This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the version available on IEEE Xplore.

Efficient Diverse Ensemble for Discriminative Co-Tracking

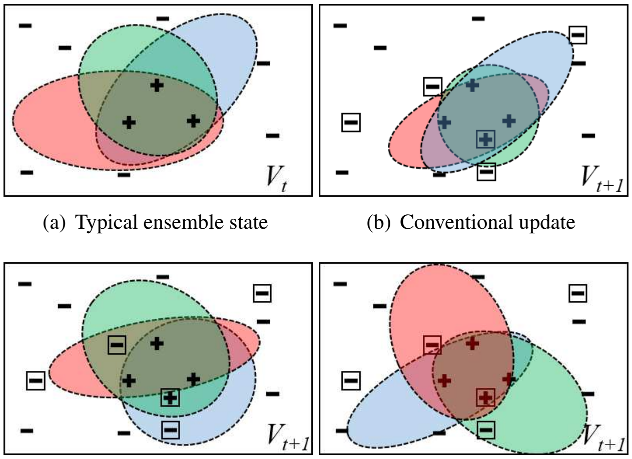
Kourosh Meshgi, Shigeyuki Oba, Shin Ishii
Graduate School of Informatics, Kyoto University
606–8501 Yoshida-honmachi, Kyoto, Japan
{meshgi-k,oba,ishii}@sys.i.kyoto-u.ac.jp

Abstract

Ensemble discriminative tracking utilizes a committee of classifiers, to label data samples, which are in turn, used for retraining the tracker to localize the target using the collective knowledge of the committee. Committee members could vary in their features, memory update schemes, or training data, however, it is inevitable to have committee members that excessively agree because of large overlaps in their version space. To remove this redundancy and have an effective ensemble learning, it is critical for the committee to include consistent hypotheses that differ from one-another, covering the version space with minimum overlaps. In this study, we propose an online ensemble tracker that directly generates a diverse committee by generating an efficient set of artificial training. The artificial data is sampled from the empirical distribution of the samples taken from both target and background, whereas the process is governed by query-by-committee to shrink the overlap between classifiers. The experimental results demonstrate that the proposed scheme outperforms conventional ensemble trackers on public benchmarks.

1. Introduction

Tracking-by-detection [3,5,19,20,22,25] as one the most popular approaches of discriminative tracking utilizes classifier(s) to perform the classification task using object detectors. In a tracking-by-detection pipeline, several samples are obtained from each frame of the video sequence, to be classified and labeled by the target detector, and this information is used to re-train the classifier in a closed feedback loop. This approach advantages from the overwhelming maturity of the object detection literature, both in the terms of accuracy and speed [11,13], yet struggles to keep up with the target evolution as it rises issues such as proper strategy, rate, and extent of the model update [32,46,55]. To adapt to object appearance changes, the tracking-by-detection methods update the decision boundary as opposed to object appearance model in generative trackers. Imperfections of target detection and model update throughout the tracking, manifest themselves as accumulating errors, which essentially drifts the model from the real target distribution, hence leads to target loss and tracking failure. Such imperfections can be caused by labeling noise, self-learning loop, sensitive online-learning schemes, improper update frequency, non-realistic assumption about the target distribution, and equal weights for all training samples.



(a) Typical ensemble state (b) Conventional update

(c) Partial update (d) Diversified update

Figure 1. Version space examples for ensemble classifiers. **(a)** All hypotheses are consistent with the previous labeled data, but each represents a different classifier in the version space V_t. In the next time step, the models are updated with the new data (boxed). **(b)** Updating with all of the data tend to make the hypothesis more overlapping. **(c)** Random subsets of training data are given to the hypotheses and they update without considering the rest of the data, the hypotheses cover random areas of the version space. **(d)** Random subsets of training data plus artificial generated data *(proposed)*, trains the hypothese to be mutually uncorrelated as much as possible, while encouraging them to cover more (unexplored) area of the version space.

Misclassification of a sample due to drastic target transformations, visual artifacts (such occlusion) or model errors not only degrades target localization accuracy, but also confuses the classifier [22] when trained by this erroneous label. Typically in tracking-by-detection, the classifier is retrained using its own output from the earlier tracking episodes (the self-learning loop), which amplitudes a training noise in the classifier and accumulate the error over time. The problem amplifies when the tracker lacks a forgetting mechanism or is unable to obtain external scaffolds. Some researchers believe in the necessity of having a "teacher" to train the classifier [20]. This inspired the use of co-tracking [50], ensemble tracking [44, 57], disabling updates during occlusions, or label verification schemes [24] to break the self-learning loop using auxiliary classifiers.

Ensemble tracking framework provides effective frameworks to tackle one or more of these challenges. In such frameworks, the self-learning loop is broken, and the labeling process is performed by leveraging a group of classifiers with different views [19, 21, 44], subsets of training data [39] or memories [38, 57]. The main challenge in ensemble methods is how to decorrelate ensemble members and diversify learned models [21]. Combining the outputs of multiple classifiers is only useful if they disagree on some inputs [27], however, individual learners with similar training data are usually highly correlated [60] (Fig. 1).

Contributions: We propose a diversified ensemble discriminative tracker (DEDT) for real-time object tracking. We construct an ensemble using various subsamples of the tracking data and maintain the ensemble throughout the tracking. This is possible by devising methods to update the ensemble to reflect target changes while keeping its diversity to achieve good accuracy and generalization. In addition, breaking the self-learning loop to avoid the potential drift of the ensemble is applied in a co-tracking framework with an auxiliary classifier. However, to avoid unnecessary computation and boost the accuracy of the tracker, an effective data exchange scheme is required. We demonstrate that learning ensembles with randomized subsets of training data along with artificial data with diverse labels in a co-tracking framework achieve superior accuracy. This paper offers the following contributions:

- We propose a novel ensemble update scheme that generates necessary samples to diversify the ensemble. Unlike the other model update schemes that ignore the correlation between classifiers of an ensemble, this method is designed to promote diversity.
- We propose a co-tracking framework that accommodates the short and long-term memory mixture, effective collaboration between classification modules, and optimized data exchange between modules by borrowing the concept of query-by-committee [49] from active learning literature.

In this view, our proposed method is distinguishable from CMT [38] that uses multiple-memory horizons for training the ensemble. It is also different from MUSTer [23] that use long-term memory to validate the results of short-memory tracker and TGPR [17], in which long-term memory regularizes the results of short-memory tracker. Furthermore, the proposed framework differs from the co-tracking elaborated in [50], as in that method two classifiers cast a weighted vote to label the target, and pass the samples they struggle with to the other one to learn. However, in our tracker, the ensemble passes the disputed samples to an auxiliary classifier which is trained on all of the data periodically, to provide the effect of long-term memory while being resistant to abrupt changes, outliers and label noise. The evaluation results of DEDT on OTB50 [55], OTB100 [56], and VOT2015 [26] datasets demonstrates competitive accuracy of DEDT compared to the state-of-the-art of tracking.

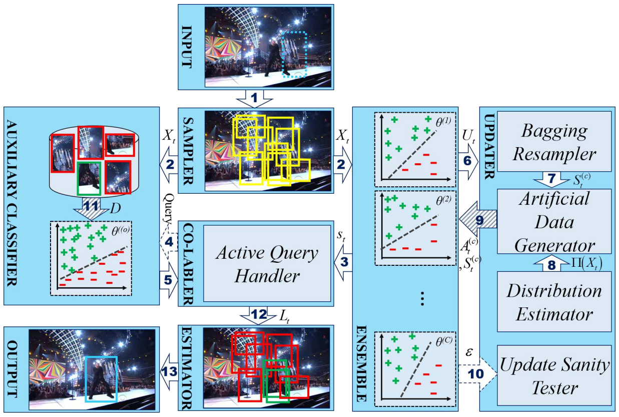


Figure 2. Schematic of the system. The proposed tracker, DEDT, labels the obtained sample using an homogeneous ensemble of the classifiers, the committee. The samples that the committee has highest disagreement upon (the uncertain samples) are queried from the auxiliary classifier, a different type of classifier. The location of the target is then estimated using the labeled target. Each member of the ensemble is then updated with a random subset of uncertain samples. By generating the diversity set (r.t. Sec 4.2), the ensemble is then diversified, yielding a more effective ensemble. For notion and procedure please r.t. Sec 4.1 and Alg. 1.

2. Prior Work

Ensemble tracking: Using a linear combination of several weak classifiers with different associated weights has been proposed in a seminal work by Avidan [2]. Following this study, constructing an ensemble by boosting [19], online boosting [31, 41], multi-class boosting [43] and multi-instance boosting [3, 58] led to the enhancement of the performance of the ensemble trackers. Despite its popularity, boosting demonstrates low endurance against label noise [47] and alternative techniques such as Bayesian ensemble weight adjustment [5] has been proposed to alleviate this shortcoming. Recently, ensemble learning based on CNNs gained popularity. Researchers make ensembles of CNNs that shares convolutional layers [40], different loss functions for each output of the feature map [54], and repeatedly subsampling different nodes and layers in fully connected layers on CNN to build an ensemble [21, 34]. Furthermore, it is proposed to exploit the power of ensembles such as feature adjustment in ensembles [16] and the addition of the ensemble's members [44, 57] over-time.

Ensemble diversity: Empirically, ensembles tend to yield better results when there is a significant diversity among the models [28]. Zhou [60] categorizes the diversity generation heuristics into *(i)* manipulation of data samples based on sampling approaches such as bagging and boosting (e.g. in [39]), *(ii)* manipulation of input features such as online boosting [19], random subspaces [45], random ferns [42] and random forests [44] or combining using different layers, neurons or interconnection layout of CNNs [21, 34], *(iii)* manipulation of learning parameter, and *(iv)* manipulation of the error representation. The literature also suggests a fifth category of manipulation of error function which encourages the diversity such as ensemble classifier selection based on Fisher linear discriminant [53].

Training data selection: A principled ordering of training examples can reduce the cost of labeling and lead to faster increases in the performance of the classifier [52], therefore we strive to use training examples based on their usefulness, and avoid using on all of them (including noisy ones and outliers) that may result in higher accuracy [14]. Starting from easiest examples (Curriculum learning) [6], pruning adversarial examples [35], excluding misclassified samples from next rounds of training [51], sorting samples by their training value [30] are some of the proposed approaches in the literature. However, the most common setting is active learning, in which the algorithm selects which training examples to label at each step for the highest gains in the performance. In this view, it may require to focus on learning the hardest examples first. For example, following the criteria of "highest uncertainty", an active learner select samples closest to the decision boundary to be labeled next. This concept can be useful in visual tracking, e.g. to measure the uncertainty caused by bags of samples [59].

Active learning for ensembles: Query-by-committee (QBC) [49] is one of the most popular ensemble-based active learning approaches, which constructs a committee of models representing competing hypotheses to label the samples. By defining a utility function on the ensemble (such as disagreement, entropy, or Q-statistics [60]), this method selects the most informative samples to be queried from the oracle (or any other collaborating classifier) in a form of the query optimization process [48]. Built upon randomized component learning algorithm, QBC involves Gibbs sampling, which requires adaptation to use deterministic classifiers. This was realized by resampling different subsets of data to construct an ensemble of deterministic base learners in query-by-bagging and query-by-boosting frameworks [1]. The set of hypotheses consistent with the data is called *version space* and by selecting the most informative samples to be labeled, QBC attempts to shrink the version space. However, only a committee of hypotheses that effectively samples the version space of all consistent hypotheses is productive for the sample selection [9]. To this end, it is crucial to promote the diversity of the ensemble [37]. In QBag and QBoost algorithms, all of the classifiers are trained on random subsets of the similar dataset, which degrade the diversity of the ensemble. Reducing the number of necessary labeled samples [29], unified sample learning and feature selection procedure [33] and reducing the sampling bias by controlling the variance [8] are some of the improvements that active learning provides for the discriminative trackers. Moreover, using diversity data to diversify the committee members [37] and promoting the classifiers that have unique misclassifications [53] are from few samples that active learning was employed to promote the diversity of the ensemble.

3. Tracking by Detection

By definition, a tracker tries to determine the state of the target $\mathbf{p}_t$ in frame F_t ($t \in \{1, \dots, T\}$) by finding the transformation $\mathbf{y}_t$ from its previous state $\mathbf{p}_{t-1}$. In tracking-by-detection formulation, the tracker employs a classifier θ_t to separate the target from the background. It is realized by evaluating possible candidates from the expected target state-space $\mathcal{Y}_t$. The candidate whose appearance resembles the target the most, is usually considered as the new target state. Finally, the classifier is updated to reflect the recent information.

To this end, first several samples $\mathbf{x}_t^{\mathbf{p}_{t-1}\circ\mathbf{y}_t^j} \in \mathcal{X}_t$ are obtained by a transformation $\mathbf{y}_t^j \in \mathcal{Y}_t$ from the previous target state, $\mathbf{p}_{t-1} \circ \mathbf{y}_t^j$. Sample $j \in \{1, \dots, n\}$ indicates the location $\mathbf{p}_{t-1} \circ \mathbf{y}_t^j$ in the frame F_t, where the image patch $\mathbf{x}_t^{\mathbf{p}_{t-1}\circ\mathbf{y}_t^j}$ is contained. Then, each sample is evaluated by the classifier scoring function $h : \mathcal{X}_t \to \mathbb{R}$ to calculate the score $s_t^j = h(\mathbf{x}_t^{\mathbf{p}_{t-1}\circ\mathbf{y}_t^j} | \theta_t)$. This score is utilized to obtain a label ℓ_t^j for the sample, typically by thresholding its score,

$$\ell_t^j = \begin{cases} +1 & , s_t^j > \tau_u \\ -1 & , s_t^j < \tau_l \\ 0 & , \text{otherwise} \end{cases} \tag{1}$$

where τ_l and τ_u serves as lower and upper thresholds respectively. Finally, the target location $\mathbf{y}_t$ is obtained by comparing the samples' classification scores. To obtain the exact target state, the sample with highest score is selected as the new target, $\mathbf{y}_t = \mathbf{y}_t^{j^*}$ s.t. $j^* = \underset{i}{\text{argmax}}\ (s_t^j)$. A subset of the samples and their labels are used to re-train the classifier's model $\theta_{t+1} = u(\theta_t, \mathcal{D}_{\xi(t)})$. Here, $\mathcal{D}_t = \{\langle \mathcal{X}_t, \mathcal{L}_t \rangle\}$ is the set of samples $\mathcal{X}_t$ and their labels $\mathcal{L}_t$, $u(.)$ is the model update function, and the $\xi(t)$ defines the subset of the samples that the tracker considers for model update.

An ensemble discriminative tracker employs a set of classifiers instead of one. These classifiers, hereafter called *committee*, are represented by $\mathcal{C}_t = \{\theta_t^{(1)}, \dots, \theta_t^{(C)}\}$, and are typically homogeneous and independent (e.g., [31, 44]). Popular ensemble trackers utilize the majority voting of the committee as their utility function,

$$s_t^j = \sum_{c=1}^{C} \text{sign}\big(h(\mathbf{x}_t^{\mathbf{p}_{t-1}\circ\mathbf{y}_t^j}|\theta_t^{(c)})\big). \tag{2}$$

Then eq(1) is used to label the samples.

The model of each classifier is updated independently, $\theta_{t+1}^{(c)} = u(\theta_t^{(c)}, \mathcal{D}_{\xi(t)})$ meaning that all of the committee members are trained with a similar set of samples and a common label for them.

4. Diverse Ensemble Discriminative Tracker

We propose a diverse ensemble tracker composed of a highly-adaptive and diverse ensemble of classifiers $\mathcal{C}$ (the *committee*), a long-term memory object detector (that serves as the *auxiliary* classifier), and an information exchange channel governed by active learning. This allows for effective diversification of the ensemble, improving the generalization of the tracker and accelerating its convergence to the ever-changing distribution of target appearance. We leveraged the complementary nature and long-term memory of the auxiliary tracker to facilitate effective model update.

One way to diversify the ensemble is to increase the number of examples they disagree upon [27]. Using bagging and boosting to construct an ensemble out of a fix sample set, ignores this critical need for diversity as all of the data are randomly sampled from a shared data distribution. However, for each committee member, there exists a set of samples that distinguish them from other committee members. One way to obtain such samples is to generate some training samples artificially to differ maximally from the current ensemble [36].

The diversified ensemble covers larger areas of the version space (i.e. the space of consistent hypotheses with the samples from current frame), however, this radical update of the ensemble may render the classifier susceptible to drastic target appearance changes, abrupt motion, and occlusions. In this case, given the non-stationary nature of the target distribution[1], the classifier should adapt itself rapidly with the target changes, yet it should keep a memory of the target to re-identify if the target goes out-of-view or got occluded (as known as stability-plasticity dilemma [20]). In addition, there are samples for which the ensemble is not unanimous and an external teacher maybe deemed required.

To amend these shortcomings, an auxiliary classifier is utilized to label the samples which the ensemble dispute upon (co-tracking). This classifier is batch-updated with all of the samples less frequently than the ensemble, realizing the longer memory for the tracker. Active query optimization is employed to query the label of the most informative samples from the auxiliary classifier, which is observed to effectively balance the stability-plasticity equilibrium of the tracker as well. Figure 2 presents the schematic of the proposed tracker.

4.1. Formalization

In this approach, if the committee comes to a solid vote about a sample, then the sample is labeled accordingly. However, when the committee disagrees about a sample, its label is queried from the auxiliary classifier $\theta_t^{(o)}$:

$$\ell_t^j = \begin{cases} +1 & , s_t^j > \tau_u \\ -1 & , s_t^j < \tau_l \\ \text{sign}\big(h(\mathbf{x}_t^{\mathbf{p}_{t-1}\circ\mathbf{y}_t^j}|\theta_t^{(o)})\big) & \text{, otherwise} \end{cases} \tag{3}$$

in which s_t^j is derived from eq(2). The uncertain samples list is defined as $\mathcal{U}_t = \{\mathbf{x}_t^{\mathbf{p}_{t-1}\circ\mathbf{y}_t^j}|\tau_l < s_t^j < \tau_u\}$.

The committee members are then updates using our proposed mechanism $f(.)$ using the uncertain samples $\mathcal{U}_t$,

$$\theta_{t+1}^{(c)} = f(\theta_t^{(1..c)}, \mathcal{U}_t, \mathcal{D}_t) \tag{4}$$

Finally, to maintain a long-term memory and slower update rate for the auxiliary classifier, it is updated every Δ frames with all of the samples from $t - \Delta$ to t.

$$\theta_{t+1}^{(o)} = \begin{cases} u(\theta_t^{(o)}, \mathcal{D}_{t-\Delta..t}) & \text{, if } t \neq k\Delta + 1 \\ \theta_t^{(o)} & \text{, if } t = k\Delta + 1 \end{cases} \tag{5}$$

Algorithm (1) summarizes the proposed tracker.

4.2. Diversifying Ensemble Update

The model updates to construct a diverse ensemble either replace the weakest or oldest classifier of the ensemble [2, 19] or creates a new ensemble in each iteration [37]. While the former lacks flexibility to adjust to the rate of target change, the latter involves a high level of computation redundancy. To alleviate these shortcomings, we create an ensemble for the first frame, update them in each frame to keep a memory of the target, and diversify them to improve the effectiveness of ensemble. The diversifying update procedure is as follows:

1. The members ensemble $\mathcal{C}_t$ is updated with a random subsets (of size m) of the uncertain data $\mathcal{U}_t$, that make them more adept in handling such samples, and generate a temporary ensemble $\mathcal{C}'_t$. Note that for certain

[1] The non-stationarity means that the appearance of an object may change so significantly that a negative sample in the current frame looks more similar to a positive example in the previous frames [4].

```
input  : Committee models $\theta_t^{(c)}$, Auxiliary model $\theta^{(o)}$
input  : Target position in previous frame $\mathbf{p}_{t-1}$
output: Target position in current frame $\mathbf{p}_t$
for $j \leftarrow 1$ to $n$ do
    Sample a transformation $\mathbf{y}_t^j \sim \mathcal{N}(\mathbf{p}_t, \Sigma_{search})$
    Calculate committee score $s_t^j$ (eq(2))
    if $\tau_l < s_t^j < \tau_u$ then sample label is uncertain
        $\ell_t^j = \mathrm{sign}\big(h(\mathbf{x}_t^{\mathbf{p}_{t-1} \circ \mathbf{y}_t^j} | \theta^{(o)})\big)$
        $\mathcal{U}_t \leftarrow \mathcal{U}_t \cup \{\langle \mathbf{x}^{\mathbf{p}_{t-1} \circ \mathbf{y}_t^j}, \ell_t^j \rangle\}$
    else
        $\ell_t^j = \mathrm{sign}(s_t^j)$
    $\mathcal{D} \leftarrow \mathcal{D} \cup \{\langle \mathbf{x}^{\mathbf{p}_{t-1} \circ \mathbf{y}_t^j}, \ell_t^j \rangle\}$
for $c \leftarrow 1$ to $C$ do
    Uniformly resample $m$ data $\mathcal{S}_t^{(c)}$ from $\mathcal{U}_t$
    $\theta_t'^{(c)} \leftarrow u(\theta_t^{(c)} | \mathcal{S}_t^{(c)})$
Calculate the prediction error of $\mathcal{C}_t$, $\epsilon(\mathcal{C}_t) = \frac{|\mathcal{U}_t|}{|\mathcal{D}_t|}$
Calculate empirical distribution of samples, $\Pi(\mathcal{X}_t)$
for $c \leftarrow 1$ to $C$ do
    do
        Draw $m'$ samples $\mathcal{A}_t^{(c)}$ from $\Pi(\mathcal{X}_t)$
        Calculate class membership probability $\hat{\ell}(\mathcal{C}_t')$
        Set the labels of samples $\propto \frac{1}{\hat{\ell}(\mathcal{C}_t')}$
        $\theta_t''^{(c)} \leftarrow u(\theta_t'^{(c)} | \mathcal{A}_t^{(c)})$
        Calculate new prediction error $\epsilon(\mathcal{C}_t'')$ (eq(6))
    while $\epsilon(\mathcal{C}_t'') \geq \epsilon(\mathcal{C}_t)$
    $\theta_t'^{(c)} \leftarrow \theta_t''^{(c)}$
All diversity sets are applied, $\mathcal{C}_{t+1} \leftarrow \mathcal{C}_t'$
if $\mathrm{mod}(t, \Delta) = 0$ then
    $\theta_{t+1}^{(o)} \leftarrow u(\theta_t^{(o)}, \mathcal{D}_{t-\Delta..t})$
Target transformation $\mathbf{y}_t = \mathbf{y}_t^{j^*}$ s.t. $j^* = \underset{i}{\mathrm{argmax}}(s_t^j)$
Calculate target position $\mathbf{p}_t = \mathbf{p}_{t-1} \circ \hat{\mathbf{y}}_t$
```

Algorithm 1: Diverse Ensemble Discriminative Tracker

samples (those not in $\mathcal{U}_t$), the committee is unanimous about the label and adding them to the training set of the committee classifiers is redundant [39].

2. The label prediction of the original ensemble $\mathcal{C}_t$ is then calculated on $\mathcal{D}_t$ w.r.t. the labels given by the whole tracker (composed of the ensemble and the auxiliary classifier), and prediction error $\epsilon(\mathcal{C}_t)$ is obtained.
3. The empirical distribution of training data, $\Pi(\mathcal{X}_t)$, is calculated to govern the creation of the artificial data.
4. In an iterative process for each of the committee members, m' samples are drawn from a $\Pi(\mathcal{X}_t)$, assuming attribute independence. Given a sample, the class membership probabilities of the temporary ensemble $\hat{\ell}(\mathcal{C}_t')$ that is the probability of selecting a label by the temporary ensemble on $\mathcal{D}_t$, is then calculated. Labels are then sampled from this distribution, such that the probability of selecting a label is inversely proportional to the temporary ensemble prediction. This set of artificial samples and their diverse labels are called the *diversity set* of committee member c, $\mathcal{A}_t^{(c)}$.
5. The classifier c of temporary ensemble is updated with $\mathcal{A}_t^{(c)}$, to obtain the diverse ensemble $\mathcal{C}_t'' = \{\theta_t''^{(c)}\}$ and calculate its prediction error $\epsilon(\mathcal{C}_t'')$. If this update increases the total prediction error of the ensemble ($\epsilon(\mathcal{C}_t'') > \epsilon(\mathcal{C}_t)$), then the artificial data is rejected and new data $\mathcal{A}_t^{(c)}$ should be generated,

$$\epsilon(\mathcal{C}_t'') = \sum_{c=1}^{C} \sum_{j=1}^{n} \mathbb{1}\big(\ell_t^j \neq h(\mathbf{x}_t^{\mathbf{p}_{t-1} \circ \mathbf{y}_t^j} | \theta_t''^{(c)})\big). \qquad (6)$$

where $\mathbb{1}(.)$ denotes the step function that returns 1 iff its argument is true/positive and 0 otherwise.

This procedure creates samples for each member of the committee that distinguish them from other members of the ensemble using a contradictory label (therefore improving the ensemble diversity [37]), but only accepts them when using such artificial data improves the ensemble accuracy.

4.3. Implementation Details

There are several parameters in the system such as the number of committee members (C), parameters of sampling step (number of samples n, effective search radius Σ_{search}), and the holding time of auxiliary classifier (Δ). Larger values of m results in temporary committee with a higher degree of overlap, thus less diverse, whereas smaller values of m tend to miss the latest changes of the quick-changing target. A Larger number of artificial samples m' result in more diversity in the ensemble, but reduce the chance of successful update (i.e. lowering the prediction error of the ensemble). These parameters were tuned using a simulated annealing optimization on a cross-validation set.

In our implementation, we used kd-tree-based KNN classifiers with HOG [10] feature for the ensemble and reused the calculations with a caching mechanism to accelerate classification. For the empirical distribution of the data, a Gaussian distribution is determined by estimating the mean and standard variation of the given training set (i.e. HOG of $\mathcal{X}_t$). In addition, to localize the target, the samples with the highest sum of confidence scores is selected as the next target position. The auxiliary classifier is a a part-based detector [15]. The features, part-base detector dictionary, and the parameters of committee members (k of KNNs), thresholds τ_l, τ_u, and the rest of above-mentioned parameters (Except for C that have been adjusted to control the speed of the tracker, here) have been adjusted using cross-validation. With $C = 15, k = 23, n = 1000, m = 80, m' = 250, \tau_u = 0.54$ and $\tau_l = -0.41$

Table 1. Quantitative evaluation of trackers under different visual tracking challenges of OTB50 [55] using AUC of success plot and their overall precision. The first, second and third best methods are shown in color. More data are available on http://ishiilab.jp/member/meshgi-k/dedt.html.

Attribute	TLD	STRK	TGPR	MEEM	MSTR	STPL	CMT	SRDCF	CCOT	Ours
IV	0.48	0.53	0.54	0.62	0.73	0.68	0.73	0.70	0.75	0.75
DEF	0.38	0.51	0.61	0.62	0.69	0.70	0.69	0.67	0.69	0.69
OCC	0.46	0.50	0.51	0.61	0.69	0.69	0.69	0.70	0.76	0.72
SV	0.49	0.51	0.50	0.58	0.71	0.68	0.72	0.71	0.76	0.74
IPR	0.50	0.54	0.56	0.58	0.69	0.69	0.74	0.70	0.72	0.73
OPR	0.48	0.53	0.54	0.62	0.70	0.67	0.73	0.69	0.74	0.74
OV	0.54	0.52	0.44	0.68	0.73	0.62	0.71	0.66	0.79	0.76
LR	0.36	0.33	0.38	0.43	0.50	0.47	0.55	0.58	0.70	0.58
BC	0.39	0.52	0.57	0.67	0.72	0.67	0.69	0.70	0.70	0.73
FM	0.45	0.52	0.46	0.65	0.65	0.56	0.70	0.63	0.72	0.74
MB	0.41	0.47	0.44	0.63	0.65	0.61	0.65	0.69	0.72	0.72
Avg. Succ	0.49	0.55	0.56	0.62	0.72	0.69	0.72	0.70	0.75	0.74
Avg. Prec	0.60	0.66	0.68	0.74	0.82	0.76	0.83	0.78	0.84	0.84
$IoU > 0.5$	0.59	0.64	0.66	0.75	0.86	0.82	0.83	0.83	0.90	0.89
Avg FPS	21.2	11.3	3.7	14.2	8.3	48.1	21.9	4.3	0.2	21.9

DEDT achieved the speed of 21.97 fps on a Pentium IV PC @ 3.5 GHz and a Matlab/C++ implementation on a CPU. Source code can be found at http://ishiilab.jp/member/meshgi-k/dedt.html.

5. Experiments

For our component analysis, we used the OTB50 [55] dataset and its subsets with a distinguishing attribute to evaluate the tracker performance. 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. Additionally, to compare our proposed algorithm against the state-of-the-art we employed OTB100 [56] and VOT2015 [26] datasets.

For this comparison, we have used success and precision plots, where their area under curve provides a robust metric for comparing tracker performances [55]. The precision plot compares the number of frames that a tracker has certain pixels of displacement, whereas the overall performance of the tracker is measured by the area under the surface of its success plot, where the success of tracker in time t is determined when the normalized overlap of the tracker target estimation $\mathbf{p}_t$ with the ground truth $\mathbf{p}_t^*$ (also known as IoU) exceeds a threshold τ_{ov}. Success plot, graphs the success of the tracker against different values of the threshold τ_{ov} and its AUC is calculated as

$$AUC = \frac{1}{T} \int_0^1 \sum_{t=1}^{T} \mathbb{1} \left(\frac{|\mathbf{p}_t \cap \mathbf{p}_t^*|}{|\mathbf{p}_t^* \cup \mathbf{p}_t^*|} > \tau_{ov} \right) d_{\tau_{ov}}, \quad (7)$$

where T is the length of sequence, $|.|$ denotes the area of the region and $\cap$ and $\cup$ stands for intersection and union of the regions respectively. We also compare all the trackers by the success rate at the conventional thresholds of 0.50 ($IoU > 0.50$) [55]. The result of the algorithms are reported as the average of five independent runs.

5.1. Effect of Diversification

To demonstrate the effectiveness of the proposed diversification method we compare the DEDT tracker with two different versions of the tracker. In the firs version, DEDT-bag, the ensemble classifiers are only updated with uniform-picked subsets of the uncertain data (step 1 in section 4.2). In the other version, DEDT-art, the committee members are only updated with artificially generated data (steps 2-5 in the same section). All three algorithms use $m + m'$ samples to update their classifiers. In addition to the overall performance of the tracker, we measure the diversity of the ensemble using the Q-statistics as elaborated in [28]. For statistically independent classifiers i and j, the expectation of $Q_{i,k} = 0$. Classifiers that tend to classify the same sample correctly will have positive values of Q, and those which commit errors on different samples have negative Q ($-1 \leq Q_{i,k} \leq +1$). For the ensemble of C classifiers, the averaged Q statistics over all pairs of classifiers is

$$Q_{av} = \frac{2}{C(C-1)} \sum_{i=1}^{C-1} \sum_{j=i+1}^{C} Q_{i,j} \text{ , s.t.} \quad (8)$$

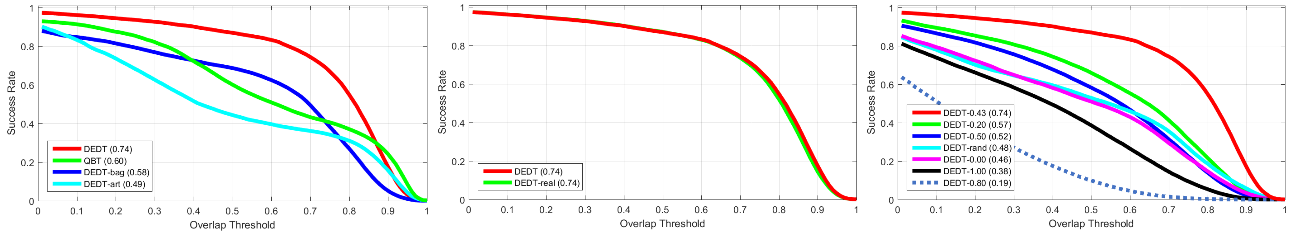
$$Q_{i,j} = \frac{N^{ff}N^{bb} - N^{fb}N^{bf}}{N^{ff}N^{bb} + N^{fb}N^{bf}} \quad (9)$$

where N^{fb} is the number of cases that classifier i classified the sample as foreground, while classifier j detected it as background, etc.

Figure 3(a) illustrates the effectiveness of the diversification mechanism in contrast with merely generating data or update the classifiers with uninformed subsamples of the data. From the experiment results, AUC(DEDT-art) $<$ AUC(DEDT-bag) $<$ AUC(DEDT) and $0 < Q_{av}$(DEDT) $< Q_{av}$(DEDT-art) $< Q_{av}$(DEDT-bag) it can be concluded that all of steps of proposed diversification are crucial to maintain an accurate and diverse ensemble. Q_{av}(DEDT-art) $< Q_{av}$(DEDT-bag) shows that the diversity of DEDT-art is better than random diversity obtain by DEDT-bag, however, AUC(DEDT-art) $<$ AUC(DEDT-bag) reveals that merely using artificial data without the samples gathered by the tracker, does not provide enough data for an accurate model update.

5.2. Effect of using Artificial Data

In the first look, using synthesized data to train the ensemble that will keep track of a real object may not seem proper. In this experiment, we look for the closest patch



(a) The diversification procedure (b) Using artificial data versus real data (c) The "activeness", i.e. the effect of thresholds

Figure 3. The effect of different components of the proposed algorithm on the overall tracking results on OTB50 [55].

of the real image (frame t of the video) to the synthesized sample, and use it as the diversity data. To this end, in each frame, a dense sampling over the frame is performed, the HOG of these image patches are calculated, and the closest match to the generated sample (using Euclidean distance) is selected. The obtained tracker is referred as DEDT-real, and its performance is compared to the original DEDT.

As Figure 3(b) shows, the use of this computationally-expensive version of the algorithm does not improve the performance significantly. However, it should be noted that generating adversarial samples of the ensemble [18] for as the diversity data of individual committee members is expected to increase the accuracy of the ensemble, yet it is out of the scope of the current research and may be considered as a future direction for this research.

5.3. Effect of "Activeness"

Labeling thresholds (τ_l and τ_u) control the "activeness" of the data exchange between the committee and the auxiliary classifier, therefore allowing the ensemble to get more/less assistance for its collaborator. In our implementation, these two values are treated independently, but for the sake of argument assume that $\tau_l = -\delta$ and $\tau_u = +\delta$ ($\delta \in [0, 1]$). Figure 3(c) compares the effects of different values of the δ, and also a "random" data exchange scheme in which the labeler gets the label of the sample from the ensemble or auxiliary classifier with the same chance. To interpret this figure it is prudent to note that $\delta \to 0$ forces the ensemble to label all of the samples without any assistance from the auxiliary classifier. By increasing δ the ensemble starts to query highly disputed samples from the auxiliary classifier, which is desired by design. If this value increases excessively, the ensemble queries even slightly uncertain samples from the auxiliary classifier, rendering the tracker prone to the labeling noise of this classifier. In addition, the tracker loses its ability to update rapidly in the case of an abrupt change in the target's appearance or location, leading to a degraded performance of the tracker. In the extreme case of $\delta \to 1$ the tracker reduces to a single object detector modeled by the auxiliary classifier.

The information exchange in one way is in the form of querying the most informative labels from the auxiliary classifier, and on the other way is re-training it with the labeled samples by the committee (for certain samples). We observed that this exchange is essential to construct a robust and accurate tracker. Moreover, such data exchange not only breaks the self-learning loop but also manages the plasticity-stability equilibrium of the tracker. In this view, lower values of δ correspond to a more-flexible tracker, while higher values make it more conservative.

5.4. Comparison with State-of-the-Art

To establish a fair comparison with the state-of-the-art, some of the most successful popular discriminative trackers (according to a recent large benchmark [26, 55, 56] and the recent literature) are selected: TLD [24], STRK [22], TGPR [17], MEEM [57], MUSTer [23], STAPLE [7], CMT [38], SRDCF [12], and CCOT [13].

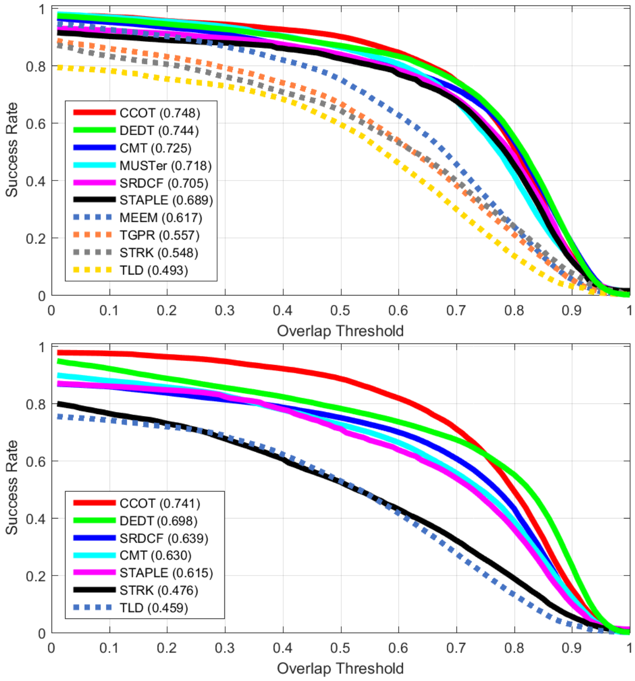


Figure 4. Quantitative performance comparison of the proposed tracker, DEDT, with the state-of-the-art trackers using success plot on OTB50 [55] (**top**) and OTB100 [56] (**bottom**).

Table 2. Quantitative evaluation of trackers under different visual tracking challenges of OTB100 [56].

	TLD	STRK	MEEM	STPL	CMT	SRDCF	CCOT	Ours
Avg. Succ	0.46	0.48	0.65	0.62	0.63	0.64	0.74	0.69
Avg. Prec	0.58	0.59	0.62	0.73	0.74	0.71	0.85	0.81
$IoU > 0.5$	0.52	0.52	0.62	0.71	0.72	0.75	0.88	0.78

Table 3. Evaluation on VOT2015 [26] by the means of robustness and accuracy.

	STRK	TGPR	MEEM	MSTR	STPL	CMT	SRDCF	CCOT	Ours
Accuracy	0.47	0.48	0.50	0.52	0.53	0.49	0.56	0.54	0.58
Robustness	1.26	2.31	1.85	2.00	1.35	1.81	1.24	0.82	1.36

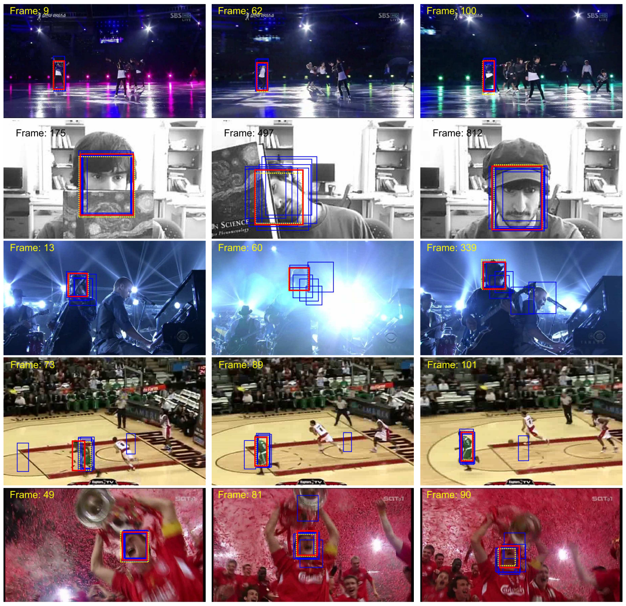


Figure 5. Sample tracking results of evaluated algorithms on several challenging video sequences, in these sequences the red box depicts the DEDT against other trackers (blue). The ground truth is illustrated with yellow dashed box. From top to bottom the sequences are *Skating1*, *FaceOcc2*, *Shaking*, *Basketball*, and *Soccer* with drastic illumination changes, scaling and out-of-plane rotations, background clutter, noise and severe occlusions.

Figure 4 presents the success and precision plots of DEDT along with other state-of-the-art trackers for all sequences. It is shown in this plot that DEDT usually keeps the localization error under 10 pixels. Table 1 presents the area under the curve of the success plot (eq(7)) for all the sequences and their subcategories, each focusing on a certain challenge of the visual tracking. As shown, DEDT has the competitive precision compared to CCOT which employs state-of-the-art multi-resolution deep feature maps, and performs better than the rest of the other investigated trackers on this dataset. The performance of DEDT is comparable with CCOT in the case of illumination variation, deformation, out-of-view, out-of-plane rotation and motion blur, while it has superior performance in handling background clutter. This indicates the effectiveness of the target vs. background detection and flexibility for accommodating rapid target changes. While the former can be attributed to effective ensemble tracking, the latter is known to be the effect of combining long and short-term memory. It is observed in the run-time that for handling extreme rotations, the ensemble heavily relies on the auxiliary tracker, which although brings the superior performance in the category, a better representation of the ensemble model may reduce the reliance of the tracker to the auxiliary tracker.The proposed algorithm shows a sub-optimal performance in low-resolution scenario compared to DCF-based trackers (SRDCF, and CCOT), and although it does not provide a high-quality localization for smaller/low-resolution targets, it is able to keep tracking them. This finding highlights the importance of further research on the ensemble-based DCF trackers. Our method also achieved the best accuracy (0.58) on VOT2015 by outperforming SRDCF, yet the highest robustness (0.82) belongs to CCOT (Table 3). Finally, a qualitative comparison of DEDT versus other trackers is presented in Figure 5.

6. Conclusion

In this study, we proposed diverse ensemble discriminative tracker (DEDT) that maintains a diverse committee of classifiers to the label of the samples and queries the most disputed labels –which are the most informative ones– from a long-term memory auxiliary classifier. By generating artificial data with diverse labels, we intended to diversify the ensemble of classifiers, efficiently covering the version space, increasing the generalization of the ensemble, and as a result, improve the accuracy. In addition, by using the query-by-committee concept in labeling and updating stages of the tracker, the label noise problem is decreased. By using the diverse committee, in turn, the problem of equal weights for the samples are addressed, and a good approximation of the target location is acquired even without dense sampling. The active learning scheme manages the balance between short-term and long-term memory by recalling the label from long-term memory when the short-term memory is not clear about the label (due to forgetting the label or insufficient data). This also reduces the dependence of the tracker on a single classifier (i.e., auxiliary classifier), yet breaking the self-learning loop to avoid accumulative model drift. The results of the experiment on OTB50, OTB100, and VOT2015 benchmarks demonstrate the competitive tracking performance of the proposed tracker compared with the state-of-the-art.

Acknowledgment

This study is partly supported by the Japan NEDO and the "Post-K application development for exploratory challenges" project of the Japan MEXT.

References

[1] N. Abe and H. Mamitsuka. Query learning strategies using boosting and bagging. In *ICML'98*, 1998. 3

[2] S. Avidan. Ensemble tracking. *PAMI*, 29, 2007. 2, 4

[3] B. Babenko, M.-H. Yang, and S. Belongie. Visual tracking with online multiple instance learning. In *CVPR'09*, 2009. 1, 2

[4] Q. Bai, Z. Wu, S. Sclaroff, M. Betke, and C. Monnier. Randomized ensemble tracking. In *ICCV'13*, 2013. 4

[5] Y. Bai and M. Tang. Robust tracking via weakly supervised ranking svm. In *CVPR'12*, 2012. 1, 2

[6] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *ICML'09*, 2009. 3

[7] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. Torr. Staple: Complementary learners for real-time tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1401–1409, 2016. 7

[8] A. Beygelzimer, S. Dasgupta, and J. Langford. Importance weighted active learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 49–56. ACM, 2009. 3

[9] D. A. Cohn, Z. Ghahramani, and M. I. Jordan. Active learning with statistical models. *Journal of artificial intelligence research*, 4(1):129–145, 1996. 3

[10] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005. 5

[11] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg. Eco: Efficient convolution operators for tracking. *arXiv preprint arXiv:1611.09224*, 2016. 1

[12] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg. Learning spatially regularized correlation filters for visual tracking. In *ICCV'15*, pages 4310–4318, 2015. 7

[13] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg. Beyond correlation filters: Learning continuous convolution operators for visual tracking. In *ECCV'16*. 1, 7

[14] F. De la Torre and M. J. Black. Robust principal component analysis for computer vision. In *ICCV'01*, 2001. 3

[15] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *PAMI*, 32, 2010. 5

[16] J. Gall, A. Yao, N. Razavi, L. Van Gool, and V. Lempitsky. Hough forests for object detection, tracking, and action recognition. *PAMI*, 2011. 3

[17] J. Gao, H. Ling, W. Hu, and J. Xing. Transfer learning based visual tracking with gaussian processes regression. In *ECCV'14*, pages 188–203. Springer, 2014. 2, 7

[18] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 7

[19] H. Grabner, M. Grabner, and H. Bischof. Real-time tracking via on-line boosting. In *BMVC'06*, volume 1, page 6, 2006. 1, 2, 3, 4

[20] H. Grabner, C. Leistner, and H. Bischof. Semi-supervised on-line boosting for robust tracking. In *ECCV'08*. 2008. 1, 2, 4

[21] B. Han, J. Sim, and H. Adam. Branchout: Regularization for online ensemble tracking with convolutional neural networks. In *Proceedings of IEEE International Conference on Computer Vision*, pages 2217–2224, 2017. 2, 3

[22] S. Hare, A. Saffari, and P. H. Torr. Struck: Structured output tracking with kernels. In *ICCV'11*, 2011. 1, 7

[23] 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*. 2, 7

[24] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *PAMI*, 34(7):1409–1422, 2012. 2, 7

[25] H. Kiani Galoogahi, A. Fagg, and S. Lucey. Learning background-aware correlation filters for visual tracking. *arXiv*, 2017. 1

[26] M. Kristan, J. Matas, A. Leonardis, and M. Felsberg. The visual object tracking vot2015 challenge results. In *ICCVw'15*. 2, 6, 7, 8

[27] A. Krogh, J. Vedelsby, et al. Neural network ensembles, cross validation, and active learning. *Advances in neural information processing systems*, 7:231–238, 1995. 2, 4

[28] L. I. Kuncheva and C. J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine learning*, 51(2):181–207, 2003. 3, 6

[29] C. H. Lampert and J. Peters. Active structured learning for high-speed object detection. In *PR*, pages 221–231. Springer, 2009. 3

[30] A. Lapedriza, H. Pirsiavash, Z. Bylinskii, and A. Torralba. Are all training examples equally valuable? *arXiv*, 2013. 3

[31] C. Leistner, A. Saffari, and H. Bischof. Miforests: Multiple-instance learning with randomized trees. In *ECCV'10*, 2010. 2, 4

[32] A. Li, M. Lin, Y. Wu, M.-H. Yang, and S. Yan. Nus-pro: A new visual tracking challenge. *PAMI*, 2016. 1

[33] C. Li, X. Wang, W. Dong, J. Yan, Q. Liu, and H. Zha. Active sample learning and feature selection: A unified approach. *arXiv preprint arXiv:1503.01239*, 2015. 3

[34] H. Li, Y. Li, and F. Porikli. Convolutional neural net bagging for online visual tracking. *Computer Vision and Image Understanding*, 153:120–129, 2016. 3

[35] J. Lu, T. Issaranon, and D. Forsyth. Safetynet: Detecting and rejecting adversarial examples robustly. *arXiv*, 2017. 3

[36] P. Melville and R. J. Mooney. Constructing diverse classifier ensembles using artificial training examples. In *IJCAI*, volume 3, pages 505–510, 2003. 4

[37] P. Melville and R. J. Mooney. Diverse ensembles for active learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 74. ACM, 2004. 3, 4, 5

[38] K. Meshgi, S. Oba, and S. Ishii. Active discriminative tracking using collective memory. In *MVA'17*. 2, 7

[39] K. Meshgi, S. Oba, and S. Ishii. Robust discriminative tracking via query-by-committee. In *AVSS'16*, 2016. 2, 3, 5

[40] H. Nam, M. Baek, and B. Han. Modeling and propagating cnns in a tree structure for visual tracking. *arXiv preprint arXiv:1608.07242*, 2016. 2

[41] N. C. Oza. Online bagging and boosting. In *SMC'05*, 2005. 2

[42] C. Rao, C. Yao, X. Bai, W. Qiu, and W. Liu. Online random ferns for robust visual tracking. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 1447–1450. IEEE, 2012. 3

[43] A. Saffari, C. Leistner, M. Godec, and H. Bischof. Robust multi-view boosting with priors. In *ECCV'10*. 2010. 2

[44] A. Saffari, C. Leistner, J. Santner, M. Godec, and H. Bischof. On-line random forests. In *ICCVw'09*. 2, 3, 4

[45] A. Salaheldin, S. Maher, and M. Helw. Robust real-time tracking with diverse ensembles and random projections. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 112–120, 2013. 3

[46] S. Salti, A. Cavallaro, and L. Di Stefano. Adaptive appearance modeling for video tracking: Survey and evaluation. *IEEE TIP*, 2012. 1

[47] J. Santner, C. Leistner, A. Saffari, T. Pock, and H. Bischof. Prost: Parallel robust online simple tracking. In *CVPR'10*. 2

[48] B. Settles. *Active learning*. Morgan & Claypool Publishers, 2012. 3

[49] H. S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *COLT'92*, pages 287–294. ACM, 1992. 2, 3

[50] F. Tang, S. Brennan, Q. Zhao, and H. Tao. Co-tracking using semi-supervised support vector machines. In *ICCV'07*. 2

[51] A. Vezhnevets and O. Barinova. Avoiding boosting overfitting by removing confusing samples. In *ECML'07*. 3

[52] S. Vijayanarasimhan and K. Grauman. Cost-sensitive active visual category learning. *IJCV*, 2011. 3

[53] I. Visentini, J. Kittler, and G. L. Foresti. Diversity-based classifier selection for adaptive object tracking. In *MCS*, pages 438–447. Springer, 2009. 3

[54] L. Wang, W. Ouyang, X. Wang, and H. Lu. Stct: Sequentially training convolutional networks for visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1373–1381, 2016. 2

[55] Y. Wu, J. Lim, and M.-H. Yang. Online object tracking: A benchmark. In *CVPR'13*, pages 2411–2418. IEEE, 2013. 1, 2, 6, 7

[56] Y. Wu, J. Lim, and M.-H. Yang. Object tracking benchmark. *PAMI*, 37(9):1834–1848, 2015. 2, 6, 7, 8

[57] J. Zhang, S. Ma, and S. Sclaroff. Meem: Robust tracking via multiple experts using entropy minimization. In *ECCV'14*. 2, 3, 7

[58] K. Zhang and H. Song. Real-time visual tracking via online weighted multiple instance learning. *PR*, 2013. 2

[59] K. Zhang, L. Zhang, M.-H. Yang, and Q. Hu. Robust object tracking via active feature selection. *IEEE CSVT*, 23(11):1957–1967, 2013. 3

[60] Z.-H. Zhou. *Ensemble methods: foundations and algorithms*. CRC press, 2012. 2, 3