wigner D-matrix. That is

$$\mathbf{D}^{\ell}_{\mathfrak{g}} \mathbf{v}^{\ell} = \langle \mathfrak{g} f, \overline{\mathbf{Y}^{\ell}} \rangle. \tag{3.10}$$

In our applications we consider two kinds of invariants that we obtain via group integration techniques: second order invariants and third order invariants. It turns out that the resulting invariants coincide with the angular power-spectrum and the angular bi-spectrum.

3.1.1.1. Second Order Spectra Invariants

The first kernel function that we use is based on a second order monomial involving two spherical harmonic projections. Let $f \in L(S_2)$ be a function on the sphere. We define a kernel $\mathcal{K}_{(2)} : L(S_2) \to \mathbb{C}$ according to

$$\begin{aligned} \mathcal{K}_{(2)}{}^{\ell_1 \ell_2}_{m_1 m_2}(f) &:= \langle f, \overline{Y^{\ell_1}_{m_1}} \rangle \overline{\langle f, \overline{Y^{\ell_2}_{m_2}} \rangle} \\ &= v^{\ell_1}_{m_1} \overline{v^{\ell_2}_{m_2}}\,, \end{aligned} \tag{3.11}$$

where the $v^{\ell}_{m} \in \mathbb{C}$ are spherical harmonic expansion coefficients of f. For better readability we attached the integer valued kernel parameters as subscript and superscript to the function name. The subscript 2 denotes that two spherical harmonic coefficients are involved. That is, the kernel is a product of two spherical harmonic projections (the second term is complex conjugated).

Now we plug Eq. (3.11) into Eq. (3.7) and integrate over the rotation group. This results in a nonlinear feature extraction $F_2(\ell_1, \ell_2) : L(S_2) \to \mathbb{C}$. The resulting features are invariant to 3D rotations. F_2 is defined by

$$\begin{aligned} F_2(f, \ell_1, \ell_2, m_1, m_2) &:= \int_{SO(3)} \mathcal{K}_{(2)}{}^{\ell_1 \ell_2}_{m_1 m_2}(\mathfrak{g} f) d\mathfrak{g} = \int_{SO(3)} \langle \mathfrak{g} f, \overline{Y^{\ell_1}_{m_1}} \rangle \overline{\langle \mathfrak{g} f, \overline{Y^{\ell_2}_{m_2}} \rangle} \\ &= \int_{SO(3)} [\mathbf{D}^{\ell_1}_{\mathfrak{g}} \mathbf{v}^{\ell_1}]_{m_1} \overline{[\mathbf{D}^{\ell_2}_{\mathfrak{g}} \mathbf{v}^{\ell_2}]_{m_2}} d\mathfrak{g} = \int_{SO(3)} \sum_{n_1, n_2} [\mathbf{D}^{\ell_1}_{\mathfrak{g}}]_{m_1, n_1} v^{\ell_1}_{n_1} \overline{[\mathbf{D}^{\ell_2}_{\mathfrak{g}}]_{m_2, n_2} v^{\ell_2}_{n_2}} d\mathfrak{g} \\ &= \sum_{n_1, n_2} v^{\ell_1}_{n_1} \overline{v^{\ell_2}_{n_2}} \underbrace{\int_{SO(3)} [\mathbf{D}^{\ell_1}_{\mathfrak{g}}]_{m_1, n_1} \overline{[\mathbf{D}^{\ell_2}_{\mathfrak{g}}]_{m_2, n_2}} d\mathfrak{g}}_{\text{(orthogonal)}} = \sum_{n_1, n_2} v^{\ell_1}_{n_1} \overline{v^{\ell_2}_{n_2}} \frac{8\pi^2}{2\ell_1 + 1} \delta_{\ell_1, \ell_2} \delta_{m_1, m_2} \delta_{n_1, n_2} \\ &= \underbrace{\frac{8\pi^2}{2\ell_1 + 1}}_{=|G| \, \left(\substack{\text{group} \\ \text{weight}}\right)} \delta_{\ell_1, \ell_2} \delta_{m_1, m_2} \langle \mathbf{v}^{\ell_1}, \mathbf{v}^{\ell_2} \rangle = \frac{8\pi^2}{2\ell_1 + 1} \delta_{\ell_1, \ell_2} \delta_{m_1, m_2} \|\mathbf{v}^{\ell_1}\|^2 \quad . \end{aligned} \tag{3.12}$$

The resulting invariant is the energy of a spherical harmonic expansion coefficient, hence they coincides with the angular power-spectrum; Def. (2.2.27) on page 68. According to this coincidence we call the resulting feature the angular power-spectrum.

Definition 3.1.1 (Angular Power-Spectrum Invariants). *Let $f \in L(S_2)$ be a function on the sphere. Let further be $\mathbf{v}^\ell$ be the expansion coefficient of f, associated with the spherical harmonic of order ℓ. Then the angular power spectrum invariants are defined by*

$$F_2(f,\ell) := \langle \mathbf{v}^\ell, \mathbf{v}^\ell \rangle \tag{3.13}$$

Note that we omit the group weight here. It represents nothing else than the energy in each expansion coefficient. The interrelation between coefficients is not preserved.

In a similar way we can derive invariants based on the projection of two different functions $f_1, f_2 \in L(S_2)$, or by using two different kinds of basis functions (varying the radial component in a 3D image scenario). The resulting invariants are then the result of the inner product of two expansion coefficients, associated with the same spherical harmonic order.

3.1.1.2. Third Order Spectra Invariants

The second kernel function that we utilize is based on a third order monomial involving three spherical harmonic projections. Let $f \in L(S_2)$ be a function on the sphere. We define a kernel $\mathcal{K}_{(3)} : L(S_2) \rightarrow \mathbb{C}$ according to

$$\begin{aligned}\mathcal{K}_{(3)}{}^{\ell_1\ell_2\ell_3}_{m_1m_2m_3}(f) &:= \langle f, \overline{Y^{\ell_1}_{m_1}} \rangle \overline{\langle f, \overline{Y^{\ell_2}_{m_2}} \rangle \langle f, \overline{Y^{\ell_3}_{m_3}} \rangle} \\ &= v^{\ell_1}_{m_1} \overline{v^{\ell_2}_{m_2} v^{\ell_3}_{m_3}}\,,\end{aligned} \tag{3.14}$$

where the $v^\ell_m \in \mathbb{C}$ are spherical harmonic expansion coefficients of f. The subscript 3 denotes that three spherical harmonic coefficients are involved. That is, the kernel is a triple-product of three spherical harmonic projections (the second two coefficients are complex conjugated).

In the same way as for the second order invariants we obtain a nonlinear rotation invariant feature from Eq. (3.14) via usage of Eq. (3.7). This results in a nonlinear feature extraction $F_3(\ell_1, \ell_2, \ell_2) : L(S_2) \rightarrow \mathbb{C}$. F_3 is defined by

$$\begin{aligned}
&F_3(f.\ell_1,\ell_2,\ell_3,m_1,m_2,m_3) := \int_{SO(3)} \mathcal{K}_{(3)}{}^{\ell_1\ell_2\ell_3}_{m_1m_2m_3}(\mathfrak{g}f)d\mathfrak{g} = \int_{SO(3)} \\
&= \int_{SO(3)} [\mathbf{D}^{\ell_1}_{\mathfrak{g}}\mathbf{v}^{\ell_1}]_{m_1} \overline{[\mathbf{D}^{\ell_2}_{\mathfrak{g}}\mathbf{v}^{\ell_2}]_{m_2}[\mathbf{D}^{\ell_3}_{\mathfrak{g}}\mathbf{v}^{\ell_3}]_{m_3}} d\mathfrak{g} \\
&= \int \sum_{n_1,n_2,n_3} [\mathbf{D}^{\ell_1}_{\mathfrak{g}}]_{m_1,n_1} a^{\ell_1}_{n_1} \overline{[\mathbf{D}^{\ell_2}_{\mathfrak{g}}]_{m_2,n_2} v^{\ell_2}_{n_2} [\mathbf{D}^{\ell_3}_{\mathfrak{g}}]_{m_3,n_3} v^{\ell_3}_{n_3}} d\mathfrak{g} \\
&= \sum_{n_1,n_2,n_3} v^{\ell_1}_{n_1} \overline{v^{\ell_2}_{n_2} v^{\ell_3}_{n_3}} \int [\mathbf{D}^{\ell_1}_{\mathfrak{g}}]_{m_1,n_1} \overline{[\mathbf{D}^{\ell_2}_{\mathfrak{g}}]_{m_2,n_2} [\mathbf{D}^{\ell_3}_{\mathfrak{g}}]_{m_3,n_3}} d\mathfrak{g} \\
&= \sum_{n_1,n_2,n_3} v^{\ell_1}_{n_1} \overline{v^{\ell_2}_{n_2} v^{\ell_3}_{n_3}} \frac{8\pi^2}{2\ell_1+1} \langle \ell_2 m_2, \ell_3 m_3 \,|\, \ell_1 m_1 \rangle \langle \ell_2 n_2, \ell_3 n_3 \,|\, \ell_1 n_1 \rangle \\
&= \underbrace{\tfrac{8\pi^2}{2\ell_1+1}}_{=|G| \, (\substack{\text{group} \\ \text{weight}})} \quad \langle \ell_2 m_2, \ell_3 m_3 \,|\, \ell_1 m_1 \rangle \langle \mathbf{v}^{\ell_1}, (\mathbf{v}^{\ell_2} \circ_{\ell_1} \mathbf{v}^{\ell_3}) \rangle \quad .
\end{aligned} \tag{3.15}$$

The resulting invariant represents the interrelation of three spherical harmonic expansion coefficients including both magnitude and phase. They coincide with the angular bi-spectrum; Def. (2.2.28) on page 68.

Definition 3.1.2 (Angular Bi-Spectrum Invariants)**.** *Let $f \in L(S_2)$ be a function on the sphere. Let further be $\mathbf{v}^{\ell}$ be the expansion coefficient of f, associated with the spherical harmonic of order ℓ. Then the angular bi-spectrum invariants are defined by*

$$F_3(f,\ell_1,\ell_2,\ell_3) := \langle \mathbf{v}^{\ell_1}, (\mathbf{v}^{\ell_2} \circ_{\ell_1} \mathbf{v}^{\ell_3}) \rangle \quad . \tag{3.16}$$

Note that we omit the weighting factor here.

In a similar way we can derive invariants based on the projection of three different functions $f_1, f_2, f_3 \in L(S_2)$ by multiplying their coefficients (which in biomedical applications might be different color channels representing different types of staining). Alternatively we can use different kinds of basis functions (varying the radial component in a 3D image scenario).

The angular bi-spectrum provides phase and magnitude preserving features. It is a much more detailed representation than the power-spectrum. But it comes with a large number of possible products between spherical harmonic expansion coefficients. Hence we end up with a large number of features. It turns out that if the input signal is real valued, then the large degree of freedom in combining tensors of different ranks and forming new tensors of varying ranks brings linear dependencies into the set of all possible features (similar to the power spectrum of a real valued 1D signal). This is, keeping memory usage and computation time in mind, a scenario that is worth avoiding. Regarding bi-spectrum features on $SO(3)$, this issue has not been addressed so far.

Swapping two multiplicands within a product might only change a products parity; see Eq. (2.95) on page 64. We consider such kinds of linear dependencies as trivial and will focus on a more specific problem that only occurs for higher order products involving more than two multiplicands.

Theorem 3.1.3 (Nontrivial Identities of Bi-Spectrum Invariants)**.** *Let $f \in L(S_2,\mathbb{R}^3)$ be a real valued function on the sphere. Then we have the following (non-trivial) identities for the bi-spectrum invariants:*

$$F_3(f,\ell_1,\ell_2,\ell_3) = \tfrac{(-1)^{\ell_1}\sqrt{2\ell_2+1}}{(-1)^{\ell_2}\sqrt{2\ell_1+1}} F_3(f,\ell_2,\ell_3,\ell_1) = \tfrac{(-1)^{\ell_1}\sqrt{2\ell_3+1}}{(-1)^{\ell_3}\sqrt{2\ell_1+1}} F_3(f,\ell_3,\ell_1,\ell_2) \quad . \tag{3.17}$$

That is, it is sufficient to consider only two cases for each tensor coefficient: being one of the tensors within the inner tensor products or not. All remaining permutations are either trivial identities (just switching to operands within a product), or one of the cases shown above. Consequently, if we only count the non-trivial identities, we save up to three times the number of all possible invariants if we make use of the equation above!

Proof: Linear Dependencies in $SO(3)$ Bi-Spectra of Real Valued Functions. The inner product can be written in terms of a tensor product (Eq. (2.2.26) on page 67). In the resulting term a tensor conjugation appears which can be split up according to Eq. (2.2.25). Hence the features written as tensor products are

$$\begin{aligned}F_3(f,\ell_1,\ell_2,\ell_3) &= \langle \mathbf{v}^{\ell_1}, (\mathbf{v}^{\ell_2} \circ_{\ell_1} \mathbf{v}^{\ell_3})\rangle \\ &= (\mathbf{v}^{\ell_1} \bullet_0 \left(\mathbf{v}^{\ell_2} \circ_{\ell_1} \mathbf{v}^{\ell_3}\right)^{\ddagger}) \\ &= (-1)^{(\ell_1+\ell_2+\ell_3)}(\mathbf{v}^{\ell_1} \bullet_0 (\left(\mathbf{v}^{\ell_2}\right)^{\ddagger} \circ_{\ell_1} \left(\mathbf{v}^{\ell_3}\right)^{\ddagger}))\,. \end{aligned} \tag{3.18}$$

Consequently, the bi-spectrum features coincide with the tensor product of three expansion coefficients, where two of them are the tensor conjugated coefficients. Let $f \in L(S_2,\mathbb{R}^3)$ be a real valued function. As a result, the tensor conjugation simplifies to $\left(\mathbf{v}^{\ell}\right)^{\ddagger} = \mathbf{v}^{\ell}$ (see Eq. (2.2.35)) and we obtain the following simplification:

$$\begin{aligned}F_3(f,\ell_1,\ell_2,\ell_3) &= \langle \mathbf{v}^{\ell_1}, (\mathbf{v}^{\ell_2} \circ_{\ell_1} \mathbf{v}^{\ell_3})\rangle \\ &= (\mathbf{v}^{\ell_1} \bullet_0 \left(\mathbf{v}^{\ell_2} \circ_{\ell_1} \mathbf{v}^{\ell_3}\right)^{\ddagger}) \\ &= (-1)^{(\ell_1+\ell_2+\ell_3)}(\mathbf{v}^{\ell_1} \bullet_0 (\left(\mathbf{v}^{\ell_2}\right)^{\ddagger} \circ_{\ell_1} \left(\mathbf{v}^{\ell_3}\right)^{\ddagger})) = (\mathbf{v}^{\ell_1} \circ_0 (v^{\ell_2} \circ_{\ell_1} v^{\ell_3}))\,. \end{aligned} \tag{3.19}$$

We can then show by considering all non-trivial permutations of ℓ_1,ℓ_2,ℓ_3 in combination with the associativity rule of spherical products (Cor. 2.2.31 on page 71) that

$$\begin{aligned}F_3(f,\ell_1,\ell_2,\ell_3) &= (-1)^{(\ell_1+\ell_2+\ell_3)}\frac{(-1)^{\ell_1}}{\sqrt{2\ell_1+1}}(\mathbf{v}^{\ell_1} \circ_0 (\mathbf{v}^{\ell_2} \circ_{\ell_1} \mathbf{v}^{\ell_3})) \qquad (3.20)\\ &= (-1)^{(\ell_1+\ell_2+\ell_3)}\frac{(-1)^{\ell_1}}{\sqrt{2\ell_1+1}}(\mathbf{v}^{\ell_2} \circ_0 (\mathbf{v}^{\ell_3} \circ_{\ell_2} \mathbf{v}^{\ell_1})) = \frac{(-1)^{\ell_1}\sqrt{2\ell_2+1}}{(-1)^{\ell_2}\sqrt{2\ell_1+1}}F_3(f,\ell_2,\ell_3,\ell_1) \\ &= (-1)^{(\ell_1+\ell_2+\ell_3)}\frac{(-1)^{\ell_1}}{\sqrt{2\ell_1+1}}(\mathbf{v}^{\ell_3} \circ_0 (\mathbf{v}^{\ell_1} \circ_{\ell_3} \mathbf{v}^{\ell_2})) = \frac{(-1)^{\ell_1}\sqrt{2\ell_3+1}}{(-1)^{\ell_3}\sqrt{2\ell_1+1}}F_3(f,\ell_3,\ell_1,\ell_2)\,. \end{aligned}$$

□

3.1.1.3. Distinguishing Reflections

One specific case which is worth further investigation is reflection. Reflection plays an important role when searching specific landmarks in images of specimens or structures showing a natural reflection axis. This is e.g. the case for many organisms of higher forms of life including images of brains, or whole organisms like zebra fish embryos (Ronneberger et al. 2012). We consider w.l.o.g the reflection about the origin (reflection about an axis is the result of reflection about the origin, followed by a rotation). Let $f \in L_2(S_2)$. The function's expansion coefficients compute according to $\mathbf{v}^{\ell} := \langle f, \overline{\mathbf{Y}^{\ell}}\rangle$; see Eq. (2.64). Let further $f'(\mathbf{n}) := f(-\mathbf{n})$ be the reflected version of f. The expansion coefficients $\mathbf{w}^{\ell}$ of f' can be determined to be

$$\begin{aligned}\mathbf{w}^{\ell} &:= \langle f', \overline{\mathbf{Y}^{\ell}}\rangle \\ &= \int_{S_2} f'(\mathbf{n})\mathbf{Y}^{\ell}(\mathbf{n})d\mathbf{n} = \int_{S_2} f(-\mathbf{n})\mathbf{Y}^{\ell}(\mathbf{n})d\mathbf{n} \\ &= \int_{S_2} f(\mathbf{n})\mathbf{Y}^{\ell}(-\mathbf{n})d\mathbf{n} = (-1)^{\ell}\int_{S_2} f(\mathbf{n})\mathbf{Y}^{\ell}(\mathbf{n})d\mathbf{n} \quad \text{(Eq. (2.68))} \\ &= (-1)^{\ell}\mathbf{v}^{\ell} \quad . \end{aligned} \tag{3.21}$$

That is, only the expansion coefficients associated with an odd frequency do differ by their sign. It is obvious that the power spectrum according to Def. 3.1.1 is the same for both functions, namely

$$\langle \mathbf{w}^\ell, \mathbf{w}^\ell \rangle = \langle (-1)^\ell \mathbf{v}^\ell, (-1)^\ell \mathbf{v}^\ell \rangle = (-1)^{(2\ell)} \langle \mathbf{v}^\ell, \mathbf{v}^\ell \rangle = \langle \mathbf{v}^\ell, \mathbf{v}^\ell \rangle \quad . \tag{3.22}$$

Hence it cannot be used to distinguish between f and f'. Contrarily, the bi-spectrum preserves phase information and thus resolves this issue (Kakarala 2012). Consider the more general triple product $(\mathbf{w}^{\ell_1} \circ_L (\mathbf{w}^{\ell_2} \circ_J \mathbf{w}^{\ell_3}))$. By substituting $\mathbf{w}^\ell$ with $(-1)^\ell \mathbf{v}^\ell$ we obtain the following equation

$$\begin{aligned}(\mathbf{w}^{\ell_1} \circ_L (\mathbf{w}^{\ell_2} \circ_J \mathbf{w}^{\ell_3})) \qquad &= ((-1)^{\ell_1} \mathbf{v}^{\ell_1} \circ_L ((-1)^{\ell_2} \mathbf{v}^{\ell_2} \circ_J (-1)^{\ell_3} \mathbf{v}^{\ell_3})) \\ &= (-1)^{(\ell_1+\ell_2+\ell_3)} (\mathbf{v}^{\ell_1} \circ_L (\mathbf{v}^{\ell_2} \circ_J \mathbf{v}^{\ell_3})) \quad .\end{aligned} \tag{3.23}$$

Hence only the triple products where $(-1)^{(\ell_1+\ell_2+\ell_3)}$ is odd can distinguish between a reflected and the original version of a function.

Moreover, it turns out that for the invariant case ($L = 0$), only a few number of odd products differ from the zero tensor, namely those where $\ell_1 \neq \ell_2 \neq \ell_3$.

Proof. Due to the symmetry of the products it is sufficient to consider the case where $(\mathbf{w}^{\ell_1} \circ_0 (\mathbf{w}^{\ell_2} \circ_{\ell_1} \mathbf{w}^{\ell_3}))$. Assume $\ell_2 = \ell_3$. If $\ell_2 + \ell_3 + \ell_1$ is odd, then $(\mathbf{w}^{\ell_2} \circ_{\ell_1} \mathbf{w}^{\ell_3}) = \mathbf{0}$ and thus the whole product (see Eq. (2.95)). □

Definition 3.1.4 (Even and Odd Invariants). *We call the bi-spectrum invariants where* $(\ell_1 + \ell_2 + \ell_3)$ *is even "even bi-spectrum invariants" and the invariants where* $(\ell_1 + \ell_2 + \ell_3)$ *is odd "odd bi-spectrum invariants".*

3.1.2. Power-spectrum versus Bi-Spectrum

The power-spectrum provides intrinsic information about the amount of a certain angular frequency within a pattern. However, we loose any phase information so that information about the interrelation of different frequency components is lost. Consider the 2D example illustrated in Fig. 3.3 on the following page: in 2D the angular harmonic functions are the so-called circular harmonics (see the example on on page 46). Figure 3.3 shows four different functions resulting from a linear combination of the same three circular harmonic functions (only the real part is shown). For all three cases the frequencies and amplitudes of the basis functions are not altered. But they are rotated (cyclic shifted) differently. Fig. 3.3 a) is the reference pattern. Fig. 3.3 b) a rotated version (all basis waves are rotated in an identical manner). In the cases shown in figures c) and d), one basis wave has been rotated differently thus the resulting patterns are not a rotated version of a) any more. Now assume we compute the power spectrum as feature. The angular power spectrum represents the power in each angular frequency within a pattern in a separate manner. Consequently, the angular power spectrum is the same for all four patterns, despite the fact that c) and d) are different to a) and b). The loss of phase information means we cannot conclude the relative orientation of different frequencies from the invariant feature. A solution is provided by the so-called (here angular) bi-spectrum. The bi-spectrum combines three expansion coefficients in a non-linear

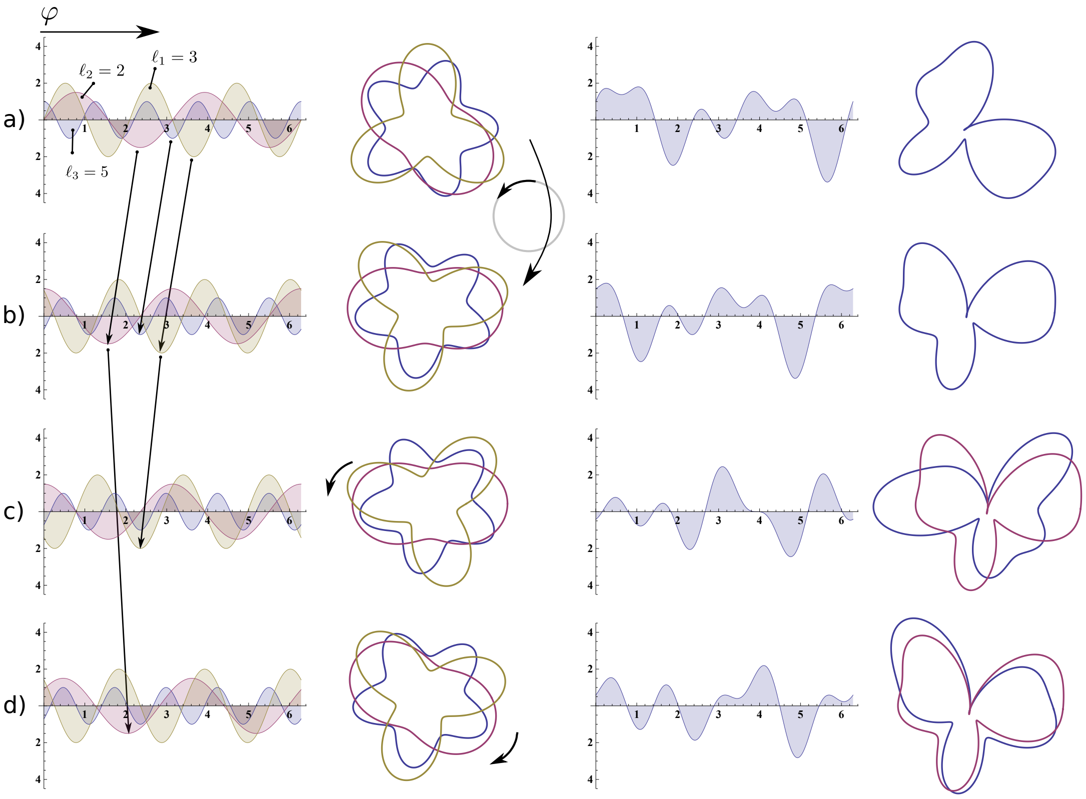


Figure 3.3.: The first and the second columns show the three basis functions forming the functions in the third and forth columns. We depict two kinds of representations, one as periodic function, the other one as function on the circle. Computing the power spectrum for all four functions would lead to the same result. This is because all four functions are the result of superimposing basis functions of the same frequency and magnitude. They are just shifted (rotated) differently. The bi-spectrum represents the interrelation of each triple of (phase)shifts and magnitudes of the basis functions such that the first two functions have the same bi-spectrum (they are just rotated versions of each other), but each spectrum of the remaining two functions would differ from all other three spectra.

way preserving the interrelation of the coefficients. For compact groups with regular expansion coefficients, like $SO(2)$, the resulting features are even *complete* (Kakarala 1992), that is, the function can be recovered up to the rotation acting on the function. For the example in Fig. 3.3 the functions a) and b) would result in the same bi-spectrum, while each of the spectra for c) and d) would be dissimilar to the bi-spectra of the remaining three functions. Note that for functions on the 2-sphere the bi-spectrum preserves phase information, too. Hence it provides a much richer description than the power-spectrum. However, the bi-spectrum is incomplete for functions on the sphere: remember that in this case the expansion coefficients of the Wigner D-matrices are irregular matrices (see Eq. (2.56)).

3.2. $SE(3)$ Covariant Filters For Rotation Invariant Detection

The $SE(3)$ covariant spherical harmonic transformation has been introduced as a tool for mapping images onto spherical tensor fields. The elements of the resulting tensor fields are local spherical harmonic expansion coefficients. The expansion coefficients are representing both, the appearance and orientation of a local image pattern. The big benefit of such a filter is the fact that the output image transforms in a predictable manner under full Euclidean motion. However, in many applications we are rather interested in the presence or absence of objects than in determining their orientation. Assume the scenario where a biologist is annotating an image with cells. The cells should be labeled according to their cell state. In such a scenario the cell orientation is not influencing its classification. For this example and similar scenarios it is worth making use of the proposed rotation invariant features to form an $SE(3)$ covariant filter which output transforms like a scalar valued image. That is, a rotation and translation of the input image leads to a pure coordinate transform of the output image. The values themselves are behaving like intensity values of an image, but representing intrinsic information of the underlying input pattern including its local neighborhood. Such a filter output can be regarded as an "intelligent saliency map" which, thanks to local rotation invariance, facilitates detection of objects and structures.

Such a filter can be created easily. We first extend the kernel function Eq. (3.11) to volumetric images. For this we just replace the spherical harmonic projection with the projection for volumetric images by introducing an additional radial component. Hence for an image $I \in L(\mathbb{R}^3)$ we get

$$\begin{aligned}\mathcal{K}_{(2)}{}^{\ell_1\ell_2}_{m_1m_2}(I,r) &:= \langle I, \overline{Y^{\ell_1}_{m_1}}\delta_r\rangle \overline{\langle I, \overline{Y^{\ell_2}_{m_2}}\delta_r\rangle} \\ &= v^{\ell_1}_{m_1}(r)\overline{v^{\ell_2}_{m_2}(r)}\,, \end{aligned} \tag{3.24}$$

where $v^{\ell}_{m}(r)$ are expansion coefficients of I. The rest can be derived in the same way as for functions on the sphere. Thus the resulting invariants are

$$F_2(I,\ell,r) := \langle \mathbf{v}^{\ell}(r), \mathbf{v}^{\ell}(r)\rangle \qquad \text{(and)} \tag{3.25}$$

$$F_3(I,\ell_1.\ell_2,\ell_3,r) := \langle \mathbf{v}^{\ell_1}(r), (\mathbf{v}^{\ell_2}(r) \circ_{\ell_1} \mathbf{v}^{\ell_3}(r))\rangle \quad . \tag{3.26}$$

The additional parameter r is the radius parameter stemming from the spherical transformation for images. Later we introduce further spherical harmonic transformations introducing different kinds of radial functions. For these alternatives the radial parameter will vary. However, the radial parameters are not angular dependent so that features can be computed in the same manner. Since the features Eq. (3.25) and Eq. (3.26) are rotation covariant (or even rotation invariant), they induce an $SE(3)$ covariant filter $\mathcal{F}_2\{\ell,r\} : L_2(\mathbb{R}^3) \rightarrow L_2(\mathbb{R}^3)$ with

$$\mathcal{F}_2\{I,\ell,r\}(\mathbf{x}) := F_2(\mathfrak{h}_{(-\mathbf{x})}I,\ell,r) \tag{3.27}$$

and $\mathcal{F}_3\{\ell_1.\ell_2,\ell_3,r\} : L_2(\mathbb{R}^3) \rightarrow L_2(\mathbb{R}^3)$ with

$$\mathcal{F}_3\{I,\ell_1.\ell_2,\ell_3,r\}(\mathbf{x}) := F_3(\mathfrak{h}_{(-\mathbf{x})}I,\ell_1.\ell_2,\ell_3,r) \quad . \tag{3.28}$$

Since the features are rotation invariant, both filters are covariant to Euclidean motion in the sense that

$$\mathcal{F}_2\{\mathfrak{h}_{\mathbf{t}}\mathfrak{g}I,\ell,r\}(\mathbf{x}) = \mathcal{F}_2\{I,\ell,r\}(\mathbf{U}_{\mathfrak{g}}(\mathbf{x}-\mathbf{t})) \quad \text{and} \tag{3.29}$$

$$\mathcal{F}_3\{\mathfrak{h}_{\mathbf{t}}\mathfrak{g}I,\ell_1.\ell_2,\ell_3,r\}(\mathbf{x}) = \mathcal{F}_3\{I,\ell_1.\ell_2,\ell_3,r\}(\mathbf{U}_{\mathfrak{g}}(\mathbf{x}-\mathbf{t})) \quad . \tag{3.30}$$

That is, they transform like scalar valued images. Hence, together with a trainable classifier, the pure spherical harmonic transformation turns into an appropriate tool for generic object detection in volumetric images. There is no need for the classifier to cope with local object rotations thus such a system is predestined for automated biomedical image annotation. Examples of such feature images are shown in Fig. 3.4 on the next page.

For the detection of complex objects a single scalar valued image of local rotation invariant features is by far not sufficient. Therefore, the output of several filters is combined to form a vector valued feature image. The single components of such a vector transform like scalar valued images so that elements of such a feature vector do not "mix" under rotation. We call such a feature vector a rotation invariant descriptor.

Definition 3.2.1 (Descriptor Images)**.** *We use a spherical harmonic transform in combination with the angular power spectrum, as well as the angular bi-spectrum for creating a new image filter. The proposed filter maps images to multidimensional saliency maps, which can be considered as a vector valued feature image. Since the single vector components of such a filter output transform like scalar valued images, the feature vectors components do not mix under rotations. We call such feature vectors rotation invariant descriptors. Accordingly, we call descriptor valued images descriptor images.*

3.2.1. Implementation

A descriptor image can be computed in two steps. We first use one of the proposed $SE(3)$ covariant spherical harmonic transformations to map input images to images containing the spherical harmonic expansion coefficients. In a second step, we use the expansion coefficients to form invariants in a voxel-by-voxel manner using the products in Eq. (3.25) and Eq. (3.26). This is often the only possible way for realizing the proposed filters, because only then we can benefit from a fast dense spherical harmonic transform allowing for simultaneously computing thousands, or even millions of spherical harmonic transformations in a voxel-by-voxel manner in reasonable time.

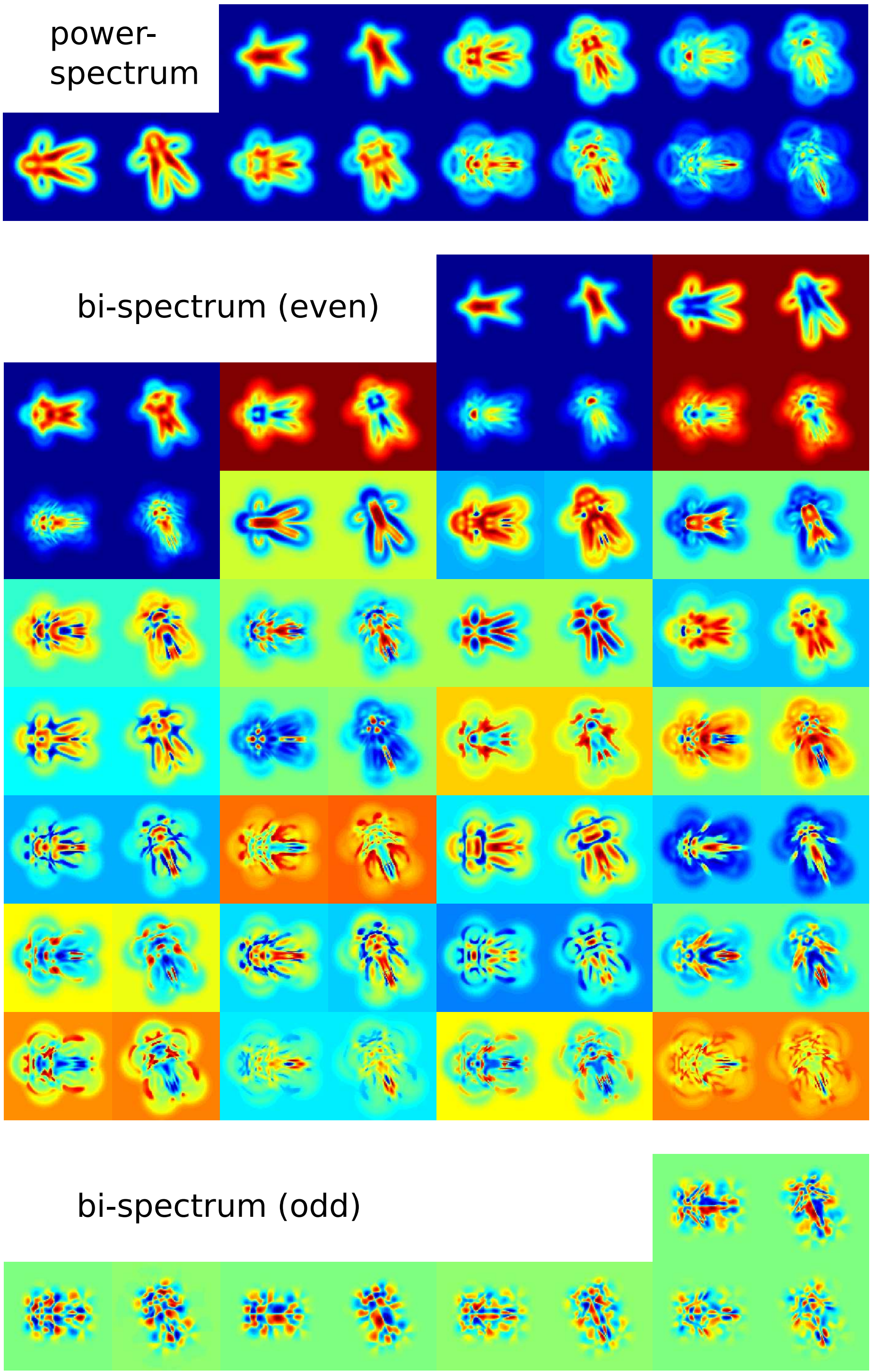


Figure 3.4.: The angular power-spectrum and the bi-spectrum of a volumetric image showing a human shaped model. The bi-spectrum is divided into its even and odd components. We only show the center slice. Note that the odd bi-spectrum resolves the reflection symmetry. (we also show a version of the features where the image has been rotated; red is highest, blue lowest value). We took the square root from the power-spectrum and third root from the bi-spectrum ($\text{sign}(f)|f|^{1/3}$).

4 Spherical Tensor Derivatives

In Cartesian tensor calculus we have differential operators like the gradient operator acting on images $I \in L_2(\mathbb{R}^3)$. The resulting functions are describing characteristics of the initial function in an analytical manner. For instance, when applying the gradient operator $\nabla = (\partial_x, \partial_y, \partial_z)^T$ to I we get the gradient image, a field with tensor rank 1. The gradients are pointing into the greatest rate of increase of I. The gradient magnitude is the rate of increase. The gradient field can be used e.g. for edge and corner detection (Canny 1986; Harris and Stephens 1988). Note that an image itself can be regarded as tensor fields of order 0; see Eq. (2.41) on page 50. Applying the gradient operator twice coincides with the Hessian matrix $H = \nabla\nabla^T$ of I and we get a Cartesian tensor field of order two. The eigenvalues of the Hessian matrices are proportional to the function's curvature. The largest eigenvector of the Hessian matrix is pointing into the direction of largest curvature, while the smallest eigenvector is pointing into the direction of smallest curvature. Therefore, the ratio of eigenvalues is widely used for blob detection; see e.g. Lowe (2004), where it is used for accurate key-point localization. Another example is the Laplace operator $\Delta = \mathrm{trace}(H) = \partial_x^2 + \partial_y^2 + \partial_z^2$. The application of the Laplace operation leaves the tensor rank unchanged.

In spherical tensor algebra we have similar operators which we call the spherical (tensor) derivatives. Spherical derivatives are strongly related to the spherical tensor gradient operator playing an important role in theoretical chemistry (see e.g. Weniger (2005)). The following introduction of spherical tensor derivatives uses the notation proposed in Reisert and Burkhardt (2009b).

We denote by ∇^{S_2} the spherical gradient operator, the spherical counterpart of the gradient operator ∇. We obtain ∇^{S_2} by a unitary coordinate transform induced by the solid harmonics $\mathbf{R}^\ell$, where

$$\nabla^{S_2} := \mathbf{R}^1(\nabla) = \left(\frac{1}{\sqrt{2}}(\partial_x - i\partial_y), \partial_z, -\frac{1}{\sqrt{2}}(\partial_x + i\partial_y)\right)^T \quad ; \tag{4.1}$$

see section 2.2.3.6 on page 77 for the definition of solid harmonics. Consequently, the spherical gradient operator ∇^{S_2} inherits the rotation properties of the solid harmonics and rotates in the harmonic domain according to (see. Eq. (2.2.2) on page 50)

$$\mathbf{D}^1_{\mathfrak{g}}((\nabla^{S_2} I)(\mathbf{U}_{\mathfrak{g}}^T x)) = \mathbf{R}^1(\mathbf{U}_{\mathfrak{g}}(\nabla I)(\mathbf{U}_{\mathfrak{g}}^T x)) \quad , \tag{4.2}$$

i.e. ∇^{S_2} can be regarded as spherical tensor of order 1. The derivative of $I \in L_2(\mathbb{R}^3)$ in Fourier domain is just a point-wise multiplication of $\widetilde{I}$ with the frequency vector

$\mathbf{k} \in \mathbb{R}^3$ (see corollary 2.1.6). Utilizing the Fourier correspondence $\widetilde{\nabla I} = \widetilde{\nabla}\widetilde{I} = i\mathbf{k}\widetilde{I}$, we obtain the Fourier correspondence of ∇^{S_2}, namely

$$\widetilde{\nabla^{S_2}} = \mathbf{R}^1(\widetilde{\nabla}) = i\mathbf{R}^1(\mathbf{k}) \quad . \tag{4.3}$$

We denote by $\partial^\ell = \mathbf{R}^\ell(\nabla)$ the higher order spherical gradient operator. The single components are denoted by $\partial^\ell = (\partial^\ell_{-\ell}, \cdots, \partial^\ell_\ell)^T$. The higher order spherical gradient inherits the recursive relationship from the solid harmonics, thus

$$\begin{aligned} \partial^1 &:= \mathbf{R}^1(\nabla) = \nabla^{S_2} \quad \text{and} \\ \partial^{\ell+1} &:= (\mathbf{R}^\ell(\nabla) \bullet_{\ell+1} \mathbf{R}^1(\nabla)) = (\partial^\ell \bullet_{\ell+1} \nabla^{S_2}) \quad . \end{aligned} \tag{4.4}$$

Then its counterpart in Fourier domain is defined by

$$\begin{aligned} \widetilde{\partial}^1 &:= \widetilde{\nabla^{S_2}} \quad \text{and} \\ \widetilde{\partial}^{\ell+1} &:= (\widetilde{\partial}^\ell \bullet_{\ell+1} \widetilde{\nabla^{S_2}}) = (i^\ell \mathbf{R}^\ell(\mathbf{k}) \bullet_{\ell+1} i\mathbf{R}(\mathbf{k})) = i^{\ell+1}\mathbf{R}^{\ell+1}(\mathbf{k}) \quad . \end{aligned} \tag{4.5}$$

Due to Eq. (4.1) (symmetry of spherical harmonics) we have

$$\overline{\partial}^\ell_m = (-1)^m \partial^\ell_{-m} \quad , \tag{4.6}$$

i.e. ∂^ℓ shows the properties of a tensor in $\mathbb{V}_\ell$ (see section 2.2.3.8 on page 79). Based on ∂^ℓ we define the spherical tensor derivative operators.

Definition 4.0.2 (Spherical Derivatives). *Let* $\mathbf{f} \in \mathcal{T}_\ell$ *be a tensor-field of order* ℓ. *The rank increasing spherical up-derivative* $\boldsymbol{\nabla}^1 : \mathcal{T}_\ell \to \mathcal{T}_{\ell+1}$ *and for* $\ell > 0$ *the rank decreasing spherical down-derivative* $\boldsymbol{\nabla}_1 : \mathcal{T}_\ell \to \mathcal{T}_{\ell-1}$ *are defined by*

$$\begin{aligned} \boldsymbol{\nabla}^1 \mathbf{f} &:= \partial^1 \bullet_{\ell+1} \mathbf{f} \qquad\qquad \textit{and} \\ \boldsymbol{\nabla}_1 \mathbf{f} &:= \partial^1 \bullet_{\ell-1} \mathbf{f} \quad . \end{aligned} \tag{4.7}$$

With $\overline{\boldsymbol{\nabla}}^1$ and $\overline{\boldsymbol{\nabla}}_1$ we denote the complex conjugate operator, where $\overline{\boldsymbol{\nabla}}^1 \mathbf{f} := \overline{\partial}^1 \bullet_{\ell+1} \mathbf{f}$. The complex conjugated spherical down-derivatives are defined accordingly.

Similarly to their Cartesian counterparts, spherical derivatives can change the rank of tensor fields. The meaning of the spherical tensor fields changes with respect to the differential operation acting on the field. For instance, given a scalar valued field, applying the spherical up-derivative operator transforms the field to a spherical tensor field of order one, the spherical counterpart of a gradient vector field. When applying the same operator twice we obtain a spherical tensor field of order two, corresponding to a traceless Hessian matrix.

In the same way we can define the spherical tensor differentiation in Fourier domain.

Definition 4.0.3 (Spherical Derivatives (Fourier Domain)). *The rank increasing spherical up-derivative* $\widetilde{\boldsymbol{\nabla}}^1 : \mathcal{T}_\ell \to \mathcal{T}_{\ell+1}$ *and the rank decreasing spherical down-derivative* $\widetilde{\boldsymbol{\nabla}}_1 : \mathcal{T}_\ell \to \mathcal{T}_{\ell-1}$ *are defined by*

$$\begin{aligned} \widetilde{\boldsymbol{\nabla}}^1 \widetilde{\mathbf{f}} &:= i\mathbf{R}^1(\mathbf{k}) \bullet_{\ell+1} \widetilde{\mathbf{f}} \qquad\qquad \textit{and} \\ \widetilde{\boldsymbol{\nabla}}_1 \widetilde{\mathbf{f}} &:= i\mathbf{R}^1(\mathbf{k}) \bullet_{\ell-1} \widetilde{\mathbf{f}} \quad . \end{aligned} \tag{4.8}$$

We often consecutively apply the up-derivatives followed by a consecutive application of the down-derivatives. Given a field $\mathbf{f} \in \mathcal{T}_J$. When applying ℓ times the spherical up-derivatives followed by n times ($n \leq \ell$) the spherical down-derivatives we write $\boldsymbol{\nabla}_n \boldsymbol{\nabla}^\ell$, or in short notation $\boldsymbol{\nabla}^\ell_n$, with

$$\begin{aligned}
\boldsymbol{\nabla}^\ell_n \mathbf{f} &:= \underbrace{(\partial^1 \bullet_{J+(\ell-n)} ..(\partial^1 \bullet_{J+(\ell-1)}}_{n \text{ times } \boldsymbol{\nabla}_1} \underbrace{(\partial^1 \bullet_{J+\ell} ..(\partial^1 \bullet_{J+1} \mathbf{f})))))}_{\ell \text{ times } \boldsymbol{\nabla}^1} \qquad (4.9)\\
&= (\partial^n \bullet_{J+(\ell-n)} (\partial^\ell \bullet_{J+\ell} \mathbf{f})) \quad .\\
\widetilde{\boldsymbol{\nabla}^\ell_n \mathbf{f}} &:= (i\mathbf{R}^1(\mathbf{k}) \bullet_{J+(\ell-n)} ..(i\mathbf{R}^1(\mathbf{k}) \bullet_{J+(\ell-1)} (i\mathbf{R}^1(\mathbf{k}) \bullet_{J+\ell} ..(i\mathbf{R}^1(\mathbf{k}) \bullet_{J+1} \widetilde{\mathbf{f}})))))\\
&= i^{\ell+n} \mathbf{R}^\ell_n(\mathbf{k}) \bullet_{J+(\ell-n)} \widetilde{\mathbf{f}} \quad \text{(according to Eq. (2.143))} \quad . \qquad (4.10)
\end{aligned}$$

For scalar valued fields $\mathbf{f}^0 \in \mathcal{T}_0$, the up-down-derivatives are related to the Laplace operator Δ. From $\widetilde{\Delta f} = -k^2 \widetilde{f}$ (see corollary 2.1.6), we can conclude that

$$\begin{aligned}
\widetilde{\boldsymbol{\nabla}^\ell_n \mathbf{f}^0} &= i^{\ell+n} \mathbf{R}^\ell_n(\mathbf{k}) \bullet_{\ell-n} \widetilde{\mathbf{f}^0}\\
&= i^{\ell-n} \mathbf{R}^{(\ell-n)}(\mathbf{k})(ik)^{2n} \widetilde{\mathbf{f}^0}\\
&= \underbrace{i^{\ell-n} \mathbf{R}^{(\ell-n)}(\mathbf{k})}_{=\widetilde{\boldsymbol{\nabla}}^{\ell-n}} (\underbrace{-k^2}_{=\widetilde{\Delta}^n})^n \widetilde{\mathbf{f}^0} \quad . \qquad (4.11)
\end{aligned}$$

As a result we get

Proposition 4.0.4 (Spherical Up-Down Derivatives). *If* $\mathbf{f}^0 \in \mathcal{T}_0$, *then we can write* $\boldsymbol{\nabla}^\ell_n \mathbf{f}^0$ *in terms of the Laplace operator:*

$$\boldsymbol{\nabla}^\ell_n \mathbf{f}^0 = \boldsymbol{\nabla}^{\ell-n} \Delta^n \mathbf{f}^0 \quad . \qquad (4.12)$$

Proposition 4.0.5 (Commutativity under Convolutions). *Let* $\mathbf{f}^{J_1} \in \mathcal{T}_{J_1}$ *and* $\mathbf{f}^{J_2} \in \mathcal{T}_{J_2}$ *be two spherical tensor fields, then*

$$\begin{aligned}
(\boldsymbol{\nabla}^\ell \mathbf{f}^{J_1}) \widetilde{\bullet}_{(J_2-(J_1+\ell))} \mathbf{f}^{J_2} &= \mathbf{f}^{J_1} \widetilde{\bullet}_{(J_2-(J_1+\ell))} (\boldsymbol{\nabla}_\ell \mathbf{f}^{J_2}) \qquad \textit{and} \qquad (4.13)\\
(\boldsymbol{\nabla}^\ell \mathbf{f}^{J_1}) \widetilde{\bullet}_{(J_1+\ell+J_2)} \mathbf{f}^{J_2} &= \mathbf{f}^{J_1} \widetilde{\bullet}_{(J_1+\ell+J_2)} (\boldsymbol{\nabla}^\ell \mathbf{f}^{J_2}) \, , \qquad (4.14)
\end{aligned}$$

which can be proved by considering the convolution in Fourier domain including the Fourier representation of the derivatives and using the associativity rules 2.2.31 on page 71.

4.1. Polar Representation of Spherical Tensor Derivatives

Deriving an analytical solution of the spherical derivatives of spherical tensor fields is in general a nontrivial cumbersome task. However, we can derive very simple differentiation rules for tensor fields $\mathbf{f}^\ell \in \mathcal{T}_\ell$ having the form $\mathbf{f}^\ell(\mathbf{r}) = \mathbf{Y}^\ell(\mathbf{r}) f^\ell(r)$, where $f^\ell : \mathbb{R} \rightarrow \mathbb{C}$ is just a 1D function representing the radial profile. Note that all scalar valued ($\ell = 0$), isotropic functions like the isotropic Gaussian belong to this group of functions. This allows us to easily derive the analytical form of their tensor derivatives. Such functions are e.g. the spherical derivatives of the

- Gaussian windowed solid harmonics $\mathbf{f}^\ell(\mathbf{r}) := \mathbf{R}^\ell(\mathbf{r})e^{-\mathbf{r}^T\mathbf{r}/2} = \mathbf{Y}^\ell(\mathbf{r})\underbrace{r^\ell e^{-\mathbf{r}^T\mathbf{r}/2}}_{=f^\ell(r)}$ (which is for $\ell = 0$ the pure isotropic 3D Gaussian (Reisert and Burkhardt 2009a))
- Spherical Bessel functions $\mathbf{f}^\ell(\mathbf{r}, k) := \mathbf{Y}^\ell(\mathbf{r})\underbrace{j_\ell(kr)}_{=f^\ell(r)}$
- Gauss-Laguerre functions $\mathbf{f}_n^\ell(\mathbf{r}) := \mathbf{Y}^{\ell-n}(\mathbf{r})\underbrace{r^{\ell-n}L_n^{(\ell-n)+\frac{1}{2}}(\frac{r^2}{2})e^{-\mathbf{r}^T\mathbf{r}/2}}_{=f_n^\ell(r)}$

All these functions will be studied in detail in section 5.1 on page 110 and section 5.2 on page 128. The results of the current subsection have been partially published in Skibbe and Reisert (2013) and in the appendix of Skibbe et al. (2012).

Lemma 4.1.1. *Given a spherical tensor field $\mathbf{f}^\ell \in \mathcal{T}_\ell$ whose angular and radial components are separable such that $\mathbf{f}^\ell(\mathbf{r}) = \mathbf{Y}^\ell(\mathbf{r})f^\ell(r)$, where $f^\ell : \mathbb{R} \to \mathbb{C}$ denotes the function representing the radial component of $\mathbf{f}^\ell$. Then the spherical up- and down-derivatives of $\mathbf{f}^\ell$ have the form*

$$(\boldsymbol{\nabla}^1\mathbf{f}^\ell)(\mathbf{r}) = \mathbf{Y}^{\ell+1}(\mathbf{r})r^\ell\frac{\partial}{\partial r}\frac{1}{r^\ell}f^\ell(r) \qquad \textit{and} \qquad (4.15)$$

$$(\boldsymbol{\nabla}_1\mathbf{f}^\ell)(\mathbf{r}) = \mathbf{Y}^{\ell-1}(\mathbf{r})\frac{1}{r^{\ell+1}}\frac{\partial}{\partial r}r^{\ell+1}f^\ell(r) \qquad , \qquad (4.16)$$

respectively.

Proof. In order to prove lemma 4.1.1 we compute the Fourier correspondence of $\mathbf{f}^\ell$ component-by-component utilizing the spherical expansion of the plane wave (Eq. (C.18)). Then we compute the derivatives of $\mathbf{f}^\ell$ in frequency domain and transform back the results into spatial domain. We finally show that the results are identical to the results we obtain when applying the differentiation rules stated in lemma 4.1.1 in spatial domain.

First the Fourier transform of $\mathbf{f}^\ell$:

$$\begin{aligned}
&\widetilde{f_m^\ell}(\mathbf{k}) = (2\pi)^{-3/2}\langle f_m^\ell, e^{i\mathbf{k}^T\mathbf{r}}\rangle = \\
&= \iiint Y_m^\ell(\mathbf{r}) f^\ell(r) \sum_{\ell'} (-i)^{\ell'}(2\ell'+1) j_{\ell'}(kr)\mathbf{Y}^{\ell'}(\mathbf{r}) \bullet_0 \mathbf{Y}^{\ell'}(\mathbf{k}) r^2 \sin\theta \underbrace{\sqrt{\tfrac{2}{\pi}}\tfrac{1}{4\pi}}_{=(2\pi)^{-3/2}} d\theta d\varphi dr \\
&= \sqrt{\tfrac{2}{\pi}} \sum_{\ell'} \int f^\ell(r)(-i)^{\ell'}(2\ell'+1) j_{\ell'}(kr) r^2 dr \tfrac{1}{4\pi} \iint \mathbf{Y}^{\ell'}(\mathbf{r}) \bullet_0 \mathbf{Y}^{\ell'}(\mathbf{k}) Y_m^\ell(\mathbf{r}) \sin\theta d\theta d\varphi \\
&= \sqrt{\tfrac{2}{\pi}} \sum_{\ell'} \int f^\ell(r)(-i)^{\ell'}(2\ell'+1) j_{\ell'}(kr) r^2 dr \sum_{m'=-\ell'}^{m'=-\ell} Y_{m'}^{\ell'}(\mathbf{k}) \tfrac{1}{4\pi} \underbrace{\iint \overline{Y_{m'}^{\ell'}(\mathbf{r})} Y_m^\ell(\mathbf{r}) d\Omega}_{=\frac{4\pi}{2\ell+1}\delta_{\ell'\ell}\delta_{m'm}} \\
&= Y_m^\ell(\mathbf{k})(-i)^\ell \underbrace{\sqrt{\tfrac{2}{\pi}} \int f^\ell(r) j_\ell(kr) r^2 dr}_{=\alpha_\ell(k)} = Y_m^\ell(\mathbf{k})(-i)^\ell \alpha_\ell(k) \quad . \qquad (4.17)
\end{aligned}$$

With $\alpha(k) \in \mathbb{C}$ we denote the expansion coefficients of the Hankel transform of the radial component f^ℓ (see. Eq. (C.16)). After transforming component-by-component we get the identity

$$\widetilde{\mathbf{f}^\ell}(\mathbf{k}) = \langle \mathbf{f}^\ell, e^{i\mathbf{k}^T\mathbf{r}}\rangle = \mathbf{Y}^\ell(\mathbf{k})(-i)^\ell \alpha_\ell(k) \,. \qquad (4.18)$$

Having the Fourier transform $\widetilde{\mathbf{f}^\ell}$ of $\mathbf{f}^\ell$ we can compute the spherical up-derivative and down-derivative of $\widetilde{\mathbf{f}^\ell}$ in frequency domain. Using the Fourier representation of the spherical derivatives (Eq. (4.8)) we get

$$\begin{aligned}
&\widetilde{\boldsymbol{\nabla}^1 \mathbf{f}^\ell} = && (4.19) \\
&\quad i\mathbf{R}^1(\mathbf{k}) \bullet_{\ell+1} \widetilde{\mathbf{f}^\ell} = i\mathbf{R}^1(\mathbf{k}) \bullet_{\ell+1} \mathbf{Y}^\ell(\mathbf{k})(-i)^\ell \alpha_\ell(k) = ik\mathbf{Y}^{\ell+1}(\mathbf{k})(-i)^\ell \alpha_\ell(k) \quad \text{and} \\
&\widetilde{\boldsymbol{\nabla}_1 \mathbf{f}^\ell} = && (4.20) \\
&\quad i\mathbf{R}^1(\mathbf{k}) \bullet_{\ell-1} \widetilde{\mathbf{f}^\ell} = i\mathbf{R}^1(\mathbf{k}) \bullet_{\ell-1} \mathbf{Y}^\ell(\mathbf{k})(-i)^\ell \alpha_\ell(k) = ik\mathbf{Y}^{\ell-1}(\mathbf{k})(-i)^\ell \alpha_\ell(k) \quad .
\end{aligned}$$

Transforming back Eq. (4.19) and Eq. (4.20) into spatial domain we separately obtain the angular part and the radial part of the spherical derivatives of $\mathbf{f}^\ell$. Again, we transform component-by-component. For the up-derivatives we get

$$\begin{aligned}
(2\pi)^{-3/2}\langle ikY_m^{\ell+1}(\mathbf{k})(-i)^\ell \alpha_\ell(k), e^{-i\mathbf{k}^T\mathbf{r}}\rangle &= Y_m^{\ell+1}(\mathbf{r})(-1)\sqrt{\tfrac{2}{\pi}} \int k\alpha_\ell(k) j_{\ell+1}(kr) k^2 dk \\
&= Y_m^{\ell+1}(\mathbf{r}) \underbrace{\sqrt{\tfrac{2}{\pi}} \int \alpha_{\ell+1}(k) j_{\ell+1}(kr) k^2 dk}_{=f^{\ell+1}(r)} \quad ,
\end{aligned}$$

where $f^{\ell+1}$ is the Hankel transform of the radial component of $\boldsymbol{\nabla}^1 \mathbf{f}^\ell$ with expansion coefficients $\alpha_{\ell+1}(k) = -k\alpha_\ell(k)$. It follows that

$$\nabla^1 \mathbf{f}^\ell(\mathbf{r}) = \mathbf{Y}^{\ell+1}(\mathbf{r}) \underbrace{\sqrt{\tfrac{2}{\pi}} \int \alpha_{\ell+1}(k) j_{\ell+1}(kr) k^2 dk}_{=f^{\ell+1}(r)} . \tag{4.21}$$

Now we show that differentiation in spatial domain using Eq. (4.15) leads to the same result. For this we represent f^ℓ by its Hankel transform (consider Eq. (C.17) and Eq. (4.18)). That is $f^\ell(r) = \sqrt{\frac{2}{\pi}} \int_0^\infty \alpha_\ell(k) j_\ell(rk) k^2 dk$. Utilizing the differential relation of the Bessel functions (Eq. (C.14)) we can show that (we only consider the radial component here)

$$\begin{aligned}
r^\ell \frac{\partial}{\partial r} \frac{1}{r^\ell} f^\ell(r) &= r^\ell \frac{\partial}{\partial r} \frac{1}{r^\ell} \sqrt{\tfrac{2}{\pi}} \int_0^\infty \alpha_\ell(k) j_\ell(rk) k^2 dk \\
&= \sqrt{\tfrac{2}{\pi}} \int_0^\infty \alpha(k) \left(r^\ell \frac{\partial}{\partial r} \frac{1}{r^\ell} j_\ell(kr) \right) k^2 dk \\
&= \sqrt{\tfrac{2}{\pi}} \int_0^\infty \alpha(k) \left(-k j_{\ell+1}(kr) \right) k^2 dk \\
&= \sqrt{\tfrac{2}{\pi}} \int_0^\infty \underbrace{(-k\alpha(k))}_{=\alpha_{\ell+1}(k)} j_{\ell+1}(kr) k^2 dk = f^{\ell+1}(r) ,
\end{aligned} \tag{4.22}$$

which proves Eq. (4.15).

In a similar way we can prove Eq. (4.16). Transforming back Eq. (4.20) into spatial domain we get the Hankel expansion of the radial component:

$$\begin{aligned}
\nabla_1 \mathbf{f}^\ell(\mathbf{r}) &= \mathbf{Y}^{\ell-1}(\mathbf{r}) \sqrt{\tfrac{2}{\pi}} \int \underbrace{k\alpha_\ell(k)}_{=\alpha_{\ell-1}(k)} j_{\ell-1}(kr) k^2 dk . \\
&= \mathbf{Y}^{\ell-1}(\mathbf{r}) \underbrace{\sqrt{\tfrac{2}{\pi}} \int \alpha_{\ell-1}(k) j_{\ell-1}(kr) k^2 dk}_{=f^{\ell-1}(r)} .
\end{aligned} \tag{4.23}$$

Utilizing the differential relation of the Bessel functions (Eq. (C.15)) we can show that (we again only consider the radial component here)

$$\begin{aligned}
\frac{1}{r^{\ell+1}} \frac{\partial}{\partial r} r^{\ell+1} f^\ell(r) &= \frac{1}{r^{\ell+1}} \frac{\partial}{\partial r} r^{\ell+1} \sqrt{\tfrac{2}{\pi}} \int_0^\infty \alpha_\ell(k) j_\ell(rk) k^2 dk \\
&= \sqrt{\tfrac{2}{\pi}} \int_0^\infty \alpha(k) \left(\frac{1}{r^{\ell+1}} \frac{\partial}{\partial r} r^{\ell+1} j_\ell(kr) \right) k^2 dk \\
&= \sqrt{\tfrac{2}{\pi}} \int_0^\infty \alpha(k) \left(k j_{\ell-1}(kr) \right) k^2 dk \\
&= \sqrt{\tfrac{2}{\pi}} \int_0^\infty \underbrace{(k\alpha(k))}_{=\alpha_{\ell-1}(k)} j_{\ell-1}(kr) k^2 dk = f^{\ell-1}(r) ,
\end{aligned} \tag{4.24}$$

which proves Eq. (4.16). □

4.2. Discrete Spherical Tensor Derivatives

Discrete versions of the spherical derivative operators can be obtained by discretizing the spherical gradient operator $\nabla^{S_2} = (\frac{1}{\sqrt{2}}(\partial_x - i\partial_y), \partial_z, -\frac{1}{\sqrt{2}}(\partial_x + i\partial_y))^T$ (Eq. (4.1)). This requires a discretization of the ordinary partial derivatives $\partial_x, \partial_y, \partial_z$ in the Cartesian image coordinate system. Regarding computational efficiency we found that simple finite difference schemes yield sufficient accuracy for our applications. Figure 4.1 shows several alternatives that we have evaluated in an experiment (results are following later in this section). In Farid and Simoncelli (2004), one can find a survey of further possible alternatives.

Consequently, the spherical derivative operators $\boldsymbol{\nabla}^1$ and $\boldsymbol{\nabla}_1$ are weighted superpositions of ordinary Cartesian finite differences. A crucial point is the computation of the weights based on the Clebsch-Gordan coefficients. They can be computed once in advance, before applying the operators to an image in a voxel-by-voxel manner. The computation scheme can easily be derived from Eq. (4.7) and Eq. (C.31), Eq. (C.32) in the appendix.

Given a tensor field $\mathbf{f}^\ell \in \mathcal{T}_\ell$. We can compute its up-derivative component-by-component in the following way (Note, that we assume that $f^\ell_m = 0$ for all $|m| > \ell$):

$$\begin{aligned}
[\boldsymbol{\nabla}^1 \mathbf{f}^\ell]_m =& [(\nabla^{S_2} \bullet_{\ell+1} \mathbf{f}^\ell)]_m \\
=& \frac{1}{\sqrt{2}}(\partial_x - i\partial_y) f^\ell_{(m+1)} \frac{\langle 1(-1), \ell(m+1) \,|\, (\ell+1)m \rangle}{\sqrt{2}\langle 10, \ell 0 \,|\, (\ell+1)0 \rangle} \\
&+ \partial_z f^\ell_m \frac{\langle 10, \ell(m) \,|\, (\ell+1)m \rangle}{\langle 10, \ell 0 \,|\, (\ell+1)0 \rangle} \\
&- \frac{1}{\sqrt{2}}(\partial_x + i\partial_y) f^\ell_{(m-1)} \frac{\langle 11, \ell(m-1) \,|\, (\ell+1)m \rangle}{\sqrt{2}\langle 10, \ell 0 \,|\, (\ell+1)0 \rangle} \\
=& \frac{\sqrt{(\ell-m)(1+\ell-m)}}{2(\ell+1)}(\partial_x - i\partial_y) f^\ell_{(m+1)} \\
&+ \frac{\sqrt{(\ell+m+1)(\ell-m+1)}}{\ell+1}\partial_z f^\ell_m \\
&- \frac{\sqrt{(\ell+m)(1+\ell+m)}}{2(\ell+1)}(\partial_x + i\partial_y) f^\ell_{(m-1)} \quad .
\end{aligned} \tag{4.25}$$

This gives us algorithm 2 on page 223 for computing the spherical up-derivatives. For the down-derivatives we get

$$\begin{aligned}
[\boldsymbol{\nabla}_1 \mathbf{f}^\ell]_m =& [(\nabla^{S_2} \bullet_{\ell-1} \mathbf{f}^\ell)]_m \\
=& \frac{\sqrt{(\ell+m)(1+\ell-m)}}{2(\ell-1)}(\partial_x - i\partial_y) f^\ell_{(m+1)} \\
&+ \frac{\sqrt{(\ell+m)(\ell-m)}}{(\ell-1)}\partial_z f^\ell_m \\
&- \frac{\sqrt{(\ell-m)(1+\ell+m)}}{2(\ell-1)}(\partial_x + i\partial_y) f^\ell_{(m-1)} \quad ,
\end{aligned} \tag{4.26}$$

leading to algorithm 3 on page 223.

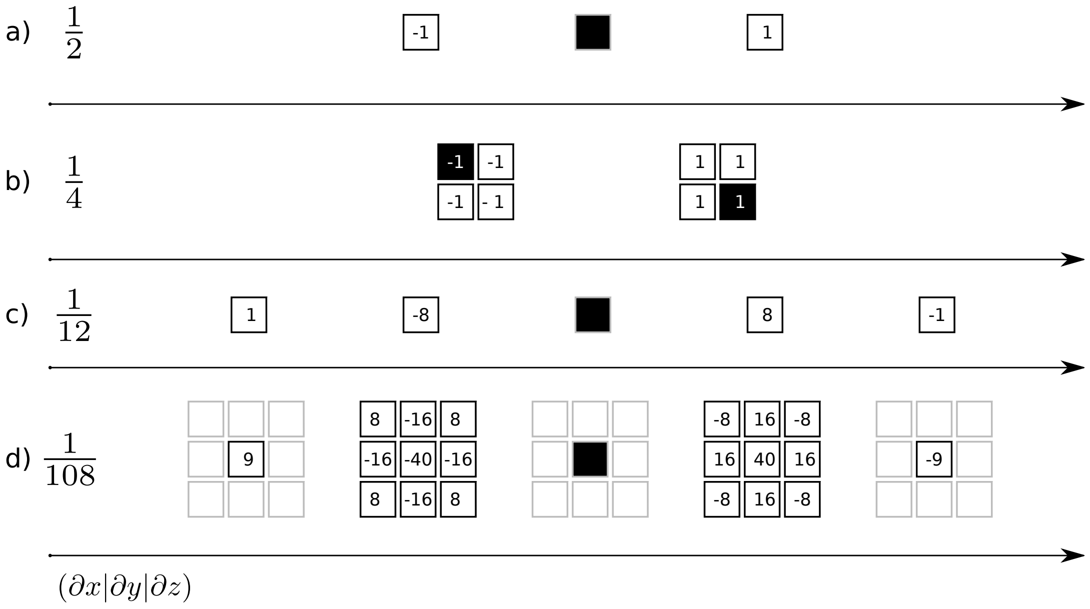


Figure 4.1.: Scheme a) and b) are the ordinary 2nd order central and forward-backward differences. Scheme c) is a 4th order central finite difference scheme. Scheme d) is a 4th order ball shaped scheme which is the most accurate scheme in our experiments.

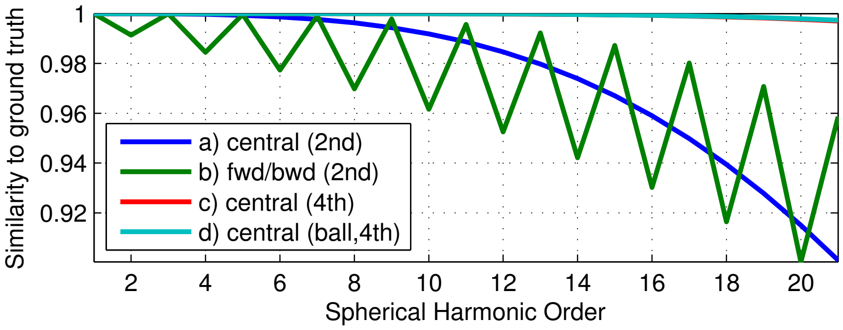
4.2.1. Experiment: Comparison of Finite Differences Schemes

In an experiment we compare four finite difference schemes with respect to accuracy and computation time. Spherical tensor derivatives in the context of image processing have been used first by (Reisert and Burkhardt 2009a). They used ordinary central differences as shown in Fig. 4.1 a) for computing the spherical gradient. We compare the existing discrete tensor derivative operators to three further alternatives showing that more accurate finite difference schemes improve the accuracy of the tensor derivatives significantly while keeping the computation time comparable.

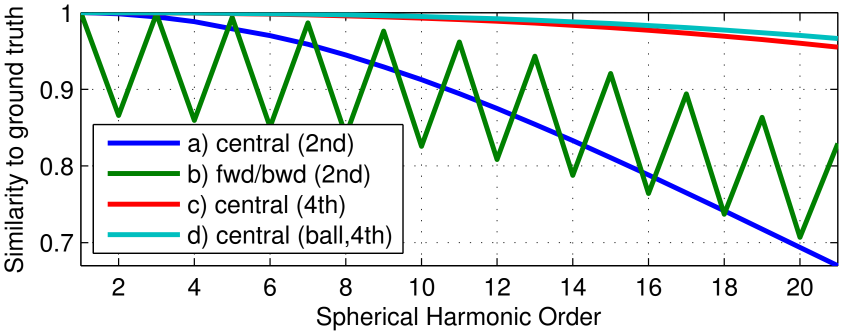
In our experiment we computed the derivatives in two manners. We first analytically derived the derivatives of a scalar valued spherical tensor field using Lemma 4.1.1 on page 100. Based on the analytical equation we rendered the derivatives into a voxel grid. In the second scenario we rendered the scalar valued function into a voxel grid and consecutively applied the discrete spherical tensor derivative operator to compute the derivatives numerically (algorithm 2). We then compared the two results using the normalized cross correlation.

Figure 4.1 shows the finite difference schemes. Scheme a) and b) are the ordinary 2nd order central and forward-backward differences. Scheme c) is a 4th order central finite difference scheme. We further use central differences based on the derivatives of a 4th order polynomial which we fit into a ball-shaped voxel neighborhood (scheme d)).

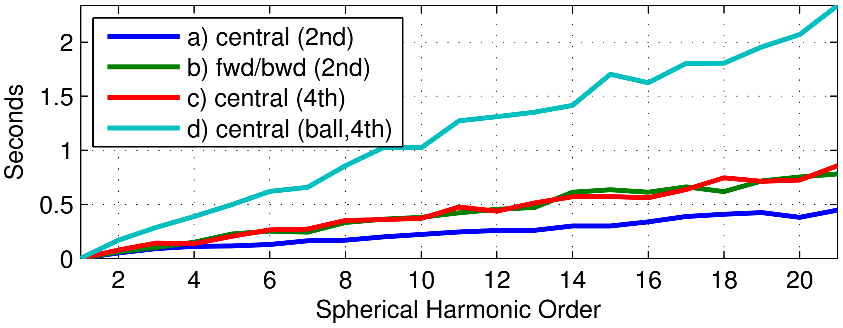
The two functions we used for evaluation are the derivatives of the 3D Gaussian (using $\sigma = 6$) and a high frequency spherical Bessel function (using $\sigma = 6$, $k = 2\pi$ and $s = \pi$). For both functions we know the analytical expression of their derivatives



(a) Accuracy (Gaussian Derivatives)



(b) Accuracy (Gauss Bessel)



(c) Computation Time (Using an Intel Quad Core i5-2400 CPU @ 3.1GHz)

Figure 4.2.: Comparison of accuracy and computation time of several differential operators based on finite differences.

(Details concerning the functions can be found in section 5.1 and section 5.2). We represented both functions in a $(2\ell+1)\times 128\times 128\times 128$ complex valued voxel grid (where ℓ is the tensor rank).

Figure 4.6 exemplarily shows centered slices of one component of the discretized analytical solution (using the GSL, Galassi et al. (2003)) together with the numerical solution based on the discrete tensor derivatives. Figure 4.2 shows the similarity of both approaches with respect to the normalized cross correlation. The curve shows that accuracy drops with the tensor order of the function. In Fig. 4.2(c) we depict

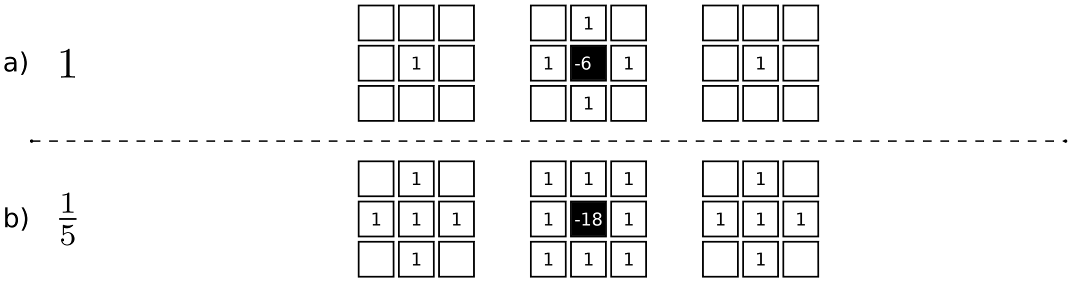


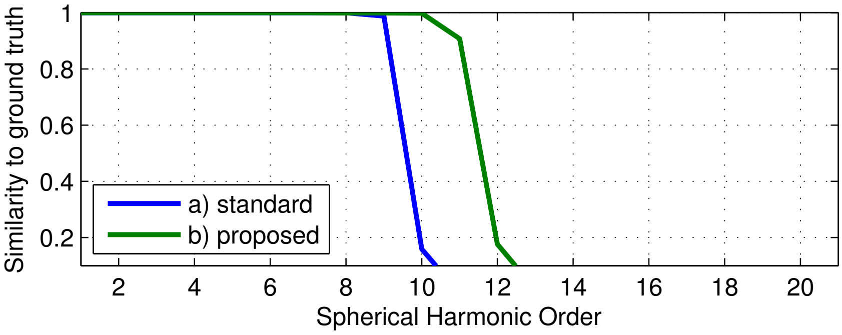
Figure 4.3.: a) The standard Laplace operator and b) the proposed Laplace operator.

the computation time for the different differentiation schemes.

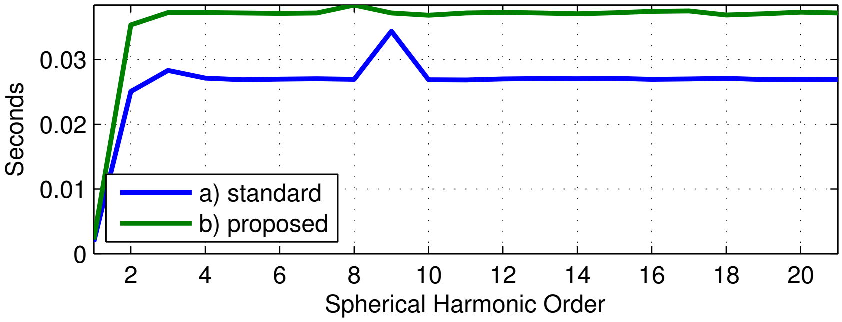
We observe that regarding the trade-off between computational efficiency and discretization accuracy the simple central difference of 2nd order accuracy (Fig. 4.1 a)) yields sufficient accuracy for derivatives up to order $\ell \leq 7$. The forward-backward scheme performs slightly better than the 2nd order central difference but comes with two major drawbacks: when using a forward-backward scheme we must deal with the alternating one-pixel shift. Furthermore, the computation time is comparable to the 4th order central differences and regarding accuracy the 4th order scheme is the favorable choice.

Very high order derivatives lead to artifacts when using only 2nd order finite differences. These artifacts can be avoided by using more accurate discrete differentiation schemes like the 4th order finite difference in Fig. 4.1 c) and d). Here the "ball-shaped" operator d) leads to the most accurate results. But compared to operator c) the accuracy doesn't differ significantly. Hence operator c) yields sufficient accuracy for derivatives up to order $\ell \leq 15$ while the computation time is comparable (about a factor of 2) to the much less accurate 2nd order central differences.

An Alternative to the Standard Laplace Operator. We made the observation that the ordinary Laplace operator (Fig. 4.3 a)) leads quite fast to numerical blow-up-effects when it is consecutively applied to an image (see Fig. 4.5 on the facing page). The Laplace operator always computes 2nd order derivatives based on a very small neighborhood operator. Therefore, after applying the operator more than eight times, we destroy the image due to numerical reasons. We propose to use a slightly larger neighborhood according to Fig. 4.3 b). As a result we can benefit from more stability while the computation time stays comparable; see Fig. 4.4.



(a) Accuracy



(b) Computation Time

Figure 4.4.: With a bit more computation time we achieve a higher accuracy using the proposed Laplace operator; see also Fig. 4.5.

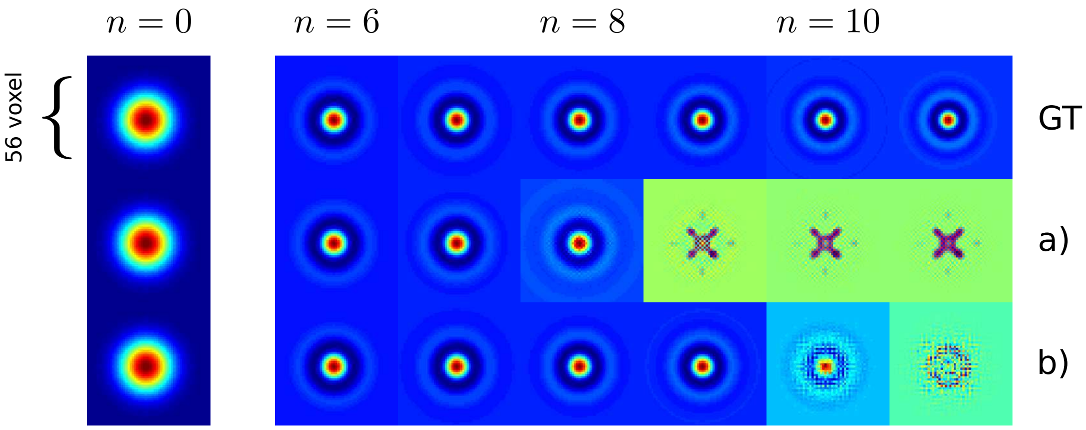


Figure 4.5.: a) Results when using the standard Laplace operator and b) when using the proposed operator; see also Fig. 4.4. The first row shows explicitly computed results.

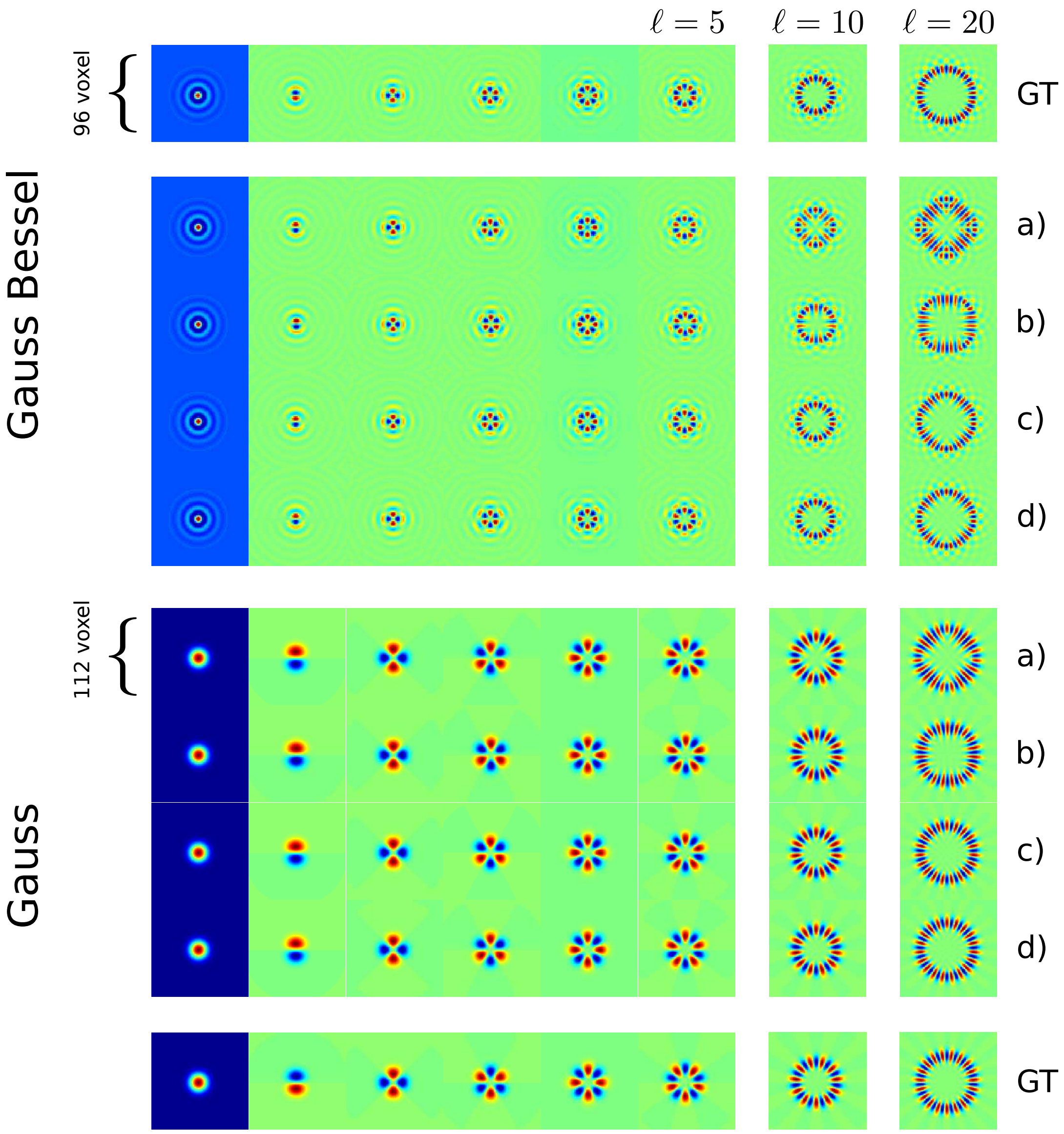


Figure 4.6.: Comparison of different finite difference schemes. The most outer tensor component for a tensor field of increasing order is shown. Upper image: derivatives of a higher frequency Gauss-Bessel kernel. Lower image: the spherical Gaussian derivatives. GT (ground truth): We analytically derived the derivatives and rendered them into a voxel grid. a-d) Numerically computed derivatives based on different finite difference schemes. a) 2^{nd} order central differences, b) 2^{nd} order forward/backward scheme, c) 4^{th} order central differences and d) 4^{th} order ball shaped scheme. Note that the functions are presented in a cubic image with an edge length of 128. The images have been cropped for this figure to 96 and 112 voxels, respectively.

5 $SE(3)$ Covariant Filters

The following chapter advocates three different kinds of $SE(3)$ covariant filters as an alternative to the "classical" dense spherical harmonic transformation $\mathcal{SH}^{\ell j}(r) : \mathcal{T}_J \to \mathcal{T}_{(\ell+j)}$; see Eq. (2.83) on page 60. The first three sections are covering the theoretical details of the transformations. A forth section will discuss important implementation details. The three transformations are

- ($\mathcal{LT}^{\ell j}\{n\} : \mathcal{T}_J \to \mathcal{T}_{(\ell+j-n)}$, **An $SE(3)$ Covariant Gauss-Laguerre Transform** is introduced in section 5.1 on the next page) The Gauss-Laguerre transformation extracts wavelet-like features based on spherical Gaussian derivatives. It extents the idea of "local jets" (Koenderink and Doorn 1987) to spherical tensor representations. A local jet is a feature vector of local derivatives. The parameter $n \in \mathbb{N}$ represents the polynomial degree of the radial Laguerre polynomial. We show how to efficiently design a filter for ordinary scalar valued images and extend it to tensor valued images.

- ($\mathcal{GT}^{\ell}\{k\} : \mathcal{T}_0 \to \mathcal{T}_\ell$, **An $SE(3)$ Covariant Spherical Gabor Transform** is introduced in section 5.2 on page 128) We further introduce a novel spherical Gabor transformation that allows for local frequency analysis. Similar to the Gauss-Laguerre transform, the proposed features can be computed efficiently via finite differences. The parameter $k \in \mathbb{R}$ represents the frequency of the radial wave function. The proposed filter maps scalar valued images to tensor fields. However, it is worth mentioning that the filter can easily be extended to tensor valued images, too.

- ($\mathcal{SHOG}^{\ell}_{w} : \mathcal{T}_0 \to \mathcal{T}_\ell$, **SHOG - Spherical Histograms of Oriented Gradients**. SHOG is introduced in section 5.3 on page 138) The SHOG transformation extents the idea of the structure tensor to higher order spherical tensors. While the ordinary structure tensor is a pure second order tensor, SHOG is represented by higher order tensors, too. Thanks to the higher order representation, a local gradient orientation distribution can be represented more precisely and allows for extracting discriminant features from volumetric images. The subscript w is an isotropic window function like a Gaussian function or Gaussian smoothed sphere.

5.1. An $SE(3)$ Covariant Gauss-Laguerre Transform

In the following section we introduce a Gauss-Laguerre transform which is efficient regarding computation time and memory usage. In our applications we use this transformation for representing local image patches of volumetric images in the Gauss-Laguerre domain. From such a representation we can analytically compute local rotation invariant features, such as the power-spectrum or the bi-spectrum, which can be used for object detection, object classification and image segmentation tasks. We show that the Gauss-Laguerre domain is optimal for representing local smooth Gaussian distributed volumetric images. We further show how the Gauss-Laguerre transformation can be extended to tensor valued images. This allows a fast transformation into the harmonic domain for tensor valued image modalities, too. Results of this section have partially been published in Skibbe et al. (2012).

The Gauss-Laguerre transform has explicitly been used for analyzing local structures in images in applications ranging from key-point detection (Sorgi et al. 2006) to advanced image filters (Reisert and Burkhardt 2008a). A theoretical discussion on this kind of functions regarding image processing can be found in the paper by Koenderink and Doorn (1992). We show that the basis functions which are spanning the Gauss-Laguerre domain, the Gauss-Laguerre functions, are the (spherical) up- and down-derivatives of the 3D Gaussian function. A subspace of the 3D Gauss-Laguerre domain is spanned by the spherical Gaussian (up-)derivatives (Reisert and Burkhardt 2009a). Their 3D applications include generic object detection (Liu et al. 2012; Reisert and Burkhardt 2009a; Schlachter et al. 2010; Skibbe and Reisert 2012b), image filtering (Reisert and Burkhardt 2009a), or object segmentation (Skibbe and Reisert 2011; Skibbe et al. 2011a). Moreover, the projections of local image patches onto these basis functions, i.e. the Gauss-Laguerre expansion coefficients, can be considered as the spherical counterpart to a "local jet" (Koenderink and Doorn 1987), a vector containing the local Cartesian derivatives of a Gaussian smoothed image. Applications involving local jets are ranging from interest point detection (Knapek et al. 2000) and image feature detection (Lillholm and Pedersen 2004), to image retrieval tasks based on local image patches (Schmid and Mohr 1997). Local rotation invariants can be formed from local jets by using tensor contraction leading to a vector containing (among others) the local average intensity value, the squared gradient magnitude, the trace (the Laplacian) and determinant of the Hessian matrix (Florack et al. 1994; Haar Romeny et al. 1992; Koenderink and Doorn 1987; Lindeberg 1993; Schmid and Mohr 1997). The images are often initially convolved with Gaussian functions of different sizes allowing for analyzing images in different scales (see particularly the paper by Lindeberg (1990)).

We propose an $SE(3)$ covariant filter that is based on the Gauss-Laguerre transformation: we use the 3D convolution to project local image pages onto the spherical Gauss-Laguerre basis functions in a voxel-by-voxel manner. Furthermore, we make

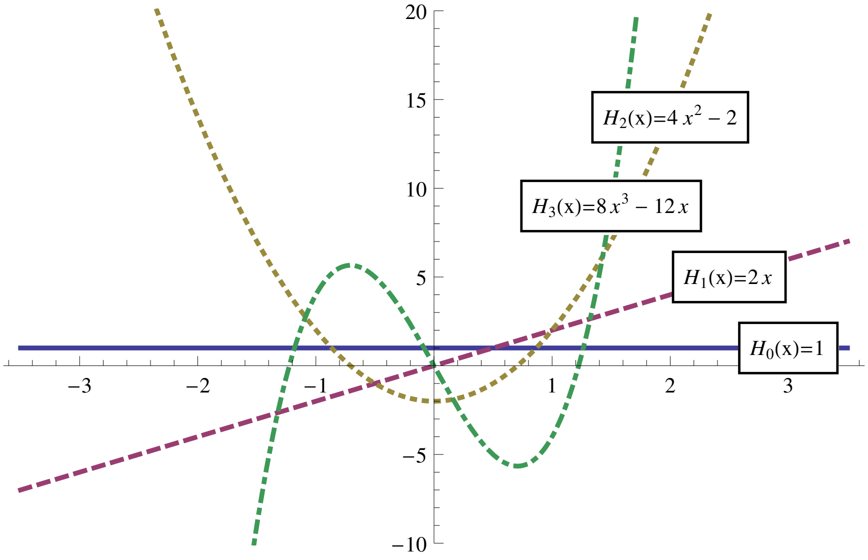


Figure 5.1.: Hermite polynomials with polynomial degree $n = 0, 1, 2, 3$.

use of the fact that the Gauss-Laguerre basis functions are the spherical derivatives of the 3D Gaussian function. Since the convolution and spherical differentiation are both linear operations, we can compute the local expansion coefficients efficiently by first smoothing the image with a Gaussian function and then locally computing the derivatives.

The 1D counterparts of the Gauss-Laguerre polynomials are the Hermite polynomials (see e.g. Koenderink and Doorn (1992)). Therefore, we use a simplified 1D scenario as an introduction to illustrate the idea behind the fast Gauss-Laguerre transform:

5.1.1. A Fast Hermite Transform for Discrete 1D Signals

Let $f \in L_2(\mathbb{R})$ be a continuous, differentiable function. The derivatives of f are discriminative features, because they are analytically describing the local characteristics of f. Moreover, we know from the Taylor expansion that f can be completely recovered by only considering the derivatives at an arbitrary point of expansion. Furthermore, from scale space theory we know that f can be studied in different granularities by initially smoothing f with an appropriate Gaussian function. The smaller the Gaussian width, the richer the local details covered by the lower order derivatives of f.

Let

$$a_n(x) = \frac{\partial^n}{\partial^n x} \underbrace{\int_{-\infty}^{\infty} f(r) e^{-(x-r)^2} dr}_{=(f*G)(x)} \tag{5.1}$$

be the nth derivative of the Gaussian smoothed function f, evaluated at the position

x (For brevity we assume w.l.o.g a Gaussian function with $G(r) := e^{-r^2}$). Then

$$(f * G)(r) = \sum_{n=0}^{\infty} a_n(x) \frac{(r-x)^n}{n!} \tag{5.2}$$

is the Taylor expansion of the smoothed function. The vector $(a_0(x), a_1(x), \cdots, a_N(x))^T$ is also known as a (scale dependent) local jet of f at position x. Because of the linearity of convolution and differentiation, we can obtain the same Taylor expansion coefficients by computing the derivatives of the Gaussian and convolving the initial function with the Gaussian derivatives, namely

$$\begin{aligned} a_n(x) &= \frac{\partial^n}{\partial^n x} \int_{-\infty}^{\infty} f(r) e^{-(x-r)^2} dr = \int_{-\infty}^{\infty} f(r) \underbrace{\frac{\partial^n}{\partial^n x} e^{-(x-r)^2}}_{=H_n(x-r)e^{-(x-r)^2}} dr \\ &= \int_{-\infty}^{\infty} f(r) H_n(x-r) e^{-(x-r)^2} dr = (f * (H_n \cdot G))(x). \end{aligned} \tag{5.3}$$

The Gaussian derivatives are known as the Gaussian weighted Hermite polynomials. We denote the Hermite polynomial by $H_n : \mathbb{R} \to \mathbb{R}$. With $(H_n \cdot G)(r) = H_n(r)e^{-(r)^2}$ we denote the point-by-point multiplication of a Hermite polynomial and the Gaussian function. The Hermite polynomials can be defined in a recursive manner, where

$$H_{n+1}(r)e^{-r^2} = \frac{\partial}{\partial r} H_n(r) e^{-r^2} \quad , \tag{5.4}$$

with $H_0 := 1$. The first four Hermite polynomials are $\{1, 2r, 4r^2 - 2, 8r^3 - 12r, \ldots\}$ (see Fig. 5.1 on the preceding page). For reflections about the origin we have $H_n(x) = (-1)^n H_n(-x)$ (see e.g. Abramowitz and Stegun (1964) for further details regarding Hermite polynomials). The Hermite polynomials $H_n : \mathbb{R} \to \mathbb{R}$ are a set of orthogonal polynomials, building an orthogonal basis for $L_2(\mathbb{R}, \mu)$. The subscript $n \in \mathbb{N}$ denotes the degree. The polynomials are orthogonal in the range $(-\infty, \infty)$ with respect to the Gaussian weighting function e^{-r^2}. This is formally expressed by

$$\langle H_m, H_n \rangle_\mu = \int_{-\infty}^{\infty} H_m(r) H_n(r) e^{-r^2} dr = \delta_{m,n} 2^n n! \sqrt{\pi}. \tag{5.5}$$

By $\delta_{m,n}$ we denote the Kronecker delta (equals 1 iff $m = n$, else 0). By orthogonal projection of f into the space spanned by the Gaussian weighted polynomials H_n we obtain the representation of f in terms of expansion coefficients

$$a_n(x) := \langle \mathfrak{t}_{(-x)} f, H_n \rangle_\mu = \int_{-\infty}^{\infty} f(r+x) H_n(r) e^{-r^2} dr = (-1)^n \int_{-\infty}^{\infty} f(r) H_n(x-r) e^{-(x-r)^2} dr, \tag{5.6}$$

where x is the center of expansion. In Fig. 5.2 we show the Gaussian weighted Hermite polynomials. Due to the orthogonality of the polynomials it holds that

$$f(r) = \sum_{n=0}^{\infty} \frac{1}{2^n n! \sqrt{\pi}} a_n(x) H_n(r-x) \,. \tag{5.7}$$

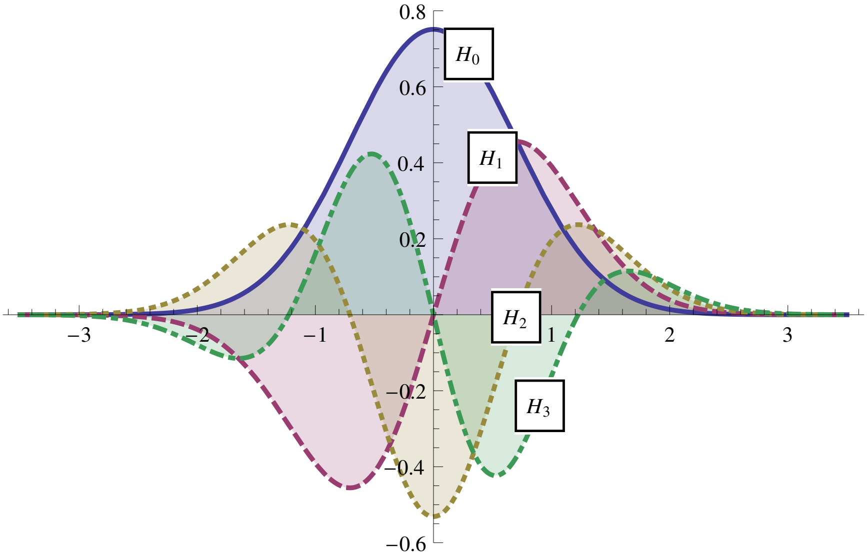


Figure 5.2.: Gaussian weighted Hermite polynomials $\frac{1}{\sqrt{2^n n! \sqrt{\pi}}} H_n(x) e^{-x^2}$ for an increasing degree $n = 0, 1, 2, 3$.

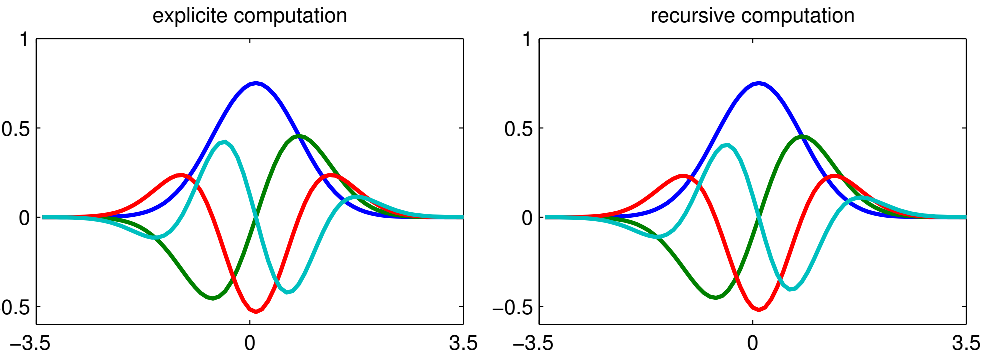
Consequently, the derivatives of a Gaussian smoothed function f are both the Taylor expansion coefficients of $(f * G)$ and the expansion coefficients of a polynomial Hermite expansion of f (compare Eq. (5.2) with Eq. (5.7)).

Suppose we aim at implementing a Hermite transformation for discrete 1D signals. Furthermore, suppose we aim at computing the coefficients in a point-by-point manner (computing $a_n(x)$ for all x) so that we can analytically locally analyze the signal by its local expansion coefficients. In this scenario we can highly benefit from the differential relationship of the Hermite polynomials: for a discrete signal f, we can realize a point-wise expansion in terms of H_n by only one initial smoothing realized via a FFT, followed by a consecutive computation of point-wise finite differences, namely

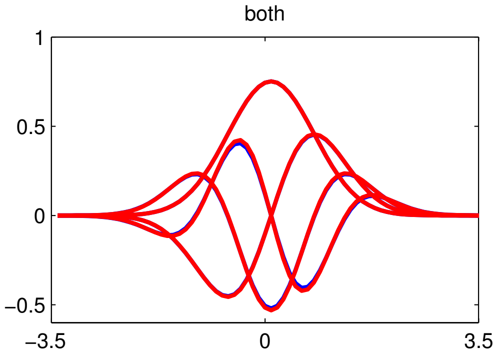
$$\begin{aligned} a_n(x) &= \langle \mathfrak{h}_x f, H_n \rangle_\mu \\ &= (-1)^n (f * (H_n \cdot G))(x) \\ &= (-1)^n (f * \tfrac{\partial^n}{\partial^n x} G)(x) \\ &= (-1)^n \tfrac{\partial^n}{\partial^n x} (f * G)(x). \end{aligned} \tag{5.8}$$

The resulting scheme for computing the expansion coefficients is illustrated in the following:

$$\underbrace{\frac{\partial}{\partial x} \cdots \underbrace{\frac{\partial}{\partial x} \underbrace{(f * G)(x)}_{=a_0(x)}}_{=-a_1(x)}}_{=(-1)^n a_n(x)} \quad . \tag{5.9}$$



(a) Explicit (b) Differential based



(c) Overlay (left red/right blue)

Figure 5.3.: a) Numerically evaluated Hermite polynomials using `Matlab` . b) We used central differences to consecutively derive the functions from the Gaussian (blue curve). Discretization steps: 0.1.

Figure 5.3 shows results when choosing as f the delta function.

In the same way as shown for the 1D scenario we can utilize the spherical derivative operator to expand a volumetric image voxel-by-voxel in terms of Gaussian weighted spherical functions having a differential relationship with respect to the spherical derivative operator $\boldsymbol{\nabla}$.

The initial Gaussian smoothing gives us two further advantages. First, the convolution with a Gaussian (or generally a Gaussian weighted continuously differentiable function) with a discrete signal ensures that the resulting signal itself is continuously differentiable. Furthermore, the Gaussian window allows for performing a local multi-scale analysis of the signal by utilizing kernels of different scale.

5.1.2. A Spherical Gauss Laguerre Transform for Discrete Images

Similarly to the example in the introduction where the Hermite polynomials arise from the Gaussian by differentiation, we obtain the orthogonal spherical Laguerre polynomials by successively applying the spherical derivative operators to the 3D Gaussian function. We will see that the angular parts of the spherical Gaussian derivatives are the spherical harmonic functions while the radial parts are Gaussian windowed Laguerre polynomials. These functions are highly related to the 2D complex derivatives of the 2D Gaussian (Reisert and Burkhardt 2008a), where the Gaussian windowed Laguerre polynomials arise from the Gaussian, too. The only differences to the 3D functions are that in 2D the angular functions are the circular harmonics and the Laguerre functions are parametrized accordingly.

The 3D Gaussian derivatives are widely known as Laguerre Gaussian-type functions in the field of theoretical chemistry (Chow Chiu L.-Y. 2001; Matsuoka 1998). In the image processing community they are used for scale and orientation analysis of images, see particularly the work of Koenderink and Doorn (1992) and have been used in a variety of applications like detection and/or classification (Liu et al. 2011b; Liu et al. 2012; Reisert and Burkhardt 2009a; Schlachter et al. 2010; Skibbe and Reisert 2011; Skibbe and Reisert 2012b; Skibbe et al. 2011a)).

Similarly to the Hermite expansion, changing the order of convolution and differentiation also works for the spherical Laguerre polynomials. Using the spherical Laguerre basis function we design a new $SE(3)$ covariant spherical harmonic based filter. The proposed filter implements an efficient spherical harmonic transform of volumetric images with almost linear complexity.

Definition 5.1.1 (Spherical Laguerre Polynomials)**.** *The 3D spherical Laguerre polynomials* $\mathcal{L}_n^\ell : \mathbb{R}^3 \times \mathbb{R} \to \mathbb{C}^{2(\ell-n)+1}$ *of polynomial degree* $\ell + n$ *are defined by*

$$\begin{aligned} \mathcal{L}_n^\ell(\mathbf{r},t) &:= \mathbf{Y}^{\ell-n}(\mathbf{r}) r^{\ell-n} L_n^{(\ell-n)+\frac{1}{2}}\left(\tfrac{r^2}{2t}\right) \\ &= \mathbf{R}^{\ell-n}(\mathbf{r}) L_n^{(\ell-n)+\frac{1}{2}}\left(\tfrac{r^2}{2t}\right), \end{aligned} \tag{5.10}$$

where $\ell, n \in \mathbb{N}_0$, $\mathbf{r} = (x,y,z)^T$ *and* $r^2 = \mathbf{r}^T\mathbf{r}$.

We denote by $L_n^\alpha : \mathbb{R} \to \mathbb{R}$ the Laguerre polynomial of order n associated with $\alpha \in \mathbb{R}_{\geq 0}$ (see Eq. (8) in the appendix). The parameter $t \in \mathbb{R}_{>0}$ corresponds to the scale or with $\sigma = \sqrt{t}$ to the standard deviation. We often neglect the second parameter, write $\mathcal{L}_n^\ell(\mathbf{r})$ and consider t w.o.l.g. to be constant. For details related to scale-space theory we refer to Lindeberg (1990). With $[\mathcal{L}_n^\ell]_m : L_2(\mathbb{R}^3)$, $|m| \leq (\ell - n)$ we denote the single components of $\mathcal{L}_n^\ell$. The polynomial degree of $\mathcal{L}_n^\ell(\mathbf{r})$ is $\ell + n$; on page 202 you can find a derivation for that fact.

The polynomials $[\mathcal{L}_n^\ell]_m : \mathbb{R}^3 \to \mathbb{C}$ are orthogonal with respect to the Gaussian weighting function $e^{-\frac{r^2}{2t}}$ spanning $L_2(\mathbb{R}^3, \mu)$, whereas

$$\langle [\mathcal{L}_n^\ell]_m, [\mathcal{L}_{n'}^{\ell'}]_{m'} \rangle_\mu = \underbrace{t^{(\ell-n)} \frac{(2\pi t)^{\frac{3}{2}} (2\ell+1)!!}{2^n n! (2(\ell-n)+1)}}_{=\frac{1}{\alpha(\ell,n,t)}} \delta_{\ell,\ell'} \delta_{m,m'} \delta_{n,n'} = \frac{\delta_{\ell,\ell'} \delta_{m,m'} \delta_{n,n'}}{\alpha(\ell,n,t)} \ . \tag{5.11}$$

With $\alpha(\ell,n,t) \in \mathbb{R}$ we denote the normalization factor, by $\langle \cdot,\cdot \rangle_\mu$ the Gaussian weighted inner product and !! denotes the double factorial. A proof of Eq. (5.11) can be found in section B.3.1 on page 202.

5.1.2.1. The Spherical Gauss Laguerre Transform

Projecting an image into the space spanned by the Gaussian windowed Laguerre polynomials is known as Gauss-Laguerre transform (see e.g. Koenderink and Doorn

(1992), or Lue-Yung Chow Chiu (1999)).

Given an image $I \in L_2(\mathbb{R}^3)$. We consider w.o.l.g. this image being a spherical tensor field $I \in \mathcal{T}_0$ of rank 0. Since the basis functions $\boldsymbol{\mathcal{L}}_n^\ell$ are spanning the function space of volumetric images, we can represent any 3D image I in terms of a linear combination of the basis functions $\boldsymbol{\mathcal{L}}_n^\ell$. Note that the literature distinguishes between two kinds of transformations which can be considered as equivalent (we will see that there exists an invertible mapping between both representations).

Half-weighted Transformation. If the Gaussian weight $e^{-\frac{r^2}{2t}}$ is equally split so that it equally weights the forward transformation and the backward transformation ($e^{-\frac{r^2}{2t}} = e^{-\frac{r^2}{4t}}e^{-\frac{r^2}{4t}}$) we call the transformation the half-weighted transformation (see particularly Koenderink and Doorn (1992) for image processing purposes). The functions together with the half Gaussian weight are known as orthogonal Laguerre functions. They are defined by

$$\boldsymbol{L}_n^\ell(\mathbf{r},t) := \boldsymbol{\mathcal{L}}_n^\ell(\mathbf{r},t)e^{-\frac{r^2}{4t}}. \tag{5.12}$$

Note that $\langle [\boldsymbol{L}_n^\ell]_m, [\boldsymbol{L}_{n'}^{\ell'}]_{m'}\rangle = \langle [\boldsymbol{\mathcal{L}}_n^\ell]_m, [\boldsymbol{\mathcal{L}}_{n'}^{\ell'}]_{m'}\rangle_\mu$. In this scenario the forward transformation is given by

$$\mathbf{v}_n^\ell := \langle I, \alpha^{\frac{1}{2}}(\ell,n,t)\overline{\boldsymbol{L}_n^\ell}\rangle = \alpha^{\frac{1}{2}}(\ell,n,t)\int_{\mathbf{r}\in\mathbb{R}^3} I(\mathbf{r})\boldsymbol{L}_n^\ell(\mathbf{r})d\mathbf{r}\,. \tag{5.13}$$

With $\mathbf{v}_n^\ell \in \mathbb{C}^{2(\ell-n)+1}$ we denote the spherical harmonic expansion coefficients of the half-weighted transformation. The expansion is given by

$$I(\mathbf{r}) = \sum_{n\leq\ell} \alpha^{\frac{1}{2}}(\ell,n,t)(\mathbf{v}_n^\ell)^T\overline{\boldsymbol{L}_n^\ell(\mathbf{r})} \quad . \tag{5.14}$$

Fully-weighted Transformation. In our applications we aim at forming invariant descriptors from the expansion coefficients. Hence we are particularly interested in a fast and efficient transformation of local image patches into the Gauss-Laguerre domain. Such a transformation exists for the fully Gaussian weighted transformation (details concerning the fast transformation are following later in this section). The back transformation plays a minor role in our context.

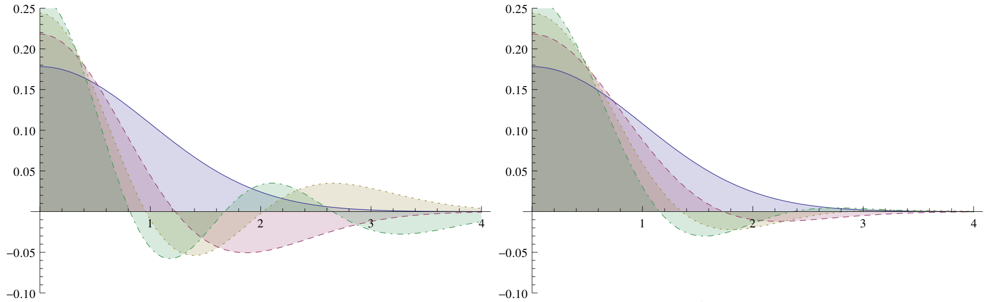
We propose a fully weighted transformation, where the full Gaussian weights the forward transformation while the Gaussian vanishes in the back transformation. In this scenario an image can be expanded in terms of the spherical Laguerre polynomials $\boldsymbol{\mathcal{L}}_n^\ell$ according to

$$I(\mathbf{r}) = \sum_{n\leq\ell} \alpha^{\frac{1}{2}}(\ell,n,t)(\mathbf{w}_n^\ell)^T\overline{\boldsymbol{\mathcal{L}}_n^\ell(\mathbf{r})} \quad . \tag{5.15}$$

The expansion coefficients $\mathbf{w}_n^\ell \in \mathbb{C}^{2(\ell-n)+1}$ of the fully-weighted transformation can be computed via the (fully) Gaussian weighted inner product of the image and the basis functions:

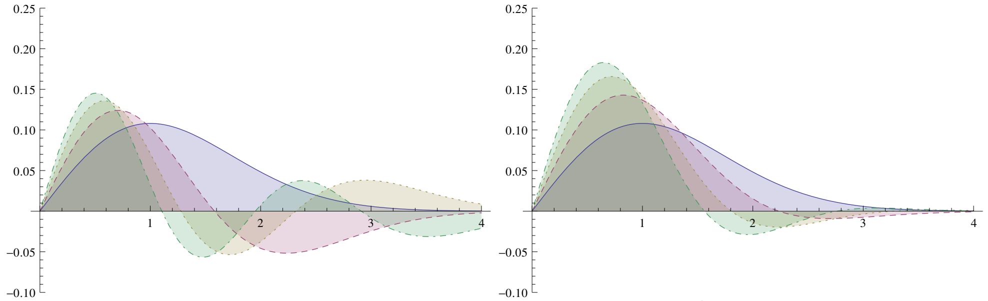
$$\mathbf{w}_n^\ell := \langle I, \alpha^{\frac{1}{2}}(\ell,n,t)\overline{\boldsymbol{\mathcal{L}}_n^\ell}\rangle_\mu = \alpha^{\frac{1}{2}}(\ell,n,t)\int_{\mathbf{r}\in\mathbb{R}^3} I(\mathbf{r})\boldsymbol{\mathcal{L}}_n^\ell(\mathbf{r})e^{\frac{-r^2}{2t}}d\mathbf{r}\,. \tag{5.16}$$

Corresponding to a spherical harmonic of order $\ell = 0$



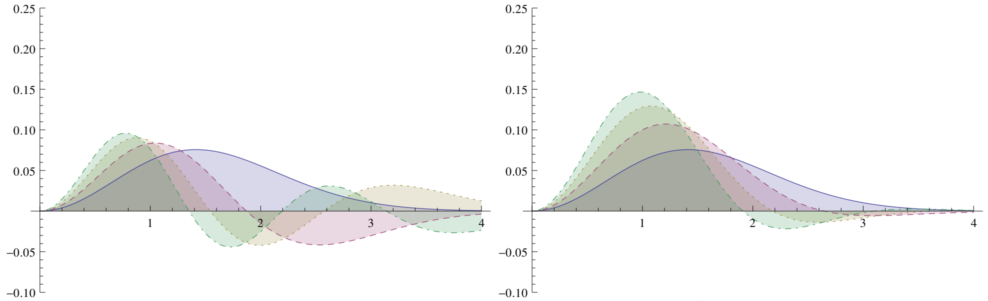
(a) $\boldsymbol{L}_i^{\ell+i}(\mathbf{r}, \frac{t}{2})$ for $\ell = 0$ and $i = 0..3$ (b) $\boldsymbol{\mathcal{L}}_i^{\ell+i}(\mathbf{r})e^{-\frac{r^2}{2t}}$ for $\ell = 0$ and $i = 0..3$

Corresponding to a spherical harmonic of order $\ell = 1$



(c) $\boldsymbol{L}_i^{\ell+i}(\mathbf{r}, \frac{t}{2})$ for $\ell = 1$ and $i = 0..3$ (d) $\boldsymbol{\mathcal{L}}_i^{\ell+i}(\mathbf{r})e^{-\frac{r^2}{2t}}$ for $\ell = 1$ and $i = 0..3$

Corresponding to a spherical harmonic of order $\ell = 2$



(e) $\boldsymbol{L}_i^{\ell+i}(\mathbf{r}, \frac{t}{2})$ for $\ell = 2$ and $i = 0..3$ (f) $\boldsymbol{\mathcal{L}}_i^{\ell+i}(\mathbf{r})e^{-\frac{r^2}{2t}}$ for $\ell = 2$ and $i = 0..3$

Figure 5.4.: The radial profiles of the Gauss-Laguerre functions $\boldsymbol{L}_i^{\ell+i}(\mathbf{r}, \frac{t}{2})$ on the left hand side are just a linear combination of the Gaussian weighted Laguerre polynomials $\boldsymbol{\mathcal{L}}_i^{\ell+i}(\mathbf{r}, t)e^{-\frac{r^2}{2t}}$ on the right hand side (and vice versa). They are spanning the same function space.

Theorem 5.1.2. *The basis functions* $\{\boldsymbol{L}_i^{\ell+i}(\mathbf{r},\frac{t}{2})\}_{i=\{0,\cdots,m\}}$ *and* $\{\boldsymbol{\mathcal{L}}_i^{\ell+i}(\mathbf{r},t)e^{-\frac{r^2}{2t}}\}_{i=\{0,\cdots,m\}}$, $m \leq \ell$, *are spanning the same function space.*

Hence if the Gaussian width is identical for both transformations, then the result of a truncated expansion with respect to the polynomial degree $(\ell+n) \leq N$ is the same for both expansions Eq. (5.15) and Eq. (5.14). That is,

$$\begin{aligned} \hat{I}(\mathbf{r}) &= \sum_{\substack{n\leq\ell \\ (n+\ell\leq N)}} \alpha^{\frac{1}{2}}(\ell,n,t)(\mathbf{v}_n^{\ell})^T \overline{\boldsymbol{L}_n^{\ell}(\mathbf{r},t/2)} \\ &= \sum_{\substack{n\leq\ell \\ (n+\ell\leq N)}} \alpha^{\frac{1}{2}}(\ell,n,t)(\mathbf{w}_n^{\ell})^T \overline{\boldsymbol{\mathcal{L}}_n^{\ell}(\mathbf{r},t)}\,. \end{aligned} \tag{5.17}$$

Consequently, the Mean-Square Error $E(\|\hat{I}-I\|^2)$ is identical for both types of truncated transformations.

Proof. There exists an invertible linear mapping between the expansion coefficients corresponding to spherical harmonic order ℓ of the half-weighted Laguerre transform to their fully-weighted counterparts: we can always express the expansion coefficients $\{\mathbf{v}_i^{(\ell+i)}\}_{i=\{0,\cdots,m\}}$, $\mathbf{v}_i^{(\ell+i)} \in \mathbb{C}^{2\ell+1}$ in terms of the coefficients $\{\mathbf{w}_i^{(\ell+i)}\}_{i=\{0,\cdots,m\}}$, $\mathbf{w}_i^{(\ell+i)} \in \mathbb{C}^{2\ell+1}$ and verse visa, where $m \leq \ell$. Since spherical harmonics of different order are orthogonal, we only need to consider coefficients sharing the same spherical harmonic order. We can show that all coefficients up to a radial polynomial degree m of the half-weighted transformation do only depend on the radial polynomials of up to order $i \leq m$ of the fully-weighted transformation (and verse visa). The connection between the coefficients is given by a linear invertible mapping M, where

$$\left[\mathbf{v}_{\ell}^{2\ell},\mathbf{v}_{\ell-1}^{2\ell-1},\cdots,\mathbf{v}_0^{\ell}\right]^T = \mathbf{M}\left[\mathbf{w}_{\ell}^{2\ell},\mathbf{w}_{\ell-1}^{2\ell-1},\cdots,\mathbf{w}_0^{\ell}\right]^T \quad . \tag{5.18}$$

The matrix

$$M = \begin{pmatrix} \binom{2\ell+\frac{1}{2}}{\ell}(-2)^0 & \binom{2\ell+\frac{1}{2}}{\ell-1}(-2)^1 & \dots & \binom{2\ell+\frac{1}{2}}{0}(-2)^{\ell} \\ 0 & \binom{(2\ell-1)+\frac{1}{2}}{\ell-1}(-2)^0 & \dots & \binom{(2\ell-1)+\frac{1}{2}}{0}(-2)^{(\ell-1)} \\ \ddots & \ddots & \ddots & \ddots \\ 0 & 0 & \dots & \binom{\ell+\frac{1}{2}}{0}(-2)^0 \end{pmatrix}_{\ell\times\ell} \begin{pmatrix} (-1)^{\ell} & 0 & \dots & 0 \\ 0 & (-1)^{(\ell-1)} & \dots & 0 \\ \ddots & \ddots & \ddots & \\ 0 & 0 & \ddots & 1 \end{pmatrix}_{\ell\times\ell} \quad . \tag{5.19}$$

is an invertible upper triangle matrix. Note that the transformation is not unitary.

We derive the mapping by using the existing relation between the half-weighted and the fully-weighted Laguerre polynomials, which is

$$\boldsymbol{L}_n^{\ell+n}(\mathbf{r},\tfrac{t}{2}) = (-1)^n \sum_{i=0}^{n} \binom{\ell+n+\frac{1}{2}}{n-i}(-2)^i \boldsymbol{\mathcal{L}}_i^{\ell+i}(\mathbf{r},t)e^{-\frac{r^2}{2t}} \quad . \tag{5.20}$$

We obtain Eq. (5.20) by using Eq. (C.26) in the appendix and multiplying both sides with $r^{(\ell-n)}\mathbf{Y}^{(\ell-n)}(\mathbf{r})e^{-\frac{r^2}{2t}}$ and choosing the indexes accordingly.

Consequently, the set of functions with increasing radial polynomial degree having the same spherical harmonic order, namely $\{\boldsymbol{L}_i^{\ell+i}(\mathbf{r},\frac{t}{2})\}_{i=\{0,\cdots,m\}}$ and

$\{\mathcal{L}_i^{\ell+i}(\mathbf{r},t)e^{-\frac{r^2}{2t}}\}_{i=\{0,\cdots,m\}}$, $m \leq \ell$, are spanning the same function space. By plugging Eq. (5.20) into the half-weighted forward transformation Eq. (5.13) we can express the expansion coefficients $\mathbf{v}_n^{\ell+n}$ in terms of the coefficients $\{\mathbf{w}_i^{(\ell+i)}\}_{i=\{0,\cdots,n\}}$ (and verse visa $\mathbf{w}_n^{\ell+n}$ via $\{\mathbf{v}_i^{(\ell+i)}\}_{i=\{0,\cdots,n\}}$):

$$\mathbf{v}_n^{\ell+n} = (-1)^n \sum_{i=0}^{n} \binom{\ell+n+\frac{1}{2}}{n-i} (-2)^i \mathbf{w}_i^{\ell+i} \quad . \tag{5.21}$$

□

Theorem 5.1.3. *The Gauss-Laguerre transform (Theorem 5.1.2.1 on page 115) is optimal for local smooth processes.*

Proof. We prove theorem 5.1.3 by showing that the Gauss-Laguerre functions $\boldsymbol{L}_n^\ell$ (Eq. (5.12)) are diagonalizing the covariance function of a local smooth process. This is sufficient for concluding that the fully-weighted transformation is optimal, too (see Eq. (5.17)). Optimal means that the resulting expansion of the original signal in terms of the optimal basis minimizes the total Mean-Square Error resulting from its truncation (with respect to the expansion coefficients we can say that the optimal transformation compacts the energy).

We simplify our proof by utilizing the fact that there exists a unitary transformation that maps the Gauss-Laguerre functions $\boldsymbol{L}_n^\ell$ to the Gaussian windowed 3D Hermite polynomials (see Koenderink and Doorn (1992)). The 3D Gaussian windowed Hermite polynomials again are a separable product of three 1D Gaussian windowed Hermite polynomials with respect to the Cartesian coordinates x,y and z. Hence it is sufficient to show that the 1D Gaussian windowed Hermite polynomials are optimal for representing local smooth 1D processes in order to conclude that the Gauss-Laguerre functions are optimal for representing local smooth 3D processes. Note that the current proof also offers an alternative to the proof in Reisert and Burkhardt (2008a) because we can draw the same conclusions for the 2D Gaussian windowed Laguerre polynomials (see also Koenderink and Doorn (1992) for the relationship between the 2D Gaussian windowed Laguerre polynomials and Gaussian windowed Hermite polynomials).

The structure of our proof is inspired by Reisert and Burkhardt (2008a). For our proof we consider the smooth and local stochastic process of random signals $g: \mathbb{R} \rightarrow \mathbb{R}$ that is designed in the following manner: suppose there exist a white process of signals $f: \mathbb{R} \rightarrow \mathbb{R}$ whose values are independently Gaussian distributed i.e. $f(x) \sim \mathcal{N}(0,1)$ (For brevity we assume zero mean and a variance of 1 because the mean has now influence on the corresponding covariance function and altering the variance just scales the covariance function). The covariance function of the white process is then given by $c_f(x,y) = \delta(x-y)$. We then construct the smooth and local stochastic process of random signals g by convolving the signals f with a Gaussian of width σ_s followed by windowing the result with a Gaussian of width σ_w. Hence g has the form

$$g(y) := e^{-\frac{y^2}{2\sigma_w^2}} \int_x e^{-\frac{(y-x)^2}{2\sigma_s^2}} f(x)dx \quad . \tag{5.22}$$

A basis for optimally representing a stochastic process is formed by the eigenfunction of the covariance function of the process. The transformation into the new basis system is known as the Karhunen-Loéve transformation. Hence we must show that the Hermite polynomials are eigenfunctions of a smooth and local stochastic process:

The covariance function c_g of the smooth and local stochastic process of random signals g is given by

$$c_g(x,y) = e^{-\frac{x^2}{2\sigma_w^2}} e^{-\frac{(x-y)^2}{4\sigma_s^2}} e^{-\frac{y^2}{2\sigma_w^2}} \quad ; \tag{5.23}$$

for details see Eq. (B.27) on page 203. Let further $t = \sigma_s^2$ be the width of the window function and let $\sigma_w^2 = (t^2-1)/t$ be the width of the smoothing. Then we can use the following identity (see auxiliary calculation Eq. (B.28) on page 204)

$$\frac{(1+t)}{\sqrt{\pi 4t}} c_g(x,y) = \sum_{n=0}^{\infty} \frac{H_n(x)H_n(y)}{\sqrt{\pi} n! 2^n} (\tfrac{t-1}{t+1})^n e^{-\frac{(x^2+y^2)}{2}} \tag{5.24}$$

to show that the Gaussian windowed Hermite polynomials $H_m(y)e^{-\frac{x^2}{2}}$ are the eigenfunctions with eigenvalues $(\frac{t-1}{t+1})^m$ of the covariance function $c_g(x,y)$, namely

$$\begin{aligned}
&\frac{(1+t)}{\sqrt{\pi 4t}} \int_x c_g(x,y) H_m(x) e^{-\frac{x^2}{2}} dx \\
&= \int_x \sum_{n=0}^{\infty} \frac{H_n(x)H_n(y)}{\sqrt{\pi} n! 2^n} (\tfrac{t-1}{t+1})^n e^{-\frac{(x^2+y^2)}{2}} H_m(x) e^{-\frac{x^2}{2}} dx \\
&= \sum_{n=0}^{\infty} (\tfrac{t-1}{t+1})^n H_n(y) e^{-\frac{y^2}{2}} \int_x \tfrac{H_n(x)H_m(x)}{\sqrt{\pi} n! 2^n} e^{-x^2} dx \\
&= \sum_{n=0}^{\infty} (\tfrac{t-1}{t+1})^n H_n(y) e^{-\frac{y^2}{2}} \delta_{mn} = (\tfrac{t-1}{t+1})^m H_m(y) e^{-\frac{y^2}{2}} \quad .
\end{aligned} \tag{5.25}$$

Consequently they are optimal for representing local and smooth processes. It is worth mentioning that we have some restrictions on t. The windowing process and the smoothing process are connected and can not be chosen independently. Furthermore, t must be > 1 because only then we have a smoothing. Moreover, for $t \to \infty$ the Eigenvalue $(\frac{t-1}{t+1})^m$ becomes 1; for $t = 1$ the Eigenvalue becomes 0. Hence the Eigenvalues for local and smooth processes are in the range $(\frac{t-1}{t+1})^m \in (0,1)$. □

5.1.2.2. Spherical Gaussian Derivatives.

It has been shown by Reisert and Burkhardt (2009a) that the Gaussian windowed solid harmonics $\mathbf{R}^\ell(\mathbf{r})e^{\frac{-r^2}{2}}$ arise from the 3D Gaussian by recursively applying the spherical up-derivatives. When additionally considering the spherical down-derivatives we get the full set of the 3D spherical Laguerre functions:

Theorem 5.1.4. *The Gaussian windowed Laguerre functions* $\mathcal{L}_n^\ell(\mathbf{r})e^{-\frac{r^2}{2t}} \in \mathcal{T}_{\ell-n}$ *are the spherical derivatives of the 3D isotropic Gaussian, namely*

$$\mathcal{L}_n^\ell(\mathbf{r})e^{-\frac{r^2}{2t}} = \frac{(-t)^\ell}{n!2^n}\boldsymbol{\nabla}_n^\ell e^{-\frac{r^2}{2t}} \quad . \tag{5.26}$$

Proof. We prove theorem 5.1.4 using mathematical induction. We first consider the case where $n = 0$. In this case $\mathcal{L}_0^\ell(\mathbf{r})e^{-\frac{r^2}{2t}} = \mathbf{Y}^\ell(\mathbf{r})r^\ell e^{-\frac{r^2}{2t}}$ (see Eq. (5.10) and Eq. (C.20)). For $\ell \to 0$ Eq. (5.26) holds. Using lemma 4.1.1 on page 100 we can show the inductive step $\ell \to \ell + 1$, with

$$\begin{aligned}
\tfrac{1}{(-t)^\ell}\boldsymbol{\nabla}^1 \mathcal{L}_0^\ell(\mathbf{r})e^{-\frac{r^2}{2t}} &= \tfrac{1}{(-t)^\ell}\mathbf{Y}^{\ell+1}(\mathbf{r})r^\ell \tfrac{\partial}{\partial r}\tfrac{1}{r^\ell}r^\ell e^{-\frac{r^2}{2t}} \\
&= \tfrac{1}{(-t)^{\ell+1}}\mathbf{Y}^{\ell+1}(\mathbf{r})r^{\ell+1}e^{-\frac{r^2}{2t}} \\
&= \tfrac{1}{(-t)^{\ell+1}}\mathcal{L}_0^{\ell+1}(\mathbf{r})e^{-\frac{r^2}{2t}} \ .
\end{aligned} \tag{5.27}$$

In combination with the spherical down-derivatives we additionally have a differential formulation for increasing the polynomial degree n of the radial Laguerre polynomials iteratively. Let us consider the inductive step $n \to n+1$. With lemma 4.1.1 we obtain the following equation:

$$\begin{aligned}
&\tfrac{n!2^n}{(-t)^\ell}\boldsymbol{\nabla}_1 \mathcal{L}_n^\ell(\mathbf{r})e^{-\frac{r^2}{2t}} = \\
&= \boldsymbol{\nabla}_1 \mathbf{Y}^{\ell-n}(\mathbf{r})r^{\ell-n}\tfrac{2^n n!}{(-t)^\ell}L_n^{(\ell-n)+\frac{1}{2}}(\tfrac{r^2}{2t})e^{-\frac{r^2}{2t}} \\
&= \mathbf{Y}^{\ell-(n+1)}(\mathbf{r})\tfrac{\frac{2^n n!}{(-t)^\ell}}{r^{(\ell-n)+1}}\tfrac{\partial}{\partial r}r^{2(\ell-n)+1}L_n^{(\ell-n)+\frac{1}{2}}(\tfrac{r^2}{2t})e^{\frac{-r^2}{2t}} .
\end{aligned} \tag{5.28}$$

Considering only the radial component we get

$$\begin{aligned}
&\tfrac{1}{r^{(\ell-n)+1}}\tfrac{\partial}{\partial r}r^{2(\ell-n)+1}L_n^{(\ell-n)+\frac{1}{2}}(\tfrac{r^2}{2t})e^{\frac{-r^2}{2t}} \\
&= \tfrac{(2t)^{(\ell-n)+\frac{1}{2}}}{r^{(\ell-n)+1}}\tfrac{\partial}{\partial r}(\tfrac{r^2}{2t})^{(\ell-n)+\frac{1}{2}}L_n^{(\ell-n)+\frac{1}{2}}(\tfrac{r^2}{2t})e^{\frac{-r^2}{2t}} \ .
\end{aligned}$$

Differentiating using the differential relation of the Laguerre polynomials (Eq. (C.22)) and combining the resulting two Laguerre polynomials utilizing the 3-point rule (C.21) leads to

$$\begin{aligned}
&\tfrac{(2t)^{(\ell-n)+\frac{1}{2}}}{r^{(\ell-n)+1}}\tfrac{\partial}{\partial r}(\tfrac{r^2}{2t})^{(\ell-n)+\frac{1}{2}}L_n^{(\ell-n)+\frac{1}{2}}(\tfrac{r^2}{2t})e^{\frac{-r^2}{2t}} \\
&= 2(n+1)r^{(\ell-(n+1))}L_{n+1}^{(\ell-(n+1))+\frac{1}{2}}(\tfrac{r^2}{2t})e^{\frac{-r^2}{2t}} \ .
\end{aligned} \tag{5.29}$$

Plugging Eq. (5.29) into (5.28) shows that

$$\tfrac{n!2^n}{(-t)^\ell}\boldsymbol{\nabla}_1 \mathcal{L}_n^\ell(\mathbf{r})e^{-\frac{r^2}{2t}} = \tfrac{(n+1)!2^{n+1}}{(-t)^\ell}\mathcal{L}_{n+1}^\ell(\mathbf{r})e^{-\frac{r^2}{2t}} \ . \tag{5.30}$$

□

5.1.2.3. An $SE(3)$ Covariant Spherical Gauss-Laguerre Transform

Similar to the "classical" dense spherical harmonic transformation $\mathcal{SH}$, the Laguerre transformation induces an $SE(3)$ covariant filter. We denote this dense Laguerre transformation by $\mathcal{LT} : \mathcal{T}_0 \to \mathcal{T}_{(\ell-n)}$. It is induced by the Laguerre transform via

$$\mathcal{LT}^{\ell}\{I,n\}(\mathbf{x}) := \langle \mathfrak{h}_{(-\mathbf{x})} I, \alpha^{\frac{1}{2}}(\ell,n,t)\overline{\boldsymbol{\mathcal{L}}_n^{\ell}} \rangle_{\mu} \quad . \tag{5.31}$$

We denote the filter output, an image of expansion coefficients, by

$$\mathbf{a}_n^{\ell} := \mathcal{LT}^{\ell m}\{I,n\}, \tag{5.32}$$

where the elements $\mathbf{a}_n^{\ell}(\mathbf{x}) \in \mathbb{C}^{(2(\ell-n)+1)}$ are local expansion coefficients according to Eq. (5.16). Apart from the 1D radial functions, which have no influence on rotations, the filter is identical to $\mathcal{SH}$. Hence it transforms under Euclidean motion in the same way as $\mathcal{SH}$. We omit showing the $SE(3)$ covariance of $\mathcal{LT}$ and refer to the preliminary chapter instead.

The Gauss-Laguerre polynomials show a certain differential relationship. Similar to the example where we used the Hermite polynomials, we can use this property to realize the projection of images onto the basis functions via finite differences. More precisely, with theorem 5.1.4 we can express the fully-weighted Gauss-Laguerre transform (Eq. (5.16)) in terms of spherical tensor derivatives of a Gaussian smoothed image, leading to

$$\begin{aligned}
\mathcal{LT}^{\ell}\{I,n\}(\mathbf{x}) &= \langle \mathfrak{h}_{(-\mathbf{x})} I, \alpha^{\frac{1}{2}}(\ell,n,t)\overline{\boldsymbol{\mathcal{L}}_n^{\ell}} \rangle_{\mu} \\
&= \alpha^{\frac{1}{2}}(\ell,n,t) \underbrace{\int_{\mathbf{r}\in\mathbb{R}^3} I(\mathbf{r}+\mathbf{x})\boldsymbol{\mathcal{L}}_n^{\ell}(\mathbf{r}) e^{\frac{-r^2}{2t}}\, d\mathbf{r}}_{\substack{\propto 2(\ell-n)+1 \text{ 3D convolutions} \\ \text{(component by component)}}} \\
&= \alpha^{\frac{1}{2}}(\ell,n,t) \int I(\mathbf{r})\boldsymbol{\mathcal{L}}_n^{\ell}(\mathbf{r}-\mathbf{x}) e^{-\|\mathbf{r}-\mathbf{x}\|^2/(2t)} d\mathbf{r} \\
&= (-1)^{\ell-n}\alpha^{\frac{1}{2}}(\ell,n,t) \int I(\mathbf{r})\boldsymbol{\mathcal{L}}_n^{\ell}(\mathbf{x}-\mathbf{r}) e^{-\|\mathbf{r}-\mathbf{x}\|^2/(2t)} d\mathbf{r} \qquad \text{(sh-parity, Eq. (2.68))} \\
&= (-1)^{\ell-n}\alpha^{\frac{1}{2}}(\ell,n,t) \frac{(-t)^{\ell}}{n!2^n} \int I(\mathbf{r}) \boldsymbol{\nabla}_n^{\ell} e^{-\|\mathbf{r}-\mathbf{x}\|^2/(2t)} d\mathbf{r} \\
&= (-1)^{\ell-n}\alpha^{\frac{1}{2}}(\ell,n,t) \frac{(-t)^{\ell}}{n!2^n} \boldsymbol{\nabla}_n^{\ell} \int I(\mathbf{r}) e^{-\|\mathbf{r}-\mathbf{x}\|^2/(2t)} d\mathbf{r} \\
&= (-1)^{\ell-n}\alpha^{\frac{1}{2}}(\ell,n,t) \frac{(-t)^{\ell}}{n!2^n} \boldsymbol{\nabla}_n^{\ell} \underbrace{\int I(\mathbf{r}) e^{-\|\mathbf{x}-\mathbf{r}\|^2/(2t)} d\mathbf{r}}_{\text{one 3D convolution}} \quad .
\end{aligned} \tag{5.33}$$

Hence from a computational point of view we only need one initial convolution with a 3D Gaussian function followed by successively applying the spherical tensor derivative operators to densely obtain all coefficients with respect to all points of

$$\begin{tikzcd}
(I * G_t) = \mathbf{a}_0^0 \arrow[r, "\boldsymbol{\nabla}^1", "\mathcal{T}_0 \to \mathcal{T}_1"'] \arrow[d, "\Delta"'] & \mathbf{a}_0^1 \arrow[r, "\boldsymbol{\nabla}^1", "\mathcal{T}_1 \to \mathcal{T}_2"'] & \mathbf{a}_0^2 \arrow[r, "\boldsymbol{\nabla}^1", "\mathcal{T}_2 \to \mathcal{T}_3"'] & \mathbf{a}_0^3 \arrow[r, "\boldsymbol{\nabla}^1", "\mathcal{T}_2 \to \mathcal{T}_3"'] & \mathbf{a}_0^4 \\
\mathbf{a}_1^1 \arrow[r, "\boldsymbol{\nabla}^1", "\mathcal{T}_0 \to \mathcal{T}_1"'] \arrow[d, "\Delta"'] & \mathbf{a}_1^2 \arrow[r, "\boldsymbol{\nabla}^1", "\mathcal{T}_1 \to \mathcal{T}_2"'] & \mathbf{a}_1^3 & & \\
\mathbf{a}_2^2 & & & &
\end{tikzcd}$$

Figure 5.5.: Transformation to the Gauss-Laguerre domain via the Laplace operator and the spherical up-derivatives. The expansion coefficients are computed up to a polynomial degree of 4 (The polynomial degree corresponding to an expansion coefficient $\mathbf{a}_n^\ell(\mathbf{x}) \in \mathbb{C}^{2(\ell-n)+1}$ is $\ell + n$). The Laplace operator increases the polynomial degree in the radial direction. The spherical tensor up-derivative operator increases the polynomial degree in angular direction.

expansion $\mathbf{x}$:

$$\begin{aligned}
\mathcal{LT}^\ell\{I, n\}(\mathbf{x}) &= \langle \mathfrak{t}_{(-\mathbf{x})} I, \alpha^{\frac{1}{2}}(\ell, n, t) \overline{\boldsymbol{\mathcal{L}}_n^\ell} \rangle_\mu \\
&= (-1)^{\ell-n} \alpha^{\frac{1}{2}}(\ell, n, t) \tfrac{(-t)^\ell}{n! 2^n} \boldsymbol{\nabla}_n^\ell (I * G_t)(\mathbf{x}) \\
&= (-1)^{\ell-n} \alpha^{\frac{1}{2}}(\ell, n, t) \tfrac{(-t)^\ell}{n! 2^n} \boldsymbol{\nabla}^{\ell-n} \Delta^n (I * G_t)(\mathbf{x}) \quad ,
\end{aligned} \tag{5.34}$$

where $G_t(\mathbf{r}) := e^{-\frac{r^2}{2t}}$ is the Gaussian function. Moreover, as shown in Eq. (5.34), we can completely omit the spherical down-derivatives by substituting $\boldsymbol{\nabla}_n^\ell = \boldsymbol{\nabla}^{\ell-n} \Delta^n$ (Eq (4.12) on page 99). As a consequence we successively apply the Laplace operator Δ to increase the polynomial degree n of the basis function in radial direction followed by successively applying the spherical derivatives $\boldsymbol{\nabla}^1$ to project onto higher order spherical harmonics of order $\ell - n$. A computation of a polynomial degree limited expansion coefficients is illustrated in Fig. 5.5.

5.1.3. An $SE(3)$ Covariant Gauss-Laguerre Transform for Tensor Fields

In many scenarios images are scalar valued. However, some image acquisition techniques or image preprocessing techniques lead to tensor valued modalities. (Note, that there always exists a spherical tensor counterpart for a Cartesian tensor valued image; see section 2.2.3.7 on page 78. For brevity we always assume a spherical tensor representation).

For instance, the gradient vector flow (Xu and Prince 1997) of an image is a tensor field of order 1. The proposed filter can be used for further analysis of such vector fields. For instance, it is possible to search for certain patterns in the field such as sources, sinks or torsions.

Another example are images acquired with the High Angular Resolution Diffusion Imaging (HARDI) technique (Tuch et al. 1999). The HARDI-technique combines different measurement parameters to infer underlying tissue properties and allows e.g. for studying the neuronal fiber architecture in the human brain without harming the patient (Reisert et al. 2010). The resulting image (the HARDI signal) is an angular dependent measurement so that spherical harmonics are a nearby tool for representing the signal (Descoteaux et al. 2007). This results in several spherical tensor fields, each representing a certain angular frequency within the signal. The proposed framework allows for further analysis of such fields. For instance, the proposed framework has been used for classifying HARDI images of human brains into gray matter and white matter tissue (Skibbe et al. 2011a), or to detect different anatomical regions (Skibbe and Reisert 2011). Both applications are introduced in chapter A on page 181.

By extending the Gauss-Laguerre transform via tensorial harmonics we obtain a Gauss-Laguerre basis for tensor fields of arbitrary tensor rank J. This basis induces a Gauss-Laguerre representation for higher order tensor fields. More precisely, by replacing the spherical harmonics with the tensorial harmonics (see section 2.2.3.4 on page 72) we gain an orthogonal basis spanning the space $\mathcal{T}_J$ of spherical tensor fields of order J. These basis functions have the form

$$\mathbf{Z}^{\ell jm}_{Jn}(\mathbf{x}) := \mathbf{e}^{(\ell-n)+j}_{m} \circ_J \overline{\boldsymbol{\mathcal{L}}^{\ell}_{n}(\mathbf{x})} \quad , \tag{5.35}$$

where $\mathbf{Z}^{\ell jm}_{Jn} : \mathbb{R}^3 \to \mathbb{C}^{2J+1}$. Note that in addition to the tensorial harmonics introduced in the preliminary section 2.2.3.4 on page 72, we have the additional parameter $n \in \mathbb{R}_{\geq 0}$ defining the degree of the Laguerre polynomials representing the radial profile of the tensor basis. The basis functions are orthogonal according to

$$\langle \mathbf{Z}^{\ell jm}_{Jn}, \mathbf{Z}^{\ell' j' m'}_{Jn'} \rangle_\mu = \frac{1}{A(\ell,j,n,t)} \delta_{\ell,\ell'} \delta_{j,j'} \delta_{m,m'} \delta_{n,n'} \, , \tag{5.36}$$

where $A(\ell,j,n,t) = \frac{(2((\ell-n)+j)+1)\alpha(\ell,n,t)}{2J+1} \in \mathbb{R}$ is a normalization factor (see auxiliary calculation on page 204). Hence any tensor field of order J can be expanded in terms of the Gauss-Laguerre basis by

$$\begin{aligned} \mathbf{f}^J(\mathbf{r}) &= \sum_{n \leq \ell} \sum_{j=-J}^{j=J} \sum_{m=-(\ell-n+j)}^{m=(\ell-n+j)} A^{\frac{1}{2}}(\ell,j,n,t) v^{\ell n}_{jm} \overline{\mathbf{Z}^{\ell jm}_{Jn}(\mathbf{r})} \\ &= \sum_{n \leq \ell} \sum_{j=-J}^{j=J} A^{\frac{1}{2}}(\ell,j,n,t) \mathbf{v}^{\ell n}_{j} \circ_J \boldsymbol{\mathcal{L}}^{\ell}_{n}(\mathbf{r}) \quad . \end{aligned} \tag{5.37}$$

We obtain the coefficients $\mathbf{v}^{\ell n}_j \in \mathbb{C}^{2((\ell-n)+j)+1}$ via orthogonal projection

$$v^{\ell n}_{jm} = \langle \mathbf{f}^J, A^{\frac{1}{2}}(\ell,j,n,t) \overline{\mathbf{Z}^{\ell jm}_{Jn}} \rangle_\mu \, . \tag{5.38}$$

It is straight forward to extend the dense Gauss-Laguerre transform $\mathcal{LT}^{\ell}(\mathbf{n}) : \mathcal{T}_0 \to \mathcal{T}_{(\ell-n)}$ to tensor fields of higher order. We obtain a new filter $\mathcal{LT}^{\ell j}(\mathbf{n}) : \mathcal{T}_J \to \mathcal{T}_{(\ell+j-n)}$ that maps spherical tensor fields of any order J to tensor fields representing the input field's local neighborhoods in a voxe-by-voxel manner by setting

$$[\mathcal{LT}^{\ell j}\{\mathbf{f}^J, n\}]_m(\mathbf{x}) := \langle \mathfrak{h}_{(-\mathbf{x})} \mathbf{f}^J, A^{\frac{1}{2}}(\ell,j,n,t) \overline{\mathbf{Z}^{\ell jm}_{Jn}} \rangle_\mu \, . \tag{5.39}$$

The filter output are images of spherical tensor valued tensorial harmonic expansion coefficients

$$\mathbf{a}_j^{\ell n} := \mathcal{LT}^{\ell j}\{\mathbf{f}^J, n\}, \tag{5.40}$$

where $\mathbf{a}_j^{\ell n} \in \mathcal{T}_{2(\ell+j-n)+1}$.

For implementation we suggest the same strategy as for the "classical" tensorial harmonic transform via weighted superposition of several scalar valued spherical harmonic transformations of the single tensor field components; see Eq. (B.18) on page 201. The explicit formular can be found on page 204.

5.1.3.1. A Fast Gauss-Laguerre Transform via Tensor Derivatives

A certain subset of the Gauss-Laguerre tensorial harmonic expansion coefficient images $\mathbf{a}_j^{\ell n}$ (Eq. (5.39)) can be computed in an even more efficient manner via tensor derivatives. These coefficients have the form $\mathbf{a}_J^{\ell n}$ and $\mathbf{a}_{J-2(\ell-n)}^{\ell n}$. This is because they can be computed by directly applying the spherical tensor derivatives to a Gaussian smoothed tensor field $\mathbf{f}^J$. We thus avoid an initial decomposition of the field into its $2J+1$ components. Furthermore, we avoid a final weighted superposition of the partial expansions. In Fig. 5.6 on the following page we have exemplarily highlighted these coefficients for tensor fields of order $J=0$, $J=1$ and $J=2$. Remember, that for $J=0$ the expansion coincides with the standard Gauss-Laguerre transform. In this trivial case, all coefficients can be computed in the proposed manner.

Such a fast transformation is very beneficial for our applications. In general, we are rather interested in efficiently computing compact, discriminative representations of local image structures than in fully restoring our image from the tensorial harmonic coefficients. Of course, by choosing only a subset of coefficients we loose information because not all information is covered by the partial expansion. However, in practice, the large number of possible tensorial harmonic coefficients demands for selecting a small finite subset out of them. Since all coefficients are representing mutual exclusive information, choosing the coefficients that can be computed efficiently satisfies both demands, reducing memory usage and reducing computational complexity.

The computation rule for the coefficient image $\mathbf{a}_J^{\ell n} := \mathcal{LT}^{\ell J}\{\mathbf{f}^J, n\}$ is

$$\begin{aligned}
&\mathbf{a}_J^{\ell n} = \\
&= A^{\frac{1}{2}}(\ell,J,n,t)\tfrac{(-t)^\ell}{n!2^n}\boldsymbol{\nabla}^{\ell-n}\Delta^n(\mathbf{f}^J\widetilde{\bullet}_J G_t)\frac{(-1)^{(\ell-n)}\sqrt{(2(\ell-n)+1)}\langle(\ell-n)0,J0\,|\,(J+(\ell-n))0\rangle}{\sqrt{N_{(\ell-n),J}}} \\
&= \boldsymbol{\nabla}^{\ell-n}\Delta^n(\mathbf{f}^J\widetilde{\bullet}_J G_t)A^{\frac{1}{2}}(\ell,J,n,t)\frac{(t)^\ell\sqrt{(2(\ell-n)+1)}\langle(\ell-n)0,J0\,|\,(J+(\ell-n))0\rangle}{n!2^n(-1)^n\sqrt{N_{(\ell-n),J}}} \\
&= \boldsymbol{\nabla}^{\ell-n}\Delta^n(\mathbf{f}^J\widetilde{\bullet}_J G_t)\frac{(t)^\ell\alpha^{\frac{1}{2}}(\ell,n,t)\langle(\ell-n)0,J0\,|\,(J+(\ell-n))0\rangle}{n!2^n(-1)^n}\ ,
\end{aligned} \tag{5.41}$$

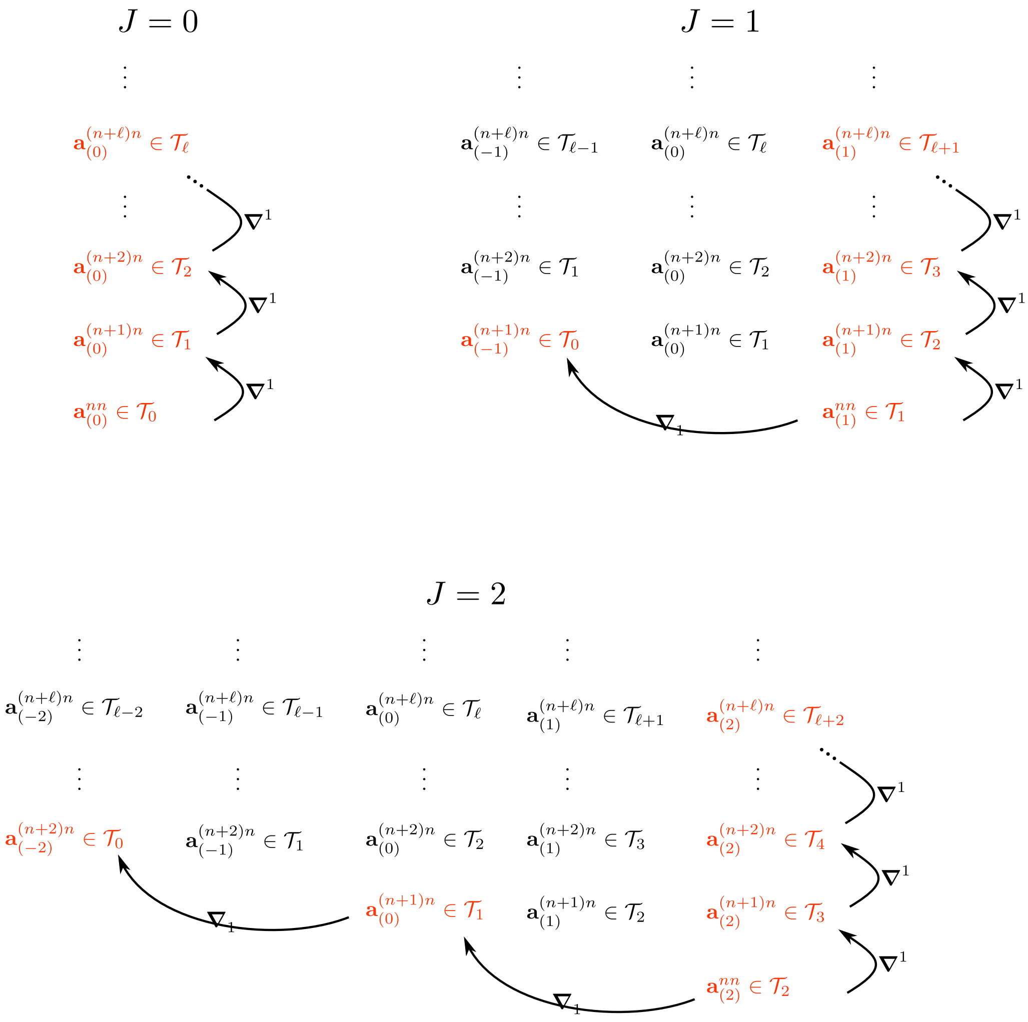


Figure 5.6.: There exists a systematic way for directly computing the coefficients shown in red in terms of tensor derivatives. After initially smoothing the tensor field with an isotropic Gaussian in a component-by-component manner, only up- and down-derivatives are required. From bottom to the top: coefficients associated with spherical harmonics of increasing order $\ell = 0, 1, \cdots$. (Note that we are not considering the normalization factors here)

and for the coefficients $\mathbf{a}^{\ell n}_{J-2(\ell-n)} := \mathcal{LT}^{\ell(J-2(\ell-n))}\{\mathbf{f}^J, n\}$

$$\begin{aligned}
\mathbf{a}^{\ell n}_{(J-2(\ell-n))} &= A^{\frac{1}{2}}(\ell,(J-2(\ell-n)),n,t)\tfrac{(-t)^\ell}{n!2^n} \times \\
&\quad \boldsymbol{\nabla}_{\ell-n}\Delta^n(\mathbf{f}^J \widetilde{\bullet}_J G_t)\frac{(-1)^{(\ell-n)}\sqrt{(2(\ell-n)+1)}\langle(\ell-n)0,J0\,|\,(J-(\ell-n))0\rangle}{\sqrt{N_{(\ell-n),J}}} \\
&= \boldsymbol{\nabla}_{\ell-n}\Delta^n(\mathbf{f}^J \widetilde{\bullet}_J G_t)A^{\frac{1}{2}}(\ell,(J-2(\ell-n)),n,t)\frac{(t)^\ell\sqrt{(2(\ell-n)+1)}\langle(\ell-n)0,J0\,|\,(J-(\ell-n))0\rangle}{n!2^n(-1)^n\sqrt{N_{(\ell-n),J}}} \\
&= \boldsymbol{\nabla}_{\ell-n}\Delta^n(\mathbf{f}^J \widetilde{\bullet}_J G_t)\frac{(t)^\ell\alpha^{\frac{1}{2}}(\ell,n,t)\sqrt{(2(J-(\ell-n))+1)}\langle(\ell-n)0,J0\,|\,(J-(\ell-n))0\rangle}{n!2^n(-1)^n\sqrt{(2(J+(\ell-n))+1)}}\ . \qquad (5.42)
\end{aligned}$$

As a consequence, only $2J+1$ initial convolutions with a Gaussian function are required (component-by-component). The remaining operators are discrete tensor derivatives and the application of the Laplace operator is realized via finite differences. For proofs see section B.3.1.2 on page 205.

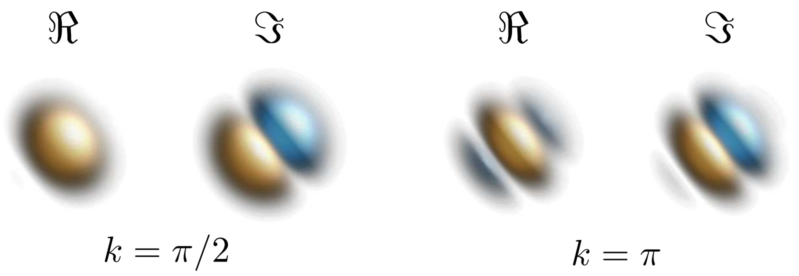


Figure 5.7.: Gabor wavelets in 3D (red: positive, blue: negative values)

5.2. An $SE(3)$ Covariant Spherical Gabor Transform

A Gabor[1] function (Gabor 1846) is a directed, Gaussian windowed plane wave suited for locally analyzing the frequency of signals. The Gaussian windowed wave is also known as Gabor wavelet, or as Gabor function. In 2D or 3D, a Gabor wave simultaneously selects a certain orientation and frequency with respect to a certain position or time point of a signal. Figure 5.8 on the facing page exemplarily shows a 2D Gabor wavelet, in Fig. 5.7 we show the real and imaginary part of Gabor wavelets in 3D.

The Gabor functions are well suited to model simple cells in the visual cortex of mammalian brains (Daugman 1985) so that Gabor functions are an appropriate mathematical tool for mimicking human visual perception. Therefore, Gabor functions are playing an important role in many early vision tasks (Daugman 1980; Daugman 1988) and became particularly important in applications like texture segmentation and classification (Bovik et al. 1990; Jain and Farrokhnia 1990; Sandler and Lindenbaum 2009). Similar to a set of Gaussian derivatives forming a "local jet" (see the previous section 5.1 on page 110), a set of Gabor features representing different angular and radial frequencies at a certain image location are also known as "Gabor-jet" (Buhmann et al. 1989). Gabor features show high robustness against noise (Kamarainen et al. 2002). Applications are ranging from texture based segmentation of single tooths from x-ray dental images (Choorat et al. 2012), face detection (Kyrki et al. 2004), character recognition (Hu et al. 2002; Yoshimura et al. 2000), Gabor feature based hash functions for purposes like watermarking (Li et al. 2012) and from the extraction of tagging sheets in tagged cardiac MR images (Qian et al. 2006) to the segmentation of MRI and CT images of human brains (Olowoyeye et al. 2009). An overview of recent applications can be found e.g. in Kamarainen et al. (2006).

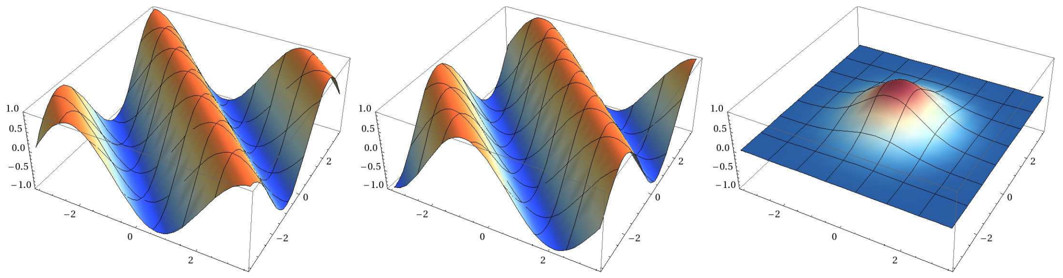
The 3D plane-wave $\omega : \mathbb{R}^3 \rightarrow \mathbb{C}$ with corresponding wave-vector $\mathbf{k} \in \mathbb{R}^3$ is defined by $\omega(\mathbf{r}, \mathbf{k}) := e^{i\mathbf{k}^T\mathbf{r}}$. Its counterpart in the frequency domain is the delta impulse having its peak at position $\mathbf{k}$, namely

$$\omega(\mathbf{r}, \mathbf{k}') = e^{i\mathbf{k}'^T\mathbf{r}} \circ\!\!-\!\!\bullet\, \widetilde{\omega}(\mathbf{k}, \mathbf{k}') = (2\pi)^{\frac{3}{2}} \delta(\mathbf{k}' - \mathbf{k}) \quad ; \tag{5.43}$$

see Eq. (B.40) in the appendix. Since point-wise multiplications in spatial domain are convolutions in the frequency domain we conclude that having a Gaussian windowed plane-wave i.e. a Gabor wavelet with a certain direction in spatial domain

---

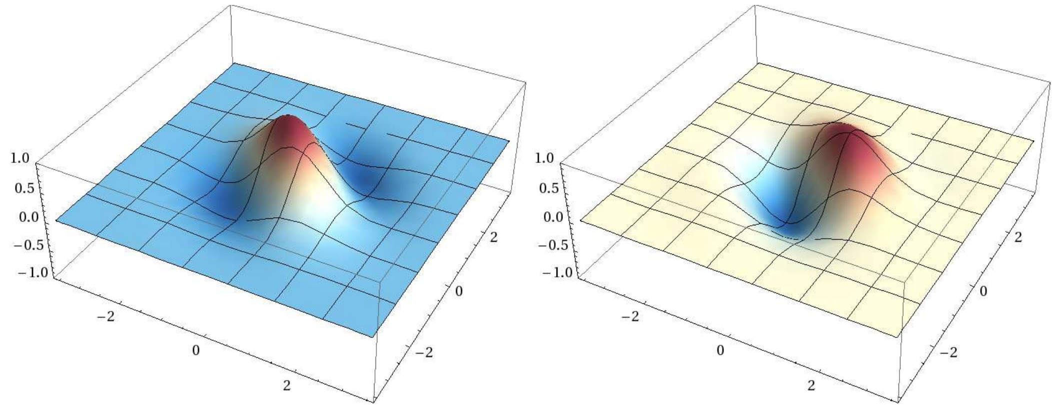
[1] named after Dennis Gabor, 1900-1979, a Hungarian electrical engineer and physicist



(a) Real part of a directed 2D plain-wave $\Re(e^{i\mathbf{k}'^T\mathbf{r}})$, where $\mathbf{k} = \frac{\pi}{2}(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^T$.

(b) Imaginary part of a directed 2D plain-wave $\Im(e^{i\mathbf{k}'^T\mathbf{r}})$, where $\mathbf{k} = \frac{\pi}{2}(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^T$.

(c) 2D Gaussian $e^{-\mathbf{r}^T\mathbf{r}/(2)}$.



(d) Real (symmetric) part of a directed 2D Gaussian windowed plain-wave $\Re(e^{-\mathbf{r}^T\mathbf{r}/2}e^{i\mathbf{k}'^T\mathbf{r}})$ (The cosine).

(e) Imaginary (antisymmetric) part of a directed 2D Gaussian windowed plain-wave $\Im(e^{-\mathbf{r}^T\mathbf{r}/2}e^{i\mathbf{k}'^T\mathbf{r}})$ (The sinus)..

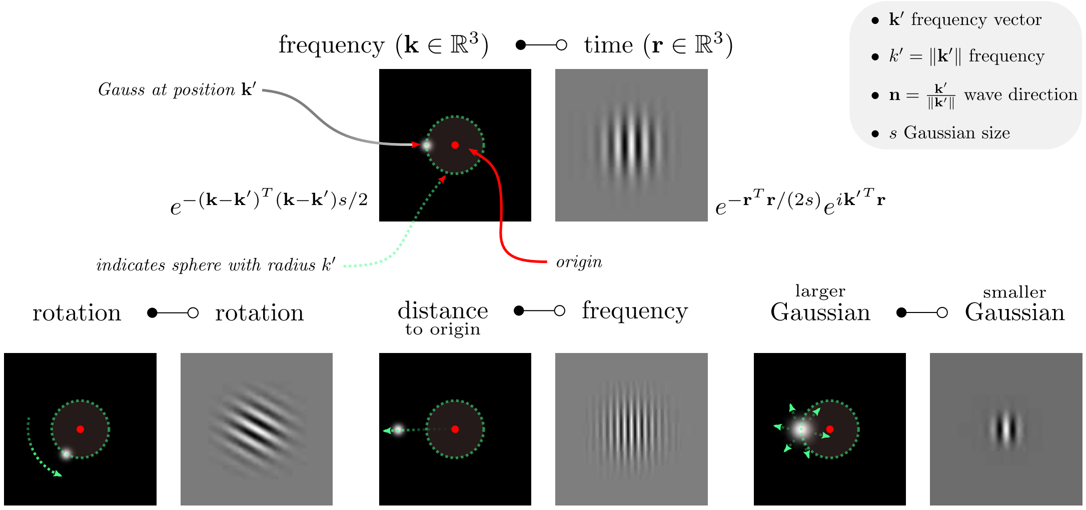
Figure 5.8.: A Gabor wave in 2D.

corresponds to a convolution of the corresponding delta peak in the frequency domain with a Gaussian function i.e. a shift of the Gaussian function. The Gabor kernel $\omega_s : \mathbb{R}^3 \rightarrow \mathbb{C}$ in spatial and frequency domain is given by

$$\omega_s(\mathbf{r},\mathbf{k}') = e^{-\mathbf{r}^T\mathbf{r}/(2s)}e^{i\mathbf{k}'^T\mathbf{r}} \circ\!\!-\!\!\!-\!\!\bullet\ \widetilde{\omega_s}(\mathbf{k},\mathbf{k}') = (2\pi s)^{\frac{3}{2}} e^{-(\mathbf{k}-\mathbf{k}')^T(\mathbf{k}-\mathbf{k}')s/2} \quad , \tag{5.44}$$

where $s \in \mathbb{R}_{>0}$ defines the width of the Gaussian window. Note that a decreasing width of the Gaussian window in spacial domain comes with an increasing size of the Gaussian function in the frequency domain (and vise versa). This behavior reflects the uncertainty of the true frequency when analyzing only a small region of a given signal. In addition to the Gaussian window size, the position in the frequency domain steers the orientation and frequency of the Gabor wavelet. In Fig. 5.9 we show the connection of frequency and time domain of Gabor functions.

A family of Gabor functions decomposes the appearance of a local structure into local frequency patterns. A family consists of Gabor waves of different orientations but having the same frequency magnitude. Hence in the frequency domain, all Gabor wavelets of a family are Gaussians whose centers are points on the same sphere. The common way to implement Gabor filters is to explicitly sample the orientation



$k'_1 = k'_2$ but $\mathbf{n}_1 \neq \mathbf{n}_2$ $\mathbf{n}_1 = \mathbf{n}_2$ but $k'_1 \neq k'_2$ $s_1 \neq s_2$

Figure 5.9.: Gabor function in frequency and spatial (time) domain

space and using a finite number of Gabor kernels, each representing a certain orientation of the plane wave; see e.g. Bigun (1994). This requires a quantization of the orientation space as illustrated in Fig. 5.10 on the next page a). Such a quantization has two shortcomings: (1) To ensure a sufficient sampling accuracy of the orientation space, a large number of Gabor functions is necessary, particularly in 3D. This negatively affects the computation time. (2) In 2D, sampling of the orientation space corresponds to an equidistant sampling of the unit-circle which can be solved easily. However, in 3D, equidistant sampling of the orientation space is equivalent to determine an equidistant distribution of points on the unit-sphere, which is, in general, a non-trivial problem known as the *Thomson problem* (Thomson 1904).

We propose a new family of functions that overcomes theses shortcomings. Our idea is inspired by Simoncelli and Freeman (1995), where frequency and orientation selection are two separate steps. We select a certain frequency range in the frequency domain by smoothing a sphere. The radius of the sphere defines the frequency k; the radial smoothing defines the frequency range (an interval around k defined by the Gaussian window); see Fig. 5.10 b). This can be regarded as an undirected Gabor wave. The counterpart in spatial domain is a Gaussian windowed Bessel function. We then use the spherical tensor derivatives to sample the orientation space in terms of spherical harmonics. Thanks to the spherical harmonics we overcome the problem of quantization in the orientation domain. The orientations are now covered continuously. Furthermore, the tensor derivatives offer an efficient way to densely compute Gabor filter responses for the whole image. With a spherical decomposition of the Gabor wavelet we gain features representing all local radial and angular frequency components separately. For details regarding rotation invariant descriptors based on spherical harmonic expansion coefficients we refer to chapter 3 on page 83.

This section is based on work that has partially been published in Skibbe et al. (2012) and also includes works that have been presented at conferences (Skibbe et al. 2010; Skibbe et al. April, 2011).

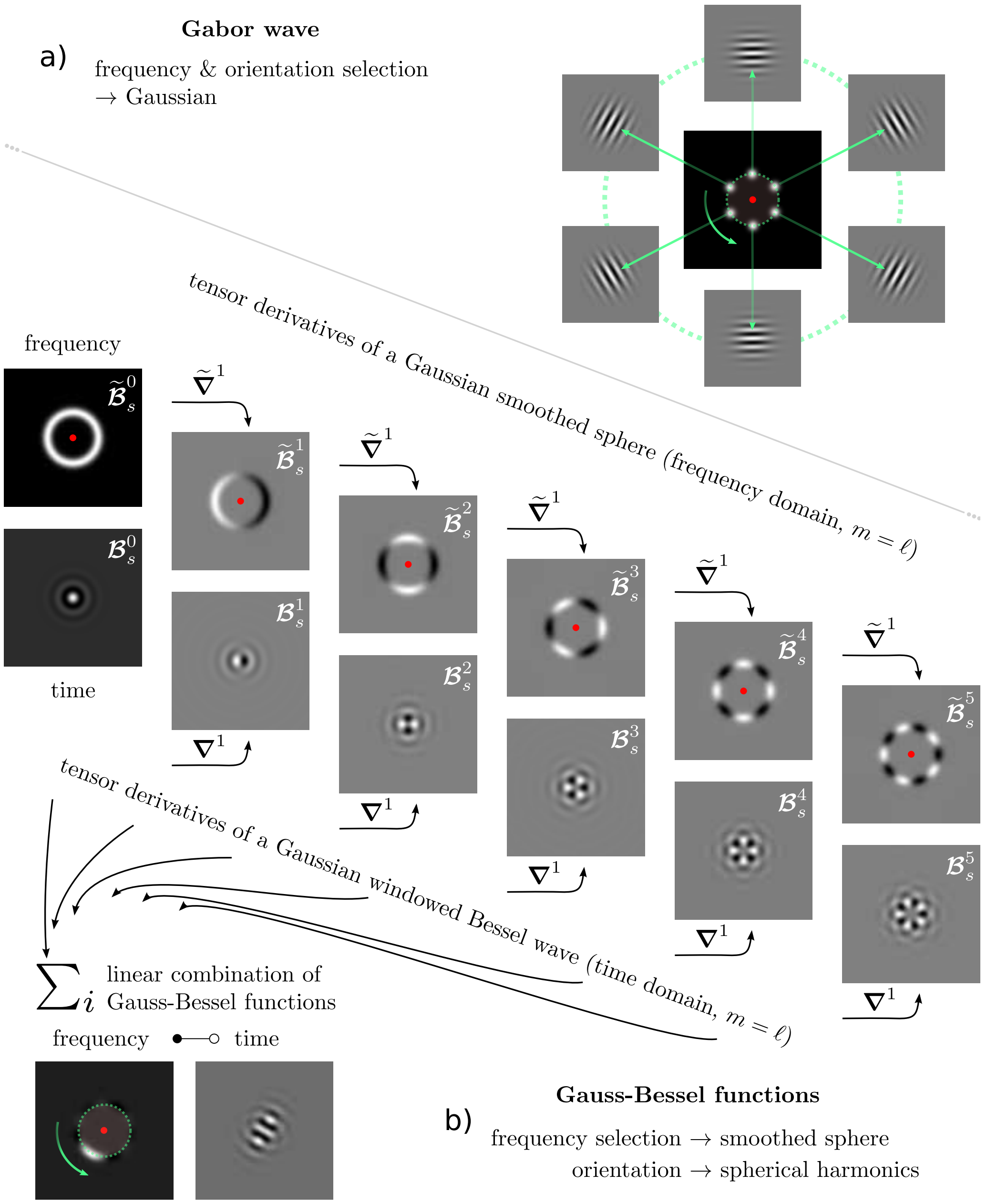


Figure 5.10.: a) For the ordinary Gabor wave, the selection of frequency and orientation is connected. This is because a Gabor wave in spatial (time) domain is an isotropic Gauss in frequency domain. (We only show the center slice of the real part of a 3D image here). b) We propose to separate the frequency selection from orientation selection. We utilize a Gaussian smoothed sphere in frequency domain to select a certain frequency. We then use tensor derivatives in frequency domain to sample the orientation space in angular direction. This corresponds to the Gauss-Bessel functions in spatial domain. A linear combination of these functions can be used to approximate a Gabor wave in any direction. (We only show the center slice of the real part for $m = 0$ of a spherical tensor field here).

5.2.1. The Fourier Transform in Terms of Spherical Derivatives

The proposed Gabor transformation makes use of the fact, that the ordinary Fourier transform can be represented in terms of spherical tensor derivatives.

Let $\omega(\mathbf{r},\mathbf{k}) := e^{\frac{i\mathbf{k}^T\mathbf{r}}{\sqrt{t}}}$ be the 3D plane wave. With $t \in \mathbb{R}$ we denote a scale parameter which we consider w.l.o.g to be constant. The plane wave ω can be represented in spherical coordinates by decomposing the wave into its angular and its radial frequency components (see Eq. (C.18) in the appendix):

$$\begin{aligned}\omega(\mathbf{r},\mathbf{k}) &= \sum_\ell (i)^\ell (2\ell+1) \underbrace{j_\ell(k\tfrac{r}{\sqrt{t}})\mathbf{Y}^\ell(\mathbf{r})}_{=\mathcal{B}^\ell(\mathbf{r},k)} \bullet_0 \mathbf{Y}^\ell(\mathbf{k}) \\ &= \sum_\ell (i)^\ell (2\ell+1)\mathbf{Y}^\ell(\mathbf{k}) \bullet_0 \mathcal{B}^\ell(\mathbf{r},k) \quad . \end{aligned} \tag{5.45}$$

With $\mathcal{B}^\ell(\mathbf{r},k) := \mathbf{Y}^\ell(\mathbf{r}) j_\ell(k\frac{r}{\sqrt{t}})$ we denote the Bessel functions representing the angular frequency component ℓ and radial frequency component k of the plane wave. They form a set of orthogonal basis functions spanning $L_2(\mathbb{R}^3)$. Note that $\mathcal{B}^\ell(\mathbf{r},k)$ is independent the direction $\mathbf{n} = \frac{\mathbf{k}}{\|\mathbf{k}\|}$ of $\mathbf{k}$. The steering of the direction of the wave is only done by the products with $\mathbf{Y}^\ell(\mathbf{k})$. We call $\mathbf{Y}^\ell(\mathbf{k})$ the angular expansion coefficients of the plane wave. The functions $j_\ell : \mathbb{R} \to \mathbb{R}$ are known as spherical Bessel functions; see Eq. (C.12) in the appendix. While the spherical harmonics build an orthogonal basis for functions on the 2-sphere, the spherical Bessel functions form an orthogonal basis for 1D functions with respect to the spherical weight r^2; see Eq. (C.13) in the appendix.

The functions $\mathcal{B}^\ell(\mathbf{r},k)$ can be used to perform a partial Fourier transform of a signal, leaving the orientation of the wave undetermined: let $I \in L_2(\mathbb{R}^3)$ be an image, then the Fourier transform of f is given by

$$\begin{aligned}\widetilde{I}(\mathbf{k}) = (2\pi)^{-3/2}\langle I, \omega(\mathbf{k})\rangle &= (2\pi)^{-3/2}\int_{\mathbb{R}^3} I(\mathbf{r}) \sum_\ell (-i)^\ell (2\ell+1)\mathbf{Y}^\ell(\mathbf{k}) \bullet_0 \mathcal{B}^\ell(\mathbf{r},k) d\mathbf{r} \\ &= (2\pi)^{-3/2} \sum_\ell (-i)^\ell (2\ell+1)\mathbf{Y}^\ell(\mathbf{k}) \bullet_0 \underbrace{\int_{\mathbb{R}^3} I(\mathbf{r})\mathcal{B}^\ell(\mathbf{r},k) d\mathbf{r}}_{=\mathbf{v}^\ell(k)} \\ &= \sum_\ell (-i)^\ell (2\ell+1)\mathbf{Y}^\ell(\mathbf{k}) \bullet_0 (2\pi)^{-3/2} \underbrace{\langle I, \overline{\mathcal{B}^\ell(k)}\rangle}_{=\mathbf{v}^\ell(k)} \quad . \end{aligned} \tag{5.46}$$

We call the coefficients $\mathbf{v}^\ell(k)$ the angular independent Fourier coefficients of I, completely representing the function I in Fourier domain (up to rotation). We obtain the coefficients $\mathbf{v}^\ell$ via orthogonal projection on $\mathcal{B}^\ell(k)$, that is

$$\mathbf{v}^\ell(k) := \langle I, \overline{\mathcal{B}^\ell(k)}\rangle \, . \tag{5.47}$$

We call the Bessel functions $\mathcal{B}^\ell(\mathbf{r},k)$ the angular Fourier basis functions. The Fourier correspondences of $\mathcal{B}^\ell$ are just spherical harmonics living on a sphere; see

Eq. (B.41) in the appendix on page 209 for the mathematical derivation. The radius of the sphere corresponds to the ratio of frequency and scale $k/\sqrt{t}$. They are defined by

$$\boldsymbol{\mathcal{B}}^{\ell}(\mathbf{r},k') = \mathbf{Y}^{\ell}(\mathbf{r}) j_{\ell}(k'\tfrac{r}{\sqrt{t}}) \circ\!\!-\!\!\bullet\, \widetilde{\boldsymbol{\mathcal{B}}}^{\ell}(\mathbf{k},k') = (-i)^{\ell}\tfrac{t\sqrt{\pi}}{(k'^{2})}\delta(k-\tfrac{k'}{\sqrt{t}})\mathbf{Y}^{\ell}_{m}(\mathbf{k})\,. \tag{5.48}$$

There exists a natural relation between the angular Fourier basis and the spherical tensor derivatives, namely

$$\boldsymbol{\nabla}^{\ell}\boldsymbol{\mathcal{B}}^{0}(k) = (\tfrac{-k}{\sqrt{t}})^{\ell}\boldsymbol{\mathcal{B}}^{\ell}(k)\,. \tag{5.49}$$

We have proven this equation in Skibbe et al. (2010). Alternatively, it can be shown by using Eq. (4.15) on page 100 and Eq. (C.14) in the appendix.

Similar to the $SE(3)$ covariant spherical harmonic transformations $\mathcal{SH}$ and $\mathcal{LT}$ (dense Gauss-Laguerre transform), the angular Fourier transform induces an $SE(3)$ covariant filter, too: the projection of images onto the Fourier basis functions $\boldsymbol{\mathcal{B}}^{\ell}(k)$ can be regarded as local feature extraction. The features are spherical tensors and thus transform under rotations of the input image in a predictable, $SO(3)$ covariant manner. Any $SO(3)$ covariant feature extraction induces a filter (see the introduction and the preliminary section for details). In this scenario here, thanks to the differential formulation of the basis function, the filter output can be computed via one initial convolution, followed by applying the spherical tensor derivatives in a recursive manner.

Given an image $I \in \mathcal{T}_0$. The dense spherical Fourier transformation is an $SE(3)$ covariant filter $\mathcal{FT}^{\ell}\{k\} : \mathcal{T}_0 \to \mathcal{T}_{\ell}$ induced by a projection of the image onto the basis functions $\boldsymbol{\mathcal{B}}^{\ell}(k)$ via

$$\begin{aligned}
\mathcal{FT}^{\ell}\{I,k\}(\mathbf{x}) &:= \langle \mathfrak{t}_{(-\mathbf{x})} f, \overline{\boldsymbol{\mathcal{B}}^{\ell}(k)}\rangle = \underbrace{\int_{\mathbf{r}\in\mathbb{R}^3} f(\mathbf{r}+\mathbf{x})\boldsymbol{\mathcal{B}}^{\ell}(\mathbf{r},k)d\mathbf{r}}_{\substack{\propto(2\ell+1)\text{ 3D convolutions}\\ \text{(component by component)}}} \\
&= (-1)^{\ell}\int f(\mathbf{r})\boldsymbol{\mathcal{B}}^{\ell}(\mathbf{x}-\mathbf{r},k)d\mathbf{r} \qquad \text{(using sh-parity, Eq. (2.68))} \\
&= (-1)^{\ell}(\tfrac{\sqrt{t}}{-k})^{\ell}\int f(\mathbf{r})\boldsymbol{\nabla}^{\ell}\boldsymbol{\mathcal{B}}^{0}(\mathbf{x}-\mathbf{r},k)d\mathbf{r} \\
&= (\tfrac{\sqrt{t}}{k})^{\ell}\boldsymbol{\nabla}^{\ell}\underbrace{\int f(\mathbf{r})\boldsymbol{\mathcal{B}}^{0}(\mathbf{x}-\mathbf{r},k)d\mathbf{r}}_{\text{one 3D convolution}} \quad .
\end{aligned} \tag{5.50}$$

This Fourier transformation can hardly be used in practice for local feature analysis, because the Bessel functions representing the radial direction are not vanishing with increasing distance to the center. However, by windowing the Bessel function with a Gaussian window function, we overcome this drawback. We obtain a new windowed Fourier basis with local support. We call the resulting functions *spherical Gabor kernels*.

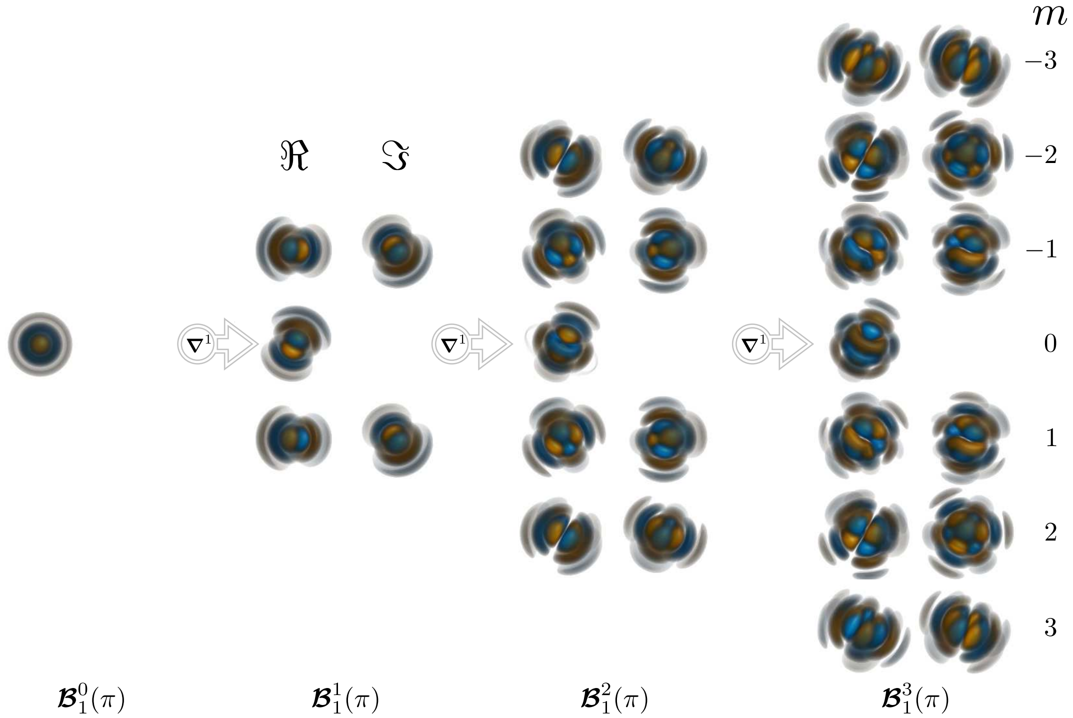


Figure 5.11.: Isosurfaces of spherical Gabor kernels of order $\ell = 0, 1, 2, 3$. Spherical harmonics represent the angular part while Gaussian windowed spherical Bessel functions cover the radial direction.

5.2.2. Spherical Gabor Kernels: Frequency Selection in the Gauss-Bessel Domain

The Fourier transform (Eq. (5.50)) transforms whole images into the Fourier domain. However, we are interested in studying the frequencies of images locally and simultaneously we want to benefit from a fast, dense transformation in terms of tensor derivatives, like we do in case of the Gauss-Laguerre transform. Therefore we will slightly alter the transformation in Eq. (5.50).

According to Eq. (5.48), the Fourier correspondence of $\mathcal{B}^0$ is just a sphere in frequency domain. Smoothing the sphere in frequency domain corresponds to windowing $\mathcal{B}^0$ in spatial domain. We call

$$\mathcal{B}_s^0(\mathbf{r}, k) := j_0(k\tfrac{r}{\sqrt{t}})e^{\frac{-r^2}{(2ts)}} \tag{5.51}$$

the 0-order spherical Gabor kernel. The parameter $s \in \mathbb{R}_{>0}$ steers the size of the Gaussian with respect to the frequency independently from the scale parameter t. With $t = \sqrt{\sigma}$ we can represent the scale in terms of the standard deviation. The kernel $\mathcal{B}_s^0$ selects a frequency range in frequency domain without making any decision about the orientation. This scenario is the initial case (middle-left) illustrated in Fig. 5.10 a) on page 131. The explicit expression for the Gaussian smoothed sphere is

$$\widetilde{\mathcal{B}}_s^0(\mathbf{k}, k') = \frac{\sqrt{2}\pi^{\frac{5}{2}} t\sqrt{s}}{kk'}(e^{-(k-k'\sqrt{t})^2 s/2} - e^{-(k+k'\sqrt{t})^2 s/2}) \quad , \tag{5.52}$$

which has a Gaussian-like profile in radial direction; see Fig. 5.12. For the derivation of the expression see Eq. (B.42) in the appendix. While the frequency is fixed by the choice of $\boldsymbol{\mathcal{B}}_s^0$, the orientation space is decomposed into its frequency components via spherical tensor derivatives. We denote by $\boldsymbol{\mathcal{B}}_s^\ell$ the ℓth tensor derivative of $\boldsymbol{\mathcal{B}}_s^0$, where $\boldsymbol{\mathcal{B}}_s^\ell(k) = (-1)^\ell \boldsymbol{\nabla}^\ell \boldsymbol{\mathcal{B}}_s^0(k)$. We call $\boldsymbol{\mathcal{B}}_s^\ell$ a spherical Gabor kernel of order ℓ. In Fig. 5.11 on the preceding page we show some 3D renderings of such kernels.

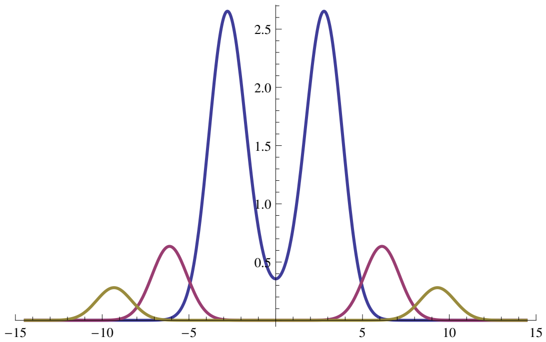


Figure 5.12.: Radial profile for a Gaussian smoothed sphere, corresponding to a radius of $\pi, 2\pi$ and 3π.

With $\widetilde{\boldsymbol{\nabla}^\ell \mathbf{f}} = i^\ell \mathbf{R}^\ell(\mathbf{k})\widetilde{\mathbf{f}}$ (according to Eq. (4.8) on page 98) we obtain the following equation of $\widetilde{\boldsymbol{\mathcal{B}}}_s^\ell$ in frequency domain:

$$\widetilde{\boldsymbol{\mathcal{B}}}_s^\ell(\mathbf{k}, k') = (-i)^\ell \mathbf{R}^\ell(\mathbf{k}) \frac{\sqrt{2}\pi^{\frac{5}{2}} t \sqrt{s}}{kk'} \left(e^{-(k-k'\sqrt{t})^2 s/2} - e^{-(k+k'\sqrt{t})^2 s/2}\right) \quad . \tag{5.53}$$

In Fig. 5.10 b) we show the center slices of $[\widetilde{\boldsymbol{\mathcal{B}}}_s^\ell(k)]_\ell$, the ℓth components of $\widetilde{\boldsymbol{\mathcal{B}}}_s^\ell(k)$ together with their spatial counterparts for an increasing order $\ell = 1 \cdots 5$. For an explicit representation of the kernels $\boldsymbol{\mathcal{B}}_s^\ell(k)$ see theorem C.3.4 on page 217.

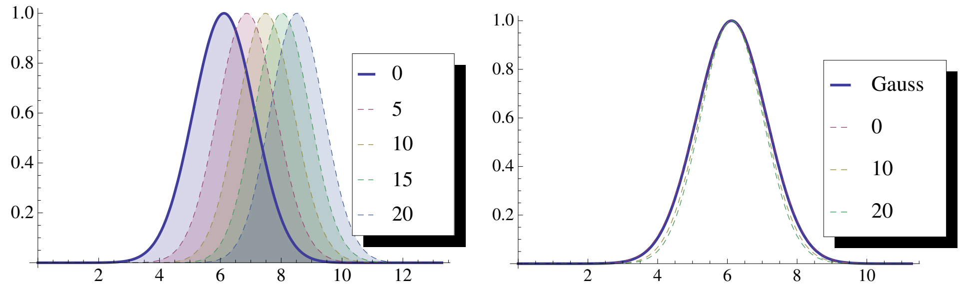
We can observe that with an increasing order ℓ the smoothed sphere is decomposed in terms of spherical harmonics, while the radial profile stays almost the same. It is worth mentioning that a small frequency shift to higher frequencies sneaks in during the successive application of the tensor derivatives. However, the effect is only noticeable for kernels of very high order. For our applications the effect is negligible, because we will never go to such high bandwidths; see Fig. 5.13 on the following page.

The ordinary Gabor wavelet with frequency k can be very well approximated by superposing a small number of Gauss-Bessel functions $\boldsymbol{\mathcal{B}}_s^\ell(k)$ according to

$$\omega_s(\mathbf{r}, \frac{\mathbf{k}}{\sqrt{t}}) \approx \sum_\ell (i)^\ell (2\ell+1) \alpha_\ell(k) \mathbf{Y}^\ell(\mathbf{k}) \bullet_0 \boldsymbol{\mathcal{B}}_s^\ell(\mathbf{r}, k) \quad , \tag{5.54}$$

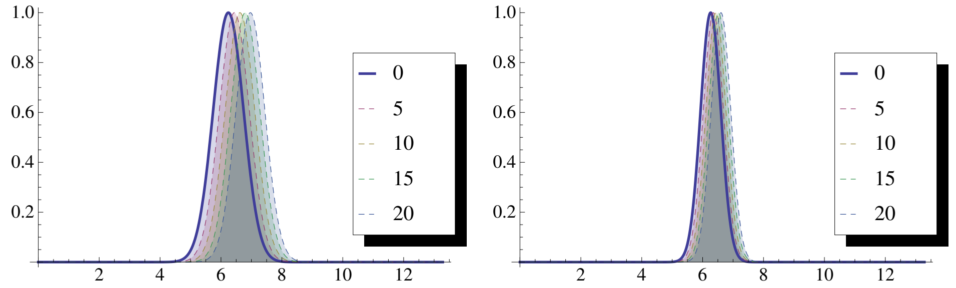
where $\alpha_\ell(k) \in \mathbb{R}$ are real valued weighting factors. In the second row of Fig. 5.14 we show a band-limited ($\ell \leq 5$) approximation of the Gabor wavelet using this spherical expansion.

We conclude that similar to the ordinary Gabor ω_s, the functions $\boldsymbol{\mathcal{B}}_s^\ell(k)$ are very well suited for the local analysis of frequencies in 3D images. Moreover, the spherical decomposition of the wave in frequency domain allows for locally studying a certain frequency with respect to the continuous range of all orientations simultaneously.



(a) The shift in frequency domain for $s = 1$, $k = 2\pi$ and $\ell = 0, 5, 10, 15, 20$.

(b) The profile of the tensor derivatives of a Gaussian smoothed sphere is similar to a Gaussian, even for derivatives of very high order (we aligned and scaled the functions with respect to their maximum).



(c) The shift in frequency domain for $s = 2$, $k = 2\pi$ and $\ell = 0, 5, 10, 15, 20$.

(d) The shift in frequency domain for $s = 3$, $k = 2\pi$ and $\ell = 0, 5, 10, 15, 20$.

Figure 5.13.: While the radial profile of the sphere stays Gaussian-like, we have a small frequency shift towards higher frequencies for higher order spherical Gabor kernels $\mathcal{B}_s^\ell$.

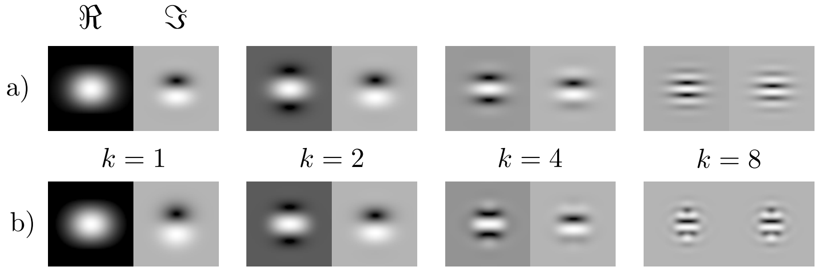


Figure 5.14.: a) Centered z-slices showing the 3D-Gabor wavelet with $\sigma = \sqrt{t} = 12, s = 0.5$. b) Proposed approximation based on the spherical derivatives of the Gaussian windowed Bessel functions (symmetric real part together with the antisymmetric imaginary part for $k = 1, 2, 4, 8$). The image size is 64^3 voxels. A spherical expansion up to order $\ell = 5$ based on Eq. (5.54) already leads to a very good approximation. For $k = 8$ the angular resolution is insufficient since the frequency is correct, while the shape differs.

$$\begin{tikzcd}(I * \boldsymbol{\mathcal{B}}_s^0(k)) = \mathbf{a}^0(k) \arrow[r, dotted, "\boldsymbol{\nabla}^1", "\mathcal{T}_0 \to \mathcal{T}_1"'] & \mathbf{a}^1(k) \arrow[r, dotted, "\boldsymbol{\nabla}^1", "\mathcal{T}_1 \to \mathcal{T}_2"'] & \mathbf{a}^2(k) \arrow[r, dotted, "\boldsymbol{\nabla}^1", "\mathcal{T}_2 \to \mathcal{T}_3"'] & \mathbf{a}^3(k)\end{tikzcd}$$

Figure 5.15.: Recursive implementation of the Gabor transform using spherical up-derivatives. Each scale and frequency requires only one initial convolution.

5.2.2.1. An $SE(3)$ Covariant Spherical Gabor Transform

We have seen that by windowing and differentiating the 0-order angular Basis functions $\boldsymbol{\mathcal{B}}^0$ we obtain a set of spherical Gabor kernels $\boldsymbol{\mathcal{B}}_s^\ell$, suitable to locally studying the frequencies of images. By accordingly replacing the functions in Eq. (5.50) , we get an $SE(3)$ covariant spherical Gabor transform $\mathcal{GT}^\ell\{k\} : \mathcal{T}_0 \to \mathcal{T}_\ell$, with

$$\begin{aligned}\mathcal{GT}^\ell\{I,k\}(\mathbf{x}) &:= \langle \mathfrak{h}_{(-\mathbf{x})} I, \overline{\boldsymbol{\mathcal{B}}_s^\ell(k)} \rangle \\ &= (\tfrac{\sqrt{t}}{k})^\ell \boldsymbol{\nabla}^\ell \int I(\mathbf{r}) \boldsymbol{\mathcal{B}}_s^0(\mathbf{x} - \mathbf{r}, k) d\mathbf{r} \\ &= (\tfrac{\sqrt{t}}{k})^\ell \boldsymbol{\nabla}^\ell (I * \boldsymbol{\mathcal{B}}_s^0(k))(\mathbf{x}) \quad . \end{aligned} \tag{5.55}$$

The filter output

$$\mathbf{a}^\ell(k) := \mathcal{GT}^\ell\{I,k\} \tag{5.56}$$

is a spherical tensor field of order ℓ, whose elements are the local Gabor frequency coefficients.

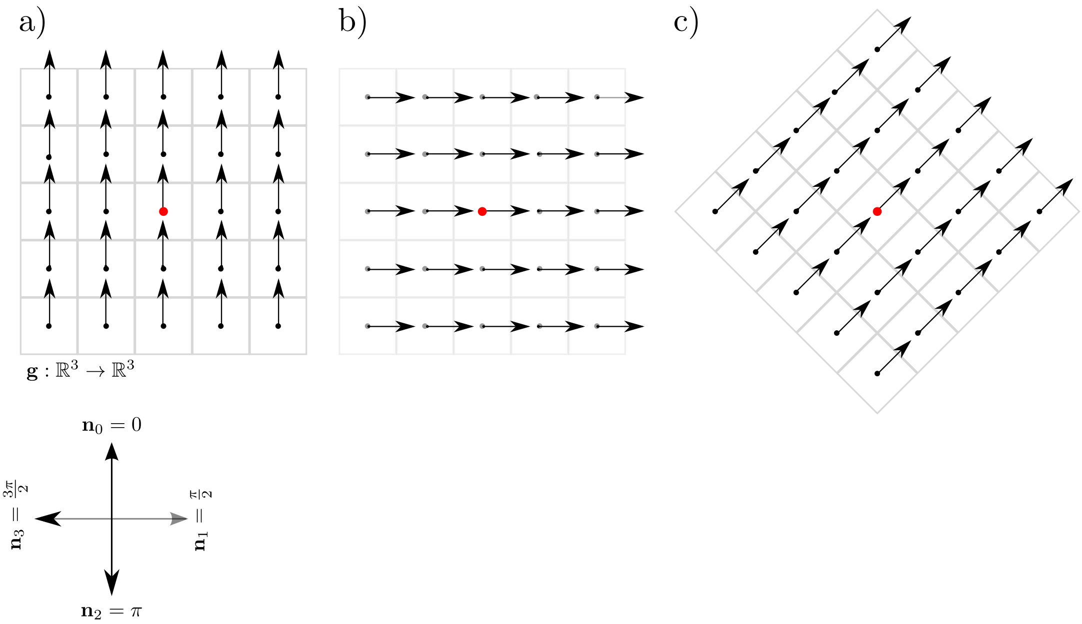


Figure 5.16.: The Histogram of oriented gradients (HOG) counts the occurrence of local gradient orientations. In 2D, HOG can be considered as function on a circle. The orientation space is commonly discretized corresponding to an equidistant sampling of the unit-circle. In the example shown here, we distinguish between four different orientations, corresponding to $\mathbf{n} = 0, \frac{\pi}{2}, \pi$ and $\frac{3\pi}{2}$. The light blue circles are representing the histograms, corresponding to the gradient directions in a-c). The red entries are representing the amount of gradients contributing to a certain direction. We see that HOG based on a discrete sampling of the orientation space is neither rotation invariant a-b) nor rotation covariant c).

5.3. SHOG - Spherical Histograms of Oriented Gradients

In this section we introduce a new sort of $SE(3)$ covariant filters for 3D images. The filter outputs are spherical tensor valued feature images. The tensors are expansion coefficients of functions representing the local occurrence of gradient orientations of the input image. We call a function representing the occurrence of 3D gradient orientations in 3D SHOG, a **S**pherical **H**istogram of **O**riented **G**radients. Parts of this section have been presented at conferences (Liu et al. 2011b; Skibbe et al. 2011b). A 2D alternative has been presented by Skibbe and Reisert (2012a).

Representations of gradient orientation distributions are widely used as reliable features for image patch description. The most frequently used representative of this group of features might be the structure tensor. The structure tensor is a Cartesian second order tensor which represents the local distribution of gradient orientations (Bigun and Granlund 1987; Förstner 1986); a definition can be found in example 2.2.4 in the preliminary chapter. The eigenvalues and eigenvectors of the structure tensor are providing reliable information about the local curvature in images and so, the structure tensor has become an indispensable, widely used tool for the detection of low-level features like corners (Förstner 1986; Noble 1988), edges (Harris and Stephens 1988; Köthe 2003) or the analysis of textures (Bigun et al. 1991; Lindeberg 1993; Rao and Schunck 1991). Further applications are ranging from diffusion filtering (Weickert 1998) to optical flow techniques (Bigun et al. 1991). A survey on structure tensors can be found e.g. in Granlund and Knutsson (1995).

The three eigenvectors of the 3D structure tensor together with their corresponding eigenvalues are parameterizing an ellipsoid that represents the local distribution of gradient orientations. Due to this very coarse representation of the distribution the structure tensor is less suited to detect and distinguish complex structures.

In many object detection tasks, a discrete representation by a histogram is used. The histogram is computed by discretization of the orientation space so that the occurrence of gradient orientations is countable in a discrete manner. Such a representation is richer in details than a structure tensor and allows for estimating the continuous gradient orientation distribution in a computationally efficient manner. Such representations are commonly used as discriminative image features with the purpose of detecting arbitrary structures. Their most famous representatives are SIFT features (Scale Invariant Feature Transform; see (Lowe 2004)) and HOG features (Histogram of Oriented Gradients; see (Dalal and Triggs 2005)). SIFT and HOG have shown a remarkable performance in various applications.

HOG features can be regarded as the most "natural" histogram that represents the occurrence of gradient orientations. In order to compute a HOG from an image patch, a (cyclic) orientation histogram is used to count the occurrence of gradient orientations. A gradient is usually contributing by its gradient magnitude so that very low gradients are only little contributing to the orientation statistic. Since the orientation space is discretized, the weights are usually distributed over adjacent bins via interpolation. The orientation space is typically coarsely sampled so that HOG histograms become robust against small deformations and small rotations. The strong advantage that HOG can be efficiently computed densely for the whole image via the integral image technique (Viola and Jones 2001) comes with the disadvantage that the HOG histogram has no well defined rotation behavior and thus rotation invariance cannot be achieved analytically; see Fig. 5.16 on the preceding page.

HOG are used in applications ranging from human detection (Dalal and Triggs 2005; Zhu et al. 2006) to state-of-the-art object recognition techniques in natural images (Bourdev and Malik 2009; Felzenszwalb et al. 2010). In recent work, HOG features have also been used for matching patches in consecutive frames in movies. These matches are used to cope with large displacements in optical flow estimation techniques (Brox and Malik 2011).

SIFT histograms can be regarded as a rotation invariant representation of HOG. That is, SIFT histograms are representing the local gradient orientation distribution in a rotation invariant manner. Rotation invariance is achieved via normalization (see Fig. 3.2 on page 86 regarding normalization versus group integration): in 2D, the SIFT histogram is computed with respect to the local reference coordinate system defined by the locally dominant gradient direction within an image patch (The gradient main direction). The gradient main direction is computed from within an isotropic region around the center of the patch. The advantage of rotation invariance comes with the disadvantage that a reliable, dominant direction cannot be guaranteed for all pixels in an image; see e.g. Lin et al. (2012). Due to this fact, SIFT histograms are less suited to be computed densely for the whole image. Moreover, compared to HOG, SIFT is much more computationally expensive. Therefore, SIFT features are only computed on certain key-points within the image. An initially applied key-point detector finds points that most likely provide a reliable reference frame. Moreover, scale-space techniques are used to make SIFT scale invariant; see the paper by Lindeberg (1990).

Applications of SIFT are ranging from object recognition (Lowe 2004; Nister and Stewenius 2006), face authentication (Bicego et al. 2006), action recognition (Scovanner et al. 2007), or for solving point matching problems in medical images (Cheung and Hamarneh 2007a). SIFT has also found its way into recent optical flow techniques, where densely computed SIFT histogram features are used instead of the raw intensity values (Liu et al. 2011a). Extension to 3D have been proposed by Allaire et al. (2008), where SIFT features are used to align medical images.

Integrating the integral image technique into SIFT leads to SURF (Speed Up Robust Features; Bay et al. (2006)). A 3D version of SURF has been introduced by Knopp et al. (2010). The resulting features can be regarded as a combination of the advantages of both histogram representations HOG and SIFT. SURF features can be computed more efficiently and they provide a rotation invariant representation.

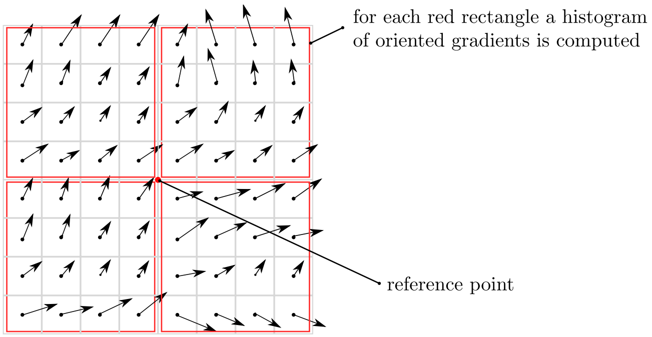


Figure 5.17.: The SIFT or HOG descriptor is based on several concatenated histograms of oriented gradients (HOG). In this example, the four red rectangles are indicating the four regions from within four HOG are computed. The final HOG descriptor is then formed by concatenating all four histograms. In case of SIFT, the grid (here 2 × 2) is aligned with respect to the local gradient main direction in order to achieve rotation invariance.

It is worth mentioning that for describing image patches in a most discriminative manner by HOG or SIFT, a grid is provided that subdivides an image patch into regions. For each region a HOG or SIFT histograms is computed. A patch descriptor is formed by concatenating the histograms; see Fig. 5.17. Such a representation considers the spatial differences of the occurrence of gradient orientations within an image patch. In case of SIFT, rotation invariance is achieved by aligning the whole grid with respect to the patch's gradient main direction. An illustration of such a SIFT descriptor for 3D images can be found in the experiment section in Fig. 6.7 on page 164. For instance, in case of the original 2D SIFT descriptor, the descriptor is formed from histograms computed on a local 4×4 grid. Alternatively, histograms can be formed from circular regions (Dalal and Triggs 2005).

The difficult part in designing rotation invariant SIFT-, HOG- or SURF-like features for volumetric images is the 3D rotation. With 3D SIFT (Allaire et al. 2008) and 3D SURF (Knopp et al. 2010) there exist 3D approaches that cope with this challenge. The 3D rotation comes mainly with two difficulties:

(1) Following the idea of 3D SIFT and 3D SURF, the gradient histograms are aligned with respect to a locally dominant direction of a local image patch. While in 2D only one dominant direction is required, two orthogonal dominant directions are necessary to define a local coordinate system for alignment in 3D (see Fig. 6.7 on page 164). Unfortunately, finding two orthogonal, reliable directions is less likely than finding one as it is the case in 2D. This is particularly the case when working with volumetric images, where classical edges do not exist. In 3D we most likely have blob like structures and/or surfaces. Particularly when working with biomedical data we seldom have clear, dominant edges or corners formed by connected surfaces.

(2) Directly extending the existing approaches from 2D to 3D requires a discrete sampling of the 3D orientation space of a 3D gradient to form the orientation bins of a histogram. An equidistant sampling of the orientation space in 2D is trivial. It is just an equidistant sampling of the circle (Fig. 5.18 a)). But equidistant sampling of the orientation space of a gradient in 3D is in general a non-trivial task. In 3D, a gradient's direction is defined by two angles (Fig. 5.18 b)). Hence the histogram of oriented gradients in 3D can be regarded as a function on the 2-sphere. However, we know that an equidistant sampling of the sphere is difficult, because in general it can not be solved explicitly (Thomson 1904). Moreover, a dense sampling of the sphere requires significantly more sampling points than a circle (compare Fig. 5.18 a) with b)). An alternative is sampling the orientation space with respect to the two angles representing a gradient direction; see Fig. 1.5 on page 21. However, in this case the poles are oversampled which must be considered in the computation (Allaire et al. 2008; Knopp et al. 2010). Furthermore, such a sampling does not solve the problem of having many more sampling points in the 3D case in comparison with the 2D scenario.

In this section we introduce SHOG, a new representation of histograms of oriented gradients in 3D that combines the structure tensor with the concepts of HOG and SIFT. We compute and represent a gradient orientation distribution directly in Fourier domain using spherical tensor representations. As a consequence, similar to the structure tensor, we densely compute a continuous representation of the local gradient orientation distribution. Similar to HOG and SIFT, our proposed representation covers finer details, too. Our SHOG representation can be regarded as a generalized form of the 3D structure tensor (we will discuss this later on in this section). Similar to HOG and the structure tensor, we compute the histograms densely for the whole image. And finally, similar to SIFT and the structure tensor, our method transforms under rotations in a predictable, continuous, well defined manner.

The major difference to HOG, SIFT and SURF is that we propose to keep the histogram continuous and realize the ”binning“ in the frequency domain by using a band limited expansion in terms of spherical harmonics. By binning in frequency domain we gain the following advantages:

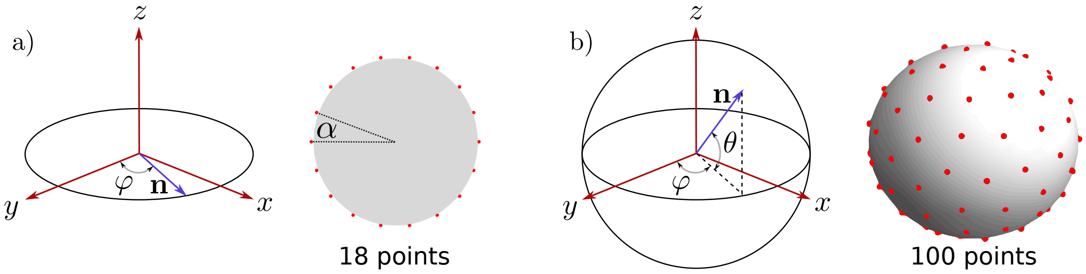


Figure 5.18.: In 3D the gradient direction is described by two angles thus a histogram of oriented gradients (HOG) can be considered as function on the sphere. We propose to represent the spherical histogram in Fourier domain in order to avoid angular discretization in spatial domain. This is mainly for two reasons: (1) in contrast to the 2D case, equidistant sampling of the orientation space of a gradient in 3D is a nontrivial task. This is the case because this problem is equivalent to distributing points on a sphere in an equidistant manner which for many configurations cannot be computed explicitly. (2) The number of necessary points required for sampling the sphere is significantly larger than for the 2D case (For $\alpha = 20°$ we have 18 points on a circle and 100 points on the sphere). This hinders a fast computation in 3D.

- No interpolation (with respect to bins) is required because, similar to the structure tensor, SHOG is based on the true continuous distribution function.
- SHOG rotates continuously with respect to rotation of its underlying data without leading to any artifacts that may occur when using a discrete representation of the orientation histogram.
- A representation in terms of spherical harmonics allows for using spherical tensor algebra. Thanks to spherical tensor calculus we can derive rotation invariants from the new HOG representation in an analytical manner.

This section focuses on the design of a filter mapping images to spherical tensor valued feature images. A feature image's elements are the expansion coefficients of the irreducible representations of local SHOG histograms (irreducible regarding 3D rotations). Details regarding invariant features derived from spherical HOG expansion coefficients can be found in chapter 3 on page 83. The design of specific region descriptors mimicking the concatenation of several neighboring histograms (like in SIFT and HOG) in a covariant manner is introduced and used in the experiments chapter 6 on page 157.

5.3.1. Spherical Histograms of Oriented Gradients

Given a volumetric image $I \in L_2(\mathbb{R}^3)$. We call the output of the mapping

$$\mathrm{SHOG}_w : L_2(\mathbb{R}^3) \rightarrow L_2(S_2), \qquad (\mathrm{SHOG}_w : \text{image} \rightarrow {}^{\text{gradient orientation}}_{\text{distribution}}) \tag{5.57}$$

a spherical histogram of oriented gradients, shortly SHOG. The spherical function SHOG is representing the occurrence of gradient orientations in an image. With S_2

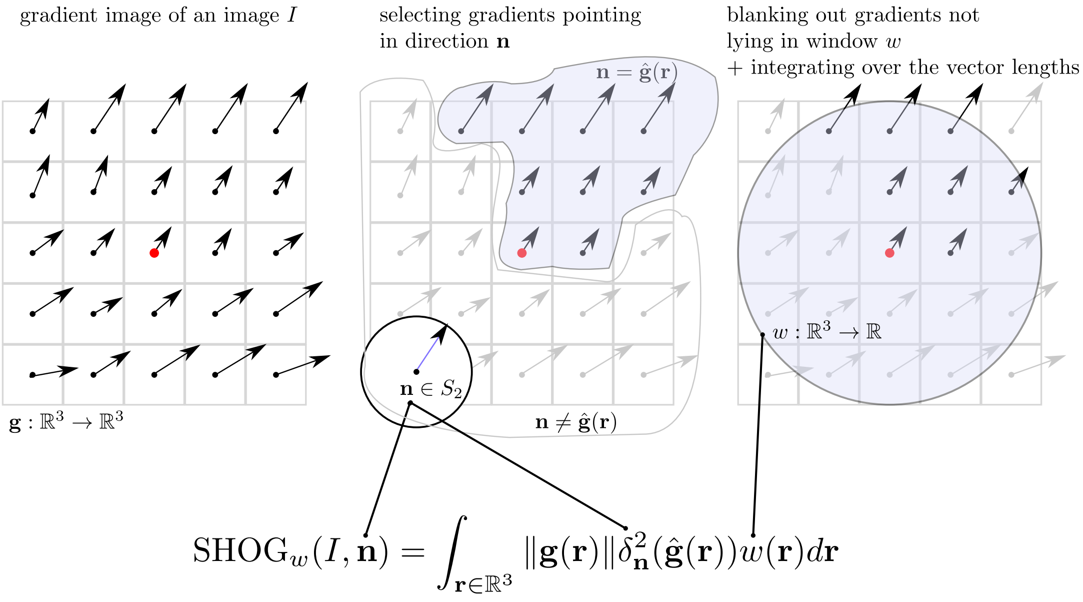


Figure 5.19.: Computation of a histogram of oriented gradients. For illustration, we consider a scenario in 2D.

we denote the unit-sphere which is used to represent the angular depended gradient orientation occurrence statistic. We denote by w an isotropic window function $w \in L_2(\mathbb{R}^3)$. The window function defines the volume within the gradient orientations are contributing to a histogram. A suitable window function is e.g. the 3D Gaussian function or a Gaussian smoothed sphere. Isotropy is required, because only then the window is not changing its shape under rotations. This facilitates the extraction of rotation invariant features. For anisotropic smoothing within the proposed concept see the paper by Reisert and Skibbe (2012).

We compute a SHOG function by counting the occurrence of gradient orientations within the window function w. Let $\mathbf{g} : \mathbb{R}^3 \to \mathbb{C}^3$, $\mathbf{g} = \nabla I$ be the gradient image of I. The resulting continuous gradient orientation distribution function has the form

$$\mathrm{SHOG}_w(I, \mathbf{n}) := \int_{\mathbf{r} \in \mathbb{R}^3} \|\mathbf{g}(\mathbf{r})\|^{\gamma} \delta^2_{\mathbf{n}}(\hat{\mathbf{g}}(\mathbf{r})) w(\mathbf{r}) d\mathbf{r} \quad . \tag{5.58}$$

An illustrative example is given in Fig. 5.19. The function $\hat{\mathbf{g}} : \mathbb{R}^3 \to S_2$, with

$$\hat{\mathbf{g}}(\mathbf{r}) := \mathbf{g}(\mathbf{r}) / \|\mathbf{g}(\mathbf{r})\| \quad , \tag{5.59}$$

is the gradient orientation field, with $\mathbf{n} \in S_2$ we denote the current histogram entry (the direction) taken into account. By $\delta^2_{\mathbf{n}} : S_2 \to \mathbb{R}$ we denote the Dirac delta function on the unit sphere (see Eq. (C.11) in the appendix) that selects those gradients out of the gradient image $\mathbf{g}$ with orientation $\mathbf{n}$. We weight the count of a gradient $\mathbf{g}(\mathbf{r})$ by its magnitude $\|\mathbf{g}(\mathbf{r})\|^{\gamma}$. With the parameter $\gamma \in \mathbb{R}_{>0}$ we can steer the gradient weights in a nonlinear manner. Similar to a gamma correction, we can use values $\gamma < 1$ to emphasize the influence of small gradients and to damp large outliers.

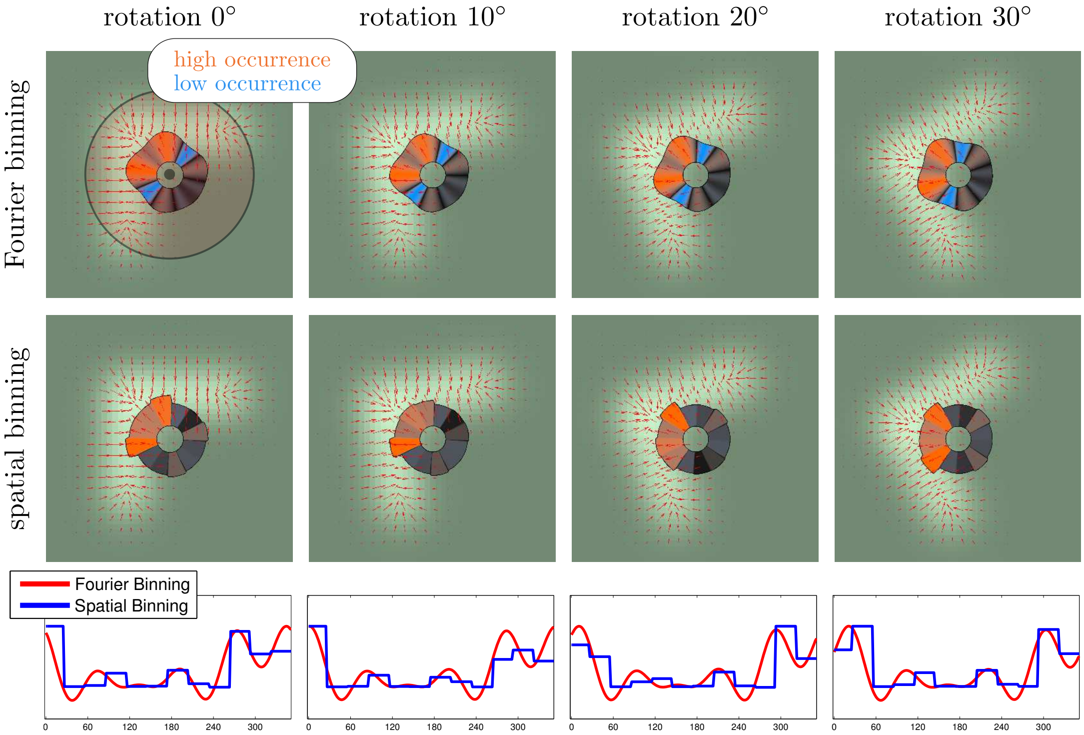


Figure 5.20.: The histogram of oriented gradients (HOG) represents the local occurrence of gradient orientations. The black circle indicates the region which is considered to compute a HOG with respect to the image center. The first row shows the proposed HOG feature (in 2D), where binning is realized in frequency domain (We used six complex valued coefficients). We see that when rotating the image, the histogram smoothly rotates with respect to the gradients. In the second row we see a HOG in its classical representation (here 12 bins). Due to binning in spatial domain, only rotations which are consistent with the number of bins (here every 30°) are rotated versions of the "original" histogram. Even soft-binning alters the appearance of the histogram during rotations. This is because rotations are acting on the histograms in a non-linear, non-invertible fashion. The third row shows the cyclic profiles of the histograms.

5.3.1.1. The Irreducible Representations of SHOG

Since SHOG is a function on the sphere, there exists a representation in terms of spherical harmonics, namely

$$\mathrm{SHOG}_w(I,\mathbf{n}) = \sum_{\ell=0}^{\infty}(2\ell+1)(\mathbf{v}_w^{\ell})^T\overline{\mathbf{Y}^{\ell}(\mathbf{n})} \quad . \tag{5.60}$$

We denote by $\mathbf{v}_w^{\ell} \in \mathbb{C}^{2\ell+1}$ the expansion coefficients, completely representing the SHOG in the spherical harmonic domain. When neglecting higher frequency components by limiting the band $\ell \leq L$ we obtain the best-approximation of SHOG_w in the finite subspace spanned by $\{\mathbf{Y}^i\}_{\{i=0\cdots L\}}$. We consider the band limiting parameter L as Fourier counterpart of the number of angular bins of the traditional, spatial HOG, because it steers the angular resolution of the histogram. In Fig. 5.21 on the following page we exemplarily depict the band limited expansion of a gradient histogram for an increasing number of frequency components with $L = 1, 2, 3, 4, 5$.

We identify the coefficients $\mathbf{v}^\ell_w$ by plugging the spherical expansion of the Dirac delta function (see section C.11 in the appendix) into Eq. (5.58). We get

$$\begin{aligned}
\mathrm{SHOG}_w(I,\mathbf{n}) &= \int_{\mathbf{r}\in\mathbb{R}^3} \|\mathbf{g}(\mathbf{r})\|^\gamma \delta^2_{\mathbf{n}}(\hat{\mathbf{g}}(\mathbf{r}))w(\mathbf{r})d\mathbf{r} \\
&= \int \|\mathbf{g}(\mathbf{r})\|^\gamma \underbrace{\sum_{\ell=0}^{\infty}(2\ell+1)(\mathbf{Y}^\ell(\hat{\mathbf{g}}(\mathbf{r})))^T\overline{\mathbf{Y}^\ell(\mathbf{n})}}_{=\delta^2_{\mathbf{n}}(\hat{\mathbf{g}}(\mathbf{r}));\text{ see Eq. (C.11) in the appendix}} \quad w(\mathbf{r})d\mathbf{r} \\
&= \sum_{\ell=0}^{\infty}(2\ell+1)\underbrace{\int \|\mathbf{g}(\mathbf{r})\|^\gamma(\mathbf{Y}^\ell(\hat{\mathbf{g}}(\mathbf{r})))^T w(\mathbf{r})d\mathbf{r}}_{=(\mathbf{v}^\ell_w)^T\in\mathbb{C}^{2\ell+1}}\overline{\mathbf{Y}^\ell(\mathbf{n})}
\end{aligned} \tag{5.61}$$

Hence the Fourier coefficients of SHOG, $\mathbf{v}^\ell_w \in \mathbb{C}^{2\ell+1}$, are the result of the angular Fourier transformation $\widetilde{SHOG}_w(\ell) : \mathcal{T}_0 \to \mathbb{C}^{(2\ell+1)}$. The transformation is defined by

$$\widetilde{SHOG}_w(I,\ell) := \int \|\mathbf{g}(\mathbf{r})\|^\gamma(\mathbf{Y}^\ell(\hat{\mathbf{g}}(\mathbf{r})))^T w(\mathbf{r})d\mathbf{r}\,. \tag{5.62}$$

That is, we compute the SHOG coefficients $\mathbf{v}^\ell_w \in \mathbb{C}^{2\ell+1}$ by aggregating the contributions of the 'higher order' gradient orientation fields $\|\mathbf{g}(\mathbf{r})\|\mathbf{Y}^\ell(\hat{\mathbf{g}}) \in \mathbb{C}^{2\ell+1}$ within the window function w in a component-by-component manner.

The SHOG coefficients $\mathbf{v}^\ell_w \in \mathbb{C}^{2\ell+1}$ are spherical tensors. Hence they rotate in a predictable manner: assume a 3D rotation denoted by $\mathfrak{g} \in SO(3)$ is acting on a volumetric image $I \in L_2(\mathbb{R}^3)$ so that $\mathfrak{g}I(\mathbf{x}) := I(\mathbf{U}^T_{\mathfrak{g}}\mathbf{x})$; see Eq. (2.38) on page 49. In this scenario the coefficients $\mathbf{v}^\ell_w$ are transforming according to

$$\mathrm{SHOG}_w(\mathfrak{g}I,\mathbf{n}) = \mathrm{SHOG}_w(I,\mathbf{U}^T_{\mathfrak{g}}\mathbf{n}) = \sum_{\ell=0}^{\infty}(2\ell+1)(\mathbf{D}^\ell_{\mathfrak{g}}\mathbf{v}^\ell_w)^T\overline{\mathbf{Y}^\ell(\mathbf{n})}\,. \tag{5.63}$$

As a consequence, the coefficients $\mathbf{v}^\ell_w$ are spherical tensors. A proof can be found on page 210.

5.3.1.2. An $SE(3)$ Covariant SHOG Filter

The angular Fourier transform of $\widetilde{SHOG}_w(I,\ell)$ maps images to rotation covariant spherical tensor valued expansion coefficients. Since any $SO(3)$ covariant local feature extraction induces an $SE(3)$ covariant filter, we use the Fourier transformation $\widetilde{SHOG}_w(I,\ell)$ to build a filter that maps images to dense SHOG coefficient feature images.

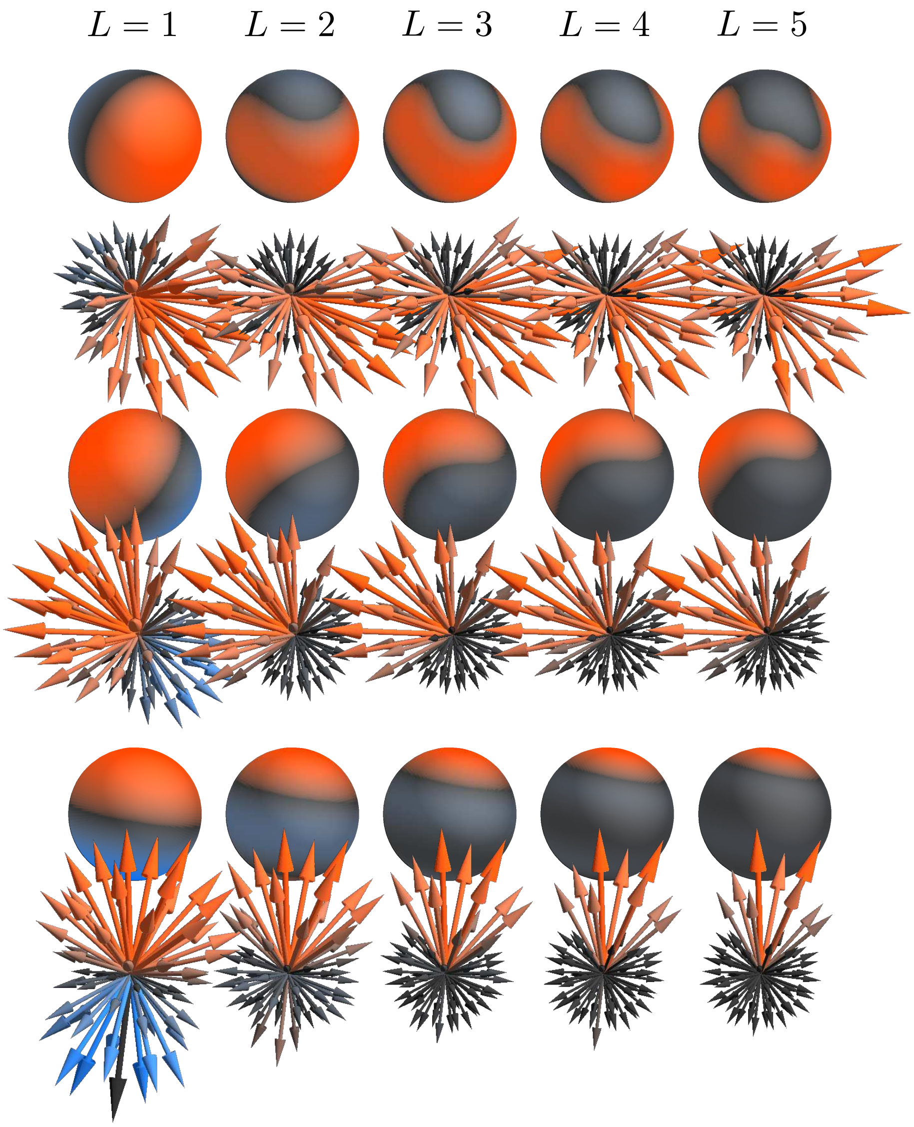


Figure 5.21.: We use a representation of spherical HOG in the spherical harmonic domain so that a few number of coefficients is sufficient to represent the characteristics of the true gradient orientation distributions. The expansion coefficients can be computed efficiently and rotate smoothly with respect to the underlying data. In this image we have visualized the spherical representations of three gradient orientation distributions for an increasing number of expansion coefficients $\ell \leq L, L = 1, 2, 3, 4, 5$. Black indicates 0, red represents positive values, blue negative values (Negative values occur because of the band-limit representation). We also show the corresponding quantized SHOG functions. Black arrows are indicating gradient orientations not contributing to the current histogram.

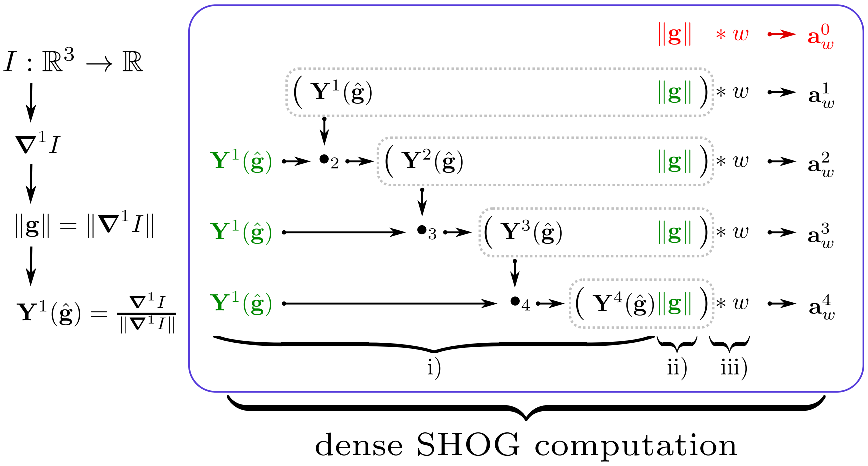


Figure 5.22.: The spherical expansion coefficients $\mathbf{a}^\ell_w$ can be computed very efficiently by utilizing a recurrence relation of the spherical harmonics and by utilizing the Fast Fourier Transform. In an initial step we compute the spherical gradient $\nabla^1 I$. We decompose the spherical gradient image into its corresponding gradient magnitude image $\|\mathbf{g}\| = \|\nabla^1 I\|$ and its spherical orientation field $\mathbf{Y}^1(\hat{\mathbf{g}})$, where $\mathbf{Y}^1(\hat{\mathbf{g}}) = \nabla^1 I/\|\nabla^1 I\|$ (see Eq. (5.68)). i) Higher angular moment states are then computed recursively utilizing tensor products $\bullet$ ii) and are then weighted by the gradient magnitude. iii) We finally use the Fast Fourier Transform to superimpose the angular moment states of neighboring gradients to form the coefficients.

We call the filter SHOG filter and denote it by $\mathcal{SHOG}^\ell : \mathcal{T}_0 \rightarrow \mathcal{T}_\ell$, where

$$\mathcal{SHOG}^\ell\{I,w\}(\mathbf{x}) := \widetilde{SHOG}_w(\mathfrak{t}_{(-\mathbf{x})}I,\ell) \quad . \tag{5.64}$$

By substituting the definition of the Fourier representation of SHOG (Eq. (5.62)) into Eq. (5.64), we obtain

$$\begin{aligned}\mathcal{SHOG}^\ell\{I,w\}(\mathbf{x}) &= \int \|\mathbf{g}(\mathbf{r}+\mathbf{x})\|^\gamma (\mathbf{Y}^\ell(\hat{\mathbf{g}}(\mathbf{r}+\mathbf{x})))^T w(\mathbf{r})d\mathbf{r} \\ &= \int \|\mathbf{g}(\mathbf{r})\|^\gamma (\mathbf{Y}^\ell(\hat{\mathbf{g}}(\mathbf{r})))^T w(\mathbf{r}-\mathbf{x})d\mathbf{r} \\ &= \int \|\mathbf{g}(\mathbf{r})\|^\gamma (\mathbf{Y}^\ell(\hat{\mathbf{g}}(\mathbf{r})))^T w(\mathbf{x}-\mathbf{r})d\mathbf{r} \quad \text{(since } w \text{ is isotropic)} \\ &= \Big(\big(\|\mathbf{g}\|^\gamma (\mathbf{Y}^\ell(\hat{\mathbf{g}}))\big) * w\Big)(\mathbf{x})\,. \end{aligned} \tag{5.65}$$

That is, we convolve the higher order spherical gradient fields $\|\mathbf{g}\|\mathbf{Y}^\ell(\hat{\mathbf{g}})$ with an isotropic window function $w \in L_2(\mathbb{R}^3)$.

Recursive Computation. Instead of explicitly computing the gradient orientation fields $\mathbf{Y}^\ell(\hat{\mathbf{g}})$, we utilize spherical tensor products in order to compute the coefficients in a recursive manner. We make use of the fact that we can recursively derive spherical harmonics of order $\ell \pm 1, \ell > 0$ by coupling two spherical harmonics of order ℓ and

1 with $\mathbf{Y}^{\ell\pm1} = \mathbf{Y}^{\ell} \bullet_{\ell\pm1} \mathbf{Y}^{1}$; see Prop. 2.2.29 on page 69 (We illustrate in Fig. 2.5 how higher order spherical harmonics can be computed recursively). By utilizing this property we gain a recursive rule with which we avoid an explicit, expensive computation of $\mathbf{Y}^{\ell}(\hat{\mathbf{g}})$. We compute the higher order orientation fields in a voxel-by-voxel way via

$$\mathbf{Y}^{\ell+1}(\hat{\mathbf{g}}(\mathbf{r})) = \mathbf{Y}^{\ell}(\hat{\mathbf{g}}(\mathbf{r})) \bullet_{\ell+1} \mathbf{Y}^{1}(\hat{\mathbf{g}}(\mathbf{r})) \quad . \tag{5.66}$$

Remember that a 3D image can be regarded as a spherical tensor field of order 0 so that $I \in \mathcal{T}_0$. Therefore, the basis case is the normalized ordinary gradient in spherical notation:

$$\mathbf{Y}^{1}(\hat{\mathbf{g}}) = \frac{(\frac{1}{\sqrt{2}}(\partial_x I - i\partial_y I), \partial_z I, -\frac{1}{\sqrt{2}}(\partial_x I + i\partial_y I))^T}{\|\mathbf{g}\|} \tag{5.67}$$

$$= \frac{(\frac{1}{\sqrt{2}}(\partial_x - i\partial_y), \partial_z, -\frac{1}{\sqrt{2}}(\partial_x + i\partial_y))^T I}{\|\mathbf{g}\|} = \frac{\boldsymbol{\nabla}^1 I}{\|\mathbf{g}\|} = \frac{\boldsymbol{\nabla}^1 I}{\|\boldsymbol{\nabla}^1 I\|}; \tag{5.68}$$

see Eq. (4.1) on page 97. The special case for $\ell = 0$ has “no orientation”, hence

$$\mathbf{Y}^{0}(\hat{\mathbf{g}}(\mathbf{r})) = \mathbf{Y}^{1}(\hat{\mathbf{g}}(\mathbf{r})) \bullet_{0} \mathbf{Y}^{1}(\hat{\mathbf{g}}(\mathbf{r})) = 1 \quad . \tag{5.69}$$

Once the gradient orientation fields $\mathbf{Y}^{\ell}(\hat{\mathbf{g}})$ are computed, the remaining computations are the convolutions with the window function w.

5.3.1.3. $\mathcal{SHOG}^{\ell}$, a Generalized 3D Structure Tensor Field

The spherical tensor field of the Fourier representations of the spherical histogram of oriented gradients (SHOG) can be regarded as generalization of the second order structure tensor field: if we consider the special case where $\gamma = 2$, the Fourier SHOG expansion coefficient fields $\mathbf{a}^0_w = \mathcal{SHOG}^0\{I, w\}$ and $\mathbf{a}^2_w = \mathcal{SHOG}^2\{I, w\}$ are identical with a spherical tensor representations of the 3D structure tensor field. That is, there exists a unitary transformation between the Cartesian structure tensor field and the zero and second order Fourier representations of SHOG.

Let $\boldsymbol{\mathcal{S}}\{I\} : \mathbb{R}^3 \to \mathbb{R}^9$ be the the structure tensor field of an image $I \in L_2(\mathbb{R}^3)$; see example 2.2.4 on page 51. We represent the structure tensor field w.l.o.g in matrix form:

$$\boldsymbol{\mathcal{S}}\{I\} = \begin{bmatrix} (\partial_x I)^2 & (\partial_x I)(\partial_y I) & (\partial_x I)(\partial_z I) \\ (\partial_x I)(\partial_y I) & (\partial_y I)^2 & (\partial_y I)(\partial_z I) \\ (\partial_x I)(\partial_z I) & (\partial_y I)(\partial_z I) & (\partial_z I)^2 \end{bmatrix} * w \quad . \tag{5.70}$$

The structure tensor is a symmetric tensor. Therefore, according to Eq. (2.148) on page 78, there exists a unique (invertible) decomposition into a traceless symmetric tensor and the trace of the structure tensor. Moreover, for both the symmetric part and the trace exists an unambiguous representation in terms of spherical tensor fields (see Eq. (2.154)), which are

$$\mathbf{s}^2 = \begin{pmatrix} \frac{1}{2}((\partial_x I)^2 - (\partial_y I)^2 + 2i(\partial_x I)(\partial_y I)) \\ ((\partial_x I)(\partial_z I) + i(\partial_y I)(\partial_z I)) \\ \frac{-1}{6}((\partial_x I)^2 + (\partial_y I)^2 - 2(\partial_z I)^2) \\ (-(\partial_x I)(\partial_z I) + i(\partial_y I)(\partial_z I)) \\ \frac{1}{2}((\partial_x I)^2 - (\partial_y I)^2 - 2i(\partial_x I)(\partial_y I)) \end{pmatrix} = \begin{pmatrix} \frac{1}{2}((\partial_x I) - i(\partial_y I))^2 \\ ((\partial_x I) + i(\partial_y I))(\partial_z I) \\ \sqrt{\frac{2}{3}}(-\frac{(\partial_x I)^2}{2} - \frac{(\partial_y I)^2}{2} + (\partial_z I)^2) \\ (-(\partial_x I) + i(\partial_y I))(\partial_z I) \\ \frac{1}{2}((\partial_x I) + i(\partial_y I))^2 \end{pmatrix}, \tag{5.71}$$

where $\mathbf{s}^2 \in \mathcal{T}_2$ represents the traceless, symmetric part of the tensor. For the trace we have

$$\mathbf{s}^0 = \frac{-((\partial_x I)^2 + (\partial_y I)^2 + (\partial_z I)^2)}{\sqrt{3}}, \tag{5.72}$$

where $\mathbf{s}^0 \in \mathcal{T}_0$ is a scalar valued image. It turns out that if $\gamma = 2$, then the corresponding SHOG contains the structure tensor in terms of the coefficients

$$\begin{aligned} \mathbf{a}^0_w &= \langle 10,10 | 00 \rangle^{(-1)} (\mathbf{s}^0 * w) \quad \text{and} \\ \mathbf{a}^2_w &= \langle 10,10 | 20 \rangle^{(-1)} (\mathbf{s}^2 * w) \quad , \end{aligned} \tag{5.73}$$

where $\langle 10,10 | 20 \rangle^{(-1)}$ and $\langle 10,10 | 00 \rangle^{(-1)}$ are two real-valued constant normalization factors.

Proof. We use Eq. (5.68) in combination with the explicit representation of the spherical derivatives, see Eq. (4.1) on page 97, to compute the weighted orientation fields for the coefficients $\mathbf{a}^2_w$ and $\mathbf{a}^0_w$ (before smoothing). The resulting tensor for $\mathbf{a}^2_w$ is

$$\begin{aligned} &\|\mathbf{g}\|^2 (\mathbf{Y}^1(\hat{\mathbf{g}}) \bullet_2 \mathbf{Y}^1(\hat{\mathbf{g}})) = \|\nabla^1 I\|^2 ((\tfrac{\nabla^1 I}{\|\nabla^1 I\|}) \bullet_2 (\tfrac{\nabla^1 I}{\|\nabla^1 I\|})) = ((\nabla^1 I) \bullet_2 (\nabla^1 I)) \\ &= \begin{pmatrix} \sqrt{\frac{3}{2}} \frac{1}{2} ((\partial_x I) - i(\partial_y I))^2 \\ \sqrt{\frac{3}{2}} ((\partial_x I) + i(\partial_y I))(\partial_z I) \\ (-\frac{(\partial_x I)^2}{2} - \frac{(\partial_y I)^2}{2} + (\partial_z I)^2) \\ \sqrt{\frac{3}{2}} (-(\partial_x I) + i(\partial_y I))(\partial_z I) \\ \sqrt{\frac{3}{2}} \frac{1}{2} ((\partial_x I) + i(\partial_y I))^2 \end{pmatrix} = \sqrt{\frac{3}{2}} \begin{pmatrix} \frac{1}{2} ((\partial_x I) - i(\partial_y I))^2 \\ ((\partial_x I) + i(\partial_y I))(\partial_z I) \\ \sqrt{\frac{2}{3}} (-\frac{(\partial_x I)^2}{2} - \frac{(\partial_y I)^2}{2} + (\partial_z I)^2) \\ (-(\partial_x I) + i(\partial_y I))(\partial_z I) \\ \frac{1}{2} ((\partial_x I) + i(\partial_y I))^2 \end{pmatrix} \\ &= \langle 10,10 | 20 \rangle^{(-1)} \begin{pmatrix} \frac{1}{2} ((\partial_x I) - i(\partial_y I))^2 \\ ((\partial_x I) + i(\partial_y I))(\partial_z I) \\ \sqrt{\frac{2}{3}} (-\frac{(\partial_x I)^2}{2} - \frac{(\partial_y I)^2}{2} + (\partial_z I)^2) \\ (-(\partial_x I) + i(\partial_y I))(\partial_z I) \\ \frac{1}{2} ((\partial_x I) + i(\partial_y I))^2 \end{pmatrix} \quad , \end{aligned} \tag{5.74}$$

which is equal to $\mathbf{s}^2$ up to the normalization factor $\langle 10,10 | 20 \rangle^{(-1)} \in \mathbb{R}$. Similarly for the spherical tensor field $\mathbf{a}^0_w$, we get

$$\begin{aligned} &\|\mathbf{g}\|^2 (\mathbf{Y}^1(\hat{\mathbf{g}}) \bullet_0 \mathbf{Y}^1(\hat{\mathbf{g}})) = \|\nabla^1 I\|^2 ((\tfrac{\nabla^1 I}{\|\nabla^1 I\|}) \bullet_0 (\tfrac{\nabla^1 I}{\|\nabla^1 I\|})) = ((\nabla^1 I) \bullet_0 (\nabla^1 I)) = \|\nabla^1 I\|^2 \\ &= ((\partial_x I)^2 + (\partial_y I)^2 + (\partial_z I)^2) = (-\sqrt{3}) \frac{-((\partial_x I)^2 + (\partial_y I)^2 + (\partial_z I)^2)}{\sqrt{3}} \\ &= \langle 10,10 | 10 \rangle^{(-1)} \frac{-((\partial_x I)^2 + (\partial_y I)^2 + (\partial_z I)^2)}{\sqrt{3}} \quad , \end{aligned} \tag{5.75}$$

which is equal to $\mathbf{s}^0$ up to the normalization factor $\langle 10,10 | 00 \rangle^{(-1)} \in \mathbb{R}$. □

5.4. Implementation: Filters for Object Detection in 3D

In this section we list up implementation related details regarding the proposed spherical harmonic transformations. This also includes the descriptors that we will use in our experiment section. The purpose of this section is to give a short guideline that should help finding suitable settings for solving an object detection task.

One issue that we should address for all of the proposed discrete spherical harmonic transformations is the choice of an upper bound for the spherical harmonic order ℓ. Since the images are presented in a discrete voxel grid, there exists a natural limitation for the representation of frequencies within a local pattern: all zero crossings appearing on a spherical harmonic function should be representable in a discrete image, otherwise we risk aliasing effects, see Nyquist (1928).

We focus on the circular harmonic $e^{im\varphi}$ appearing in the term of the spherical harmonic function, see section 2.2.2.1 on page 54. It represents the equator of a spherical harmonic function. We focus on its highest frequency $m = \ell$. In this case, $e^{i\ell\varphi}$ has 2ℓ zero crossings. Hence if we assume a sampling on a sphere with radius $\sigma \in \mathbb{R}_{>0}$,

$$\ell < \pi\sigma \tag{5.76}$$

is a good estimation for an upper bound of ℓ. To give some numbers: if the radius of an object is 2 voxels, then $\ell \leq 6$ would be the maximum angular frequency that should be taken into account.

5.4.1. A Spherical Laguerre Transform for Object Detection in 3D

Regarding computational complexity, the Gauss-Laguerre transformation is the most efficient transformation among the proposed transformations. A big advantage of this transformation is the small number of parameters: we simply define a scale and an upper bound for the polynomial degree.

In Skibbe et al. (2012) we used Gauss-Laguerre descriptors for the detection and classification of cells in microscopical images of Arabidopsis root tips; see chapter A on page 181. We found that representing local image patches in a coarse to fine manner is crucial for successfully distinguishing between the cells.

If we have objects of varying size, then we suggest the following strategy for selecting a suited set of scale parameters: let $\sigma_0 \in \mathbb{R}$ be the radius of the biggest structures or objects that we are looking for. Then $t = \sigma_0^2$ gives us the size of the Gaussian kernel that covers these structures. We then build a whole cascade of different scales in a coarse to fine manner by repeatedly dividing σ by two, namely $\sigma_{n+1} = \sigma_n/2$. We stop when $\sigma_n \leq 2$.

In an object detection task, a training set and/or a validation set can be used to find a suitable σ_0. An upper bound for the polynomial degree is found by increasing the degree until no improvements can be achieved on the training set (or the condition Eq. (5.76) is violated).

Implementation For an efficient discrete realization of the spherical Gauss-Laguerre transformation (Eq. (5.34)) we use the discrete fast Fourier transform (we use the FFTW (Frigo and Johnson 2005)) for the initial convolution and the discrete

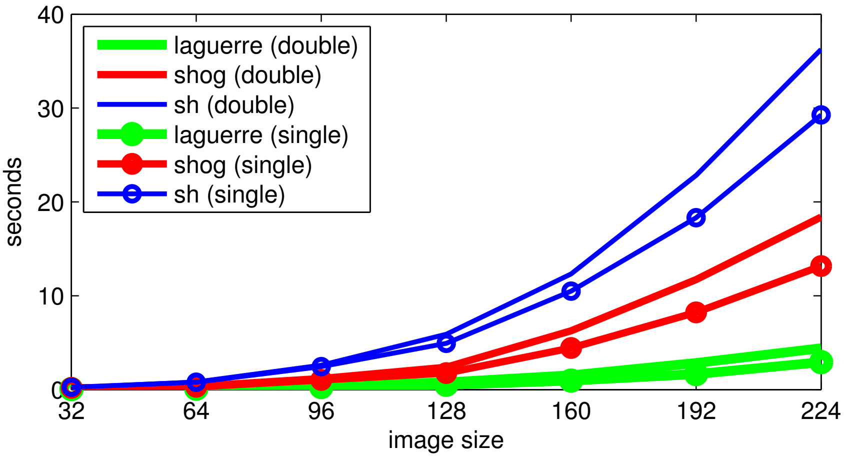


Figure 5.23.: Computation time: comparing the derivative based transformations (Gauss-Laguerre and Gabor, denoted as *laguerre*) to SHOG (denoted as *shog*) and the "standard" spherical harmonic transformation (denoted as *sh*). We compared the following scenarios: a transformation up to order 5 for datasets of the sizes 32^3,...,224^3 voxels using single and double precision numbers. The derivate based approaches clearly perform best. The SHOG expansion coefficients must be computed in a nonlinear manner which is computationally expensive. The standard transformation is based on a large number of convolutions with many different kernels which also requires a large amount of computation time and memory intensive operations. (making use of 8 cores of an Intel(R) Core(TM) i7-3770S CPU @ 3.10GHz with 32 GB of Ram via OpenMP and the multi-threaded FFTW(Frigo and Johnson 2005))

spherical tensor derivative operators introduced in section 4.2 on page 103. For a comparison of the computation time of the derivative based transformation with the SHOG transformation and the purely convolution based transformation see Fig. 5.23.

Computation Complexity Apart from one initial discrete convolution with complexity $\mathcal{O}(N \log N)$, where N is the number of voxels, only finite differences are required. The finite differences can be computed in $\mathcal{O}(MN)$, where M is the small number of scalar valued basis functions which we consider to be constant.

5.4.1.1. Gauss Laguerre Descriptors

For our experiments we use the power-spectrum (Eq. 3.27) and the bi-spectrum (Eq. 3.28) to form descriptor images from spherical Gauss Laguerre expansion coefficients. We also considered cross-spectrum invariants by combining expansion coefficients associated with different radial polynomial degree n. For the details regarding the parameters we refer to the experiment chapter.

5.4.2. A Spherical Gabor Transform for Object Detection in 3D

The spherical Gabor transformation allows for computing local features representing the frequency components of local patterns. Regarding this transformation, there are two major issues worth further investigation: scale and frequency. Since in our definition of spherical Gabor kernels scale and frequency are multiplicatively connected, we keep the scale constant in the following consideration and consider a varying frequency. The frequency parameter k determines the frequency of the radial spherical Bessel functions. To avoid artifacts induced by undersampling the radial profile of a spherical Gabor kernel, the zeros of the Bessel function should be clearly represented by its discrete representation. The common characteristic of all spherical Gabor functions of increasing order in frequency domain is that they are all lying on the same sphere (we are ignoring the small frequency shift here, because it has only little influence on the frequencies we take into account). Hence it is sufficient to focus on one representative. For brevity we focus on the zero crossings of the zero order Bessel function which has the simple form $j_0(k) = \frac{\sin k}{k}$, i.e. it is the sinc function. Hence the zeros of this Bessel function (considering only $k > 0$) are $\{\pi n \mid n \in \mathbb{N}_{>0}\}$. The corresponding spherical Gabor kernel is

$$\mathcal{B}_s^0(\mathbf{r}, k) := j_0(k\frac{r}{\sqrt{t}})e^{\frac{-r^2}{(2ts)}} \quad . \tag{5.77}$$

(see Eq. (5.51) on page 134). Therefore, taking the scale parameter t into account, we get the following upper bound which we must obey in order to avoid undersampling artifacts:

$$k \leq \sqrt{t}\pi \quad . \tag{5.78}$$

Similar to the Gauss-Laguerre transformation, we use a multi-scale representation for the spherical Gabor features: let $\sigma_0 \in \mathbb{R}$ be the radius of the biggest structures or objects that we are looking for. Then $t = \sigma_0^2$ gives us the size of the kernel that covers these structures. Given a frequency k, we build the whole cascade of different scales by repeatedly dividing by two, namely $\sigma_{n+1} = \sigma_n/2$, until Eq. (5.78) is violated. A training set or a validation set is used to find a suited σ_0.

For the selection of the frequencies we propose the following strategy: since the zero crossings of the zero order Bessel functions are a multiple of π, we suggest selecting the frequencies out of this set. In most cases it is sufficient to choose either $k \in \{n\pi \mid n \in \mathbb{N}_{>0}, n < N\}$, or in a more conservative setting $k \in \{n\frac{\pi}{2} \mid n \in \mathbb{N}_{>0}, n < N\}$, where N is a small constant.

An upper bound for the spherical harmonic order is found by increasing the order until no improvements can be achieved on a training set (and/or the condition Eq. (5.76) is violated).

In Fig. 5.24 on the next page we exemplarily show a multi-scale cascade that we use in our experiments, with $\ell \leq 5$ (only $\ell = 0, 5$ are depicted), $\sigma \in \{8, 4, 2\}$ and $k \in \{\pi, 2\pi, 3\pi, 4\pi, 5\pi, 6\pi\}$. We see that scale and frequency are well covered by these settings.

Implementation For an efficient discrete realization of the spherical Gabor transformation Eq. (5.55) we use the discrete fast Fourier transform for the initial

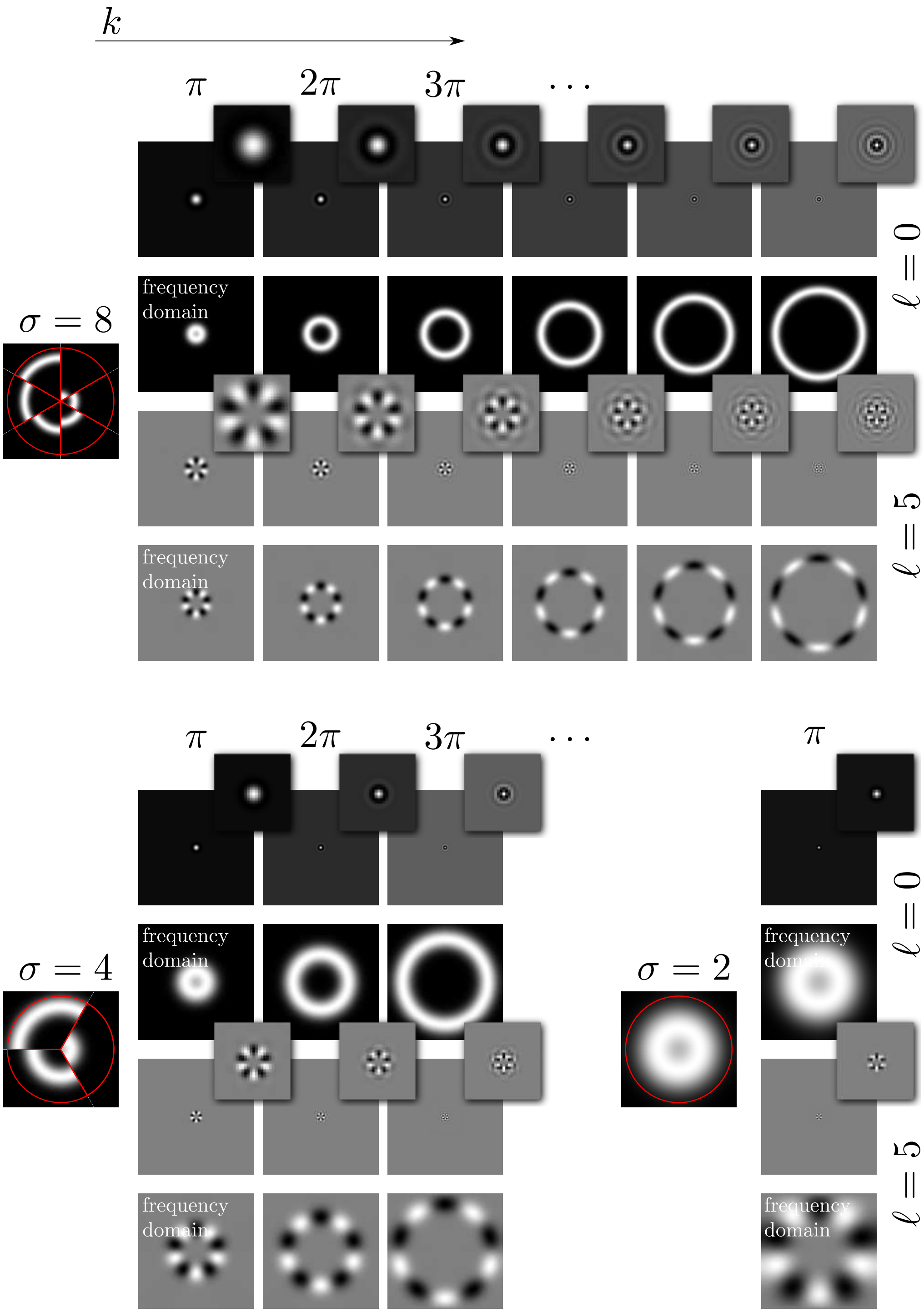


Figure 5.24.: Spherical Gabor kernels: a multi-scale cascade with parameters selected via the proposed strategy. Here we see the kernels in spatial and frequency domain for $\ell \leq 5$ (only $\ell = 0, 5$ are depicted), $\sigma \in \{8, 4, 2\}$ and $k \in \{\pi, 2\pi, 3\pi, 4\pi, 5\pi, 6\pi\}$. The red circles illustrate the sampling of the frequency in frequency domain for $\ell = 0$. We see, that scale and frequency are well covered by these settings.

convolution with the 0-order Gabor kernel $\mathcal{B}_s^0$. We then use the finite central differences (see section 4.2 on page 103) for realizing the spherical tensor derivative operators. Fig. 5.15 on page 137 illustrates the operations required for the discrete Gabor transform. For a comparison of the computation time of the derivative based transformation with the SHOG transformation and the purely convolution based transformation see Fig. 5.23 on page 151.

Computation Complexity The complexity is similar to the complexity of the Gauss-Laguerre transform; see section 5.1. Apart from one initial cross-correlation per radial frequency with each complexity $\mathcal{O}(N \log N)$, where N is the number of voxels, only finite differences are required. The finite differences can be computed in $\mathcal{O}(MN)$, where M is the small number of scalar valued basis functions per radial frequency which we consider to be constant.

5.4.2.1. Gabor Descriptors

Similar to the Gauss-Laguerre descriptor, we propose to form Gabor descriptors using the power-spectrum (Eq. 3.27) and the bi-spectrum (Eq. 3.28). In our experiments we also considered cross-spectrum invariants by combining expansion coefficients associated with different radial frequencies k. For the details regarding the parameters we refer to the experiment section.

5.4.3. A SHOG Transform for Object Detection in 3D

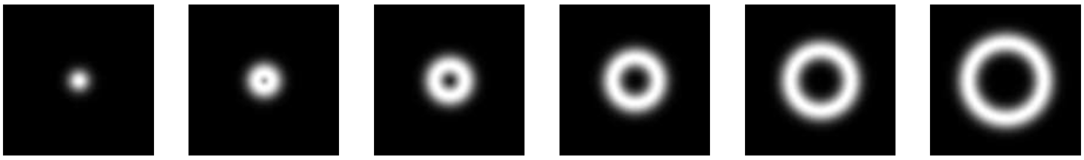
The SHOG transformation allows for mapping images to non-linear feature images in an $SE(3)$ covariant manner. The single values of the feature images are representing the local gradient orientation occurrence around image points. Compared to the previous two transformations, the nonlinearity comes with additional computational costs. However, gradient histogram based features have shown superior performance in a broad variety of applications (particularly in 2D) that might justify the computational complexity.

For the choice of the window function we propose the usage of a Gaussian smoothed sphere. Such a window functions has shown superior performance over a simple Gaussian function in preliminary experiments on real microscopical data (Skibbe et al. 2011b). Instead of using its correct analytical expression (see Eq. (B.42) in the appendix), we use the following approximation

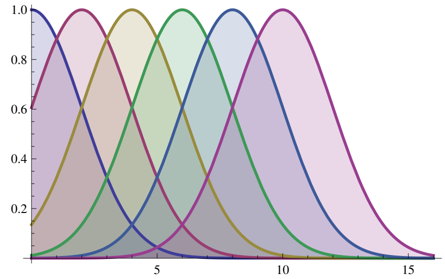
$$w(\mathbf{r}, d, t) := e^{-(d-r)^2/(t2)}\,, \tag{5.79}$$

where $r = \|\mathbf{r}\|$ is the running parameter, $d \in \mathbb{R}$ is the sphere radius and $t \in \mathbb{R}$ steers the smoothing of radial Gaussian profile. A reason for using this approximation is the fact that for the case where the radius d equals 0 the equation above turns into a pure Gaussian window function. In practice we normalize the area under the window function to 1.

We propose the following strategy for selecting an appropriate set of parameters: suppose $d \in \mathbb{R}$ is the radius of the biggest structures or objects that we are looking for. We propose to use several radial window functions with different radii to cover the whole volume within the ball of radius r. So we experimentally determine a suitable



(a) 2D profiles



(b) Radial profiles

Figure 5.25.: The proposed kinds of window functions for SHOG for radius $d = 10$ and devisions $D = 6$.

number $D \in \mathbb{N}_{>1}$ by dividing d equally into D parts. Let D be the number of window functions that should be taken into account. Then we choose $\{(i-1)*d/(D-1)\}_{i=1,\cdots,D}$ for the radii and $t = (d/(D-1))^2$ for the smoothing. That is, we always have an overlap of adjacent Gaussian windowed spheres and the most inner window is a small Gaussian window. For an example see Fig. 5.25.

An upper bound for the spherical harmonic order is found by increasing the order until no improvements can be achieved on a training set (and/or the condition Eq. (5.76) is violated).

The parameter γ weights the influence of the gradient magnitude in a nonlinear manner. For $\gamma = 1$, orientations are weighted by the gradients magnitude. For $\gamma = 2$, the weights are corresponding to the squared magnitude so that SHOG becomes a generalized higher order structure tensor. A value $\gamma < 1$ behaves similar to a gamma correction: high values are damped, small values amplified.

Implementation In our implementation we realize the convolutions by utilizing the Fast Fourier Transform (Frigo and Johnson 2005). For a comparison of the computation time with the derivative based transformation and the purely convolution based intensity transformation see Fig. 5.23 on page 151.

Computation Complexity The computation of the SHOG filter is in $O(MN \log N)$, where N is the number of voxels. This is because the most time consuming parts are the convolutions. We must convolve all single, scalar-valued components of all expansion coefficients up to order L with the window functions, that is, $M = (L+1)^2$ convolutions (for real valued images we can benefit from a sparse tensor representation so that $M = (L+1)(L+2)/2$; please consider the last section of chapter 2 for details). The orientation fields $\|\mathbf{g}\|\mathbf{Y}^{\ell}(\hat{\mathbf{g}})$ are computed independently in a voxel-by-voxel manner by tensor products with an $\mathbf{Y}^1$. Hence this computation step linearly depends on the number of voxels so that the computational complexity is in $O(MN)$.

5.4.3.1. SHOG Descriptors

For creating a SHOG descriptor we use the power-spectrum (Eq. 3.27) and the bispectrum (Eq. 3.28). In our experiments we also considered cross-spectrum invariants by combining expansion coefficients associated window functions of different radii. For the details regarding the parameters we refer to the experiment section.

6 EXPERIMENTS

The proposed spherical harmonic framework has shown superior performance in various recent applications, outperforming state-of-the-art features like 3D SIFT. Such applications are the detection and classification of cells in microscopical images of Arabidopsis root tips (Skibbe et al. 2012), or the detection of landmarks in zebrafish embryos for the purpose of image registration (Liu et al. 2012). An overview of further biomedical applications making use of the proposed spherical harmonic framework is e.g. given in Skibbe and Reisert (2013).

In addition to experiments that are part of several conference and journal publications (two examples are introduced in chapter A on page 181), we conducted three further experiments based on datasets consisting of volumetric images with high intra-class and extra-class variation. We used three different datasets, two of them based on data simulating images strongly distorted by noise and absorption effects. The third dataset consists of microscopical images of allergically relevant pollen grains.

The aim of the experiments is to complete the results of previously conducted experiments by tackling some unanswered issues:

We show that the proposed techniques outperform a dense version of 3D SIFT (Allaire et al. 2008) and 3D SURF (Knopp et al. 2010) in object detection tasks. We also compare with existing spherical harmonic based features including descriptors based on the spherical Gaussian derivatives (Reisert and Burkhardt 2009a) and the standard spherical harmonic transformation (Kazhdan et al. 2003). We further bring

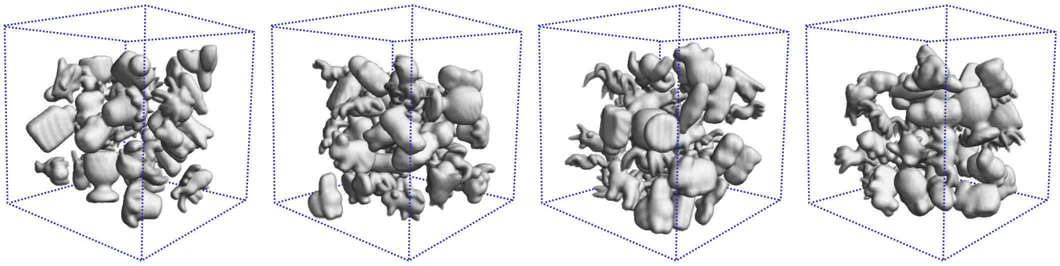


Figure 6.1.: The training set of the PSB dataset (In total: 174 objects). For the experiments we distorted the images by simulating absorption and noise.

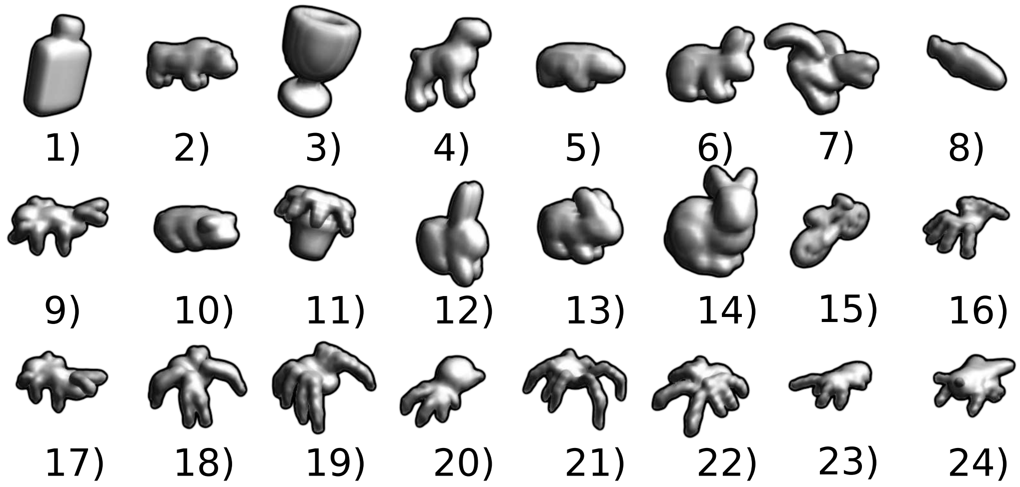


Figure 6.2.: The 24 object classes used in the first experiment on simulated data.

theory into praxis and show that using the proposed coupling rules for tensors significantly reduces the number of products in the bi-spectrum of real valued functions (see section 3.1.1.2 on page 88 for the theoretical background). This leads to a much smaller image descriptor. The experiments show that the performance of the “sparse“ bi-spectrum is comparable to a ”full“ but linearly dependent spectrum.

6.1. Datasets

The advantages of simulated datasets are manifold. The ground truth can be generated along the data generation process. This is not only more convenient than manually labeling the dataset, it also allows for precisely positioning the labels. Furthermore, we can distort the data in a predictable manner. This eases the comparison of the proposed techniques with the existing 3D SIFT and 3D SURF features in varying scenarios. The third database based on microscopical data is used for verifying the outcome on simulated data with experiments on real data.

6.1.1. PSB Dataset

The first of the two simulated datasets consists of volumetric images, each containing about 60 objects out of 24 different object categories; see Fig. 6.2 for an overview of the categories. The objects were generated via rendering of 3D mesh models from the Princeton shape benchmark (PSB) (Shilane et al. 2004) into a discrete voxel grid. In total we have 174 objects in four images for training and 682 objects in 15 images for evaluation. The different categories are equally distributed in each image. Objects are randomly painted into an image with respect to position, rotation and a varying scale (95%-105%). The images are tightly rendered into the target image. However, single parts (like the legs of an insect) do not intersect with neighboring objects. The size of an image was 64^3 voxels. The goal is, to detect the objects with respect to position and class membership.

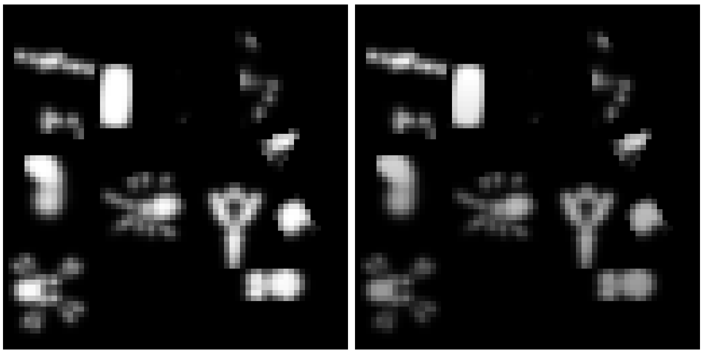
An isosurface rendering of the training set is shown in Fig. 6.1 on page 157. The whole test-set is shown in Fig. E.6 on page 241. A single slice of one of the training images is depicted in Fig. 6.3(a) on the following page. We distorted the images in the following manner: we first simulated absorption which often occurs in biological probes; Fig. 6.3(b). For this, we multiplied the image in one dimension (depth direction, or Z-direction) with a decreasing Gaussian function ($e^{(-x)}$). The function has been scaled so that the uppermost part is not penalized by the effect of absorption and the lowest part is damped by a factor of 0.5. As a result, absolute intensity values do not provide reliable information for discriminating the objects. We then constructed three scenarios by adding mean free Gaussian noise to the images ($f(x) \sim \mathcal{N}(0,\sigma)$). All images are in the range of $[0,1]$ so that we define a signal-to-noise ratio by SNR=$1/\sigma$. The three scenarios are: (1) A bit noisy: a signal to noise ratio (SNR) of 40; Fig. 6.3(c). (2) Noisy, SNR of 20, Fig. 6.3(d). (3) Very noisy, SNR of 6.5, Fig. 6.3(e). Note that the SNR is given with respect to the image without absorption. Hence due to the nonlinear change of illumination in the depth direction, the SNR decreases significantly in a nonlinear way in depth direction, making the detection of the objects even more challenging.

It is further worth mentioning that each image was stored in its undistorted state. Noise and absorption were separately added for each dataset in each experiment. While the effect of absorption was always the same for all experiments, noise was added in a random manner. This slightly influenced the outcome of the experiments. Repeating experiments several times showed a maximum variation of the equal error rate in the range of ±1.5% (in the worst case). This effect should be taken into account when considering the PR (precision recall) curves of the experiments on simulated data.

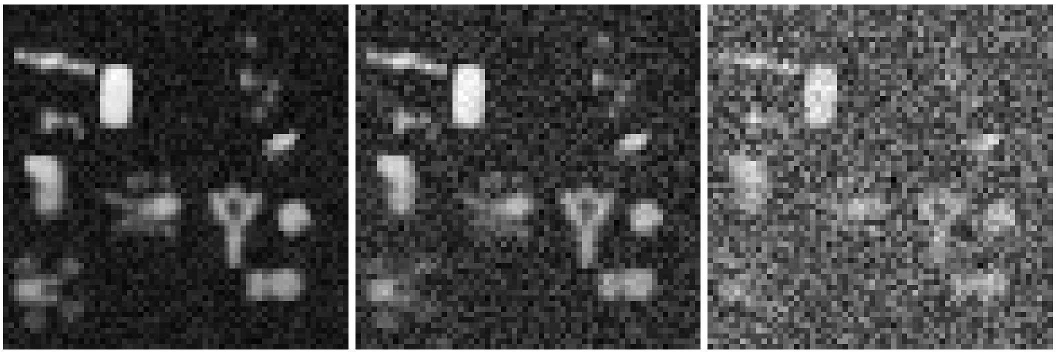
6.1.2. Simulated Pollen Dataset

Palynology is the study and analysis of pollen. Palynology has a broad diversity of applications like paleoclimatology or forensics. In this experiment we aim at detecting pores in pollen grains, small pores on the surface of the grain whose number and shape is a crucial feature for the determination of the species.

This dataset consists of simulated airborne pollen. The goal is to detect and distinguish between specific structures within and on the surface of the pollen. The shape of these structures is inspired by pollen pores of “real” pollen (Ronneberger et al. 2007). The dataset contains 30 training images and 165 test images for evaluation. The image size is 64^3 voxels. The dataset is divided into 10 different types of species. We aimed at detecting and distinguishing 10 different kinds of structures with a high intra-class variation of scale (±15%) and shape. The different species together with the landmarks are depicted in Fig. 6.4 on page 161. An overview of the inner structures of the different species can be found in Figs. E.18 to E.21 on pages 254–257. Note that landmark 9 is a mirrored version of landmark 10. Hence power-spectrum features are not capable to distinguish between them. The whole test-dataset is depicted on pages 262–269. The total number of landmarks for training and testing are listed in the table below.



(a) original (b) +absorption



(c) +absorption +noise SNR 40 (d) +absorption +noise SNR 20 (e) +absorption +noise SNR 6.5

Figure 6.3.: A slice of one of the datasets of PSB experiment with different levels of distortion.

Landmark	1	2	3	4	5	6	7	8	9	10	total
Training	48	48	96	24	24	24	192	96	12	6	**570**
Evaluation	192	192	384	96	96	96	768	384	35	37	**2280**

Similar to the PSB dataset, we conducted experiments with respect to different levels of distortion. We simulated absorption (details in the description of PSB dataset) and added Gaussian noise to the images leading to two scenarios: (1) SNR of 40 and (2) SNR of 6.5.

6.1.3. Real Pollen Dataset

The third dataset is consisting of microscopic images of pollen grains, acquired with a confocal laser scanning microscope (Ronneberger 2007; Ronneberger et al. 2007). It includes five different pollen species. The aim is to detect manually labeled pollen pores near the pollen surface. In contrast to existing publications on this task (Reisert and Burkhardt 2009a; Skibbe et al. 2011b; Skibbe et al. April, 2011), we put all species into one dataset and complemented them with additional pollen types. Hence the different kinds of landmarks must be detected in a mutually exclusive manner further complicating the task. The challenge is to detect and discriminate between all five different pore types. The landmarks are depicted in Fig. 6.6. This

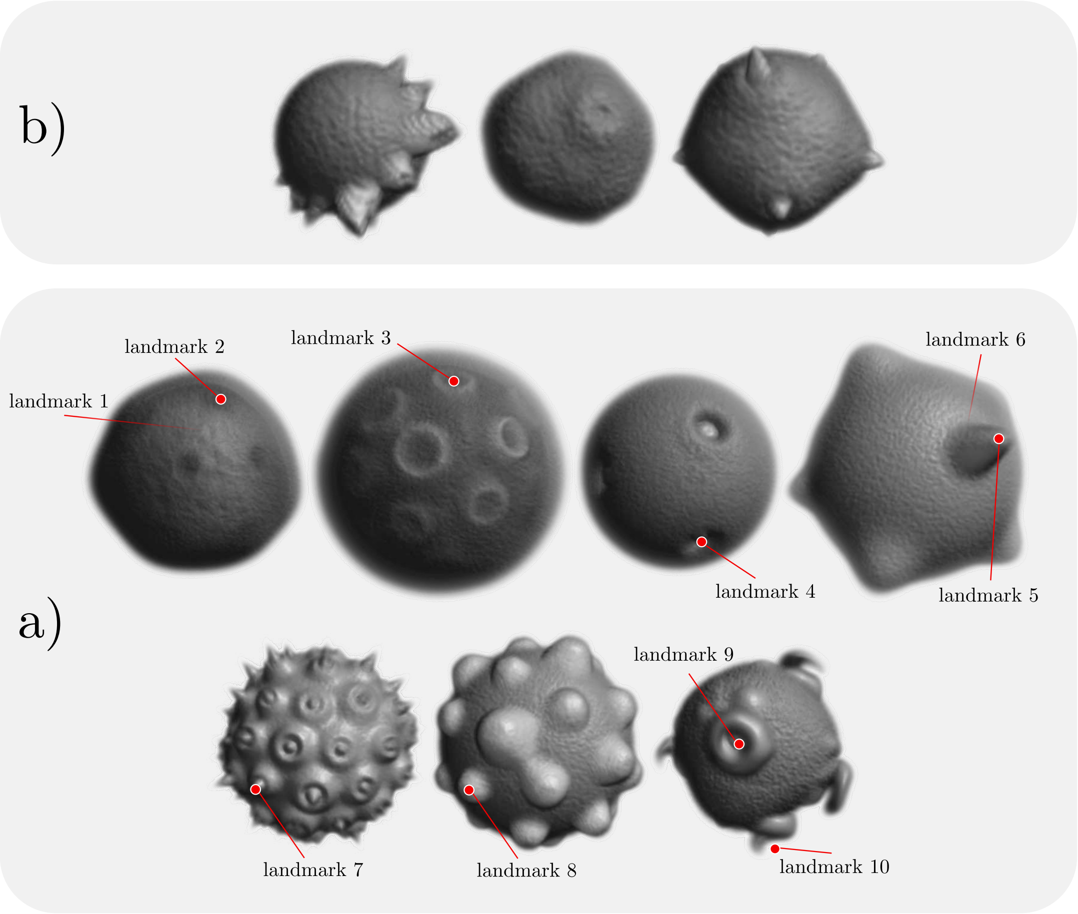


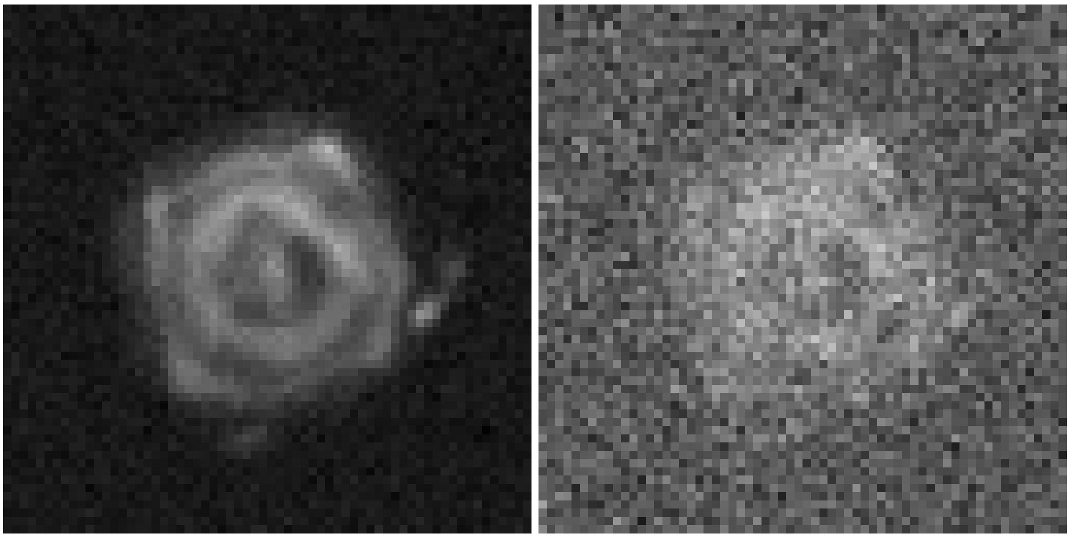
Figure 6.4.: a) The different pollen species together with the landmarks. b) We further added some other types of images with similar structures to make the task more challenging.

datasets contains 20 images with 65 pores for training, and 51 images with 163 pores for evaluation. The diameter of a voxel corresponds to $0.4\mu m$. The smallest image contains $72 \times 72 \times 64$ voxels, the largest image $80 \times 80 \times 90$ voxels. The total number of landmarks for training and testing is listed in the table below.

Landmark	1	2	3	4	5	total
Training	17	12	12	12	12	**65**
Evaluation	45	33	31	33	21	**163**

6.1.4. Experiment Setup

Since we are rather interested in the performance of the features than evaluating different types of classifiers, we used a linear classifier for the first two datasets. It is worth mentioning that a more sophisticated classifier, like a support vector machine or a random forest (Breiman 2001), significantly improves the results. However, repeating the experiments with a random forest classifier did not change the relative


(a) SNR 40

(b) SNR 6.5

Figure 6.5.: A slice of one of the simulated pollen datasets after distortion via absorption and noise (The two noise levels used in the experiments).

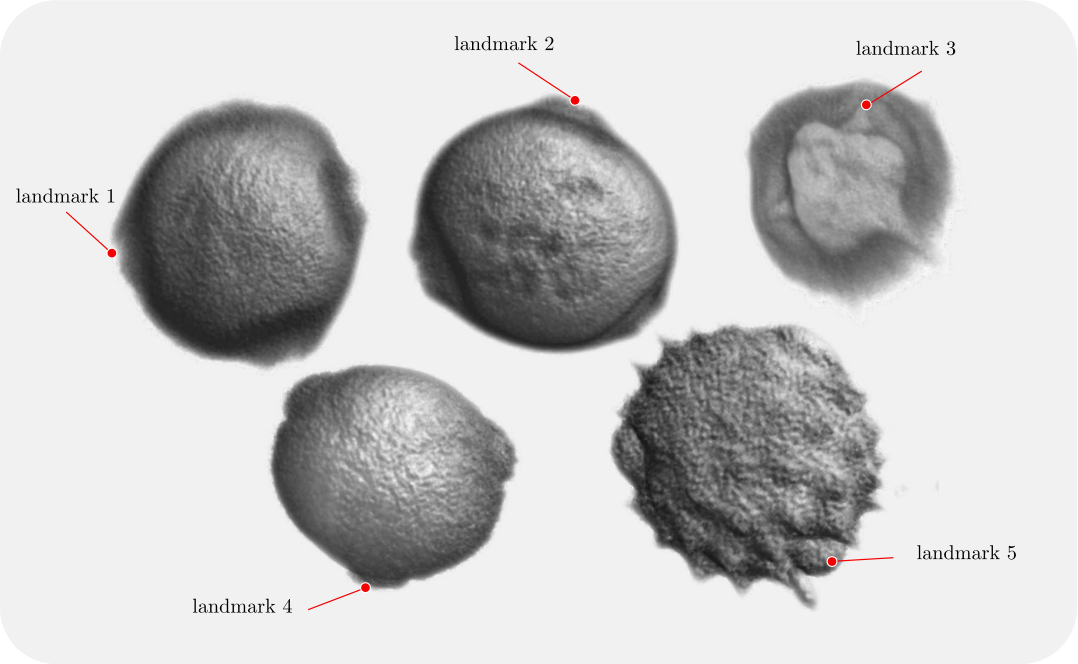


Figure 6.6.: Types of allergically relevant pollen grains used in the experiment: Alder pollen (landmark 1), Birch pollen (landmark 2), Murgwort pollen (landmark 3), Composite pollen (landmark 4) and Buckeye pollen (landmark 5).

ordering of the different descriptor types regarding the detection performance (we will show results from such an experiment, too). For the real pollen dataset we only show the results for the random forest classifier.

In our experiments we considered two groups of features: (1) several kinds of spherical harmonic based power-spectrum and bi-spectrum descriptors. This includes features based on our new proposed transformations as well as features computed via the standard spherical harmonic transformation. (2) For comparison we used a densely computed version of 3D SIFT and 3D SURF. There are mainly three hypothesis that should be supported by the experiment:

- Since the bi-spectrum preserves phase information, bi-spectrum descriptors should perform better than power-spectrum based descriptors.
- Removing the linearly dependent two third of all possible third order invariants is not negatively effecting the classification and detection performance.
- The spherical harmonic based features perform at least as good as, or even better, than state-of-the-art 3D SIFT and 3D SURF features.
- The Gaussian-Laguerre, Gabor and SHOG based invariants perform equivalent to, or better, than invariants based on the existing spherical harmonic transformation. This would be especially welcome since it provides computationally much more efficient alternatives.

When searching biomedical landmarks or small objects, scale invariance is seldom required. However, robustness to small variations in size is important. For the spherical harmonic based approaches as well as for the two reference approaches we determined one fixed scale (per approach and experiment) that works best on the training set using one half of the training images for training and the remaining training images for evaluating the settings (or 3 to 1, respectively). This was the procedure for determining all free parameters. Since the biggest occurring size of landmarks was similar for all three datasets, it turned out that the best performing scales were very similar for all datasets. Further descriptor parameters performing best on the training set slightly varied for the different datasets. But for most of the descriptors we found and used a setting that worked best for all three datasets. This facilitated the comparison of the results (with the cost that a descriptor dimension was slightly larger than necessary for the real pollen dataset).

The details regarding the descriptors we used for the experiments are listed below:

3D SIFT and 3D SURF. Apart from scale, we used the parameters proposed in Flitton et al. (2010) and Knopp et al. (2010) for SIFT and SURF, respectively. That is, a concatenation of $4 \times 4 \times 4$ gradient orientation histograms. Usually both techniques rely on a saliency detector for stable keypoint detection, pose normalization and scale selection. A pre-selection of saliency points is indispensable for achieving a reasonable computation complexity. However, to make the descriptors comparable to the dense spherical harmonic based features, we computed SIFT and SURF densely with one fixed scale, too. This ensures that no object is left out during the classification. Furthermore, it ensures that we evaluate the features and not

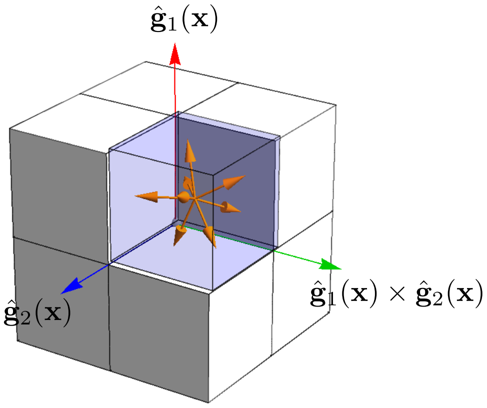


Figure 6.7.: Illustration of 3D SIFT, consisting of eight gradient histograms, each representing eight directions. A local coordinate system around an image point **c** is used for pose normalization. Therefore, the gradient main direction $\hat{\mathbf{g}}_1$ and a second, orthogonal dominant gradient direction $\hat{\mathbf{g}}_2$ are locally determined in a voxel-by-vexel manner.

the saliency detectors. While SIFT is a concatenation of gradient orientation histograms like HOG and SHOG, SURF is rather an approximation of such a distribution. We expected the letter one to be less sensitive to noise than SIFT with the cost of minor loss of discriminability. The resulting feature dimension for SIFT was 512 ($4 \times 4 \times 4$ histograms $\times 8$ gradient directions) and 384 for SURF, respectively ($4 \times 4 \times 4$ histograms bins $\times 6$ signed Cartesian coordinate bins). A histogram arrangement of 3D SIFT is exemplarily illustrated in Fig. 6.7. Before computing the gradients, we initially smoothed the image with a small Gaussian of width 1.5. For the PSB dataset we obtained the best results on the training set when using a radius of 7 voxel for the descriptor, but a smaller radius of 3.5 voxel for determining the dominant gradient directions. For the two pollen datasets we used 8 voxel and 4 voxel (direction histogram) radii, respectively. The descriptors were normalized with respect to the L_2-norm.

SH-Inv Descriptor. The spherical harmonic descriptor has been proposed by Kazhdan et al. (2003): a volumetric image is represented in terms of spherical harmonic functions via projection of the image's intensity values onto a (theoretically) infinite number of nested spheres. The spherical fractions of the image can then be represented in terms of spherical harmonics. We used the training set to determine the number of spheres which performs best on our data. In our case, six different radii $\{0, 1.6, 3.2, 4.8, 6.4, 8.0\}$. We regarded the discrete voxel grid by using a Gaussian function of width 1.6 to smooth the radial profiles of the spheres (the most inner profile is simply a Gaussian function). This ensures a small overlap of neighboring spheres increasing the descriptor's robustness against small deformations. We determined an upper bandwidth limit of $\ell \leq 5$ for the spherical harmonic expansion. Note that a 0 order coefficient represents the signal's mean. Hence we achieve invariants against additive illumination changes by neglecting this coefficient for building the descriptor. On the other hand, the mean is often an additional important feature. For our experiments we considered the mean value for forming the features in all three experiments, performing significantly better than when not including the mean value into the descriptor. The expansion coefficients were normalized using the L_2-norm with respect to the whole expansion to achieve invariance against local, multiplicative illumination changes. The power spectrum was used to form rotation invariant features (denoted by the postfix o2). We further computed cross-spectrum invariants, additionally combining coefficients of different radii (denoted by the postfix o2c). In addition to the second order spectra, we computed (even) bi-spectrum invariants. We computed them twice: once a set of linearly independent features using the proposed coupling rules (denoted by the postfix o3(`coupling rules`)) and a second time computing all possible features (denoted by the postfix o3(`all`)). We additionally computed bi-spectrum invariants including odd products, essential for solving reflection symmetries (denoted by the postfix o3(`odd, coupling rules`)). We took the square-root from the components of the power-spectrum based descriptors and the signed third root from the bi-spectrum based descriptors ($\mathrm{sign}(f)|f|^{1/3}$).

SHOG-Inv Descriptor. The SHOG invariants are based on local SHOG expansion coefficients. We experimentally determined an upper bandwidth limit of $\ell \leq 5$ for the spherical harmonic expansion. As window functions we used the same spherical Gaussian profile as for the spherical harmonic descriptors SH-Inv. That is, six different radii $d = \{0, 1.6, 3.2, 4.8, 6.4, 8.0\}$ with a smoothing of the radial profile with $t = 1.6$; for details see section 5.4.3 on page 154. We initially smoothed the image with a small Gaussian of width 1.5. The spherical harmonic representations of SHOG were normalized via L_2-norm. The gradients have been weighted with $\gamma = 1$. Rotation invariant feature representations where directly derived from the SHOG expansion coefficients via the power spectrum (denoted by the postfix o2). We further computed cross-spectrum invariants, additionally combining coefficients of different radii (denoted by the postfix o2c). Similar to SH, we computed three types of bi-spectrum features: a set of linearly independent even products using the proposed coupling rules (denoted by the postfix o3(`coupling rules`)) and a second time computing all possible features (denoted by the postfix o3(`all`)). Odd products are denoted by o3(`odd, coupling rules`). We took the square-root from the components of the power-spectrum based descriptors and the signed third root ($\mathrm{sign}(f)|f|^{1/3}$) from the bi-spectrum based descriptors.

SHOG Region-Descriptors. SHOG expansion coefficients are representing the gradient orientation distribution within a window function. The window function must be isotropic, otherwise the tensor representations would not be rotation covariant anymore. One drawback of directly computing descriptors from the SHOG expansion coefficients, like for the SHOG-Inv descriptors, is the fact that from the spatial distribution of gradients within a window only the distance to the center is represented by the descriptor (if we use smoothed spheres as window functions). This is a wanted feature for small regions, because it makes the features robust against small deformations. However, for larger objects the spatial distribution of gradients is also an important feature. This drawback hinders a fair comparison with the Gauss-Laguerre, Gabor, or even SIFT descriptors because they do all not suffer in that way.

Due to this fact we considered an additional kind of descriptor in our experiments: we introduce a second smoothing step that aggregates all SHOG features in an image point‘s neighborhood; see Skibbe et al. (2011b). This is one way for mimicking the concatenation of several histograms like in SIFT or HOG via tensor operations; see Fig. 5.17 on page 140 for an illustration of how SIFT is composed upon several histograms of oriented gradients. The new SHOG-region descriptor is inspired by the harmonic filter framework (Reisert and Burkhardt 2009a). The theoretical details of the proposed descriptor go beyond the scope of this theses, however, the interested reader may consult the brief introduction for the filters on page 225. An alternative for the filter approach is an expansion of the SHOG coefficient image via tensorial harmonics before forming invariants (Liu et al. 2011b). We included the SHOG-Region descriptors in order to show that SHOG based descriptors can reasonably compete with the intensity based approaches, but with the cost of additional processing steps.

The SHOG-Region descriptor is based on SHOG expansion coefficients combined in a nonlinear manner via second order tensor products. A smoothing step in cooperation with tensor derivatives is aggregating the outcome of the products in an image points neighborhood. The descriptor image is based on the concatenation of all combinations of the following filter $\mathcal{F}^L : L_2(\mathbb{R}^3) \rightarrow L_2(\mathbb{R}^3)$:

$$\mathcal{F}^L\{I, \ell_1, \ell_2, \nu\} := G_\nu * \left(\boldsymbol{\nabla}_L (\mathbf{a}_w^{\ell_1} \circ_L \mathbf{a}_w^{\ell_2}) \right) \quad , \tag{6.1}$$

where the $\mathbf{a}_w^\ell \in \mathcal{T}_\ell$ are the SHOG expansion coefficients according to Eq (5.64). While w is the window function we used for the SHOG-Inv descriptors, the parameter $\nu \in \mathbb{R}$ determines the Gaussian width of a second smoothing step. The two “smoothing” parameters are mutually influencing each other. Hence we searched for the best setting based on both feature related and smoothing related parameters leading to the following settings working best on the training sets of each dataset:

PSB Dataset: $\gamma = 1$, $\ell \leq 5$, $d = \{0, 1.6, 3.2, 4.8, 6.4, 8.0\}$, $t = 1.6$ and $\nu = 2$; Simulated Pollen Dataset: $\gamma = 1$, $\ell \leq 5$, $d = \{0, 2, 4, 6\}$, $t = 2$ and $\nu = 3$; Real Pollen Dataset: $\gamma = 1$, $\ell \leq 5$, $d = \{0, 2, 4\}$, $t = 2$ and $\nu = 4$.

Gabor-Inv Descriptor. The Gabor descriptor is based on power- and bi-spectrum invariants representing local frequency components within an image in a rotation invariant manner. The spectra are derived from the coefficients computed via the proposed spherical Gabor transformation. We accounted for the absorption effect by using a soft normalization proposed by Reisert and Burkhardt (2009a): the expansion coefficients of order $\ell = 1$ where normalized with respect to the local standard deviation according to $\frac{1}{\gamma + s_{\mathrm{dev}}(\mathbf{r})}$ before computing the remaining coefficients via tensor derivatives. The standard deviation s_{dev} was computed within a Gaussian window of the size of the scale that we determined for the Gabor kernel. Note that this step only requires two additional scalar valued convolutions. The parameter $\gamma \in \mathbb{R}$ is a small constant regularization parameter which we determined experimentally ($\gamma = 10^{-4}$). Theoretically, only the “constant” plane-wave with frequency 0 is sensitive to additive illumination changes. However, for the Gabor function all 0 order coefficients are slightly sensitive to additive intensity changes since the radial waves vanish due to the Gaussian window. We tested both the performance with and without considering the 0 order coefficients. Since the performance was better with all coefficients, we also accounted for the 0 order coefficients when forming the descriptors. We also normalized the 0 order coefficients with respect to the local standard deviation (separately to the normalization of higher order coefficients).

We determined the parameters for extracting the Gabor frequency coefficients along the guidelines in section 5.4.2 on page 152. We determined a scale of 8 resulting in a scale pyramid with $\sigma_0 = 8, \sigma_1 = 4, \sigma_2 = 2$, radial frequencies $k \in \{0.5\pi, \pi, 1.5\pi\}$ and an upper angular bandwidth of $\ell \leq 6$.

Similar to the SH-Inv and the SHOG-Inv we computed the power spectrum invariants (o2), cross spectrum invariants considering different radial frequencies (o2c) and the bi-spectrum invariants (postfixes o3(`coupling rules`), o3(`all`) and the odd products o3(`odd, coupling rules`)). We took the square-root from the components of the power-spectrum based descriptors and the signed third root from the bi-spectrum based descriptors ($\mathrm{sign}(f)|f|^{1/3}$).

Laguerre-Inv Descriptor. The Laguerre descriptors are based on the spherical derivatives of the Gaussian. We can regard them as rotation invariant representations of local jets, Gaussian derivatives computed with respect to different scales (see the introduction of section 5.1 on page 110 for related literature). We achieved invariance against multiplicative illumination changes via normalization with respect to the local standard deviation of the first order coefficients (see the description of the Gabor invariants above). Similar to the Gabor coefficients, the 0 order coefficients were separately normalized with respect to the standard deviation. Note that here only the initial feature (the mean) is sensitive to additive illumination changes. All remaining coefficients are based on differentiation and therefore invariant against additive illumination changes. For our experiments the performance was better when making use of all coefficients thus we kept all for building the descriptors.

We determined the parameters for extracting the coefficients following the guideline in section 5.4.1 on page 150. We determined a scale of $\sigma_0 = 6$ leading to a scale pyramid of $\sigma_0 = 6, \sigma_1 = 3, \sigma_2 = 1.5$ and an upper polynomial degree of $(\ell + n) \leq 6$.

We computed the power spectrum invariants (o2), cross spectrum invariants considering different radial polynomials (o2c) and the bi-spectrum invariants (denoted by the postfixes o3(`coupling rules`), o3(`all`) and for the odd products o3(`odd, coupling rules`)). We took the square-root from the components of the power-spectrum based descriptors and the signed third root ($\mathrm{sign}(f)|f|^{1/3}$) from the bi-spectrum based descriptors.

Gauss-Inv Descriptor. The Gauss descriptors are a subset of the Laguerre descriptors. A Gauss descriptor is build upon the Gauss-Laguerre coefficients with a fixed radial polynomial degree of $n = 0$. They have been proposed by Reisert and Burkhardt (2009a). A direct comparison with the Laguerre-Inv descriptors should show the performance gain we achieve when considering additional radial functions for the Laguerre invariant descriptor. The parameters are identical with those for the Laguerre descriptor.

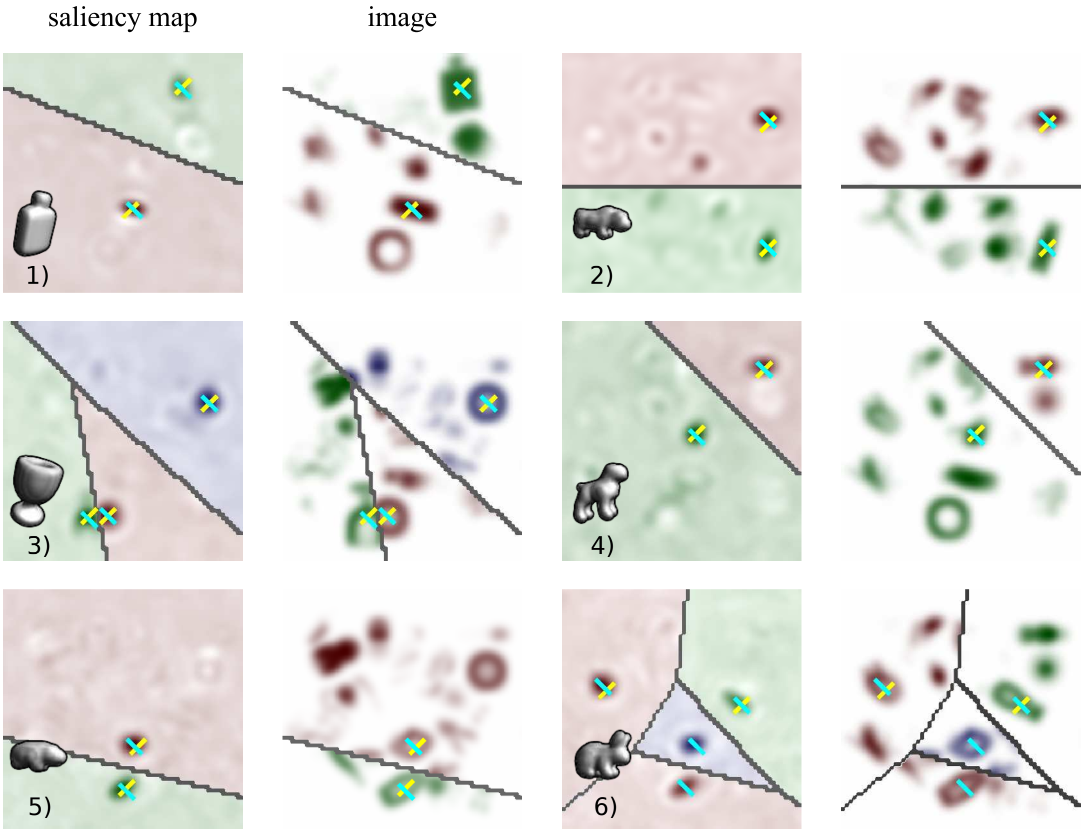


Figure 6.8.: Detection of single classes of the PSB dataset (SNR 40) using the Laguerre descriptor based on bi-spectrum invariants. The slice close to a detection or ground-truth label is shown for both the filter response (saliency map, darker represents higher values for better readability) and the image (without noise and absorption). Yellow lines are indicating the ground truth labels, magenta lines detections (after classification and local maxima detection). In the perfect case they should form a cross. The black lines are separating regions showing slices from different depths. The images show results corresponding to the first image of test set for the object categories 1-6. In this example we have two false positive detections for class 6. The corresponding PR curves can be found on page 244. The filter responses for the remaining categories can be found on pages 247–249

			Equal Error Rate (EER) in %				
Descriptor		Dim	PSB SNR 40	PSB SNR 6.5	P1 SNR 40	P1 SNR 6.5	P2
Laguerre-Inv	o2	48	88	89	57	70	38
	o2c	126	38	64	33	52	42
	o3	150	18	51	20	38	25
Gabor-Inv	o2	49	81	93	57	70	45
	o2c	196	39	65	40	62	33
	o3	210	20	53	30	52	30
SHOG-Inv	o2	36	84	94	96	97	68
	o2c	126	56	80	95	97	68
	o3	120	18	67	64	95	73
SHOG-Region	o2 (PSB)	252	26	78			
	o2 (POLLEN)	168			18	84	31
(Reference Approaches)							
Gauss-Inv	o2	21	94	96	78	77	44
	o2c	42	89	94	70	73	31
	o3	90	36	59	25	45	31
SH-Inv	o2	36	96	97	85	88	61
	o2c	126	86	92	79	87	53
	o3	120	56	71	71	86	45
3D SIFT		512	33	86	43	64	42
3D SURF		384	35	83	76	79	54

Table 6.1.: The EER for all datasets and all approaches. PSB with a SNR of 40 and 6.5, the simulated pollen dataset (P1) with a SNR of 40 and 6.5 and the real pollen dataset (P2). The Upper row shows results based on descriptors formed from expansion coefficients of the proposed techniques. The lower table shows reference approaches we used for comparison. For the results in the table we tolerated a maximum displacement to the ground truth of 3 voxels. The rates in red are the best (lowest) rates for each experiment.

6.2. Results

We evaluated the detection performance of each descriptor on all three datasets. In total, we had six different scenarios: PSB dataset with a SNR of 40, 20 and 6.5, the simulated pollen dataset with an SNR of 40 and 6.5, as well as the real pollen dataset. A short overview showing the most important results (the equal error rates (EER)) is given in table 6.1.

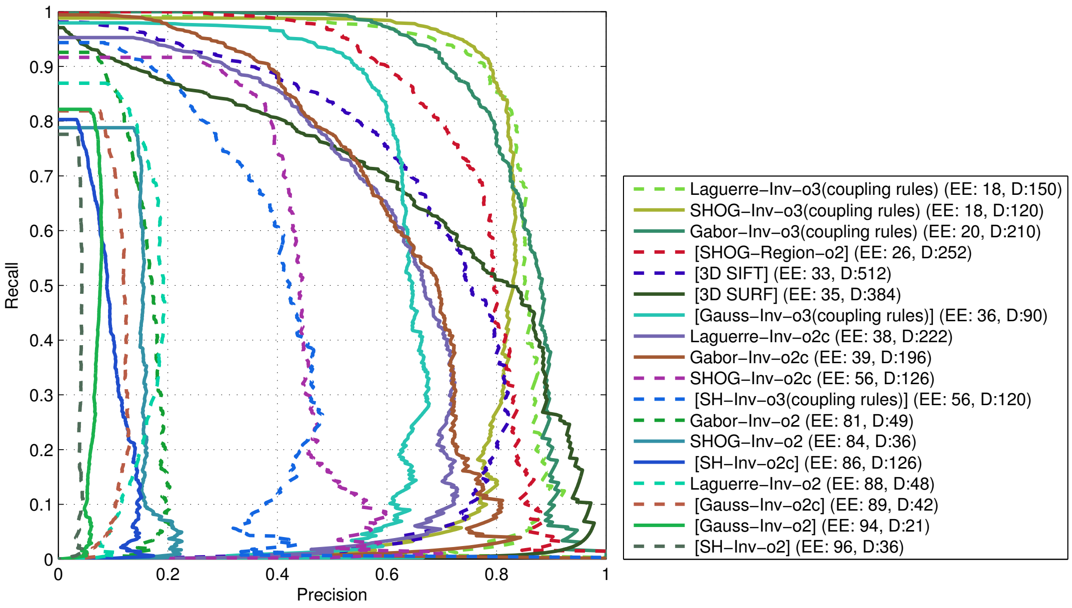
For the linear classifier we smoothed all descriptors (except for the SHOG-Region descriptor, because it applies already final smoothing steps) in a component-by-component manner with a small Gaussian (width of 1.5 voxels). This was particularly important for the SIFT and SURF descriptors. One linear classifier was trained per class by fitting the descriptor images to a label image, where 0 denoted background and 1 a detection. Note that the output was continuous and could directly be used as saliency map. For experiments where we used a random forest classifier we used 500 trees. The number of variables considered for splitting at each node was empirically determined for each dataset using the training datasets. The saliency map for each class was build upon the number of trees voting for a certain class.

For determining a possible detection and building the PR curves we classified each voxel and determined local maxima in the saliency maps. For all experiments we counted an object as detected if a local maxima was in a 3 voxel surrounding of the position of a ground truth label. We also show results for a 6 voxel threshold.

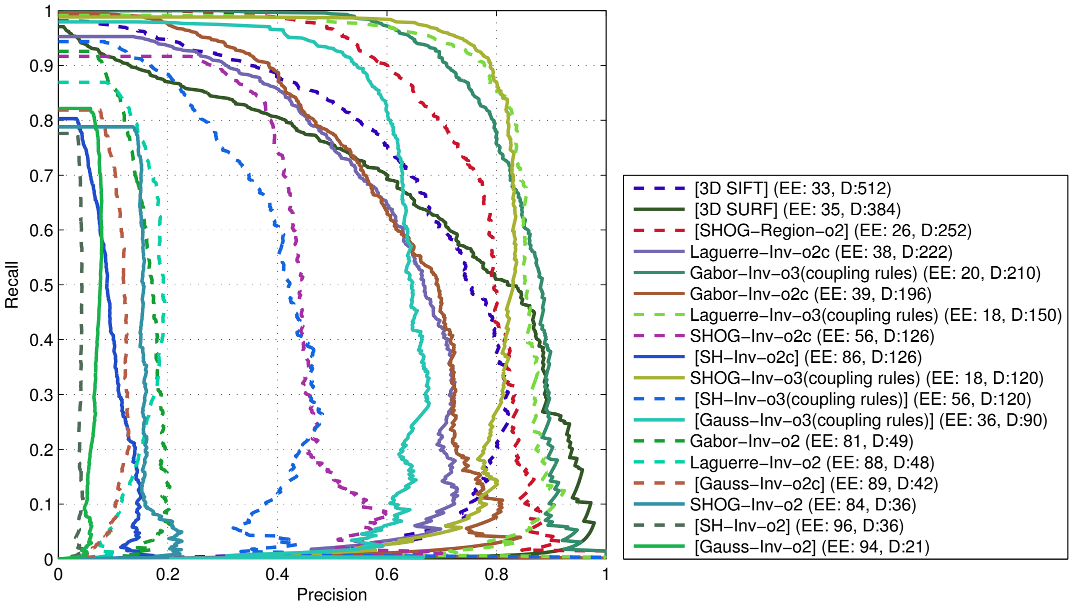
The best performing descriptor was the Laguerre descriptor based on the bi-spectrum invariants. The Laguerre descriptor did not only benefit from the most efficient transformation (among the tested descriptors), it also showed the best performance regarding the difficult scenarios with a very low signal to noise ratio. It also performed significantly better than its Gauss descriptor counterpart in most of the experiments. Only in the real pollen dataset the Gauss descriptor showed a comparable performance. It seems that as soon finer details become important for discriminating the classes, the Laguerre descriptor benefits from a more detailed radial representation. However, if only the coarse shape of an object is important for discrimination, the Laguerre descriptor can benefit only little from higher order radial polynomials and the performance drops toward the performance of the Gauss descriptor. Consequently, if image details are important, we have a big performance gain over the Gauss descriptor because of a richer description in radial direction. In a direct comparison to SIFT and SURF, the Laguerre descriptors based on the power-spectrum performed significantly worse on the PSB dataset. On the simulated and real pollen dataset, the power-spectrum based descriptors performed similarly to SIFT and better than SURF. In all experiments, the bi-spectrum based Laguerre descriptors performed significantly better than both SIFT and SURF, see Fig. E.9 on page 244 for single class plots on the PSB database and Figs. E.25 to 6.11 on pages 261–174 for single results on the simulated pollen datasets. Considering the fact that even the dimension of the bi-spectrum Laguerre descriptor is less half the size as for the SURF descriptor, the Laguerre descriptor significantly outperforms the existing two approaches regarding detection performance, computational complexity and memory consumption.

The worst performance showed the standard spherical harmonic descriptor. We tried several settings such as removing the zero order frequency coefficients from the descriptor, normalizing the final descriptor and/or the coefficients and tried to normalize with respect to the local standard deviation (like for the Laguerre descriptor), but always obtained similar results. For the PSB dataset we tried removing more than half of the classes (mainly those classes which can only be distinguished when considering details, keeping classes 1,2,3,4,6,11,14,19) and the performance was much better (EER 31% with a SNR 20, bi-spectrum). But such a scenario is not challenging enough for revealing the differences of the approaches. Therefore we didn't consider it for further comparison. We observed that the SH-Inv descriptor seems to be less suited to represent finer details and it showed, compared to the other approaches, a higher sensitivity to noise and scale variations.

The Gabor descriptors performed slightly worse than the Laguerre descriptor. We



(a) ordering: EER



(b) ordering: Feature dimension

Figure 6.9.: PR curves for the PSB dataset with a SNR 40

were surprised, because this was in contrast to previous results on real datasets in a cell detection and classification task (Skibbe et al. 2012). A closer look on the detailed results of the PSB dataset revealed that in contrast to the Laguerre descriptor, the Gabor descriptor was confused by the object classes (16-22); see the PR curves in Figs. E.15 to E.17 on pages 250–252 for details.

For the gradient histogram based descriptors including 3D SIFT, 3D SURF and the proposed SHOG-Inv descriptor the performance was quite diverse for the different datasets and scenarios. While the performance was good on the PSB dataset with a SNR of 40, the performance for all three approaches dropped significantly with the increasing noise level (e.g. 3D SIFT, EER 33% → 86%). As we expected, SIFT suffered slightly more from noise than SURF, but in general, SIFT showed a better performance than SURF. A coarse binning compensates effects caused by noise up to a certain extent, but it is not surprising that the gradient orientation based approaches are much more sensitive to noise than the intensity based approaches representing the image patches in a global manner. While SHOG-Inv features performed superior over SIFT and SURF in the PSB database, the performance was poor in the two pollen experiments. SIFT and SURF can clearly benefit from the fact that they divide the surroundings of an image point into different regions: several disjunct gradient histograms are covering different image regions. Hence there is a clear relation between 3D gradient positions and gradient orientations. SHOG-Inv descriptors can only sense the gradient orientation distribution with respect to a certain distance to an image point. The angular distribution of the gradient positions is not represented. Hence only very local structures are covered in detail. The larger the distance to the object center, the weaker its ability of discrimination. The results for the SHOG-Region descriptor confirmed this assumption. Mimicking a SIFT-like concatenation of histograms lead to a significant performance boost so that the SHOG-Region descriptor even outperformed, or performed similar to the Laguerre descriptor in the low noise scenarios.

Regarding the SHOG gradient weighting factor γ and the feature normalization we tried different settings. For the parameter γ and the normalization we compared the performance of the following combinations: $\gamma = 1$ for HOG-like, $\gamma = 2$ for generalized structure tensor-like and $\gamma = 0.5$, in combination with no normalization, L_2 normalization and similar to the Laguerre descriptor and Gabor descriptor, using a normalization of the gradients with respect to the local standard deviation. The results on the PSB are depicted in Figs. E.7 to E.8 on pages 242–243. Since simulated absorption demands a normalization, it is not surprising that normalizing the descriptor led to significantly better results. Both normalization techniques led to comparable results. Considering all three scenarios with increasing noise levels, then $\gamma = 1$ seems to be a good choice. When choosing $\gamma = 2$ (structure tensor-like), the performance was slightly worse than for the other two settings.

The detailed results for the PSB dataset using a linear classifier can be found in the precision recall curves (PR curves) in Fig. 6.9 on the previous page and Figs. E.1 to E.2 on pages 236–237. In Fig. 6.8 on page 168 and Figs. E.12 to E.13 on pages 247–248 we exemplarily show a filter response of a Laguerre based filter. In Figs E.4 and E.5 we compare the detection accuracy for a tolerated 3 voxel displacement to a 6 voxel displacement. Results obtained via a random forest classifier can be found in the PR curve shown in Fig. E.3. The results for the simulated pollen dataset in the curves in Figs. E.22 to E.23 on pages 258–259. Details for the real pollen dataset in Fig. E.34.

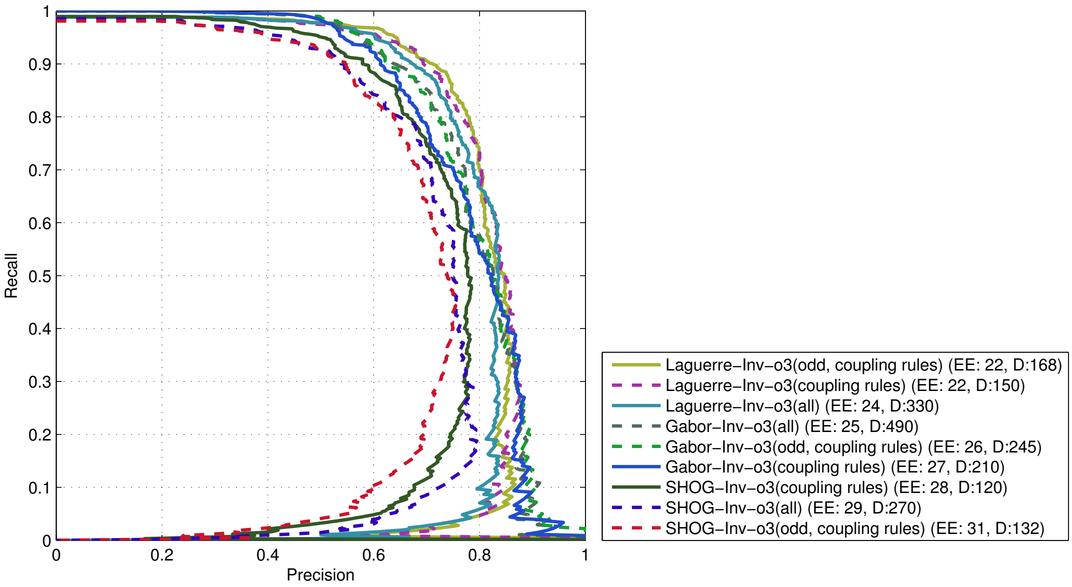
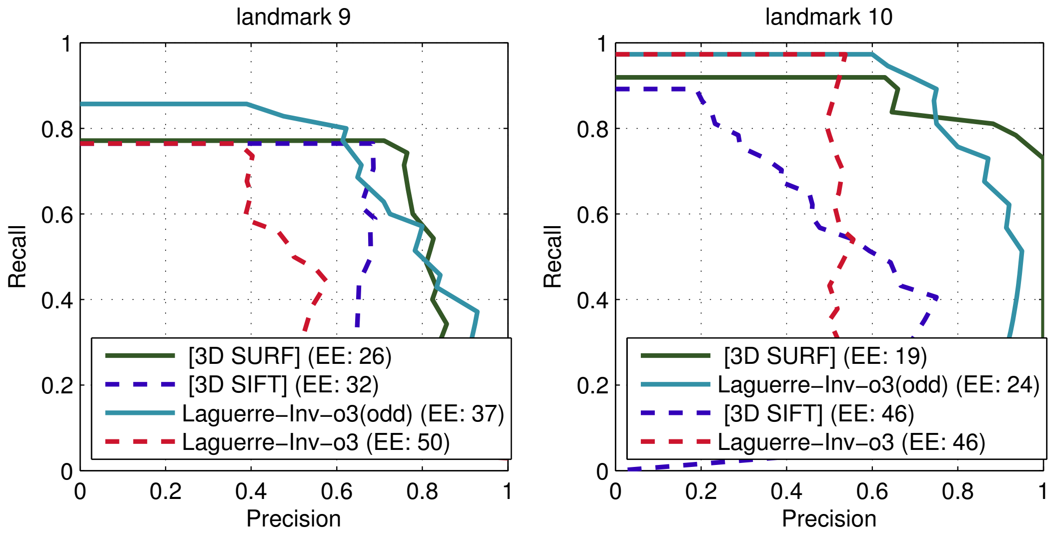


Figure 6.10.: PR curves for the PSB dataset with a SNR of 20. When building all 3rd order invariants the feature dimension is more than double as large as when building features according to the re-coupling rules. However, both features are covering the same information. Moreover, odd products are indispensable for distinguishing between objects which are identical up to reflections. But if this is not the case, odd products do not significantly contribute to the detection performance.

6.2.1. Power-Spectrum versus Bi-Spectrum

There are various possible ways to combine spherical harmonic expansion coefficients to form invariants. We focused on three kinds of invariants: the power-spectrum, the cross-spectrum and the bi-spectrum. This list of invariants can easily be extended. For instance, by combining different radial profiles in a bi-spectrum we obtain some kind of bi-cross-spectrum. Preliminary results on such Gabor features have shown a big performance boost with the cost of a noticeable increase of the descriptor dimension.

In all experiments bi-spectrum based descriptors performed better than the power-spectrum based descriptors. However, the performance gain comes with the cost of a three times higher descriptor dimension (considering the even products only). When including the odd products, the dimension becomes even higher. Moreover, the choice between the power-spectrum and the bi-spectrum is not always a pure tradeoff between memory consumption and detection performance. In many tasks the performance of the power-spectra based descriptors can become close to the performance of their bi-spectrum counterparts; compare Gauss-Inv-o2c (EER 31%) with Gauss-Inv-o3 (EER 31%) in the results of the real pollen dataset shown by the PR curves in Fig. E.34 on page 270). Hence if the structure of objects is simple and if the number of classes is manageable, then it is worth trying both second order spectra like the power- or cross-spectrum and bi-spectrum based descriptors.



(a) landmark 9　(b) landmark 10

Figure 6.11.: Evaluating single classes of the simulated pollen dataset (SNR 40). Comparing Laguerre descriptors with SIFT and SURF. Landmark 9 is a reflected version of landmark 10. While the odd bi-spectrum, SIFT and SURF can resolve reflections, the even bi-spectrum confounds the two classes.

6.2.2. Bi-Spectrum & Coupling Rules

We considered three representations of bi-spectra: two containing only even products, where the first contained all possible products (but without trivial identities, see table 6.2 on the facing page and table 6.3 on page 176, where we exemplarily show a list of products up to order 4), the second one only the linearly independent subset (see table 6.2 on the facing page). The third kind is a linearly independent spectrum including the odd products. The odd products are indispensable for distinguishing a mirrored copy of an object from its original counterpart; see section 3.1.1.3 on page 90. However, many of the object categories within the experiments show a reflection symmetry. Since odd products are 0 on a symmetry axis, they do not provide information for these object categories.

Hence it is not surprising that if reflection does not play a role, all spectra perform similarly; see Fig 6.10 on the preceding page. That is, it is never necessary to compute the full spectrum. This saves us about one third of memory and computation time. Moreover, since odd products only contribute in specific scenarios, we can also neglect them in many applications. In our experiments, the simulated pollen database contains one pair of landmarks which are reflected versions of each other. Here the odd products contributed decisively: landmark (9) EER 50% → 37% and landmark (10) EER 46% → 24%; see Fig. 6.11. A further example is given in section A.2 on page 184: odd products are indispensable for distinguishing between the left and right hemisphere of a human brain.

odd	=**0**	(all)	(coupling rules)
		$((\mathbf{a}^0 \circ_0 \mathbf{a}^0) \circ_0 \mathbf{a}^0)$	$((\mathbf{a}^0 \circ_0 \mathbf{a}^0) \circ_0 \mathbf{a}^0)$
		$((\mathbf{a}^0 \circ_1 \mathbf{a}^1) \circ_0 \mathbf{a}^1)$	
		$((\mathbf{a}^0 \circ_2 \mathbf{a}^2) \circ_0 \mathbf{a}^2)$	
		$((\mathbf{a}^0 \circ_3 \mathbf{a}^3) \circ_0 \mathbf{a}^3)$	
		$((\mathbf{a}^0 \circ_4 \mathbf{a}^4) \circ_0 \mathbf{a}^4)$	
		$((\mathbf{a}^1 \circ_1 \mathbf{a}^0) \circ_0 \mathbf{a}^1)$	
		$((\mathbf{a}^1 \circ_0 \mathbf{a}^1) \circ_0 \mathbf{a}^0)$	$((\mathbf{a}^1 \circ_0 \mathbf{a}^1) \circ_0 \mathbf{a}^0)$
✓	✓	$((\mathbf{a}^1 \circ_1 \mathbf{a}^1) \circ_0 \mathbf{a}^1)$	
		$((\mathbf{a}^1 \circ_2 \mathbf{a}^1) \circ_0 \mathbf{a}^2)$	
		$((\mathbf{a}^1 \circ_1 \mathbf{a}^2) \circ_0 \mathbf{a}^1)$	$((\mathbf{a}^1 \circ_1 \mathbf{a}^2) \circ_0 \mathbf{a}^1)$
✓	✓	$((\mathbf{a}^1 \circ_2 \mathbf{a}^2) \circ_0 \mathbf{a}^2)$	
		$((\mathbf{a}^1 \circ_3 \mathbf{a}^2) \circ_0 \mathbf{a}^3)$	
		$((\mathbf{a}^1 \circ_2 \mathbf{a}^3) \circ_0 \mathbf{a}^2)$	
✓	✓	$((\mathbf{a}^1 \circ_3 \mathbf{a}^3) \circ_0 \mathbf{a}^3)$	
		$((\mathbf{a}^1 \circ_4 \mathbf{a}^3) \circ_0 \mathbf{a}^4)$	
		$((\mathbf{a}^1 \circ_3 \mathbf{a}^4) \circ_0 \mathbf{a}^3)$	
✓	✓	$((\mathbf{a}^1 \circ_4 \mathbf{a}^4) \circ_0 \mathbf{a}^4)$	
		$((\mathbf{a}^2 \circ_2 \mathbf{a}^0) \circ_0 \mathbf{a}^2)$	
		$((\mathbf{a}^2 \circ_1 \mathbf{a}^1) \circ_0 \mathbf{a}^1)$	
✓	✓	$((\mathbf{a}^2 \circ_2 \mathbf{a}^1) \circ_0 \mathbf{a}^2)$	
		$((\mathbf{a}^2 \circ_3 \mathbf{a}^1) \circ_0 \mathbf{a}^3)$	
		$((\mathbf{a}^2 \circ_0 \mathbf{a}^2) \circ_0 \mathbf{a}^0)$	$((\mathbf{a}^2 \circ_0 \mathbf{a}^2) \circ_0 \mathbf{a}^0)$
✓	✓	$((\mathbf{a}^2 \circ_1 \mathbf{a}^2) \circ_0 \mathbf{a}^1)$	
		$((\mathbf{a}^2 \circ_2 \mathbf{a}^2) \circ_0 \mathbf{a}^2)$	$((\mathbf{a}^2 \circ_2 \mathbf{a}^2) \circ_0 \mathbf{a}^2)$
✓	✓	$((\mathbf{a}^2 \circ_3 \mathbf{a}^2) \circ_0 \mathbf{a}^3)$	
		$((\mathbf{a}^2 \circ_4 \mathbf{a}^2) \circ_0 \mathbf{a}^4)$	
		$((\mathbf{a}^2 \circ_1 \mathbf{a}^3) \circ_0 \mathbf{a}^1)$	
✓	✓	$((\mathbf{a}^2 \circ_2 \mathbf{a}^3) \circ_0 \mathbf{a}^2)$	
		$((\mathbf{a}^2 \circ_3 \mathbf{a}^3) \circ_0 \mathbf{a}^3)$	
✓		$((\mathbf{a}^2 \circ_4 \mathbf{a}^3) \circ_0 \mathbf{a}^4)$	
		$((\mathbf{a}^2 \circ_2 \mathbf{a}^4) \circ_0 \mathbf{a}^2)$	$((\mathbf{a}^2 \circ_2 \mathbf{a}^4) \circ_0 \mathbf{a}^2)$
✓		$((\mathbf{a}^2 \circ_3 \mathbf{a}^4) \circ_0 \mathbf{a}^3)$	
		$((\mathbf{a}^2 \circ_4 \mathbf{a}^4) \circ_0 \mathbf{a}^4)$	
		$((\mathbf{a}^3 \circ_3 \mathbf{a}^0) \circ_0 \mathbf{a}^3)$	
		$((\mathbf{a}^3 \circ_2 \mathbf{a}^1) \circ_0 \mathbf{a}^2)$	
✓	✓	$((\mathbf{a}^3 \circ_3 \mathbf{a}^1) \circ_0 \mathbf{a}^3)$	
		$((\mathbf{a}^3 \circ_4 \mathbf{a}^1) \circ_0 \mathbf{a}^4)$	
		$((\mathbf{a}^3 \circ_1 \mathbf{a}^2) \circ_0 \mathbf{a}^1)$	$((\mathbf{a}^3 \circ_1 \mathbf{a}^2) \circ_0 \mathbf{a}^1)$
✓	✓	$((\mathbf{a}^3 \circ_2 \mathbf{a}^2) \circ_0 \mathbf{a}^2)$	
		$((\mathbf{a}^3 \circ_3 \mathbf{a}^2) \circ_0 \mathbf{a}^3)$	
✓		$((\mathbf{a}^3 \circ_4 \mathbf{a}^2) \circ_0 \mathbf{a}^4)$	
		$((\mathbf{a}^3 \circ_0 \mathbf{a}^3) \circ_0 \mathbf{a}^0)$	$((\mathbf{a}^3 \circ_0 \mathbf{a}^3) \circ_0 \mathbf{a}^0)$
✓	✓	$((\mathbf{a}^3 \circ_1 \mathbf{a}^3) \circ_0 \mathbf{a}^1)$	
		$((\mathbf{a}^3 \circ_2 \mathbf{a}^3) \circ_0 \mathbf{a}^2)$	$((\mathbf{a}^3 \circ_2 \mathbf{a}^3) \circ_0 \mathbf{a}^2)$
✓	✓	$((\mathbf{a}^3 \circ_3 \mathbf{a}^3) \circ_0 \mathbf{a}^3)$	
		$((\mathbf{a}^3 \circ_4 \mathbf{a}^3) \circ_0 \mathbf{a}^4)$	
		$((\mathbf{a}^3 \circ_1 \mathbf{a}^4) \circ_0 \mathbf{a}^1)$	
✓		$((\mathbf{a}^3 \circ_2 \mathbf{a}^4) \circ_0 \mathbf{a}^2)$	
		$((\mathbf{a}^3 \circ_3 \mathbf{a}^4) \circ_0 \mathbf{a}^3)$	$((\mathbf{a}^3 \circ_3 \mathbf{a}^4) \circ_0 \mathbf{a}^3)$
✓	✓	$((\mathbf{a}^3 \circ_4 \mathbf{a}^4) \circ_0 \mathbf{a}^4)$	
		$((\mathbf{a}^4 \circ_4 \mathbf{a}^0) \circ_0 \mathbf{a}^4)$	
		$((\mathbf{a}^4 \circ_3 \mathbf{a}^1) \circ_0 \mathbf{a}^3)$	
✓	✓	$((\mathbf{a}^4 \circ_4 \mathbf{a}^1) \circ_0 \mathbf{a}^4)$	
		$((\mathbf{a}^4 \circ_2 \mathbf{a}^2) \circ_0 \mathbf{a}^2)$	
✓		$((\mathbf{a}^4 \circ_3 \mathbf{a}^2) \circ_0 \mathbf{a}^3)$	
		$((\mathbf{a}^4 \circ_4 \mathbf{a}^2) \circ_0 \mathbf{a}^4)$	
		$((\mathbf{a}^4 \circ_1 \mathbf{a}^3) \circ_0 \mathbf{a}^1)$	$((\mathbf{a}^4 \circ_1 \mathbf{a}^3) \circ_0 \mathbf{a}^1)$
✓		$((\mathbf{a}^4 \circ_2 \mathbf{a}^3) \circ_0 \mathbf{a}^2)$	$((\mathbf{a}^4 \circ_2 \mathbf{a}^3) \circ_0 \mathbf{a}^2)$
		$((\mathbf{a}^4 \circ_3 \mathbf{a}^3) \circ_0 \mathbf{a}^3)$	
✓	✓	$((\mathbf{a}^4 \circ_4 \mathbf{a}^3) \circ_0 \mathbf{a}^4)$	
		$((\mathbf{a}^4 \circ_0 \mathbf{a}^4) \circ_0 \mathbf{a}^0)$	$((\mathbf{a}^4 \circ_0 \mathbf{a}^4) \circ_0 \mathbf{a}^0)$
✓	✓	$((\mathbf{a}^4 \circ_1 \mathbf{a}^4) \circ_0 \mathbf{a}^1)$	
		$((\mathbf{a}^4 \circ_2 \mathbf{a}^4) \circ_0 \mathbf{a}^2)$	$((\mathbf{a}^4 \circ_2 \mathbf{a}^4) \circ_0 \mathbf{a}^2)$
✓	✓	$((\mathbf{a}^4 \circ_3 \mathbf{a}^4) \circ_0 \mathbf{a}^3)$	
		$((\mathbf{a}^4 \circ_4 \mathbf{a}^4) \circ_0 \mathbf{a}^4)$	$((\mathbf{a}^4 \circ_4 \mathbf{a}^4) \circ_0 \mathbf{a}^4)$

Table 6.2.: In this table we depict all possible 65 triple-products for tensors (or tensor fields) up to order 4 together with a linearly independent set of products (coupling rules) spanning the set of all products (all). Assuming that the coefficients are the results of a spherical harmonic transform of a real valued image, the number of linearly independent products is 15, significantly smaller. We obtain the linearly independent set of products by removing the odd zero products, removing trivial identities (just switching the order of tensors in the most inner product) and removing non-trivial identities using the coupling rules for triple products. In the case we don't count the trivial identities and remove the zero tensors we get 30 products out of 65; see table 6.3.

odd	=**0**	(all)
		$((\mathbf{a}^0 \circ_0 \mathbf{a}^0) \circ_0 \mathbf{a}^0)$
		$((\mathbf{a}^0 \circ_1 \mathbf{a}^1) \circ_0 \mathbf{a}^1)$
		$((\mathbf{a}^0 \circ_2 \mathbf{a}^2) \circ_0 \mathbf{a}^2)$
		$((\mathbf{a}^0 \circ_3 \mathbf{a}^3) \circ_0 \mathbf{a}^3)$
		$((\mathbf{a}^0 \circ_4 \mathbf{a}^4) \circ_0 \mathbf{a}^4)$
		$((\mathbf{a}^1 \circ_0 \mathbf{a}^1) \circ_0 \mathbf{a}^0)$
		$((\mathbf{a}^1 \circ_2 \mathbf{a}^1) \circ_0 \mathbf{a}^2)$
		$((\mathbf{a}^1 \circ_1 \mathbf{a}^2) \circ_0 \mathbf{a}^1)$
		$((\mathbf{a}^1 \circ_3 \mathbf{a}^2) \circ_0 \mathbf{a}^3)$
		$((\mathbf{a}^1 \circ_2 \mathbf{a}^3) \circ_0 \mathbf{a}^2)$
		$((\mathbf{a}^1 \circ_4 \mathbf{a}^3) \circ_0 \mathbf{a}^4)$
		$((\mathbf{a}^1 \circ_3 \mathbf{a}^4) \circ_0 \mathbf{a}^3)$
		$((\mathbf{a}^2 \circ_0 \mathbf{a}^2) \circ_0 \mathbf{a}^0)$
		$((\mathbf{a}^2 \circ_2 \mathbf{a}^2) \circ_0 \mathbf{a}^2)$
		$((\mathbf{a}^2 \circ_4 \mathbf{a}^2) \circ_0 \mathbf{a}^4)$
		$((\mathbf{a}^2 \circ_1 \mathbf{a}^3) \circ_0 \mathbf{a}^1)$
		$((\mathbf{a}^2 \circ_3 \mathbf{a}^3) \circ_0 \mathbf{a}^3)$
✓		$((\mathbf{a}^2 \circ_4 \mathbf{a}^3) \circ_0 \mathbf{a}^4)$
		$((\mathbf{a}^2 \circ_2 \mathbf{a}^4) \circ_0 \mathbf{a}^2)$
✓		$((\mathbf{a}^2 \circ_3 \mathbf{a}^4) \circ_0 \mathbf{a}^3)$
		$((\mathbf{a}^2 \circ_4 \mathbf{a}^4) \circ_0 \mathbf{a}^4)$
		$((\mathbf{a}^3 \circ_0 \mathbf{a}^3) \circ_0 \mathbf{a}^0)$
		$((\mathbf{a}^3 \circ_2 \mathbf{a}^3) \circ_0 \mathbf{a}^2)$
		$((\mathbf{a}^3 \circ_4 \mathbf{a}^3) \circ_0 \mathbf{a}^4)$
		$((\mathbf{a}^3 \circ_1 \mathbf{a}^4) \circ_0 \mathbf{a}^1)$
✓		$((\mathbf{a}^3 \circ_2 \mathbf{a}^4) \circ_0 \mathbf{a}^2)$
		$((\mathbf{a}^3 \circ_3 \mathbf{a}^4) \circ_0 \mathbf{a}^3)$
		$((\mathbf{a}^4 \circ_0 \mathbf{a}^4) \circ_0 \mathbf{a}^0)$
		$((\mathbf{a}^4 \circ_2 \mathbf{a}^4) \circ_0 \mathbf{a}^2)$
		$((\mathbf{a}^4 \circ_4 \mathbf{a}^4) \circ_0 \mathbf{a}^4)$

Table 6.3.: Even in the case we omit the trivial identities and remove the zero tensors (so we get 30 products out of 65), the resulting set of linearly independent products is up to three times smaller than this set (depending on the maximum tensor rank). In our experiments we always compared the completely reduced set of products (denoted as coupling-rules) to the type of products shown here (denoted as all).

7 CONCLUSIONS

In this thesis we introduced the theoretical background for building $SE(3)$ covariant filters using spherical tensor algebra. We focused on the rotation invariant detection of objects in volumetric images. We highlighted the benefits of spherical tensors over “classical” Cartesian tensors, being the most natural and sparse representations regarding full 3D rotations. We have shown how to transform volumetric images into spherical tensor representations so that we can utilize spherical tensor algebra for local feature and object detection. Given a spherical tensor representation of images, locally rotation invariant features can be derived via an established group integration technique. We have shown how to make use of this framework for deriving two kinds of widely used invariants: the angular power-spectrum based invariants and the angular bi-spectrum based invariants. We have shown that similar to the symmetry of the Fourier coefficients of real valued cyclic functions, there exists a symmetry interrelation of spherical tensor representation of real valued functions on the sphere. This implies the same symmetry for the spherical tensor coefficients of real valued volumetric images. We further have shown that this fact induces a certain kind of non-trivial symmetries within a function’s angular bi-spectrum. We further have shown that many components of this bi-spectrum are identical to the zero tensor. The consideration of both facts is indispensable for an efficient and memory friendly implementation, because it significantly reduces the amount of memory and computation time by a factor of about three without weaken the ability of discrimination.

We have contributed with several new theoretical insights and new algorithms. We introduced three new types of transformations mapping images to fields which elements are spherical tensor representations. Each transformation extracts intrinsic features from local image patches in a voxel-by-voxel manner: (1) the first kind of transformation extracts features based on the derivatives of a Gaussian smoothed image, a spherical counterpart of a “local jet“ (Koenderink and Doorn 1987). The resulting features are image representations in the 3D Gauss-Laguerre domain which we have shown to be optimal for local, smooth processes. We introduced an efficient implementation for discrete images via finite difference operators. Apart from one initial convolution, the computational complexity becomes linear in the number of voxels leading to the fastest transformation among the proposed techniques. We further introduced an extension that extracts features from higher order tensor fields. (2) The second kind of transformation is a Gabor transformation. The extracted features are representing local frequency components of image patches. We have shown that the underlying basis functions are providing a very accurate selectivity regarding frequency and orientation in frequency domain. Similar to the Gauss-Laguerre transformation, the spherical Gabor transformation benefits from an implementation via finite differences. (3) The third transformation, the SHOG transformation, is a generalization of the structure tensor (see section 5.3.1.3 on page 148). SHOG represents the local gradient orientation distribution of an image patch in a similar manner than HOG. However, in contrast to HOG and SIFT, the computation and representation of the histogram is based on a continuous function and all the computations are realized in Fourier domain. We have shown how to compute the histograms in an efficient manner using spherical tensor algebra.

We complemented the theoretical background regarding these transformations with a framework for forming image descriptors from the resulting spherical tensor representations. We particularly addressed the limitations existing when working on a discrete voxel grid and have shown how to select the parameters to avoid under-sampling artifacts.

We further conducted experiments, particularly addressing a comparison of the proposed techniques to a dense version of 3D SIFT and 3D SURF in several scenarios. The results clearly support the hypotheses that spherical tensor based approaches perform not only comparable to the state-of-the-art of SIFT and SURF, but they outperformed them significantly. We further compared the ability of discrimination of the power-spectrum based descriptors to several types of bi-spectrum based descriptors. In most cases the performance of the bi-spectrum based features was significantly better than for the power-spectrum based descriptors. On the other hand we observed that the power-spectrum based descriptors can show a good performance and thus might be very well suited for object detection tasks where the structure that is important for discrimination is simple. The results have further shown that applying the proposed rules for removing linear dependencies within a bi-spectrum shrinks the dimension of descriptors without any loss of discrimination ability. Furthermore, it turns out that the odd tensor products, which are important for discriminating reflected versions of structures, do not noticeably contribute to the performance when reflection plays not a role and can also be omitted in such tasks.

Mathematical facts allow us to draw the following major conclusions: (1) spherical tensors are the most sparse representations of the 3D rotation and therefore the natural tool for coping with higher order tensor representations. (2) The proposed Gauss-Laguerre transformation is one of the most efficient transformation mapping images to spherical harmonic expansion coefficients in a voxel-by-voxel manner. It is further optimal for representing local smooth processes. In the time of writing, we are not aware of any alternative that performs similarly fast and in a more memory efficient manner. (3) There are linear dependencies within the angular bi-spectrum for real-valued functions that can be removed in an analytical, straight forward manner. Furthermore, we can easily identify non-contributing components within the odd components of the spectrum.

We can further draw the following conclusions from experiments on simulated and real data: (1) For an efficient implementation of the proposed transformations, the usage of simple central difference schemes is sufficient for our applications. For the Laplace operations, which are accelerating the computation of the Gauss-Laguerre expansion coefficients, a new differentiation scheme shows a better numerical stability over a "standard operator" without significantly increasing the computation time. (2) All three proposed transformations can be computed in significantly less time than a purely convolution based approach, where a convolution kernel has to be provided for each single tensor component. (3) We proposed descriptors based on the power-spectrum and the bi-spectrum. They outperform 3D SIFT and 3D SURF as well as existing spherical harmonic based descriptors. In contrast to the

Gauss-Laguerre transform, the SHOG transform seems to be much more sensitive to noise. It further turned out that that for forming a SHOG descriptor it is highly beneficial to use a two step approach where neighboring SHOG expansion coefficients are combined to mimic the concatenation of several histograms like in HOG or SIFT.

It is difficult to give an advice or to draw a rule for choosing one of the proposed transformation techniques. Several "real world" applications have shown diverse results on Gauss-Laguerre based descriptors (including the Gauss descriptor), spherical Gabor descriptors and SHOG based descriptors. From a computational point of view, the Gauss-Laguerre transformation is the best choice. The performance of the Gauss-Laguerre descriptor was remarkable in our experiments. However, it difficult to transfer such a ranking to all possible scenarios. In Liu et al. (2011b); Skibbe et al. (2011b), SHOG based descriptors showed a very good performance.

All three types of transformation are covering different kinds of features: local derivatives for the Gauss-Laguerre transform, frequency selective features for the spherical Gabor transform and representations of local gradient orientation distributions for the SHOG transformation. It is worth trying all three types of transformations when aiming to solve a detection and/or classification problem. Since all three transformations are build upon the same mathematical framework, the remaining computations steps, like the computation of the spectra or the application of filters, are identical for all three candidates and therefore additional implementation effort is not required.

For the feature computation itself the answer is clear. Descriptors based on the bispectrum have a big advantage over second order power-spectrum features. From a theoretical point of view they preserve phase information which is lost in the power spectrum, but also in the experiments they performed best.

This thesis has coved the theoretical and practical background to build $SE(3)$ covariant filters for rotation invariant object detection based on spherical tensor algebra, complementing existing approaches and mathematical concepts with new theoretical insights, algorithms and techniques. We believe that spherical tensor algebra for 3D image analysis will be of similar importance than “standard“ Fourier analysis and due to the increasing importance of volumetric biomedical images in the life-sience, we expect that many applications will make use and benefit from this framework.

A Biomedical Applications

In this chapter we exemplarily introduce two applications were we made use of the proposed spherical tensor techniques. This includes one typical biological scenario, where we aimed at cell detection and cell classification in microscopical biological images and a second scenario where we worked with medical DTI images of the human brain. The tasks on medical images included a voxel-by-voxel classification of the images by different tissue types and anatomical regions, as well as the detection of point landmarks.

A.1. Confocal Recordings of Arabidopsis Root Tips

In this section we give a brief summary of previous experiments on biological data published in Skibbe et al. (2010); Skibbe et al. (2012). Please refer these references for further details.

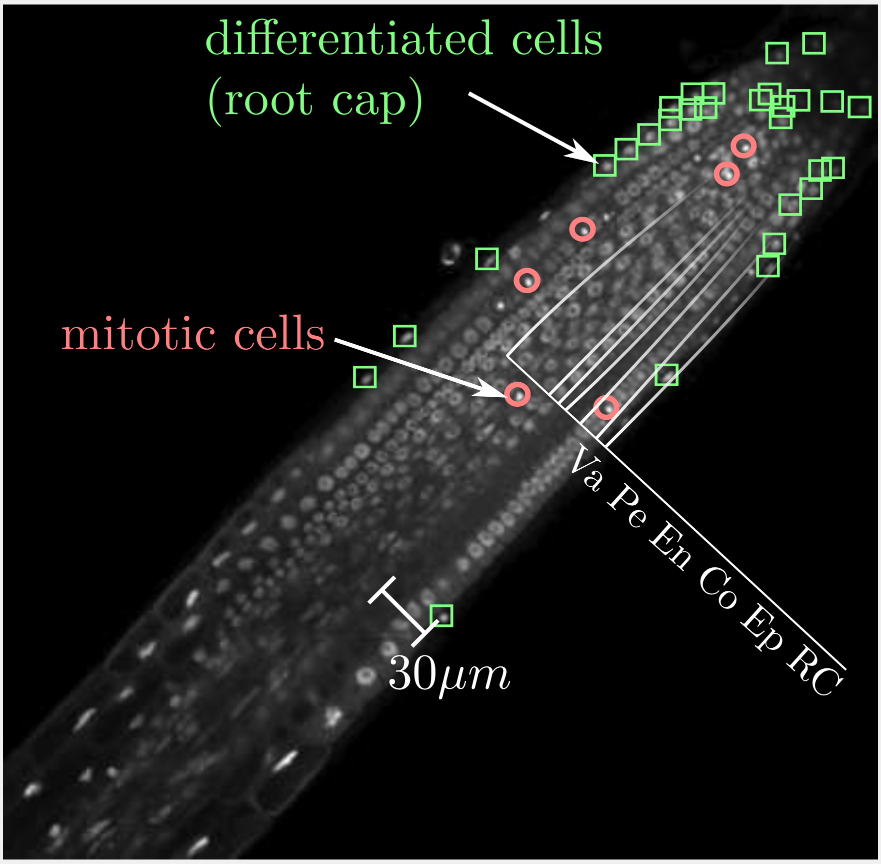
Manual annotation and evaluation of large cell populations in biological 3D images is a very time consuming, tedious and error-prone task. Analyzing a single 3D image by an expert often takes hours or even days. In this application we demonstrated how dense features derived via the spherical Laguerre transform and the spherical 3D Gabor transform allow for automated and objective detection and classification of a large number of cells in volumetric images.

Two recordings of Arabidopsis root tips, acquired via a confocal laser scanning microscope, were considered. The images were of the sizes $429 \times 419 \times 98$ and $461 \times 470 \times 101$. The cell nuclei were stained with DAPI[1]. The goal was to detect and classify the cell nuclei with respect to their cell state and correspondence to a cell layer. For our experiment we used the first dataset for training and the second dataset for evaluation. A slice of the training root is shown in Fig. A.1 a). We classed the cells by the cell layers (from outer to the inner) Root Cap (RC), Epidermis (Ep), Cortex (Co), Endodermis (En), Pericycle (Pe) and the Vasculature (Va); We further classed the cells by their cell states: an expert can distinguish cells that have already been differentiated (RC-cells), mitotic cells and stem cells. The training dataset contained 3614 stained nuclei and the test dataset 3608 stained nuclei.

We formed Gabor-Inv descriptors and Laguerre-Inv descriptors based on three different scales. One coarse scale covering a cells surrounding so that a cell's position is encoded into the descriptors and two finer scales representing the detailed shape of the cells for cell state classification. The descriptor was formed by using the power-spectrum. For the detection we used a random forest classifier Breiman (2001). The number of trees voting for a class was used to build a saliency map. We considered local maxima as detection candidates. For further details regarding the

---

[1] 4'-6-Diamidino-2-phenylindole



a) We aimed at detecting and classifying cells according to their state (mitotic/differentiated) and position (layer)

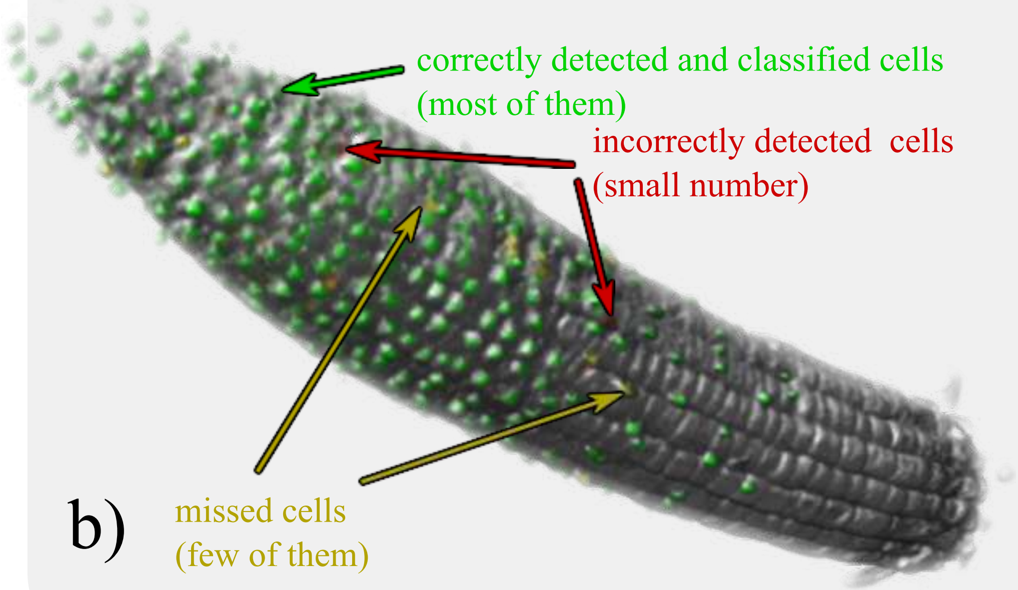
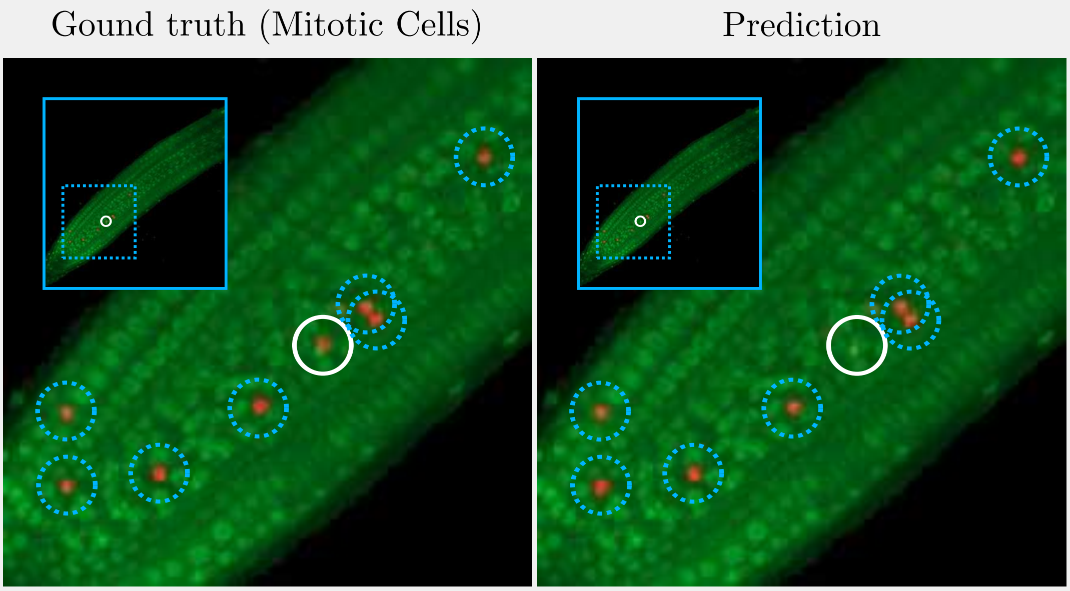
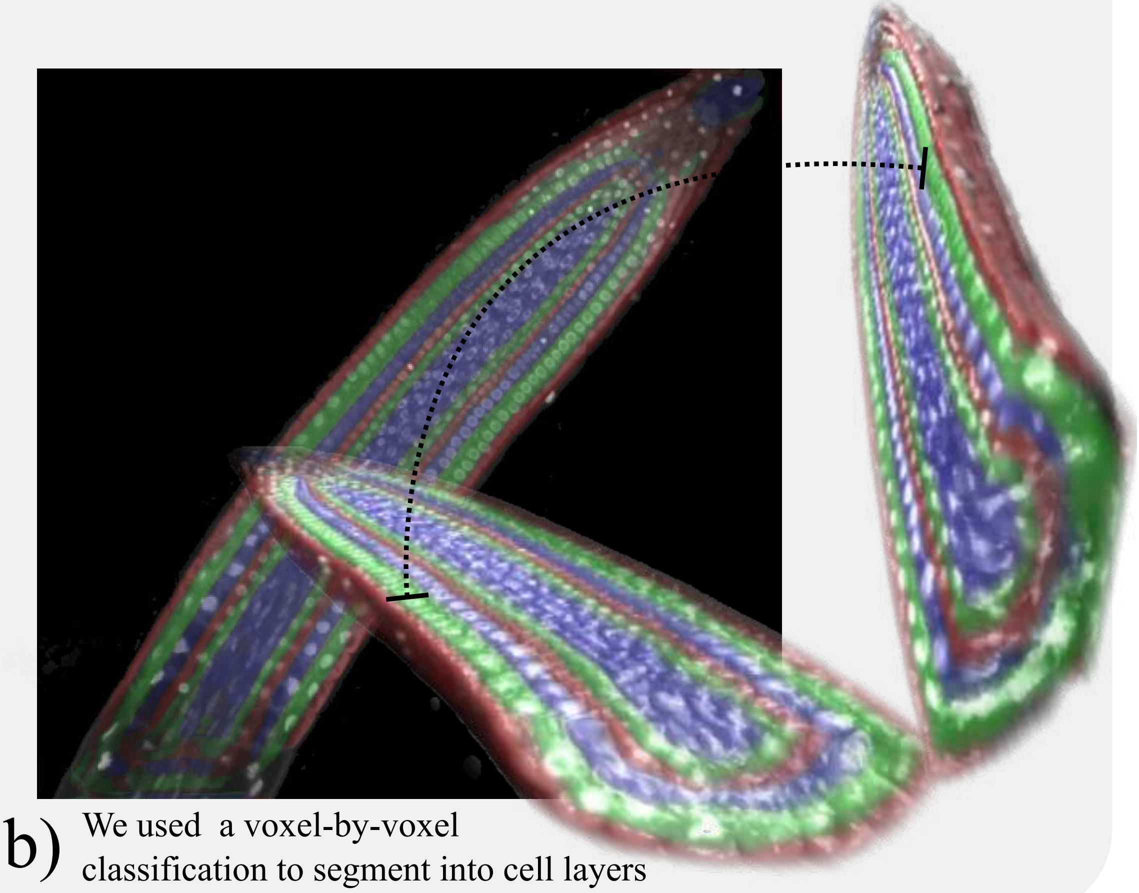


Figure A.1.: We aim at detecting and classifying cells



a) Detection of mitotic cells (left: manual labels, right: detections)
We missed one candidate



b) We used a voxel-by-voxel
classification to segment into cell layers

Figure A.2.: Cells are classified with respect to cell state and cell layer experiment setup we refer to Skibbe et al. (2010); Skibbe et al. (2012). In the experiments we correctly detected and classified most of the cells. Moreover, our approach performed significantly better than existing approaches like SIFT or the standard spherical harmonic transformation (In our experiment called SH-Inv). We depict an isosurface rendering showing detected and classified root-cap cells in Fig. A.1 b). False positive detections are colored in red, false negative detections in yellow. Correctly detected and classified cells in green. In Fig. A.2 a) we exemplarily show the detected mitotic cells in one slice of the test dataset.

A voxel-by-voxel classification can also be utilized for image segmentation. We exemplarily demonstrated a segmentation of the root image into different cell layers; see Fig. A.2 b). It is quite remarkable that we obtained spatially compact segments without exploiting any explicit neighborhood consistency as, for example, in a markov random field setting.

A.2. DTI Analysis

In this section we give a brief summary of previous experiments on DTI data of the human brain published in Skibbe and Reisert (2011); Skibbe and Reisert (2012b); Skibbe and Reisert (2013); Skibbe et al. (2011a). Please refer these references for further details.

The study of Magnetic Resonance (MR) imaging modalities is of great interest in fundamental neuroscience and medicine. MR imaging offers a wide range of different contrasts, providing scientists insights into anatomical and functional properties of the human brain. One may distinguish between anatomical contrasts (T1-weighted, T2-weighted) and derived contrasts that use combinations of different measurement parameters to infer underlying tissue properties. Examples of the latter are are Diffusion Weighted Imaging (DTI), High Angular Resolution Diffusion Imaging (HARDI Tuch et al. (1999)), functional MRI (fMRI) and Arterial Spin Labeling. The main difference to anatomical images is the rather bad image quality in terms of resolution and signal to noise ratio.

For several applications it is necessary to transfer information from the anatomical images to, for example, the low resolution DTI images. Usually co-registration is employed. One possible kinds of applications are group studies, where the determination of corresponding anatomical regions is necessary for inter subject comparisons. High quality anatomical images are used to find inter-subject correspondences, which are transfered to the low resolution contrasts by co-registration. Another application scenario is seed region generation for fiber tracking analysis. Seeds are selected on a manually labeled atlas. Then, the seeds are transfered to the DTI or HARDI contrast by co-registration with a template. In Fig. A.3 a) we visualized a result obtained via a state-of-the-art fiber tracking technique.

In our papers we proposed to omit the co-registration step and to directly realize a parcellation of, or a landmark detection in the HARDI signal. We proposed a two-steps approach based on rotation invariant image descriptors and machine learning techniques. We first learn the connection of voxel-label and voxel-appearance in a supervised manner using a co-registered atlas as label image and a descriptor image derived form the HARDI signal. Both are used for training a classifier. The second step is simply a voxel-by-voxel classification of new, unlabeled HARDI signals. Currently, a co-registration is necessary for creating a training set. Future work will focus on an unsupervised creation of a training set based on an automated determination of voxel correspondences based on both spatial regularization (difficult to omit completely) and appearance. A big advantage of our approach over registration based approaches: the success of our approach does not rely much on an existing, nearly perfect global transformation between two images. Hence it can easier cope with partial corrupted data, or with data stemming from patients with pathologies; see e.g. Fig. A.7 on page 190.

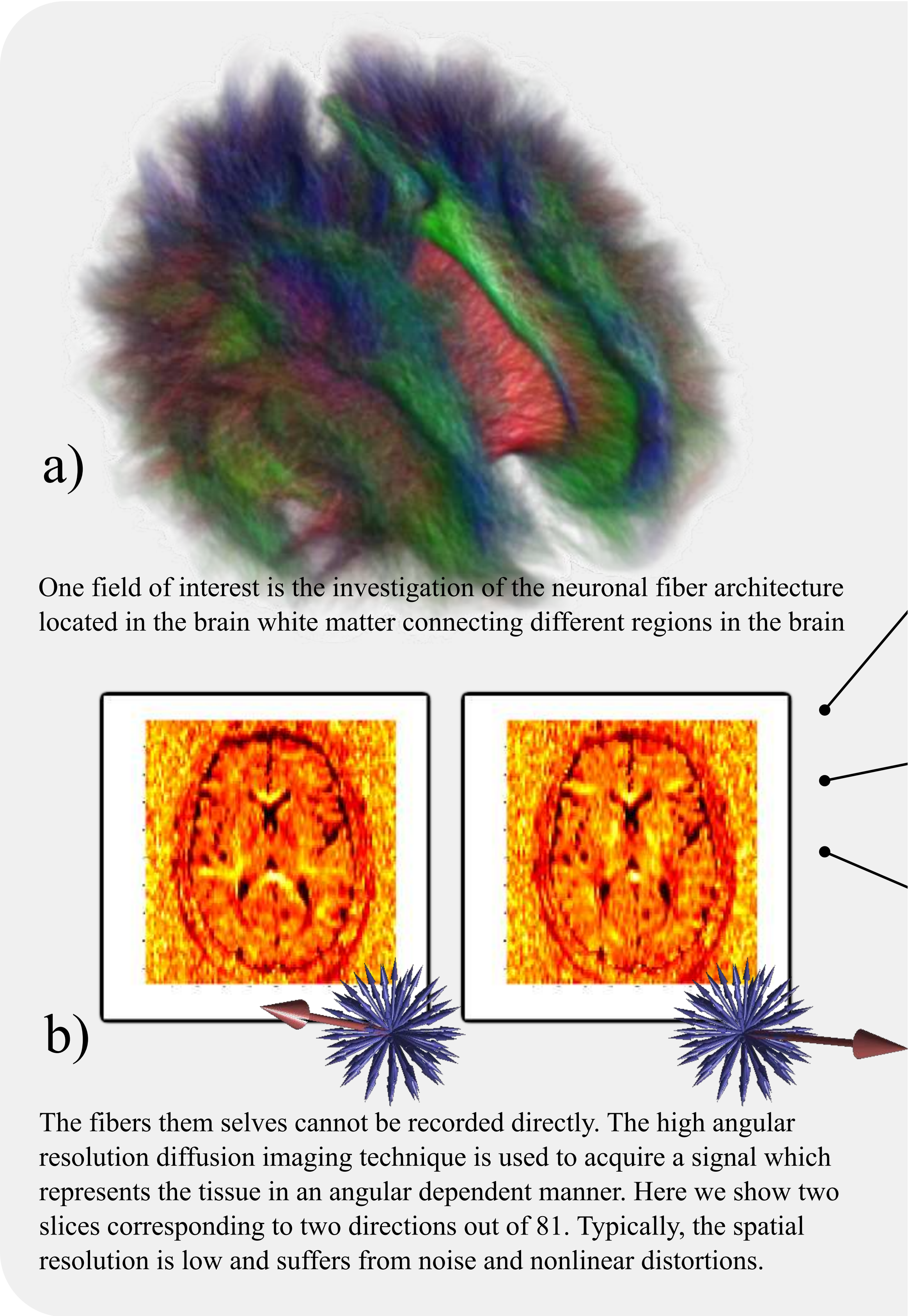
The HARDI signal is an angular dependent measurement HARDI : $\mathbb{R}^3 \to S_2$; see Fig. A.3 b) for an example. Therefore, spherical harmonics are a common representation of the HARDI signal. Since diffusion is symmetric, only symmetric spherical harmonic functions are necessary to represent the signal. A representation of a HARDI signal in terms of spherical harmonics has the form

$$\mathrm{HARDI}(\mathbf{x},\mathbf{n}) = \sum_{\ell \geq 0 \text{ is even}} \mathbf{a}^{\ell}(\mathbf{x})^{T} \mathbf{Y}^{\ell}(\mathbf{n}) \quad . \tag{A.1}$$

In our applications we used the partial Gauss-Laguerre transform for higher order tensor fields to generate new covariant neighborhood features from the expansion coefficients $\mathbf{a}^{\ell} \in \mathcal{T}_{\ell}$; see section 5.1.3.1 on page 125. We did this for several scales so that we were representing both in the descriptors local structural details of the tissue and the location in the brain. Depending on the applications, we formed different kinds of locally rotation invariant descriptor images.

Tissue Classification. The neuronal fiber architecture is located in the brain white matter. Hence the classification into different tissue types is an indispensable prerequisite for many further studies. In a first experiment we aimed at the classification into two different tissue types: brain white matter and brain gray matter. We further used a background class representing the remaining classes such as water or air. We formed rotation invariant descriptors based on the power spectrum. For the machine learning part we used a random forest classifier. In our experiments we performed superior over a comparable approach proposed by Schnell et al. (2009). They solely computed the power spectrum based on the expansion coefficients $\mathbf{a}^{\ell}$ without considering a voxels neighborhood which we belief is the reason for our superior performance. Our approach highly benefits form the tensor expansion in different scales. We show an isorendering of a classification result in Fig. A.4 c). For further details regarding the experiments see Skibbe et al. (2011a).

Parcellation in Anatomical Relevant Regions. The parcellation of the brain white matter and gray matter into different anatomically relevant regions play an important role in group studies. For instance, suppose you are interested in all fibers connecting with the frontal lobe. In such a scenario a classification into different tissue types is by far not sufficient. In our paper Skibbe and Reisert (2011) we showed that the proposed technique can also cope with this challenge. However, we observed that we need even larger scales covering the size of the whole brain in order to precisely encode the position into the descriptors. We exemplarily depict two classification results in Figs. A.4 a-b). In the first image we show a parcellation into 54 anatomical relevant regions, in the second image a parcelltation into 11 regions. For generating



One field of interest is the investigation of the neuronal fiber architecture located in the brain white matter connecting different regions in the brain

The fibers them selves cannot be recorded directly. The high angular resolution diffusion imaging technique is used to acquire a signal which represents the tissue in an angular dependent manner. Here we show two slices corresponding to two directions out of 81. Typically, the spatial resolution is low and suffers from noise and nonlinear distortions.

Figure A.3.: (Visualization of fibers based on tracking results of Reisert et al. (2010))

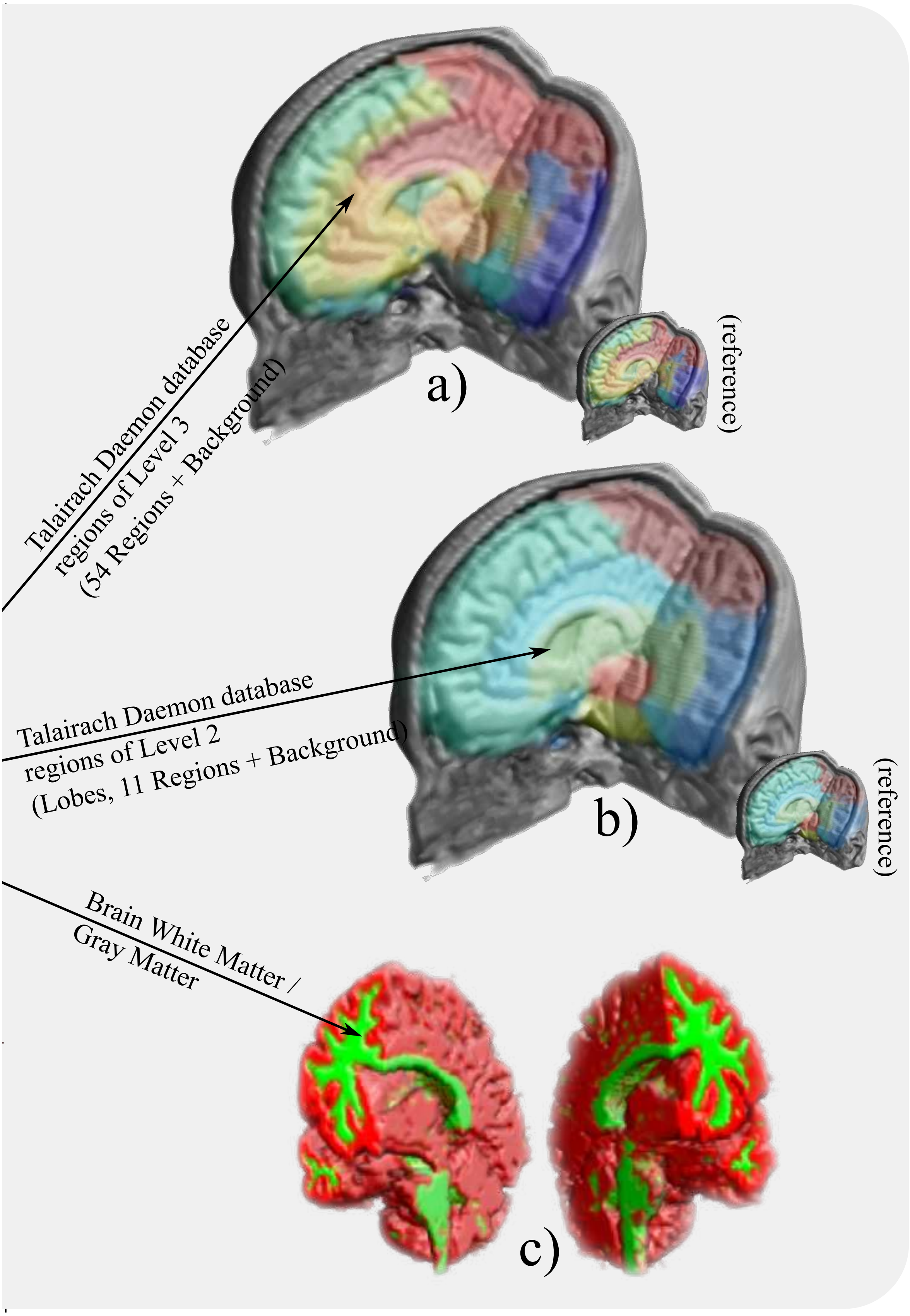


Figure A.4.: A voxel-by-voxel classification is used for image parcellation

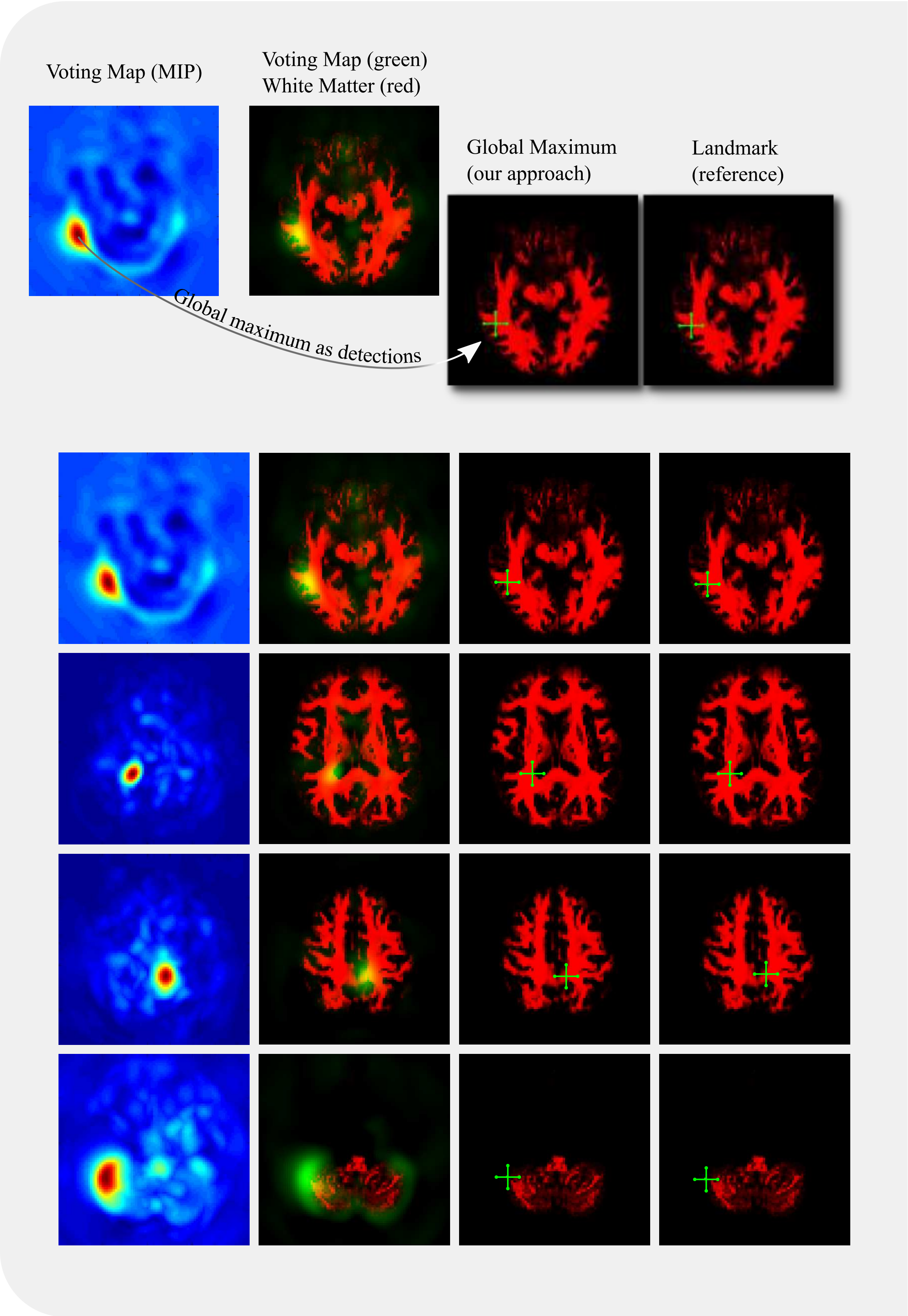


Figure A.5.: Landmark detection: the symmetry of the brain requires odd third order tensor products to distinguish between the left and the right hemisphere.

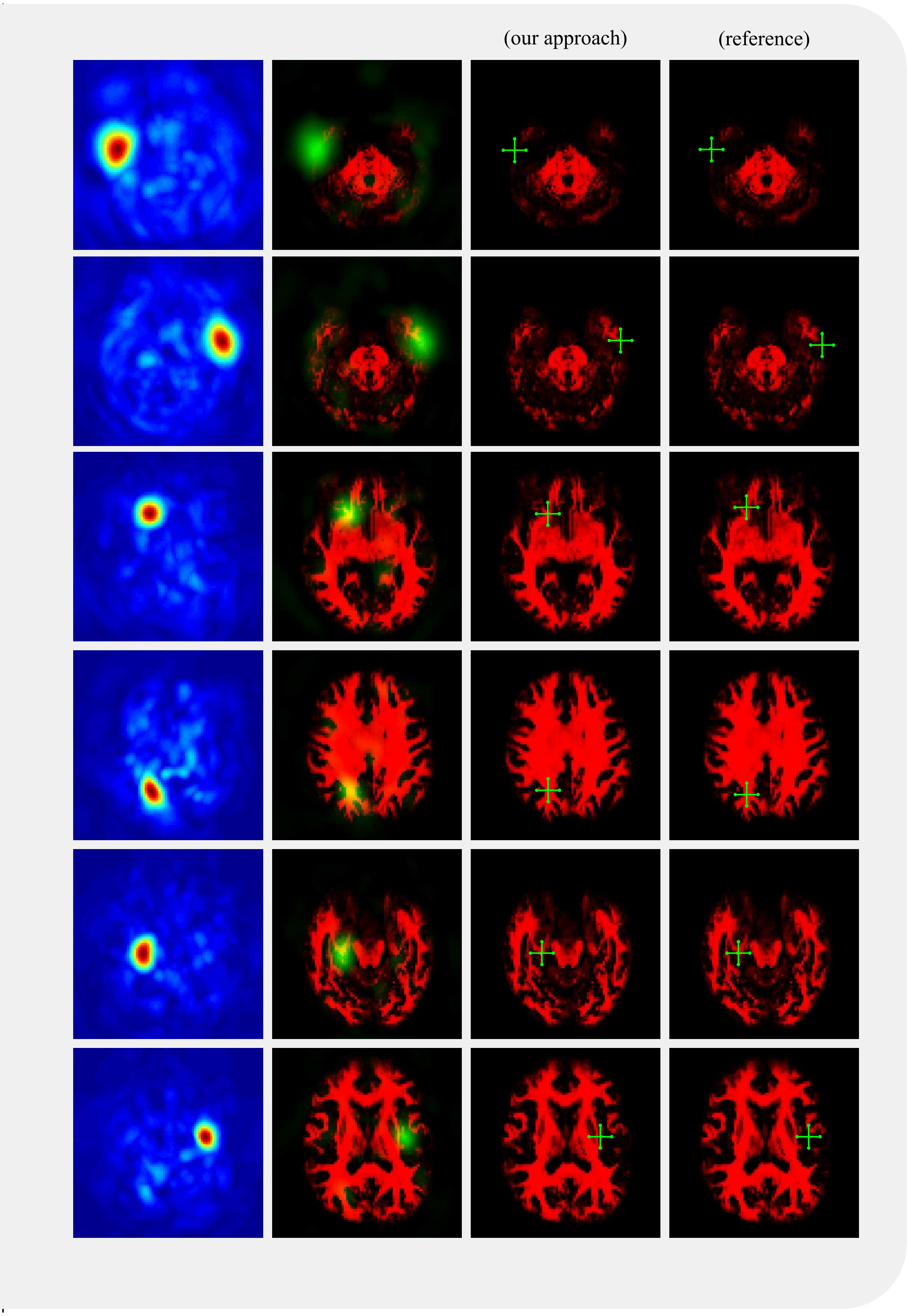


Figure A.6.: We use a coarse to fine approach to detect more than 20000 landmarks within a few minutes

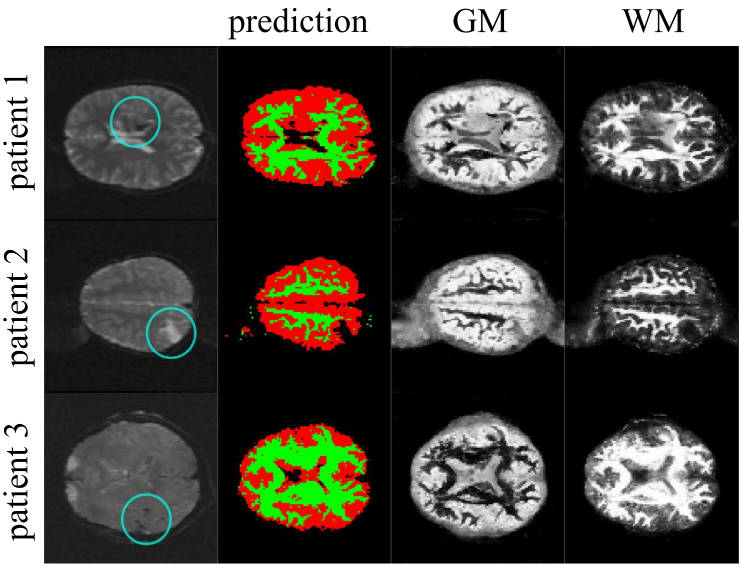


Figure A.7.: Predictions and probability maps for gray matter (GM) and white matter (WM) on a patient database.

the regions we used atlas information provided by the Talairach Daemon database (we used the WFU PickAtlas[2], a software that provides a method for generating ROI masks based on the Talairach Daemon database; see Lancaster (1997)).

In Figs. A.8 to A.9 on pages 191–192 we show the votes of a random forest classifier for the 54 regions as heat maps. We show the corresponding T1-weighted image in the background.

Unique Point Landmark Detection. In our most recent experiment we put the focus on single landmark detection. The goal was to distinguish and detect more than 20 thousand landmarks within a HARDI signal of a human brain in reasonable time. For solving this challenging task we made use of region features based on the harmonic filter framework (see appendix D on page 225 for a brief introduction in harmonic filters). Instead of scalar valued images, we used the expansion images $\mathbf{a}^\ell$ of the HARDI signal as input images. In order to distinguish the left and the right hemisphere we used a combination of both, the second order region features Eq. (D.10) and only odd products of the third order region features Eq. (D.11). We omitted computing the even third order products to keep memory usage low.

For the experiment we placed about 20000 landmarks within the brain gray and white matter in an equidistant manner. For each dataset, the computation of the features and the detection of of all landmarks took about 5 minutes. We exemplarily depict detection results in Figs. A.5 to A.6 on pages 188–189. We further show some feature images in Figs. A.10 to A.11 on pages 193–194.

---

[2] `http://fmri.wfubmc.edu/software/PickAtlas`

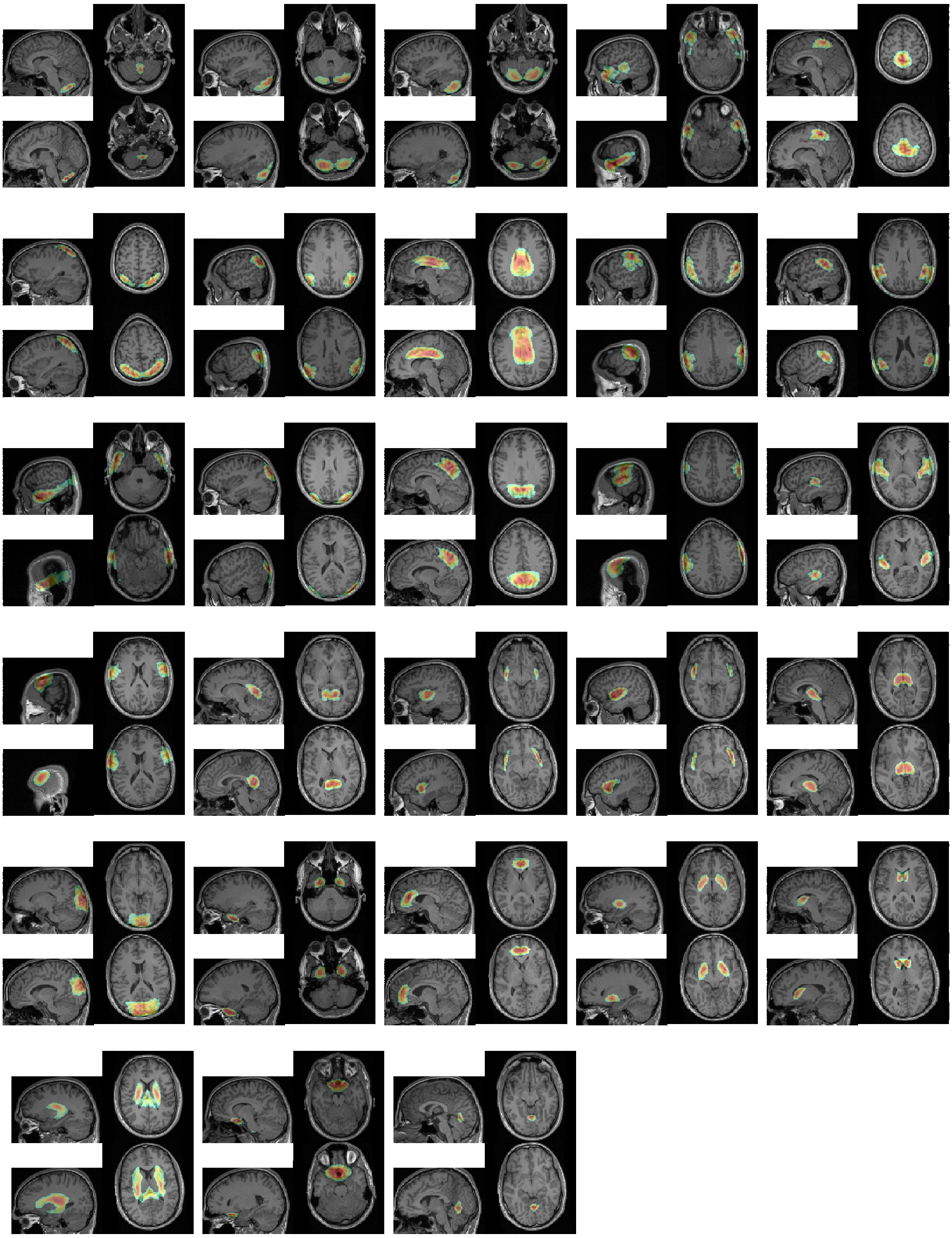


Figure A.8.: Heat maps representing the probability for all 54 regions of Level 3 of the Talairach daemon database exemplarily shown for two datasets (continued in figure A.9).

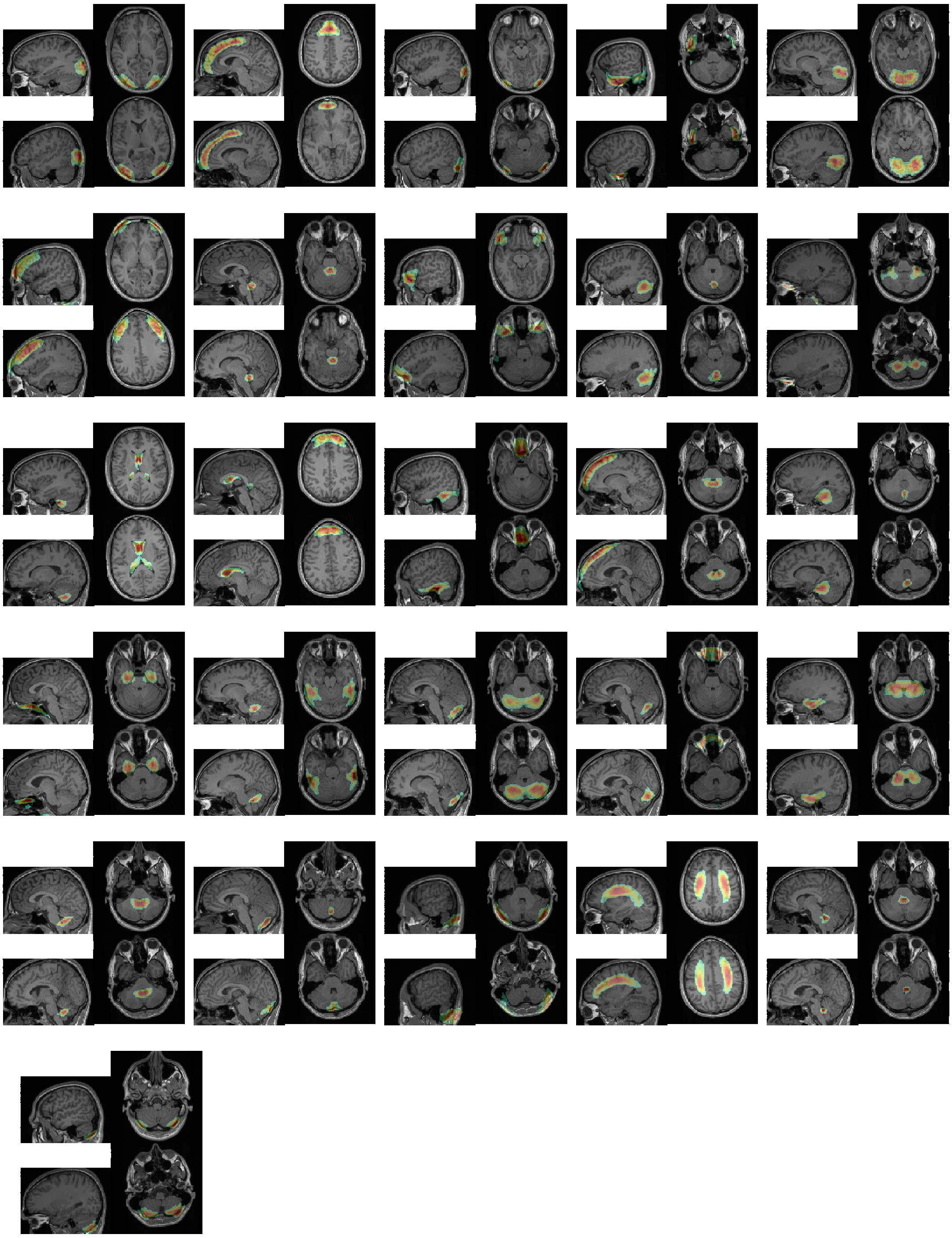
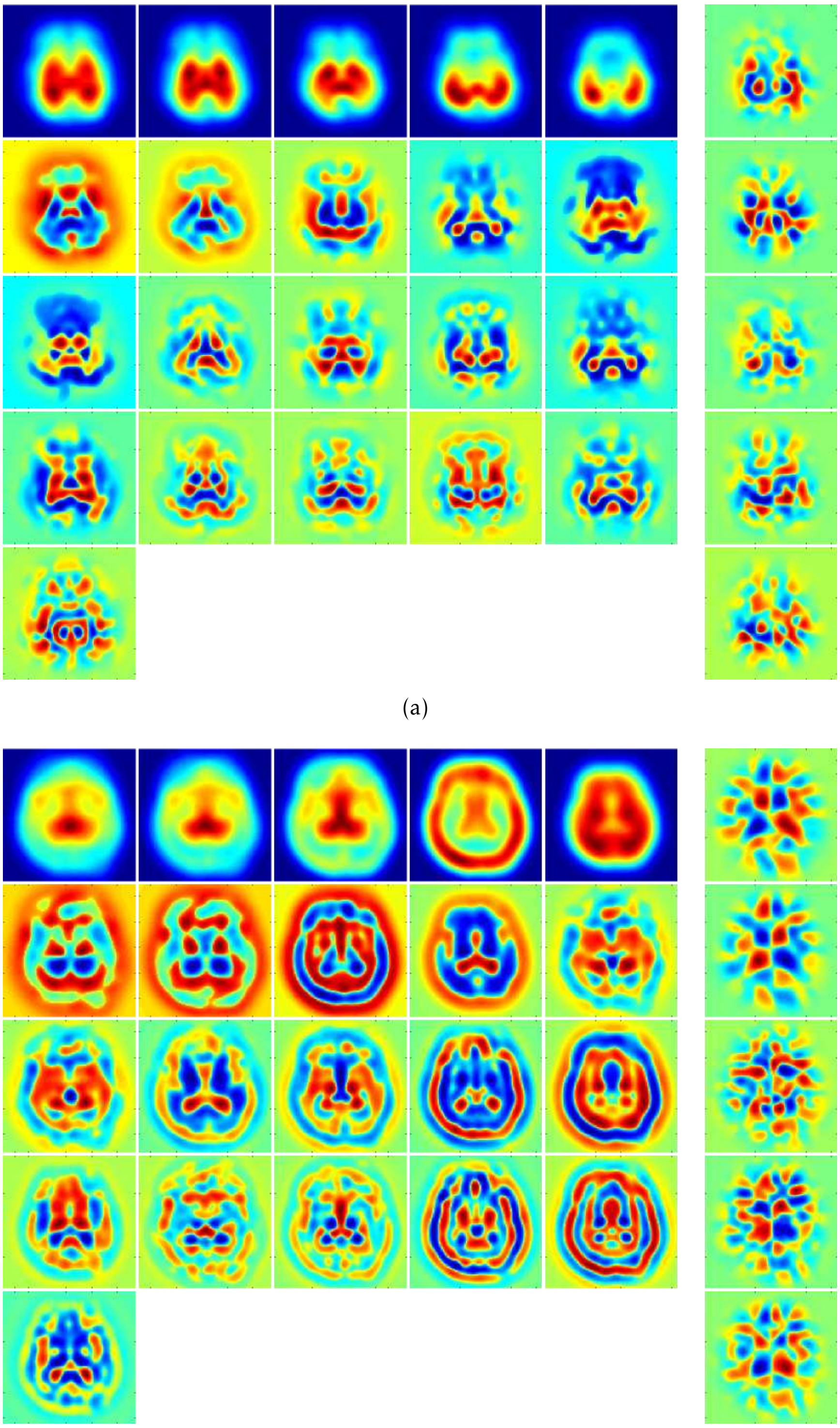


Figure A.9.: Heat maps representing the probability for all 54 regions of Level 3 of the Talairach daemon database exemplarily shown for two datasets (starting in figure A.8).



(a)

(b)

Figure A.10.: Features images used in the landmark detection task

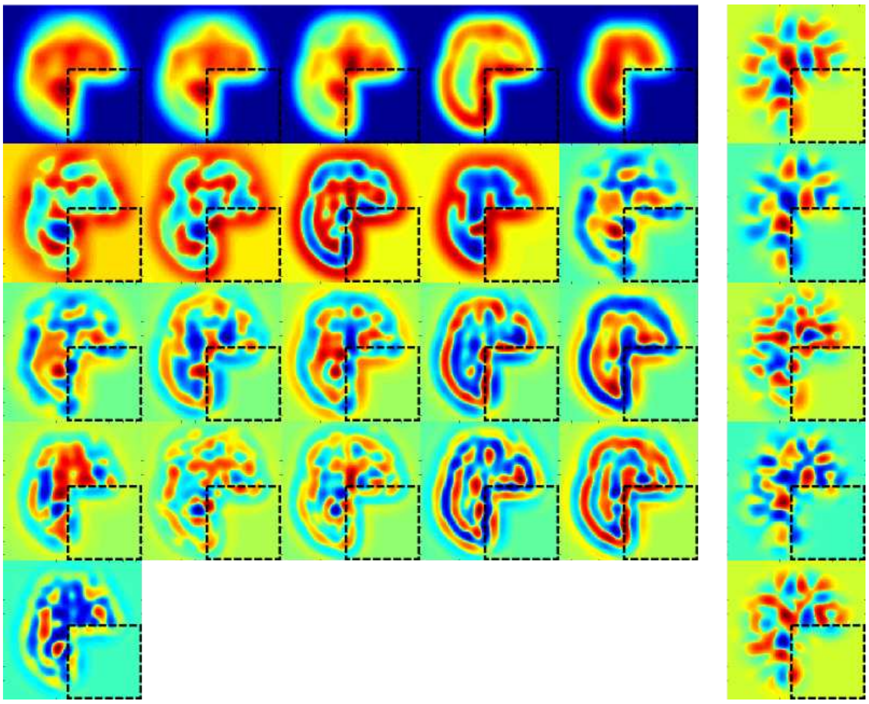


Figure A.11.: The proposed region features are robust against strong distortions. Here we cropped one forth of the image. The remaining feature images are still very similar to the feature images of the unaltered image; see on the preceding page. Note that thees features are incorporating a large voxel neighborhood.

B Auxiliary Calculations

B.1. Introduction

Proof: $SE(3)$ covariance of $\mathcal{F}$; Eq. (1.9).

$$\mathcal{F}\{\boldsymbol{\tau}^P_{\mathbf{t}}\boldsymbol{\tau}^P_{\theta} I\}(\mathbf{r}) = F(\boldsymbol{\tau}^P_{(-\mathbf{r})}\boldsymbol{\tau}^P_{\mathbf{t}}\boldsymbol{\tau}^P_{\theta} I) = F(\boldsymbol{\tau}^P_{(\mathbf{t}-\mathbf{r})}\boldsymbol{\tau}^P_{\theta} I) = F(\boldsymbol{\tau}^P_{\theta}\boldsymbol{\tau}^P_{(\mathbf{U}^T_{\theta}(\mathbf{t}-\mathbf{r}))} I)$$

$$\overset{\text{(Eq. (1.7))}}{=} \boldsymbol{\tau}^{\mathbb{C}^n}_{\theta} F(\boldsymbol{\tau}^P_{(\mathbf{U}^T_{\theta}(\mathbf{t}-\mathbf{r}))} I) \overset{\text{(Eq. (1.8))}}{=} \boldsymbol{\tau}^{\mathbb{C}^n}_{\theta}\mathcal{F}\{I\}(\mathbf{U}^T_{\theta}(\mathbf{r}-\mathbf{t})) = (\boldsymbol{\tau}^{\hat{\mathbf{V}}}_{\mathbf{t}}\boldsymbol{\tau}^{\hat{\mathbf{V}}}_{\theta}\mathcal{F}\{I\})(\mathbf{r}).$$

$\square$

Proof: $SO(3)$ covariance of F and $\mathcal{F}$ can be fully recovered from F; Eq. (1.11). $SO(3)$ covariance of F:

$$F(\boldsymbol{\tau}^P_{\theta} I) = \mathcal{F}\{\boldsymbol{\tau}^P_{\theta} I\}(\mathbf{0}) \overset{\text{(Eq. (1.9))}}{=} (\boldsymbol{\tau}^{\hat{\mathbf{V}}}_{\theta}\mathcal{F}\{I\})(\mathbf{0}) \overset{\text{(Eq. (1.10))}}{=} \boldsymbol{\tau}^{\mathbb{C}^n}_{\theta}(\mathcal{F}\{I\}(\mathbf{0})) = \boldsymbol{\tau}^{\mathbb{C}^n}_{\theta} F(I)\,.$$

If $F(I) := \mathcal{F}\{I\}(\mathbf{0})$, then

$$\mathcal{F}'\{I\}(\mathbf{r}) := F(\boldsymbol{\tau}^P_{(-\mathbf{r})} I) \overset{\text{(Def. } F)}{=} \mathcal{F}\{\boldsymbol{\tau}^P_{(-\mathbf{r})} I\}(\mathbf{0}) \overset{\text{(Eq. (1.10))}}{=} \mathcal{F}\{I\}(\mathbf{r}), \tag{B.1}$$

that is, $\mathcal{F}$ can be fully recovered given F. $\square$

B.2. Harmonic Analysis in 3D

B.2.1. Rotations in 3D: The Reducible Representations of $SO(3)$

In Eq. (2.38) we only implicitly define the linear transformation that rotates the image. Actually, this transformation is infinitely dimensional. We can find an explicit representation by expanding I in terms of its derivatives, i.e. by performing a Taylor expansion and see how a rotation transforms the derivatives. Let $\mathbf{x} = (x_0, x_1, x_2)^T \in \mathbb{R}^3$ be the point of expansion. The Taylor expansion of I is then defined by

$$
\begin{aligned}
I(\mathbf{r}) &= \sum_{j=0}^{\infty} \left\{ \frac{1}{j!} \left[\sum_{n=0}^{2} (r_n - x_n) \frac{\partial}{\partial r'_n} \right]^j I(r'_0, r'_1, r'_2) \right\}_{\substack{r'_0 = x_0 \\ r'_1 = x_1 \\ r'_2 = x_2}} \\
&= \sum_{j=0}^{\infty} \frac{1}{j!} \mathcal{D}^j (I, \mathbf{x}) \,,
\end{aligned}
\tag{B.2}
$$

where

$$
\mathcal{D}^j (I, \mathbf{x}) := \sum_{\substack{i_0, \cdots, i_j \\ i_k \in \{0,1,2\}}} [\tfrac{\partial}{\partial r_{i_0}} \cdots \tfrac{\partial}{\partial r_{i_j}} f](\mathbf{x})(r_{i_0} - x_{i_0}) \cdots (r_{i_j} - x_{i_j}) \,,
\tag{B.3}
$$

are the terms containing all derivatives and monomials of order j. Now suppose we are rotating the image. Following Eq. (2.38) we get

$$
(\mathfrak{g}I)(\mathbf{r}) = \sum_{j=0}^{\infty} \frac{1}{j!} \mathcal{D}^j (\mathfrak{g}I, \mathbf{x}) \,.
\tag{B.4}
$$

Consecutively computing the derivatives according to

$$
\frac{\partial}{\partial r_i} (\mathfrak{g}I)(\mathbf{r}) = \left(\frac{\partial}{\partial r_i} (\mathbf{U}_{\mathfrak{g}}^T \mathbf{r}) \right) \left[\frac{\partial}{\partial r_n} I \right] (\mathbf{U}_{\mathfrak{g}}^T \mathbf{r}) = U_{\mathfrak{g}_{(i,n)}} \left[\frac{\partial}{\partial r_n} I \right] (\mathbf{U}_{\mathfrak{g}}^T \mathbf{r}) \,,
\tag{B.5}
$$

we get

$$
\begin{aligned}
\mathcal{D}^j (\mathfrak{g}I, \mathbf{x}) &= \sum_{\substack{i_0, \cdots, i_j \\ i_k \in \{0,1,2\}}} [\tfrac{\partial}{\partial r_{i_0}} \cdots \tfrac{\partial}{\partial r_{i_j}} (\mathfrak{g}f)](\mathbf{x})(r_{i_0} - x_{i_0}) \cdots (r_{i_j} - x_{i_j}) \\
&= \sum_{\substack{i_0, \cdots, i_j \\ n_0, \cdots, n_j \\ i_k, n_k \in \{0,1,2\}}} U_{\mathfrak{g}_{(i_0,n_0)}} \cdots U_{\mathfrak{g}_{(i_j,n_j)}} [\tfrac{\partial}{\partial r_{n_0}} \cdots \tfrac{\partial}{\partial r_{n_j}} I](\mathbf{U}_{\mathfrak{g}}^T \mathbf{x})(r_{i_0} - x_{i_0}) \cdots (r_{i_j} - x_{i_j}) \\
&= \sum_{\substack{i_0, \cdots, i_j \\ i_k \in \{0,1,2\}}} (r_{i_0} - x_{i_0}) \cdots (r_{i_j} - x_{i_j}) \underbrace{\sum_{\substack{n_0, \cdots, n_j \\ n_k \in \{0,1,2\}}} U_{\mathfrak{g}_{(i_0,n_0)}} \cdots U_{\mathfrak{g}_{(i_j,n_j)}} [\tfrac{\partial}{\partial r_{n_0}} \cdots \tfrac{\partial}{\partial r_{n_j}} I](\mathbf{U}_{\mathfrak{g}}^T \mathbf{x})}_{\text{rotation of derivatives of order } j} \,.
\end{aligned}
\tag{B.6}
$$

Therefore, Eq. (B.4) together with Eq. (B.6) gives us an explicit representation of a 3D image rotation. Moreover, considering Eq. (B.6) more closely, we see that the 3D image rotation is a direct sum of finite dimensional linear transformations, separately acing on the derivatives of I, with

$$
[\tfrac{\partial}{\partial r_{n_0}} \cdots \tfrac{\partial}{\partial r_{n_j}} (\mathfrak{g}I)](\mathbf{x}) = \sum_{\substack{n_0, \cdots, n_j \\ n_k \in \{0,1,2\}}} U_{\mathfrak{g}_{(i_0,n_0)}} \cdots U_{\mathfrak{g}_{(i_j,n_j)}} [\tfrac{\partial}{\partial r_{n_0}} \cdots \tfrac{\partial}{\partial r_{n_j}} I](\mathbf{U}_{\mathfrak{g}}^T \mathbf{x}) \quad .
\tag{B.7}
$$

B.2.2. Spherical Harmonic Functions

Proof: The feature extraction $F(I,\ell,r)$ is covariant to $SO(3)$. We show that $F(I,\ell,r)$ defined in Eq. (2.73) is covariant to 3D rotations; see Def. 1.2.1 on page 18 for covariance. Let us consider the action of an element of the 3D rotation group $SO(3)$ on an image $I \in L_2(\mathbb{R}^3)$. Then Eq. (2.73) transforms according to

$$\begin{aligned}
F(\mathfrak{g}I,\ell,r) &= \langle \mathfrak{g}I, \overline{\mathbf{Y}^\ell \delta_r}\rangle \\
&= \int_{\mathbf{r}'\in\mathbb{R}^3} \underbrace{(\mathfrak{g}I)(\mathbf{r}')}_{\substack{SO(3)\text{ group action}\\ \text{on images}}} \mathbf{Y}^\ell(\mathbf{r}')\delta(\|\mathbf{r}'\|-r)d\mathbf{r}' = \int_{\mathbf{r}'\in\mathbb{R}^3} \underbrace{I(\mathbf{U}_{\mathfrak{g}}^T\mathbf{r}')}_{\substack{SO(3)\text{ group action}\\ \text{image domain}}} \mathbf{Y}^\ell(\mathbf{r}')\delta(\|\mathbf{r}'\|-r)d\mathbf{r}' \\
&= \int_{\mathbf{r}'\in\mathbb{R}^3} I(\mathbf{r}')\mathbf{Y}^\ell(\mathbf{U}_{\mathfrak{g}}\mathbf{r}')\delta(\|\mathbf{r}'\|-r)d\mathbf{r}' \overset{(\text{Eq. (2.62)})}{=} \mathbf{D}^\ell_{\mathfrak{g}} \int_{\mathbf{r}'\in\mathbb{R}^3} I(\mathbf{r}')\mathbf{Y}^\ell(\mathbf{r}')\delta(\|\mathbf{r}'\|-r)d\mathbf{r}' \\
&= \underbrace{\mathbf{D}^\ell_{\mathfrak{g}} F(I,\ell,r)}_{\substack{SO(3)\text{ group action}\\ \text{harmonic domain}}} = \underbrace{\mathfrak{g}F(I,\ell,r)}_{\substack{SE(3)\text{ group action}\\ \text{on feature}}} \quad .
\end{aligned} \tag{B.8}$$

□

$SE(3)$ Covariant Spherical Harmonics Transformation $\mathcal{SH}^\ell$. We show that the spherical harmonic transform defined in Eq. (2.75) is covariant to Euclidean motion; see Def. 1.2.1 on page 18 for covariance. Let us consider the action of an element of the Euclidean motion group $SE(3)$ on an image $I \in L_2(\mathbb{R}^3)$. We separately consider a rotation $\mathfrak{g} \in SO(3)$ and a translation $\mathfrak{h}_{\mathbf{t}} \in T(3)$. Then Eq. (2.75) transforms according to

$$\begin{aligned}
\mathbf{a}'^\ell(\mathbf{x},r) &= \mathcal{SH}^\ell\{\mathfrak{h}_{\mathbf{t}}\mathfrak{g}I,r\}(\mathbf{x}) = \langle \mathfrak{h}_{(-\mathbf{x})}\mathfrak{h}_{\mathbf{t}}\mathfrak{g}I, \overline{\mathbf{Y}^\ell\delta_r}\rangle \\
&= \int_{\mathbf{r}'\in\mathbb{R}^3} \underbrace{(\mathfrak{h}_{\mathbf{t}}\mathfrak{g}I)(\mathbf{r}'+\mathbf{x})}_{\substack{SE(3)\text{ group action}\\ \text{on images}}} \mathbf{Y}^\ell(\mathbf{r}')\delta(\|\mathbf{r}'\|-r)d\mathbf{r}' \\
&= \int_{\mathbf{r}'\in\mathbb{R}^3} (\mathfrak{g}I)(\mathbf{r}'+\mathbf{x}-\mathbf{t})\mathbf{Y}^\ell(\mathbf{r}')\delta(\|\mathbf{r}'\|-r)d\mathbf{r}' \\
&= \int_{\mathbf{r}'\in\mathbb{R}^3} \underbrace{I(\mathbf{U}_{\mathfrak{g}}^T\mathbf{r}'+\mathbf{U}_{\mathfrak{g}}^T(\mathbf{x}-\mathbf{t}))}_{\substack{SE(3)\text{ group action}\\ \text{image domain}}}\mathbf{Y}^\ell(\mathbf{r}')\delta(\|\mathbf{r}'\|-r)d\mathbf{r}' \\
&= \int_{\mathbf{r}'\in\mathbb{R}^3} I(\mathbf{r}'+\mathbf{U}_{\mathfrak{g}}^T(\mathbf{x}-\mathbf{t}))\mathbf{Y}^\ell(\mathbf{U}_{\mathfrak{g}}\mathbf{r}')\delta(\|\mathbf{r}'\|-r)d\mathbf{r}' \\
&= \mathbf{D}^\ell_{\mathfrak{g}} \int_{\mathbf{r}'\in\mathbb{R}^3} I(\mathbf{r}'+\mathbf{U}_{\mathfrak{g}}^T(\mathbf{x}-\mathbf{t}))\mathbf{Y}^\ell(\mathbf{r}')\delta(\|\mathbf{r}'\|-r)d\mathbf{r}' \\
&= \underbrace{\mathbf{D}^\ell_{\mathfrak{g}}\mathbf{a}^\ell(\mathbf{U}_{\mathfrak{g}}^T(\mathbf{x}-\mathbf{t}),r)}_{\substack{SE(3)\text{ group action}\\ \text{harmonic domain}}} = (\mathfrak{g}\mathbf{a}^\ell)(\mathbf{x}-\mathbf{t},r) = \underbrace{(\mathfrak{h}_{\mathbf{t}}\mathfrak{g}\mathbf{a}^\ell)(\mathbf{x},r)}_{\substack{SE(3)\text{ group action}\\ \text{on coefficients}}} \quad .
\end{aligned} \tag{B.9}$$

□

B.2.3. Spherical Tensor Algebra

B.2.3.1. The Angular Power- and Bi-Spectrum

Proof. Proving theorem 2.2.27 and 2.2.28 on page 68. Let $f^1 \in L_2(SO(3))$ and $f^2 \in L_2(SO(3))$ be two functions defined on the rotation group $SO(3)$.

According to Eq. (2.55),

$$f(\mathfrak{g}) = \sum_{\ell=0}^{\infty} \tfrac{(2\ell+1)}{8\pi^2} \sum_{m,n=-\ell}^{\ell} A^{\ell}_{m,n} [\mathbf{D}^{\ell}_{\mathfrak{g}}]_{mn} \quad . \tag{B.10}$$

Hence

$$f(\mathfrak{g}\mathfrak{g}_1) = \sum_{\ell=0}^{\infty} \tfrac{(2\ell+1)}{8\pi^2} \sum_{m,n=-\ell}^{\ell} A^{\ell}_{m,n} [\mathbf{D}^{\ell}_{\mathfrak{g}} \mathbf{D}^{\ell}_{\mathfrak{g}_1}]_{mn} = \sum_{\ell=0}^{\infty} \tfrac{(2\ell+1)}{8\pi^2} \sum_{m,n,i=-\ell}^{\ell} A^{\ell}_{m,n} [\mathbf{D}^{\ell}_{\mathfrak{g}}]_{mi} [\mathbf{D}^{\ell}_{\mathfrak{g}_1}]_{in} \ , \tag{B.11}$$

(just imaging the case where f is an element of the Wigner-D matrices itself).

We define the angular correlation by

$$\begin{aligned}
C(\mathfrak{g}') &= \int_{SO(3)} f^1(\mathfrak{g}) \overline{f^2(\mathfrak{g}\mathfrak{g}')} d\mathfrak{g} \\
&= \int_{SO(3)} \sum_{\ell_1,m_1,n_1} \tfrac{2\ell_1+1}{8\pi^2} [\mathbf{D}^{\ell_1}_{\mathfrak{g}}]_{m_1,n_1} A^{\ell_1}_{m_1,n_1} \sum_{\ell_2,m_2,n_2,i_2} \tfrac{2\ell_2+1}{8\pi^2} \overline{[\mathbf{D}^{\ell_2}_{\mathfrak{g}}]_{m_2,i_2} [\mathbf{D}^{\ell_2}_{\mathfrak{g}'}]_{i_2,n_2} B^{\ell_2}_{m_2,n_2}} d\mathfrak{g} \\
&= \sum_{\substack{\ell_1,m_1,n_1, \\ \ell_2,m_2,n_2,i_2}} \tfrac{2\ell_1+1}{8\pi^2} A^{\ell_1}_{m_1,n_1} \tfrac{2\ell_2+1}{8\pi^2} \overline{[\mathbf{D}^{\ell_2}_{\mathfrak{g}'}]_{i_2,n_2} B^{\ell_2}_{m_2,n_2}} \underbrace{\int_{SO(3)} [\mathbf{D}^{\ell_1}_{\mathfrak{g}}]_{m_1,n_1} [\overline{\mathbf{D}^{\ell_2}_{\mathfrak{g}}}]_{m_2,i_2} d\mathfrak{g}}_{= \frac{8\pi^2}{2\ell_1+1} \delta_{\ell_1 \ell_2} \delta_{m_1,m_2} \delta_{n_1,i_2}} \\
&= \sum_{\substack{\ell_1,m_1, \\ n_1,n_2}} \tfrac{2\ell_1+1}{8\pi^2} A^{\ell_1}_{m_1,n_1} \tfrac{2\ell_1+1}{8\pi^2} \overline{[\mathbf{D}^{\ell_1}_{\mathfrak{g}'}]_{n_1,n_2} B^{\ell_1}_{m_1,n_2}} \frac{8\pi^2}{2\ell_1 + 1} \\
&= \sum_{\substack{\ell_1,m_1, \\ n_1,n_2}} \tfrac{2\ell_1+1}{8\pi^2} A^{\ell_1}_{m_1,n_1} \overline{[\mathbf{D}^{\ell_1}_{\mathfrak{g}'}]_{n_1,n_2} B^{\ell_1}_{m_1,n_2}} \ ,
\end{aligned} \tag{B.12}$$

where we use the expansion of f^1 and f^2 in terms of the irreducible representations of $SO(3)$ according to Eq. (2.55) on page 54 (Note that we use the Schmidt semi-normalization here). The Fourier correspondence of the correlation can be obtained by projecting the angular correlation onto the irreducible representations of $SO(3)$

(the Wigner D-Matrices). We get

$$
\begin{aligned}
C^{\ell}_{mn} &= \int_{SO(3)} C(\mathfrak{g}, r)[\overline{\mathbf{D}^{\ell}_{\mathfrak{g}}}]_{mn} d\mathfrak{g} \\
&= \int_{SO(3)} \sum_{\substack{\ell_1, m_1, n_1, \\ n_2}} \tfrac{2\ell_1+1}{8\pi^2} A^{\ell_1}_{m_1,n_1} \overline{[\mathbf{D}^{\ell_1}_{\mathfrak{g}'}]_{n_1,n_2} B^{\ell_1}_{m_1,n_2}} [\overline{\mathbf{D}^{\ell}_{\mathfrak{g}}}]_{mn} d\mathfrak{g} \\
&= \sum_{\substack{\ell_1, m_1, n_1, \\ n_2}} A^{\ell_1}_{m_1,n_1} \tfrac{2\ell_1+1}{8\pi^2} \overline{B^{\ell_1}_{m_1,n_2}} (-1)^{(n1+n2)} \underbrace{\int_{SO(3)} [\mathbf{D}^{\ell_1}_{\mathfrak{g}}]_{(-n_1),(-n_2)} [\overline{\mathbf{D}^{\ell}_{\mathfrak{g}}}]_{mn} d\mathfrak{g}}_{= \frac{8\pi^2}{2\ell+1} \delta_{\ell_1 \ell} \delta_{(-n_1),m} \delta_{(-n_2),n}} \\
&= \sum_{m_1} (-1)^{(m+n)} A^{\ell}_{m_1,(-m)} \overline{B^{\ell}_{m_1,(-n)}} \, .
\end{aligned}
\tag{B.13}
$$

If $f^1 \in L_2(S_2)$ and $f^2 \in L_2(S_2)$, then the expansion of f^1 (and f^2, respectively) simplifies to (see Eq. (2.56) on page 54 and Eq. (2.57) on page 54)

$$
f^1(\mathfrak{g}) = \sum_{\ell,m,n} \tfrac{2\ell+1}{8\pi^2} A^{\ell}_{mn} [\mathbf{D}^{\ell}_{\mathfrak{g}}]_{mn} = \sum_{\ell,m,n} \tfrac{2\ell+1}{8\pi^2} \delta_{n,0} a^{\ell}_{m} [\mathbf{D}^{\ell}_{\mathfrak{g}}]_{mn} \, . \tag{B.14}
$$

Substituting the expansion coefficients in (B.12) and (B.13) according to (B.14) ($A^{\ell}_{mn} = \delta_{n,0} a^{\ell}_{m}$) proofs theorem 2.2.27.

Let $f^1, f^2, f^3 \in L_2(SO(3))$ be functions defined on the rotation group $SO(3)$. The triple-correlation is then defined by

$$
\begin{aligned}
C(\mathfrak{g}_1, \mathfrak{g}_2) &= \int_{SO(3)} f^1(\mathfrak{g}) \overline{f^2(\mathfrak{g}\mathfrak{g}_1) f^3(\mathfrak{g}\mathfrak{g}_2)} d\mathfrak{g} \\
&= \sum_{\substack{\ell_1, m_1, n_1 \\ \ell_2, m_2, n_2, i_2 \\ \ell_3, m_3, n_3, i_3}} \tfrac{2\ell_1+1}{8\pi^2} A^{\ell_1}_{m_1,n_1} \tfrac{2\ell_2+1}{8\pi^2} \overline{[\mathbf{D}^{\ell_2}_{\mathfrak{g}_1}]_{i_2,n_2} B^{\ell_2}_{m_2,n_2}} \tfrac{2\ell_3+1}{8\pi^2} \overline{[\mathbf{D}^{\ell_3}_{\mathfrak{g}_2}]_{i_3,n_3} C^{\ell_3}_{m_3,n_3}} \\
&\quad \times \int_{SO(3)} [\mathbf{D}^{\ell_1}_{\mathfrak{g}}]_{m_1,n_1} \overline{[\mathbf{D}^{\ell_2}_{\mathfrak{g}}]_{m_2,i_2} [\mathbf{D}^{\ell_3}_{\mathfrak{g}}]_{m_3,i_3}} d\mathfrak{g} \\
&= \sum_{\substack{\ell_1, m_1, n_1 \\ \ell_2, m_2, n_2, i_2 \\ \ell_3, m_3, n_3, i_3}} \tfrac{2\ell_1+1}{8\pi^2} A^{\ell_1}_{m_1,n_1} \tfrac{2\ell_2+1}{8\pi^2} \overline{[\mathbf{D}^{\ell_2}_{\mathfrak{g}_1}]_{i_2,n_2} B^{\ell_2}_{m_2,n_2}} \tfrac{2\ell_3+1}{8\pi^2} \overline{[\mathbf{D}^{\ell_3}_{\mathfrak{g}_2}]_{i_3,n_3} C^{\ell_3}_{m_3,n_3}} \\
&\quad \underbrace{\times \int_{SO(3)} [\mathbf{D}^{\ell_1}_{\mathfrak{g}}]_{m_1,n_1} \overline{[\mathbf{D}^{\ell_2}_{\mathfrak{g}}]_{m_2,i_2} [\mathbf{D}^{\ell_3}_{\mathfrak{g}}]_{m_3,i_3}} d\mathfrak{g}}_{\text{simplifies using Eq. (C.42) in the appendix}} \\
&= \sum_{\substack{\ell_1, m_1, n_1 \\ \ell_2, m_2, n_2, i_2 \\ \ell_3, m_3, n_3, i_3}} A^{\ell_1}_{m_1,n_1} \tfrac{2\ell_2+1}{8\pi^2} \overline{[\mathbf{D}^{\ell_2}_{\mathfrak{g}_1}]_{i_2,n_2} B^{\ell_2}_{m_2,n_2}} \tfrac{2\ell_3+1}{8\pi^2} \overline{[\mathbf{D}^{\ell_3}_{\mathfrak{g}_2}]_{i_3,n_3} C^{\ell_3}_{m_3,n_3}} \\
&\quad \times \langle \ell_2 m_2, \ell_3 m_3 \,|\, \ell_1 m_1 \rangle \langle \ell_2 i_2, \ell_3 i_3 \,|\, \ell_1 n_1 \rangle.
\end{aligned}
\tag{B.15}
$$

The Fourier correspondence of the triple-correlation is the bi-spectrum, given by

$$
\begin{aligned}
C^{\ell j}_{mnop} &= \iint\limits_{SO(3)} D(\mathfrak{g}_1, \mathfrak{g}_2, r)[\overline{\mathbf{D}^{\ell}_{\mathfrak{g}_1}}]_{mn} d\mathfrak{g}_1 [\overline{\mathbf{D}^{j}_{\mathfrak{g}_2}}]_{op} d\mathfrak{g}_2 \\
&= \sum_{\substack{\ell_1, m_1, n_1 \\ \ell_2, m_2, n_2, i_2 \\ \ell_3, m_3, n_3, i_3}} A^{\ell_1}_{m_1,n_1} \tfrac{2\ell_2+1}{8\pi^2} \overline{B^{\ell_2}_{m_2,n_2}} \tfrac{2\ell_3+1}{8\pi^2} \overline{C^{\ell_3}_{m_3,n_3}} \langle \ell_2 m_2, \ell_3 m_3 \,|\, \ell_1 m_1 \rangle \langle \ell_2 i_2, \ell_3 i_3 \,|\, \ell_1 n_1 \rangle \\
&\qquad \times \underbrace{\iint\limits_{SO(3)} \overline{[\mathbf{D}^{\ell_2}_{\mathfrak{g}_1}]}_{i_2,n_2} [\overline{\mathbf{D}^{\ell}_{\mathfrak{g}_1}}]_{mn} d\mathfrak{g}_1 \overline{[\mathbf{D}^{\ell_3}_{\mathfrak{g}_2}]}_{i_3,n_3} [\overline{\mathbf{D}^{j}_{\mathfrak{g}_2}}]_{op} d\mathfrak{g}_2}_{= (-1)^{(m+n+o+p)} \frac{8\pi^2}{2\ell+1} \frac{8\pi^2}{2j+1} \delta_{\ell_2,\ell} \delta_{i_2,(-m)} \delta_{n_2,(-n)} \delta_{\ell_3,j} \delta_{i_3,(-o)} \delta_{n_3,(-p)}} \\
&= \sum_{\substack{\ell_1, m_1, n_1 \\ m_2, m_3}} A^{\ell_1}_{m_1,n_1} \overline{B^{\ell}_{m_2,(-n)}} \overline{C^{j}_{m_3,(-p)}} \langle \ell m_2, j m_3 \,|\, \ell_1 m_1 \rangle \langle \ell(-m), j(-o) \,|\, \ell_1 n_1 \rangle \\
&= \sum_{\ell_1, m_2, m_3} A^{\ell_1}_{(m_2+m_3),(m+o)} \overline{B^{\ell}_{m_2,(-n)}} \overline{C^{j}_{m_3,(-p)}} \langle \ell m_2, j m_3 \,|\, \ell_1 (m_2+m_3) \rangle \langle \ell(-m), j(-o) \,|\, \ell_1 (-o-m) \rangle.
\end{aligned}
\tag{B.16}
$$

If $f^1, f^2, f^3 \in L_2(S_2)$ we can again simplify the bi-spectrum according to Eq. (B.14) and we get

$$
\begin{aligned}
&c^{\ell j}_{m} = \\
&\sum_{\ell_1, m_2, m_3} a^{\ell_1}_{(m_2+m_3)} \delta_{0,(m+o)} \overline{b^{\ell}_{m_2}} \delta_{0,n} \overline{c^{j}_{m_3}} \delta_{0,p} \langle \ell m_2, j m_3 \,|\, \ell_1 (m_2+m_3) \rangle \langle \ell(-m), j(-o) \,|\, \ell_1 - (m+o) \rangle \\
&= \delta_{0,n} \delta_{0,p} \sum_{\ell_1, m_2, m_3} \langle \ell m, j(-m) \,|\, \ell_1 0 \rangle a^{\ell_1}_{(m_2+m_3)} \overline{b^{\ell}_{m_2}} \overline{c^{j}_{m_3}} \langle \ell m_2, j m_3 \,|\, \ell_1 (m_2+m_3) \rangle \\
&= \sum_{\ell_1} \langle \ell m, j(-m) \,|\, \ell_1 0 \rangle \langle \mathbf{a}^{\ell_1}, (\mathbf{b}^{\ell} \circ_{\ell_1} \mathbf{c}^{j}) \rangle \quad ,
\end{aligned}
\tag{B.17}
$$

which is the bi-spectrum for functions on the 2-sphere according to theorem 2.2.28 on page 68. □

B.2.3.2. Tensorial Harmonic Functions

In Skibbe et al. (2009b) we have shown how to compute the expansion coefficients in two steps in terms of an ordinary spherical harmonic transform: the expansion coefficients $\mathbf{a}^{\ell}_{j}(\mathbf{x}, r) \in \mathbb{C}^{2(\ell+j)+1}$ can be computed by first projecting $\mathbf{f}^{J} = \{f^{J}_{-J}, \ldots, f^{J}_{M}, \cdots, f^{J}_{J}\}$ into the space spanned by spherical harmonics in a component-by-component manner. Finally, a weighted superposition forms the coefficients.

Theorem B.2.1. *Computation of tensorial harmonic expansion coefficients*

$$\begin{aligned} a^{\ell}_{jm}(\mathbf{x},r) &= \langle \mathfrak{h}_{(-\mathbf{x})}\mathbf{f}^J, \overline{\mathbf{Z}^{\ell jm}_J}\delta_r\rangle \\ &= \sum_{M=-J}^{J}\sum_{n=-\ell}^{\ell} \underbrace{\langle \mathfrak{h}_{(-\mathbf{x})} f^J_M, Y^{\ell}_n \delta_r\rangle}_{\textit{standard SH trafo}} \underbrace{\langle(\ell+j)m, \ell n | JM\rangle}_{\textit{weights}} \,. \end{aligned} \tag{B.18}$$

Proving Eq. (B.18) *on this page.* Let

$$\mathbf{b}^{\ell}_M(\mathbf{x},r) = \langle \mathfrak{h}_{(-\mathbf{x})} f^J_M, \mathbf{Y}^{\ell}\delta_r\rangle \quad , \tag{B.19}$$

where the $\mathbf{b}^{\ell}_M \in \mathcal{T}_\ell$ are the spherical harmonic coefficients of the contra-variant spherical harmonic expansion (compared to the standard spherical harmonic expansion, the spherical harmonics are complex conjugated). Combining Eq. (2.125) and Eq. (2.77) we obtain a system of equations which allow us to determine the relation between the tensorial harmonic coefficients $\mathbf{a}^{\ell}_j(\mathbf{x},r)$ and the spherical harmonic coefficients $\mathbf{b}^{\ell}_M(\mathbf{x},r)$:

$$\begin{aligned} \mathbf{f}^J(\mathbf{r}) &= \sum_{\ell=0}^{\infty}\sum_{j=-J}^{j=J} N_{\ell,j}\mathbf{a}^{\ell}_j(\mathbf{x},\|\mathbf{r}-\mathbf{x}\|) \circ_J \mathbf{Y}^{\ell}(\mathbf{r}-\mathbf{x}) \\ &= \sum_{\ell=0}^{\infty}\sum_{j=-J}^{j=J} N_{\ell,j} \sum_{m=-(\ell+j)}^{m=(\ell+j)}\sum_{n=-\ell}^{n=\ell} a^{\ell}_{jm}(\mathbf{x},\|\mathbf{r}-\mathbf{x}\|)\langle(\ell+j)m,\ell n|JM\rangle Y^{\ell}_n(\mathbf{r}-\mathbf{x}) \\ &= \sum_{\ell=0}^{\infty}\sum_{n=-\ell}^{n=\ell} Y^{\ell}_n(\mathbf{r}-\mathbf{x}) \underbrace{\sum_{j=-J}^{j=J} N_{\ell,j}\sum_{m=-(\ell+j)}^{m=(\ell+j)} a^{\ell}_{jm}(\mathbf{x},\|\mathbf{r}-\mathbf{x}\|)\langle(\ell+j)m,\ell n|JM\rangle}_{=(2\ell+1)b^{\ell}_{M,n}(\mathbf{x},\|\mathbf{r}-\mathbf{x}\|)} \\ &= \sum_{\ell=0}^{\infty}\sum_{n=-\ell}^{n=\ell}(2\ell+1)b^{\ell}_{M,n}(\mathbf{x},\|\mathbf{r}-\mathbf{x}\|)Y^{\ell}_n(\mathbf{x}-\mathbf{r}) \\ &= \sum_{\ell=0}^{\infty}(2\ell+1)\mathbf{b}^{\ell}_M(\mathbf{x},\|\mathbf{r}-\mathbf{x}\|)^T\mathbf{Y}^{\ell}(\mathbf{r}-\mathbf{x}) \end{aligned} \tag{B.20}$$

With use of Eq. (B.20) we can directly observe that

$$(2\ell+1)b^{\ell}_{M,n}(\mathbf{x},r) = \sum_{j=-J}^{j=J} N_{\ell,j}\sum_{m=-(\ell+j)}^{m=(\ell+j)} a^{\ell}_{jm}(\mathbf{x},r)\langle(\ell+j)m,\ell n|JM\rangle \tag{B.21}$$

Multiplying both sides with $\langle(\ell+j')m',\ell n|JM\rangle$ results in

$$\begin{aligned} &(2\ell+1)b^{\ell}_{M,n}(\mathbf{x},r)\langle(\ell+j')m',\ell n|JM\rangle \\ &\quad = \sum_{j=-J}^{j=J} N_{\ell,j}\sum_{m=-(\ell+j)}^{m=(\ell+j)} a^{\ell}_{jm}(\mathbf{x},r)\langle(\ell+j)m,\ell n|JM\rangle\langle(\ell+j')m',\ell n|JM\rangle \end{aligned} \tag{B.22}$$

Summarizing over all n and M leads to

$$
\begin{aligned}
&\sum_{M,n} b^{\ell}_{M,n}(\mathbf{x},r)\langle(\ell+j')m',\ell n|JM\rangle(2\ell+1)\\
&\quad=\sum_{M,n}\sum_{j=-J}^{j=J} N_{\ell,j}\sum_{m=-(\ell+j)}^{m=(\ell+j)} a^{\ell}_{jm}(\mathbf{x},r)\langle(\ell+j)m,\ell n|JM\rangle\langle(\ell+j')m',\ell n|JM\rangle\\
&\quad=\sum_{j=-J}^{j=J} N_{\ell,j}\sum_{m=-(\ell+j)}^{m=(\ell+j)} a^{\ell}_{jm}(\mathbf{x},r)\underbrace{\sum_{M,n}\langle(\ell+j)m,\ell n|JM\rangle\langle(\ell+j')m',\ell n|JM\rangle}_{\delta_{j,j'}\delta_{m,m'}\frac{2J+1}{2(\ell+j')+1}}
\end{aligned}
\tag{B.23}
$$

Due to the orthogonality of the Clebsch-Gordon coefficients (Eq. (C.33) in the appendix) all addends with $m \neq m'$ or $j \neq j'$ vanish:

$$
\begin{aligned}
\sum_{M,n} b^{\ell}_{M,n}(\mathbf{x},r)\langle(\ell+j)m,\ell n|JM\rangle &= N_{\ell,j}\frac{2J+1}{(2(\ell+j)+1)(2\ell+1)}a^{\ell}_{jm}(\mathbf{x},r)\\
&= a^{\ell}_{jm}(\mathbf{x},r).
\end{aligned}
\tag{B.24}
$$

□

B.3. $SE(3)$ Covariant Filters

B.3.1. An $SE(3)$ Covariant Gauss-Laguerre Transform

Proof: The polynomial degree of $\mathcal{L}^{\ell}_{n}(\mathbf{r})$ is $\ell+n$. Using the Cartesian representation of the spherical harmonics (Eq. (C.10) in the appendix) we get

$$
\begin{aligned}
[\mathcal{L}^{\ell}_{n}]_m(\mathbf{r},t) = &\underbrace{\sqrt{((\ell-n)-m)!((\ell-n)+m)!}\sum_{i=0}^{\ell}\sum_{j=0}^{\ell}\sum_{k=0}^{\ell}\frac{\delta_{i+j+k,(\ell-n)}\delta_{i-j,m}}{i!j!k!2^i2^j}(x-iy)^j(-x-iy)^i z^k}_{=\mathbf{Y}^{\ell-n}(\mathbf{r})r^{\ell-n}\text{ with polynomial degree }\ell-n\text{ (according to }\delta_{i+j+k,(\ell-n)})}\\
\times\quad &\underbrace{L_n^{(\ell-n)+\frac{1}{2}}(\tfrac{r^2}{2t})}_{\text{polynomial degree }2n}\quad .
\end{aligned}
\tag{B.25}
$$

Consequently, the polynomial degree of $\mathcal{L}^{\ell}_{n}$ is $\ell-n+2n=\ell+n$. □

Proof: Orthogonality of spherical Laguerre polynomials (Eq. (5.11) *on page 115).*

$$
\begin{aligned}
&\langle [\boldsymbol{\mathcal{L}}_n^{\ell}]_m, [\boldsymbol{\mathcal{L}}_{n'}^{\ell'}]_{m'} \rangle_\mu = \\
&= \int_{\mathbb{R}^3} Y_m^{\ell-n}(\mathbf{r}) r^{\ell-n} L_n^{(\ell-n)+\frac{1}{2}}(\tfrac{r^2}{2t}) \overline{Y_{m'}^{\ell'-n'}(\mathbf{r}) r^{\ell'-n'} L_{n'}^{(\ell'-n')+\frac{1}{2}}(\tfrac{r^2}{2t})} e^{-\frac{r^2}{2t}} d\mathbf{r} \\
&= \int r^{\ell-n} L_n^{(\ell-n)+\frac{1}{2}}(\tfrac{r^2}{2t}) r^{\ell'-n'} L_{n'}^{(\ell'-n')+\frac{1}{2}}(\tfrac{r^2}{2t}) r^2 e^{-\frac{r^2}{2t}} dr \iint \underbrace{Y_m^{\ell-n}(\mathbf{r}) \overline{Y_{m'}^{\ell'-n'}}(\mathbf{r}) d \sin\theta\theta d\varphi}_{=\frac{4\pi}{(2(\ell-n)+1)}\delta_{\ell-n,\ell'-n'}\delta_{m,m'}} \\
&= \frac{4\pi}{(2(\ell-n)+1)} \quad \underbrace{\int (r^2)^{(\ell-n)} L_n^{(\ell-n)+\frac{1}{2}}(\tfrac{r^2}{2t}) L_{n'}^{(\ell-n)+\frac{1}{2}}(\tfrac{r^2}{2t}) r^2 e^{-\frac{r^2}{2t}} dr}_{=2^{(\ell-n+1+\frac{1}{2})} t^{(\ell-n+1+\frac{1}{2})} \frac{\Gamma(\ell+\frac{1}{2}+1)}{2n!} \delta_{n,n'}, \text{(Eq. (C.24) in the appendix)}} \quad \delta_{\ell-n,\ell'-n'}\delta_{m,m'} \\
&= \frac{4\pi}{(2(\ell-n)+1)} 2^{(\ell-n+1+\frac{1}{2})} t^{(\ell-n+1+\frac{1}{2})} \frac{\Gamma(\ell+\frac{1}{2}+1)}{2n!} \delta_{\ell,\ell'}\delta_{m,m'}\delta_{n,n'} \\
&= \frac{4\pi}{(2(\ell-n)+1)} 2^{(\ell-n+1+\frac{1}{2})} t^{(\ell-n+1+\frac{1}{2})} \frac{(2\ell+1)!!\sqrt{\pi}}{2^{(}\ell+1)2n!} \delta_{\ell,\ell'}\delta_{m,m'}\delta_{n,n'} \\
&= t^{(\ell-n)} \frac{(2\pi t)^{\frac{3}{2}}(2\ell+1)!!}{2^n n!(2(\ell-n)+1)} \delta_{\ell,\ell'}\delta_{m,m'}\delta_{n,n'} \quad .
\end{aligned}
\tag{B.26}
$$

□

Auxiliary calculation for for Eq. (5.23) on page 120:

$$
\begin{aligned}
c_g(x,y) &= \tfrac{1}{\sigma_s\sqrt{\pi}} \int_{a,b,c,d} \underbrace{\delta(c-x)e^{-\frac{c^2}{2\sigma_w^2}}}_{\text{window}} \underbrace{e^{-\frac{(c-a)^2}{2\sigma_s^2}}}_{\text{smooth}} c_f(a,b) \underbrace{e^{-\frac{(d-b)^2}{2\sigma_s^2}}}_{\text{smooth}} \underbrace{e^{-\frac{d^2}{2\sigma_w^2}}\delta(d-y)}_{\text{window}} dadbdcde \\
&= \frac{1}{\sigma_s\sqrt{\pi}} \int_{a,b,c,d} \delta(c-x) e^{-\frac{c^2}{2\sigma_w^2}} e^{-\frac{(c-a)^2}{2\sigma_s^2}} \delta(a,b) e^{-\frac{(d-b)^2}{2\sigma_s^2}} e^{-\frac{d^2}{2\sigma_w^2}} \delta(d-y) dadbdcde \\
&= \frac{1}{\sigma_s\sqrt{\pi}} \int_a e^{-\frac{x^2}{2\sigma_w^2}} e^{-\frac{(x-a)^2}{2\sigma_s^2}} e^{-\frac{(y-a)^2}{2\sigma_s^2}} e^{-\frac{y^2}{2\sigma_w^2}} da \\
&= \frac{1}{\sigma_s\sqrt{\pi}} e^{-\frac{x^2}{2\sigma_w^2}} e^{-\frac{y^2}{2\sigma_w^2}} \int_a e^{-\frac{(x-a)^2}{2\sigma_s^2}} e^{-\frac{(y-a)^2}{2\sigma_s^2}} da \\
&= e^{-\frac{x^2}{2\sigma_w^2}} e^{-\frac{y^2}{2\sigma_w^2}} e^{-\frac{(x-y)^2}{4\sigma_s^2}} = e^{-\frac{x^2}{2\sigma_w^2}} e^{-\frac{(x-y)^2}{4\sigma_s^2}} e^{-\frac{y^2}{2\sigma_w^2}} \quad .
\end{aligned}
\tag{B.27}
$$

Auxiliary calculation for Eq. (5.24) on page 120.

$$\begin{aligned}
& \sum_{n=0}^{\infty}\frac{H_n(x)H_n(y)}{n!2^n}\lambda^n && =(1-\lambda^2)^{\frac{-1}{2}}e^{\frac{2xy\lambda-(x^2+y^2)\lambda^2}{1-\lambda^2}} && |\times e^{-\frac{(x^2+y^2)}{2}} \\
\Leftrightarrow\ & \sum_{n=0}^{\infty}\frac{H_n(x)H_n(y)}{n!2^n}\lambda^n e^{-\frac{(x^2+y^2)}{2}} && =(1-\lambda^2)^{\frac{-1}{2}}e^{\frac{2xy\lambda-(x^2+y^2)\lambda^2}{1-\lambda^2}}e^{-\frac{(x^2+y^2)}{2}} && \\
=\ & \sum_{n=0}^{\infty}\frac{H_n(x)H_n(y)}{n!2^n}\lambda^n e^{-\frac{(x^2+y^2)}{2}} && =(1-\lambda^2)^{\frac{-1}{2}}e^{-\frac{1-\lambda}{1+\lambda}\frac{(x+y)^2}{4}-\frac{1+\lambda}{1-\lambda}\frac{(x-y)^2}{4}} && |x\rightarrow(-x) \\
\Leftrightarrow\ & (-1)^n\sum_{n=0}^{\infty}\frac{H_n(-x)H_n(y)}{n!2^n}\lambda^n e^{-\frac{(x^2+y^2)}{2}} && =(1-\lambda^2)^{\frac{-1}{2}}e^{-\frac{1-\lambda}{1+\lambda}\frac{(y-x)^2}{4}-\frac{1+\lambda}{1-\lambda}\frac{(-x-y)^2}{4}} && |\lambda=\frac{t-1}{t+1} \\
\Leftrightarrow\ & (-1)^n\sum_{n=0}^{\infty}\frac{H_n(-x)H_n(y)}{n!2^n}(\tfrac{t-1}{t+1})^n e^{-\frac{(x^2+y^2)}{2}} && =\frac{(1+t)}{\sqrt{4t}}e^{\frac{-(1+t^2)x^2}{4t}\frac{-(t^2-1)2xy}{4t}\frac{-(1+t^2)y^2}{4t}} && \\
=\ & (-1)^n\sum_{n=0}^{\infty}\frac{H_n(-x)H_n(y)}{n!2^n}(\tfrac{t-1}{t+1})^n e^{-\frac{(x^2+y^2)}{2}} && =\frac{(1+t)}{\sqrt{4t}}e^{\frac{-x^2}{2t}}e^{\frac{-(t^2-1)(x+y)^2}{4t}}e^{\frac{-y^2}{2t}} && |x\rightarrow(-x) \\
\Leftrightarrow\ & \sum_{n=0}^{\infty}\frac{H_n(x)H_n(y)}{n!2^n}(\tfrac{t-1}{t+1})^n e^{-\frac{(x^2+y^2)}{2}} && =\frac{(1+t)}{\sqrt{4t}}e^{\frac{-x^2}{2t}}e^{\frac{-(t^2-1)(y-x)^2}{4t}}e^{\frac{-y^2}{2t}} && |\times\frac{1}{\sqrt{\pi}} \\
\Leftrightarrow\ & \sum_{n=0}^{\infty}\frac{H_n(x)H_n(y)}{\sqrt{\pi}n!2^n}(\tfrac{t-1}{t+1})^n e^{-\frac{(x^2+y^2)}{2}} && =\frac{(1+t)}{\sqrt{\pi 4t}}\underbrace{e^{\frac{-x^2}{2t}}e^{\frac{-(t^2-1)(y-x)^2}{4t}}e^{\frac{-y^2}{2t}}}_{=c(x,y)} && \\
\Leftrightarrow\ & \sum_{n=0}^{\infty}\frac{H_n(x)H_n(y)}{\sqrt{\pi}n!2^n}(\tfrac{t-1}{t+1})^n e^{-\frac{(x^2+y^2)}{2}} && =\frac{(1+t)}{\sqrt{\pi 4t}}c(x,y) && \qquad (B.28)
\end{aligned}$$

Auxiliary calculation for Eq. (5.36) on page 124.

$$\begin{aligned}
\langle \mathbf{Z}^{\ell jm}_{Jn}, \mathbf{Z}^{\ell' j' m'}_{Jn'}\rangle_\mu &= \sum_{M=-J}^{J}\int_{\mathbb{R}^3} Z^{\ell jm}_{JnM}(\mathbf{r})\overline{Z^{\ell' j' m'}_{Jn'M}(\mathbf{r})}d\mathbf{r} \\
&= \underbrace{\sum_{M=-J}^{J}\langle(\ell-n+j)m,(\ell-n)(M-m)\,|\,JM\rangle\langle(\ell'-n'+j')m',(\ell'-n')(M-m')\,|\,JM\rangle\times}_{\frac{2J+1}{2((\ell-n)+j)+1}\delta_{m,m'}\delta_{j,j'}} \\
&\quad \underbrace{\int_{\mathbb{R}^3}\overline{[\boldsymbol{\mathcal{L}}^{\ell}_{n}]_{(M-m)}(\mathbf{x})}[\boldsymbol{\mathcal{L}}^{\ell'}_{n'}]_{(M-m')}(\mathbf{x})e^{\frac{-r^2}{2t}}d\mathbf{r}}_{\frac{\delta_{\ell,\ell'}\delta_{m,m'}\delta_{n,n'}}{\alpha(\ell,n,t)}} \\
&= \frac{1}{A(\ell,j,n,t)}\delta_{\ell,\ell'}\delta_{j,j'}\delta_{m,m'}\delta_{n,n'}\ . \qquad (B.29)
\end{aligned}$$

B.3.1.1. An $SE(3)$ Covariant Gauss-Laguerre Transform for Tensor Fields

In the preliminary section we introduced a transformation into the tensorial harmonic domain via scalar valued spherical harmonic transformations. For this, the single components of a spherical tensor field were transformed into the harmonic domain in a component by component manner; see Eq. (B.18) on page 201. Similarly, we can compute the transformation of higher order spherical tensor fields into the Gauss-Laguerre domain $\mathcal{LT}^{\ell j}$ based on the Gauss-Laguerre transformation for scalar valued images $\mathcal{LT}^{\ell}$. Let $\mathbf{f}^J = \{f^J_{-J},, \cdots, f^J_M, \cdots, f^J_J\}$, $f^J_M : \mathbb{R}^3 \to \mathbb{C}$ be the single components of a spherical tensor field $\mathbf{f}^J \in \mathcal{T}_J$. Then, according to Eq. (B.18) on page 201,

$$\begin{aligned}
&[\mathcal{LT}^{\ell j}\{\mathbf{f}^J, n\}]_m(\mathbf{x}) = \langle \mathfrak{t}_{(-\mathbf{x})}\mathbf{f}^J, A^{\frac{1}{2}}(\ell, j, n, t)\overline{\mathbf{Z}^{\ell jm}_{Jn}}\rangle_\mu \qquad \text{(B.30)}\\
&= A^{\frac{1}{2}}(\ell, j, n, t) \sum_{M=-J}^{J} \sum_{N=-(\ell-n)}^{(\ell-n)} \underbrace{[\mathcal{LT}^{\ell}\{f^J_M, n\}]_N(\mathbf{x})}_{=\text{Trafo Eq. (5.34)}} \langle (\ell - n + j)m, (\ell - n)N \,|\, JM\rangle\\
&= (-1)^{\ell-n} A^{\frac{1}{2}}(\ell, j, n, t) \sum_{M,N} \langle (\ell - n + j)m, (\ell - n)N \,|\, JM\rangle \tfrac{(-t)^\ell}{n!2^n} \overline{\nabla}^{\ell-n} \Delta^n (f^J_M * e^{-\frac{r^2}{2t}})(\mathbf{x})\,.
\end{aligned}$$

Hence we can benefit from the same fast Gauss-Laguerre transform as for pure scalar valued images. It is worth mentioning, that for $J = 0$ the transformation coincides (up to a factor) with Eq. (5.34).

B.3.1.2. A Fast Gauss-Laguerre Transform via Tensor Derivatives

In the following we prove Eq. (5.41) on page 125 and Eq. (5.42) on page 127. Let

$$\mathbf{Z}^{\ell jm}_J(\mathbf{x}) := \mathbf{e}^{(\ell-n)+j}_m \circ_J \overline{\mathbf{B}^\ell(\mathbf{x})} \qquad \text{(B.31)}$$

be a tensorial harmonic basis, where $\mathbf{B}^\ell = \boldsymbol{\nabla}^\ell \mathbf{B}^0$ are spherical harmonic basis functions having a differential relationship. Let further be $\mathbf{B}^0 \in \mathcal{T}_0$. Then for the tensorial harmonic expansion coefficients $\mathbf{a}^\ell_J$ and $\mathbf{a}^\ell_{(J-2\ell)}$ it holds that

$$\mathbf{a}^\ell_J = \boldsymbol{\nabla}^\ell(\mathbf{f}^J \widetilde{\bullet}_J \mathbf{B}^0)\frac{(-1)^\ell\sqrt{(2\ell+1)}\langle \ell 0, J0 \,|\, (J+\ell)0\rangle}{\sqrt{N_{\ell,J}}} \qquad \text{(B.32)}$$

$$\mathbf{a}^\ell_{(J-2\ell)} = \boldsymbol{\nabla}_\ell(\mathbf{f}^J \widetilde{\bullet}_J \mathbf{B}^0)\frac{(-1)^\ell\sqrt{(2\ell+1)}\langle \ell 0, J0 \,|\, (J-\ell)0\rangle}{\sqrt{N_{\ell,J}}}\,. \qquad \text{(B.33)}$$

Proof. Since we have the differential relationship of the basis functions we can rewrite the projection onto the tensorial harmonics in terms of tensor derivatives:

$$
\begin{aligned}
a^{\ell}_{jm}(\mathbf{x}) &= \langle \tau_{\mathbf{x}}\mathbf{f}^J, \overline{\mathbf{Z}^{\ell jm}_J}\rangle \\
&= \int_{\mathbf{r}\in\mathbb{R}^3} (\mathbf{f}^J(\mathbf{r}-\mathbf{x}))^T \mathbf{Z}^{\ell jm}_J(\mathbf{r})d\mathbf{r} \\
&= \int_{\mathbf{r}\in\mathbb{R}^3} (\mathbf{f}^J(\mathbf{r}))^T \mathbf{Z}^{\ell jm}_J(\mathbf{r}+\mathbf{x})d\mathbf{r} \\
&= \int_{\mathbf{r}\in\mathbb{R}^3} \sum_{M=-J}^{M=J} f^J_M(\mathbf{r}) Z^{\ell jm}_{JM}(\mathbf{r}+\mathbf{x})d\mathbf{r} \\
&= \int_{\mathbf{r}\in\mathbb{R}^3} \sum_{M=-J}^{M=J} f^J_M(\mathbf{r}) \overline{Z^{\ell jm}_{J(-M)}(\mathbf{r}+\mathbf{x})}(-1)^M d\mathbf{r} \\
&= \tfrac{\sqrt{2J+1}}{(-1)^J}(\mathbf{f}^J \widetilde{\circ}_0 \overline{\mathbf{Z}^{\ell jm}_J})(\mathbf{x}) \\
&= \tfrac{\sqrt{2J+1}}{(-1)^J}(\mathbf{f}^J \widetilde{\circ}_0 \underbrace{(\mathbf{e}^{\ell+j}_m \circ_J \overline{\mathbf{B}^{\ell}})}_{=\mathbf{Z}^{\ell jm}_J})(\mathbf{x}) \\
&= \tfrac{\sqrt{2J+1}}{(-1)^J}((\overline{\mathbf{B}^{\ell}} \widetilde{\circ}_{\ell+j} \mathbf{f}^J) \circ_0 \mathbf{e}^{\ell+j}_m)(\mathbf{x}) \\
&= \tfrac{\sqrt{2J+1}}{(-1)^J}((\underbrace{(\overline{\partial}^{\ell} \circ_{\ell} \mathbf{B}^0)}_{=\overline{\mathbf{B}^{\ell}}} \widetilde{\circ}_{\ell+j} \mathbf{f}^J) \circ_0 \mathbf{e}^{\ell+j}_m)(\mathbf{x}) \\
&= \tfrac{\sqrt{2J+1}}{(-1)^J}((\overline{\partial}^{\ell} \circ_{\ell+j} (\mathbf{f}^J \widetilde{\bullet}_J \mathbf{B}^0)) \circ_0 \mathbf{e}^{\ell+j}_m)(\mathbf{x}) \\
&= \tfrac{\sqrt{2J+1}}{(-1)^J} \tfrac{(-1)^{\ell+j}}{\sqrt{2(\ell+j)+1}}((\overline{\partial}^{\ell} \circ_{\ell+j} (\mathbf{f}^J \widetilde{\bullet}_J \mathbf{B}^0)) \bullet_0 \mathbf{e}^{\ell+j}_m)(\mathbf{x}) \\
&= \tfrac{\sqrt{2J+1}}{(-1)^J} \tfrac{(-1)^{\ell+j}}{\sqrt{2(\ell+j)+1}}[(\partial^{\ell} \circ_{\ell+j} (\mathbf{f}^J \widetilde{\bullet}_J \mathbf{B}^0))]_m(\mathbf{x}) \\
&= \tfrac{\sqrt{2J+1}}{(-1)^J} \tfrac{(-1)^{\ell+j}}{\sqrt{2(\ell+j)+1}}[(\partial^{\ell} \circ_{\ell+j} (\mathbf{f}^J \widetilde{\bullet}_J \mathbf{B}^0))]_m(\mathbf{x}) \\
&= \tfrac{1}{\sqrt{N_{\ell,j}}}\sqrt{(2\ell+1)}(-1)^{J+\ell+j}[(\partial^{\ell} \circ_{\ell+j} (\mathbf{f}^J \widetilde{\bullet}_J \mathbf{B}^0))]_m(\mathbf{x}) \quad .
\end{aligned}
\tag{B.34}
$$

For the special case where $j = J$ we get

$$
\begin{aligned}
a^{\ell}_{Jm} &= \frac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}}(-1)^{\ell}[(\partial^{\ell} \circ_{J+\ell} (\mathbf{f}^J \widetilde{\bullet}_J \mathbf{B}^0))]_m \\
&= \frac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}}(-1)^{\ell}\langle \ell 0, J0 | (J+\ell)0\rangle[\boldsymbol{\nabla}^{\ell}(\mathbf{f}^J \widetilde{\bullet}_J \mathbf{B}^0)]_m \quad ,
\end{aligned}
\tag{B.35}
$$

and

$$
\begin{aligned}
a^{\ell}_{(J-2\ell)m} &= \frac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}}(-1)^{J+\ell+(J-2\ell)}[(\partial^{\ell} \circ_{\ell+(J-2\ell)} (\mathbf{f}^J \widetilde{\bullet}_J \mathbf{B}^0))]_m(\mathbf{x}) \\
&= \frac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}}(-1)^{\ell}[(\partial^{\ell} \circ_{J-\ell} (\mathbf{f}^J \widetilde{\bullet}_J \mathbf{B}^0))]_m(\mathbf{x}) \\
&= \frac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}}(-1)^{\ell}\langle \ell 0, J0 | (J-\ell)0\rangle[\boldsymbol{\nabla}_{\ell}(\mathbf{f}^J \widetilde{\bullet}_J \mathbf{B}^0)]_m \quad .
\end{aligned}
\tag{B.36}
$$

Alternatively, we can use Eq. (B.18) on page 201 and get

$$\begin{aligned}
a^{\ell}_{jm}(\mathbf{x}) &= \sum_{M=-J}^{M=J} \sum_{n=-\ell}^{n=\ell} \langle \boldsymbol{\tau}_{(-\mathbf{x})} f^{J}_{M}, B^{\ell}_{n} \rangle \langle (\ell+j)m', \ell n | JM \rangle \\
&= \sum_{M=-J}^{M=J} \sum_{n=-\ell}^{n=\ell} b^{\ell}_{M,n}(\mathbf{x}) \langle (\ell+j)m', \ell n | JM \rangle \\
&= \sum_{n=-\ell}^{n=\ell} b^{\ell}_{m+n,n}(\mathbf{x}) \langle (\ell+j)m, \ell n | J(m+n) \rangle \\
&= \sum_{n=-\ell}^{n=\ell} b^{\ell}_{m+n,n}(\mathbf{x}) \langle (\ell+j)m, \ell n | J(m+n) \rangle \\
&= \sqrt{\tfrac{2J+1}{2(\ell+j)+1}} \sum_{M=-J}^{M=J} \sum_{n=-\ell}^{n=\ell} (-1)^{\ell+n} b^{\ell}_{M,n}(\mathbf{x}) \langle \ell(-n), JM | (\ell+j)m \rangle \\
&= \frac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}} \textstyle\sum_{M=-J}^{M=J} \sum_{n=-\ell}^{n=\ell} (-1)^{\ell+n} b^{\ell}_{M,n}(\mathbf{x}) \langle \ell(-n), JM | (\ell+j)m \rangle \quad . \qquad \text{(B.37)}
\end{aligned}$$

Since $\mathbf{b}^{\ell}_{M} = \boldsymbol{\nabla}^{\ell}(f^{J}_{M} * \mathbf{B}^{0})$ we obtain

$$\begin{aligned}
a^{\ell}_{Jm} \quad &= \frac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}} (-1)^{\ell} \textstyle\sum_{n=-\ell}^{n=\ell} (-1)^{n} [\overline{\boldsymbol{\nabla}}^{\ell} \mathbf{b}^{0}_{m+n}]_{n} \langle \ell(-n), J(m+n) | (\ell+J)m \rangle \\
&= \frac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}} (-1)^{\ell} \textstyle\sum_{n=-\ell}^{n=\ell} (-1)^{n} [(\overline{\partial}^{\ell} \bullet_{\ell} \mathbf{b}^{0}_{m+n})]_{n} \langle \ell(-n), J(m+n) | (\ell+J)m \rangle \\
&= \frac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}} (-1)^{\ell} \textstyle\sum_{n=-\ell}^{n=\ell} (-1)^{n} \overline{\partial}^{\ell}_{n} b^{0}_{m+n} \langle \ell(-n), J(m+n) | (J+\ell)m \rangle \\
\text{(Eq. (4.6))} \quad &= \frac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}} (-1)^{\ell} \textstyle\sum_{n=-\ell}^{n=\ell} \partial^{\ell}_{-n} b^{0}_{m+n} \langle \ell(-n), J(m+n) | (J+\ell)m \rangle \\
&= \frac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}} (-1)^{\ell} \textstyle\sum_{n=-\ell}^{n=\ell} \partial^{\ell}_{n} b^{0}_{m-n} \langle \ell n, J(m-n) | (J+\ell)m \rangle \\
&= \frac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}} (-1)^{\ell} \textstyle\sum_{n=-\ell}^{n=\ell} \partial^{\ell}_{n} (f^{J}_{m-n} * \mathbf{B}^{0}) \langle \ell n, J(m-n) | (J+\ell)m \rangle \\
&= \frac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}} (-1)^{\ell} [(\partial^{\ell} \circ_{(J+\ell)} (\mathbf{f}^{J} \widetilde{\bullet}_{J} B^{0}))]_{m} \\
&= \frac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}} (-1)^{\ell} \langle \ell 0, J0 | (\ell+J)0 \rangle [\boldsymbol{\nabla}^{\ell} (\mathbf{f}^{J} \widetilde{\bullet}_{J} \mathbf{B}^{0})]_{m} \quad , \qquad \text{(B.38)}
\end{aligned}$$

and

$$\begin{aligned}
a^{\ell}_{(J-2\ell)m} \quad &= \tfrac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}}(-1)^{\ell}\textstyle\sum_{n=-\ell}^{n=\ell}(-1)^{n}[\overline{\boldsymbol{\nabla}}^{\ell}\mathbf{b}^{0}_{m+n}]_{n}\langle \ell(-n),J(m+n)\,|\,((J-2\ell)+\ell)m\rangle \\
&= \tfrac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}}(-1)^{\ell}\textstyle\sum_{n=-\ell}^{n=\ell}(-1)^{n}[(\overline{\partial}^{\ell}\bullet_{\ell}\mathbf{b}^{0}_{m+n})]_{n}\langle \ell(-n),J(m+n)\,|\,(J-\ell)m\rangle \\
&= \tfrac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}}(-1)^{\ell}\textstyle\sum_{n=-\ell}^{n=\ell}(-1)^{n}\overline{\partial}^{\ell}_{n}b^{0}_{m+n}\langle \ell(-n),J(m+n)\,|\,(J-\ell)m\rangle \\
\text{(Eq. (4.6))} \quad &= (-1)^{\ell}\sum_{n=-\ell}^{n=\ell}\partial^{\ell}_{-n}b^{0}_{m+n}\langle \ell(-n),J(m+n)\,|\,(J-\ell)m\rangle \\
&= \tfrac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}}(-1)^{\ell}\textstyle\sum_{n=-\ell}^{n=\ell}\partial^{\ell}_{n}b^{0}_{m-n}\langle \ell n,J(m-n)\,|\,(J-\ell)m\rangle \\
&= \tfrac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}}(-1)^{\ell}[(\partial^{\ell}\circ_{(J-\ell)}(\mathbf{f}^{J}\widetilde{\bullet}_{J}\mathbf{B}^{0}))]_{m} \\
&= \tfrac{\sqrt{(2\ell+1)}}{\sqrt{N_{\ell,J}}}(-1)^{\ell}\langle \ell 0,J0\,|\,(J-\ell)0\rangle[\boldsymbol{\nabla}_{\ell}(\mathbf{f}^{J}\widetilde{\bullet}_{J}\mathbf{B}^{0})]_{m} \quad . \qquad \text{(B.39)}
\end{aligned}$$

□

B.3.2. An $SE(3)$ Covariant Spherical Gabor Transform

Needed by Eq. (5.48) on page 133: the Fourier representation of the plane wave ω is

$$\begin{aligned}
&\mathcal{FT}\{\omega(\mathbf{k}')\}(\mathbf{k}) = (2\pi)^{-3/2}\langle\omega(\mathbf{k}'),e^{i\mathbf{k}^{T}\mathbf{r}}\rangle = \widetilde{\omega}(\mathbf{k},\mathbf{k}') = \\
&= \sum_{\substack{\ell_1,\ell_2,\\ m_1,m_2}}\int_{\mathbb{R}^3}(-i)^{\ell_1}(i)^{\ell_2}\tfrac{(2\ell_1+1)(2\ell_2+1)}{(2\pi)^{3/2}}j_{\ell_1}(kr)j_{\ell_2}(k'r)\overline{Y^{\ell_1}_{m_1}(\mathbf{r})}Y^{\ell_1}_{m_1}(\mathbf{k})Y^{\ell_2}_{m_2}(\mathbf{r})\overline{Y^{\ell_2}_{m_2}(\mathbf{k}')}d\mathbf{r} \\
&= (2\pi)^{-3/2}\sum_{\substack{\ell_1,\ell_2,\\ m_1,m_2}}\int_{r=0}^{\infty}(2\ell_1+1)(2\ell_2+1)j_{\ell_1}(kr)j_{\ell_2}(k'r)Y^{\ell_1}_{m_1}(\mathbf{k})\overline{Y^{\ell_2}_{m_2}(\mathbf{k}')} \\
&\qquad \times(-i)^{\ell_1}(i)^{\ell_2}\underbrace{\iint\overline{Y^{\ell_1}_{m_1}(\mathbf{r})}Y^{\ell_2}_{m_2}(\mathbf{r})\sin(\theta)d\theta d\varphi}_{=\frac{4\pi}{2\ell_1+1}\delta_{\ell_1,\ell_2}\delta_{m_1,m_2}}r^2dr \\
&= \tfrac{2}{\sqrt{\pi}}\sum_{\ell_1}\underbrace{(-i)^{\ell_1}(i)^{\ell_1}}_{=1}(2\ell_1+1)\underbrace{\int_{r=0}^{\infty}j_{\ell_1}(kr)j_{\ell_1}(k'r)r^2dr}_{=\frac{\pi}{2(k')^2}\delta(k-k')}\sum_{m_1}Y^{\ell_1}_{m_1}(\mathbf{k})\overline{Y^{\ell_1}_{m_1}(\mathbf{k}')} \\
&= \tfrac{t}{(k')^2}\delta(k-k')\sqrt{\pi}\underbrace{\sum_{\ell_1}(2\ell_1+1)\sum_{m_1}Y^{\ell_1}_{m_1}(\mathbf{k})\overline{Y^{\ell_1}_{m_1}(\mathbf{k}')}}_{=4\pi\delta^2_{\mathbf{k}}(\mathbf{k}'),\ \text{(Eq. (C.11))}} \\
&= (2\pi)^{\frac{3}{2}}\tfrac{1}{(k')^2}\delta(k-k')\delta^2_{(\mathbf{k}')}(\mathbf{k}) = (2\pi)^{\frac{3}{2}}\delta(\mathbf{k}-\mathbf{k}')\,. \qquad \text{(B.40)}
\end{aligned}$$

As we expected, the Fourier transform of the plane wave ω is just a delta impulse in the frequency domain (see Eq. (C.11) in the appendix).

The coefficients $\widetilde{\boldsymbol{\mathcal{B}}}^{\ell}(\mathbf{k},k') \in \mathbb{C}^{2\ell+1}$ are the Fourier representations of the Bessel functions $\boldsymbol{\mathcal{B}}^{\ell}(\mathbf{r},k')$:

$$
\begin{aligned}
&\mathcal{FT}\{\mathcal{B}^{\ell}_{m}(k')\}(\mathbf{k}) = \sum_{\ell_1,m_1}\int_{\mathbb{R}^3}(-i)^{\ell_1}\tfrac{(2\ell_1+1)}{(2\pi)^{3/2}}j_{\ell_1}(kr)j_{\ell}(k'\tfrac{r}{\sqrt{t}})\overline{Y^{\ell_1}_{m_1}(\mathbf{r})}Y^{\ell_1}_{m_1}(\mathbf{k})Y^{\ell}_{m}(\mathbf{r})d\mathbf{r}\\
&= (2\pi)^{-3/2}\sum_{\ell_1,m_1}\int_{r=0}^{\infty}(2\ell_1+1)j_{\ell_1}(kr)j_{\ell}(k'\tfrac{r}{\sqrt{t}})Y^{\ell_1}_{m_1}(\mathbf{k})\\
&\qquad\times(-i)^{\ell_1}\underbrace{\iint\overline{Y^{\ell_1}_{m_1}(\mathbf{r})}Y^{\ell}_{m}(\mathbf{r})\sin(\theta)d\theta d\varphi r^2 dr}_{=\frac{4\pi}{2\ell_1+1}\delta_{\ell_1,\ell}\delta_{m_1,m}}\\
&= \tfrac{2}{\sqrt{\pi}}(-i)^{\ell}\underbrace{\int_{r=0}^{\infty}j_{\ell}(kr)j_{\ell}(k'\tfrac{r}{\sqrt{t}})r^2 dr}_{=\frac{t\pi}{2(k')^2}\delta(k-\frac{k'}{\sqrt{t}})}Y^{\ell}_{m}(\mathbf{k}) = (-i)^{\ell}\tfrac{t\sqrt{\pi}}{(k')^2}\delta(k-\frac{k'}{\sqrt{t}})Y^{\ell}_{m}(\mathbf{k}) = \widetilde{\boldsymbol{\mathcal{B}}}^{\ell}(\mathbf{k},k') \quad .
\end{aligned}
\tag{B.41}
$$

Similar to the spherical representation in spatial domain, $\widetilde{\boldsymbol{\mathcal{B}}}^{\ell}(\mathbf{k},k')$ is independent the direction of the frequency vector $\mathbf{n} = \frac{\mathbf{k}'}{\|\mathbf{k}'\|}$ of $\mathbf{k}'$. Accordingly, the steering of the direction of the wave in frequency domain depends only on the term $\mathbf{Y}^{\ell}(\mathbf{k})$.

B.3.2.1. Gaussian Smoothed Sphere

The Fourier correspondence of a spherical Bessel function of order 0 and frequency k is a sphere in frequency domain with radius k; see Eq. (5.48) on page 133. Hence a Gaussian smoothed sphere is the Fourier representation of the Gaussian windowed Bessel function. Therefore, we get the explicit formula for the smoothed sphere by transforming the Gaussian windowed Bessel function into the Fourier domain.

$$
\begin{aligned}
&\mathcal{FT}\{\boldsymbol{\mathcal{B}}^{0}_{s}(k')\}(\mathbf{k}) = \int_{\mathbb{R}^3}e^{i\mathbf{k}^T\mathbf{r}}\boldsymbol{\mathcal{B}}^{0}_{s}(\mathbf{r},k')e^{-\frac{\|\mathbf{r}\|^2}{2st}}d\mathbf{r} = \int_{\mathbb{R}^3}e^{i\mathbf{k}^T\mathbf{r}}j_0(\frac{k'\|\mathbf{r}\|}{\sqrt{t}})e^{-\frac{\|\mathbf{r}\|^2}{2st}}d\mathbf{r} =\\
&= \iiint e^{ikr\cos\theta}j_0(\tfrac{k'r}{\sqrt{t}})e^{-\frac{r^2}{2st}}\sin(\theta)r^2 d\theta d\varphi dr\\
&= \iint\underbrace{\int_{\theta=0}^{\pi}e^{ikr\cos\theta}\sin(\theta)d\theta}_{=\frac{2\sin(kr)}{kr}}\underbrace{j_0(\tfrac{k'r}{\sqrt{t}})}_{=\frac{\sin(\frac{k'r}{\sqrt{t}})}{\frac{k'r}{\sqrt{t}}}}e^{-\frac{r^2}{2st}}r^2 d\varphi dr\\
&= \frac{2\sqrt{t}}{k'}\int\tfrac{\sin(kr)}{k}\sin(\tfrac{k'r}{\sqrt{t}})e^{-\frac{r^2}{2st}}\underbrace{\int_{\varphi=0}^{2\pi}d\varphi}_{=2\pi^2}dr = \frac{4\pi^2\sqrt{t}}{k'}\int\tfrac{\sin(kr)}{k}\sin(\tfrac{k'r}{\sqrt{t}})e^{-\frac{r^2}{2st}}dr\\
&= \tfrac{4\pi^2\sqrt{t}}{k'}\tfrac{(e^{-(k-k'\sqrt{t})^2 s/2}-e^{-(k+k'\sqrt{t})^2 s/2})}{2k}\sqrt{\tfrac{\pi}{2}}\sqrt{s}\sqrt{t}\\
&= \tfrac{\sqrt{2}\pi^{\frac{5}{2}}t\sqrt{s}}{kk'}(e^{-(k-k'\sqrt{t})^2 s/2}-e^{-(k+k'\sqrt{t})^2 s/2}) \quad .
\end{aligned}
\tag{B.42}
$$

B.3.3. SHOG - Spherical Histograms of Oriented Gradients

Proof: $\mathbf{v}_w^\ell$ *is a spherical tensor.* It is sufficient to show that the coefficients are rotating according to Eq. (2.84) on page 62.

Assume a 3D rotation denoted by $\mathfrak{g} \in SO(3)$ is acting on a volumetric image $I \in L_2(\mathbb{R}^3)$ so that $(\mathfrak{g}I)(\mathbf{x}) := I(\mathbf{U}_\mathfrak{g}^T\mathbf{x})$. We denote the gradient image of I by $\mathbf{g} : \mathbb{R}^3 \to \mathbb{C}^3$, $\mathbf{g} = \nabla I$. Let $\mathrm{SHOG}_w(I)$ be the SHOG feature representing I. Then $\mathrm{SHOG}_w(\mathfrak{g}I)$, the SHOG representation of $\mathfrak{g}I$ can be determined quite easily by considering the fact that the gradient directions are rotating, too; namely $(\mathfrak{g}\mathbf{g})(\mathbf{x}) := \mathbf{U}_\mathfrak{g}\mathbf{g}(\mathbf{U}_\mathfrak{g}^T\mathbf{x})$; see Eq. (2.45). Thus we can identify the new expansion coefficients of $\mathrm{SHOG}(\mathfrak{g}I)_w$ to be:

$$
\begin{aligned}
&\mathrm{SHOG}_w(\mathfrak{g}I,\mathbf{n}) = \\
&= \int \underbrace{\|\mathbf{U}_\mathfrak{g}\mathbf{g}(\mathbf{r})\|^\gamma}_{=\|\mathbf{g}(\mathbf{r})\|^\gamma} \delta_\mathbf{n}^2(\mathbf{U}_\mathfrak{g}\hat{\mathbf{g}}(\mathbf{r}))w(\mathbf{r})d\mathbf{r} \\
&= \int \|\mathbf{g}(\mathbf{r})\|^\gamma \underbrace{\sum_{\ell=0}^{\infty}(2\ell+1)\left(\mathbf{Y}^\ell(\mathbf{U}_\mathfrak{g}\hat{\mathbf{g}}(\mathbf{r}))\right)^T \overline{\mathbf{Y}^\ell(\mathbf{n})}}_{=\delta_\mathbf{n}^2;\text{ see Eq. (C.11) in the appendix}} w(\mathbf{r})d\mathbf{r} \\
&= \sum_{\ell=0}^{\infty}(2\ell+1)\int \|\mathbf{g}(\mathbf{r})\|^\gamma \underbrace{\left(\mathbf{Y}^\ell(\mathbf{U}_\mathfrak{g}\hat{\mathbf{g}}(\mathbf{r}))\right)^T}_{=\mathbf{D}_\mathfrak{g}^\ell\mathbf{Y}^\ell(\hat{\mathbf{g}})} w(\mathbf{r})d\mathbf{r}\overline{\mathbf{Y}^\ell(\mathbf{n})} \\
&= \sum_{\ell=0}^{\infty}(2\ell+1)\underbrace{\int \|\mathbf{g}(\mathbf{r})\|^\gamma \left(\mathbf{D}_\mathfrak{g}^\ell\mathbf{Y}^\ell(\hat{\mathbf{g}}(\mathbf{r}))\right)^T w(\mathbf{r})d\mathbf{r}}_{=\mathbf{D}_\mathfrak{g}^\ell{\mathbf{v}^\ell}^T}\overline{\mathbf{Y}^\ell(\mathbf{n})} \\
&= \underbrace{\sum_{\ell=0}^{\infty}(2\ell+1)(\mathbf{D}_\mathfrak{g}^\ell\mathbf{v}_w^\ell)^T\overline{\mathbf{Y}^\ell(\mathbf{n})}}_{=\text{expansion in Eq. (5.63)}} = \sum_{\ell=0}^{\infty}(2\ell+1)(\mathbf{v}_w^\ell)^T\overline{((\mathbf{D}_\mathfrak{g}^\ell)^*\mathbf{Y}^\ell(\mathbf{n}))} \\
&= \sum_{\ell=0}^{\infty}(2\ell+1)(\mathbf{v}_w^\ell)^T(\overline{\mathbf{Y}^\ell(\mathbf{U}_\mathfrak{g}^T\mathbf{n})}) = \mathrm{SHOG}_w(I,\mathbf{U}_\mathfrak{g}^T\mathbf{n})\,.
\end{aligned}
$$

□

C Functions and Formulas

C.1. Symbols

Symbol	Description	Page(s)
$\mathcal{FT}$	The Fourier transformation	39
$\widetilde{f}$	The Fourier representation of f	39
$\mathfrak{g} \in G$	A group element. Mostly a rotation $\mathfrak{g} \in SO(n)$	49
$\mathfrak{h}_{\mathbf{t}} \in T(n)$	Denotes a translation by $\mathbf{t}$	39
$\mathbf{D}^\ell$	Wigner D-Matrix	53
Y^ℓ_m	Spherical harmonic function	55
$\mathbf{Y}^\ell$	Vector representation of spherical harmonic functions	55
$\mathbf{R}^\ell$	Vector representation of solid harmonic functions	77
$\mathbf{Z}^{\ell j m}_J$	Tensorial harmonic functions	72
$\left(\mathbf{v}^\ell\right)^\ddagger$	The spherical tensor conjugation	66
$\partial^\ell, \boldsymbol{\nabla}^\ell_n$	Spherical derivatives	97
$\circ_\ell, \bullet_\ell$	Spherical tensor products	64
$\widetilde{\circ}_\ell, \widetilde{\bullet}_\ell$	Spherical tensor convolution	65
$\mathcal{SH}, \mathcal{LT}, \mathcal{GT}, \mathcal{GT}$	Dense spherical harmonic transformations	57,109

C.2. Preliminaries

Definition C.2.1 (Vector Space). *A nonempty set $\mathcal{X}$ is called a **vector space** over a field $\mathbb{F}$, if for all $\mathbf{x}, \mathbf{y} \in X$ and for all $\alpha \in \mathbb{F}$, there exist a sum $\mathbf{x} + \mathbf{y} \in X$ and a product $\alpha\mathbf{x} \in X$*

fulfilling the following vector space axioms:

- $\forall \mathbf{x},\mathbf{y},\mathbf{z} \in X : \mathbf{x} + (\mathbf{y} + \mathbf{z}) = (\mathbf{x} + \mathbf{y}) + \mathbf{z}$ *(associativity)*
- $\forall \mathbf{x},\mathbf{y} \in X : \mathbf{x} + \mathbf{y} = \mathbf{y} + \mathbf{x}$ *(commutativity)*
- $\exists \mathbf{0} \in X : \forall \mathbf{x} \in X : \mathbf{x} + \mathbf{0} = \mathbf{x}$ *(identity element of addition, the zero vector)*
- $\forall \mathbf{x} \in X : \exists (-\mathbf{x}) \in X : \mathbf{x} + (-\mathbf{x}) = \mathbf{0}$ *(additive inverse)*
- $\forall \mathbf{x}, \forall \alpha, \beta \in \mathbb{F} : \alpha(\beta \mathbf{x}) = (\alpha\beta)\mathbf{x}$ *(associativity)*
- $\forall \mathbf{x},\mathbf{y}, \forall \alpha \in \mathbb{F} : \alpha(\mathbf{x} + \mathbf{y}) = \alpha\mathbf{x} + \alpha\mathbf{y}$ *(distributivity I)*
- $\forall \mathbf{x}, \forall \alpha, \beta \in \mathbb{F} : (\alpha + \beta)\mathbf{x} = \alpha\mathbf{x} + \beta\mathbf{x}$ *(distributivity II)*
- $\exists 1 \in \mathbb{F} : \forall \mathbf{x} \in \mathbb{F} : 1\mathbf{x} = \mathbf{x}$ *(multiplicative identity)*

Definition C.2.2 (Vector Sub Space)**.** *Let X be a vector space over the field $\mathbb{F}$. A proper (nonempty) subset $Y \subset X$ of X is called sub space of X, iff*

$$\forall \alpha, \beta \in \mathbb{F} : \mathbf{x},\mathbf{y} \in Y \Rightarrow \alpha\mathbf{x} + \beta\mathbf{y} \in Y \quad . \tag{C.1}$$

Definition C.2.3 (Inner Product Space)**.** *Let X be a vector space X over the field $\mathbb{F}$. Let $\mathbb{F}$ be either the complex numbers $\mathbb{C}$ or the real numbers $\mathbb{R}$. X is called an inner product space, if there exists a Hermitian form $\langle\cdot,\cdot\rangle : X \times X \to \mathbb{F}$ fulfilling the following properties*

- $\forall \mathbf{x},\mathbf{y} \in X : \langle \mathbf{x},\mathbf{y}\rangle = \overline{\langle \mathbf{y},\mathbf{x}\rangle}$ *(conjugate symmetry, Hermitian (if $\mathbb{R}$, symmetric))*
- $\forall \mathbf{x},\mathbf{y},\mathbf{z} \in X : \langle \mathbf{x},\mathbf{y} + \mathbf{z}\rangle = \langle \mathbf{x},\mathbf{y}\rangle + \langle \mathbf{x},\mathbf{z}\rangle$
  $\forall \mathbf{x},\mathbf{y} \in X, \forall \alpha \in \mathbb{F} : \langle \alpha\mathbf{x},\mathbf{y}\rangle = \alpha\langle \mathbf{x},\mathbf{y}\rangle$ *(linearity in the first argument)*
  $\Rightarrow \forall \mathbf{x},\mathbf{y} \in X, \forall \alpha \in \mathbb{F} : \langle \mathbf{x},\alpha\mathbf{y}\rangle = \overline{\alpha}\langle \mathbf{x},\mathbf{y}\rangle = \langle \overline{\alpha}\mathbf{x},\mathbf{y}\rangle$
- $\forall (\mathbf{x} \in X : \langle \mathbf{x},\mathbf{x}\rangle \geq 0) \quad \wedge \quad (\langle \mathbf{x},\mathbf{x}\rangle = 0 \Leftrightarrow \mathbf{x} = 0)$ *(positive definiteness)*

The inner product induces a norm $||\cdot|| : X \to \mathbb{R}$ together with a distance measure

- $||\mathbf{x}|| = \sqrt{\langle \mathbf{x},\mathbf{x}\rangle}$ *(the norm is induced by the inner product)*
- $d(\mathbf{x},\mathbf{y}) = ||\mathbf{x} - \mathbf{y}||$ *(distance between two points in the vector space)*

The vector space X together with $||\cdot||$ is called a normed space. The norm is positive definite (see Def. of the inner product), it also holds that $\forall \mathbf{x} \in X, \alpha \in \mathbb{F} : ||\alpha\mathbf{x}|| = |\alpha|||\mathbf{x}||$ and it obeys the triangle inequality $\forall \mathbf{x},\mathbf{y} \in X : ||\mathbf{x} + \mathbf{y}|| \leq ||\mathbf{x}|| + ||\mathbf{y}||$.

Definition C.2.4 (Hilbert Space)**.** *A complete normed inner product space X is called a Hilbert space. A normed inner product space X is complete, if every Cauchy sequence converges with respect to the norm of X.*

Example C.2.5 (n-Dimensional Vector Space over $\mathbb{R}$ or $\mathbb{C}$)**.** *Let X be the set of all n-tuples $\mathbf{x} \in \mathbb{R}^n$ (or $\mathbb{C}^n$, respectively), where $n \in \mathbb{N}_{>0}$. An element of X can be written as vector*

$$\mathbf{x} = (x_1, x_2, \cdots, x_n)^T,$$

where $x_i \in \mathbb{R}$ *(or* $\mathbb{C}$*, respectively). The vector space addition and multiplication are defined in an element by element manner:*

$$\begin{aligned}\mathbf{x}+\mathbf{y} &= (x_1+y_1, x_2+y_2, \cdots, x_n+y_n)^T, \\ \alpha\mathbf{x} &= (\alpha x_1, \alpha x_2, \cdots, \alpha x_n)^T.\end{aligned}$$

The inner product is defined by

$$\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T \overline{\mathbf{y}} = \sum_{i=1}^{n} x_i \overline{y_i} \quad . \tag{C.2}$$

The norm and the distance measure are defined in terms of the inner product. If the vector space is over $\mathbb{R}$*, then it is also called the Euclidean space.*

Definition C.2.6 (The n-Dimensional Local Linear (compact) Lie Groups)**.** *Let* W *be an open connected set containing the zero tuple* $\mathbf{e} = (0, \cdots, 0)$ *in the space* F_n *of all n-tuples* $\mathbf{g} = (g_1, \cdots, g_n)$*. The tuples* $\mathbf{g}$ *can be real or complex valued. Given a set* V *build upon* $m \times m$ *non-singular matrices* $\mathbf{A}(\mathbf{g}) = \mathbf{A}(g_1, \cdots, g_n)$ *defined for each* $\mathbf{g} \in W$*. We call such a set an n-dimensional local linear Lie algebra, if*

1. $\mathbf{A}(\mathbf{e}) = \mathcal{I}$ *(the matrix identity)*
2. *The matrix elements of* $\mathbf{A}(\mathbf{g})$ *are analytic functions of the parameters* $g_1, \cdots, g_n$ *and the map* $\mathbf{g} \to \mathbf{A}(\mathbf{g})$ *is injective.*
3. *The n derivatives of* $\mathbf{A}(\mathbf{g})$*, which are the matrices* $\frac{\partial \mathbf{A}(\mathbf{g})}{\partial g_j}$*,* $j = 1, \cdots, n$*, are linearly independent for each* $\mathbf{g} \in W$*. As a result, these matrices span an n-dimensional subspace of the* m^2*-dimensional space of all* $m \times m$ *matrices.*
4. *There exists a neighborhood* w' *of* $\mathbf{e}$ *in* F_n*,* $W' \subset W$*, with the property that for every pair of n-tuples* $\mathbf{g}, \mathbf{h} \in W'$ *there exists an n-tuple* $\mathbf{k}$ *in* W *satisfying the matrix product* $\mathbf{A}(\mathbf{g})\mathbf{A}(\mathbf{h}) = \mathbf{A}(\mathbf{k})$*.*

If V *is bounded and closed, then* V *is called* ***compact****. (Bounded means that* $\exists k \in \mathbb{R}_{>0} : \forall \mathbf{A}(\mathbf{g}) \in V, 1 \leq i,j \leq m : |A(\mathbf{g})_{i,j}| \leq k$*. Closed means that each Cauchy sequence in* V *converges to a matrix in* V*, (also in a matrix component by component manner))*

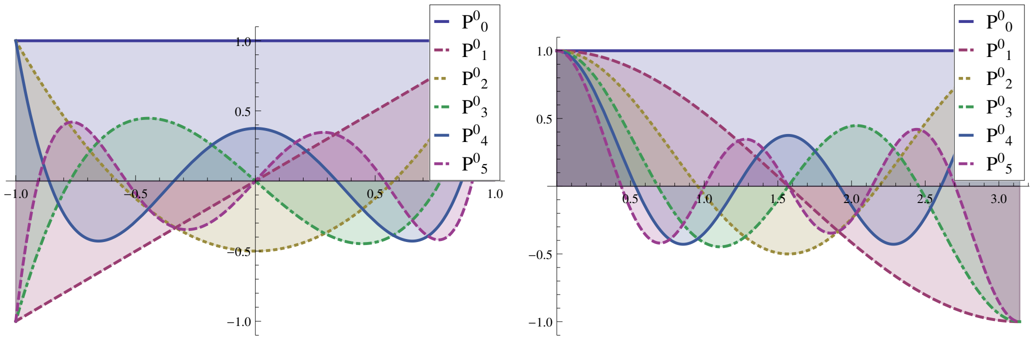
For further details we refer to the lecture notes of Miller (1991), on which this definition is based on.

C.3. Irreducible Representation of $SO(3)$

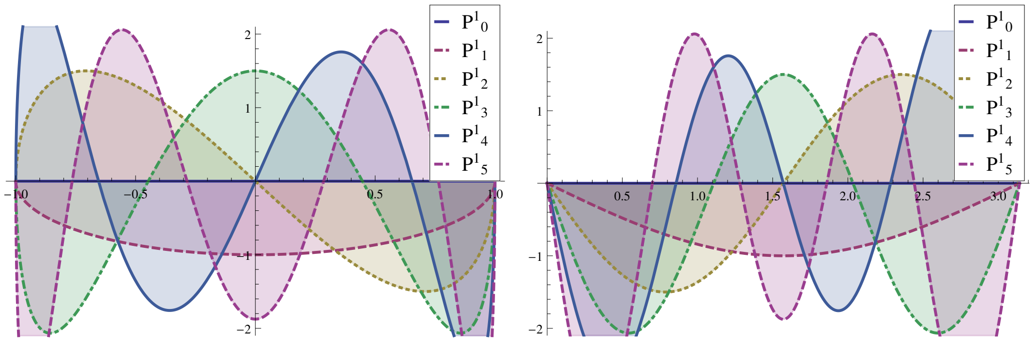
C.3.1. Volume of $SO(3)$ and Surface Area of S_2

$$\text{Volume}(SO(3)) = \int_{SO(3)} d\mathfrak{g} = \underbrace{\int_{\varphi=0}^{2\pi} d\varphi}_{=2\pi} \underbrace{\int_{\theta=0}^{\pi} \sin(\theta) d\theta}_{=2} \underbrace{\int_{\psi=0}^{2\pi} d\psi}_{=2\pi} = 8\pi^2 \quad . \tag{C.3}$$

$$\text{Area}(S_2) = \int_{S_2} d\mathfrak{g} = \underbrace{\int_{\varphi=0}^{2\pi} d\varphi}_{=2\pi} \underbrace{\int_{\theta=0}^{\pi} \sin(\theta) d\theta}_{=2} = 4\pi \quad . \tag{C.4}$$



(a) $P_\ell^m(x)$ for $\ell = 0, \cdots, 5$ and $m = 0$ (b) $P_\ell^m(\cos(x))$



(c) $P_\ell^m(x)$ for $\ell = 0, \cdots, 5$ and $m = 1$ (d) $P_\ell^m(\cos(x))$

Figure C.1.: Associated Legendre Polynomials for $\ell = 0, \cdots, 5$ and $m = 0, 1$.

C.3.2. Legendre Polynomials

For details see e.g. Abramowitz and Stegun (1964).

Definition C.3.1 (Legendre Polynomials)**.** *The Legendre polynomials are defined by*

$$P_n(x) := \frac{1}{2^n} \sum_{k=0}^{n} \binom{n}{k}^2 (x-1)^{n-k} (x+1)^k \quad , \tag{C.5}$$

where $n \geq 0$.

Definition C.3.2 (Associated Legendre Polynomials)**.** *The associated Legendre polynomials are defined by*

$$P_\ell^m(x) := (-1)^m (1-x^2)^{m/2} \frac{\partial^m}{\partial x^m} P_\ell(x) \quad , \tag{C.6}$$

where P_ℓ *are the ordinary Legendre polynomials and* $m \geq 0$. *We have depicted some Legendre polynomials in Fig. C.1. The Legendre polynomials for* $m < 0$ *are then defined by*

$$P_\ell^{(-m)}(x) = (-1)^m \frac{(l-m)!}{(l+m)!} P_\ell^m(x) \quad . \tag{C.7}$$

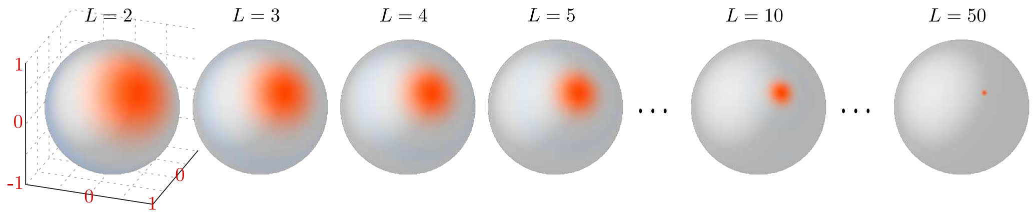


Figure C.2.: Band limited expansion of the Dirac delta $\delta^2_{\mathbf{n}} : S_2 \to \mathbb{R}$ on the unit-sphere. A limit of $\ell \leq L$ determines the granularity of the binning of our representation of the spherical histograms of oriented gradients (SHOG, see chapter 5.3 on page 138).

For $0 \leq m \leq \min(\ell, \ell')$ *the associated Legendre polynomials are orthogonal according to*

$$\int_{x=-1}^{1} P^m_\ell(x) P^m_{\ell'}(x) dx = \frac{2(\ell+m)!}{(2\ell+1)!(\ell-m)!} \delta_{\ell'_\ell} \quad . \tag{C.8}$$

The parity of Legendre polynomials is given by

$$P^m_\ell(x) = (-1)^{\ell-m} P^m_\ell(-x) \quad . \tag{C.9}$$

C.3.3. Cartesian representation of Spherical Harmonics

For $\mathbf{r} = (x, y, z)^T$ the spherical harmonics are defined by

$$Y^\ell_m(\mathbf{r}) = \frac{\sqrt{(\ell-m)!(\ell+m)!}}{(\sqrt{x^2+y^2+z^2})^\ell} \sum_{i=0}^{\ell} \sum_{j=0}^{\ell} \sum_{k=0}^{\ell} \frac{\delta_{i+j+k,\ell} \delta_{i-j,m}}{i!j!k!2^i 2^j} (x - iy)^j (-x - iy)^i z^k \quad . \tag{C.10}$$

(The Wolfram Functions Site, `http://functions.wolfram.com/05.10.06.0027.01`)

C.3.4. Spherical Expansion of the Dirac Delta Function

Let $\mathbf{n}, \mathbf{n}' \in S_2$ be two unit-length vectors representing points on the 2-sphere. We denote by $\delta^2_{\mathbf{n}} : S_2 \to \mathbb{R}$ the delta function on the 2-sphere, whereas $\delta^2_{\mathbf{n}}(\mathbf{n}') = \delta(\theta - \theta')\delta(\varphi - \varphi')$ and $\int_{S_2} \delta^2_{\mathbf{n}}(\mathbf{n}') d\mathbf{n}' = 1$. According to Arfken and Weber (2005, p. 792) we have the following expansion of $\delta^2_{\mathbf{n}}$ in terms of spherical harmonics:

$$\delta^2_{\mathbf{n}}(\mathbf{n}') := \sum_{\ell=0}^{\infty} \tfrac{(2\ell+1)}{4\pi} \left(\overline{\mathbf{Y}^\ell}(\mathbf{n}')\right)^T \mathbf{Y}^\ell(\mathbf{n}) \quad . \tag{C.11}$$

A limit of $\ell \leq L$ leads to the best-approximation of $\delta^2_{\mathbf{n}}$ in a finite subspace spanned by $Y^\ell_m, \ell \leq L$ (see figure C.2).

C.3.5. Spherical Bessel Functions

Definition C.3.3 (The spherical Bessel functions)**.** *The spherical Bessel functions* $j_\ell : \mathbb{R}_{\geq 0} \to \mathbb{R}$ *are related to the Bessel functions of the first kind* J_ν *(For further details see. e.g.*

Abramowitz and Stegun (1964)) by $j_\ell(r) := \sqrt{\frac{\pi}{2r}} J_{\ell+1/2}(r)$ *and can be expanded by*

$$j_\ell(r) = r^\ell \sum_{m=0}^{\infty} \frac{(-1)^m}{2^m m!(2(\ell+m)+1)!!} r^{2m}, \tag{C.12}$$

where

$$\int_0^\infty j_\ell(kr) j_\ell(k'r) r^2 dr = \frac{\pi}{2k^2} \delta(k-k') \quad . \tag{C.13}$$

In Fig. C.3 on page 219 you can find plots of the Bessel functions for order $\ell = 0, 1, 2$. For the spherical Bessel functions we have the following differential relations:

$$\tfrac{\partial}{\partial r}\left[r^{-\nu} j_\nu\right] = -r^{-\nu} j_{\nu+1} \qquad \text{and} \tag{C.14}$$

$$\tfrac{\partial}{\partial r}\left[r^{\nu+1} j_\nu\right] = r^{\nu+1} j_{\nu-1} \quad . \tag{C.15}$$

The Hankel Transform (also known as Fourier-Bessel transform, Bronshtein and Semendyayev (1997)) of order ℓ in terms of the spherical Bessel functions is given by

$$\alpha_\ell(k) = \sqrt{\frac{2}{\pi}} \int_0^\infty f(r) j_\ell(kr) r^2 dr \, , \tag{C.16}$$

and its corresponding inverse transformation is given by

$$f(r) = \sqrt{\frac{2}{\pi}} \int_0^\infty \alpha_\ell(k) j_\ell(rk) k^2 dk \, , \tag{C.17}$$

which both follows from Eq. (C.13).

C.3.6. The Plane Wave

Using the addition theorem of the spherical harmonics we can express the spherical expansion of the plane wave (see e.g Rose (1995), p. 136) in terms of the tensor product $\bullet_0$ leading to:

$$\begin{aligned} e^{i\mathbf{k}^T\mathbf{r}} &= \sum_{\ell=0}^{\infty} (i)^\ell (2\ell+1) j_\ell(kr) \mathbf{Y}^\ell(\mathbf{r}) \bullet_0 \mathbf{Y}^\ell(\mathbf{k}) \\ &= \sum_\ell (i)^\ell (2\ell+1) j_\ell(kr) \sum_{m=-\ell}^{m=\ell} \overline{Y_m^\ell}(\mathbf{r}) Y_m^\ell(\mathbf{k}) \\ &= \sum_\ell (i)^\ell (2\ell+1) j_\ell(kr) \sum_{m=-\ell}^{m=\ell} Y_m^\ell(\mathbf{r}) \overline{Y_m^\ell}(\mathbf{k}) \\ &= \sum_\ell (i)^\ell (2\ell+1) j_\ell(kr) P_\ell(\mathbf{r}^T\mathbf{k}) \\ &= \sum_\ell (i)^\ell (2\ell+1) \langle j_\ell(kr) \mathbf{Y}^\ell(\mathbf{r}), \mathbf{Y}^\ell(\mathbf{k}) \rangle \quad , \end{aligned} \tag{C.18}$$

where $\mathbf{Y}^\ell = \left(Y_{-\ell}^\ell, \ldots, Y_\ell^\ell\right)^T$ are the semi-Schmidt normalized spherical harmonics written as vector.

C.3.7. Spherical Gabor Kernels

Theorem C.3.4. *The spherical derivatives $\mathcal{B}_s^\ell$ of the Gaussian windowed 0-ordered Bessel functions $\mathcal{B}_s^0(\mathbf{r},k) := j_0(k\frac{r}{\sqrt{t}})e^{\frac{-r^2}{(2ts)}}$ are given by*

$$\mathcal{B}_s^\ell(\mathbf{r},k) = \mathbf{Y}^\ell(\mathbf{r})\left[\sum_{i=0}^{\ell}\binom{\ell}{i}(\tfrac{r}{ts})^{\ell-i}(\tfrac{k}{\sqrt{t}})^i j_i(k\tfrac{r}{\sqrt{t}})\right]e^{\frac{-r^2}{2ts}} \quad . \tag{C.19}$$

(See Fig. C.3 on page 219)

Proof. In the following we prove the explicit representation of the derivatives of the Gaussian windowed Bessel functions (theorem C.3.4) using mathematical induction. Consider the spherical derivatives (see lemma 4.1.1) acting on the radial component (inductive step $\ell \rightarrow \ell+1$) $(r^\ell\frac{\partial}{\partial r}\frac{1}{r^\ell})r^\ell\left[\sum_{i=0}^{\ell}\binom{\ell}{i}(\frac{1}{ts})^{\ell-i}(\frac{k}{r\sqrt{t}})^i j_i(k\frac{r}{\sqrt{t}})\right]e^{\frac{-r^2}{2ts}} =$

$$\cdots + r^\ell\frac{\partial}{\partial r}\binom{\ell}{i}(\tfrac{1}{ts})^{\ell-i}(\tfrac{k}{r\sqrt{t}})^i j_i(k\tfrac{r}{\sqrt{t}})e^{\frac{-r^2}{(2ts)}}$$
$$+ r^\ell\frac{\partial}{\partial r}\binom{\ell}{i+1}(\tfrac{1}{ts})^{\ell-(i+1)}(\tfrac{k}{r\sqrt{t}})^{i+1} j_{i+1}(k\tfrac{r}{\sqrt{t}})e^{\frac{-r^2}{(2ts)}}\cdots$$

Due to the product rule we derivate with respect to both the Gaussian and the Bessel functions (Eq. (C.14)), splitting each addend into two new addends, here exemplarily shown for i and $i+1$, with

$$\cdots - r^{\ell+1}\binom{\ell}{i}(\tfrac{1}{ts})^{\ell-(i-1)}(\tfrac{k}{r\sqrt{t}})^i j_i(k\tfrac{r}{\sqrt{t}})e^{\frac{-r^2}{(2ts)}}$$
$$- r^{\ell+1}\binom{\ell}{i}(\tfrac{1}{ts})^{\ell-i}(\tfrac{k}{r\sqrt{t}})^{i+1} j_{i+1}(k\tfrac{r}{\sqrt{t}})e^{\frac{-r^2}{(2ts)}}$$
$$- r^{\ell+1}\binom{\ell}{i+1}(\tfrac{1}{ts})^{\ell-i}(\tfrac{k}{r\sqrt{t}})^{i+1} j_{i+1}(k\tfrac{r}{\sqrt{t}})e^{\frac{-r^2}{(2ts)}}$$
$$- r^{\ell+1}\binom{\ell}{i+1}(\tfrac{1}{ts})^{\ell-i}(\tfrac{k}{r\sqrt{t}})^{i+2} j_{i+2}(k\tfrac{r}{\sqrt{t}})e^{\frac{-r^2}{(2ts)}} - \cdots \, .$$

Neighboring addends having Bessel functions of equal rank (in the term above the two middle terms) can be merged as shown in the next step, where

$$\cdots - r^{\ell+1}\binom{\ell}{i}(\tfrac{1}{ts})^{\ell-(i-1)}(\tfrac{k}{r\sqrt{t}})^i j_i(k\tfrac{r}{\sqrt{t}})e^{\frac{-r^2}{(2ts)}}$$
$$- r^{\ell+1}\binom{\ell+1}{i+1}(\tfrac{1}{ts})^{(\ell+1)-(i+1)}(\tfrac{k}{r\sqrt{t}})^{i+1} j_{i+1}(k\tfrac{r}{\sqrt{t}})e^{\frac{-r^2}{(2ts)}}$$
$$- r^{\ell+1}\binom{\ell}{i+1}(\tfrac{1}{ts})^{\ell-i}(\tfrac{k}{r\sqrt{t}})^{i+2} j_{i+2}(k\tfrac{r}{\sqrt{t}})e^{\frac{-r^2}{(2ts)}} - \cdots \, .$$

Doing so for all pairs of addends associated with a Bessel function of equal rank leads to the final sum showing our assumption, namely

$$-r^{\ell+1}\left[\sum_{i=0}^{\ell+1}\binom{\ell+1}{i}(\tfrac{1}{ts})^{\ell+1-i}(\tfrac{k}{r\sqrt{t}})^i j_i(k\tfrac{r}{\sqrt{t}})\right]e^{\frac{-r^2}{2ts}} \quad . \qquad \square$$

C.3.8. Associated Laguerre Polynomials

Definition C.3.5 (The Associated Laguerre Polynomials). *The associated Laguerre polynomials are defined by*

$$L_n^k(x) = \sum_{i=0}^{n} (-1)^i \binom{n+k}{n-i} \frac{x^i}{i!} \, . \tag{C.20}$$

For details see Abramowitz and Stegun (1964).

The following 3-point-rule is used in this work:

$$nL_n^k(x) = (n+k)L_{n-1}^k(x) - xL_{n-1}^{k+1}(x) \quad . \tag{C.21}$$

We further need the following differential equation:

$$\tfrac{1}{m!}\tfrac{d^m}{dx^m} x^k L_n^k(x) = \binom{n+k}{m} x^{(k-m)} L_n^{(k-m)}(x) \quad . \tag{C.22}$$

The polynomials are orthogonal over $[0,\infty)$ with respect to the weight $x^k e^{-x}$, namely

$$\int_0^\infty x^k e^{-x} L_n^k(x) L_{n'}^k(x) dx = \frac{\Gamma(n+k+1)}{n!} \delta_{n,n'}. \tag{C.23}$$

Substituting $x \rightarrow \frac{r^2}{2t}$ and $k \rightarrow \frac{1}{2}$ we get

$$\int_0^\infty (r^2)^k L_n^{k+\frac{1}{2}}(\tfrac{r^2}{2t}) L_{n'}^{k+\frac{1}{2}}(\tfrac{r^2}{2t}) e^{\frac{-r^2}{2t}} r^2 dr = 2^{(k+1+\frac{1}{2})} t^{(k+1+\frac{1}{2})} \tfrac{\Gamma(n+k+1+\frac{1}{2})}{2n!} \delta_{n,n'} \quad . \tag{C.24}$$

There exists the following scaling rule:

$$L_n^k(\alpha x) = \sum_{i=0}^{n} \tbinom{n+k}{n-i} \alpha^i (1-\alpha)^{(n-i)} L_i^k(x) \quad . \tag{C.25}$$

(The Wolfram Functions Site, `http://functions.wolfram.com/07.03.16.0005.01`)

Setting $k = (\ell - n) + \frac{1}{2}$, $x = \frac{r^2}{2t}$ and $\alpha = 2$ we get

$$L_n^{(\ell-n)+\frac{1}{2}}(\tfrac{r^2}{t}) = (-1)^n \sum_{i=0}^{n} \tbinom{\ell+\frac{1}{2}}{n-i} (-2)^i L_i^{(\ell-n)+\frac{1}{2}}(\tfrac{r^2}{2t}) \quad . \tag{C.26}$$

C.3.9. Clebsch Gordan Coefficients

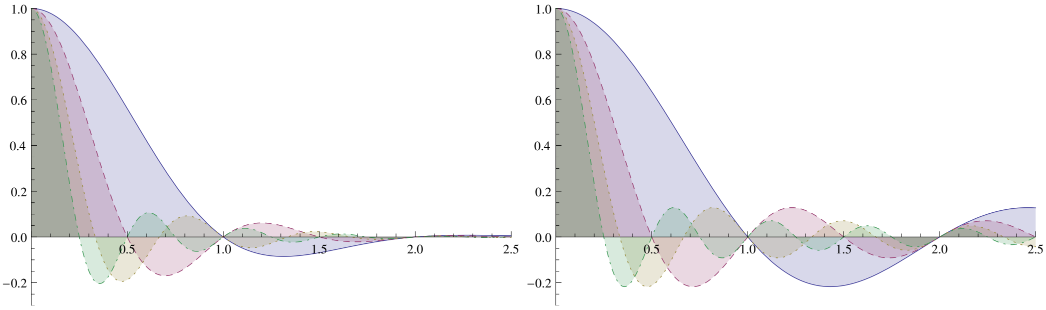
Definition C.3.6 (Explicit Expressions of Clebsch Gordan Coefficients). *The Clebsch Gordan coefficients are defined by*

$$\langle \ell_1 m_1, \ell_2 m_2 | \ell m \rangle = \delta_{m,(m_1+m_2)} \frac{\sqrt{\binom{2\ell_1}{-\ell+\ell_1+\ell_2}\binom{2\ell_2}{-\ell+\ell_1+\ell_2}}}{\sqrt{\binom{\ell+\ell_1+\ell_2+1}{-\ell+\ell_1+\ell_2}\binom{2\ell_1}{\ell_1-m_1}\binom{2\ell_2}{\ell_2-m_2}\binom{2\ell}{\ell-m}}} \times$$

$$\sum_{k=\max(0,-\ell+\ell_2-m_1,-\ell+\ell_1+m_2)}^{k=\min(-\ell+\ell_1+\ell_2,\ell_1-m_1,\ell_2+m_2)} (-1)^k \binom{-\ell+\ell_1+\ell_2}{k} \binom{\ell+\ell_1-\ell_2}{-k+\ell_1-m_1} \binom{\ell-\ell_1+\ell_2}{-k+\ell_2+m_2} . \tag{C.27}$$

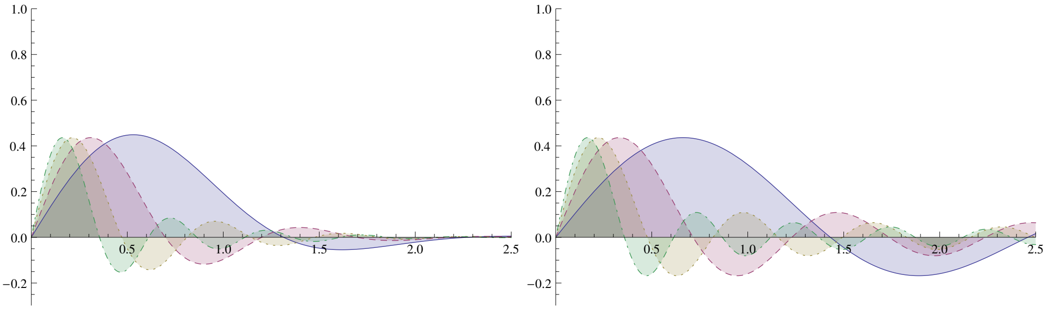
See e.g. (The Wolfram Functions Site, `http://functions.wolfram.com/07.38.06.0003.01` *or Abramowitz and Stegun (1964)).*

Corresponding to a spherical harmonic of order $\ell = 0$



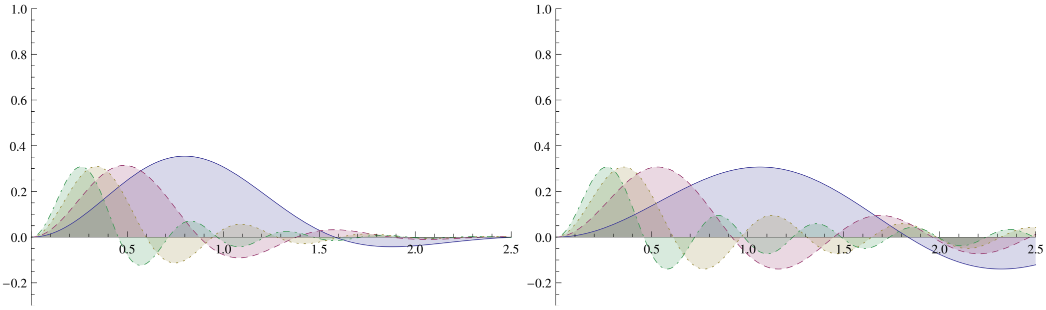
(a) $\mathcal{B}_s^0(\mathbf{r}, k)$ for $\ell = 0$ and $k = 1\pi, 2\pi, 3\pi, 4\pi$ (b) $j_\ell(kr)$ for $\ell = 0$ and $k = 1\pi, 2\pi, 3\pi, 4\pi$

Corresponding to a spherical harmonic of order $\ell = 1$



(c) $\mathcal{B}_s^0(\mathbf{r}, k)$ for $\ell = 1$ and $k = 1\pi, 2\pi, 3\pi, 4\pi$ (d) $j_\ell(kr)$ for $\ell = 1$ and $k = 1\pi, 2\pi, 3\pi, 4\pi$

Corresponding to a spherical harmonic of order $\ell = 2$



(e) $\mathcal{B}_s^0(\mathbf{r}, k)$ for $\ell = 2$ and $k = 1\pi, 2\pi, 3\pi, 4\pi$ (f) $j_\ell(kr)$ for $\ell = 2$ and $k = 1\pi, 2\pi, 3\pi, 4\pi$

Figure C.3.: The radial profile of the Gauss-Bessel functions (our proposed spherical Gabor kernels) on the left, together with the ordinary non-vanishing spherical Bessel functions on the right. Their profiles of the waves are very similar.

C.3.9.1. Symmetry Properties

$$\begin{aligned}
\langle \ell_1 m_1, \ell_2 m_2 | LM\rangle &= (-1)^{\ell_1+\ell_2-L}\langle \ell_1(-m_1), \ell_2(-m_2) | L(-M)\rangle \\
&= (-1)^{\ell_1+\ell_2-L}\langle \ell_2 m_2, \ell_1 m_1 | LM\rangle \\
&= (-1)^{\ell_1-m_1}\sqrt{\frac{2L+1}{2\ell_2+1}}\langle \ell_1 m_1, L(-M) | \ell_2(-m_2)\rangle \\
&= (-1)^{\ell_2+m_2}\sqrt{\frac{2L+1}{2\ell_1+1}}\langle L(-M), \ell_2 m_2 | \ell_1(-m_1)\rangle \\
&= (-1)^{\ell_1-m_1}\sqrt{\frac{2L+1}{2\ell_2+1}}\langle LM, \ell_1(-m_1) | \ell_2 m_2\rangle \\
&= (-1)^{\ell_2+m_2}\sqrt{\frac{2L+1}{2\ell_1+1}}\langle \ell_2(-m_2), LM | \ell_1 m_1\rangle
\end{aligned} \tag{C.28}$$

(The Wolfram Functions Site, `http://functions.wolfram.com/HypergeometricFunctions/ClebschGordan/17/02/07/`)

C.3.9.2. Special Cases

$$\langle \ell_1 m_1, \ell_2 m_2 | 00\rangle = \frac{(-1)^{\ell_1-m_1}}{\sqrt{2\ell_1+1}}\delta_{m_1,(-m_2)}\delta_{\ell_1,\ell_2} \tag{C.29}$$

(The Wolfram Functions Site, `http://functions.wolfram.com/07.38.03.0002.01`) From Eq. (C.28) and Eq (C.29) we can conclude that

$$\langle \ell_1 m_1, 00 | \ell_2 m_2\rangle = \sqrt{\frac{2\ell_2+1}{2\ell_1+1}}\delta_{m_1,m_2}\delta_{\ell_1,\ell_2} = \delta_{m_1,m_2}\delta_{\ell_1,\ell_2} \quad . \tag{C.30}$$

$$\begin{aligned}
&\langle \ell_1 m_1, \ell_2 m_2 | (\ell_1-\ell_2)(m_1+m_2)\rangle = \\
&= \frac{(-1)^{\ell_2+m_2}\sqrt{(\ell_1+m_1)!(\ell_1-m_1)!(2\ell_2)!(2\ell_1-2\ell_2+1)!}}{\sqrt{(2\ell_1+1)!(\ell_2+m_2)!(\ell_2-m_2)!(\ell_1-\ell_2+m_1+m_2)!(\ell_1-\ell_2-m_1-m_2)!}}
\end{aligned} \tag{C.31}$$

(The Wolfram Functions Site, `http://functions.wolfram.com/07.38.03.0008.01`)

$$\begin{aligned}
&\langle \ell_1 m_1, \ell_2 m_2 | (\ell_1+\ell_2)(m_1+m_2)\rangle = \\
&= \frac{\sqrt{(2\ell_1)!(2\ell_2)!(\ell_1+\ell_2+m_1+m_2)!(\ell_1+\ell_2-m_1-m_2)!}}{\sqrt{(2\ell_1+2\ell_2)!(\ell_1+m_1)!(\ell_1-m_1)!(\ell_2+m_2)!(\ell_2-m_2)!}}
\end{aligned} \tag{C.32}$$

(The Wolfram Functions Site, `http://functions.wolfram.com/07.38.03.0003.01`)
The Clebsch-Gordan coefficients are orthogonal according to

$$\sum_{m_1,m}\langle \ell_1 m_1, \ell_2 m_2 | \ell m\rangle\langle \ell_1 m_1, \ell_2' m_2' | \ell m\rangle = \frac{2\ell+1}{2\ell_2'+1}\delta_{\ell_2,\ell_2'}\delta_{m_2,m_2'} \, . \tag{C.33}$$

C.3.10. Wigner 6j-Symbols

Definition C.3.7 (Wigner 6j-Symbols). *The Wigner 6j-Symbols are defined by*

$$\begin{Bmatrix} \ell_1 & \ell_2 & \ell_4 \\ J & \ell_3 & \ell_5 \end{Bmatrix} = \frac{(-1)^{\ell_1+\ell_2+\ell_3+J}}{\sqrt{(2\ell_4+1)(2\ell_5+1)}} \sum_{\substack{m_1,m_2,m_3 \\ m_4,m_5}} \langle \ell_1 m_1, \ell_2 m_2 \,|\, \ell_4 m_4 \rangle$$
$$\times \langle \ell_3 m_3, \ell_4 m_4 \,|\, JM \rangle \langle \ell_1 m_1, \ell_3 m_3 \,|\, \ell_5 m_5 \rangle \langle \ell_2 m_2, \ell_5 m_5 \,|\, JM \rangle \,. \tag{C.34}$$

(Taken from Krainov et al. (2005), page 1, Eq. (D.2))

The permutation of any pairs

$$\left.\begin{matrix} \ell_1 \\ \ell_2 \end{matrix}\right\} \rightleftarrows \left\{\begin{matrix} \ell_3 \\ \ell_4 \end{matrix}\right. \qquad \text{or} \qquad \left.\begin{matrix} \ell_1 \\ \ell_2 \end{matrix}\right\} \rightleftarrows \left\{\begin{matrix} \ell_2 \\ \ell_1 \end{matrix}\right. \tag{C.35}$$

leaves the value of the 6j symbol unaltered (see Krainov et al. (2005)).

$$\begin{Bmatrix} \ell_1 & \ell_2 & (\ell_1+\ell_2) \\ (\ell_1+\ell_2+\ell_5) & \ell_5 & (\ell_1+\ell_5) \end{Bmatrix} = \frac{(-1)^{2\ell_1+2\ell_2+2\ell_5}}{\sqrt{2(\ell_1+\ell_2)+1}\sqrt{2(\ell_1+\ell_5)+1}} \tag{C.36}$$

(The Wolfram Functions Site, `http://functions.wolfram.com/07.40.03.0017.01`)

$$\begin{Bmatrix} \ell_1 & \ell_2 & \ell_3 \\ 0 & \ell_5 & \ell_6 \end{Bmatrix} = \frac{(-1)^{\ell_1+\ell_2+\ell_5} \delta_{\ell_2,\ell_6} \delta_{\ell_3,\ell_5}}{\sqrt{2\ell_2+1}\sqrt{\ell_3+1}} \tag{C.37}$$

(The Wolfram Functions Site, `http://functions.wolfram.com/07.40.03.0006.01`)

$$\begin{Bmatrix} \ell_1 & \ell_2 & (\ell_1+\ell_2) \\ (\ell_3-\ell_2-\ell_1) & \ell_3 & (\ell_3-\ell_1) \end{Bmatrix} = \frac{1}{\sqrt{2(\ell_1+\ell_2)+1}\sqrt{2(\ell_3-\ell_1)+1}} \tag{C.38}$$

(A direct conclusion from Eq (2.2.31))

C.3.11. Wigner D-Matrix

The small Wigner d-matrix (Wigner 1931; Wigner and Griffin 1959) is defined by

$$d^{\ell}_{mn}(\theta) = \sqrt{(\ell+m)!(\ell-m)!(\ell+n)!(\ell-n)!} \quad \times \tag{C.39}$$
$$\sum_k \frac{(-1)^{m-n+k}}{(\ell+n-k)!k!(m-n+s)!(\ell-m-k)!} (\cos(\tfrac{\theta}{2}))^{2\ell+n-m-2k} (\sin(\tfrac{\theta}{2}))^{m-n+2k}$$

The Wigner D-Matrices obey the following parity

$$[\mathbf{D}^{\ell}_{\mathfrak{g}}]_{mn} = (-1)^{m-n} \overline{[\mathbf{D}^{\ell}_{\mathfrak{g}}]_{(-m),(-n)}} \tag{C.40}$$

Recursive relation of Wigner D-matrices:

$$[\mathbf{D}^{\ell_1}_{\mathfrak{g}}]_{m_1,n_1} [\mathbf{D}^{\ell_2}_{\mathfrak{g}}]_{m_2,n_2} = \sum_{\ell,m,n} \langle \ell_1 m_1, \ell_2 m_2 \,|\, \ell m \rangle \langle \ell_1 n_1, \ell_2 n_2 \,|\, \ell n \rangle [\mathbf{D}^{\ell}_{\mathfrak{g}}]_{m,n} \,. \tag{C.41}$$

Utilizing the orthogonality relation of Wigner D-matrices (Eq. (2.22) on page 45) we get

$$\begin{aligned}
&\int_{SO(3)} \overline{[\mathbf{D}_{\mathfrak{g}}^{\ell_1}]_{m_1,n_1}} [\mathbf{D}_{\mathfrak{g}}^{\ell_2}]_{m_2,n_2} [\mathbf{D}_{\mathfrak{g}}^{\ell_3}]_{m_3,n_3} d\mathfrak{g} \\
&= \int_{SO(3)} \overline{[\mathbf{D}_{\mathfrak{g}}^{\ell_1}]_{m_1,n_1}} \sum_{\ell,m,n} \langle \ell_2 m_2, \ell_3 m_3 \,|\, \ell m \rangle \langle \ell_2 n_2, \ell_3 n_3 \,|\, \ell n \rangle [\mathbf{D}_{\mathfrak{g}}^{\ell}]_{m,n} d\mathfrak{g} \\
&= \sum_{\ell,m,n} \langle \ell_2 m_2, \ell_3 m_3 \,|\, \ell m \rangle \langle \ell_2 n_2, \ell_3 n_3 \,|\, \ell n \rangle \int_{SO(3)} \overline{[\mathbf{D}_{\mathfrak{g}}^{\ell_1}]_{m_1,n_1}} [\mathbf{D}_{\mathfrak{g}}^{\ell}]_{m,n} d\mathfrak{g} \\
&= \langle \ell_2 m_2, \ell_3 m_3 \,|\, \ell_1 m_1 \rangle \langle \ell_2 n_2, \ell_3 n_3 \,|\, \ell_1 n_1 \rangle \frac{8\pi^2}{2\ell_1 + 1} \quad . \qquad \text{(C.42)}
\end{aligned}$$

C.3.12. Spherical Derivatives

Algorithm 2 (spherical up-derivatives) $\boldsymbol{\nabla}^1 : \mathcal{T}_\ell \to \mathcal{T}_{(\ell+1)}$

```
input: $\mathbf{f}^\ell \in \mathcal{T}_\ell$
for $\forall \mathbf{r} \in \mathbf{N}^3$ do
  for $m = -(\ell+1)$ to $(\ell+1)$ do
    $\mathbf{f}^{\ell+1}_m(\mathbf{r}) = 0$
    if $|m+1| \leq \ell$ then
      $\mathbf{f}^{\ell+1}_m(\mathbf{r}) = \mathbf{f}^{\ell+1}_m(\mathbf{r}) + \frac{\sqrt{(\ell-m)(1+\ell-m)}}{2(\ell+1)}(\partial_x \mathbf{f}^\ell_{(m+1)} - i\partial_y \mathbf{f}^\ell_{(m+1)})(\mathbf{r})$
    end if
    if $|m| \leq \ell$ then
      $\mathbf{f}^{\ell+1}_m(\mathbf{r}) = \mathbf{f}^{\ell+1}_m(\mathbf{r}) + \frac{\sqrt{(\ell+m+1)(\ell-m+1)}}{\ell+1}(\partial_z \mathbf{f}^\ell_m)(\mathbf{r})$
    end if
    if $|m-1| \leq \ell$ then
      $\mathbf{f}^{\ell+1}_m(\mathbf{r}) = \mathbf{f}^{\ell+1}_m(\mathbf{r}) - \frac{\sqrt{(\ell+m)(1+\ell+m)}}{2(\ell+1)}(\partial_x \mathbf{f}^\ell_{(m-1)} + i\partial_y \mathbf{f}^\ell_{(m-1)})(\mathbf{r})$
    end if
  end for
end for
return $\mathbf{f}^{\ell+1}$
```

Algorithm 3 (spherical down-derivatives) $\boldsymbol{\nabla}_1 : \mathcal{T}_\ell \to \mathcal{T}_{(\ell-1)}$

```
input: $\mathbf{f}^\ell \in \mathcal{T}_\ell$, $\ell \geq 1$
for $\forall \mathbf{r} \in \mathbf{N}^3$ do
  for $m = -(\ell-1)$ to $(\ell-1)$ do
    $\mathbf{f}^{\ell-1}_m(\mathbf{r}) = 0$
    if $|m+1| \leq \ell$ then
      $\mathbf{f}^{\ell-1}_m(\mathbf{r}) = \mathbf{f}^{\ell-1}_m(\mathbf{r}) + \frac{\sqrt{(\ell+m)(1+\ell-m)}}{2(\ell-1)}(\partial_x \mathbf{f}^\ell_{(m+1)} - i\partial_y \mathbf{f}^\ell_{(m+1)})(\mathbf{r})$
    end if
    if $|m| \leq \ell$ then
      $\mathbf{f}^{\ell-1}_m(\mathbf{r}) = \mathbf{f}^{\ell-1}_m(\mathbf{r}) + \frac{\sqrt{(\ell+m)(\ell-m)}}{\ell-1}(\partial_z \mathbf{f}^\ell_m)(\mathbf{r})$
    end if
    if $|m-1| \leq \ell$ then
      $\mathbf{f}^{\ell-1}_m(\mathbf{r}) = \mathbf{f}^{\ell-1}_m(\mathbf{r}) - \frac{\sqrt{(\ell-m)(1+\ell+m)}}{2(\ell-1)}(\partial_x \mathbf{f}^\ell_{(m-1)} + i\partial_y \mathbf{f}^\ell_{(m-1)})(\mathbf{r})$
    end if
  end for
end for
return $\mathbf{f}^{\ell-1}$
```

D Harmonic Filters

D.1. $SE(3)$ Covariant Filters

In this thesis we proposed filters mapping images to locally rotation invariant saliency maps. The elements of these saliency maps are power-spectrum and bi-spectrum features; rotation invariant image patch representations that are analytically derived from spherical tensor valued expansion coefficients. Each feature represents the surrounding of an image point. In a second step, a trainable classifier evaluates the saliency map in a voxel-by-voxel manner. In this way, objects or structures can be detected in an $SE(3)$ covariant way.

The proposed spectra based invariant features are collecting informations in a voxel's neighborhood and encoding them in a rotation invariant way. That is, the process can be regarded as some kind of information implosion: concentrating all information in one point. The harmonic filters (Reisert and Burkhardt 2009a) can be regarded as an extension to such power- and bi-spectrum based filters. These harmonic filters however do not directly compute invariant features from expansion coefficients, but combine the information at each image point in a nonlinear manner. Afterwards they spread this information into a points neighborhood in a steerable way. This can be regarded as a kind of feature collection followed by a voting step. The steerability of the votes can be utilized to build a filter

$$\mathcal{H} : L_2(\mathbb{R}^3) \rightarrow L_2(\mathbb{R}^3) \,, \tag{D.1}$$

mapping images to $SE(3)$ covariant saliency maps. These maps are directly representing the evidence for the presence or absence of objects; see Fig. D.1 for an example.

The trainable harmonic filter for generic object detection in 3D is an extension of the 2D holomorphic filter (Reisert and Burkhardt 2008c) and has been invented by Reisert and Burkhardt (2009a). The harmonic filter and its variations have been suc-

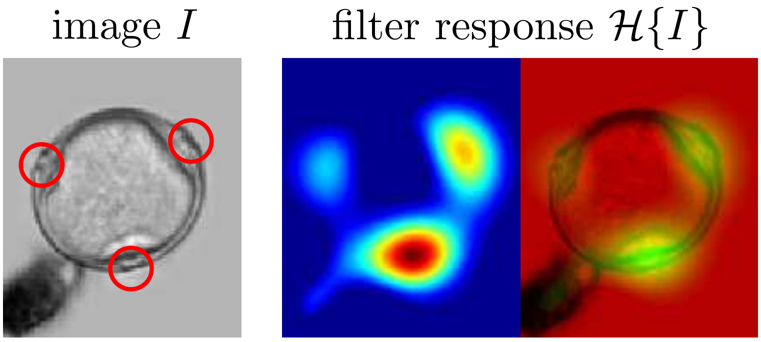


Figure D.1.: Saliency map of a filter for pore detection in microscopical images of pollen-grains; Image shows results presented in Skibbe and Reisert (2012a).

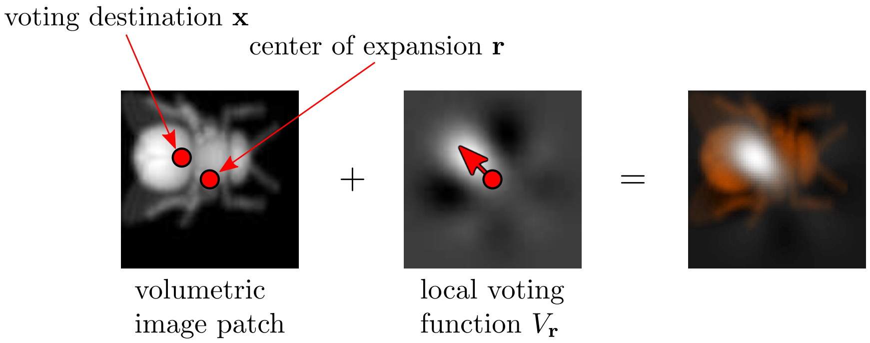


Figure D.2.: Voting function of the filter.

cessfully used for the detection of landmarks and objects in a broad variety of applications. This includes the detection of mitoses in colorectal cancer (Schlachter et al. 2010), the detection of landmarks in zebrafish embryos (Liu et al. 2012; Ronneberger et al. 2012), the detection of landmarks in airborne pollen (Skibbe and Reisert 2013; Skibbe et al. 2011b; Skibbe et al. April, 2011), or the detection of landmarks in MRI data of human brains (Skibbe and Reisert 2012b; Skibbe and Reisert 2013).

The harmonic filter can be derived from nonlinear polynomial filters via group integration techniques (see Reisert and Burkhardt (2008c)). We will focus on a more intuitive interpretation as an $SE(3)$ covariant voting filter for generic object detection. We further extend the originally used spherical Gaussian derivatives via a complete basis spanned by the Gauss-Laguerre polynomials in the following derivation.

Suppose we want to detect the center of an image of a fly as illustrated in Fig. D.2. In this scenario we are considering the voting scenario from an image point $\mathbf{r} \in \mathbb{R}^3$. Since the center of the fly is at position $\mathbf{x} \in \mathbb{R}^3$, we aim at creating a voting function that casts votes from $\mathbf{r}$ to $\mathbf{x}$. The Gauss-Laguerre functions $\{\boldsymbol{L}_n^\ell(\frac{t}{2})\}$ provide an orthogonal basis for freely designing such a voting function. We denote the voting function at position $\mathbf{r}$ by $V_{\mathbf{r}} \in L_2(\mathbb{R}^3)$, defined by

$$V_{\mathbf{r}}(\mathbf{x}) := \sum_{n \leq \ell} \mathbf{b}_n^\ell(\mathbf{r})^T \overline{\boldsymbol{L}_n^\ell(\mathbf{r} - \mathbf{x}, \tfrac{t}{2})} \quad . \tag{D.2}$$

With $\mathbf{b}_n^\ell(\mathbf{r}) \in \mathbb{C}^{2(\ell-n)+1}$ we denote the expansion coefficients steering the shape of the voting function. Since the basis functions $\{\boldsymbol{L}_n^\ell(\frac{t}{2})\}$ and $\{\boldsymbol{\mathcal{L}}_n^\ell(t) \cdot G_t\}$ are spanning the same function space (Thm. 5.1.2 on page 118), we can rewrite the voting function into

$$
\begin{aligned}
V_{\mathbf{r}}(\mathbf{x}) &= \sum_{n\leq\ell} \mathbf{b}_n^{\ell}(\mathbf{r})^T \overline{\boldsymbol{L}_n^{\ell}(\mathbf{r}-\mathbf{x},\tfrac{t}{2})} \\
V_{\mathbf{r}}(\mathbf{x}) &= \sum_{n\leq\ell} (-1)^{(\ell+n)} \mathbf{b}_n^{\ell}(\mathbf{r})^T \overline{\boldsymbol{L}_n^{\ell}(\mathbf{x}-\mathbf{r},\tfrac{t}{2})} \\
&= \sum_{n\leq\ell} \mathbf{V}_n^{\ell}(\mathbf{r})^T \overline{\boldsymbol{\mathcal{L}}_n^{\ell}(\mathbf{x}-\mathbf{r},t)} e^{-\frac{\|\mathbf{x}-\mathbf{r}\|^2}{2t}} \qquad \text{(according to Eq. (5.1.2))} \\
&= \sum_{n\leq\ell} \mathbf{V}_n^{\ell}(\mathbf{r}) \bullet_0 \boldsymbol{\mathcal{L}}_n^{\ell}(\mathbf{x}-\mathbf{r},t) e^{-\frac{\|\mathbf{x}-\mathbf{r}\|^2}{2t}} \quad , \qquad \text{(D.3)}
\end{aligned}
$$

where $\mathbf{V}_n^{\ell}(\mathbf{r}) \in \mathbb{C}^{2(\ell-n)+1}$ are new expansion coefficients (this will allow for an efficient implementation via tensor derivatives). We skip the discussion about the determination of these expansion coefficients and implementation details for the moment. The idea of the filter is that every voxel casts votes for the absence or presence of objects. The filter response itself is a superposition of all votes from all image points. This can be formulated as an integral which, plugging into it the spherical expansions of the voting function, turns out to be

$$
\mathcal{H}(\mathbf{x}) := \int_{\mathbb{R}^3} V_{\mathbf{r}}(\mathbf{x}) d\mathbf{r} = \left(\sum_{n\leq\ell} \mathbf{V}_n^{\ell} \widetilde{\bullet}_0 \boldsymbol{\mathcal{L}}_n^{\ell}(t) e^{\frac{-r^2}{2t}} \right)(\mathbf{x}) \quad . \qquad \text{(D.4)}
$$

At this point the filter is just a pure application of tensor operations. The steering of the function is done via the coefficients $\mathbf{V}_n^{\ell}(\mathbf{r})$. For simple filtering purposes, the coefficients $\mathbf{V}_n^{\ell}(\mathbf{r})$ might be even chosen manually (Reisert and Burkhardt 2009a). However, for the generic detection of objects, the coefficients are learned in a data driven way. Moreover, the filter output should depend on its input image in a covariant manner. Let $I \in L_2(\mathbb{R}^3)$ be an input image. Then the expansion coefficients are typically themselves the result of an $SE(3)$ covariant mapping

$$
\mathbf{V}_n^{\ell} : L_2(\mathbb{R}^3) \to \mathcal{T}_{(\ell-n)} \qquad \text{(D.5)}
$$

fulfilling

$$
\mathbf{V}_n^{\ell}\{\mathfrak{g}I\}(\mathbf{x}) = \mathbf{D}_{\mathfrak{g}}^{(\ell-n)} \mathbf{V}_n^{\ell}\{I\}(\mathbf{U}_{\mathfrak{g}}^T \mathbf{x}) \quad . \qquad \text{(D.6)}
$$

The trainable filter becomes

$$
\mathcal{H}\{I\} := \left(\sum_{n\leq\ell} \mathbf{V}_n^{\ell}\{I, \xi_{\ell,n}\} \widetilde{\bullet}_0 \boldsymbol{\mathcal{L}}_n^{\ell}(t) e^{\frac{-r^2}{2t}} \right) \quad , \qquad \text{(D.7)}
$$

where $\xi_{\ell,n}$ is a coefficient dependent parameter vector. These parameter vectors are the free filter parameters that can be adapted to specific detection tasks. The performance of the filter highly depends on the design of $\mathbf{V}_n^{\ell}$.

For instance, the following non-linear mapping $\mathbf{V}_0^{\ell} : L_2(\mathbb{R}^3) \to \mathcal{T}_{\ell}$ is proposed by Reisert and Burkhardt (2009a). It is based on summarizing over second order tensor products of spherical harmonic expansion coefficients with equal rank:

$$\mathbf{V}^{\ell}_{i=0,\cdots,N}\{I,\alpha_i\} := \sum_{\substack{|\ell_1-\ell_2|\leq\ell\leq\ell_1+\ell_2 \\ \ell_1+\ell_2+\ell \text{ even}}} \alpha^{\ell}_{\ell_1,\ell_2}(\mathbf{a}^{\ell_1} \circ_{\ell} \mathbf{a}^{\ell_2}) \quad . \tag{D.8}$$

Each addend is weighted with a scalar valued factor $\alpha^{\ell}_{\ell_1,\ell_2} \in \mathbb{R}$. The weights are initially determined in a training step via least square fit to a (manually) labeled ground truth image representing the "optimal" filter response. Note that in this case, the radial polynomial degree n has been set to zero, resulting in a simplified voting function purely based on Gaussian derivatives. This is a good trade-of regarding detection performance and computation time.

The voting process is illustrated in Figs. D.3 on page 230- D.7 on page 234.

Alternatives for a mapping $\mathbf{V}^{\ell}_{n}$ have been proposed by Schlachter et al. (2010), where microscopical images with multiple fluorescent markers have been combined to form multichannel features. They further extended the feature extraction to vector valued fields. It is also possible to use higher order tensor fields as input and using third order products to cope with object symmetries (Skibbe and Reisert 2012b). Recent work by Liu et al. (2012) has shown a remarkable performance gain by incorporating power-spectrum features in combination with unsupervised clustering techniques into a covariant mapping.

One big advantage for the filter is its fast application. Thanks to the linearity of convolution and differentiation, the voting itself can also be formalized in terms of spherical tensor derivatives and one single convolution:

$$\begin{aligned}
\mathcal{H}\{I\} &= \sum_{n\leq\ell} \mathbf{V}^{\ell}_{n}\{I\}\widetilde{\bullet}_0 \boldsymbol{\mathcal{L}}^{\ell}_{n}(t) G_t = \sum_{n\leq\ell} \mathbf{V}^{\ell}_{n}\{I\}\widetilde{\bullet}_0 \left(\tfrac{(-t)^{\ell}}{n!2^n} \boldsymbol{\nabla}^{\ell}_{n} G_t\right) \\
&= \sum_{n\leq\ell} \mathbf{V}^{\ell}_{n}\{I\}\widetilde{\bullet}_0 \left(\tfrac{(-t)^{\ell}}{n!2^n} (\partial^{(\ell-n)} \bullet_{(\ell-n)} (\Delta^n G_t))\right) \\
&= \sum_{n\leq\ell} \Delta^n \left(\partial^{(\ell-n)} \bullet_0 (\tfrac{(-t)^{\ell}}{n!2^n} \mathbf{V}^{\ell}_{n}\{I\})\right)\widetilde{\bullet}_0 G_t) && \text{(using Eq. (2.2.31))} \\
&= \sum_{n\leq\ell} \left(\Delta^n \boldsymbol{\nabla}_{(\ell-n)} \mathbf{V}'^{\ell}_{n}\{I\}\right)\widetilde{\bullet}_0 G_t = G_t * \sum_{n\leq\ell} \left(\Delta^n \boldsymbol{\nabla}_{(\ell-n)} \mathbf{V}'^{\ell}_{n}\{I\}\right) \quad .
\end{aligned} \tag{D.9}$$

D.1.1. Region Descriptors

The non-linear mappings $\mathbf{V}^{\ell}_{n}\{I\}$ used for forming the voting function in combination with the final smoothing $G_t * \left(\Delta^n \boldsymbol{\nabla}_{(\ell-n)} \mathbf{V}'^{\ell}_{n}\right)$ can be regarded as $SE(3)$ covariant filters by them selves. The output is a scalar valued saliency map. The filter output of $\mathcal{H}\{I\}$ is indeed just a linear combination of several of such saliency maps. Hence similar to the power- and bi-spectrum based filters, the output of several filters can be concatenated to create descriptor images. The vector valued components of such descriptor images do not mix under rotations.

In the consideration above, the final smoothing step in the filter distributes the nonlinear spherical tensor features into a voxel's surrounding. On the other hand, this convolution can be considered as an aggregation of features. In this consideration, first, local spherical harmonic expansion coefficients are combined in a nonlinear way. The following smoothing step collects all these features in a voxel's surrounding and forms a scalar valued quantity representing a voxel's region. Because the features combine several spherical harmonic expansion coefficients over a small region, we call them region features (in contrast with the pure power- and bi-spectra invariants considering each only expansion coefficients at one single voxel).

Definition D.1.1 (Covariant Filters based on Region Features)**.** *In our applications we mainly consider two types of such filters* $\mathcal{F}^J : \mathcal{T}_J \to \mathcal{T}_0$*, mapping tensor fields to scalar valued saliency maps. We call the elements of these maps region features. A counterpart of the power-spectrum based filters are the filters involving a second order product,*

$$\mathcal{F}^L\{\mathbf{f}^J, \ell_1, \ell_2, \nu\} := G_\nu * \left(\boldsymbol{\nabla}_L(\mathbf{a}^{\ell_1} \circ_L \mathbf{a}^{\ell_2})\right) \quad . \tag{D.10}$$

The counterpart of the bi-spectrum is a filter involving a triple product:

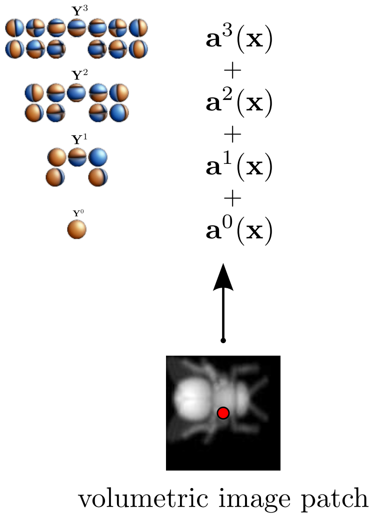
$$\mathcal{F}^L\{\mathbf{f}^J, \ell_1, \ell_2, \ell_3, \ell, \nu\} := G_\nu * \left(\boldsymbol{\nabla}_L(\mathbf{a}^{\ell_1} \circ_L (\mathbf{a}^{\ell_2} \circ_\ell \mathbf{a}^{\ell_3}))\right) \quad . \tag{D.11}$$

Note that similar to the angular power-spectrum and the angular bi-spectrum, the feature vectors of filters involving second order products (Eq. (D.10)) are invariant to reflections, while odd filters involving triple products (Eq. (D.11)) are variant to reflections. We call a filter *odd*, iff $\ell_1 + \ell_2 + \ell_3 + L$ is odd. Otherwise it will be called *even* because it then will not resolve reflections.

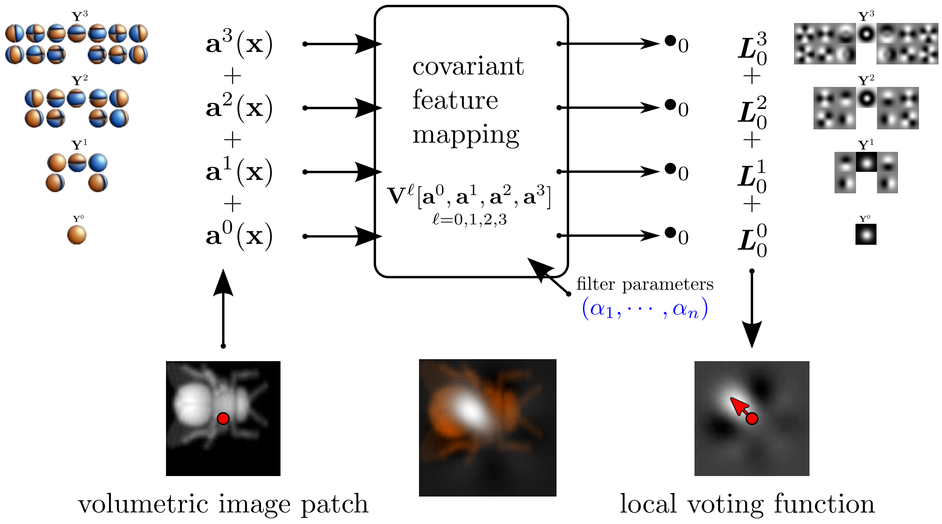
Proof: Variant to Reflections. Our proof is based on the assumptions and definitions in section 3.1.1.3 on page 90. So let $I, I' \in L_2(\mathbb{R}^3)$ and I' be the mirrored version of I. The expansion coefficients of I at image position $\mathbf{x}$ are $\mathbf{a}^\ell(\mathbf{x})$, consequently the expansion coefficients of I' are $\mathbf{b}^\ell(\mathbf{x}) = (-1)^\ell \mathbf{a}^\ell(-\mathbf{x})$. We consider two scenarios. First, the odd derivatives of a mirrored function are pointing in the opposite direction of its original counterpart. Since the tensor derivatives are just a combination of Cartesian derivatives, we can conclude that $(\boldsymbol{\nabla}_\ell \mathbf{f}'^\ell)(\mathbf{x}) = (-1)^\ell (\boldsymbol{\nabla}_\ell \mathbf{f}^\ell)(-\mathbf{x})$ for some spherical tensor fields $\mathbf{f}^\ell, \mathbf{f}'^\ell$ with $\mathbf{f}'^\ell(\mathbf{x}) := \mathbf{f}^\ell(-\mathbf{x})$. The second scenario is the triple product, which is odd if $\ell_1 + \ell_2 + \ell_3$ is odd (see Eq. (3.23) on page 91). Consequently

$$(\boldsymbol{\nabla}_L(\mathbf{b}^{\ell_1} \circ_L (\mathbf{b}^{\ell_2} \circ_\ell \mathbf{b}^{\ell_3})))(\mathbf{x}) = (-1)^{(\ell_1+\ell_2+\ell_3+L)} (\boldsymbol{\nabla}_L(\mathbf{a}^{\ell_1} \circ_L (\mathbf{a}^{\ell_2} \circ_\ell \mathbf{a}^{\ell_3})))(-\mathbf{x}). \tag{D.12}$$

□

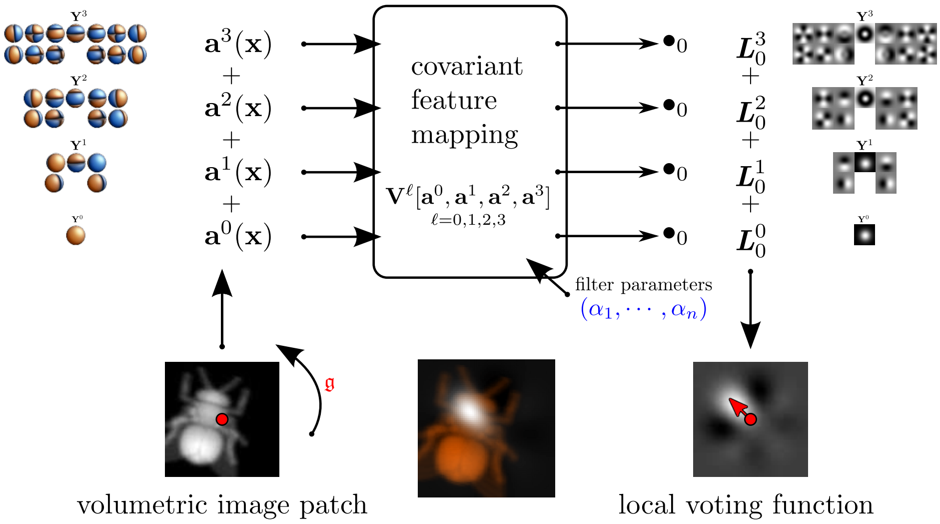


(a) Computing features from local image patches via a local spherical harmonic transform.

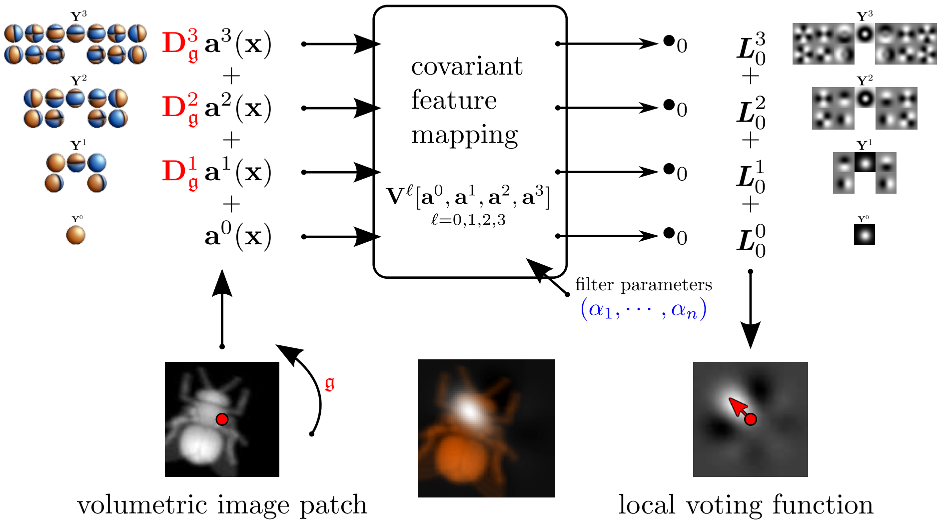


(b) The coefficients are combined in a non-linear manner and form expansion coefficients for a spherical voting function.

Figure D.3.: The Harmonic Filter Framework Fig. D.3(a): Local image patches are represented in terms of spherical tensor coefficients using, e.g. the Gauss-Laguerre transform, Gabor transform or SHOG. The coefficients are densely computed for the whole image using the proposed algorithms. Fig. D.3(b): The expansion coefficients of the patches are combined in a non-linear way forming new spherical tensor coefficients. The resulting coefficients are again (together with trainable filter parameters $alpha_i$) used as expansion coefficients for a volumetric spherical harmonic basis (like the Gauss-Laguerre or Gauss-Bessel functions). The weights α_i are used to steer the shape of the voting function.)

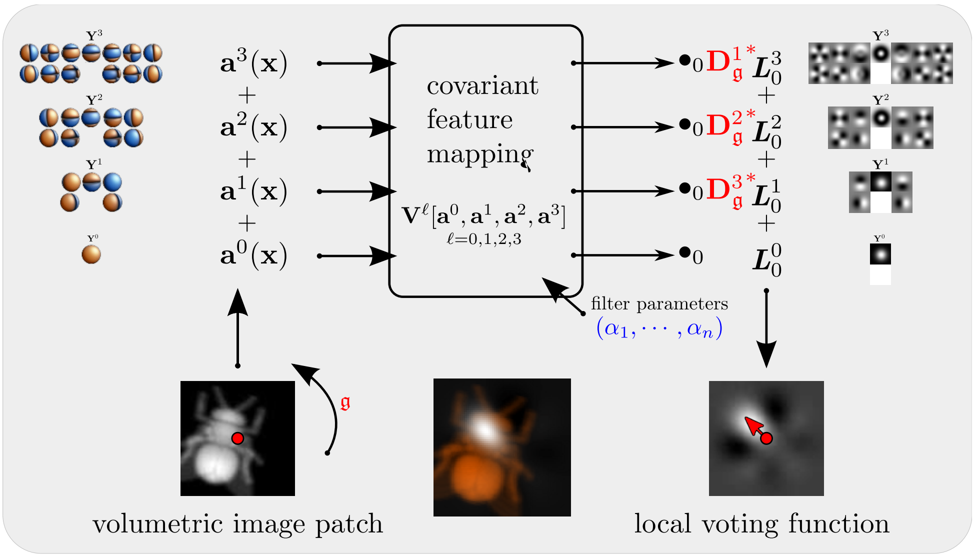


(a) Thanks to the spherical tensor representation: if the local image patch rotates, ...

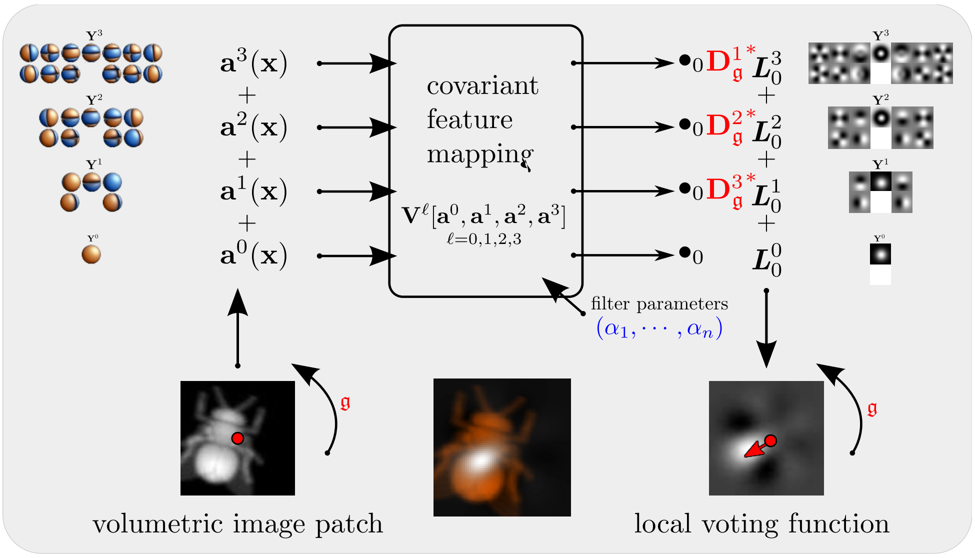


(b) ... the rotation transforms the expansion coefficients.

Figure D.4.: (see Fig. D.3 on the preceding page) Fig. D.4(a): If the image rotates, then the rotation is acting on the expansion coefficients in form of Wigner-D rotation matrices; Fig. D.4(b).

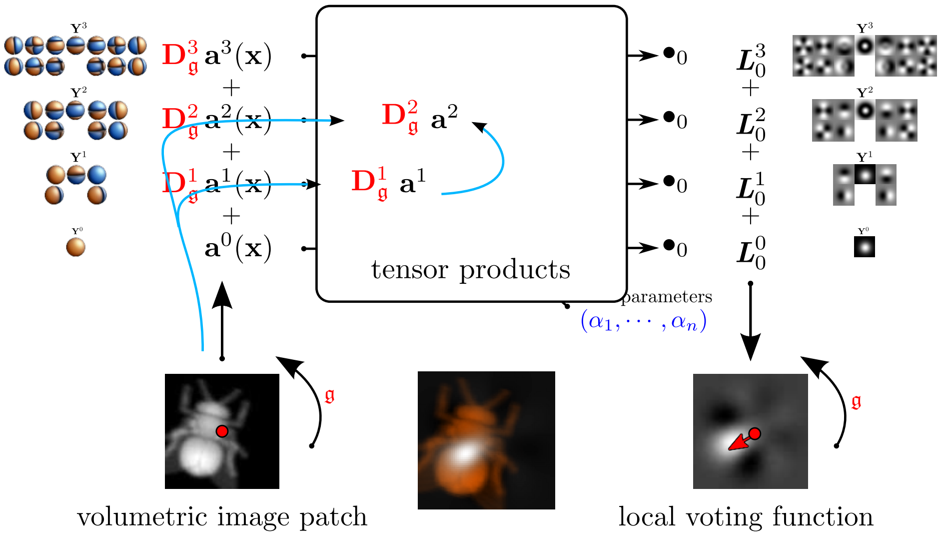


(a) A rotation of the expansion coefficients let the basis function of the voting function rotate.

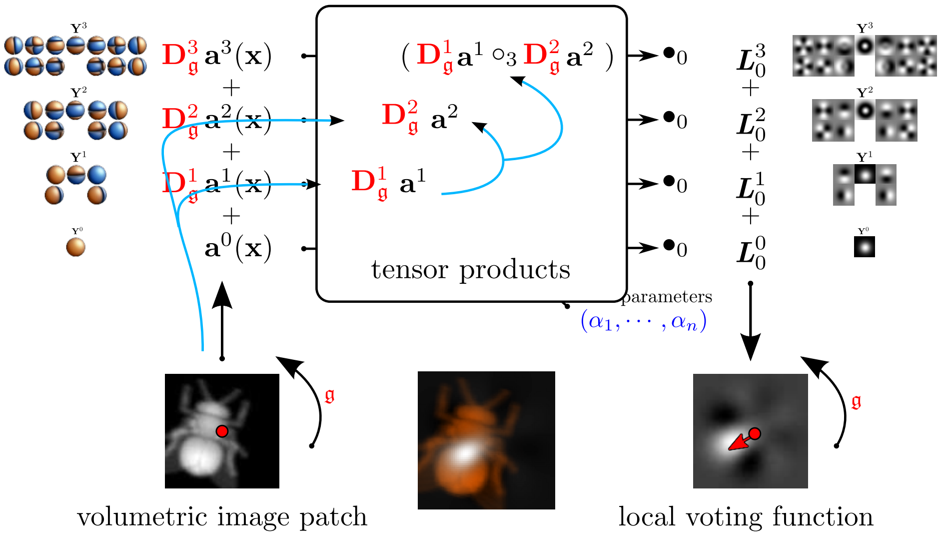


(b) As a result, the voting function rotates.

Figure D.5.: (see Fig. D.4 on the preceding page) The computation of non-linear features must be rotation covariant thus the rotation is acting on the basis functions of the voting function, too; Fig. D.5(a). As a result, the voting function is rotating according to its corresponding image patch; Fig. D.5(b).

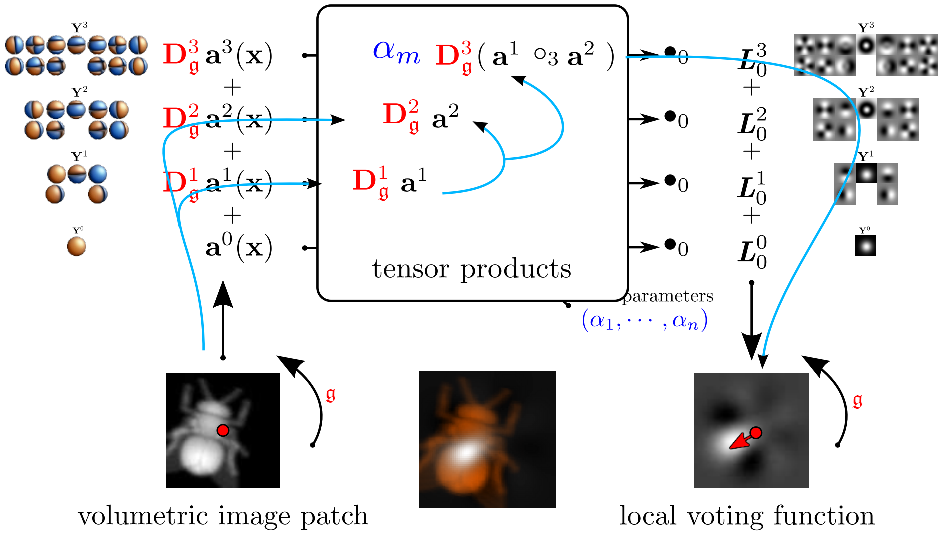


(a) We compute rotation covariant features ....



(b) ... via tensor products.

Figure D.6.: One solution for combining the expansion coefficients in a non-linear way while fulfilling the covariance criteria are spherical tensor products, see Fig. D.6(a) and Fig. D.6(b). Depending on the type of expansion there arise many non-linearly dependent possible products.

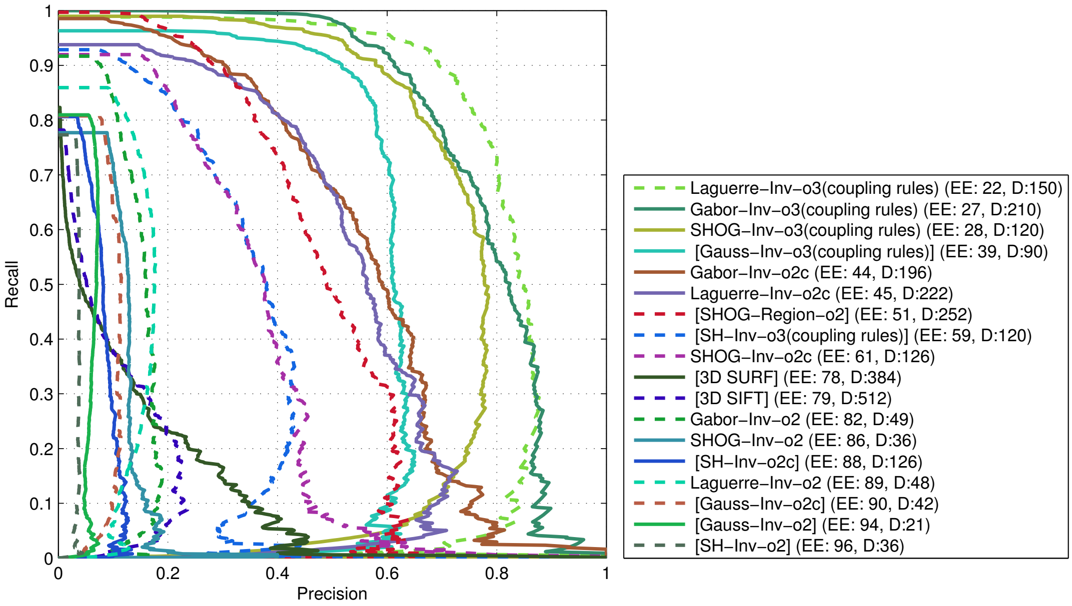


(a) Each tensor product influences the shape and orientation of the voting function via a weighting factor α_m. The weights are learned in a training step.

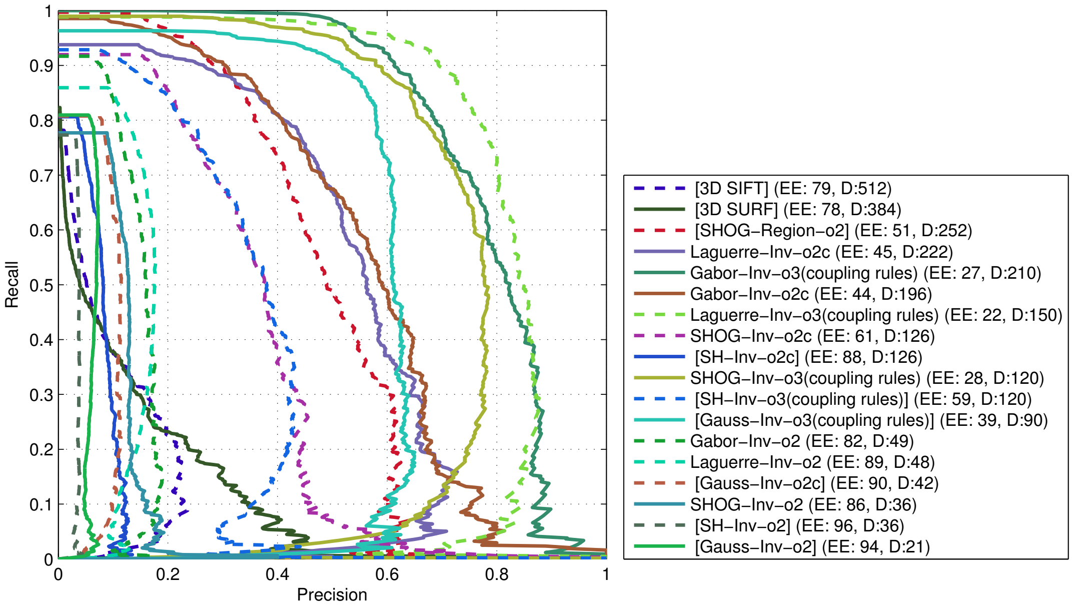
Figure D.7.: Moreover, third order products can be used to solve reflection symmetries within local structures, Fig. D.7(a): each non-linear feature is itself a spherical tensor of order ℓ having a direction and magnitude. Therefore, we can use this information to alter the ℓth order expansion coefficient of the voting function. This influences the voting function's shape. The amount of contribution (the free filter parameter α_m) of each coefficient is learned in a training step.

E Experiments: Plots and Figures

E.1. Plots and Figures: PSB Dataset

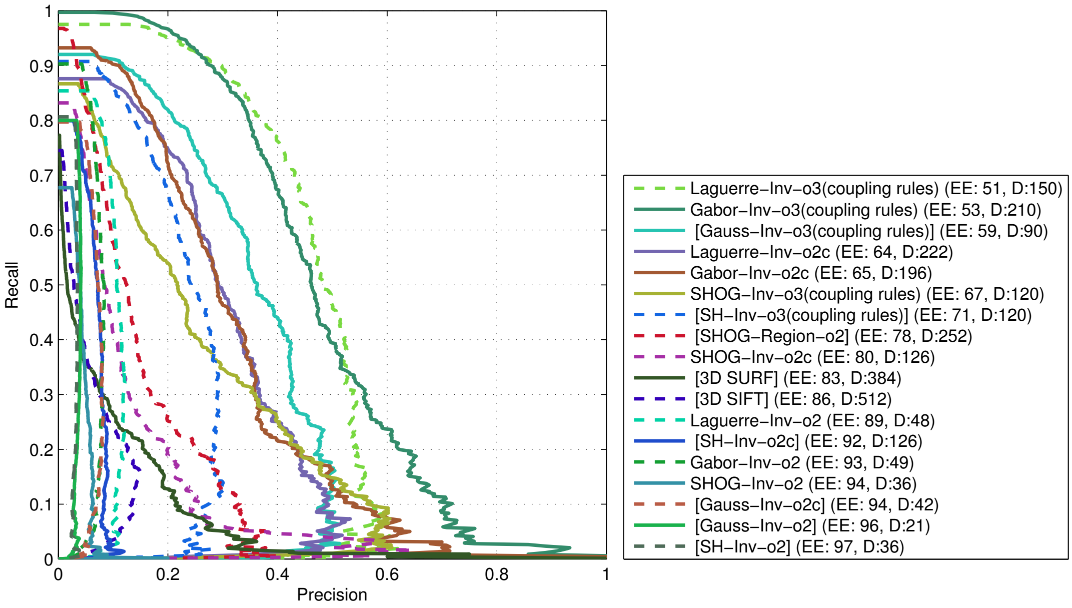


(a) ordering: EER

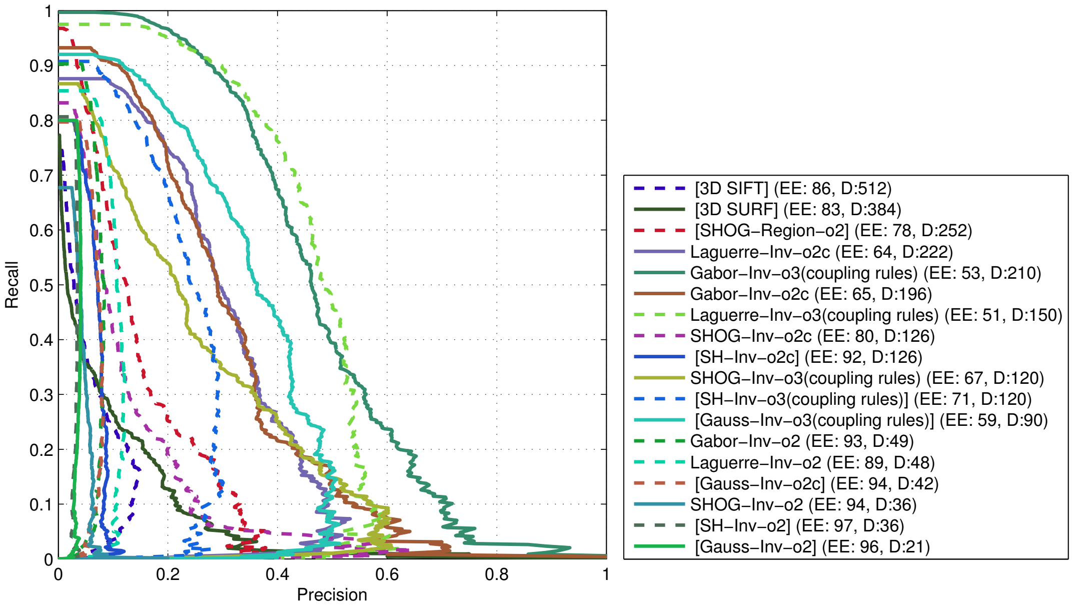


(b) ordering: Dimension

Figure E.1.: PR curves for the PSB dataset with a SNR 20



(a) ordering: EER



(b) ordering: Dimension

Figure E.2.: PR curves for the PSB dataset with a SNR 6.5

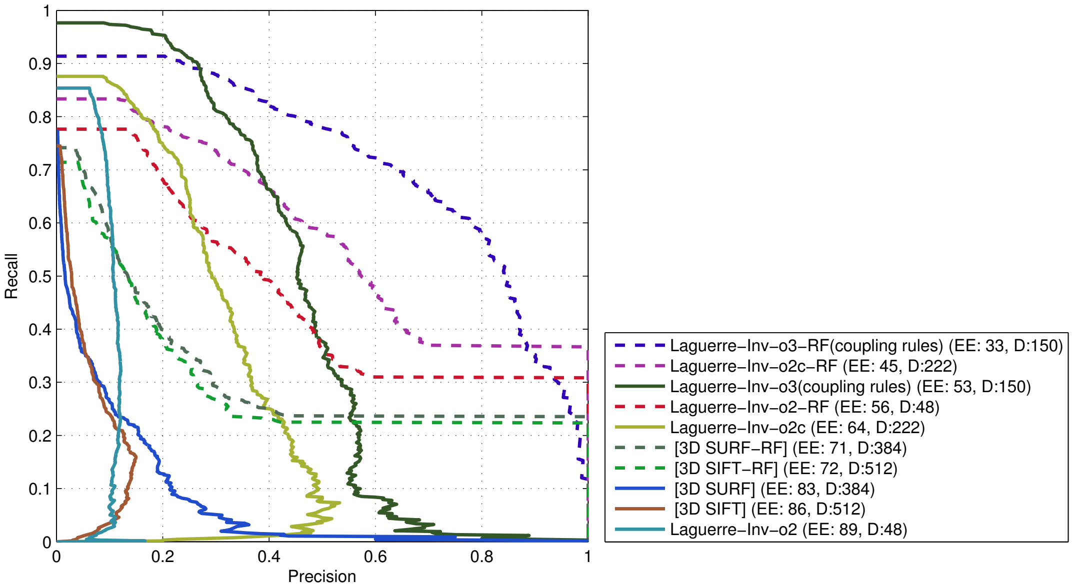
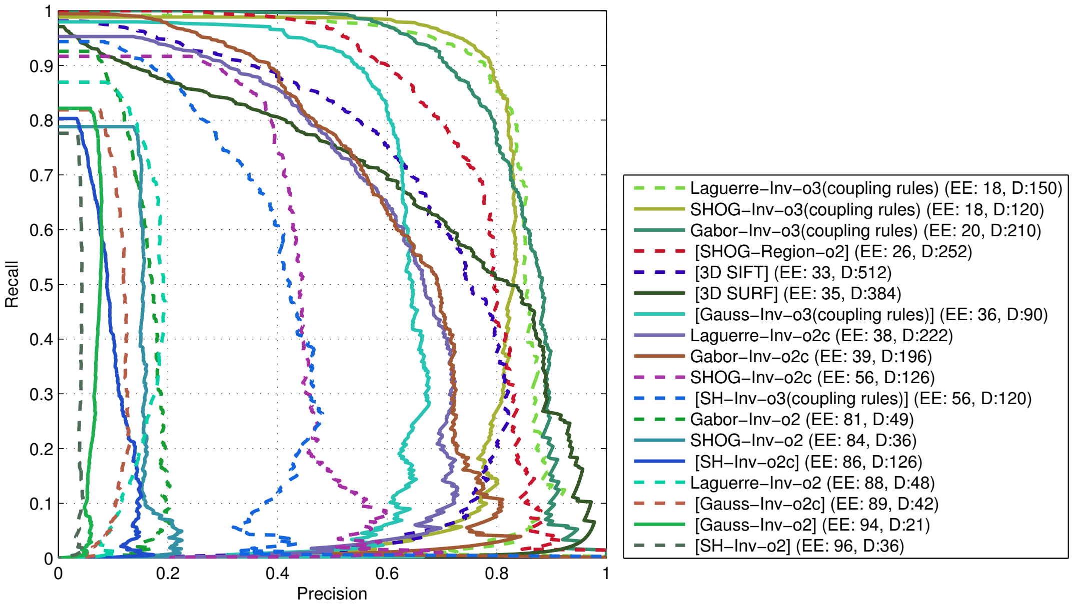
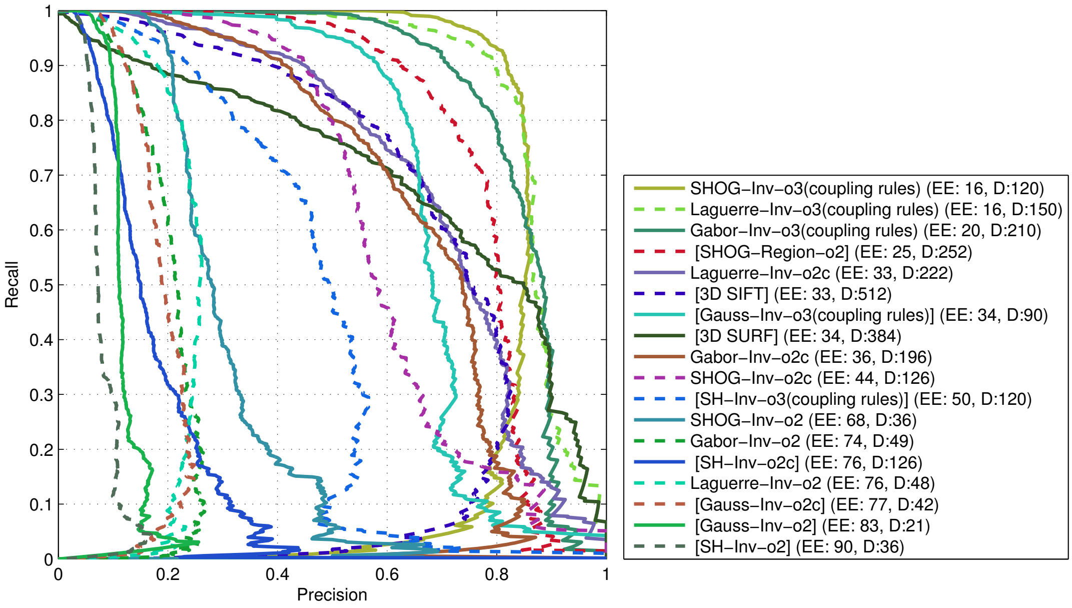


Figure E.3.: PR curves for the PSB dataset with a SNR 6.5. Classification results for the Laguerre descriptors, 3D SIFT and 3D SURF with a linear classifier and with a random forest classifier (postfix RF).

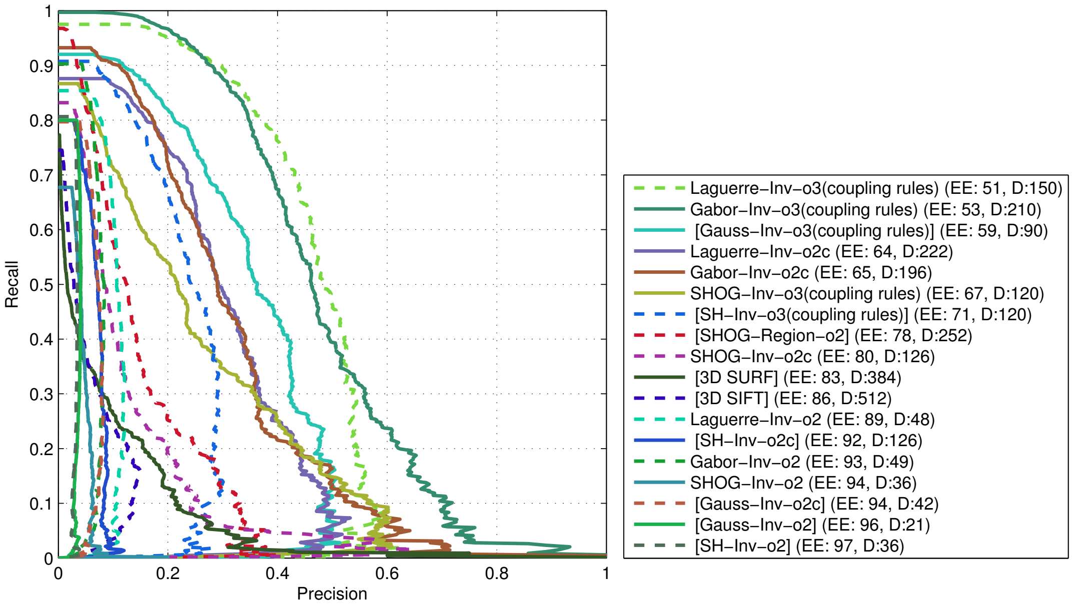


(a) tolerated displacement of 3 voxels

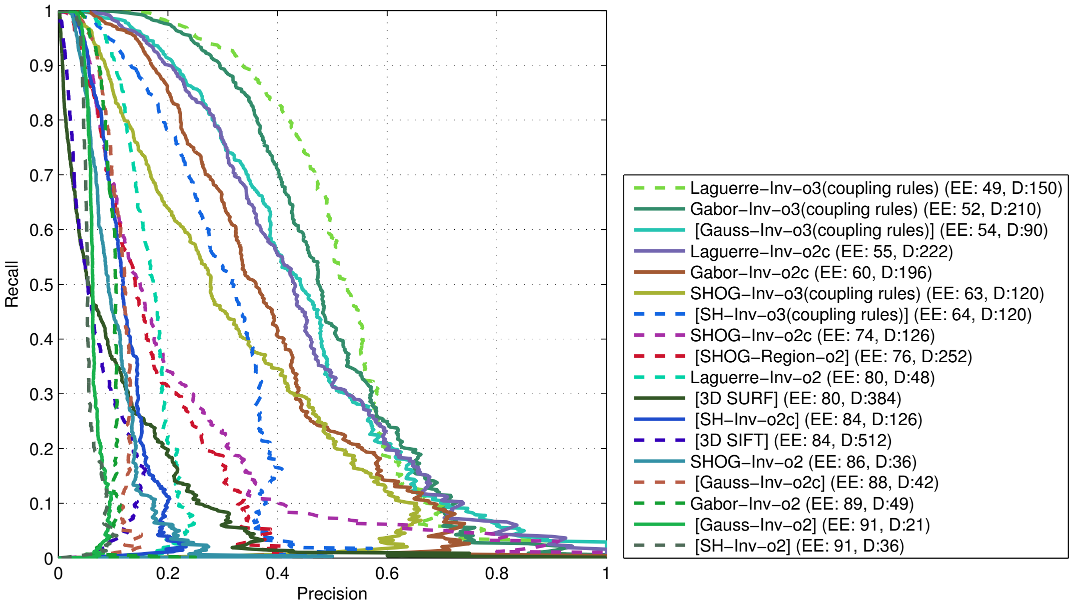


(b) tolerated displacement of 6 voxels

Figure E.4.: PR curves for the PSB dataset with a SNR 40



(a) tolerated displacement of 3 voxels



(b) tolerated displacement of 6 voxels

Figure E.5.: PR curves for the PSB dataset with a SNR 6.5

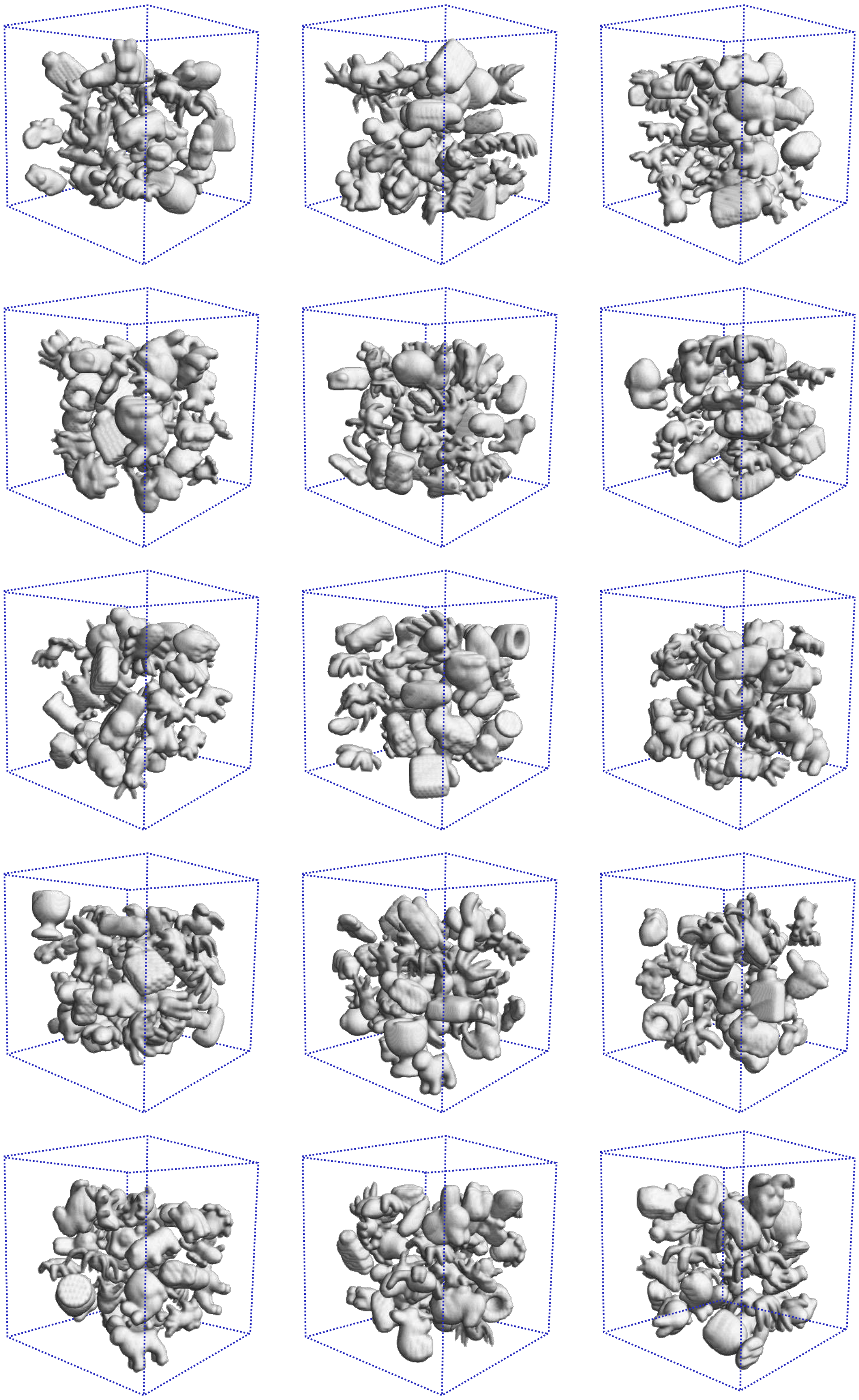
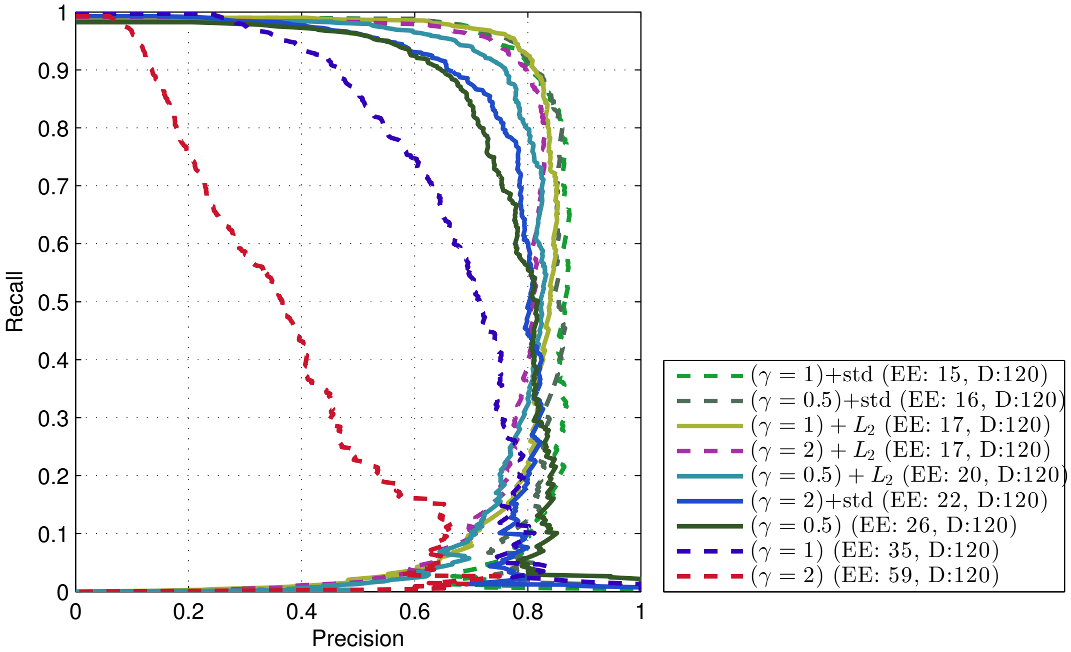
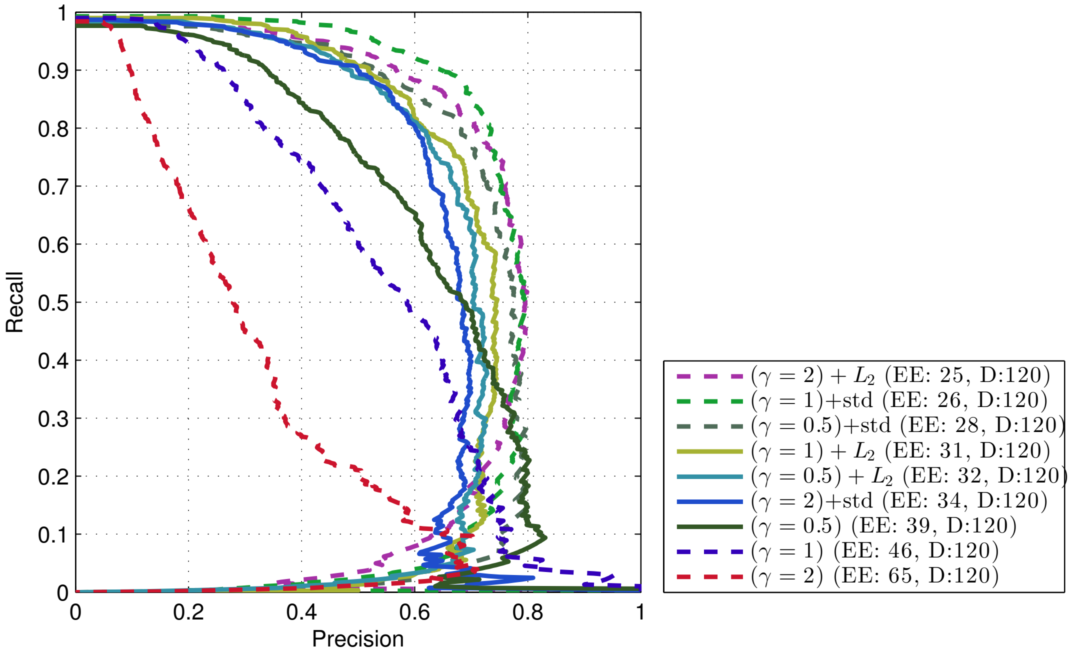


Figure E.6.: The test dataset of the PSB dataset (In total: 682 objects)

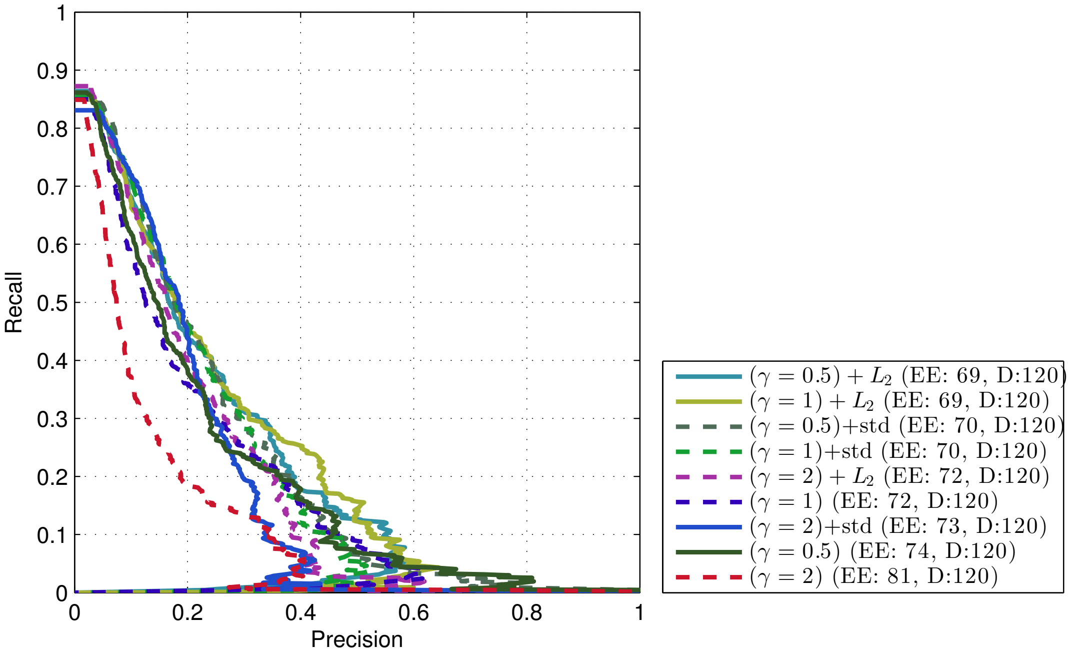


(a) SNR 40



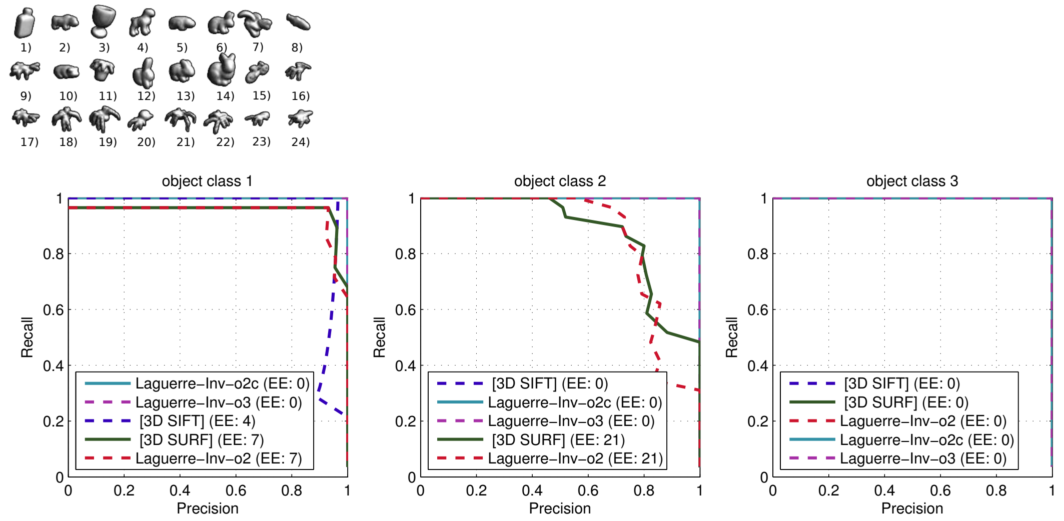
(b) SNR 20

Figure E.7.: SHOG-Inv-o3 descriptor with varying settings for γ (1 for HOG-like, 2 for generalized structure tensor-like and 0.5) in combination with no normalization, L_2, and using a normalization of the gradients with respect to the local standard deviation denoted by the postfix “std”.

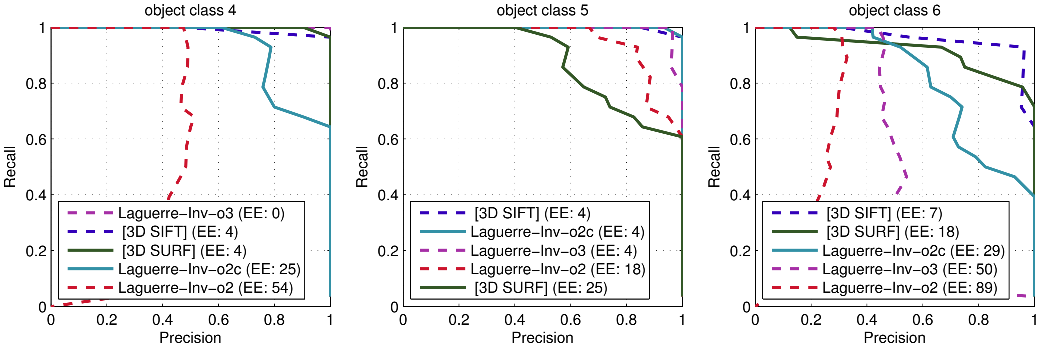


(a) SNR 6.5

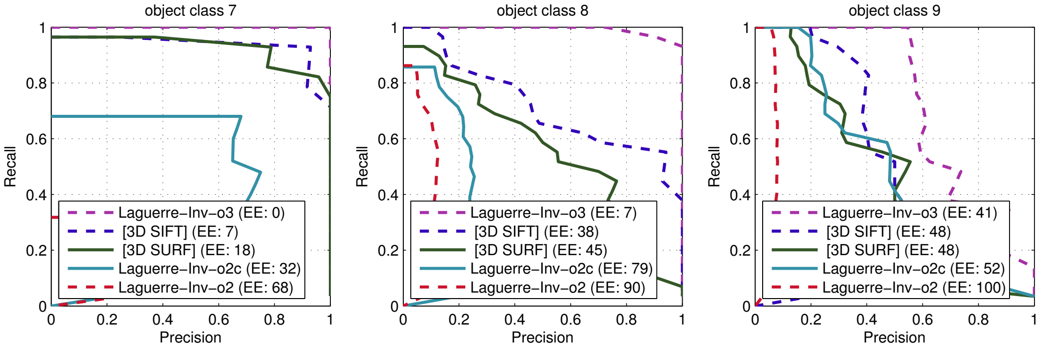
Figure E.8.: SHOG-Inv-o3 descriptor with varying settings for γ (1 for HOG-like, 2 for generalized structure tensor-like and 0.5) in combination with no normalization, L_2, and using a normalization of the gradients with respect to the local standard deviation denoted by the postfix “std”.



(a) object 1 (b) object 2 (c) object 3

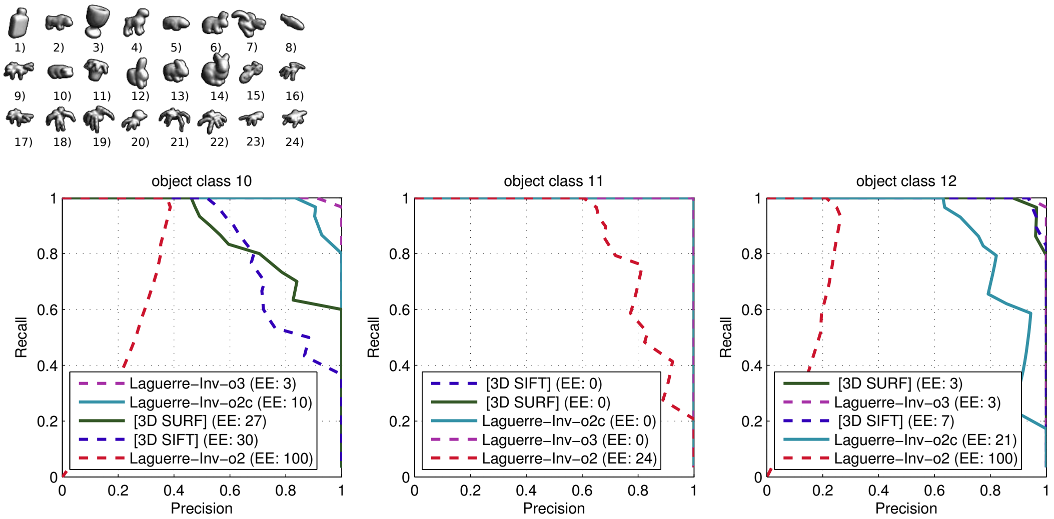


(d) object 4 (e) object 5 (f) object 6

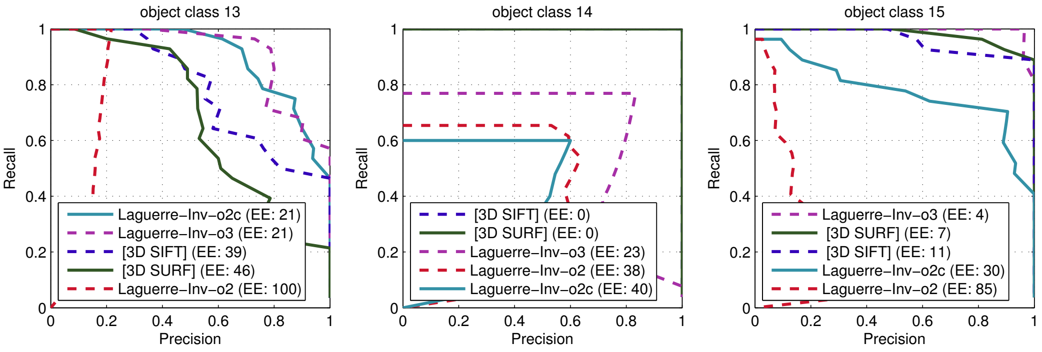


(g) object 7 (h) object 8 (i) object 9

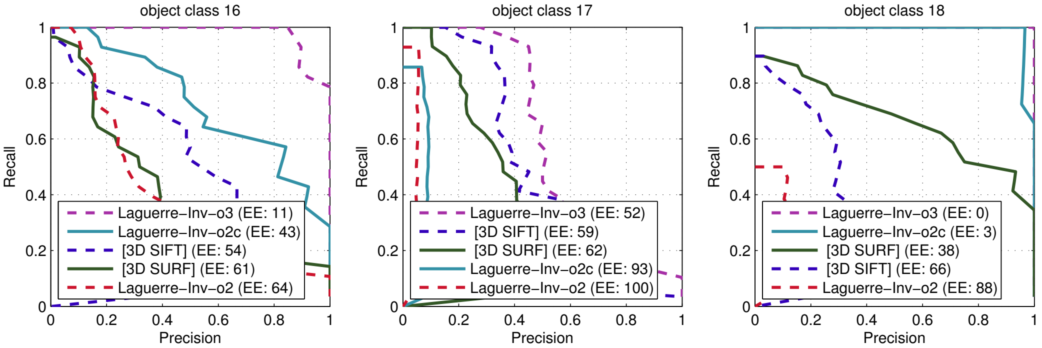
Figure E.9.: Evaluating single classes of the PSB dataset (SNR 40). Comparing Laguerre descriptors with SIFT and SURF. The bi-spectrum descriptor (o3) clearly outperforms 3D SIFT and SURF, while the power spectrum based descriptor (o2) is not discriminative enough for distinguishing all 24 classes.



(a) object 10 (b) object 11 (c) object 12

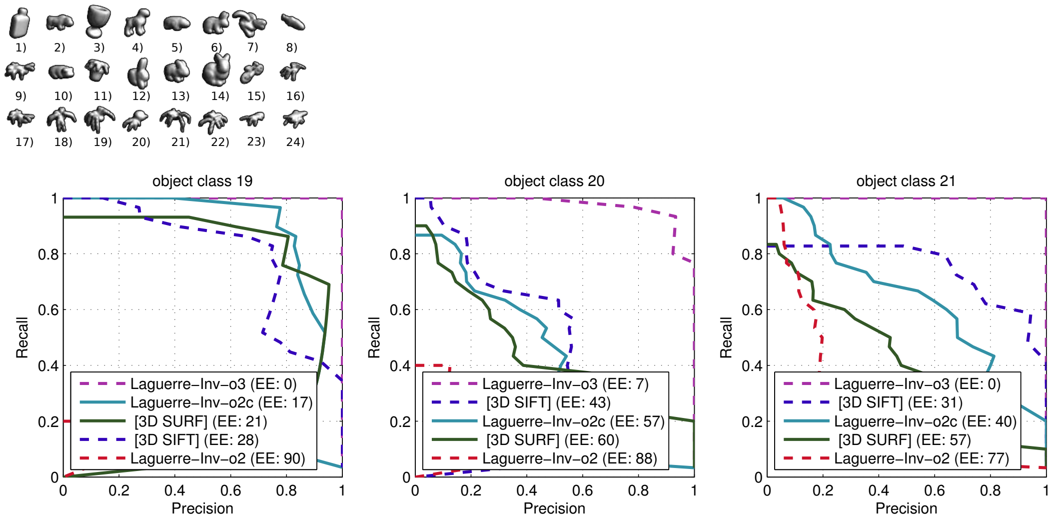


(d) object 13 (e) object 14 (f) object 15

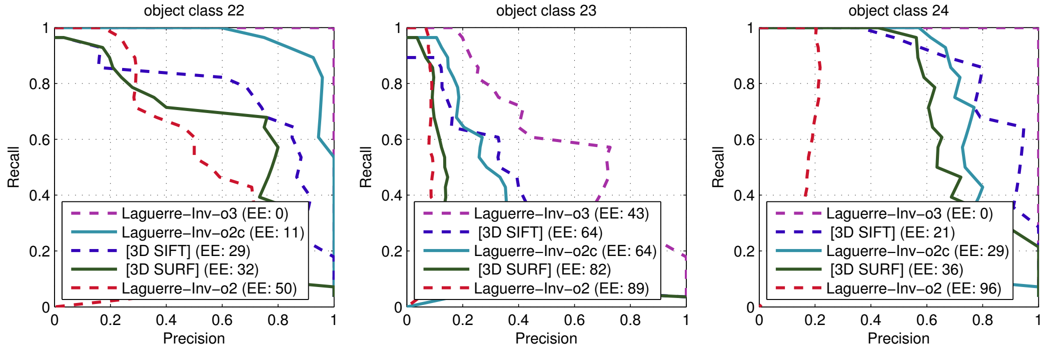


(g) object 16 (h) object 17 (i) object 18

Figure E.10.: Evaluating single classes of the PSB dataset (SNR 40). Comparing Laguerre descriptors with SIFT and SURF. The bi-spectrum descriptor (o3) clearly outperforms 3D SIFT and SURF, while the power spectrum based descriptor (o2) is not discriminative enough discriminative enough for distinguishing all 24 classes.



(a) object 19 (b) object 20 (c) object 21



(d) object 22 (e) object 23 (f) object 24

Figure E.11.: Evaluating single classes of the PSB dataset (SNR 40). Comparing Laguerre descriptors with SIFT and SURF. The bi-spectrum descriptor (o3) clearly outperforms 3D SIFT and SURF, while the power spectrum based descriptor (o2) is not discriminative enough discriminative enough for distinguishing all 24 classes.

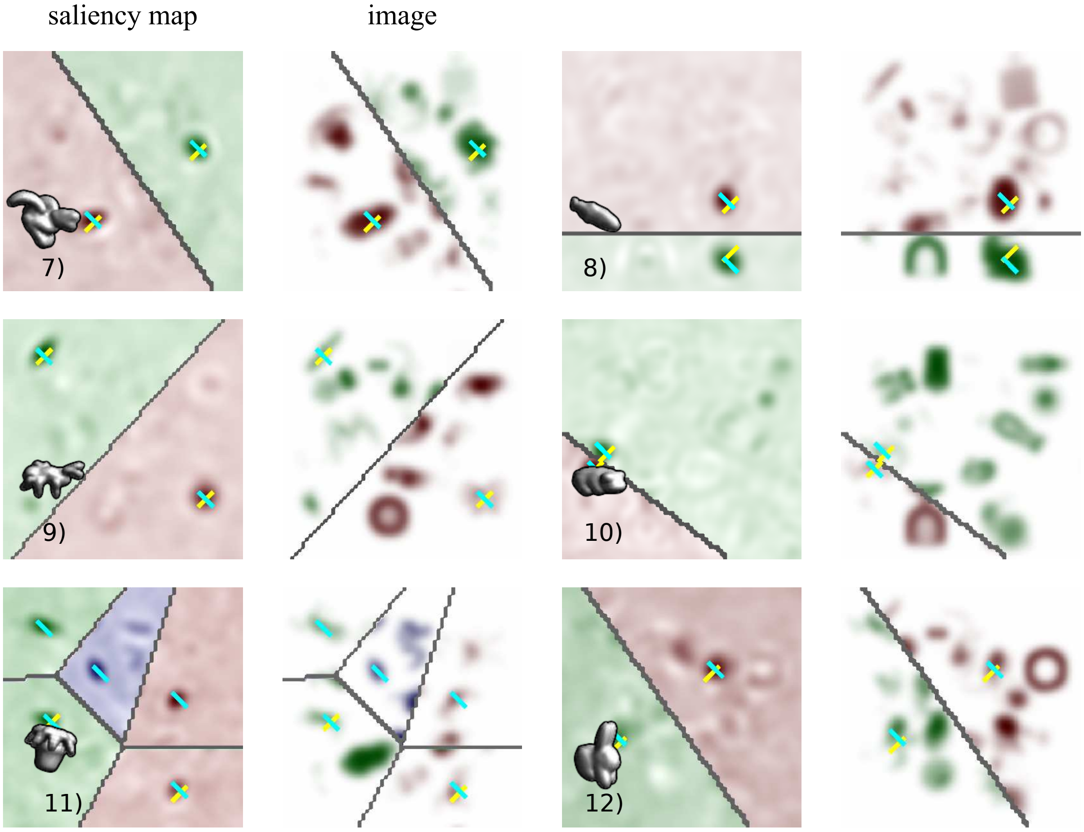


Figure E.12.: Detection of single classes of the PSB dataset (SNR 40) using the Laguerre descriptor based on bi-spectrum invariants. The slice close to a detection or ground-truth label is shown for both the filter response (saliency map, darker represents higher values for better readability) and the image (without noise and absorption). Yellow lines are indicating the ground truth labels, magenta lines detections (after classification). Results for the categories 7-12 are shown. The corresponding PR curves can be found on pages 244–245. The filter responses for the first six classes on page 168

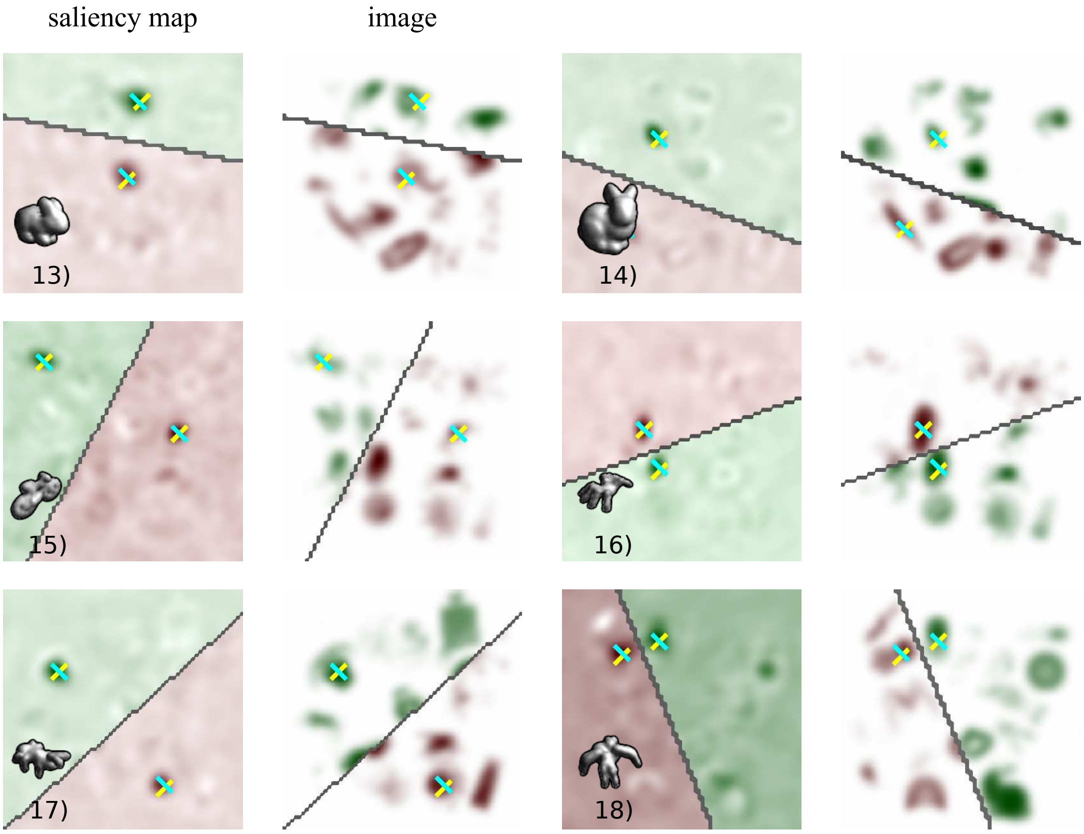


Figure E.13.: Detection of single classes of the PSB dataset (SNR 40) using the Laguerre descriptor based on bi-spectrum invariants. The slice close to a detection or ground-truth label is shown for both the filter response (saliency map, darker represents higher values for better readability) and the image (without noise and absorption). Yellow lines are indicating the ground truth labels, magenta lines detections (after classification). Results for the categories 13-18 are shown. The corresponding PR curves can be found on pages 245–246.

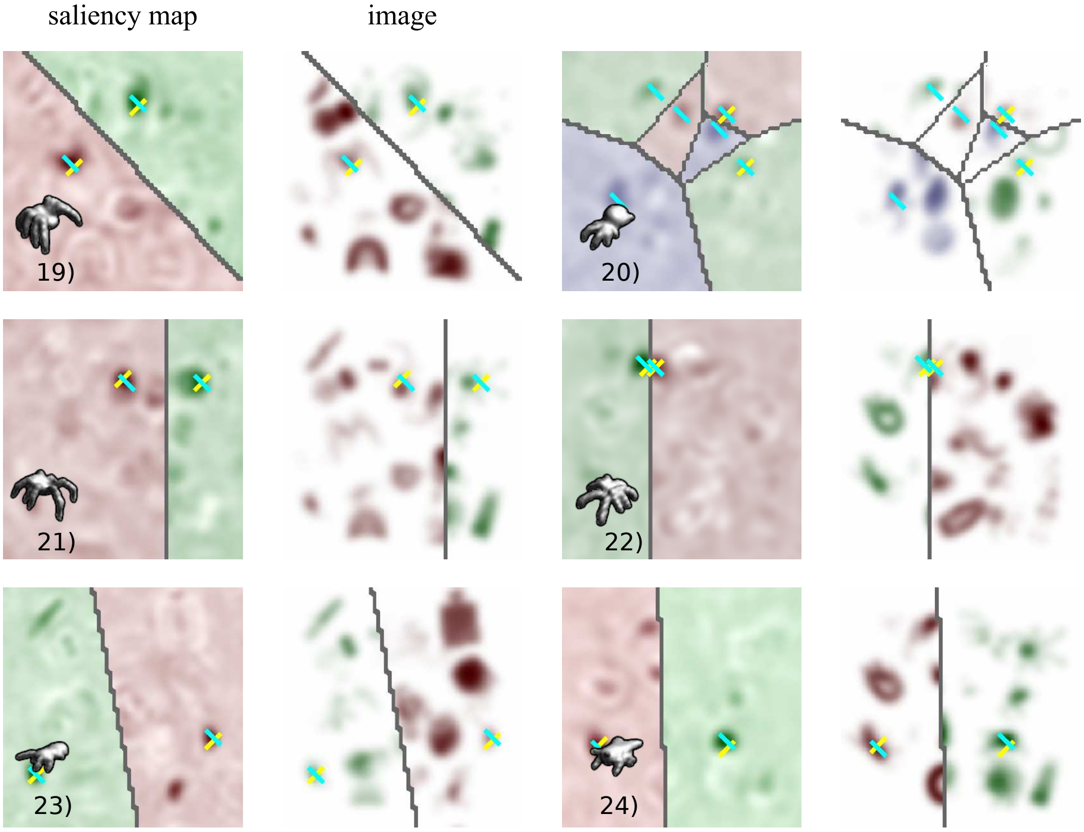
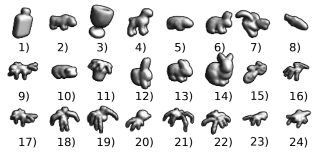
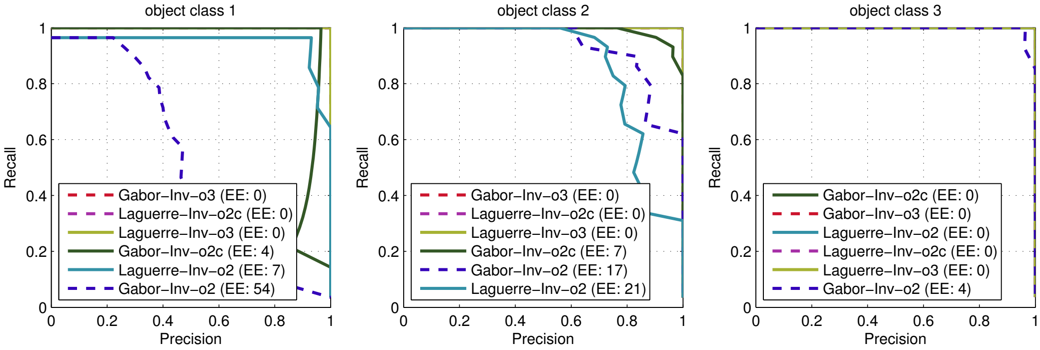
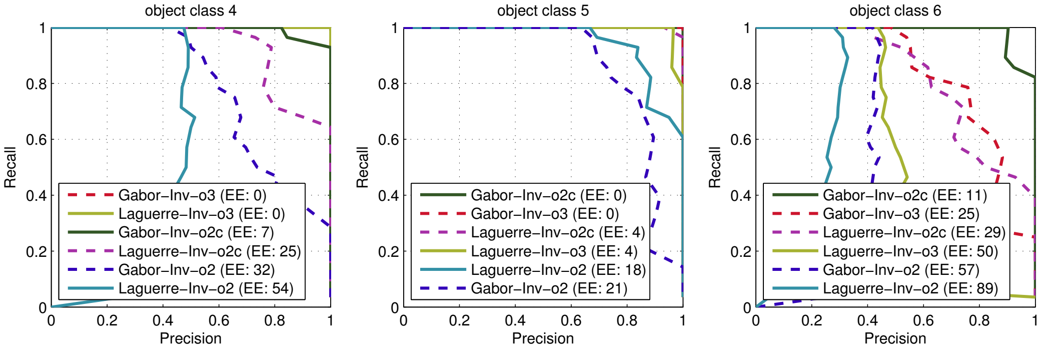


Figure E.14.: Detection of single classes of the PSB dataset (SNR 40) using the Laguerre descriptor based on bi-spectrum invariants. The slice close to a detection or ground-truth label is shown for both the filter response (saliency map, darker represents higher values for better readability) and the image (without noise and absorption). Yellow lines are indicating the ground truth labels, magenta lines detections (after classification). Results for the categories 18-24 are shown. The corresponding PR curves can be found on page 246.

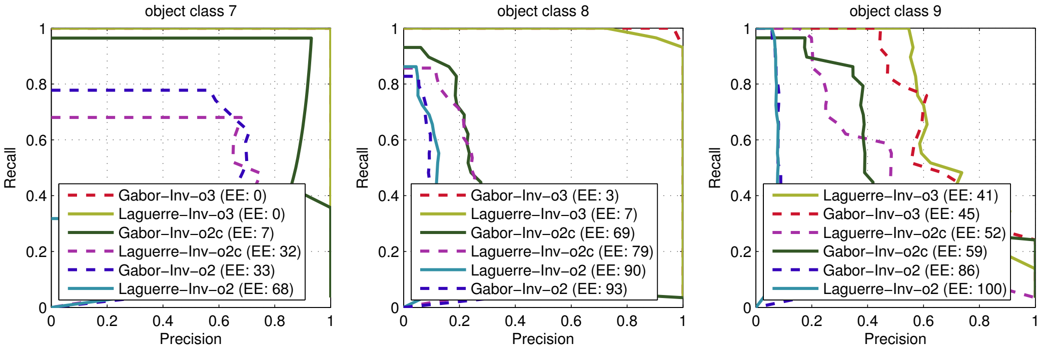




(a) object 1 (b) object 2 (c) object 3

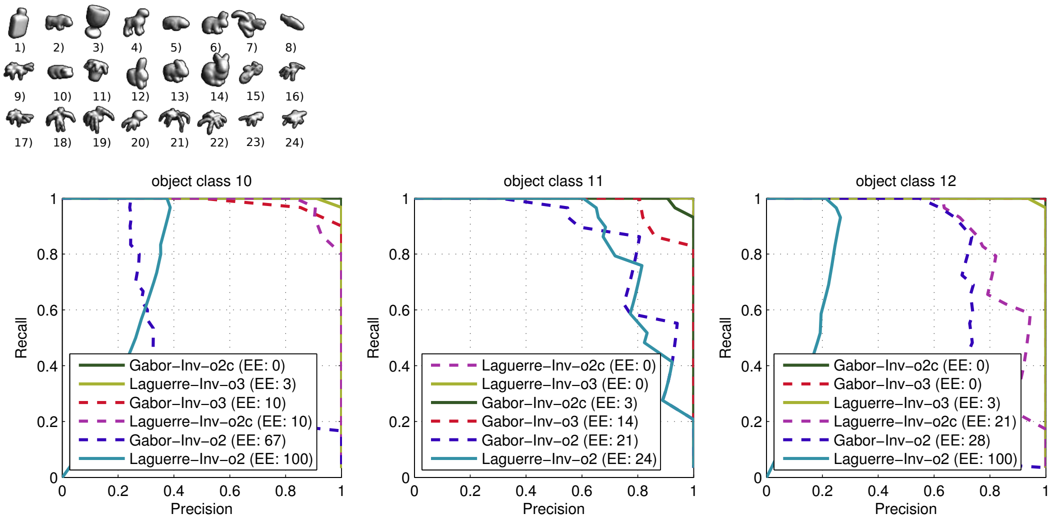


(d) object 4 (e) object 5 (f) object 6

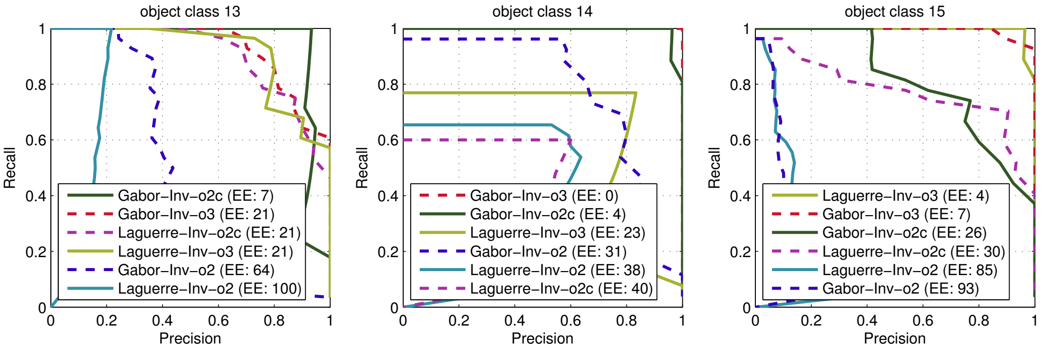


(g) object 7 (h) object 8 (i) object 9

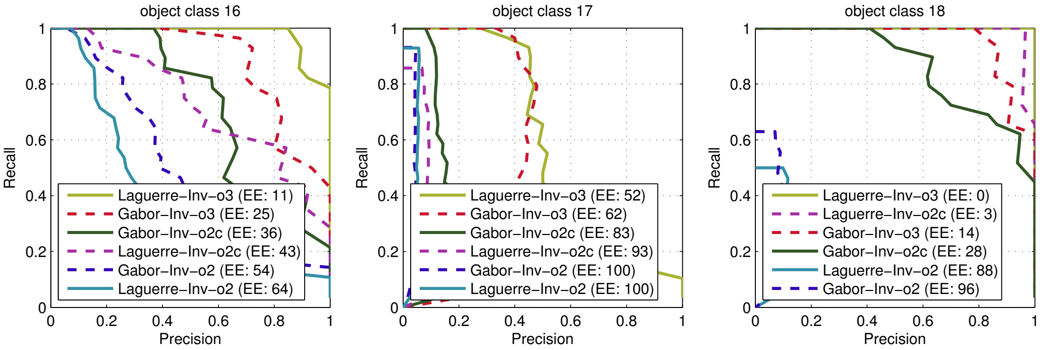
Figure E.15.: Evaluating single classes of the PSB dataset (SNR 40). Comparing Laguerre descriptors with Gabor descriptors. We observed that Gabor confounded the classes (16-22) which manly differ in the leg poses and leg lengths. Laguerre confounded the classes 6,13 and 14, which all are rabbit shaped images. Overall, Laguerre performed sightly better than Gabor.



(a) object 10 (b) object 11 (c) object 12

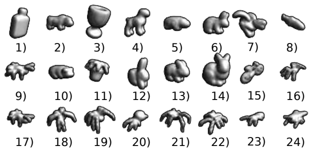


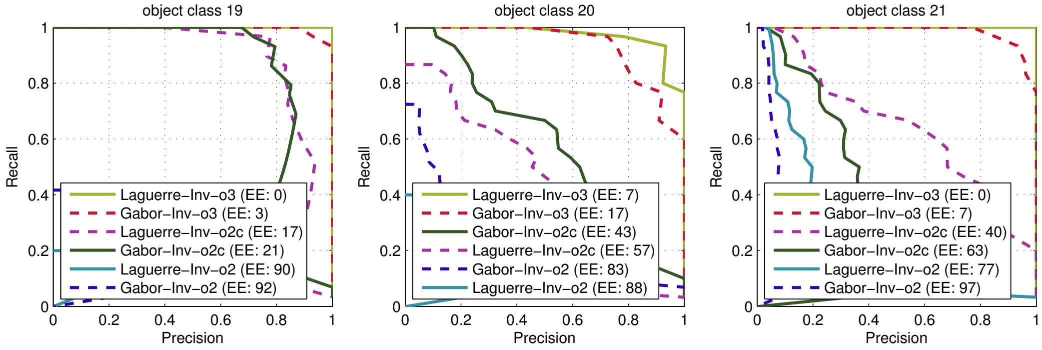
(d) object 13 (e) object 14 (f) object 15



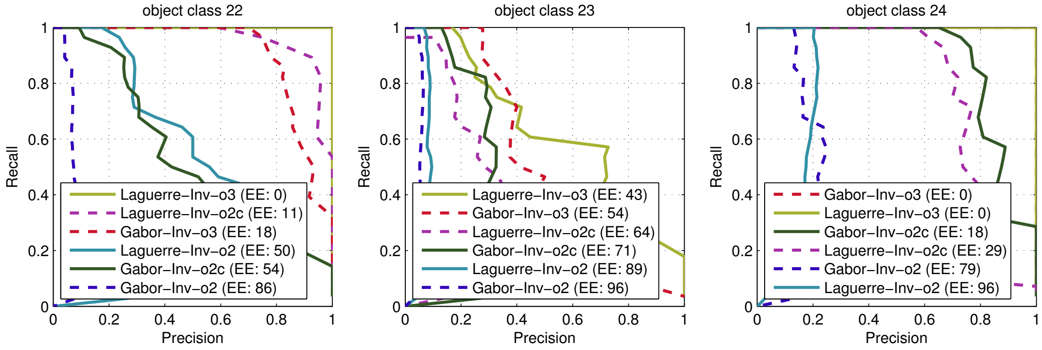
(g) object 16 (h) object 17 (i) object 18

Figure E.16.: Evaluating single classes of the PSB dataset (SNR 40). Comparing Laguerre descriptors with Gabor descriptors. We observed that Gabor confounded the classes (16-22) which manly differ in the leg poses and leg lengths. Laguerre confounded the classes 6,13 and 14, which all are rabbit shaped images. Overall, Laguerre performed sightly better than Gabor.





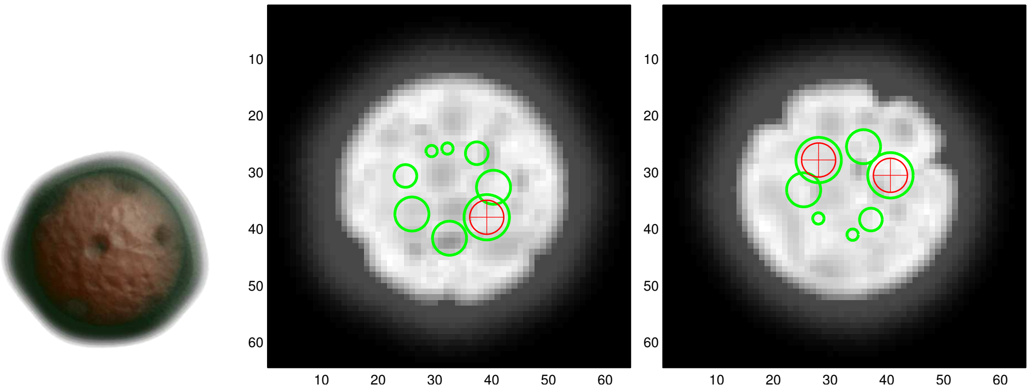
(a) object 19 (b) object 20 (c) object 21



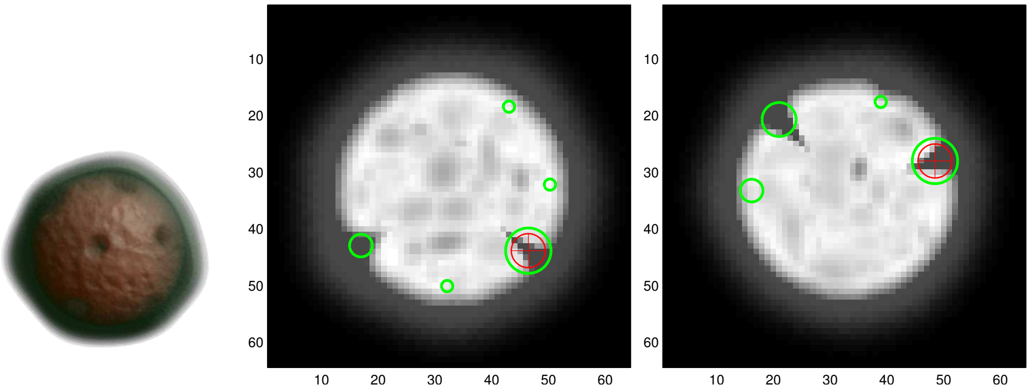
(d) object 22 (e) object 23 (f) object 24

Figure E.17.: Evaluating single classes of the PSB dataset (SNR 40). Comparing Laguerre descriptors with Gabor descriptors. We observed that Gabor confounded the classes (16-22) which manly differ in the leg poses and leg lengths. Laguerre confounded the classes 6,13 and 14, which all are rabbit shaped images. Overall, Laguerre performed sightly better than Gabor.

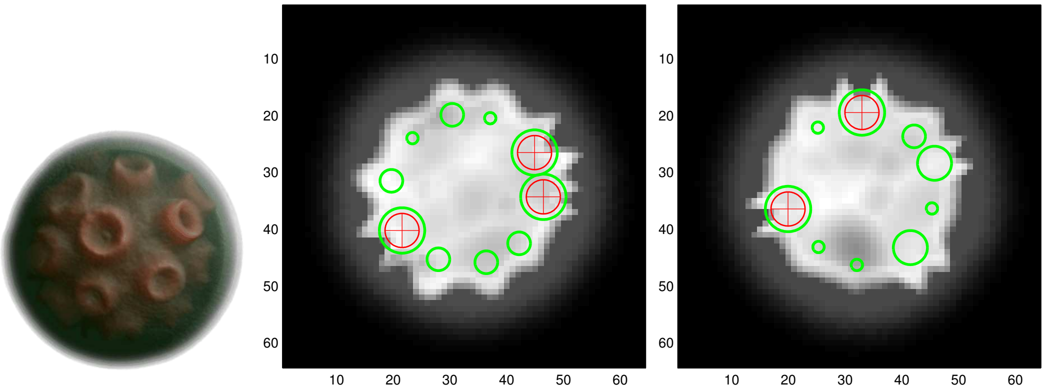
E.2. Plots and Figures: Simulated Pollen Dataset



(a) landmark 1 (b) landmark 1 (c) landmark 1

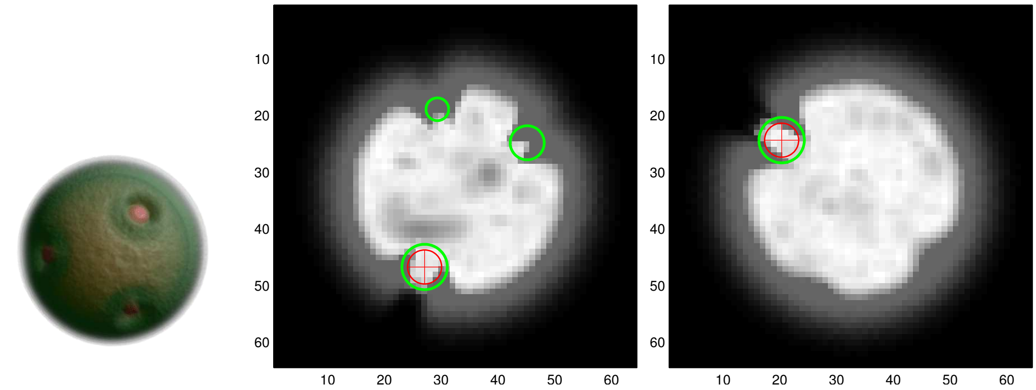


(d) landmark 2 (e) landmark 2 (f) landmark 2

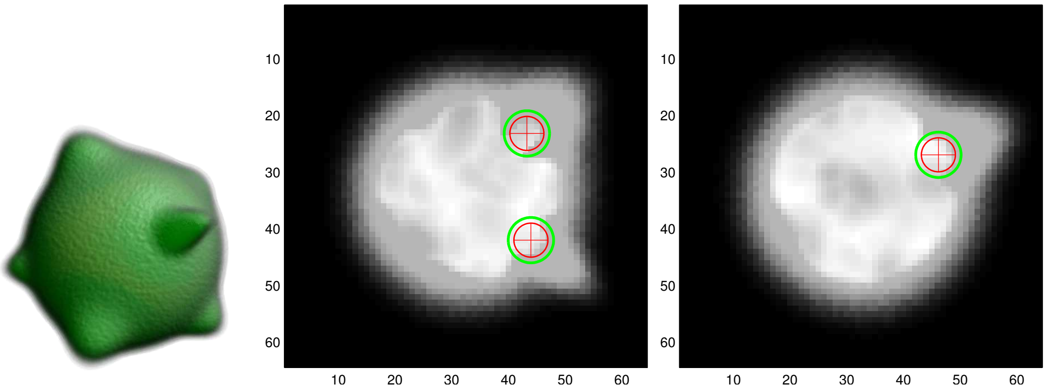


(g) landmark 3 (h) landmark 3 (i) landmark 3

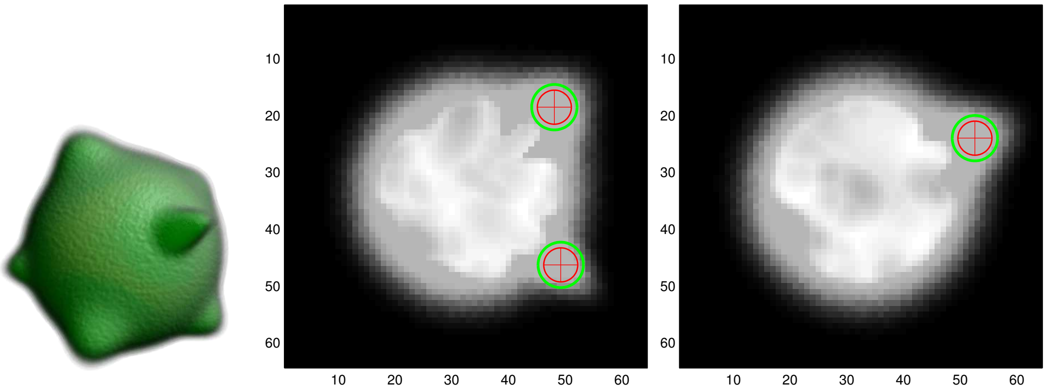
Figure E.18.: We exemplarily show two slices for each landmark of the simulated pollen database. The size of the green circle indicates the distance to a landmark (if close to the current slice). A red circle indicates that the landmark is in the current slice.



(a) landmark 4 (b) landmark 4 (c) landmark 4

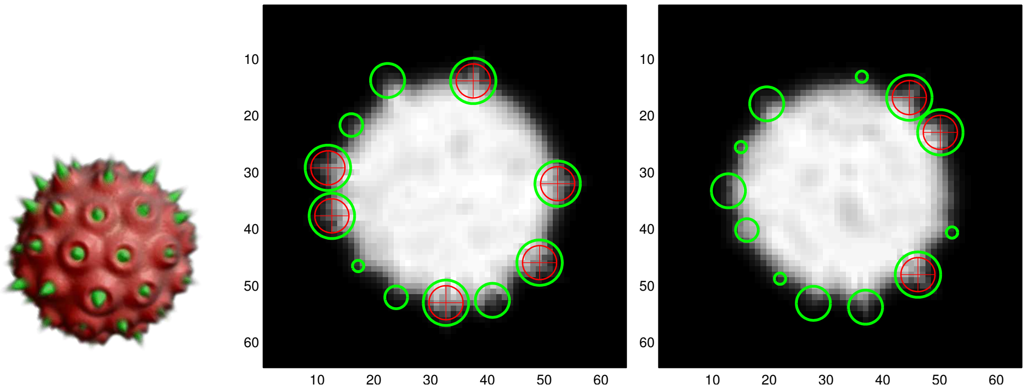


(d) landmark 5 (e) landmark 5 (f) landmark 5

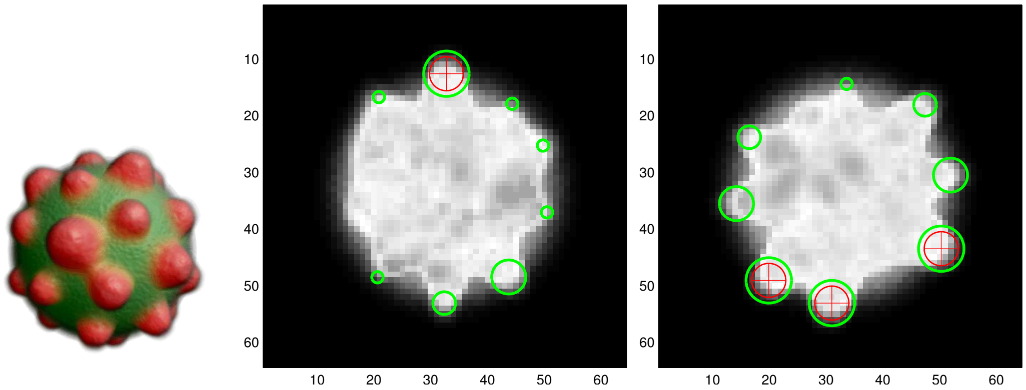


(g) landmark 6 (h) landmark 6 (i) landmark 6

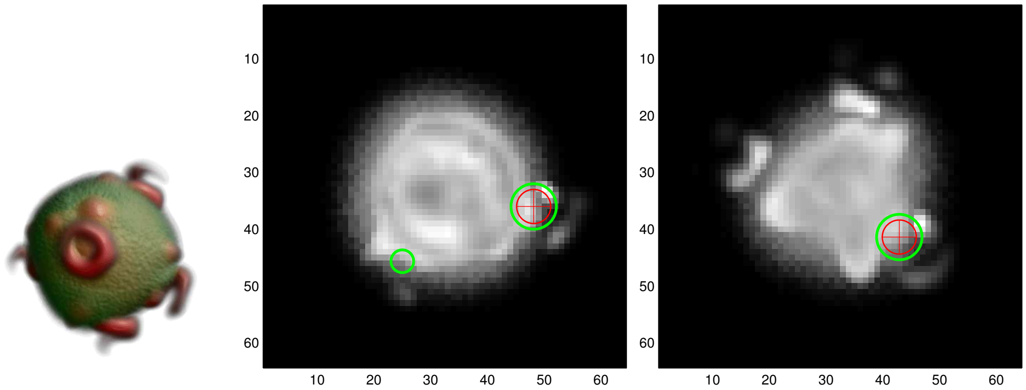
Figure E.19.: We exemplarily show two slices for each landmark of the simulated pollen database. The size of the green circle indicates the distance to a landmark (if close to the current slice). A red circle indicates that the landmark is in the current slice.



(a) landmark 7 (b) landmark 7 (c) landmark 7

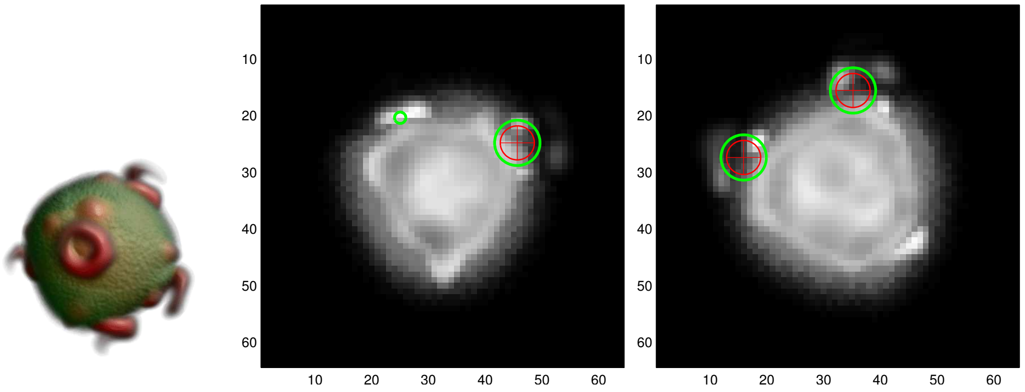


(d) landmark 8 (e) landmark 8 (f) landmark 8



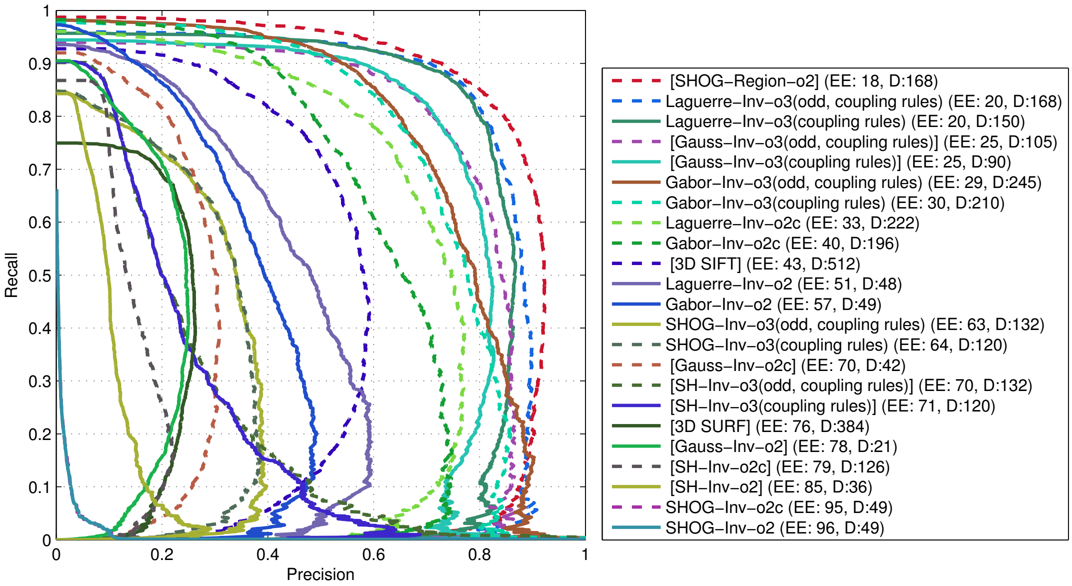
(g) landmark 9 (h) landmark 9 (i) landmark 9

Figure E.20.: We exemplarily show two slices for each landmark of the simulated pollen database. The size of the green circle indicates the distance to a landmark (if close to the current slice). A red circle indicates that the landmark is in the current slice.

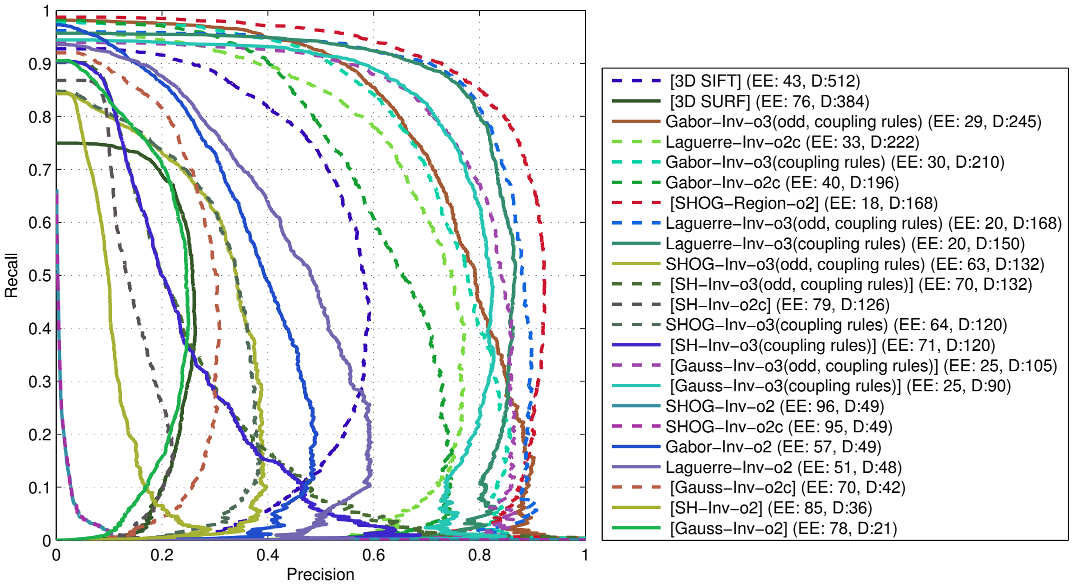


(a) landmark 10 (b) landmark 10 (c) landmark 10

Figure E.21.: We exemplarily show two slices for each landmark of the simulated pollen database. The size of the green circle indicates the distance to a landmark (if close to the current slice). A red circle indicates that the landmark is in the current slice.

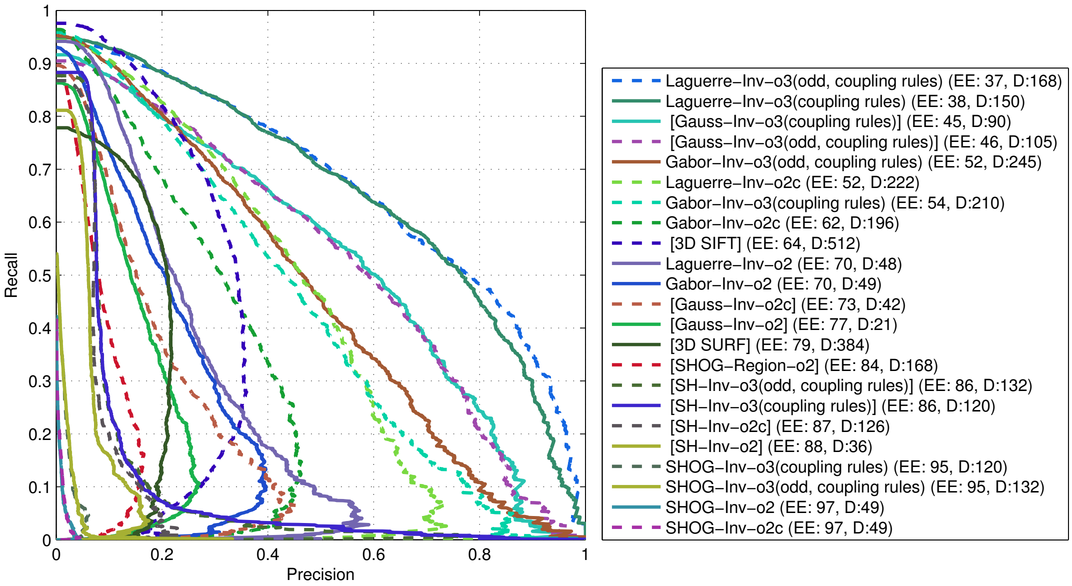


(a) ordering: EER

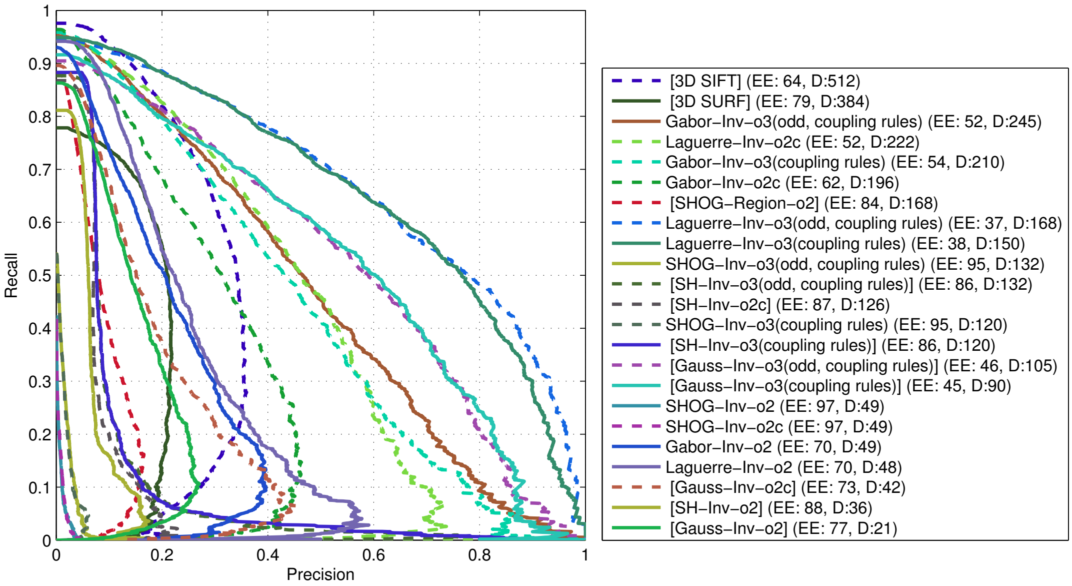


(b) ordering: Dimension

Figure E.22.: PR curves for the simulated pollen database with a SNR 40

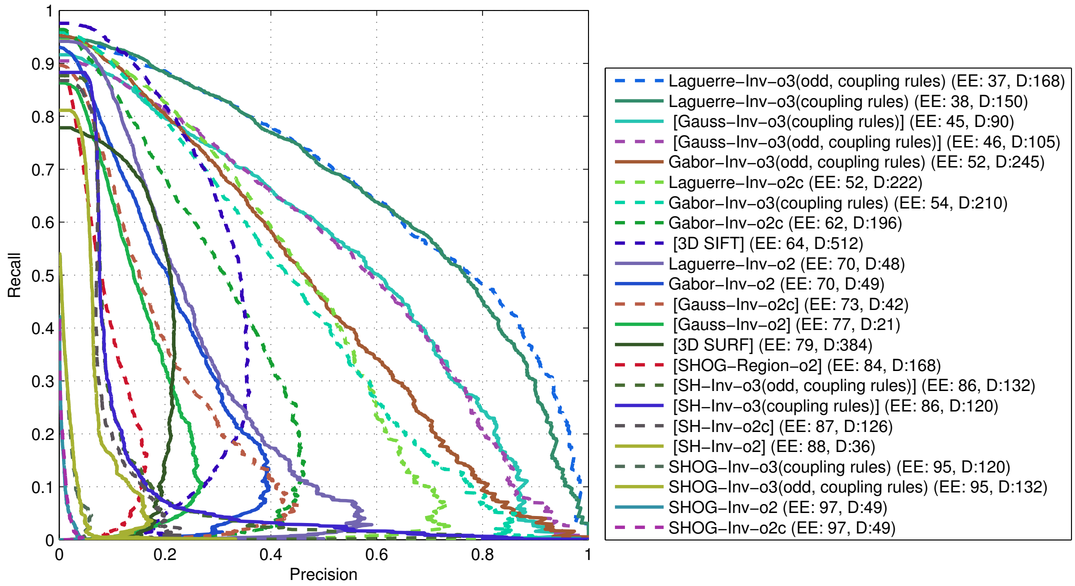


(a) ordering: EER

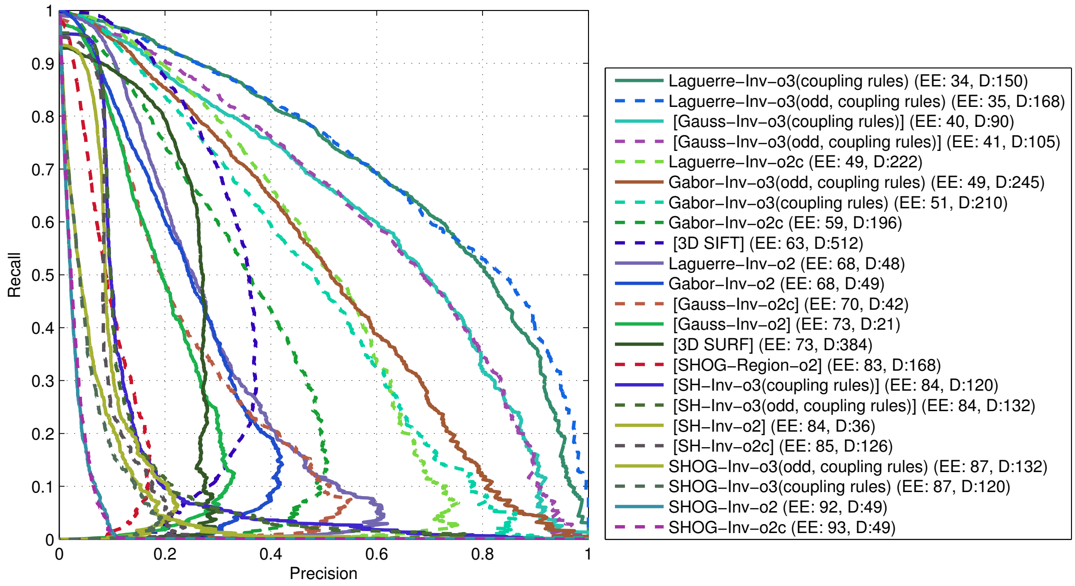


(b) ordering: Dimension

Figure E.23.: PR curves for the simulated pollen database with a SNR 6.5

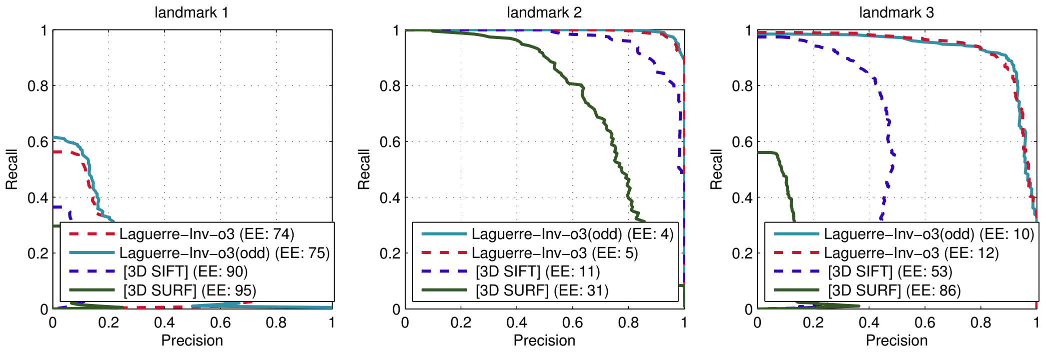


(a) tolerated displacement of 3 voxels

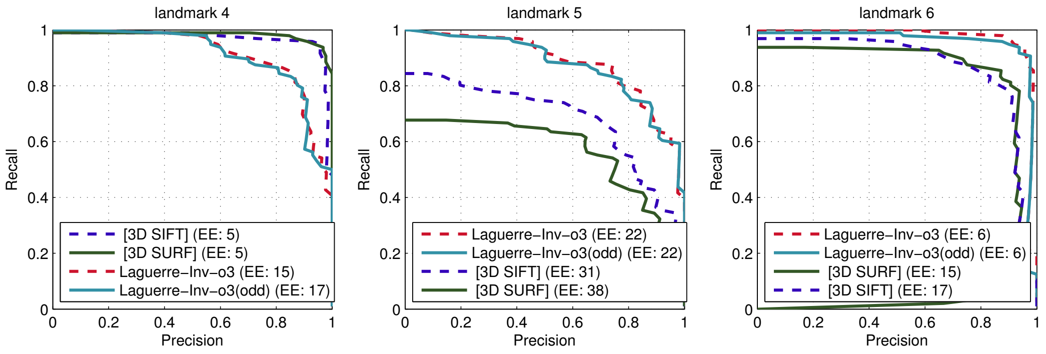


(b) tolerated displacement of 6 voxels

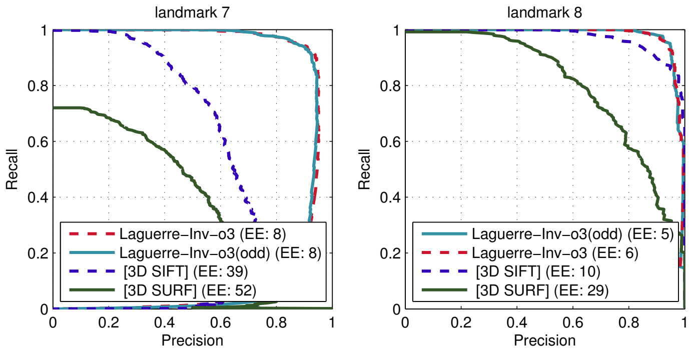
Figure E.24.: PR curves for the simulated pollen database with a SNR 6.5



(a) landmark 1 (b) landmark 2 (c) landmark 3

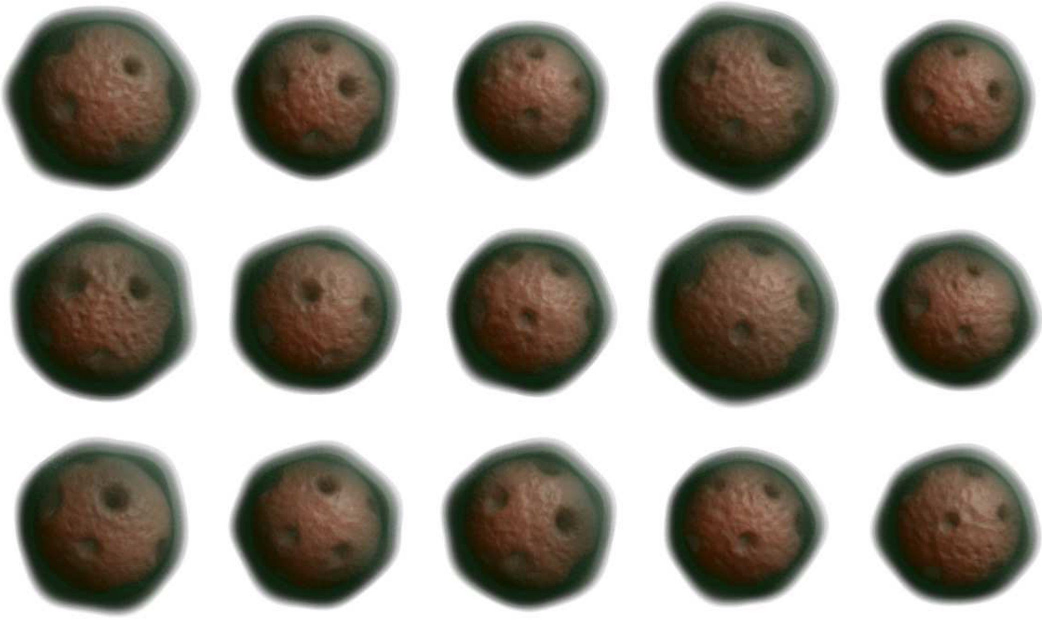


(d) landmark 4 (e) landmark 5 (f) landmark 6

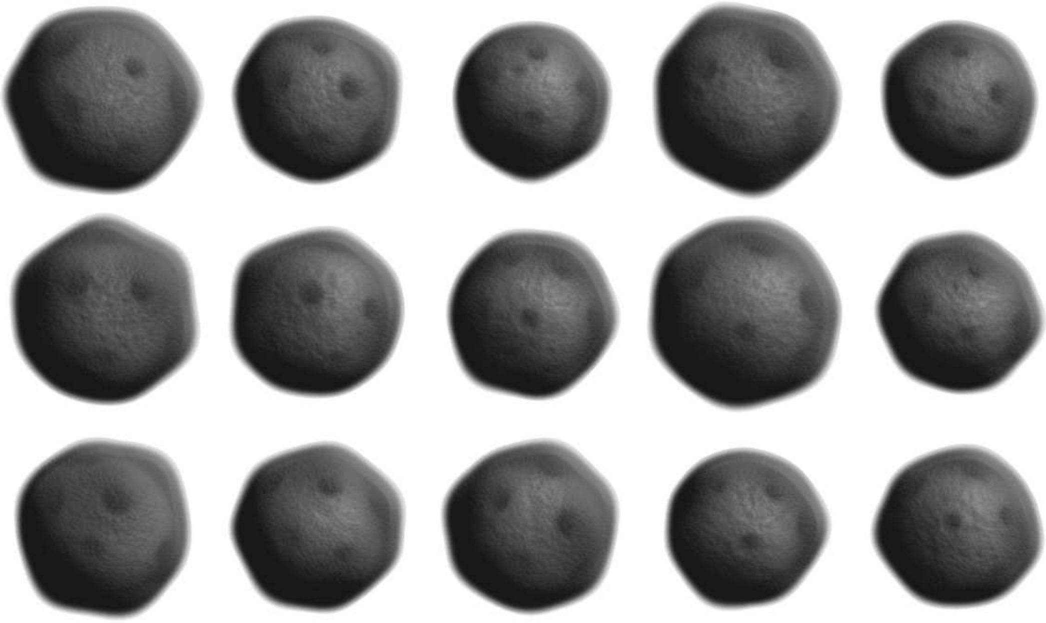


(g) landmark 7 (h) landmark 8

Figure E.25.: Evaluating single classes of the simulated pollen dataset (SNR 40). Comparing Laguerre descriptors with SIFT and SURF. The bi-spectrum descriptor (o3) clearly outperforms 3D SIFT and SURF.



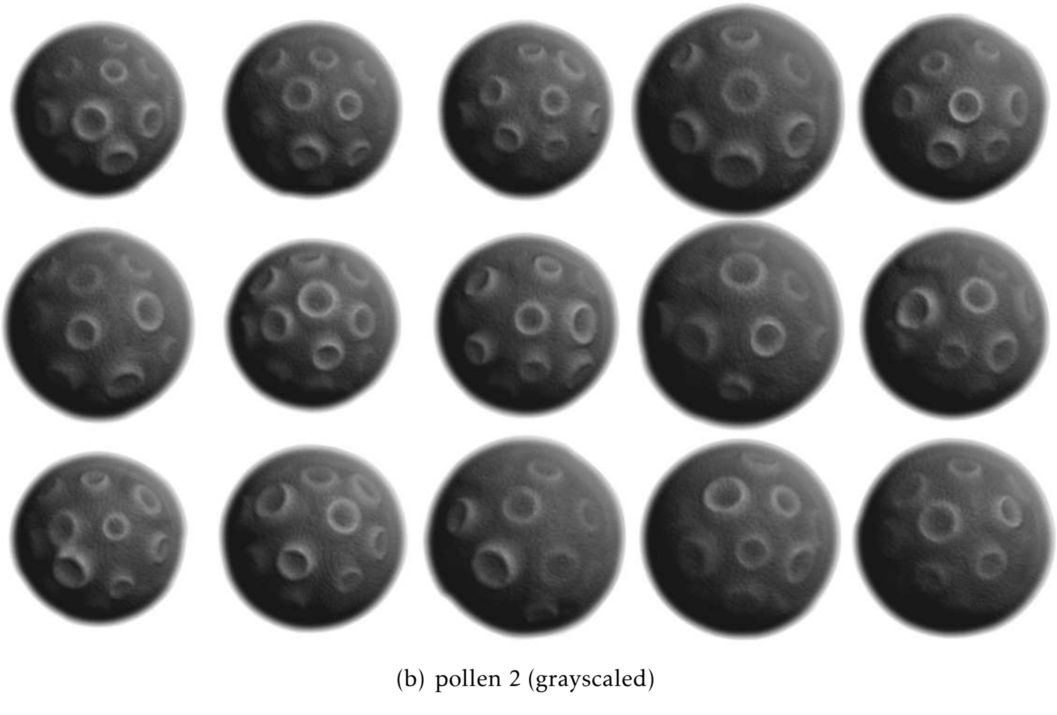
(a) pollen 1 (generated images)



(b) pollen 1 (grayscaled)

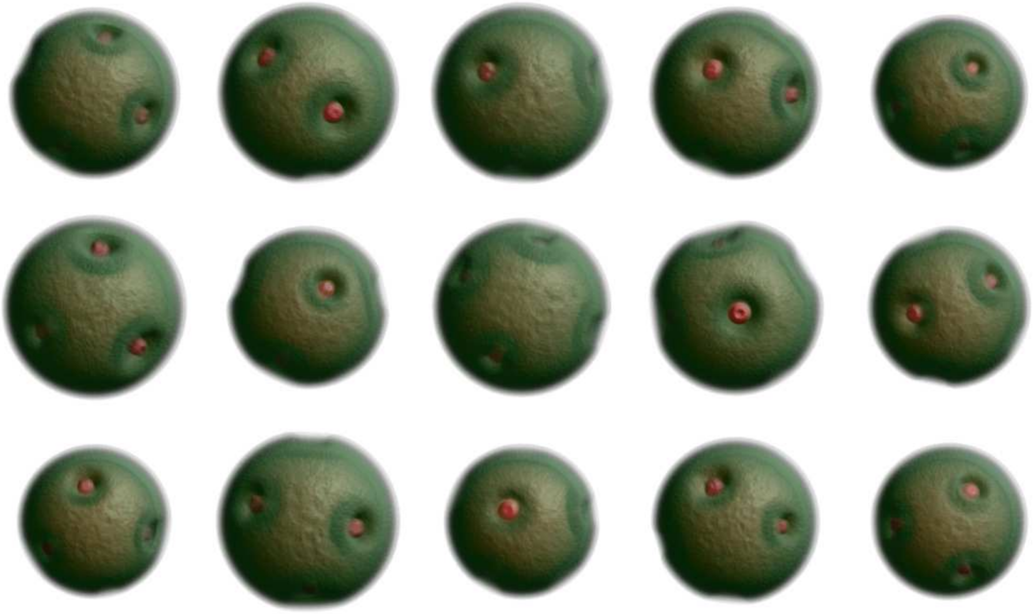
Figure E.26.: The Simulated Pollen Dataset

(a) pollen 2 (generated images)

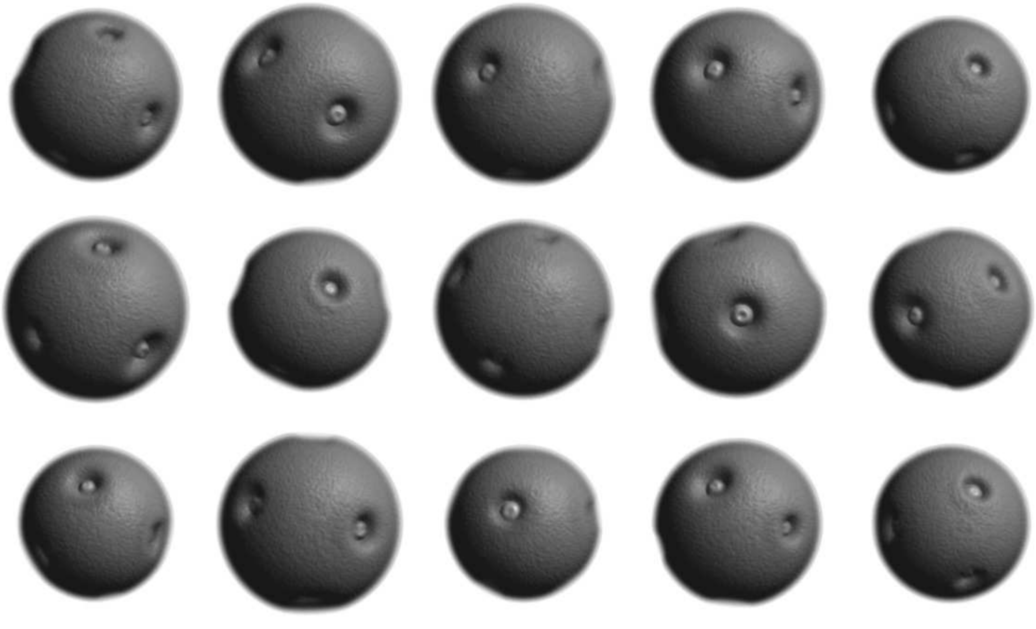


(b) pollen 2 (grayscaled)

Figure E.27.: The Simulated Pollen Dataset

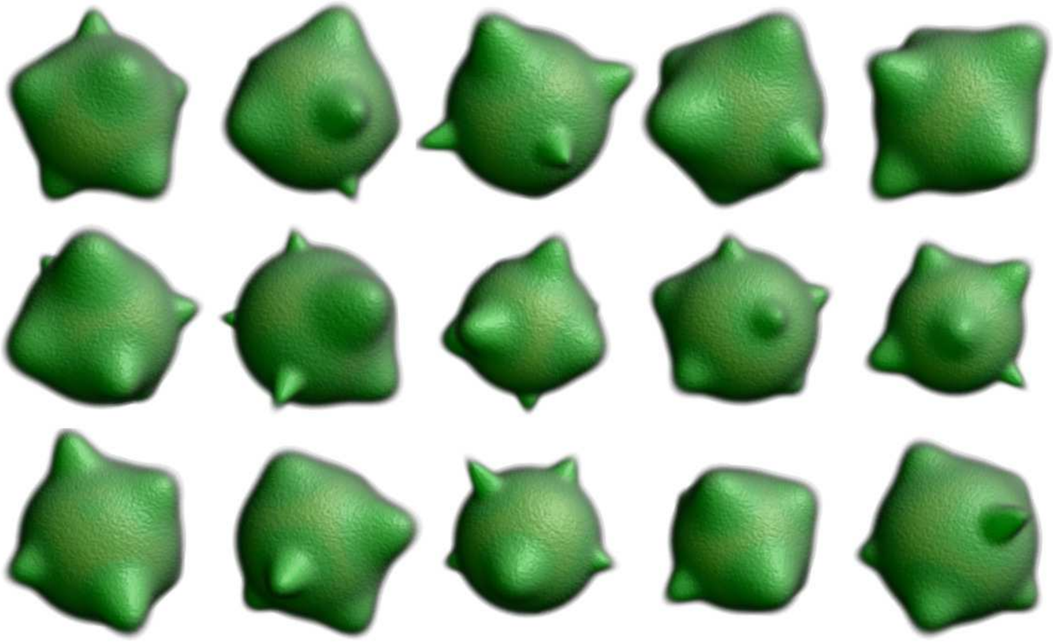


(a) pollen 3 (generated images)

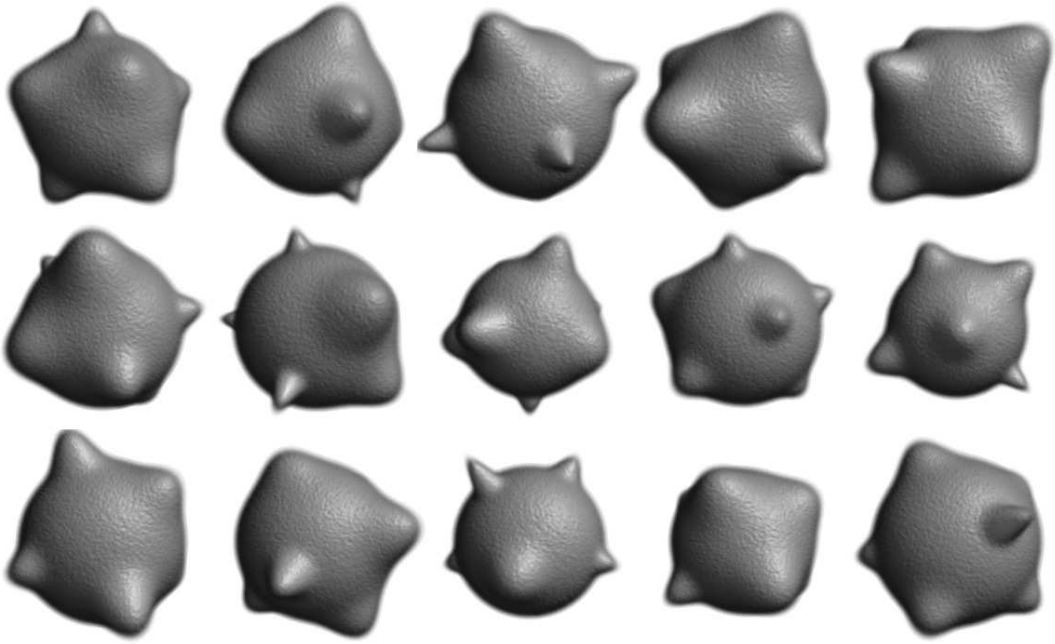


(b) pollen 3 (grayscaled)

Figure E.28.: The Simulated Pollen Dataset

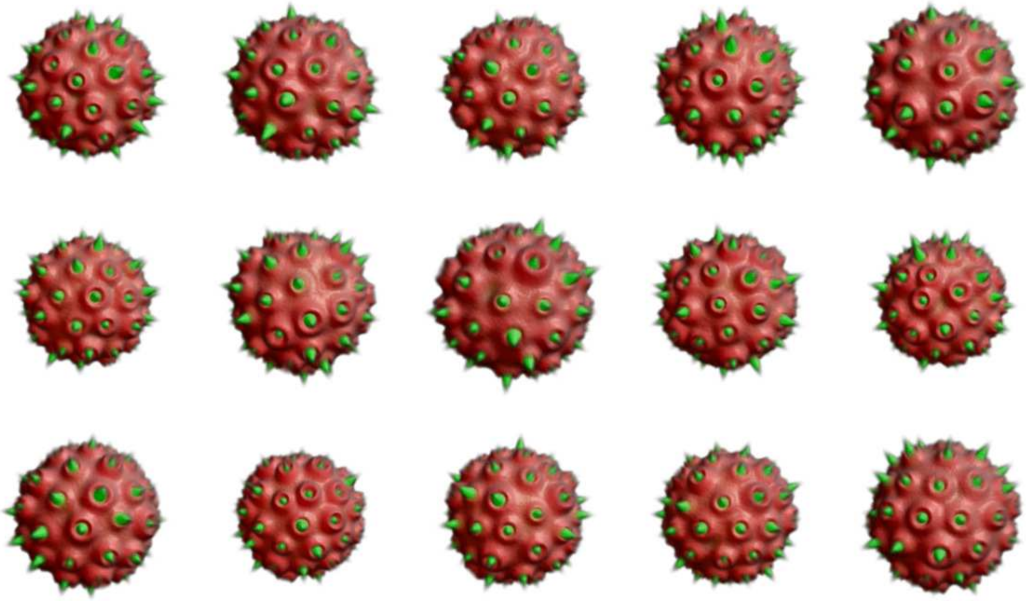


(a) pollen 4 (generated images)

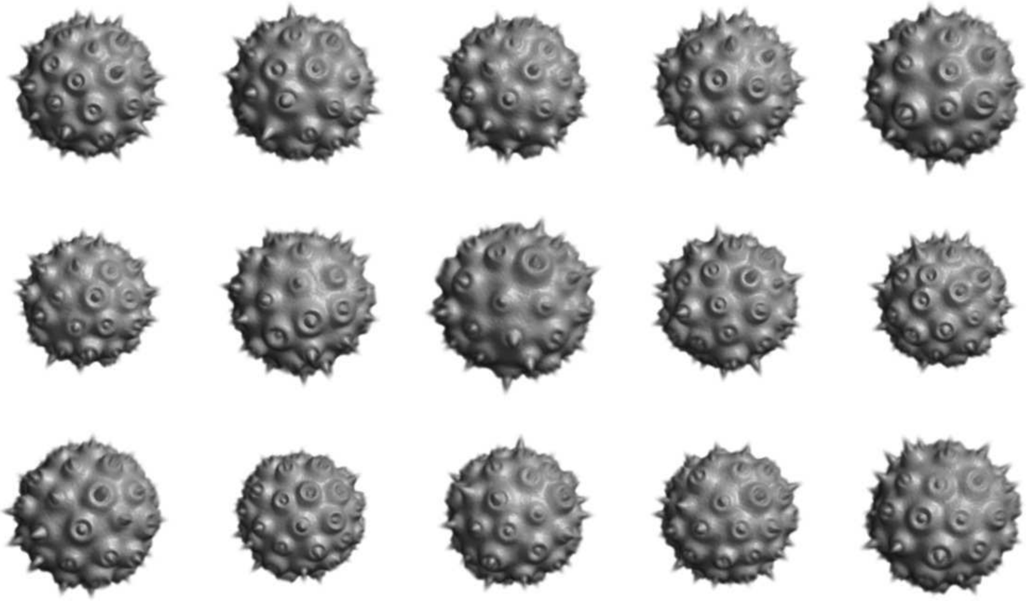


(b) pollen 4 (grayscaled)

Figure E.29.: The Simulated Pollen Dataset

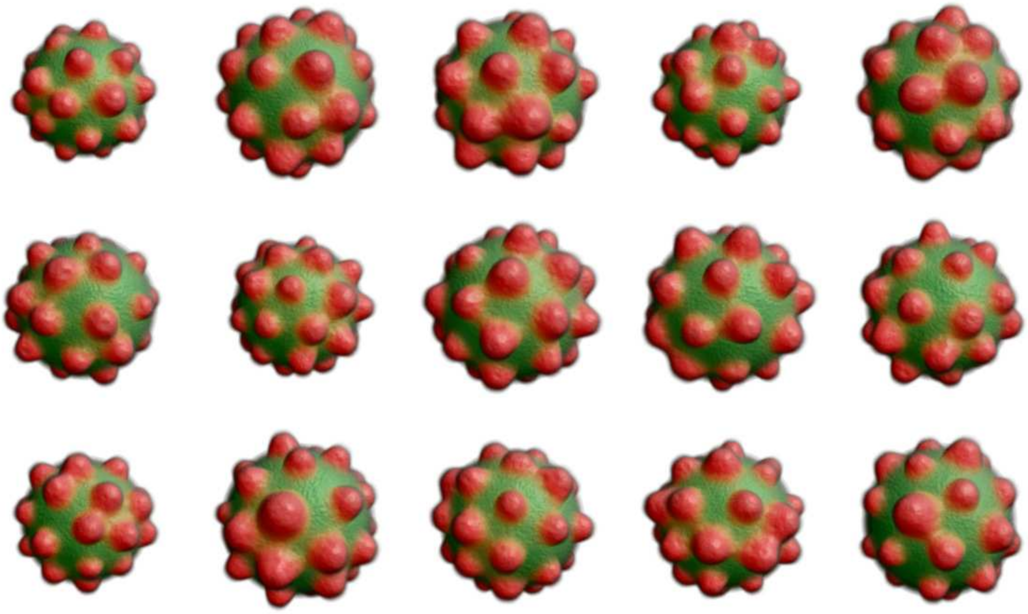


(a) pollen 5 (generated images)

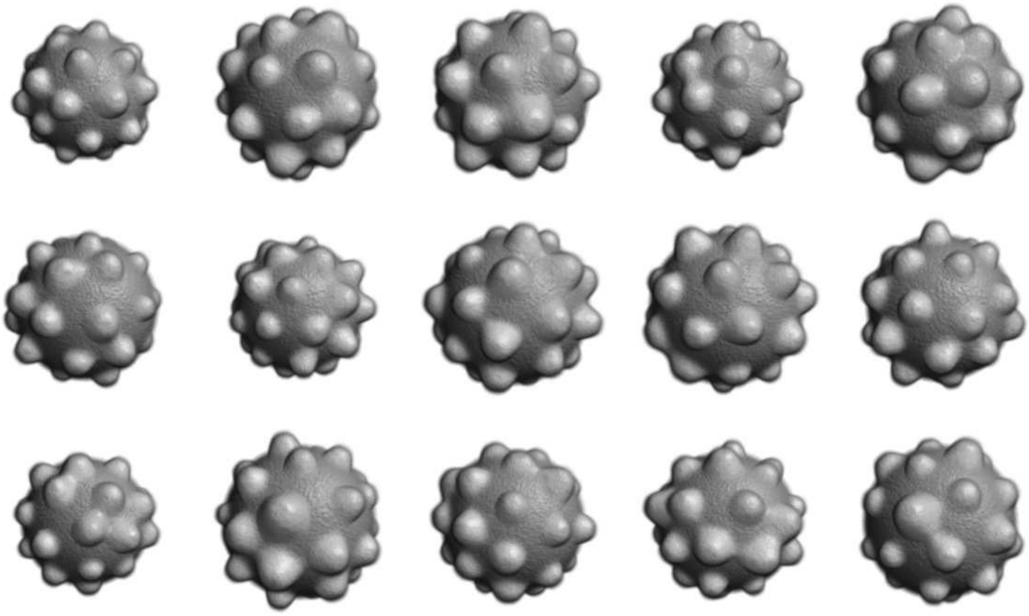


(b) pollen 5 (grayscaled)

Figure E.30.: The Simulated Pollen Dataset



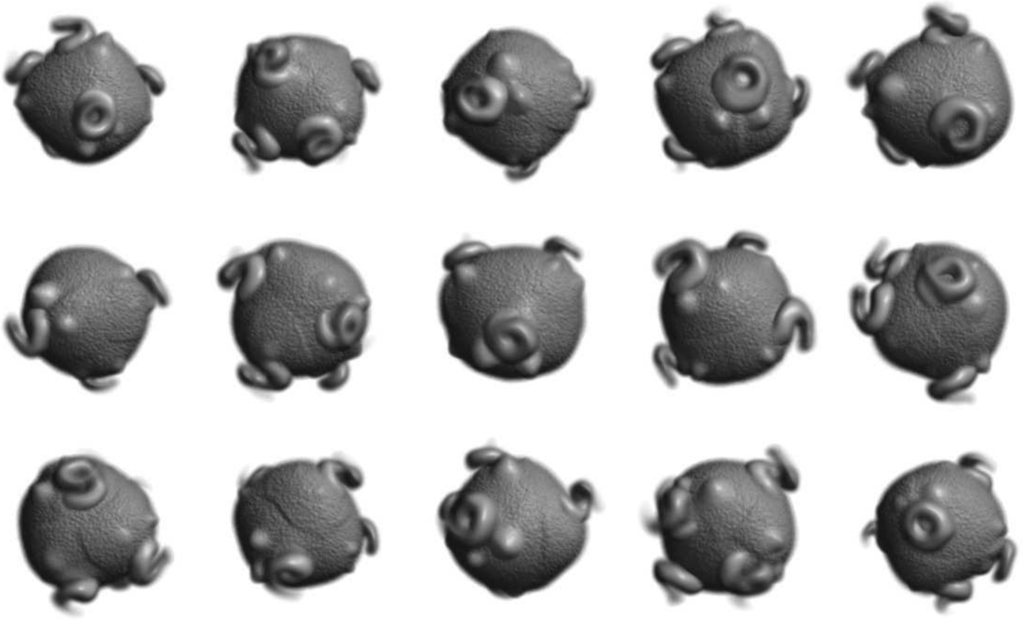
(a) pollen 6 (generated images)



(b) pollen 6 (grayscaled)

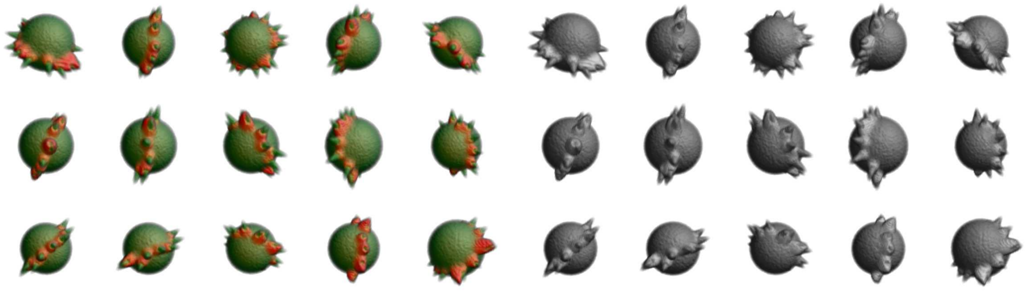
Figure E.31.: The Simulated Pollen Dataset

(a) pollen 7 (generated images)



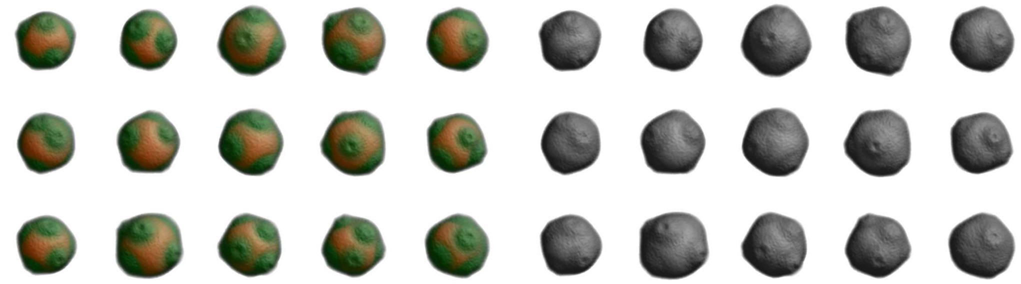
(b) pollen 7 (grayscaled)

Figure E.32.: The Simulated Pollen Dataset



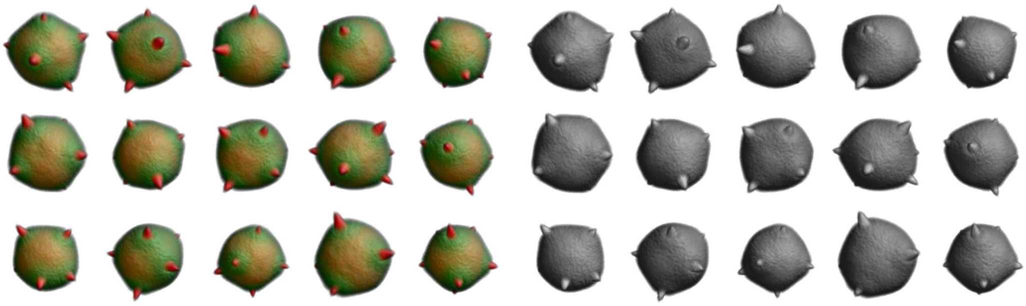
(a) pollen 8 (generated images)

(b) pollen 8 (grayscaled)



(c) pollen 9 (generated images)

(d) pollen 9 (grayscaled)

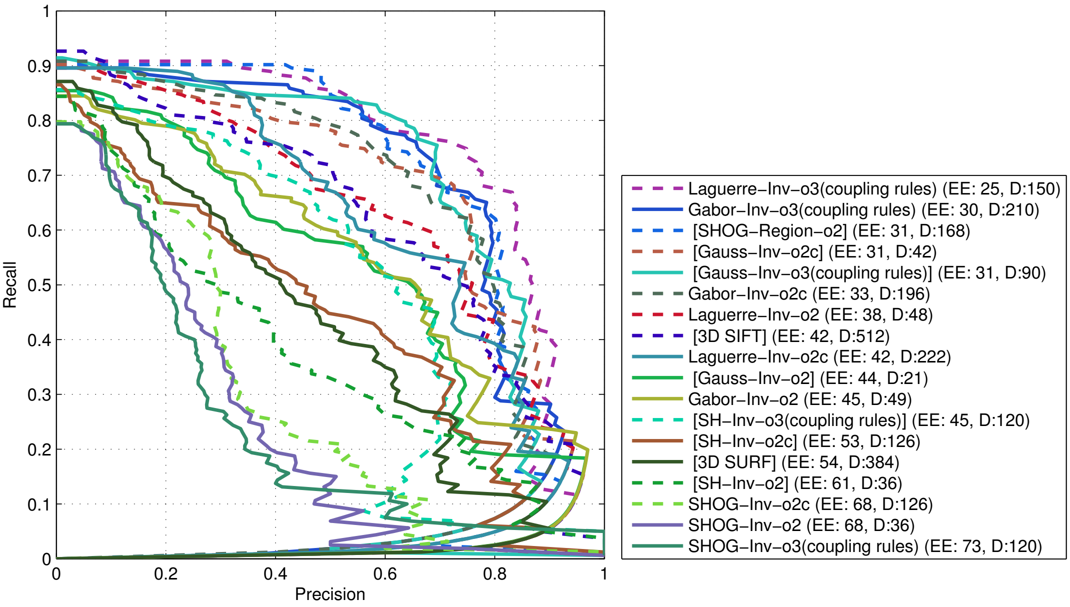


(e) pollen 10 (generated images)

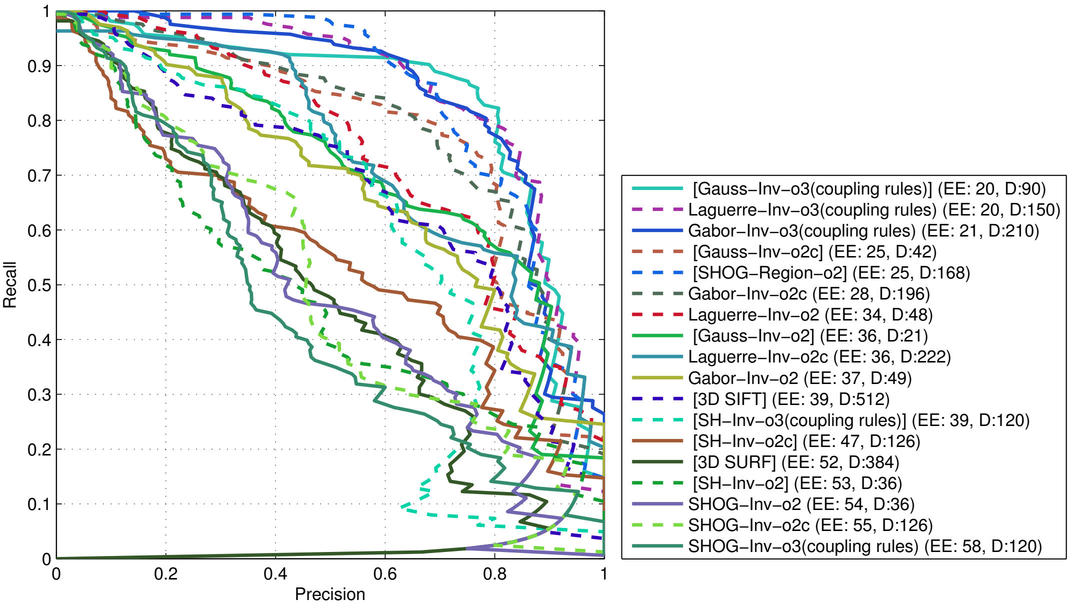
(f) pollen 10 (grayscaled)

Figure E.33.: (Additional Negative Datasets)

E.3. Plots and Figures: Real Pollen Dataset

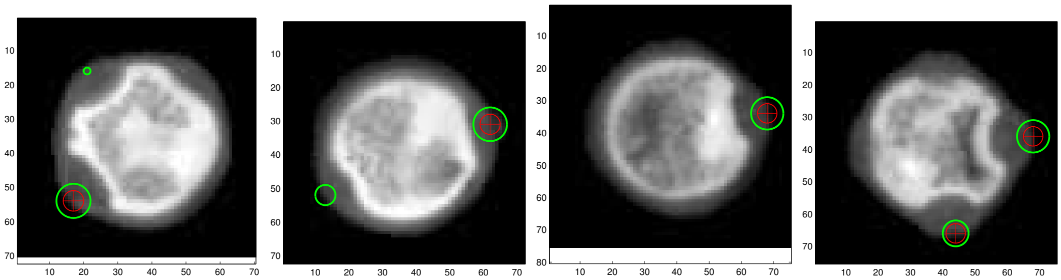


(a) tolerated displacement of 3 voxels

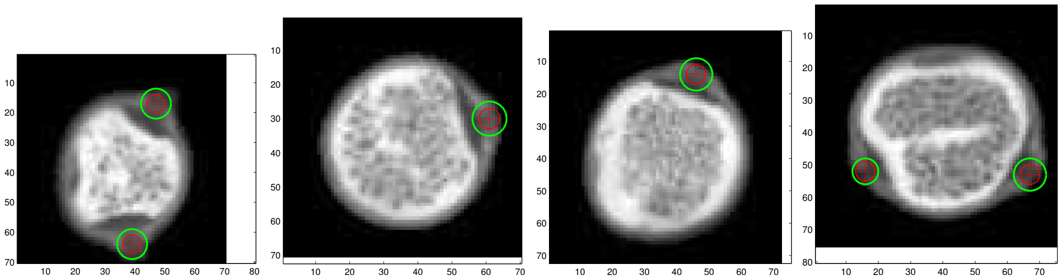


(b) tolerated displacement of 6 voxels

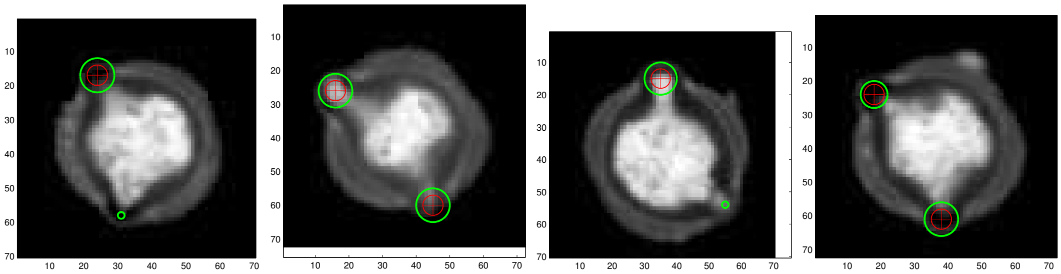
Figure E.34.: PR curves for the real pollen dataset



(a) landmark 1 (b) landmark 1 (c) landmark 1 (d) landmark 1

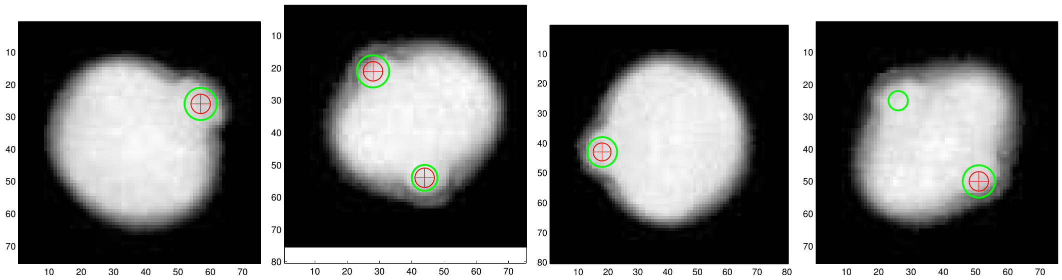


(e) landmark 2 (f) landmark 2 (g) landmark 2 (h) landmark 2

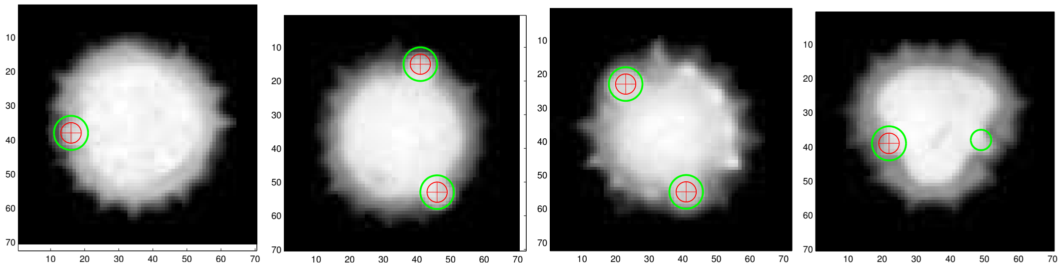


(i) landmark 3 (j) landmark 3 (k) landmark 3 (l) landmark 3

Figure E.35.: Slices from the images in the training set of the real pollen dataset. The size of the green circle indicates the distance to a landmark (if close to the current slice). A red circle indicates that the landmark is in the current slice.



(a) landmark 4 (b) landmark 4 (c) landmark 4 (d) landmark 4



(e) landmark 5 (f) landmark 5 (g) landmark 5 (h) landmark 5

Figure E.36.: Slices from the images in the training set of the real pollen dataset. The size of the green circle indicates the distance to a landmark (if close to the current slice). A red circle indicates that the landmark is in the current slice.

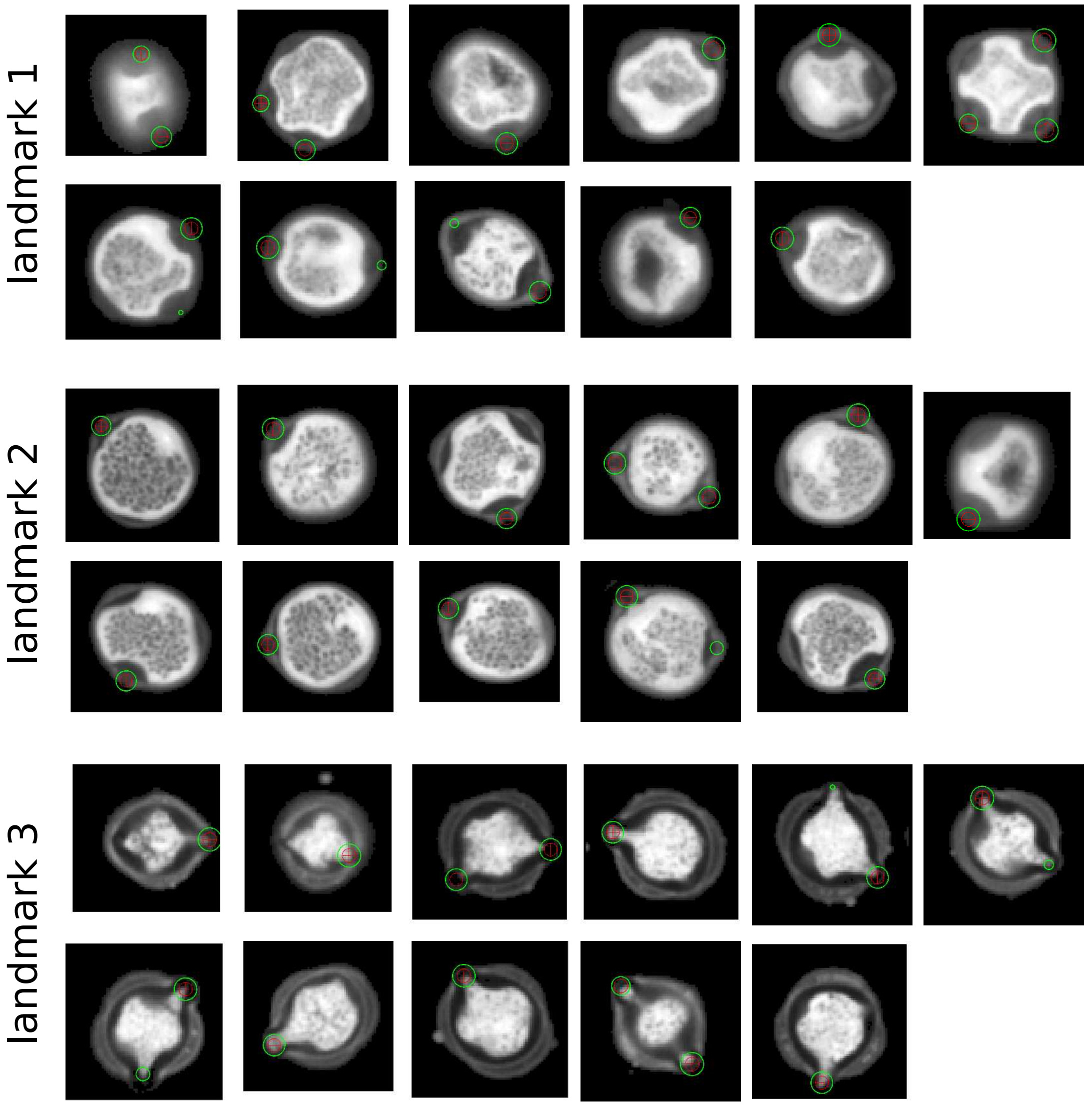


Figure E.37.: Slices from the images in the test set of the real pollen dataset. The size of the green circle indicates the distance to a landmark (if close to the current slice). A red circle indicates that the landmark is in the current slice.

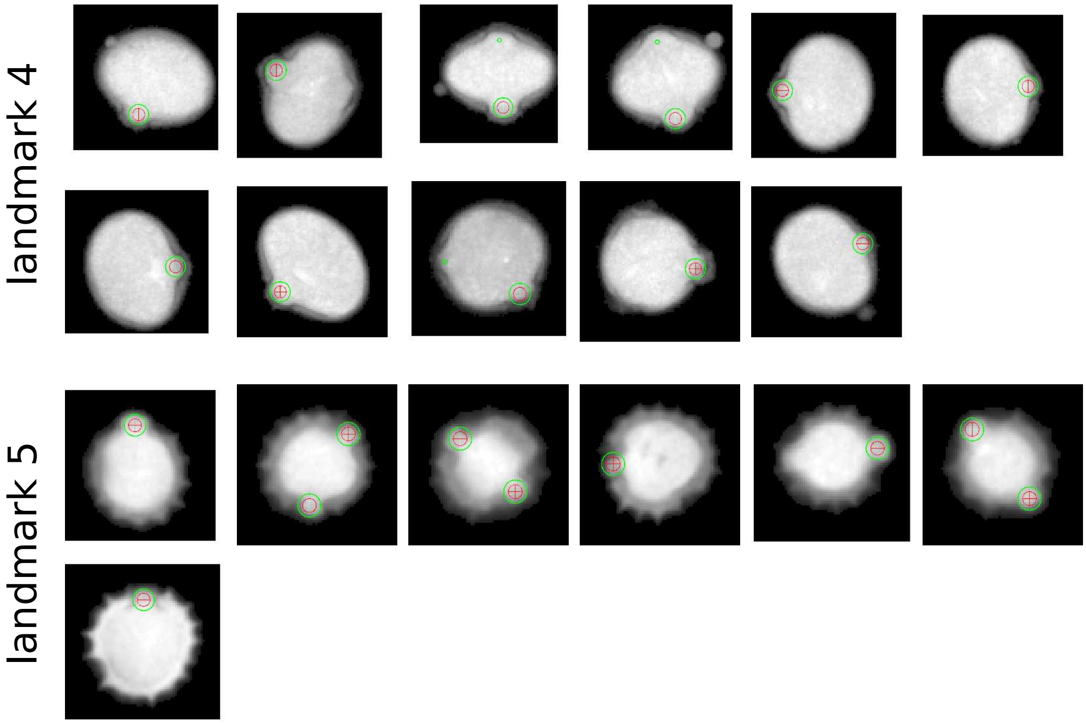


Figure E.38.: Slices from the images in the test set of the real pollen dataset. The size of the green circle indicates the distance to a landmark (if close to the current slice). A red circle indicates that the landmark is in the current slice.

BIBLIOGRAPHY

Abramowitz, Milton and Irene A. Stegun (1964). *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. ninth Dover printing, tenth GPO printing. New York: Dover (cit. on pp. 112, 214, 216, 218).

Allaire, S., J.J. Kim, S.L. Breen, D.A. Jaffray, and V. Pekar (2008). „Full orientation invariance and improved feature selectivity of 3D SIFT with application to medical image analysis.“ In: *Proc. of the MMBIA*, pp. 1–8 (cit. on pp. 29, 140, 141, 157).

Arfken, George B. and Hans J. Weber (July 2005). *Mathematical Methods for Physicists, Sixth Edition: A Comprehensive Guide*. 6th ed. Academic Press (cit. on p. 215).

Bay, H., T. Tuytelaars, and L. Van Gool (2006). „SURF: Speeded up robust features.“ In: *Proc. of the ECCV*. Graz, Austria: Springer, pp. 404–417 (cit. on pp. 29, 140).

Bicego, M., A. Lagorio, E. Grosso, and M. Tistarelli (2006). „On the use of SIFT features for face authentication.“ In: *Proc. of the CVPR Workshops*. IEEE, pp. 35–35 (cit. on pp. 29, 140).

Bigun, J. (1994). „Speed, frequency, and orientation tuned 3-D Gabor filter banks and their design.“ In: *Proc. of ICPR*. IEEE. Jerusalem , Israel, C:184–187 (cit. on p. 130).

Bigun, J. and G.H. Granlund (1987). „Optimal orientation detection of linear symmetry.“ In: *Proc. of the ICCV*. IEEE. London, UK, pp. 433–438 (cit. on pp. 51, 138).

Bigun, J., G.H. Granlund, and J. Wiklund (1991). „Multidimensional orientation estimation with applications to texture analysis and optical flow.“ In: *IEEE Trans. Pattern Anal. Mach. Intell.* 13.8, pp. 775–790 (cit. on p. 139).

Bourdev, L. and J. Malik (2009). „Poselets: Body part detectors trained using 3D human pose annotations.“ In: *Proc. of the ICCV*. IEEE. Kyoto, Japan, pp. 1365–1372 (cit. on pp. 20, 139).

Bovik, A. C., M. Clark, and W. S. Geisler (1990). „Multichannel Texture Analysis Using Localized Spatial Filters.“ In: *IEEE Trans. Pattern Anal. Mach. Intell.* 12.1, pp. 55–73 (cit. on p. 128).

Breiman, Leo (Oct. 2001). „Random Forests.“ In: *Mach. Learn.* 45 (1), pp. 5–32 (cit. on pp. 16, 161, 181).

Brink, D. M. and G.R. Satchler (1993). *Angular Momentum*. Oxford Science Publications (cit. on pp. 78, 80).

Bronshtein, I. N. and K. A. Semendyayev (1997). *Handbook of mathematics (3rd ed.)* Ed. by K. A. Kirsch. London, UK: Springer-Verlag (cit. on p. 216).

Brown, J. and A. Carrington (2003). *Rotational Spectroscopy of Diatomic Molecules*. Cambridge University Press (cit. on p. 61).

Brox, T. and J. Malik (2011). „Large displacement optical flow: descriptor matching in variational motion estimation." In: *IEEE Trans. Pattern Anal. Mach. Intell.* 33.3, pp. 500–513 (cit. on p. 139).

Brox, T., J. Weickert, B. Burgeth, and P. Mrázek (2006). „Nonlinear structure tensors." In: *Image and Vision Computing* 24.1, pp. 41–55 (cit. on p. 51).

Buhmann, J., J. Lange, and C. von der Malsburg (1989). „Distortion invariant object recognition by matching hierarchically labeled graphs." In: *Proc. of the IJCNN, the International Joint Conference on Neural Networks*. IEEE, pp. 155–159 (cit. on p. 128).

Burkhardt, H. and S. Siggelkow (2001). „Invariant features in pattern recognition–fundamentals and applications." In: *Nonlinear model-based image/video processing and analysis*, pp. 269–307 (cit. on pp. 28, 83, 84).

Byerly, W.E. (2003). *An Elementary Treatise on Fourier's Series: And Spherical, Cylindrical, and Ellipsoidal Harmonics, with Applications to Problems in Mathematical Physics*. Dover Phoenix Editions. Dover Publications (cit. on p. 77).

Canny, J. (1986). „A computational approach to edge detection." In: *IEEE Trans. Pattern Anal. Mach. Intell.* 6, pp. 679–698 (cit. on p. 97).

Canterakis, N. (1999). „3D Zernike Moments and Zernike Affine Invariants for 3D Image Analysis and Recognition." In: *Proc. of the 11th Scandinavian Conf. on Image Analysis*. Kangerlussuaq, Greenland (cit. on p. 32).

Cheung, W. and G. Hamarneh (2007a). „N-SIFT: n-dimensional scale invariant feature transform for matching medical images." In: *Proc. of the ISBI*. IEEE. Washington, DC, USA, pp. 720–723 (cit. on pp. 29, 140).

– (2007b). „Scale invariant feature transform for n-dimensional images (n-SIFT)." In: *The Insight Journal* (cit. on p. 29).

Choorat, P., W. Chiracharit, and K. Chamnongthai (2012). „A single tooth segmentation using structural orientations and statistical textures." In: *Proc. of the Biomedical Engineering International Conference (BMEiCON), 2011*. IEEE, pp. 294–297 (cit. on p. 128).

Chow Chiu L.-Y., Moharerrzadeh M. (Feb. 2001). „The addition theorem of spherical Laguerre Gaussian functions and its application in molecular integrals." In: *Journal of Molecular Structure: THEOCHEM* 536, 263–267(5) (cit. on p. 115).

Cooley, J.W. and J.W. Tukey (1965). „An algorithm for the machine calculation of complex Fourier series." In: *Mathematics of computation*, pp. 297–301 (cit. on pp. 30, 40).

Cortes, C. and V. Vapnik (1995). „Support-vector networks.“ In: *Mach. Learn.* 20.3, pp. 273–297 (cit. on p. 16).

Dalal, N. and B. Triggs (2005). „Histograms of Oriented Gradients for Human Detection.“ In: *Proc. of the CVPR*. IEEE, pp. 886–893 (cit. on pp. 23, 139, 141).

Daugman, J. (1980). „Two-dimensional spectral analysis of cortical receptive field profiles.“ In: *Vision Research* 20.10, pp. 847–856 (cit. on p. 128).

Daugman, J. G. (1985). „Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters.“ In: *Journal of the Optical Society of America A: Optics, Image Science, and Vision* 2.7, pp. 1160–1169 (cit. on p. 128).

Daugman, J.G (July 1988). „Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression.“ In: *IEEE Transactions on Acoustics, Speech and Signal Processing* 36, pp. 1169–1179 (cit. on p. 128).

Descoteaux, M., E. Angelino, S. Fitzgibbons, and R. Deriche (2007). „Regularized, fast, and robust analytical Q-ball imaging.“ In: *Magnetic Resonance in Medicine* 58.3, pp. 497–510 (cit. on p. 124).

Dollár, P., C. Wojek, B. Schiele, and P. Perona (2009). „Pedestrian detection: A benchmark.“ In: *Proc. of the CVPR*. IEEE, pp. 304–311 (cit. on p. 19).

Farid, Hany and Eero P. Simoncelli (2004). „Differentiation of Discrete Multidimensional Signals.“ In: *IEEE Trans. Image Process.* 13, pp. 496–508 (cit. on p. 103).

Fehr, J. (2009). „Local Invariant Features for 3D Image Analysis.“ PhD thesis. Albert-Ludwigs-Universität Freiburg (cit. on pp. 32, 83).

– (2010). „Local Rotation Invariant Patch Descriptors for 3D Vector Fields.“ In: *Proc. of the ICPR*. IEEE. Istanbul, Turkey, pp. 1381–1384 (cit. on pp. 32, 83).

Fehr, J. and H. Burkhardt (2006). „Phase based 3D Texture Features.“ In: *Proc. of the DAGM*. Berlin, Germany: LNCS, Springer, pp. 263–272 (cit. on pp. 32, 83).

– (2007). „Harmonic Shape Histograms for 3D Shape Classification and Retrieval.“ In: *Proc. of the MVA*. IAPR. Tokyo, Japan (cit. on p. 32).

Fehr, J., M. Reisert, and H. Burkhardt (2009). „Cross-Correlation and Rotation Estimation of Local 3D Vector Field Patches.“ In: *Proc. of the 5th International Symposium on Advances in Visual Computing: Part I*. ISVC ’09. Las Vegas, Nevada: Springer-Verlag, pp. 287–296 (cit. on p. 75).

Felzenszwalb, P.F., R.B. Girshick, D. McAllester, and D. Ramanan (2010). „Object detection with discriminatively trained part-based models.“ In: *IEEE Trans. Pattern Anal. Mach. Intell.* 32.9, pp. 1627–1645 (cit. on pp. 20, 139).

Flitton, G., T.P. Breckon, N. Megherbi, and UK Cranfield (2010). „Object recognition using 3D SIFT in complex CT volumes.“ In: *Proc. of the BMVC*. BMVA. Aberystwyth, UK, pp. 11–1 (cit. on pp. 29, 163).

Florack, LMJ, BM ter Haar Romeny, JJ Koenderink, and MA Viergever (1994). „General intensity transformations and differential invariants.“ In: *Journal of Mathematical Imaging and Vision* 4.2, pp. 171–187 (cit. on p. 110).

Flusser, J. and T. Suk (2006). „Rotation moment invariants for recognition of symmetric objects.“ In: *IEEE Trans. Image Process.* 15.12, pp. 3784–3790 (cit. on p. 50).

Flusser, J., B. Zitova, and T. Suk (2009). *Moments and moment invariants in pattern recognition*. Wiley (cit. on p. 31).

Förstner, W. (1986). „A feature based correspondence algorithm for image matching.“ In: *International Archives of Photogrammetry and Remote Sensing* 26.3, pp. 150–166 (cit. on pp. 23, 27, 51, 138, 139).

Frigo, Matteo and Steven G. Johnson (2005). „The Design and Implementation of FFTW3.“ In: *Proc. of the IEEE* 93.2. Special issue on “Program Generation, Optimization, and Platform Adaptation”, pp. 216–231 (cit. on pp. 40, 60, 150, 151, 155).

Gaal, Steven A. (1973). *Linear Analysis and Representation Theory*. Springer (cit. on p. 45).

Gabor, D. (1846). „Theory of communication.“ In: *J. Inst. of Elect. Eng.* 93, pp. 429–441 (cit. on p. 128).

Galassi, Mark, Jim Davies, James Theiler, Brian Gough, Gerard Jungman, Michael Booth, and Fabrice Rossi (Feb. 2003). *Gnu Scientific Library: Reference Manual*. Network Theory Ltd. (cit. on p. 105).

Geronimo, D., A.M. Lopez, A.D. Sappa, and T. Graf (2010). „Survey of pedestrian detection for advanced driver assistance systems.“ In: *IEEE Trans. Pattern Anal. Mach. Intell.* 32.7, pp. 1239–1258 (cit. on p. 19).

Granlund, Gösta H. and Hans Knutsson (1995). *Signal processing for computer vision*. Kluwer (cit. on p. 139).

Haar Romeny, L.M.J.F.B.M. ter, JJ Koenderink, and MA Viergever (1992). „General Intensity Transformations and Second Order Invariants.“ In: *Theory & Applications of Image Analysis: Selected Papers from the 7th Scandinavian Conference on Image Analysis*. Vol. 2. World Scientific Publishing Company Incorporated, p. 22 (cit. on p. 110).

Haasdonk, B., A. Halawani, and H. Burkhardt (2004). „Adjustable invariant features by partial Haar-integration.“ In: *Proc. of the ICPR*. Vol. 2. IEEE. Cambridge, UK, pp. 769–774 (cit. on p. 30).

Haasdonk, B., A. Vossen, and H. Burkhardt (2005). „Invariance in kernel methods by Haar-integration kernels.“ In: *Image Analysis*, pp. 213–236 (cit. on p. 30).

Harris, C. and M. Stephens (1988). „A combined corner and edge detector.“ In: *Alvey vision conference*. Vol. 15. Manchester, UK, p. 50 (cit. on pp. 23, 27, 51, 97, 139).

Hasselmann K. Munk, W. H. and G. J. F. MacDonald (1963). „Bi-spectra of ocean waves,“ in: *Time Series Analysis*. Ed. by M. Rosenblatt, p. 125 (cit. on p. 68).

Heitz, G. and D. Koller (2008). „Learning spatial context: Using stuff to find things.“ In: *Proc. of the ECCV*, pp. 30–43 (cit. on p. 20).

Hoang, T.V. (2011). „Représentations d'images pour la reconnaissance de formes.“ PhD thesis. Université Nancy II (cit. on p. 31).

Hoang, T.V. and S. Tabbone (2011). „Generic polar harmonic transforms for invariant image description.“ In: *Proc. of the ICIP*. IEEE. Brussels, Belguim, pp. 829–832 (cit. on p. 31).

Horn, B.K.P. and B.G. Schunck (1981). „Determining optical flow.“ In: *Artificial intelligence* 17.1, pp. 185–203 (cit. on p. 18).

Hsu, Y.N., H.H. Arsenault, and G. April (1982). „Rotation-invariant digital pattern recognition using circular harmonic expansion.“ In: *Applied Optics* 21.22, pp. 4012–4015 (cit. on p. 31).

Hu, M.K. (1962). „Visual pattern recognition by moment invariants.“ In: *Information Theory, IRE Transactions on* 8.2, pp. 179–187 (cit. on p. 50).

Hu, P., Y. Zhao, Z. Yang, and J. Wang (2002). „Recognition of gray character using Gabor filters.“ In: *Information Fusion, 2002. Proceedings of the Fifth International Conference on*. Vol. 1. IEEE, pp. 419–424 (cit. on p. 128).

Hurwitz, A. (1897). „Über die Erzeugung von Invarianten durch Integration.“ In: *Nachrichten von der Kgl. Gesellschaft der Wissenschaften zu Göttingen - Mathematisch-Physikalische Klasse*, pp. 71–90 (cit. on p. 30).

Jain, A. K. and F. Farrokhnia (1990). „Unsupervised texture segmentation using Gabor filters.“ In: *Proc. of the IEEE International Conference on Systems, Man and Cybernetics*, pp. 14–19 (cit. on p. 128).

Jain, A.K. and R.C. Dubes (1988). *Algorithms for clustering data*. Prentice-Hall, Inc. (cit. on p. 16).

John, G.H. and P. Langley (1995). „Estimating continuous distributions in Bayesian classifiers.“ In: *Proc. of the eleventh conference on uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., pp. 338–345 (cit. on p. 16).

Kakarala, R. (1992). „Triple correlation on groups.“ PhD thesis. University of California, Irvine (cit. on pp. 30, 33, 47, 68, 92).

Kakarala, R. (1993). „A group-theoretic approach to the triple correlation.“ In: *Higher-Order Statistics, IEEE Signal Processing Workshop on*. South Lake Tahoe, CA, USA, pp. 28 –32 (cit. on p. 68).

– (2012). „The Bispectrum as a Source of Phase-Sensitive Invariants for Fourier Descriptors: A Group-Theoretic Approach.“ In: *Journal of Mathematical Imaging and Vision*, pp. 1–13 (cit. on pp. 68, 91).

Kamarainen, J.K., V. Kyrki, and H. Kalviainen (2002). „Noise tolerant object recognition using Gabor filtering.“ In: *Proc. of the DSP 2002, the 14th International Conference on Digital Signal Processing*. Vol. 2. IEEE, pp. 1349–1352 (cit. on p. 128).

– (2006). „Invariance properties of Gabor filter-based features-overview and applications.“ In: *IEEE Trans. Image Process.* 15.5, pp. 1088–1099 (cit. on p. 128).

Kazhdan, M., T. Funkhouser, and R. Szymon (2003). „Rotation invariant spherical harmonic representation of 3D shape descriptors.“ In: *SGP ’03: Proc. of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing*. Aachen, Germany: Eurographics Association, pp. 156–164 (cit. on pp. 32, 58, 83, 157, 164).

Ke, Y. and R. Sukthankar (2004). „PCA-SIFT: A more distinctive representation for local image descriptors.“ In: *Proc. of the CVPR*. Vol. 2. IEEE, pp. II–506 (cit. on p. 29).

Khotanzad, A. and Y.H. Hong (1990). „Invariant image recognition by Zernike moments.“ In: *IEEE Trans. Pattern Anal. Mach. Intell.* 12.5, pp. 489–497 (cit. on pp. 20, 31).

Kim, YC and EJ Powers (1978). „Digital bispectral analysis of self-excited fluctuation spectra.“ In: *Physics of Fluids* 21, p. 1452 (cit. on p. 68).

Knapek, Markus, Ricardo Swain Oropeza, Ricardo Swain, Oropeza David, and David J. Kriegman (2000). „Selecting Promising Landmarks.“ In: *Proc. of the IEEE International Conference on Robotics and Automation*, pp. 3771–3777 (cit. on p. 110).

Knopp, J., M. Prasad, G. Willems, R. Timofte, and L. Van Gool (2010). „Hough Transform and 3D SURF for robust three dimensional classification.“ In: *Proc. of the ECCV*. Crete, Greece: Springer (cit. on pp. 29, 140, 141, 157, 163).

Koenderink, J. J. and A. J. van Doorn (1987). „Representation of local geometry in the visual system.“ In: *Biological Cybernetics* 55, pp. 367–375 (cit. on pp. 23, 33, 109, 110, 177).

– (June 1992). „Generic neighborhood operators.“ In: *IEEE Trans. Pattern Anal. Mach. Intell.* 14.6, pp. 597–605 (cit. on pp. 32, 110, 111, 115, 116, 119).

Kondor, R. (2008). „Group theoretical methods in machine learning.“ PhD thesis. Columbia University (cit. on p. 68).

Köthe, U. (2003). „Edge and junction detection with an improved structure tensor.“ In: *Pattern Recognition*, pp. 25–32 (cit. on p. 139).

Krainov, V. P., H. R. Reiss, and B. M. Smirnov (2005). *Appendix D: Wigner 6j Symbols, in Radiative Processes in Atomic Physics*. Ed. by FRG. Wiley-VCH Verlag GmbH & Co. KGaA Weinheim. Radiative Processes in Atomic Physics (cit. on p. 221).

Kyrki, V., J.K. Kamarainen, and H. Kälviäinen (2004). „Simple Gabor feature space for invariant object recognition.“ In: *Pattern Recognition Letters* 25.3, pp. 311–318 (cit. on p. 128).

Lampert, C.H., M.B. Blaschko, and T. Hofmann (2008). „Beyond sliding windows: Object localization by efficient subwindow search.“ In: *Proc. of the CVPR*. IEEE, pp. 1–8 (cit. on p. 29).

Lancaster J.L., Summerln J.L. Rainey L. Freitas C.S. Fox P.T. (1997). „The Talairach Daemon, a database server for Talairach Atlas Labels.“ In: *Neuroimage* (cit. on p. 190).

Leibe, Bastian, Ales Leonardis, and Bernt Schiele (2004). „Combined Object Categorization and Segmentation With An Implicit Shape Model.“ In: *Proc. of the ECCV workshop on statistical learning in computer vision*. Prague, Czech Republic: Springer, pp. 17–32 (cit. on p. 19).

Li, Y. (1992). „Reforming the theory of invariant moments for pattern recognition.“ In: *Pattern Recognition* 25.7, pp. 723–730 (cit. on p. 50).

Li, Y., Z. Lu, C. Zhu, and X. Niu (2012). „Robust Image Hashing Based on Random Gabor Filtering and Dithered Lattice Vector Quantization.“ In: *IEEE Trans. Image Process.* 21.4, pp. 1963–1980 (cit. on p. 128).

Lillholm, Martin and Kim Steenstrup Pedersen (2004). „Jet Based Feature Classification.“ In: *Proc. of the ICPR*. IEEE. Washington, DC, USA, pp. 787–790 (cit. on p. 110).

Lin, Hsuan-Tien, Chih-Jen Lin, and Ruby C. Weng (Oct. 2007). „A note on Platt's probabilistic outputs for support vector machines.“ In: *Mach. Learn.* 68 (3), pp. 267–276 (cit. on p. 26).

Lin, W.Y., S. Liu, Y. Matsushita, T.T. Ng, and L.F. Cheong (2011). „Smoothly varying affine stitching.“ In: *Proc. of the CVPR*. IEEE, pp. 345–352 (cit. on p. 29).

Lin, W.Y., L. Liu, Y. Matsushita, K.L. Low, and S. Liu (2012). „Aligning images in the wild.“ In: *Proc. of the CVPR*. IEEE, pp. 1–8 (cit. on p. 140).

Lindeberg, T. (1990). „Scale-Space for Discrete Signals.“ In: *IEEE Trans. Pattern Anal. Mach. Intell.* 12, pp. 234–254 (cit. on pp. 110, 115, 140).

– (1993). *Scale-Space Theory in Computer Vision*. Kluwer international series in engineering and computer science. Springer (cit. on pp. 19, 110, 139).

Liu, C., J. Yuen, and A. Torralba (2011a). „SIFT flow: Dense correspondence across scenes and its applications.“ In: *IEEE Trans. Pattern Anal. Mach. Intell.* 33.5, pp. 978–994 (cit. on pp. 29, 140).

Liu, K., H. Skibbe, T. Schmidt, T. Blein, K. Palme, and O. Ronneberger (Aug. 2011b). „3D Rotation-Invariant Description from Tensor Operation on Spherical HOG Field.“ In: *Proc. of the BMVC*. Ed. by Manuel Trucco Jesse Hoey Stephen McKenna and Jianguo Zhang. BMVA. Dundee, UK (cit. on pp. 32, 83, 115, 138, 166, 179).

Liu, K., Q. Wang, W. Driever, and O. Ronneberger (2012). „2D/3D Rotation-Invariant Detection using Equivariant Filters and Kernel Weighted Mapping.“ In: *Proc. of the CVPR*. IEEE (cit. on pp. 29, 31, 32, 110, 115, 157, 226, 228).

Lowe, David G (2004). „Distinctive image features from scale-invariant keypoints.“ In: *Int. J. Comput. Vision* 60.2, pp. 91–110 (cit. on pp. 19, 20, 23, 27, 28, 85, 97, 139, 140).

Lucas, B.D., T. Kanade, et al. (1981). „An iterative image registration technique with an application to stereo vision.“ In: *Proceedings of the 7th international joint conference on Artificial intelligence* (cit. on p. 18).

Lue-Yung Chow Chiu, Mohammad Moharerrzadeh (1999). „Fourier transform of spherical Laguerre Gaussian functions and its application in molecular integrals.“ In: vol. 73, pp. 265–273 (cit. on p. 116).

Ma, L., M. Reisert, and H. Burkhardt (2012). „RENNSH: A Novel Alpha-helix Identification Approach for Intermediate Resolution Electron Density Maps.“ In: *Computational Biology and Bioinformatics, IEEE/ACM Transactions on* 9.1, pp. 228–239 (cit. on p. 32).

Mak, L., S. Grandison, and R.J. Morris (2008). „An extension of spherical harmonics to region-based rotationally invariant descriptors for molecular shape description and comparison.“ In: *Journal of Molecular Graphics and Modelling* 26.7, pp. 1035–1045 (cit. on p. 32).

Mann, Steve and Simon Haykin (1991). „The Chirplet Transform: A Generalization of Gabor’s Logon Transform.“ In: *Vision Interface ’91 proceedings/compte rendu, Calgary* (cit. on p. 31).

Matsuoka, Osamu (1998). „Molecular integrals over real solid spherical Gaussian-type functions.“ In: *J. Chem. Phys* (cit. on p. 115).

Mikolajczyk, K., B. Leibe, and B. Schiele (2006). „Multiple Object Class Detection with a Generative Model.“ In: *Proc. of the CVPR*. Vol. 1. IEEE, pp. 26–36 (cit. on p. 20).

Miller, Willard (1991). „Topics In Harmonic Analysis With Applications To Radar And Sonar.“ In: *RADAR and SONAR, Part 1, IMA Volumes in Mathematics and its Applications*. Springer-Verlag, pp. 66–168 (cit. on pp. 41, 42, 44, 213).

Morel, J.M. and G. Yu (2009). „ASIFT: A new framework for fully affine invariant image comparison.“ In: *SIAM Journal on Imaging Sciences* 2.2, pp. 438–469 (cit. on pp. 20, 29).

Morse, P. M. and H. Feshbach (1953). *Methods of Theoretical Physics, Part II.* New York: McGraw-Hill, pp. 1898–1901 (cit. on p. 75).

Nachbin, L. (1965). *The Haar Integral.* (Translated by Bechtolsheim, L.) Princeton, NJ: van Nostrand (cit. on p. 42).

Naĭmark, M.A. (1959). *Normed rings.* Groningen, The Netherlands: English Transl. P. Noordhoff (cit. on p. 45).

Niemeijer, M., M.K. Garvin, K. Lee, B. van Ginneken, M.D. Abràmoff, and M. Sonka (2009). „Registration of 3D spectral OCT volumes using 3D SIFT feature point matching.“ In: *Proc. of the SPIE*. Vol. 7259, p. 72591I (cit. on p. 29).

Nister, D. and H. Stewenius (2006). „Scalable recognition with a vocabulary tree.“ In: *Proc. of the CVPR*. Vol. 2. IEEE, pp. 2161–2168 (cit. on pp. 29, 140).

Noble, J.A. (1988). „Finding corners.“ In: *Image and Vision Computing* 6.2, pp. 121–128 (cit. on p. 139).

Noronha, S. and R. Nevatia (1997). „Detection and description of buildings from multiple aerial images.“ In: *Proc. of the CVPR*. IEEE, pp. 588–594 (cit. on p. 20).

Novotni, M. and R. Klein (2004). „Shape retrieval using 3D Zernike descriptors.“ In: *Computer-Aided Design* 36.11, pp. 1047–1062 (cit. on p. 32).

Nyquist, H. (1928). „Certain topics in telegraph transmission theory.“ In: *American Institute of Electrical Engineers, Transactions of the* 47.2, pp. 617–644 (cit. on p. 150).

Olowoyeye, A., M. Tuceryan, and S. Fang (2009). „Medical volume segmentation using bank of Gabor filters.“ In: *Proc. of the ACM symposium on Applied Computing*. ACM, pp. 826–829 (cit. on p. 128).

Osada, K., T. Furuya, and R. Ohbuchi (2008). „SHREC’08 entry: Local volumetric features for 3D model retrieval.“ In: *Proc. of the Shape Modeling and Applications, SMI*. New York, USA, pp. 245–246 (cit. on p. 29).

Paganelli, C., M. Peroni, F. Pennati, G. Baroni, P. Summers, M. Bellomi, and M. Riboldi (2012). „Scale Invariant Feature Transform as feature tracking method in 4D imaging: A feasibility study.“ In: *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*. IEEE, pp. 6543–6546 (cit. on p. 29).

Peter, F and H Weyl (1927). „Die Vollständigkeit der primitiven Darstellungen einer geschlossenen kontinuierlichen Gruppe.“ In: *Mathematische Annalen* 97.1, pp. 737–755 (cit. on p. 45).

Platt, John C. (1999). „Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods.“ In: *Advances in large margin classifiers* 10.3, pp. 61–74 (cit. on p. 26).

Qian, Z., D. Metaxas, and L. Axel (2006). „Extraction and Tracking of MRI Tagging Sheets Using a 3D Gabor Filter Bank.“ In: *Proc. of the IEEE Eng Med Biol Soc* 1 (cit. on p. 128).

Racah, Giulio (Nov. 1942). „Theory of Complex Spectra. II.“ In: *Phys. Rev.* 62 (9-10), pp. 438–462 (cit. on p. 70).

Rao, A.R. and B.G. Schunck (1991). „Computing oriented texture fields.“ In: *CVGIP: Graphical Models and Image Processing* 53.2, pp. 157–185 (cit. on p. 139).

Reddi, S. S. (1981). „Radial and angular moment invariants for image identification.“ In: *IEEE Trans. Pattern Anal. Mach. Intell.* 2, pp. 240–242 (cit. on p. 50).

Reisert, M. (2007). *Spherical Derivatives for Steerable Filtering in 3D*. Tech. rep. 3. Albert-Ludwigs-Universität Freiburg (cit. on p. 28).

– (2008). „Group Integration Techniques in Pattern Analysis - A Kernel View.“ PhD thesis. Albert-Ludwigs-Universität Freiburg (cit. on pp. 18, 29, 83).

Reisert, M. and H. Burkhardt (2006). „Invariant features for 3D-data based on group integration using directional information and spherical harmonic expansion.“ In: *Proc. of the ICPR*. Vol. 4. IEEE. Hong Kong, pp. 206–209 (cit. on pp. 32, 83).

– (2008a). „Complex Derivative Filters.“ In: *IEEE Trans. Image Process.* 17.12, pp. 2265–2274 (cit. on pp. 110, 115, 119).

– (2008b). „Efficient Tensor Voting with 3D tensorial harmonics.“ In: *Proc. of the CVPR Workshops*. IEEE, pp. 1–7 (cit. on pp. 28, 72, 74).

– (2008c). „Equivariant Holomorphic Filters for Contour Denoising and Rapid Object Detection.“ In: *IEEE Trans. Image Process.* 17.2, pp. 190–203 (cit. on pp. 31, 225, 226).

– (2009a). „Harmonic Filters for Generic Feature Detection in 3D.“ In: *Proc. of the DAGM*. Jena, Germany: LNCS, Springer, pp. 131–140 (cit. on pp. 27, 32, 34, 100, 104, 110, 115, 120, 157, 160, 166, 167, 225, 227).

– (2009b). „Spherical Tensor Calculus for Local Adaptive Filtering.“ In: *Tensors in Image Processing and Computer Vision*. Ed. by S. Aja-Fernández, R. de Luis García, D. Tao, and X. Li. Springer (cit. on pp. 52, 55, 61, 74, 78, 80, 81, 97).

Reisert, M. and H Skibbe (2012). „Left-Invariant Diffusion on the Motion Group in terms of the Irreducible Representations of SO(3).“ In: *arXiv:submit/0423757 [math.AP]* (cit. on pp. 51, 61, 143).

Reisert, M., I. Mader, C. Anastosoulus, M. Weigel, S. Schnell, and V.Kiselev (2010). „Global Fiber Reconstruction Becomes Practical." In: *Neuroimage* (cit. on pp. 124, 186).

Reiss, T.H. (1991). „The revised fundamental theorem of moment invariants." In: *IEEE Trans. Pattern Anal. Mach. Intell.* 13.8, pp. 830–834 (cit. on p. 50).

Ronneberger, O. (2007). „3D invariants for automated pollen recognition." PhD thesis. Albert-Ludwigs-Universität Freiburg (cit. on pp. 32, 83, 160).

Ronneberger, O., H. Burkhardt, and E. Schultz (2002). „General-purpose Object Recognition in 3D Volume Data Sets using Gray-Scale Invariants – Classification of Airborne Pollen-Grains Recorded with a Confocal Laser Scanning Microscope." In: *Proc. of the ICPR*. IEEE. Quebec, Canada (cit. on pp. 30, 83).

Ronneberger, O., J. Fehr, and H. Burkhardt (2005). „Voxel-Wise Gray Scale Invariants for Simultaneous Segmentation and Classification." In: *Proc. of the DAGM*. Vienna, Austria: LNCS, Springer, pp. 85–92 (cit. on pp. 27, 32, 60, 83).

Ronneberger, O., Q. Wang, and H. Burkhardt (2007). „3D invariants with high robustness to local deformations for automated pollen recognition." In: *Proc. of the DAGM*, Heidelberg, Germany: LNCS, Springer, pp. 455–435 (cit. on pp. 32, 159, 160).

Ronneberger, O., K. Liu, M. Rath, D. Rueß, T. Mueller, H. Skibbe, B. Drayer, T. Schmidt, A. Filippi, R. Nitschke, T. Brox, H. Burkhardt, and W. Driever (2012). „ViBE-Z: a framework for 3D virtual colocalization analysis in zebrafish larval brains." In: *Nature Methods* 9.7, pp. 735–742 (cit. on pp. 15, 32, 90, 226).

Rose, M. (1995). *Elementary Theory of Angular Momentum*. Dover Publications (cit. on pp. 78, 80, 216).

Rosenblatt, F. (1958). „The perceptron: a probabilistic model for information storage and organization in the brain." In: *Psychological review* 65.6, p. 386 (cit. on p. 16).

Rudin, W. (2006). *Real and Complex Analysis*. Tata McGraw-Hill Education (cit. on p. 39).

Sandler, R. and M. Lindenbaum (2009). „Optimizing Gabor Filter Design for Texture Edge Detection and Classification." In: *Int. J. Comput. Vision* 84.3, pp. 308–324 (cit. on p. 128).

Schael, M. (2002). „Invariant texture classification using group averaging with relational kernel functions." In: *Texture 2002 the 2nd international workshop on texture analysis and synthesis*, pp. 129–134 (cit. on p. 83).

Schael, M. and S. Siggelkow (2000). „Invariant grey-scale features for 3D sensordata." In: *Proc. of the ICPR*. Vol. 2. IEEE. Barcelona, Spain, pp. 531–535 (cit. on pp. 30, 83).

Schlachter, M, M Reisert, C Herz, F Schluermann, S Lassmann, M Werner, H Burkhardt, and O Ronneberger (2010). „Harmonic Filters for 3D Multi-Channel Data: Rotation Invariant Detection of Mitoses in Colorectal Cancer." In: *IEEE Trans. Med. Imag.* 29.8, pp. 1485–1495 (cit. on pp. 32, 110, 115, 226, 228).

Schmid, Cordelia and Roger Mohr (1997). „Local grayvalue invariants for image retrieval." In: *IEEE Trans. Pattern Anal. Mach. Intell.* 19, pp. 530–535 (cit. on p. 110).

Schmidt, U. and S. Roth (2012). „Learning rotation-aware features: From invariant priors to equivariant descriptors." In: *Proc. of the CVPR*. IEEE, pp. 2050–2057 (cit. on pp. 20, 31).

Schnell, S., D. Saur, B.W. Kreher, J. Hennig, H. Burkhardt, and V.G. Kiselev (2009). „Fully automated classification of HARDI in vivo data using a support vector machine." In: *Neuroimage* 46, pp. 642–651 (cit. on pp. 32, 83, 185).

Schulz-Mirbach, H. (1994). „Constructing Invariant Features by Averaging Techniques." In: *Proc. of the ICPR*. Vol. II. Jerusalem , Israel, pp. 387–390 (cit. on p. 29).

– (1995a). „Anwendung von Invarianzprinzipien zur Merkmalgewinnung in der Mustererkennung." Reihe 10, Nr. 372, VDI-Verlag. PhD thesis. Technische Universität Hamburg-Harburg (cit. on pp. 30, 83, 85).

– (1995b). „Invariant features for gray scale images." In: *Proc. of the DAGM*. Ed. by G. Sagerer, S. Posch, and F. Kummert. Bielefeld, Germany: Reihe Informatik aktuell, Springer, pp. 1–14 (cit. on p. 83).

– (1996). *Invariant gray scale features*. Tech. rep. 8/96. Technische Informatik I, Technische Universität Hamburg-Harburg (cit. on p. 83).

Scovanner, P., S. Ali, and M. Shah (2007). „A 3-dimensional SIFT descriptor and its application to action recognition." In: *Proc. of the 15th international conference on Multimedia*. ACM Press New York, NY, USA, pp. 357–360 (cit. on pp. 29, 140).

Seber, G.A.F. (1984). *Multivariate observations*. Vol. 41. Wiley New York etc. (cit. on p. 16).

Shaw, P.J. and D.J. Rawlins (1991). „The point-spread function of a confocal microscope: its measurement and use in deconvolution of 3-D data." In: *Journal of Microscopy* 163.2, pp. 151–165 (cit. on p. 15).

Sheng, Y. and L. Shen (1994). „Orthogonal Fourier-Mellin moments for invariant pattern recognition." In: *JOSA A* 11.6, pp. 1748–1757 (cit. on pp. 20, 31).

Shilane, Philip, Patrick Min, Michael Kazhdan, and Thomas Funkhouser (2004). „The Princeton Shape Benchmark." In: *Proc. of the Shape Modeling and Applications, SMI*. IEEE. Genova, Italy, pp. 167–178 (cit. on p. 158).

Siggelkow, S. (2002). „Feature Historgrams for Content-Based Image Retrieval." PhD thesis. Albert-Ludwigs-Universität Freiburg (cit. on p. 83).

Simoncelli, E.P. and W.T. Freeman (1995). „The steerable pyramid: A flexible architecture for multi-scale derivative computation." In: *Proc. of the ICIP*. Vol. 3. IEEE. Washington, DC, USA, pp. 444–447 (cit. on p. 130).

Skibbe, H. and M. Reisert (2011). „Dense Rotation Invariant Brain Pyramids for Automated Human Brain Parcellation." In: *Proc. of the Informatik 2011, Workshop on Emerging Technologies for Medical Diagnosis and Therapy*. Berlin, Germany (cit. on pp. 32–34, 110, 115, 124, 184, 185).

– (2012a). „Circular Fourier-HOG features for rotation invariant object detection in biomedical images." In: *Proc. of the ISBI*. IEEE. Barcelona, Spain, pp. 450–453 (cit. on pp. 31, 34, 138, 225).

– (2012b). „Detection of Unique Point Landmarks in HARDI Images of the Human Brain." In: *Proc. of the Workshop on Computational Diffusion MRI (CDMRI'12), part of the MICCAI 2012*. Nice, France (cit. on pp. 32–34, 110, 115, 184, 226, 228).

– (2013). „Rotation Covariant Image Processing for Biomedical Applications." In: *Computational and Mathematical Methods in Medicine, Special Issue on Mathematical Methods in Biomedical Imaging* 2013 (cit. on pp. 34, 62, 100, 157, 184, 226).

Skibbe, H., M. Reisert, Q. Wang, O. Ronneberger, and H. Burkhardt (2009a). „Fast Computation of 3D Spherical Fourier Harmonic Descriptors - A Complete Orthonormal Basis for a Rotational Invariant Representation of Three-Dimensional Objects." In: *Proc. of the 3DIM, part of the ICCV*. IEEE. Kyoto, Japan (cit. on pp. 32, 33, 61).

Skibbe, H., M. Reisert, O. Ronneberger, and H. Burkhardt (2009b). „Increasing the Dimension of Creativity in Rotation Invariant Feature Design Using 3D Tensorial Harmonics." In: *Proc. of the DAGM*. Jena, Germany: LNCS, Springer, pp. 141–150 (cit. on pp. 32, 33, 75, 77, 79, 83, 200).

Skibbe, H., M. Reisert, T. Schmidt, K. Palme, O. Ronneberger, and H. Burkhardt (2010). „3D Object Detection using a Fast Voxel-Wise Local Spherical Fourier Tensor Transformation." In: *Proc. of the DAGM*. Darmstadt, Germany: LNCS, Springer, pp. 412–421 (cit. on pp. 33, 34, 83, 130, 133, 181, 184).

Skibbe, H., M. Reisert, and H. Burkhardt (2011a). „Gaussian Neighborhood Descriptors for Brain Segmentation." In: *Proc. of the MVA*. IAPR. Nara, Japan (cit. on pp. 32–34, 110, 115, 124, 184, 185).

– (2011b). „SHOG - Spherical HOG Descriptors for Rotation Invariant 3D Object Detection." In: *Proc. of the DAGM*. Frankfurt, Germany: LNCS, Springer, pp. 142–151 (cit. on pp. 32–34, 138, 154, 160, 166, 179, 226).

Skibbe, H., M. Reisert, T. Schmidt, T. Brox, O. Ronneberger, and H. Burkhardt (2012). „Fast Rotation Invariant 3D Feature Computation utilizing Efficient Local Neighborhood Operators." In: *IEEE Trans. Pattern Anal. Mach. Intell.* (Cit. on pp. 29, 32–34, 100, 110, 130, 150, 157, 171, 181, 184).

Skibbe, H., M. Reisert, O. Ronneberger, and H. Burkhardt (April, 2011). „Spherical Bessel Filter for 3D Object Detection.“ In: *Proc. of the ISBI*. IEEE. Chicago,Illinois, USA (cit. on pp. 33, 130, 160, 226).

Sorgi, L., N. Cimminiello, and A. Neri (2006). „Keypoints Selection in the Gauss Laguerre Transformed Domain.“ In: *Proc. of the BMVC*. BMVA. Edinburgh, UK, II:539 (cit. on p. 110).

Spath, H. (1985). *The Cluster Dissection and Analysis Theory FORTRAN Programs Examples*. Prentice-Hall, Inc. (cit. on p. 16).

Thomson, J. J. (Mar. 1904). „On the Structure of the Atom: an Investigation of the Stability and Periods of Oscillation of a number of Corpuscles arranged at equal intervals around the Circumference of a Circle; with Application of the Results to the Theory of Atomic Structure.“ In: *Philosophical Magazine Series 6* 7.39, pp. 237–265 (cit. on pp. 130, 141).

Tola, E., V. Lepetit, and P. Fua (2008). „A fast local descriptor for dense matching.“ In: *Proc. of the CVPR*. IEEE, pp. 1–8 (cit. on p. 29).

Tuch, DS., RM. Weisskoff, JW. Belliveau, and VJ. Wedeen (1999). „High angular resolution diffusion imaging of the human brain.“ In: *Proc. of the ISMRM*. Philadelphia,USA (cit. on pp. 123, 184).

Tuszynski, J.A. (1990). *Spherical Tensor Operators: Tables of Matrix Elements and Symmetries*. World Scientific (cit. on p. 62).

Tuytelaars, T. and K. Mikolajczyk (2008). „Local invariant feature detectors: a survey.“ In: *Foundations and Trends® in Computer Graphics and Vision* 3.3, pp. 177–280 (cit. on p. 28).

Vedaldi, A., M. Blaschko, and A. Zisserman (2011). „Learning equivariant structured output SVM regressors.“ In: *Proc. of the ICCV*. IEEE. Barcelona, Spain, pp. 959–966 (cit. on p. 20).

Venkatraman, V., L. Sael, and D. Kihara (2009). „Potential for protein surface shape analysis using spherical harmonics and 3D Zernike descriptors.“ In: *Cell biochemistry and biophysics* 54.1, pp. 23–32 (cit. on p. 32).

Viola, P. and M. Jones (2001). „Rapid object detection using a boosted cascade of simple features.“ In: *Proc. of the CVPR*. Vol. 1. IEEE, pp. I–511 (cit. on p. 139).

Wang, Q., O. Ronneberger, and H. Burkhardt (2009). „Rotational Invariance Based on Fourier Analysis in Polar and Spherical Coordinates.“ In: *IEEE Trans. Pattern Anal. Mach. Intell.* 31, pp. 1715–1722 (cit. on p. 32).

Weickert, J. (1998). *Anisotropic diffusion in image processing*. Vol. 1. Teubner Stuttgart (cit. on p. 139).

Weniger, Ernst Joachim (2005). „The Spherical Tensor Gradient Operator.“ In: *Collect. Czech. Chem. Commun.* Vol. 70, pp. 1225–1271. arXiv:`0505018v1 [math-ph]` (cit. on p. 97).

Wigner, E.P. (1931). *Gruppentheorie und ihre Anwendungen auf die Quantenmechanik der Atomspektren*. Braunschweig: Vieweg Verlag (cit. on pp. 53, 221).

Wigner, E.P. and J.J. Griffin (1959). *Group Theory and Its Application to the Quantum Mechanics of Atomic Spectra*. Pure and applied Physics. New York: Academic Press (cit. on p. 221).

Wormer, Paul E.S. „Angular Momentum Theory.“ In: *Lecture Notes - University of Nijmegen Toernooiveld, 6525 ED Nijmegen, The Netherlands* (cit. on pp. 62, 70).

Xu, C. and J.L. Prince (1997). „Gradient vector flow: A new external force for snakes.“ In: *Proc. of the CVPR*. IEEE, pp. 66–71 (cit. on p. 123).

Yap, P.T., X. Jiang, and A.C. Kot (2010). „Two-dimensional polar harmonic transforms for invariant image representation.“ In: *IEEE Trans. Pattern Anal. Mach. Intell.* 32.7, pp. 1259–1270 (cit. on p. 31).

Yoshimura, H., M. Etoh, K. Kondo, and N. Yokoya (2000). „Gray-scale character recognition by Gabor jets projection.“ In: *Proc. of the ICPR*. Vol. 2. IEEE. Barcelona, Spain, pp. 335–338 (cit. on p. 128).

Zahn, C.T. (1971). „Graph-theoretical methods for detecting and describing gestalt clusters.“ In: *Computers, IEEE Transactions on* 100.1, pp. 68–86 (cit. on p. 16).

Zhu, Q., M.C. Yeh, K.T. Cheng, and S. Avidan (2006). „Fast human detection using a cascade of histograms of oriented gradients.“ In: *Proc. of the CVPR*. Vol. 2. IEEE, pp. 1491–1498 (cit. on p. 139).