15th IAPR International Conference on Machine Vision Applications (MVA)
Nagoya University, Nagoya, Japan, May 8-12, 2017.

Active Discriminative Tracking using Collective Memory

Kourosh MESHGI, Shigeyuki OBA, Shin ISHII
Graduate School of Informatics
Kyoto University, Kyoto, Japan
{meshgi-k,ishii,oba}@sys.i.kyoto-u.ac.jp

Abstract

Ever changing appearance of the targets in real-world scenarios mandates a discriminative tracker to update its classifier(s) on-the-fly, a process during which the model could be updated with irrelevant/noisy data, causing the tracker to drift away from the target over time. The updates should be frequent enough to reflect the latest changes in the target's appearance, whereas the tracker should keep the memory of previous templates to recover from occlusions or temporal variations in appearance of the target (aka the plasticity-stability dilemma). In this study, we proposed a committee of classifiers with different memory spans, to address the appearance changes with various durations. An active learning scheme selects the most disputed samples and queries their labels from a less-frequently updated long-term memory oracle. This combination of memory spans balances the plasticity-stability equilibrium as demonstrated by the experiments and provides a comparable performance to the state-of-the-art trackers with a relatively simple implementation.

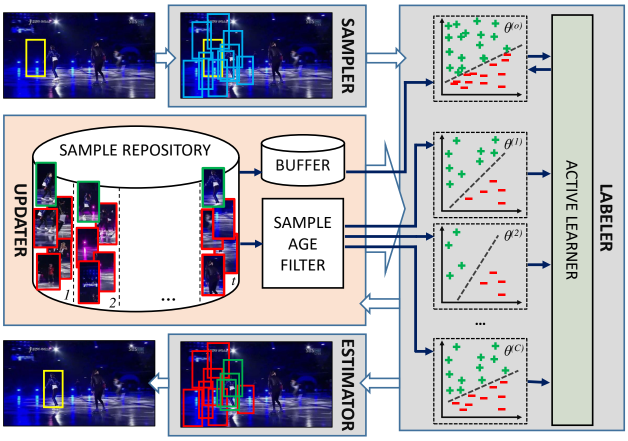


Figure 1. Schematic of proposed tracker, Collective Memory Tracker (CMT). In this tracker, the samples are labeled by a novel labeler unit, which uses a committee of classifiers with different short-term memory spans, a long-term memory classifier, and an active learner to balance the interaction of these two memories.

1 Introduction

Discriminative tracking is the task of separating the target object from its background [1, 2, 3, 4, 5, 6, 7, 8, 9], which is usually in contrast with generative tracking that focuses on the target itself [10]. Numerous approaches have been proposed in the discriminative tracking framework ranging from simple object detector-based methods [11] to context-aware trackers [4], discriminative correlation filters [12], and ensemble tracking [2, 9].

Ensemble tracking employs a committee of (independent) classifiers to express their ideas about target location or every single sample of data. Such methods select the best classifier to represent the target [7], combine the long-term and short-term memory by fusing different trackers [8], or re-evaluate classifier's labels with auxiliary detectors [6].

A key component for achieving a robust longterm tracking is the tracker's capability of updating its internal representation of targets (the appearance model) to changing conditions. This update should accommodate the rapid changes in the target's shape and appearance, however, an unsupervised rapid update scheme renders the tracker prone to be updated with misclassified data samples. On the other hand, temporal appearance changes and occlusions, may drift the target model away quickly if the update frequency is too high. In such cases, the trackers may need to remember the only user-annotated frame ($t = 1$) or some early tracking results to recover from such situation. The need to simultaneous fulfillment of these contradicting goals of rapid learning and stable memory may be called the "stability-plasticity dilemma"[13]. To address this challenge, different approaches were proposed. TLD [6] utilizes two auxiliary detectors (for false-positive and false-negatives) to verify the labeling process, STRUCK [5] avoids the labeling process to alleviate the model drift problem, MEEM [7] proposed a restoration mechanism to roll-back the faulty updates, and MUSTer [8] utilizes a bio-inspired memory model to combine short and long-term memories.

In this study, we propose an intuitive way to incorporate different memories into a tracker and balance the stability-plasticity equilibrium. The proposed tracker utilizes the shared pool of knowledge and information in the memories of two or more members of a committee to label a data sample. Beside having this collective memory to handle short-term target variations, another long-term memory classifier is used to enable the tracker to handle temporal variations and occlusions, and re-identify the target in the less-likely scenario of target loss. To balance these two memories, an active learning mechanism is proposed in this tracker, in which, the most disputed samples in the short-memory (thus the most informative ones), are queried from the long-term memory module.

Following this, the proposed method, Collective Memory Tracker (CMT) is elaborated and compared against the best-performing discriminative trackers on a large public dataset. The findings reveal that CMT's performance is comparable to the state-of-the-art, yet this tracker could be enhanced in various ways to achieve higher performances.

2 Collective Memory Tracker (CMT)

In this section, an adaptive discriminative tracker is formulated and later expanded to ensemble trackers. Built on this foundation, the proposed method, CMT, is elaborated.

2.1 Discriminative Tracking

To determine the position of the target $\mathbf{p}_t$ at time $t \in \{1, \dots, T\}$, a discriminative tracker strives to find a transformation $\mathbf{y}_t$ relative to the target's last known position $\mathbf{p}_{t-1}$, where $\mathbf{p}_t = \mathbf{p}_{t-1} \circ \mathbf{y}_t$. The transformation $\mathbf{y}_t \in \mathcal{Y}$ is a member of transformation space $\mathcal{Y}$, which is commonly induced by 2D translations, but could be generalized to more sophisticated spaces (e.g. 2D affine transformations space) without the loss of generality. In addition, the motion models, context [4], and generated confidence maps [14] may be considered in selecting samples.

A common approach to estimate $\mathbf{y}_t$ is by generating a set of n different samples $\{\mathbf{y}_t^1, \dots, \mathbf{y}_t^n\}$, and aggregating them based on a utility function $f(\mathbf{y}_t^j)$, i.e.,

$$\mathbf{y}_t = g(\mathbf{y}_t^1, \dots, \mathbf{y}_t^n, f(\mathbf{y}_t^1), \dots, f(\mathbf{y}_t^1)). \tag{1}$$

Each sample $\mathbf{y}_t^j$ indicates the location $\mathbf{p}_{t-1} \circ \mathbf{y}_t^j$ in the video frame I_t, where the image patch $\mathbf{x}_t^{\mathbf{p}_{t-1} \circ \mathbf{y}_t^j} \in \mathcal{X}_t$ is contained. The samples are evaluated by the tracker's classifier, θ_t, which makes its predictions via its classification confidence function $h : \mathcal{X}_t \to \mathbb{R}$. In traditional tracking-by-detection algorithms this confidence score serves as the utility for eq(1),

$$\mathbf{s}_t^j = h(\mathbf{x}_t^{\mathbf{p}_{t-1} \circ \mathbf{y}_t^j} | \theta_t). \tag{2}$$

from which the label of the sample j can be inferred,

$$\ell_t^j = \operatorname{sign}(\mathbf{s}_t^j). \tag{3}$$

In turn, all of the samples and their labels are used to re-train the classifier's model θ_t,

$$\theta_{t+1} = u(\theta_t, \mathcal{X}_{t-\Delta:t}, \mathcal{L}_{t-\Delta:t}) \tag{4}$$

in which $\mathcal{L}_t$ denotes the set of labels of the samples $\mathcal{X}_t$, $u(.)$ is the model update function, and the Δ is the history that a tracker considers in it re-training.

A typical discriminative tracker tries to maximize the response of the classifier, i.e., (1) becomes

$$\hat{\mathbf{y}}_t = \underset{\mathbf{y}_t^j \in \mathcal{Y}}{\operatorname{argmax}}(f(\mathbf{y}_t^j)), \tag{5}$$

however other methods such as weighted averaging of the transfomations are proposed in the literature [9]. Furthermore, many of the adaptive trackers utilize online-learning classifiers [5, 10] in which only the data from the recent frame ($\Delta = 1$) is used.

Algorithm 1: Collective Memory Tracker (CMT)

```
input  : Target position in last frame $\mathbf{p}_{t-1}$
output: Target position in current frame $\mathbf{p}_t$
for $j \leftarrow 1$ to $n$ do
    Sample transformation $\mathbf{y}_t^j \sim \mathcal{N}(\mathbf{p}_t, \Sigma_{search})$
    Calculate committee score $s_t^j$ (eq(6))
    Label the sample $\ell_t^j$ (eq(9))
    Archive $\langle t, s_t^j, \ell_t^j \rangle$ in $\mathcal{D}$
for $c \leftarrow 1$ to $C$ do
    Retrain $\theta_t^{(c)}$ by $\mathcal{D}$ considering $\Delta^{(c)}$ (eq(8))
if $\mathrm{mod}(t, \Delta^{(o)}) = 0$ then update the oracle
    Retrain $\theta^{(o)}$ considering all samples of $\mathcal{D}$
Estimate transformation $\hat{\mathbf{y}}_t$ (eq(10))
Calculate target position $\mathbf{p}_t = \mathbf{p}_{t-1} \circ \hat{\mathbf{y}}_t$
```

2.2 Ensemble Discriminative Tracking

An ensemble discriminative tracker employs a set of classifiers instead of one. These classifiers, hereafter called *committee*, are represented by $\mathcal{C} = \{\theta^{(1)}, \dots, \theta^{(C)}\}$, and are typically homogeneous and independent. Popular ensemble trackers utilize the majority voting of the committee as their utility function,

$$\mathbf{s}_t^j = \sum_{c=1}^{C} \operatorname{sign}\big(h(\mathbf{x}_t^{\mathbf{p}_{t-1} \circ \mathbf{y}_t^j} | \theta_t^{(c)})\big). \tag{6}$$

and eq(3) is used to label the samples. Finally, the model is updated for each classifier independently,

$$\theta_{t+1}^{(c)} = u(\theta_t^{(c)}, \mathcal{X}_{t-\Delta:t}, \mathcal{L}_{t-\Delta:t}) \tag{7}$$

meaning that all of the committee members are trained with a similar set of samples and a common label for them. On the other hand, in co-tracking algorithms (such as [14]), different classifiers have different sample set $\mathcal{X}_t^{(c')}$ and label them based on their own models ($\mathcal{L}_t^{(c)}$).

2.3 Proposed Method

The proposed algorithm, Collective Memory Tracker (CMT), is based on the premise that different memory spans throughout the tracking, result in different classifiers. By leveraging different classifiers induced by different depth of the memory, this tracker harness the power of collective memory to balance the stability-plasticity equilibrium, i.e., this mixture of memory spans strives to balance the adaptation power of the tracker to recent changes of the target, with its memory of its initial template and earlier tracking results. The different memory spans, $\Delta^{(c)}$, for the committee members ($1 \leq \Delta^{(c)} \leq T$) promote the diversity of the committee by providing different training data,

$$\theta_{t+1}^{(c)} = u(\theta_t^{(c)}, \mathcal{X}_{t-\Delta^{(c)}:t}, \mathcal{L}_{t-\Delta^{(c)}:t}) \tag{8}$$

It is natural that rapid appearance changes, occlusions, and permanent target variations cause the increase of the disagreement about some samples. To address this issue, inspired by query-by-committee [15] we select the most disputed samples in each frame (with s_t^j closer to zero), and label them using an auxiliary classifier ($\theta^{(o)}$) with a long-term memory, hereafter

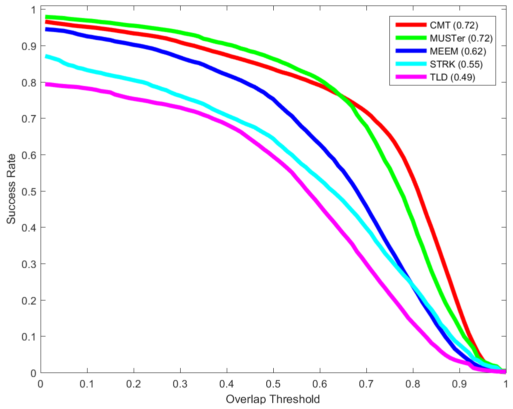


Figure 2. Quantitative performance comparison of the proposed tracker, CMT, with the state-of-the-art trackers using success plot and its AUC.

called the *oracle*. Therefore, eq(3) can be written as

$$\ell_t^j = \begin{cases} +1 & \mathbf{s}_t^j > \tau_u \\ -1 & \mathbf{s}_t^j < \tau_l \\ \text{sign}(h(\mathbf{x}^{\mathbf{p}_{t-1} \circ \mathbf{y}_t^i}) | \theta_t^{(o)}) & \text{otherwise} \end{cases} \tag{9}$$

where τ_u and τ_l are thresholds with which the tracker controls its reliance on the oracle. To estimate the next target location, the patch that maximizes the sum of raw responses of the committee is used

$$\hat{\mathbf{y}}_t = \underset{\mathbf{y}_t^j \in \mathcal{Y}}{\text{argmax}} \left(\sum_{c=1}^{C} h(\mathbf{x}_t^{\mathbf{p}_{t-1} \circ \mathbf{y}_t^j} | \theta_t^{(c)}) \right). \tag{10}$$

Algorithm 1 summarizes the proposed tracker.

As illustrated in Figure 1, the samples along with their labels are stored in a repository $\mathcal{D}$, and the committee members are trained with the samples in their memory span every frame. In addition, the oracle is retrained every $\Delta^{(o)}$ frames with all the samples, that renders the tracker robust against the temporal variations of target appearance and occlusions. The the memory spans $\Delta^{(c)} = \{2, 3, 5, 7, 11\}$ and $\Delta^{(o)} = 15$, thresholds $\tau_u = 0.39$ and $\tau_l = -0.43$, and the search radius $\Sigma_{search} = 6$ pixels are tuned by cross-validation.

In this study, we used homogeneous classifiers (KNNs) for the committee members with the similar set of features (HOG+HOC), thus, the features for each sample is calculated once but classified C times by committee members. This design also facilitates the storage of the samples in $\mathcal{D}$, and speed up classifier retraining by reusing the computations for trackers with shorter memory spans for those with longer ones. The oracle is a modified part-based classifier based on [11].

3 Experiments

To establish a fair comparison with the state-of-the-art, we select several discriminative trackers with active memory management: MUSTer [8], MEEM [7], STRUCK [5], and TLD [6]. We perform a benchmark on the 50 videos of the Object Tracking Benchmark [16], along with partial subsets of the dataset with a distinguishing attribute to evaluate the tracker performance under different situations. These attributes are illumination variation (IV), scale variation (SV), occlusions (OCC), deformation (DEF), motion blur (MB), fast motion (FM), in-plane-rotation (IPR), out-of-plane rotation (OPR), out-of-view (OV), low resolution (LR), and background clutter (BC), defined based on the biggest challenges that a tracker may face throughout tracking. CMT achieved the speed of 21.97 fps on a Pentium IV PC @ 3.5 GHz and a Matlab/C++ implementation with no code optimization.

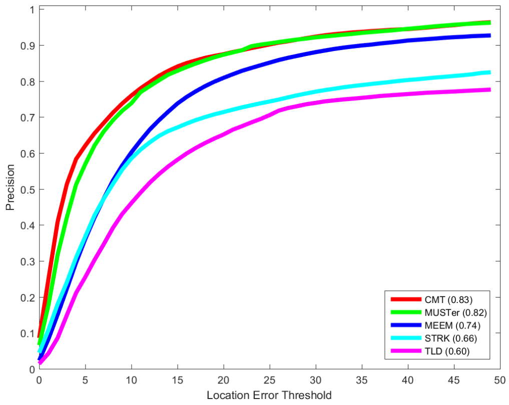


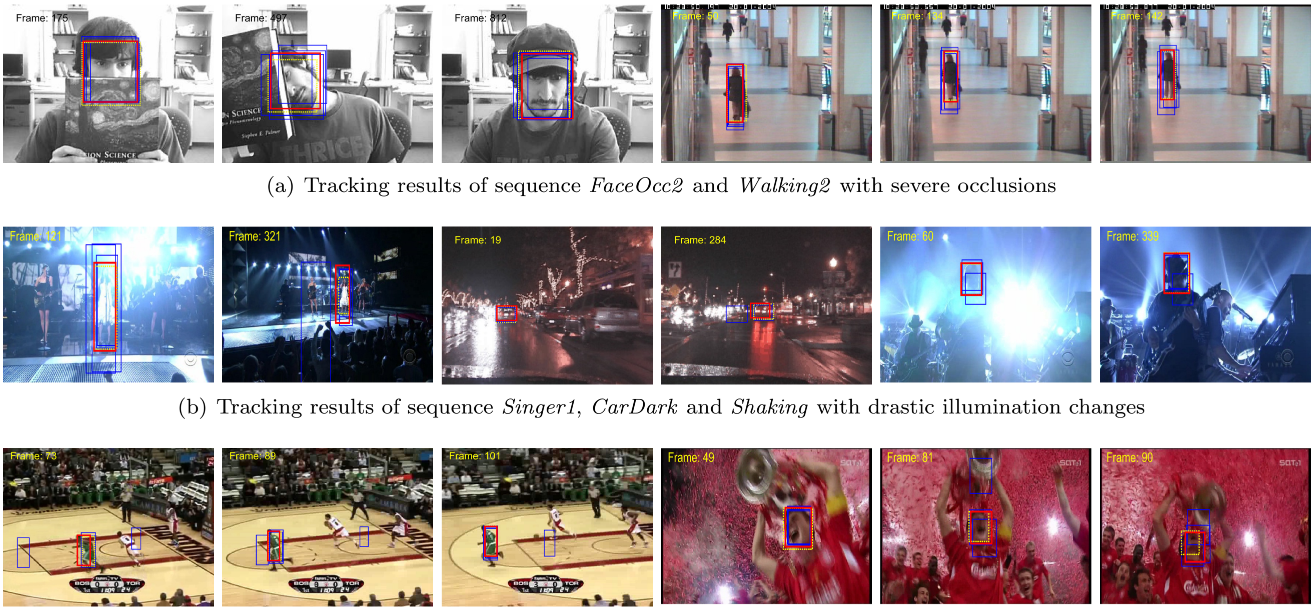
Figure 3. Quantitative localization accuracy comparison of CMT, with the state-of-the-art trackers using success plot and its AUC.

For this comparison, we have used success and precision plots, where their area under curve provides a robust metric for comparing tracker performances [16].

The success plot depicted in Figure 2, demonstrates that CMT (proposed) and MUSTer outperformed other trackers considering all of the videos. Table 1 provides a detailed analysis of the trackers based on the aforementioned attributes, where in all categories these two trackers dominate the other trackers. This leading performance could be attributed to the effective methods CMT and MUSTer employ to balance the stability-plasticity equilibrium. This superior performance is especially evident in the categories in which the appearance of the target underwent transformations (IV, DEF, SV, IPR, and OPR) or was partially invisible to the camera (OCC and OV). Whilst Table 1 shows a comparable performance between CMT and MUSTer, the precision plot (Figure 3) illustrates that CMT is slightly more accurate than MUSTer, though the difference is not significant.

Table 1. Quantitative evaluation of trackers under different visual tracking challenges using AUC of success plot. The best performance for each attribute is shown in **bold**.

Attribute	TLD	STRK	MEEM	MUSTer	Ours
IV	0.48	0.53	0.62	**0.73**	**0.73**
DEF	0.38	0.51	0.62	**0.69**	**0.69**
OCC	0.46	0.50	0.61	0.69	**0.71**
SV	0.49	0.51	0.58	0.71	**0.72**
IPR	0.50	0.54	0.58	0.69	**0.74**
OPR	0.48	0.53	0.62	0.70	**0.73**
OV	0.54	0.52	0.68	**0.73**	0.71
LR	0.36	0.33	0.43	0.50	**0.55**
BC	0.39	0.52	0.67	**0.72**	0.69
FM	0.45	0.52	0.65	0.65	**0.70**
MB	0.41	0.47	0.63	**0.65**	**0.65**
ALL	0.49	0.55	0.62	**0.72**	**0.72**



(a) Tracking results of sequence *FaceOcc2* and *Walking2* with severe occlusions

(b) Tracking results of sequence *Singer1*, *CarDark* and *Shaking* with drastic illumination changes

(c) Tracking results of sequence *Basketball* and *Soccer* with deformation

Figure 4. Sample tracking results of evaluated algorithms on several challenging video sequences, in these sequences the red box depicts the CMT against other trackers (blue). The ground truth is illustrated with yellow dashed box. The results are available in the http://ishiilab.jp/member/meshgi-k/cmt.html.

4 Conclusion

This study proposed to employ a committee of classifiers with different memory spans to epitomize a collective memory. Different memory spans diversify the committee members, which boost the performance of the resulting ensemble tracker. Furthermore, it enables the use of deterministic classifiers in the query-by-committee learning framework. This active learning framework efficiently detects the most informative samples (i.e., the most disputed ones in this study) and query their labels from a long-term memory oracle. The balance in stability-plasticity equilibrium is achieved by the combination of several short-term classifiers with a long-term classifier, and managing their interaction with an active learning mechanism.

The proposed tracker, CMT, outperforms traditional discriminative trackers and achieve a comparable performance with MUSTer that utilizes hybrid memory schemes to bolster the importance of this aspect of tracker designs. Results of the tracking on a large video dataset revealed that this method is effective in handling various tracking challenges especially occlusions and target appearance changes.

Acknowledgment

This article is based on results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

References

[1] R. T. Collins, Y. Liu, and M. Leordeanu, “Online selection of discriminative tracking features,” *PAMI*, vol. 27, no. 10, pp. 1631–1643, 2005.

[2] S. Avidan, “Ensemble tracking,” *PAMI*, vol. 29, 2007.

[3] H. Grabner, C. Leistner, and H. Bischof, “Semi-supervised on-line boosting for robust tracking,” in *ECCV'08*, 2008.

[4] H. Grabner, J. Matas, L. Van Gool, and P. Cattin, “Tracking the invisible: Learning where the object might be,” in *CVPR'10*, 2010.

[5] S. Hare, A. Saffari, and P. H. Torr, “Struck: Structured output tracking with kernels,” in *ICCV'11*, 2011.

[6] Z. Kalal, K. Mikolajczyk, and J. Matas, “Tracking-learning-detection,” *PAMI*, vol. 34, no. 7, 2012.

[7] J. Zhang, S. Ma, and S. Sclaroff, “Meem: Robust tracking via multiple experts using entropy minimization,” in *ECCV'14*. Springer, 2014, pp. 188–203.

[8] Z. Hong, Z. Chen, C. Wang, X. Mei, D. Prokhorov, and D. Tao, “Multi-store tracker (muster): a cognitive psychology inspired approach to object tracking,” in *CVPR'15*, 2015, pp. 749–758.

[9] K. Meshgi, S. Oba, and S. Ishii, “Robust discriminative tracking via query-by-committee,” in *AVSS'16*, 2016.

[10] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, “Incremental learning for robust visual tracking,” *IJCV*, vol. 77, no. 1-3, pp. 125–141, 2008.

[11] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part-based models,” *PAMI*, vol. 32, 2010.

[12] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg, “Learning spatially regularized correlation filters for visual tracking,” in *ICCV'15*, 2015, pp. 4310–4318.

[13] S. Salti, A. Cavallaro, and L. Di Stefano, “Adaptive appearance modeling for video tracking: Survey and evaluation,” *IEEE TIP*, 2012.

[14] F. Tang, S. Brennan, Q. Zhao, and H. Tao, “Co-tracking using semi-supervised support vector machines,” in *ICCV'07*, 2007.

[15] H. S. Seung, M. Opper, and H. Sompolinsky, “Query by committee,” in *COLT'92*. ACM, 1992.

[16] Y. Wu, J. Lim, and M.-H. Yang, “Online object tracking: A benchmark,” in *CVPR'13*, 2013.