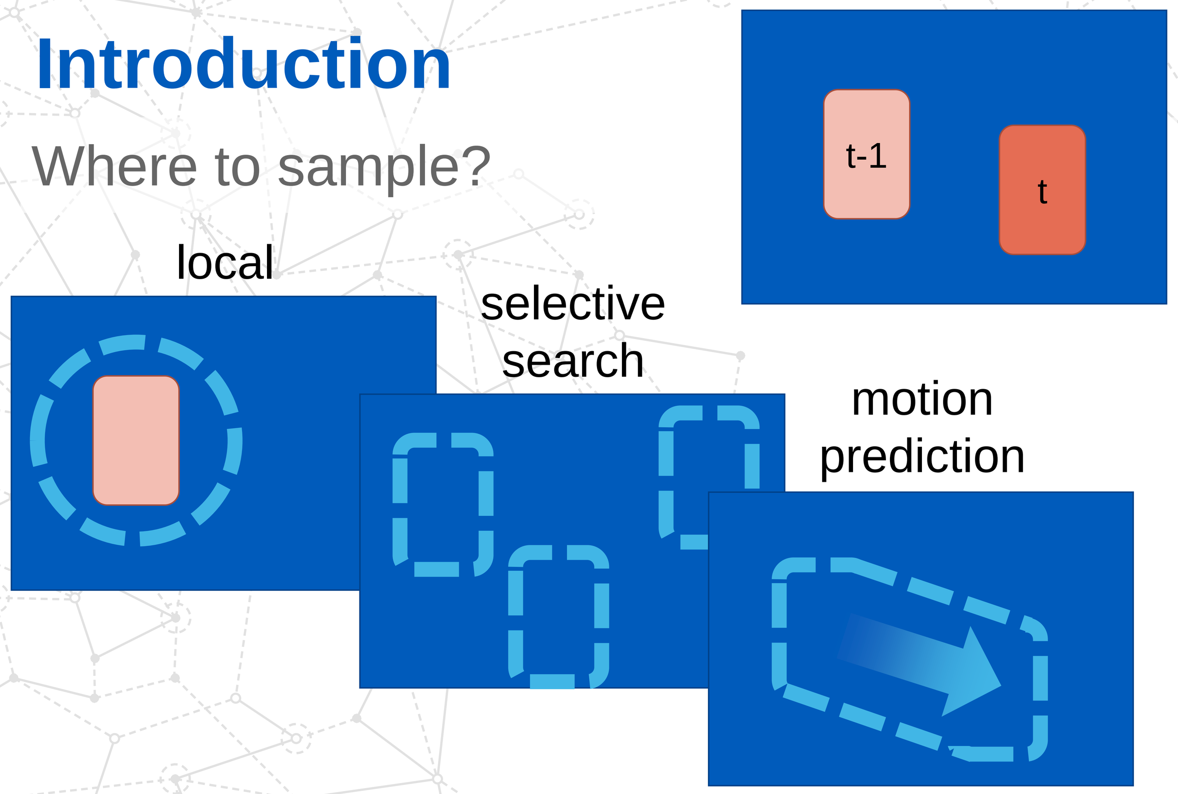
INFORMATION-MAXIMIZING SAMPLING TO PROMOTE TRACKING-BY-DETECTION

Kourosh Meshgi [1] Maryam Sadat Mirzaei [1,2] Shigeyuki Oba [1]

[1] Kyoto University, Graduate School of Informatics [2] Riken Center for Advanced Artificial Intelligence Project

Introduction

Where to sample?



Intuition

Sampling affect **accuracy**, **robustness** and **time complexity**:

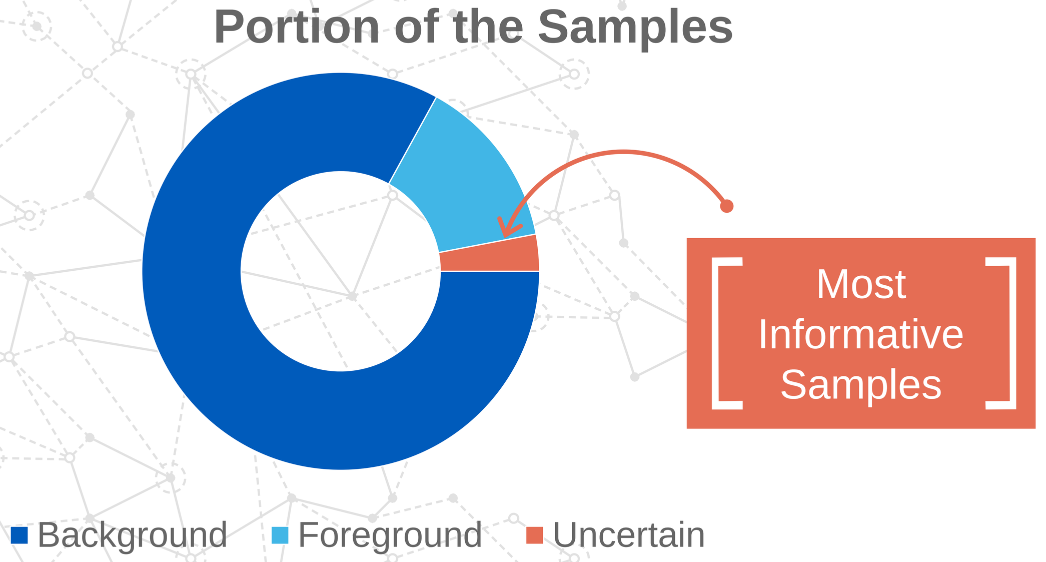
UNINFORMED SAMPLING

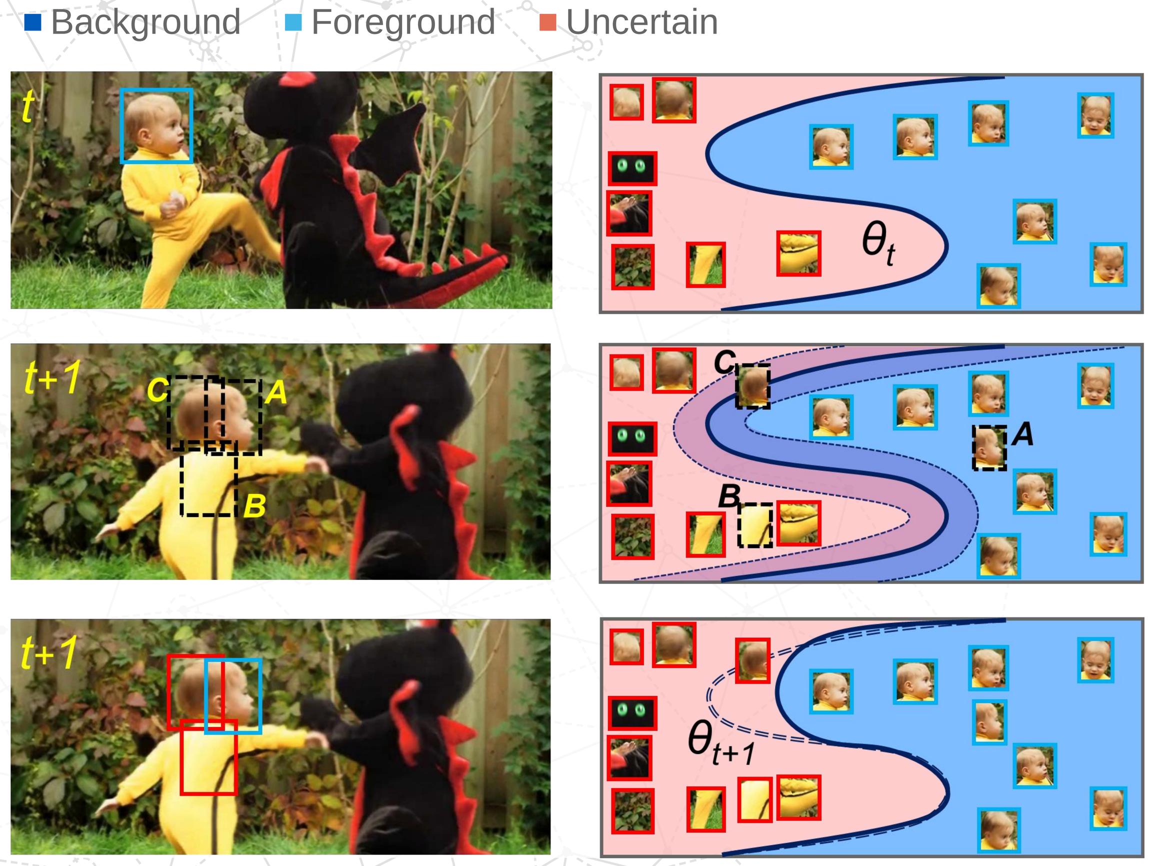
- **Exhaustive:** Sliding Window, Dense Sampling, etc.
- **Stochastic:** Motion model, Context, Saliency

INFORMED SAMPLING

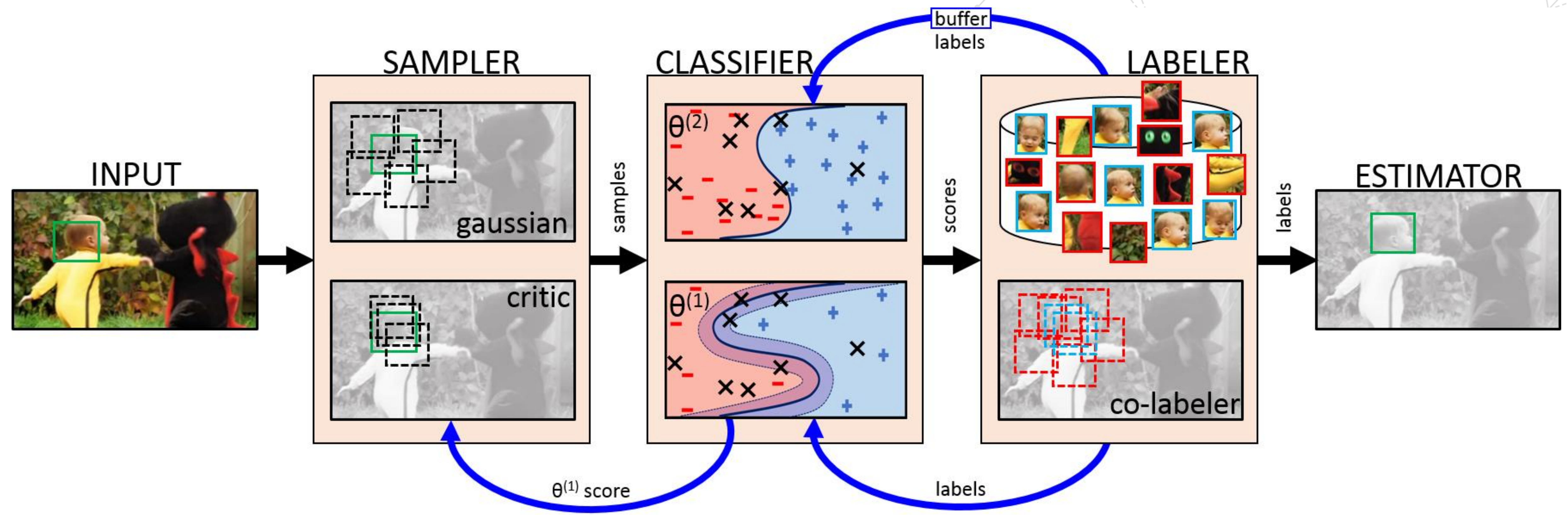
- **Object Proposals:** Edge Boxes, CPMC, BING, etc. [1]
- **Classifier Induced:** R-CNN based, Label Uncertainty

Portion of the Samples





Method

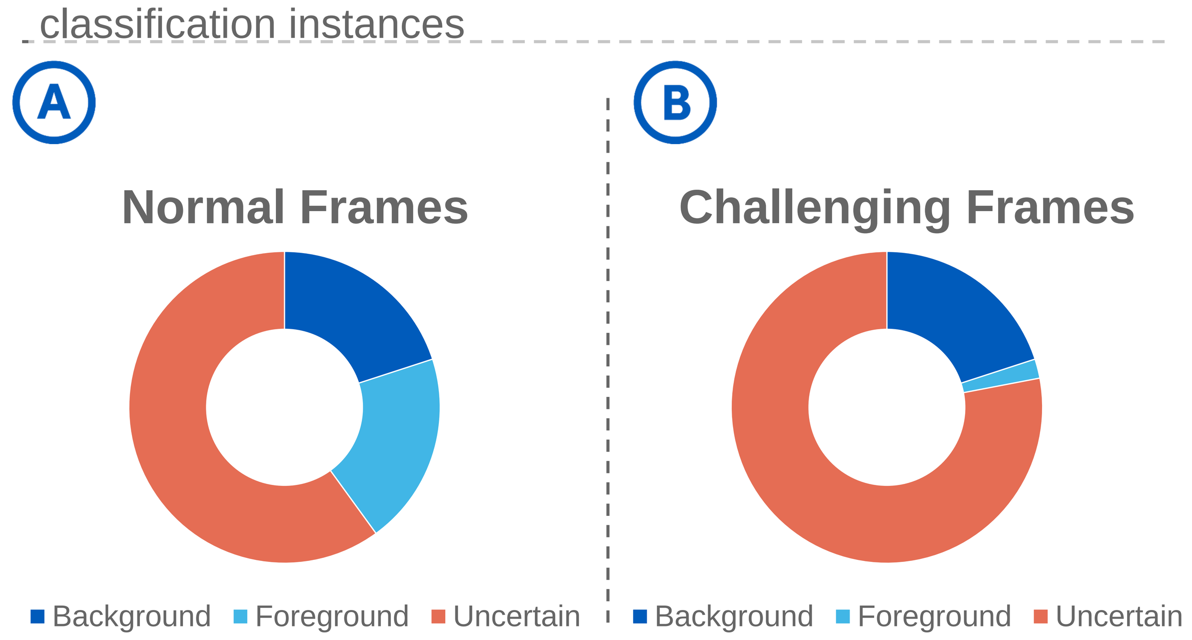


Proposal

Information Maximizing Sampling Strategy: by exploiting the uncertainties of the classifier, we try to obtain samples that knowing their labels, would maximally improve the classification accuracy, in other words, most informative samples.

Proposed: A "critic" that tries to expose the weaknesses of the tracker's classifier, and the classifier tries to improve its classification in those area to provide a good classification for those sort of samples. (**Adversarial Search,** Zero-sum game, GAN style)

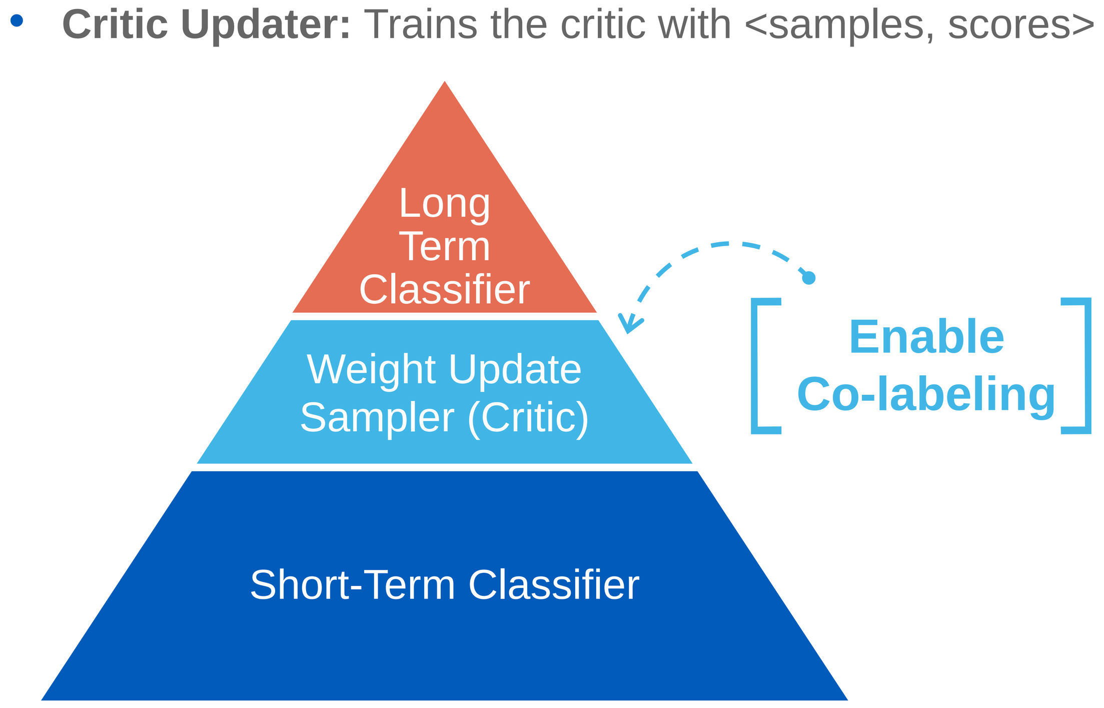
Query optimization (Active Learning): Select only the samples that knowing their label maximize the information of the classifier (optimize its training) to handle the next classification instances



The **sampling** and **classification** have two different objectives. While the former tries to provide better samples from the target, the latter tries to construct a better classifier, demanding representative negative samples and supports for defining an accurate classification boundary.

Architecture

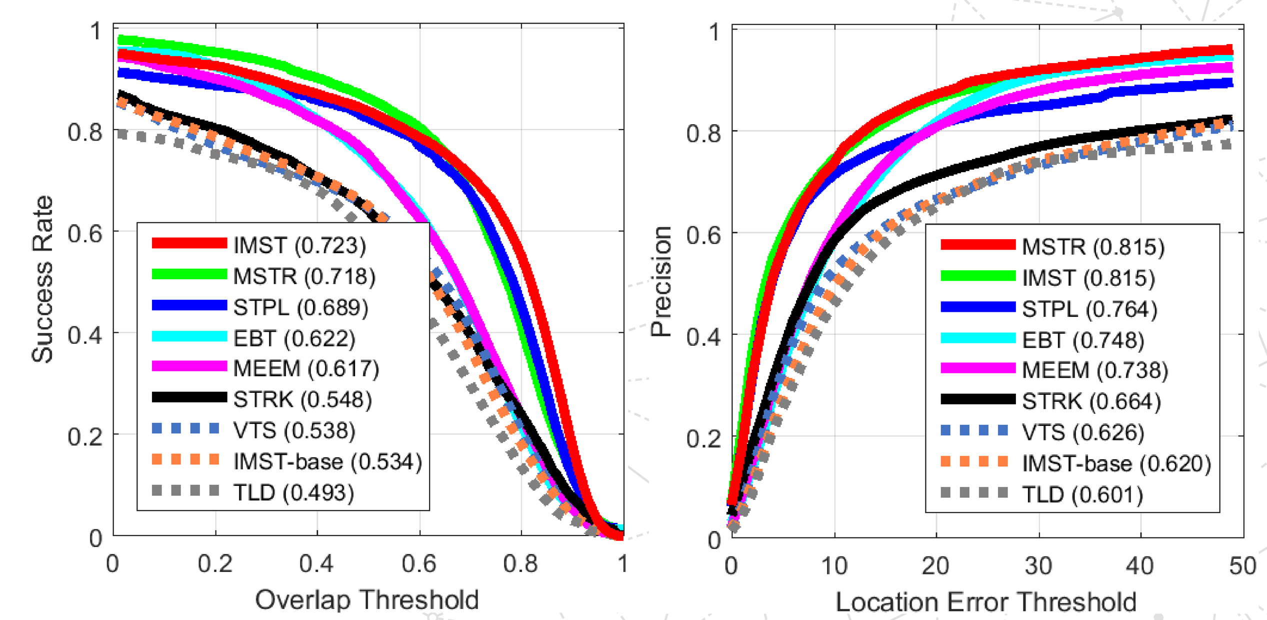
- **Gaussian Sampler:** based on the last known target position, first half of the samples
- **Critic Sampler:** Second half of the samples that challenges the classifier to improve its decision boundary
- **Short-term Classifier:** quick naïve feature-based classifier
- **Long-term Classifier:** slow sophisticated classifier [2]
- **Query Optimizer:** the data they want the other classifier to label, queries the most uncertain samples
- **Weight Adjuster:** Calculate the weights of classifiers
- **Co-labler:** Label each sample based on the outputs of two classifiers and their weight [3]
- **Short Updater:** Updates short-term classifier every frame
- **Long Updater:** Updates long-term classifier every several frames
- **Critic Updater:** Trains the critic with <samples, scores>



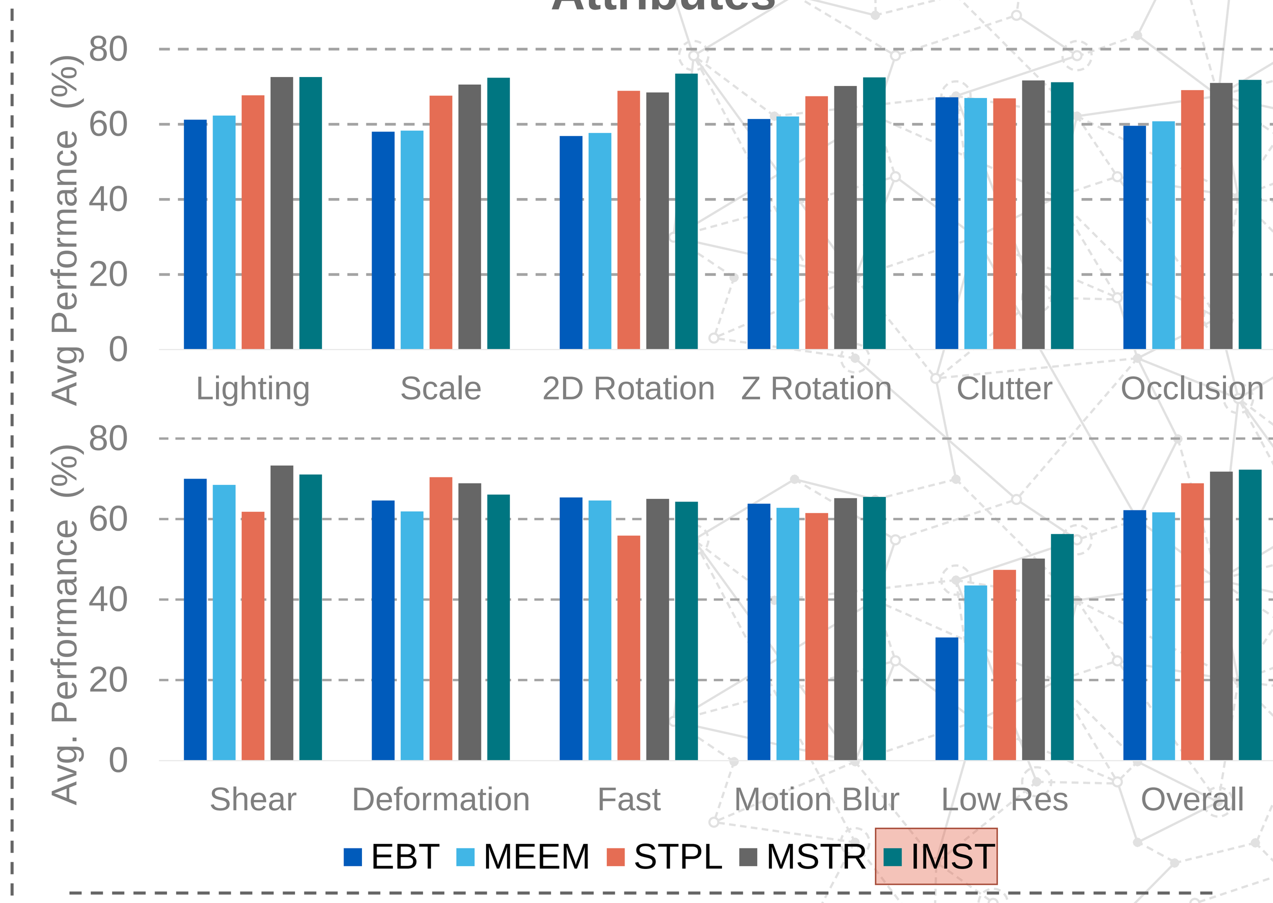
CRITIC IMPLEMENTATION

- **Structured output SVM** a discriminant that can predict if a sample is challenging for short-term classifier
- **Budgeted update** [4] with every sample and its score

Results



Attributes



Conclusion

obtain samples by considering target's spatiotemporal properties & uncertainty-analysis of the classifier, provides the required labels from a long-memory auxiliary classifier:

- **Generalization & speed-up** by querying only the most informative samples from the long-memory complex classifier
- **Balance** between long and short term memory
- **Uncertainty reduction for label classification**
- **Breaks** self learning loop

References

1. Zhu et al, Beyond local search: Tracking objects everywhere with instance specific proposals, CVPR'16.
2. Meshgi et al., Robust discriminative tracking via query-by-committee, AVSS'16.
3. Tang et al., Co-tracking using semi-supervised support vector machines, ICCV'07.
4. Hare et al., Struck: Structured output tracking with kernels, ICCV'11.

http://ishiilab.jp/member/meshgi-k/imst.html