Pattern Recognition 47 (2014) 1395–1410

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Visual tracking via weakly supervised learning from multiple imperfect oracles

Bineng Zhong [a,f], Hongxun Yao [b], Sheng Chen [c], Rongrong Ji [d], Tat-Jun Chin [e], Hanzi Wang [f,*]

[a] Department of Computer Science and Engineering, Huaqiao University, China
[b] Department of Computer Science and Engineering, Harbin Institute of Technology, China
[c] Oregon State University, USA
[d] Department of Electronic Engineering, Columbia University, USA
[e] School of Computer Science, The University of Adelaide, Australia
[f] School of Information Science and Technology, Xiamen University, China

ARTICLE INFO

Article history:
Received 6 June 2012
Received in revised form
28 April 2013
Accepted 2 October 2013
Available online 11 October 2013

Keywords:
Visual tracking
Weakly supervised learning
Information fusion
Online learning
Adaptive appearance model
Drift problem
Online evaluation

ABSTRACT

Notwithstanding many years of progress, visual tracking is still a difficult but important problem. Since most top-performing tracking methods have their strengths and weaknesses and are suited for handling only a certain type of variation, one of the next challenges is to integrate all these methods and address the problem of long-term persistent tracking in ever-changing environments. Towards this goal, we consider visual tracking in a novel weakly supervised learning scenario where (possibly noisy) labels but no ground truth are provided by multiple imperfect oracles (i.e., different trackers). These trackers naturally have intrinsic diversity due to their different design strategies, and we propose a probabilistic method to simultaneously infer the most likely object position by considering the outputs of all trackers, and estimate the accuracy of each tracker. An online evaluation strategy of trackers and a heuristic training data selection scheme are adopted to make the inference more effective and efficient. Consequently, the proposed method can avoid the pitfalls of purely single tracking methods and get reliably labeled samples to incrementally update each tracker (if it is an appearance-adaptive tracker) to capture the appearance changes. Extensive experiments on challenging video sequences demonstrate the robustness and effectiveness of the proposed method.

1. Introduction

Visual tracking has attracted significant attention due to its wide variety of applications, including intelligent video surveillance, human machine interfaces, robotics, and so on. Much progress has been made in the last two decades; an overview is given in the next section. However, designing robust visual tracking methods is still a challenging problem. Challenges in visual tracking problems include non-rigid shapes, appearance variations, occlusions, illumination changes, cluttered scenes, low frame rate, etc.

Years of research in visual tracking have demonstrated that significant improvements may be achieved by using more sophisticated feature selection or target representation, more elaborate synergies between tracking and classification, segmentation or detection, and taking into account prior information of the scenes and the tracked objects. Since each kind of tracking method has its strengths and weaknesses and is applicable for handling one or a few types of challenges, it is difficult, if not impossible, for a single tracking method to work under a variety of tracking scenarios. Many methods often use sequentially cascaded or parallel majority voting frameworks to fuse the outputs of a number of tracking methods. One of the main challenges affecting these two kinds of fusing schemes is how to measure the performance of a tracker when there is no ground truth available.

In this paper, the proposed tracking method is conceptually different and is based on a new strategy; in contrast to using sequentially cascaded or parallel majority voting schemes, we consider visual tracking in a novel weakly supervised learning scenario where labels are provided by multiple imperfect oracles (i.e., different trackers), and no ground truth is given. A probabilistic method is proposed to explore the alternatives of fusing multiple imperfect oracles for visual tracking, and simultaneously infer the most likely object position and the accuracy of each imperfect oracle.

The inspiration for this work comes from a recently developed machine learning area in weak supervision, where the task is to jointly learn from multiple labeling sources [1–6]. This task

* Correspondence to: School of Information Science and Technology, Haiyun Campus, Xiamen University, Xiamen 361005, China. Tel.: +86 592 2580063.
E-mail address: hanzi.wang@xmu.edu.cn (H. Wang).

0031-3203/$ - see front matter Crown Copyright © 2013 Published by Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.patcog.2013.10.002

underlies several subfields such as data fusion, active learning, transfer learning, multitask learning, multiview learning, learning under covariate shift and distributed inference, which are receiving increasing interest in the machine learning community. The problem of learning from multiple labeling sources is different from the unsupervised, supervised, semi-supervised or transductive learning problems, in that each training instance is given a set of candidate class labels provided by different labelers with varying accuracy, and the ground truth label of each instance is unknown. In practice, a variety of real-world problems can be formalized as multi-labeler problems. For example, there have been an increasing number of experiments using Amazon's Mechanical Turk [7] for annotation. In situations like these, the performance of different annotators can vary widely. Without the ground truth, how to learn classifiers, evaluate the annotators, infer the ground truth label of each data point, and estimate the labeling difficulty of each data point are the main issues addressed by the task of learning from multiple labeling sources. Other examples of a multi-labeler scenario include reCAPTCHA [1], computer-aided diagnosis [4] and search-engine optimizers [5].

To the best of our knowledge, none of the earlier studies have viewed visual tracking as the problem of learning from multiple labeling sources. A new weakly supervised learning based information fusion method is proposed for integrating trackers and leads to encouraging results when it is applied to the task of visual tracking. While most existing fusion-based tracking methods utilize multiple features, the proposed method integrates the results of existing tracking methods which naturally have intrinsic diversity due to their different design strategies. The proposed method has the following advantages:

(1) The proposed method presents a natural way of fusing multiple imperfect oracles to get a final reliable and accurate tracking result. The imperfect oracles can be some imperfect tracking methods in the literature. This avoids the pitfalls of depending on a single tracking method.
(2) The proposed method gives an estimate of the ground truth labeling of training data during tracking in a robust probabilistic inference manner and thus can alleviate the tracker drift problem.
(3) The proposed method can evaluate online the accuracy and trustworthiness of the different tracking methods in the absence of ground truth [8,9]. This allows the best individual method to be used at each time instance.

It is vital to recognize that we are not proposing a single new tracking method, but a new weakly supervised framework to integrate the results of multiple trackers. Therefore, previously established and newly developed trackers can also be potentially exploited by our framework.

Our work is based on the initial version [10]. However, there are a lot of important differences between this work and our previous work, which can be summarized as follows. First, the present paper is rewritten to make it more clear and include a more comprehensive review of related works. Second, additional experimental comparisons with more other state-of-the-art methods on more challenging video sequences are performed to better illustrate the superiority of the proposed method. Third, more discussions about the robustness of the proposed method and perturbations of the solution under simulated dummy trackers have been discussed in the paper.

The rest of the paper is organized as follows. An overview of the related work is given in Section 2. Section 3 introduces our weakly supervised learning formulation for visual tracking, and presents the probabilistic method that jointly estimates the most likely object position and each tracker' accuracy. The detailed tracking method is then described in Section 4. Experimental results are given in Section 5. Finally, it concludes this work in Section 6 and gives suggestions for future research in Section 7.

2. Related work

This section gives a brief review of related tracking methods using online learning and multi-cue fusion techniques.

Although numerous methods have been proposed, robust visual tracking remains a significant challenge. Difficulties in visual tracking include non-rigid shapes, appearance variations, occlusions, illumination changes, cluttered scenes, low frame rate, etc. To solve these challenges, most top-performing methods rely on online learning-based methods [11–15] to adaptively update target appearance. In these methods, visual tracking is formulated as an online binary classification problem and the target model is updated using the images tracked from previous frames. Compared with the methods using fixed target models, such as [16–18], these adaptive methods are more robust to appearance changes. However, the main drawback of these appearance-adaptive methods is their sensitivity to drift, i.e., they may gradually adapt to non-targets.

One popular technique to avoid tracker drift is to make sure the current tracker does not stray too far from the initial appearance model. Matthews et al. [19] are among the first to utilize that technique and provide a partial solution for template trackers. In [20], discriminative attentional regions are chosen on-the-fly as those that best discriminate the current object area from the background region. In that work, tracker drift is unlikely, since no on-line updates of the attentional regions. Furthermore, Fan et al. [63] propose a robust tracking method based on discriminative spatial attention. Grabner et al. formulate tracking as an online semi-supervised learning problem [21]. Combining with a prior classifier, this method takes all incoming samples as unlabeled and uses them to update the tracker. Despite their success, these methods are limited by the fact that they cannot accommodate very large changes in appearance.

To balance semi-supervised and fully adaptive tracking, Stalder et al. [22] present a method using object specific and adaptive priors. In [23], Babenko et al. propose to use a multiple instance learning based appearance model for object tracking. Instead of using a single positive image patch to update a traditional discriminative classifier, they use one positive bag consisting of several image patches to update a multiple instance learning classifier. This method is robust but can lose accuracy if the patches do not precisely capture the object appearance information. In [24–26], co-training is applied to online multiple-tracker learning with different features. The trackers collaboratively classify the new unlabeled samples and use these newly labeled samples with high confidence to update each other. However, independence among different features is required in co-tracking and this condition is too strong to be met in practice.

To incrementally learn from multiple noised data, Lou and Hamprecht [27] propose a structured learning method for cell tracking which formulates tracking by assignment as a constrained binary energy minimization problem. However, the method requires exhaustive assignment annotations of pairs of frames. To address this limitation, they further propose a structured learning method [28] using partial annotations, which has achieved a performance comparable to that obtained from exhaustive annotation. Wang et al. [29] propose an active learning method for solving the incomplete data problem in facial age classification by the furthest nearest-neighbor criterion. In [30], a novel active learning framework is also proposed for video annotation and interactive tracking, in which active learning is used to intelligently query a worker to label only certain objects at

only certain frames that are likely to improve performance. Yao [31] et al. propose a time-weighted appearance model for object tracking. The model is obtained through weighted online structure learning that avoids drift and emphasizes most recent observations. To capture the important dependency among associations, Yang et al. [32] propose a learning-based Conditional Random Field (CRF) model for tracking multiple targets, in which the CRF model is adopted to consider both tracklet affinities and dependencies among them. However, the method is an offline method that combines multiple cues on pre-labeled ground truth data.

Since shape is a powerful tool in image representation [33–35], segmentation-based methods have also been proposed for alleviating the drift problem. Ren and Malik [36] propose a paradigm of tracking by repeatedly segmenting figures from the background, which alleviates the drift problem through accurate spatial support obtained in segmentation. Global shape information is an effective top-down complement to bottom-up figure-ground segmentation as well as a useful constraint to avoid drift during adaptive tracking. Yin and Collins [37] propose a novel method to embed global shape information into local graph links in a Conditional Random Field (CRF) framework. Fan et al. [38] propose a matting-based tracking method, which relies on trackable points on both sides of the object boundary. Wang et al. [39] propose a superpixel tracking method by using mid-level cues that capture spatial information to some extent. However, in general, segmentation based methods only benefit from the situation when the foreground is in high contrast to the background, which is not always the case in natural scenes.

Hare et al. [40] and He et al. [41] use a tracker based on online learning for key-point matching. They perform tracker update only when the motion consensus of local descriptors is verified. These two methods can alleviate the drift problem to some extent but they only work well for the tracking of texture-rich objects.

A number of attempts have been made to utilize multiple observation models to improve the performance of a tracker. The literature on the tracking methods using multiple cues essentially demonstrates the concept of different cues complementing each other and thus overcoming the failure cases of individual cues. Multiple cue integration can be done by assuming that different cues are independent of or dependent on each other. Considering cue independence, Birchfield [42] combines the intensity gradients around an object boundary and the color histogram of the object region interior with equal weights, in a head tracker. Although robust to some extent, the method seems to lack solid theory. Considering cue dependence, Wu and Huang [43] formulate the integration of multiple cues by a factorized graphical model (i.e., a variant of particle filter) and use a variational analysis method to approximate the probabilistic inference. Instead of integrating multiple cues in a single graphical model, Moreno-Noguer et al. [44] represent each cue by a different Bayesian filter and assume sequentially conditional dependent among them, thus, cue dependence is considered in the re-sampling stage in the particle filtering. Despite their success, these two methods are limited by the fact that they cannot accommodate very large changes in appearance due to using fixed appearance models. Yang et al. [45] propose an adaptive way to integrate multi-cues in tracking multiple humans driven by human detectors. Nonetheless, it is extremely hard to design a perfect human detector with both high detection rate and precision rate. Recently, Stenger et al. [46] investigate different combinations of tracking methods. Given a particular tracking scenario, they try to learn which methods are useful and how they can be combined to yield good results by evaluating all pairs/triplets (using 2 different schemes). However, one limitation of that method is the requirement of an extra offline training step. Santner et al. [47] propose a method which combines three trackers (i.e., an online random forest-based tracker, a correlation-based template tracker and an optical-flow-based mean shift tracker) in a cascade-style. However, how to set the overlapping and confidence thresholds that trigger the cascading process is crucial. In [48], a visual tracker sampler is proposed, in which multiple appearance models, motion models, state representation types, and observation types are sampled via Markov Chain Monte Carlo to generate the sampled trackers. Wang et al. [49] and Erdem et al. [50] present a method which dynamically integrates multiple cues in a particle filter respectively.

Lu and Hager [51] propose a model adaptation method driven by feature matching and feature distinctiveness that is robust to drift. Oron et al. [52] develop another method to deal with the drift problem by automatically estimating the amount of local (dis) order in an object. In [53], a discriminative metric is learned for robust visual tracking. In addition, with the popularity of low-rank subspaces and sparse representations in image processing and machine learning, a variety of low-rank and sparse representations based tracking methods have been recently proposed [54–56] for robust object tracking. For a full review of the tracking literature, please refer to [57].

3. An weakly supervised view of visual tracking

3.1. The problem

For online learning based visual tracking, a tracker observes samples (typically image patches) in each frame and predicts their labels to be either foreground or background. At the end of each frame, the adaptive tracker uses the newly obtained sample-label pairs to improve its prediction rule for the following frames. However, due to the challenges in natural scenes and shape changes, the tracker may gradually adapt to non-targets, i.e., the model drift problem. The main reason behind model drift is that the tracker is updated using ***a self-learning policy in the absence of ground truth***. Many demonstrations have shown that aggregating the judgments of a number of components (e.g., detectors, trackers and recognizers) enhances the tracking performance to some degree. This phenomenon is known as the "wisdom of crowds". Thus, the performance of one tracker may be assessed by using a majority voting scheme based on a number of other trackers, which is often utilized in tracking applications. However, it is not clear how one can measure the performance of the other trackers when there is no ground truth available. This is chicken-and-egg problem. So the question is, in the absence of ground truth, how to effectively address the above-mentioned problem in order to adapt to the appearance changes of a target while at the same time avoiding the model drift problem.

3.2. Optimal integration of labels from labelers of unknown expertise

Recently, weakly supervised learning from multiple labeling sources has received the interest of many researchers. Before introducing our work, we briefly review the work of [6] as it forms the basis of the proposed method.

Following the notations of Whitehill et al. [6], a typical weakly supervised learning scenario from m labelers consists of a training set $S=\{(x_j,l_j)\}_{j=1}^{n}$ containing n samples, where $x_j \in X$ is a sample (typically a d-dimensional feature vector) and $l_j=\{l_{ij'}|j'=j\}$ is the set of all given labels for a sample $x_j l_{ij} \in \{0,1\}$ is the label assigned to the sample x_j by the ith labeler. It should be noted that not all labelers are required to label all the samples. In this case, the index variable i in $l_{ij'}$ refers only to those labelers who labeled sample x_j. Several factors determine the observed labels $L=\{l_{ij}\}$: (1) the labeling difficulty of each sample; (2) the actual true label; and (3) the expertise/accuracy of each labeler. According to the training

set $S=\{(x_j,l_j)\}_{j=1}^{n}$, the task is to jointly learn the labeling difficulty of each sample, the expertise/accuracy of each labeler, and the true label of each sample.

Denote $1/\beta_j \in (0,+\infty)$ as the labeling difficulty of the sample x_j, $z_j \in \{0,1\}$ as the true label of the sample x_j and $\alpha_i \in (-\infty,+\infty)$ as the expertise of the ith labeler. The probability of the correct label l_{ij} assigned to the sample x_j by the ith labeler is generated as follows:

$$p(l_{ij}=z_j|\alpha_i,\beta_j)=\frac{1}{1+e^{-\alpha_i\beta_j}} \tag{1}$$

Based on the model, we obtain a high probability of labeling correctly when the labelers' accuracies are high and the sample labeling difficulties are low.

Fig. 1 shows the structure of the model, in which a Gaussian prior is used for α_i and $\beta_j'(=In\beta_j)$ respectively. It is worth emphasizing that the parameter β_j is re-parameterized $\beta_j=e^{\beta_j'}$ since we need a prior that does not generate a negative value. In addition, when the true label z_j of a particular sample is known, we can use it to better estimate the other model parameters.

An Expectation–Maximization algorithm (EM) is used to obtain maximum likelihood estimates of the parameters of interest: $Z=\{z_j\in\{0,1\}\}$, $\alpha=\{\alpha_i\}$ and $\beta=\{\beta_j\}$. The EM algorithm is an efficient iterative procedure to compute the maximum-likelihood solution in the presence of hidden data. The authors of [6] use variables Z, α and β as the hidden data. Each iteration of the EM algorithm consists of an Expectation (E) step and a Maximization (M) step.

E step: Assuming that we have a current estimate of the values of α, β from the last M step and the observed labels, we can compute the posterior probabilities of all $z_j\in\{0,1\}$:

$$\begin{aligned} p(z_j|L,\alpha,\beta) &= p(z_j|l_j,\alpha,\beta_j) \propto p(z_j|\alpha,\beta_j)p(l_j|z_j,\alpha,\beta_j) \\ &\propto p(z_j)\textstyle\prod_i p(l_{ij}|z_j,\alpha_i,\beta_j) \end{aligned} \tag{2}$$

where $p(z_j|\alpha,\beta_j)=p(z_j)$ because true sample labels, sample difficulties and labelers' accuracies are assumed to be generated independently in the graphical model.

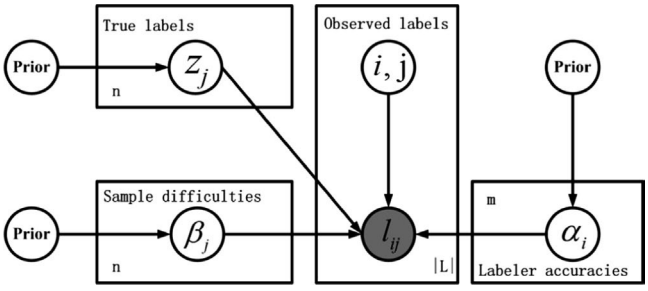

Fig. 1. Graphical model of sample difficulties, true sample labels, observed labels, and labelers' accuracies. Only the shaded variables are observed.

M step: To estimate the values of the parameters (α,β), we maximize an auxiliary function Q,

$$(\alpha^*,\beta^*)=\text{argmax}_{(\alpha,\beta)}Q(\alpha,\beta) \tag{3}$$

where the function Q is defined as the expectation of the joint log-likelihood of the observed and hidden variables (L,Z) given the parameters (α,β), w.r.t. the posterior probabilities of the Z estimated in the last E step:

$$\begin{aligned} Q(\alpha,\beta) &= E[Inp(L,Z|\alpha,\beta)] = E[In\textstyle\prod_j(p(z_j)\prod_i p(l_{ij}|z_j,\alpha_i,\beta_j))] \\ &\quad \text{(since } l_{ij} \text{ are cond. indep. given } Z,\ \alpha,\ \beta) \\ &= \textstyle\sum_j E[Inp(z_j)]+\sum_{ij}E[Inp(l_{ij}|z_j,\alpha_i,\beta_j)] \end{aligned} \tag{4}$$

where the expectation is taken over Z given the previous parameter values α^{old}, β^{old} as estimated at the last E-step. Using gradient ascent, we can estimate the values of α^* and β^* that locally maximize Q.

The convergence rate of the E-step is linear in the number of samples and the total number of labels. For the M-step, the computational time for computing the values of Q and its gradient increases linearly with the number of samples, number of labelers, and total number of sample labels. For simplicity, we denote the

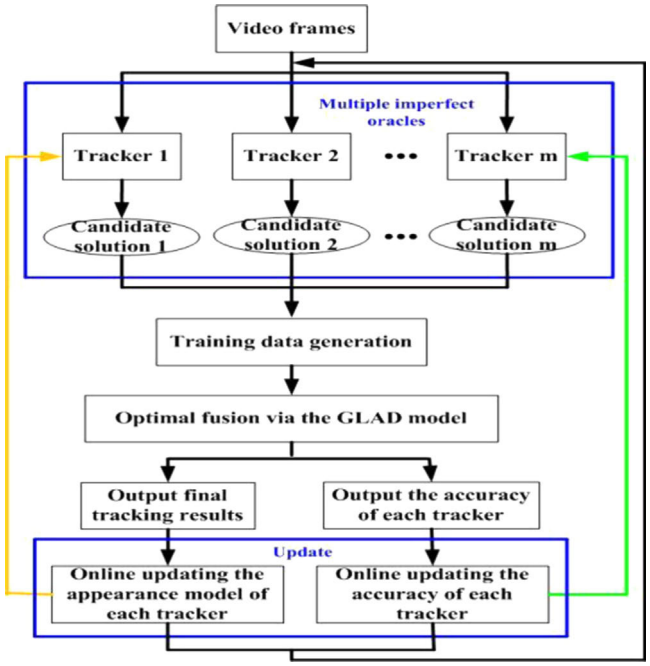

Fig. 3. Overview of the proposed tracking method. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

For t = 1, 2, …

1. Generate a set of samples S_t $\{x_t^1, x_t^2, \ldots, x_t^{N_t}\}$.
2. Predict L_t^j $\{l_t^{ij'} \mid j' \quad j\}$ for each sample x_t^j using multiple imperfect oracles.
3. Jointly estimate the ground truth label of each sample and the accuracy of each imperfect oracle from the current observed sample set S_t.
4. Update multiple imperfect oracles based on the estimated tracking results.

End

Fig. 2. Weakly supervised learning setting for visual tracking.

above inference method as Generative model of Labels, Abilities, and Difficulties (GLAD), a name termed in [6].

3.3. Visual tracking via learning from an oracle set

A unified weakly supervised learning framework for visual tracking, in which multiple imperfect oracles cooperate on tracking objects, can be formulated as the following problem:

Consider that m imperfect oracles observe an input sequence $s_1, s_2, ..., s_T$, where $s_t = \{x_t^1, x_t^2, ..., x_t^{N_t}\}$ is a sample set obtained in frame t and x_t^j is the jth sample in s_t. Denote $L_t^j = \{l_t^{ij} | j' = j\}$ as a set of candidate class labels for the sample x_t^j, where $l_t^{ij} \in \{0, 1\}$ is the label provided by the ith oracle. Our weakly supervised learning-based tracker setting is depicted in Fig. 2.

4. Robust visual tracking

In this section, we develop the proposed tracking method based on the weakly supervised learning setting described above. The basic idea is to embed a heterogeneous set of trackers into the proposed weakly supervised learning setting to form a robust tracker. The proposed tracking method is schematically shown in Fig. 3.

Specifically, the proposed tracking method works as follows: for the current video frame, we first estimate a set of candidate solutions that are obtained by a heterogeneous set of tracking methods. Then, the training data are adaptively selected according to a heuristic strategy and used for GLAD. After the training data generation, the GLAD model is used to simultaneously infer the most likely object position and the accuracy of each tracker. A testing sample with the maximum probability of belonging to the positive sample set is chosen to be the new object position, and is also retained as a positive training sample for the tracker update in the following step. An online evaluation strategy is developed to incrementally update the accuracy of each tracker. Meanwhile, the target appearance model of a candidate tracker is updated if it is an appearance-adaptive tracker. The tracking procedure continues in this iterative fashion until the end of the video. Below we give a detailed description of each component in this framework, and then the proposed method is summarized.

4.1. A heterogeneous set of oracles

The key part of the proposed method proceeds by first obtaining a set of different tracking results that serve as proposal solutions, and then optimally fusing them using the GLAD model. The success of the method thus depends on the availability of good proposal solutions.

It is important to note that the proposal solutions need not to be good in the whole tracking process in order to be "useful". Instead, each solution may contribute to a particular phase in the whole tracking process, if it contains reasonable tracking results for that phase. This suggests the use of different tracking methods with different strengths and weaknesses for computing the proposals. In our experiments, we use the following six tracking methods as the selected proposal solutions:

(1) **Fragments-based Tracker (FT)** [18]. The tracker uses static appearance models to obtain solutions. Due to using integral gray histograms and part based appearance model, such solutions are very efficient and robust to occlusions. However, the method tends to have difficulties in tracking objects that exhibit significant appearance changes.
(2) **Online Boosting Tracker (OBT)** [14]. The tracker uses online boosting to obtain solutions. Due to the properties of the method, such solutions are able to adapt to appearance changes of the object, but unfortunately suffer from the model drifting problem.
(3) **Semi-Supervised Online Boosting Tracker (SSOBT)** [21]. The tracker uses semi-supervised online boosting method to obtain solutions. Such solutions can alleviate the model drifting problem since the tracker cannot get too far away from the prior. However the prior might be too strong (i.e., limited appearance changes and partial occlusions) and generic (i.e., no discrimination between different objects from one class).
(4) **Beyond Semi-Supervised Tracker (BSST)** [22]. The tracker balances between semi-supervised and fully adaptive tracking to obtain solutions.
(5) **Online Multiple Instance Learning-based Tracker (OMILT)** [23]. The tracker uses online multiple instance learning (MIL), in which one positive bag consisting of several image patches is used to update a MIL classifier, to obtain solutions.
(6) **SURF Tracker (ST)** [41]. The tracker uses the SURF descriptors to obtain solutions. Such solutions often contain good results for textured objects but are virtually useless for textureless objects.

The reasons that we choose these proposal solutions are:

(1) These solutions consist of a variety of complementary cues, such as gray histogram-based patch matching [18], motion consensus of local descriptors [41] and online classification using Haar features [14,21–23].
(2) The features used in the six tracking methods can be extracted in an efficient manner by using the integral image technique.
(3) The source codes of these methods are publicly available. This makes parameter tuning to achieve the results reported in original literature and the comparison with these methods fair.
(4) Candidate solutions of each tracking method can be obtained independently, which allows for a high-speed parallel implementation if needed.

It is important to note that other (potentially more efficient and robust) methods may also be considered. The proposed method provides a principled way of fusing proposal solutions from various tracking methods in the literature.

4.2. Heuristic selection of training data for optimal fusion

It can be seen from Section 3 (B) that the computational complexity of the E-Step is linear to the number of patches and the total number of labels. For the M-Step, the values of Q and ∇Q must be computed repeatedly until convergence. Thus, the computations to compute each function are linear to the number of patches, the number of labelers, and the number of image labels. To make the proposed tracking method more efficient in practice, we develop in this section a heuristic way to select training data for the GLAD model.

Consider that m imperfect oracles receive an image patch set $s_t = \{x_t^1, x_t^2, ..., x_t^{N_t}\}$ within the current search window in the tth frame, where x_t^j is the jth image patch of interest. Denote $P_t = \{p_t^1, p_t^2, ..., p_t^m\}$ as a set of proposal solutions' bounding boxes obtained via m imperfect trackers $T = \{T_1, T_2, ..., T_m\}$ in the tth frame, $D(\cdot, \cdot)$ as the distance between the centers of two image patches, and $l_t^{ij} \in \{0, 1\}$ as the label of the image patch x_t^j obtained by the tracker T_i. For each tracker T_i, denote Top_t^i as the set of image patches having the 10 highest confidences (estimated by the tracker T_i) belonging to the positive samples in the tth frame. The training data is selected as shown in Fig. 4:

In addition, we initialize the parameter β_t^j of the labeling difficulty of the image patch x_t^j by using the majority voting

strategy:

$$\beta_t^j = \exp\left(k_1 \times \left|\left(\frac{2}{m}\sum_{i=1}^{m} l_t^{ij}\right) - 1\right|\right) \quad (5)$$

where k_1 is a normalized factor and typically set as 1.6 in our case. In other words, we have computed β_t^j according to the degree of agreement among all trackers. If most trackers label the image patch x_t^j as 0, then it has a high β_t^j (easier to assign label). Similarly, Eq. (5) will set β_t^j high if most trackers label the image patch x_t^j as 1.

4.3. Online evaluation of trackers

Automatic evaluation of visual tracking methods in the absence of ground truth is a challenging and important problem. As the accuracy α_i of each tracker can be inferred in each frame and the Q function in Eq. (4) can be modified straightforwardly to handle a prior over each α_i by adding a log-prior term for each of these variables, we can determine online the most accurate trackers. We are interested in the trackers which have the most supporting evidence over time. After the tracking of each frame, we first check m proposal solutions of the imperfect trackers against the fusion results. A tracker is deemed to fail if the center location error (pixels) between its proposal bounding box and the final bounding box obtained by fusion is greater than a distance threshold (typically set as 5). This threshold has relatively small influence on the performance of the tracker. Then, the prior accuracies of the m trackers at time t are adjusted as follows:

$$\alpha_i^t = (1 - w_i^t)\alpha_i^{t-1} + w_i^t M_i^t \quad (6)$$

where w_i^t is the learning rate and M_i^t is 1 for the tracker which it succeeds and 0 for the remaining trackers.

In addition, in order to avoid wrong updating and improve robustness, we adaptively adjust the learning rate w_i^t of the ith tracker as follows:

$$w_i^t = \begin{cases} \frac{k_2}{1+e^{-\alpha_i}}, & \text{if} \quad M_i^t = 1 \\ \frac{k_2}{1+e^{\alpha_i}}, & \text{if} \quad M_i^t = 0 \end{cases} \quad (7)$$

where k_2 (typically its value is set to 0.1) is a normalized factor and α_i is the accuracy of the ith tracker, which is one of the output of the GLAD model. In the case that the ith tracker succeeds (i.e., $M_i^t = 1$), if the accuracy of the ith tracker α_i is large, the learning rate w_i^t is set to be large to quickly adapt. Otherwise, w_i^t is set to small.

- **For** j $1,\ldots,N_t$
  - **If** x_t^j P_t $\{p_t^1, p_t^2, \ldots, p_t^m\}$
    - **For** i $1,\ldots,m$
      - **If** ($D(x_t^j, p_t^i)$ a threshold $\|$ x_t^j Top_t^i)
        l_t^{ij} 1.
      - **Else**
        l_t^{ij} 0.
    - **End for**
  - **Else** ignore the image patch x_t^j .
- **End for**

Fig. 4. Illustration of the heuristic training data selection for optimal fusion.

4.4. Summary of the proposed method

A summary of our weakly supervised learning based tracking method is described in Fig. 5. From Fig. 5, we can see that the important dependency among oracles is modeled by the proposed method via the GLAD model. For example, the proposed method can be understood as the following: if one tracker (denoted as A) disagrees with the other trackers while the other trackers are confident and agree with each other, the expertise/accuracy of the tracker A will be reduced and the prediction of the tracker A will be influenced towards the agreed location (depending on the overall probability map). Meanwhile, the expertise of the other trackers is increased. Otherwise, the tracker A gives its prediction and updates its appearance model as if there were no other trackers. Therefore, the trackers can keep relative independence and also maintain mutually beneficial interactions with each other. This makes the proposed method robust to inaccurate trackers.

5. Experiments

In this section, we first introduce the setting of our experiments. Then, we apply the proposed tracking method to a number of challenging video sequences, and systematically compare it with several state-of-the-art trackers. Furthermore, some specific

Algorithm 1 Weakly Supervised Learning from Multiple Imperfect Oracles for Visual Tracking

Initialization:

1. Acquire one manually labels the first frame.
2. Construct the target appearance model for each candidate tracker.
3. Initialize the accuracy of each candidate tracker .

for t = 2 to the end of the video

1. Generate a set of proposal solutions via a heterogeneous set of trackers.
2. Select training data for optimal fusion according to a heuristic strategy.
3. Run the GLAD model.
4. Localize the new target position as the sample with the maximum probability belonging to the positive sample class.
5. Update the accuracy of each candidate tracker.
6. Update target appearance model of each candidate tracker if it is an appearance-adaptive tracker.

end for

Fig. 5. Summary of the proposed tracking method.

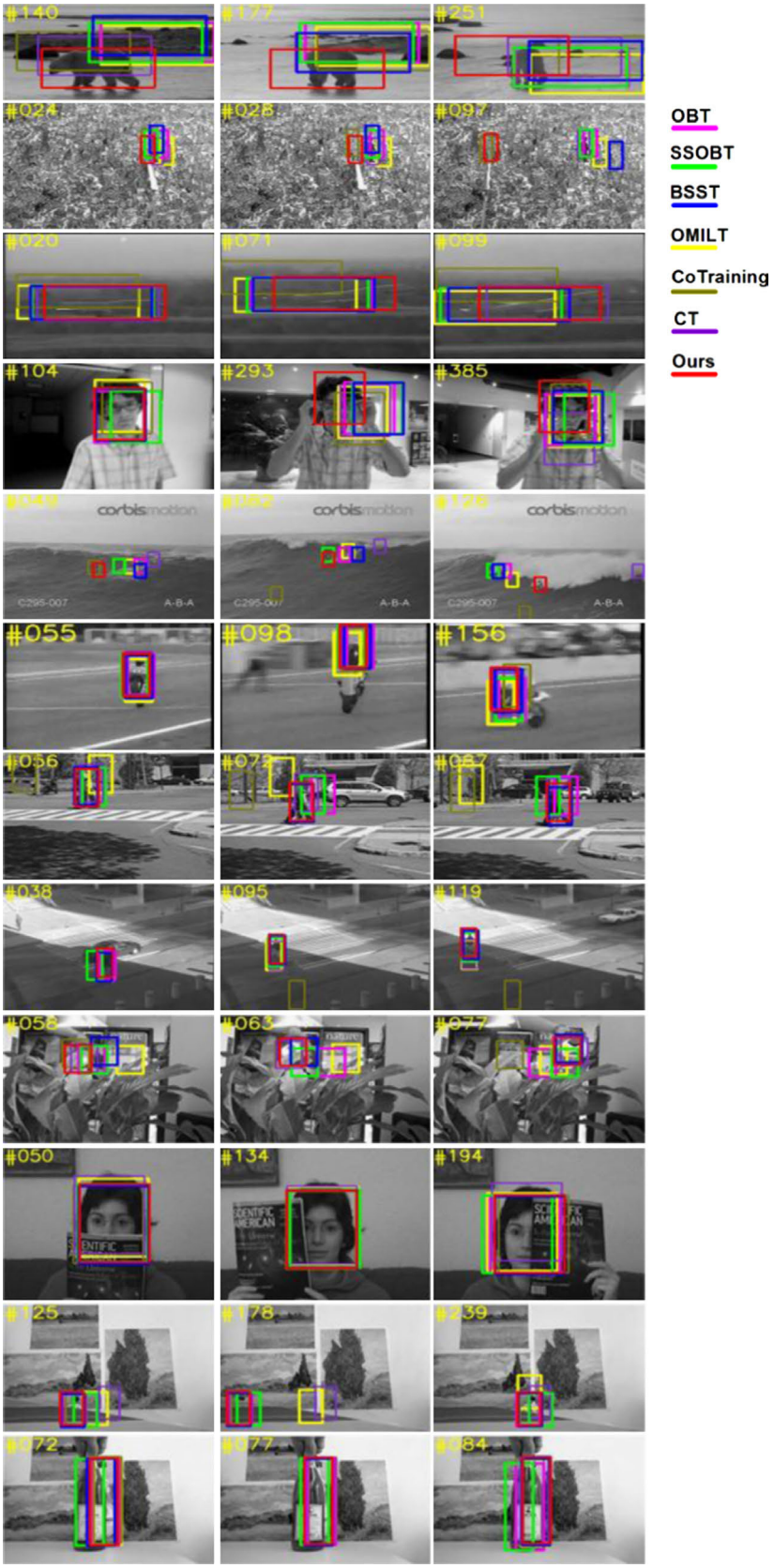

Fig. 6. Qualitative comparison of our method, **OBT**, **SSOBT**, **BSST**, **OMILT**, **CoTraining** and **CT** on the twelve challenging video sequences. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

properties of our tracker are empirically evaluated by a group of carefully designed experiments using different subsets of the imperfect oracles, some of which may be rather incompetent, or even malicious. The reason why these different subsets are chosen is that we want to evaluate several issues of the proposed method. Please see Sections 5.3–5.7 for further explanations. Before

describing the evaluation and analysis, we first introduce the setting used in our experiments.

5.1. Experiment setting

In our experiments, we choose to track only the location for simplicity and computational efficiency reasons. Thus, no scale and rotation adaption are implemented. In any case, both can be incorporated with slight modification of the proposed method. We also do not use any motion model to predict the new position. The centroid of the search window is the same as that of the previous target bounding box. The processing speed depends on the size of the search window which we have defined by enlarging the target region by half of its size in each direction. In addition, multiple imperfect oracles are parallelly implemented. Typically, the number of iterations taken to converge to the optimization of $Q(\alpha,\beta)$ (i.e., Eq. (4)) is less than 100 while the fusion time on our computer is about 10 ms when we use the training data obtained by using our selection scheme. Therefore, the running time of the proposed tracking method is determined by the slowest tracker since the fusion time is negligible and all trackers are run in parallel. We have achieved the processing speed of 10 fps at the resolution of 320×240 pixels (the running time could be reduced substantially by using multiple cores). The initial accuracy α_i^0 of each oracle is 1. The proposed method is implemented by using C++, on a machine with Intel Pentium Dual 2.0 GHz processor.

For performance evaluation, we compare the proposed method against several state-of-the-art methods in visual tracking – **FT** [18], **OBT** [14], **SSOBT** [21], **BSST** [22], **OMILT** [23], **CoTraining** [26], **CT** [56], and **ST** [41]. For the first seven methods, we use the same parameters as the authors have given on their websites [58–62] for all of the experiments. **ST** is implemented by ourselves according to [41]. In addition, to more clearly illustrate that we can give proper labels to new samples for further tracker training, we also implement the variations of the six methods (i.e., **FT**, **OBT**, **SSOBT**, **BSST**, **OMILT, ST**) separately. More specifically, the variation of each method updates itself with reliable labeled samples obtained by the final fusing results. We refer to the variations of the six trackers as **FT_V**, **OBT_V**, **SSOBT_V**, **BSST_V**, **OMILT_V** and **ST_V** respectively.

We have tested the tracking methods by using a number of challenging video sequences from the existing literature as well as our own collection. Both qualitative and quantitative comparisons are done to evaluate the involved tracking methods. The quantitative performance is measured by the center location errors (pixels) for each frame and the averaged center location errors for the whole sequences. The ground truth is achieved by manually labeling all frames from the video sequences. Due to the space limitation, we only show twelve challenging video sequences (see Fig. 6) in this paper. The *Bear* sequence on the first row is taken from the internet. It contains a running bear captured by a moving camera. This sequence is challenging since it suffers from both light and pose changes, and dramatic figure/ground appearance pattern changes. The *Back_Clutter* sequence on the second row is taken from [59]. This sequence contains a "Waldo" doll moving in front of a background with very similar texture. Tracking the "Waldo" doll is a very difficult visual tracking task. A tracking method should neither track the needle to which the target object is attached nor track an object in the background. The *Airplane* sequence on the third row is a low quality surveillance video captured by a PTZ camera watching a runway. The *David_ Indoor* sequence on the fourth row is taken indoor and contains a person moving from dark toward bright area with large illumination and pose changes. The *Person_Surf* sequence on the fifth row contains a surfer riding a wave. The turbulent wakes created by sweeping wave and the surfer create significant challenges for tracking. The *Motor* sequence on the sixth row contains a motorcycle moving with large pose and lighting variation in a cluttered background. The sequences on the next two rows are from [15]. The *Two_Pedestrian* sequence on the seventh row is captured with a moving camera, which contains two walkers. This is a very difficult video sequence for visual tracking due to dramatic figure/ground appearance pattern changes in this sequence; The *One_Pedestrian* sequence on the eighth row is a low resolution and low figure-ground contrast surveillance video where the person's leg part has similar color to the ground and the person's upper part is also similar to the passing car's color. The *Tiger* sequence on the ninth row is taken from [60]. This sequence suffers from frequent occlusions, fast motion, pose and appearance changes. The *Face* sequence on the tenth row is from [58], which suffers from partial occlusions. The sequences on the last two rows are taken from [59]: The *Toy* sequence on the eleventh row contains a toy which should be tracked while moving behind a static object. A tracking method should continuously track the object without being affected by the static object; The *Bottle* sequence on the last row contains a static bottle which is gradually occluded by another moving bottle. A tracking method should not be affected by the distracting bottle. A brief summary of the above-mentioned challenging video sequences based on the types of specific tracking problems is listed in Table 1.

5.2. Comparison with other trackers

To show the advantage of the proposed method over the other competing trackers, we perform a number of experiments using a rich set of imperfect oracles consisting of **OBT_V**, **SSOBT_V**, **BSST_V**, and **OMILT_V.** We report the results in this section.

According to Table 1, the used test sequences can be classified into two categories based on if the sequences contain occlusions.

Table 1
Classification of the used test sequences based on the types of specific tracking problems.

Image sequence	Moving camera	Light changes	Pose changes	Scale changes	Clutter	Appearance changes	Occlusion
Bear	√	√	√	√	×	√	×
Back_Clutter	×	×	×	×	√	×	×
Airplane	√	√	√	√	√	√	×
David_Indoor	√	√	√	√	×	√	×
Person_Surf	√	√	×	×	√	√	×
Motor	√	×	√	√	×	√	×
Two_Pedestrian	√	√	√	√	×	√	×
One_Pedestrian	×	√	×	√	√	×	×
Tiger	×	√	√	√	×	√	√
Face	×	×	×	×	×	×	√
Toy	×	×	×	×	×	×	√
Bottle	×	×	×	×	×	×	√

(1) The image sequences without occlusions contain the *Bear*, *Back_Clutter*, *Airplane*, *David_Indoor*, *Person_Surf*, *Motor*, *Two_Pedestrian* and *One_Pedestrian* sequences. (2) The image sequences with occlusions contain the *Tiger*, *Face*, *Toy* and *Bottle* sequences.

Fig. 6 shows the qualitative comparison results of the trackers, i.e., our method (in red), the **OBT** (in magenta), **SSOBT** (in green), **BSST** (in blue), **OMILT** (in yellow), **CoTraining** (in dark yellow), and **CT** (in violet), on the twelve challenging video sequences.

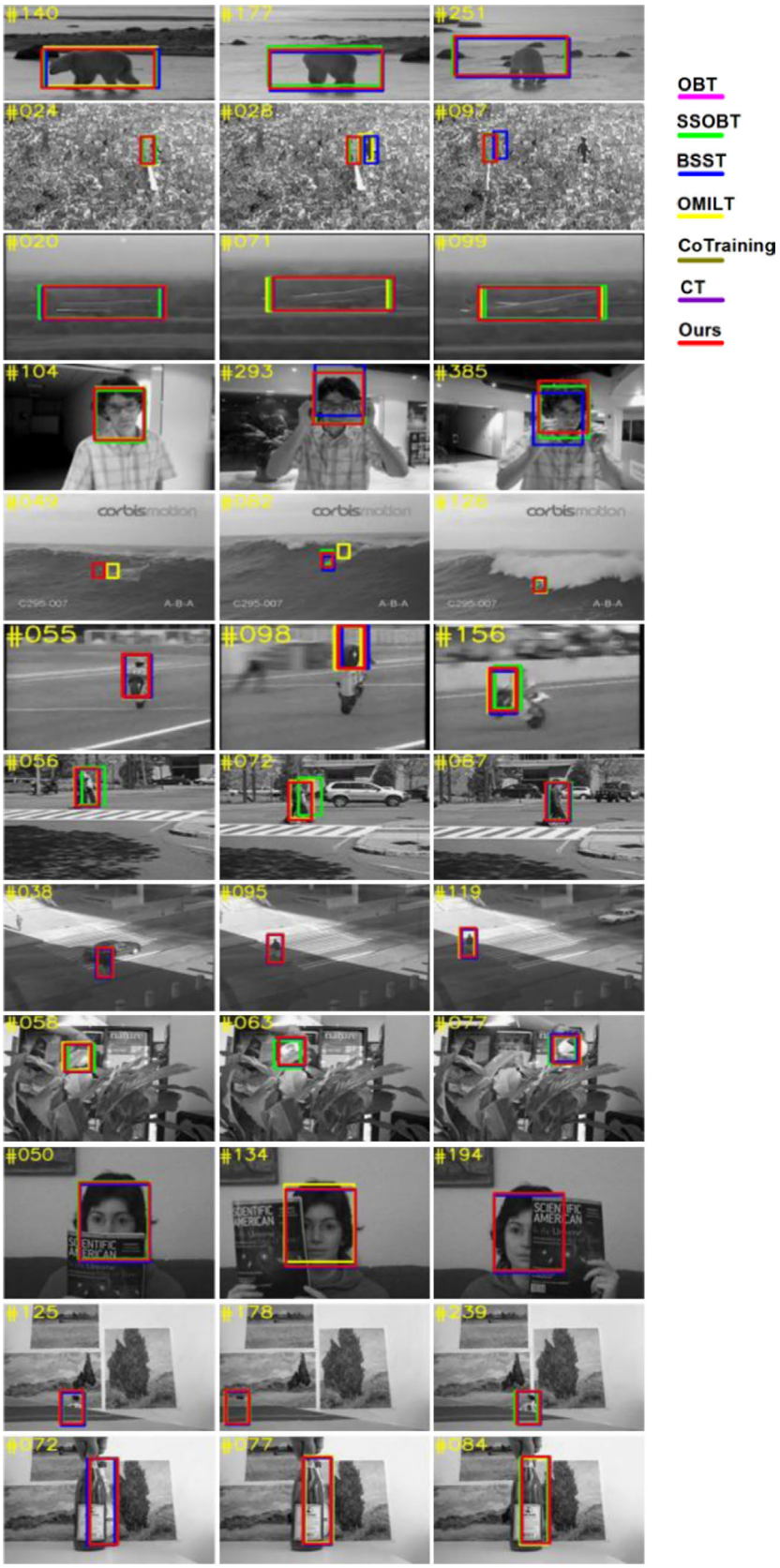

Fig. 7. Qualitative comparison of our method, **OBT_V**, **SSOBT_V**, **BSST_V** and **OMILT_V** on the twelve challenging video sequences.

Difficulties in the eight image sequences without occlusions include non-rigid shape and appearance variations of the object, pose changes, scale changes, illumination changes, cluttered scenes, etc. These eight image sequences include both indoor and outdoor scenes. As can be seen from Fig. 6, the proposed method efficiently combines the proposal solutions provided by multiple imperfect oracles (i.e., **OBT_V**, **SSOBT_V**, **BSST_V**, and **OMILT_V**), and is able to obtain better results than any of the comparison methods (i.e., **OBT**, **SSOBT**, **BSST**, **OMILT, CoTraining,** and **CT**) for these eight image sequences without occlusions. The competing methods either perform poorly in some cases (i.e., for the *Bear*, *Back_Clutter*, *Airplane*, *Person_Surf* and *Two_Pedestrian* sequences) or achieve comparable performance to our method (i.e., for the *David_Indoor*, *Motor* and *One_Pedestrian* sequences). As expected, the proposed method can avoid the pitfalls of purely single tracking methods and get reliably labeled samples to incrementally update each tracker (if it is an appearance-adaptive tracker) to capture object appearance changes.

So far, we have only considered the image sequences without occlusions. To illustrate the performance of our method for image sequences with significant occlusions, we further evaluate the proposed method by using four sequences including occlusions (i.e., the *Tiger*, *Face*, *Toy* and *Bottle* sequences). Based on the experimental results shown in Fig. 6, our method outperforms the competing methods or achieves comparable performance to that of the competing methods for handling occlusions. For these sequences, our tracker could achieve similar results to those obtained by the best imperfect oracle in each time step. For example, for the *Toy* sequence shown in the eleventh row of Fig. 6, it is obvious to see that our tracker (in red) can always obtain similar results to those obtained by the best imperfect oracles in frames #125, #178 and #239. Moreover, for the *Tiger* sequence, the frequent occlusions continuously change in the object's pose and quick variation in the foreground appearance results in the early failure of the single imperfect oracle (i.e., **OBT**, **SSOBT**, **BSST**, **OMILT**, **CoTraining**, and **CT**).

According to the overall results, the proposed method outperforms the competing methods because it considers visual tracking in a weakly supervised learning scenario where (possibly noisy) labels are provided by multiple imperfect oracles. By simultaneously inferring the most likely object position and the accuracy of each imperfect oracle via the proposed probabilistic method, we can obtain training data during tracking in a robust manner for further tracker update. Therefore, the model drift problem is alleviated in the proposed method. This is further verified in the experiments on the four variations (i.e., **OBT_V**, **SSOBT_V**, **BSST_V** and **OMILT_V**) of the competing trackers (i.e., **OBT**, **SSOBT**, **BSST** and **OMILT**). As shown in Fig. 7, it is obvious that the four variations outperform their corresponding trackers. Note that the four variations are not independent trackers. They are merely component trackers within our method which receive more reliable samples from our method (see Section 5.1 for the description).

The quantitative comparison results of the competing trackers are listed in Table 2 and Fig. 8 respectively. As we can see, the continuously changing background and quick variation in foreground result in the failure of some imperfect oracles, e.g., the **OBT**, **SSOBT**, **BSST, OMILT, CoTraining** and **CT**; while our method can successfully track the targets for almost the full length of all these sequences.

Table 2
The average center location errors in pixels. The quantitative comparison results on the twelve challenging sequences obtained by our method, the **OBT**, **SSOBT**, **BSST**, **OMILT, CoTraining** and **CT**, respectively.

Image sequence	OBT	SSOBT	BSST	OMILT	CoTraining	CT	Ours
Bear	76	75	63	85	72	37	18
Back_Clutter	89	94	111	101	6	35	3
Airplane	28	27	30	45	55	102	7
David_Indoor	16	24	15	18	23	17	11
Person_Surf	35	37	42	34	90	55	5
Motor	22	21	22	21	20	6	23
Two_Pedestrian	20	21	16	60	62	36	15
One_Pedestrian	4	4	4	5	43	3	5
Tiger	13	13	14	27	46	26	9
Face	2	2	3	2	8	9	7
Toy	4	6	3	45	5	41	5
Bottle	4	3	2	5	4	2	2

5.3. How many oracles?

Since the proposed method is an open framework in which any tracking methods can be integrated and its success depends on the availability of good proposal solutions, the following questions arise: (1) How sensitive is our method to the number of oracles and the robustness of a single imperfect oracle? (2) How difficult is it to obtain a good set of imperfect oracles? To answer these two questions, we calculate the tracker accuracies in different configurations of an oracle set. Instead of examining all the possible combinations of trackers, we use a simple greedy heuristic, i.e., increase the oracles set by adding one oracle at a time. The measurements are made for several image sequences. The results for the *Back_Clutter* sequence are plotted in Fig. 9. Obviously, for different combinations of multiple imperfect oracles, a good combination can be chosen across a wide range of oracle sets. Similar observations are made for all the other test sequences. This property significantly eases the selection of a rich set of imperfect oracles. Furthermore, the experiments have shown that a good set of multiple oracles for a sequence usually performs well for other sequences (please see Fig. 6).

5.4. Multiple imperfect oracles: whom to trust?

One big advantage of multiple oracle based tracking lies in that the proposal solutions do not necessarily have to be good in the whole tracking process in order to be “useful”. To verify this advantage, we check the accuracies of the oracles over time for all the test sequences and give a typical example in Fig. 10. In our method, if one oracle gives correct tracking results in current frame, then its accuracy is increased; otherwise, its accuracy is decreased. As shown in Fig. 10, each oracle may contribute to a particular time in the whole tracking process, if it contains reasonable tracking results for that time.

5.5. Illustration of the fusion process

In this section, we show the fusion process through two tracking examples and how the training samples are reliably labeled to alleviate the tracker drift problem. In Fig. 11, we visualize some representative image patches and their (possibly noisy) labels (i.e., 0 or 1) that are provided by the multiple imperfect oracles from the frame #189 in the *David_Indoor* sequence and the frame #31 in the *Person_Surf* sequence respectively. As expected, though the positive/negative ratio of an image patch is 1:3 (see the fourth row in Fig. 11(a)) or 2:2 (see the second row in Fig. 11(b)), we can get the most likely target samples for further training. Meanwhile, these two case studies also show the proposed method is robust to noisy (or adversarial) labeling and the advantage of the weak supervised learning scheme over the majority voting scheme for fusing the tracking results. For example, the image patch on the fourth row in Fig. 11 (a) is labeled as 0 by **OBT_V**, **SSOBT_V** and **OMILT_V**. Only **BSST_V** labels it as 1.

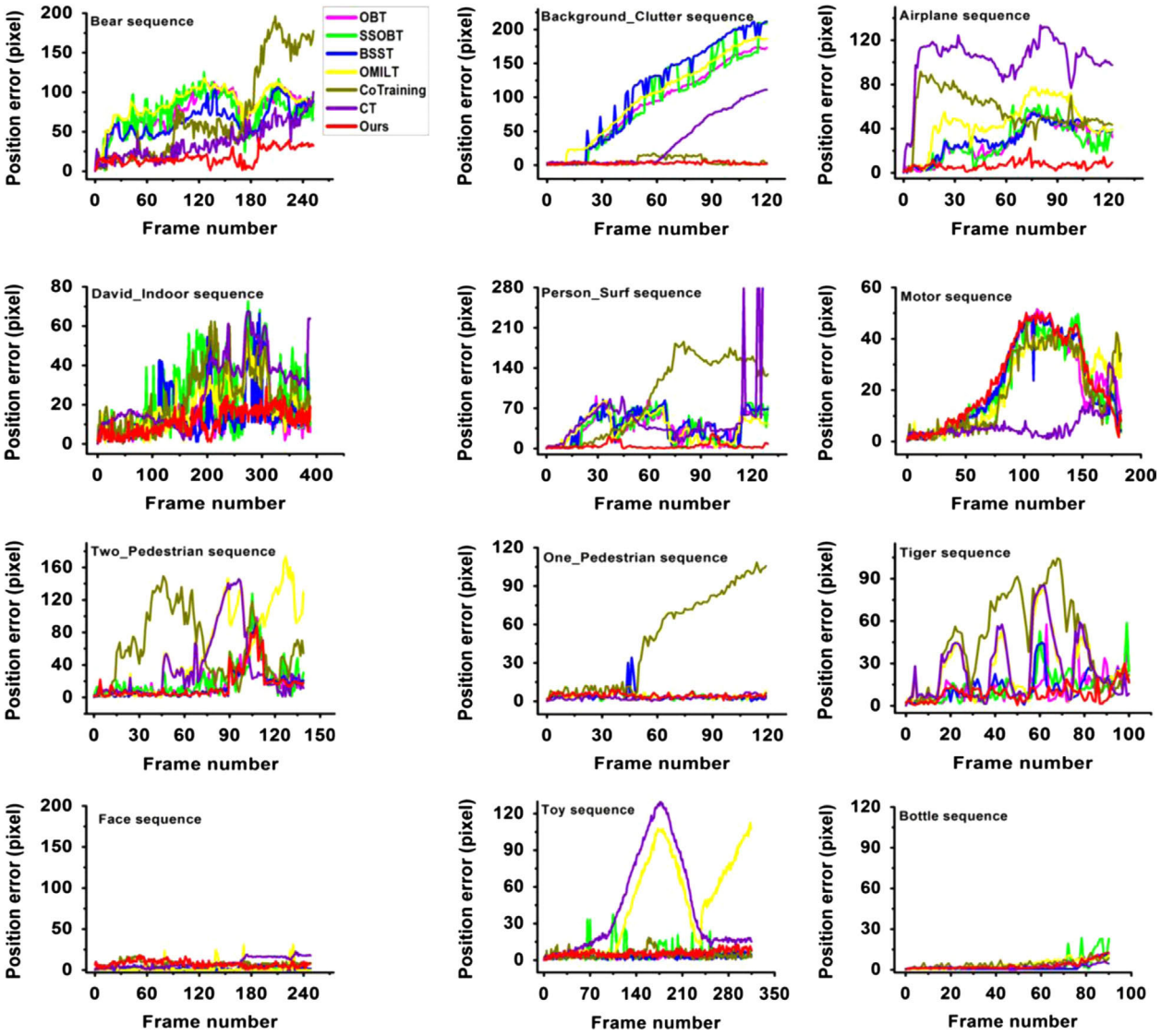

Fig. 8. The position error curves for the twelve sequences.

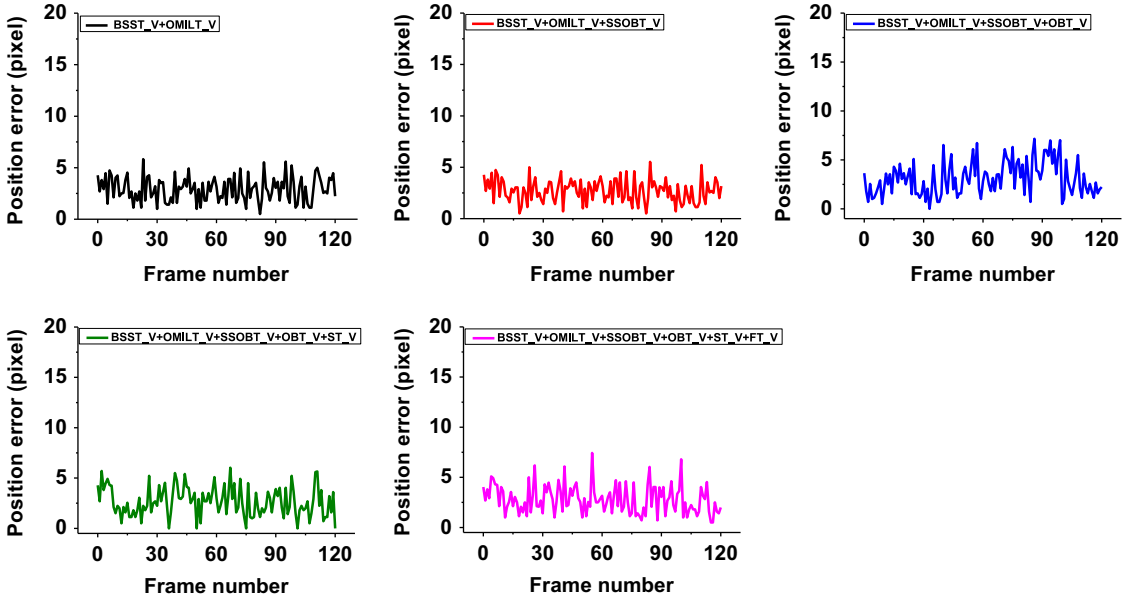

Fig. 9. The position error curves for different combinations of multiple imperfect oracles for the *Back_Clutter* sequence. The oracle set is increased in a sequential way.

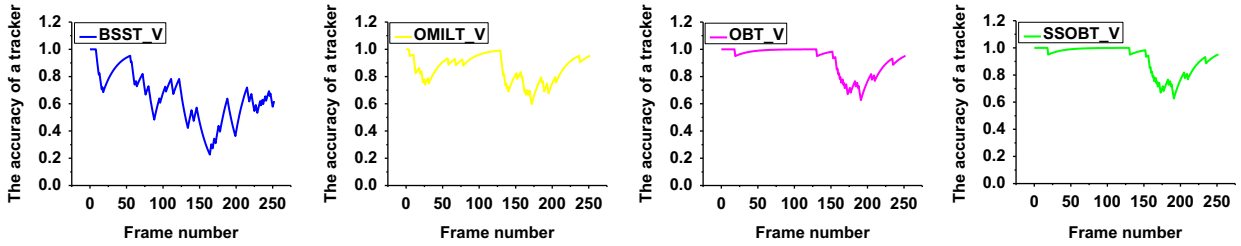

Fig. 10. The evolving curves of the accuracies obtained by the **OBT_V**, **SSOBT_V**, **BSST_V** and **OMILT_V** for the *Bear* sequence.

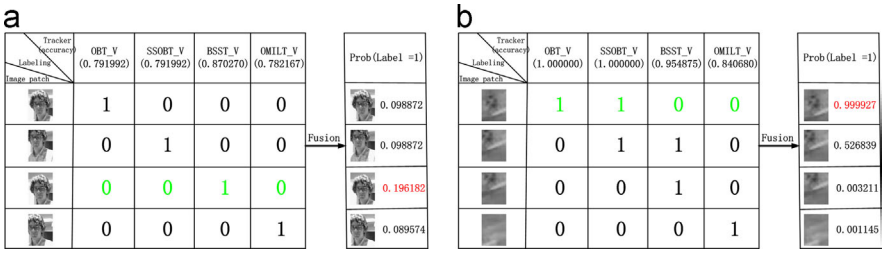

Fig. 11. Illustration of the fusion process. Red indicates the highest probability belonging to positive sample. (a) Frame #189 from the *David_Indoor* sequence. (b) Frame #31 from the *Person_Surf* sequence. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

After fusion, we can still give the correct labeling (i.e., 1) to the image patch.

5.6. Simulated dummy trackers

In the following experiments, we analyze the robustness of the proposed method to dummy trackers which report random target positions (i.e., very incompetent oracles). Specifically, we are interested in several aspects of the method: Is the expertise of a dummy tracker reduced as a function of time? What happens if several such incompetent oracles are introduced?

We perform a number of experiments using imperfect oracles (i.e., the **OBT_V**, **SSOBT_V**, **BSST_V**, and **OMILT_V** trackers) and dummy trackers which report random target positions within the search window. The measurements are made for several image sequences. Due to randomness in our stimulation, the tracking results are similar but not exactly the same for one particular image sequence. The visualized stimulating results for the *Bear* sequence are shown in Fig. 12, in which the red, magenta, green, blue, yellow and darkgreen rectangles represent the tracking results obtained by our tracker, **OBT_V**, **SSOBT_V**, **BSST_V**, **OMILT_V** and the dummy tracker respectively. From the top row to the bottom row in Fig. 12, the number of different combinations of the dummy trackers is 1, 3, 5, 8 and 10 respectively.

We check the expertise of different combinations of the dummy trackers over time and give the results in Fig. 13. As expected, the expertise of most dummy trackers is reduced with time for a variety of different combinations. In some time instances, the expertise of a dummy tracker is increased with time due to the dummy tracker reporting the correct target positions in that period. This is in consistence with the fact that our simulation of the dummy trackers is totally random. Overall, the expertise of a dummy tracker is generally reduced with time on the whole sequence. As shown in Fig. 14, the performance of our method is robust to the dummy trackers. When the dummy trackers are introduced, our method can evaluate online that some trackers are purposely giving adversarial target positions and automatically weaken their influence on inference.

In conclusion, the experiments using dummy trackers verify that our method allow a large number of imperfect trackers and even dummy trackers to cooperate to reach robust decisions by simultaneously inferring the most likely object position and the accuracy of each imperfect oracle.

5.7. Perturbations of the current solution

Finally, one requirement of the proposed method is that there must exist at least one tracker that produces an accurate enough suggestion, thus one would naturally ask for an experimental evaluation of the limitations of the method. When does the proposed method break down? What happens if, especially at the beginning of the tracking process, a consensus is based on a wrong solution? Will the proposed method recover, or may it drift the solution far from the correct one?

To evaluate the performance of the proposed method under these conditions, we perform a set of experiments in which the tracking positions are artificially perturbed at the beginning of tracking. More specifically, our experimental settings are as follows: the proposed method uses the **OBT_V**, **SSOBT_V**, **BSST_V**, and **OMILT_V** trackers. Consider the state of an object is represented by $s=\{x,y\}$, where x and y denote the horizontal and vertical locations of the object respectively. The object of interest is manually selected in the first frame. For the video frame at time t, we first obtain a solution $s_t=\{x_t,y_t\}$ that is estimated by our method. Then, the current solution is artificially perturbed by changing its state from $s_t=\{x_t,y_t\}$ to $s_t'=\{x_t-d,y_t-d\}$, where d denotes the bias to the correct location of the object. After the perturbation of the current solution, we take the image patch of the permuted position as a positive sample to update the target appearance model for each imperfect tracker (i.e., **OBT_V**, **SSOBT_V**, **BSST_V**, and **OMILT_V**) if it is an appearance-adaptive tracker. The perturbation procedure continues for N frames. After N frames, the tracking procedure continues without perturbation until the end of video. We report the results on the *Airplane* sequence in Figs. 15 and 16.

Fig. 15 shows the results of the perturbation and tracking. Denote **PM_d_N** as a tracker in which the tracking positions are artificially perturbed with d pixels at the first N frames for the *Airplane* sequence. The quantitative comparison results of the proposed tracker and **PM_d_N** are shown in Fig. 16. It can be seen

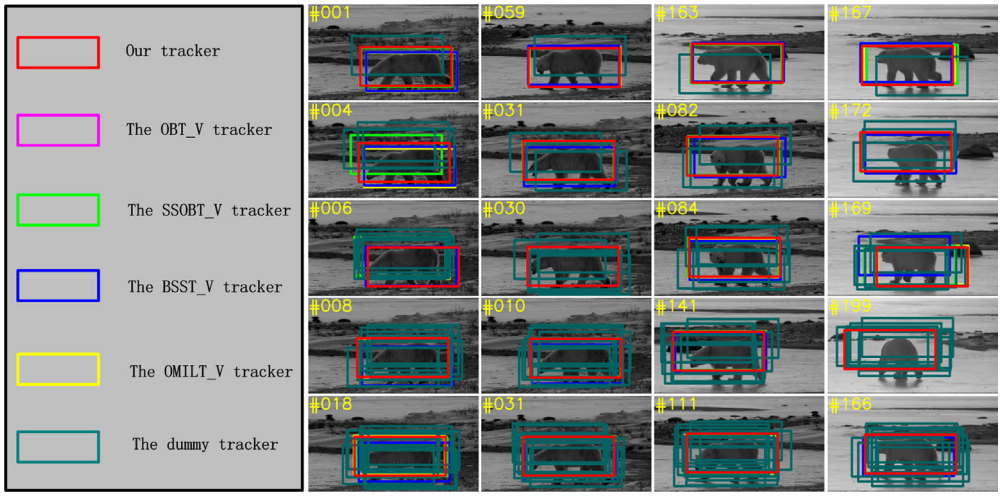

Fig. 12. Visualized illustration of the stimulating process of dummy trackers which yield random target positions on the *Bear* sequence. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

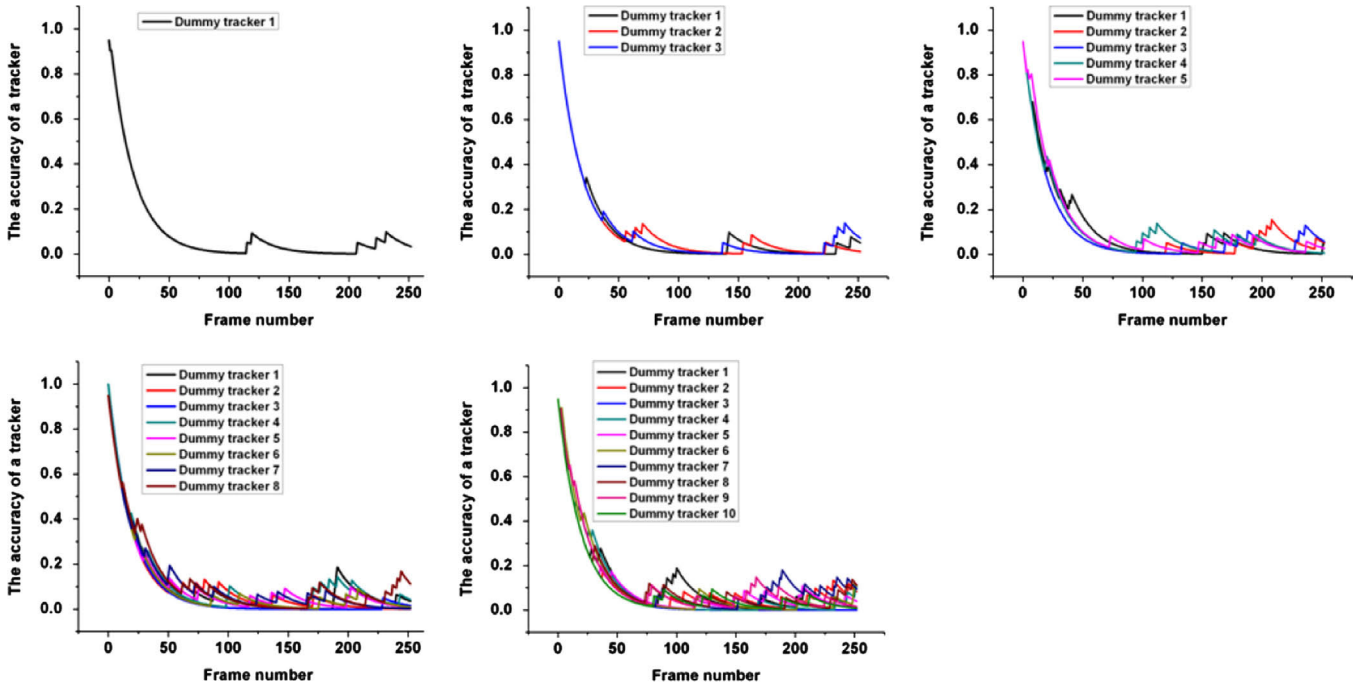

Fig. 13. The evolving curves obtained by using a variety of different combinations of dummy trackers for the *Bear* sequence. The number of different combinations of dummy trackers is 1, 3, 5, 8 and 10 respectively.

that the proposed tracker can still recover the correct positions even in the case that the values of d and N are as large as 10.

Recall that one requirement of the proposed tracker is that there must exist at least one tracker that produces an accurate suggestion in a particular phase. Fig. 17 shows the position error curves of the proposed tracker, **PM_10_10**, **OBT_V_PM_10_10**, **SSOBT_V_PM_10_10**, **BSST_V_PM_10_10** and **OMILT_V_PM_10_10** for the *Airplane* sequences. **PM_10_10** denotes a tracker in which the tracking positions are artificially perturbed with 10 pixels at the first 10 frames for the *Airplane* sequence. **X_V_PM_10_10** denotes a tracker in which the tracker is updated according to the training data obtained during the tracking process of **PM_10_10**. As can be seen, at the beginning the proposed tracker gradually drifts the solution from the correct one due to the artificial perturbation. However, after the disturbing process (e.g., frame #30), **SSOBT_V_PM_10_10** plays an important role in the tracking, because it contains a fixed prior classifier, which can successfully re-detect the object and continue tracking. When more and more appearance changes occur (e.g., frame #90), **SSOBT_V_PM_10_10** is limited by the fact that they cannot accommodate very large changes in appearance. At this phase, the role of **OMILT_V_PM_10_10** increases to provide better object/background separation.

Qualitatively, there are two aspects of the oracle set that are relevant to the success of the proposed tracker, i.e., the quality of each individual oracle and the diversity between different oracles. In our experiments, we have taken into account both aspects when choosing the oracles set, and the proposed tracker works well during the whole tracking process. Moreover, we have also conducted a set of experiments in which the tracking positions are artificially perturbed at the beginning of tracking. Fortunately, we can still solve the perturbation problem under the proposed tracking framework via weakly supervised learning from imperfect oracles, and observe from the results that in such a case the

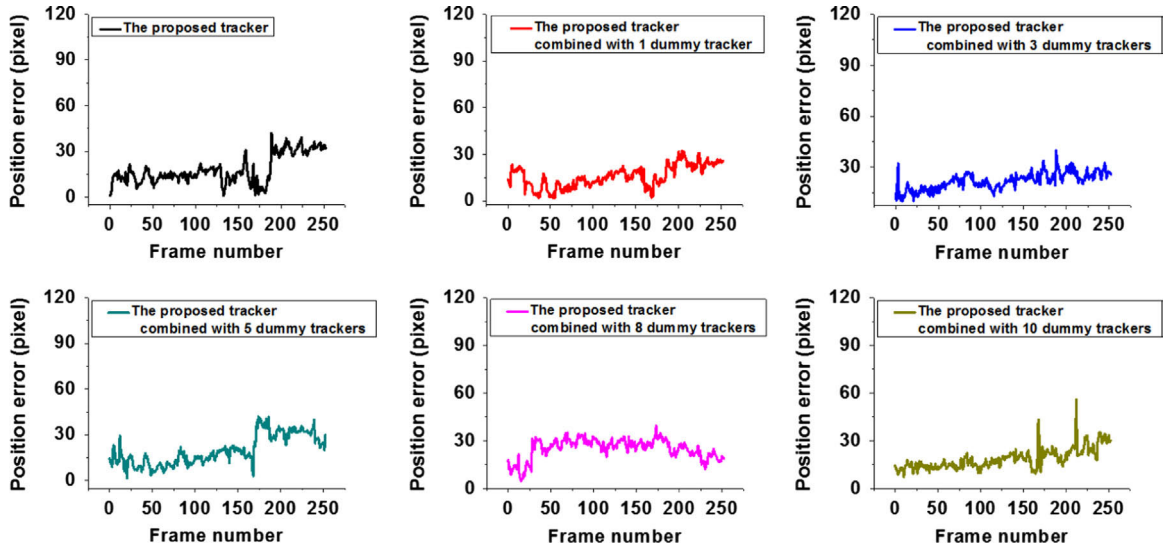

Fig. 14. The position error curves obtained by the proposed tracker and the proposed tracker combined with 1, 3, 5, 8 or 10 dummy trackers respectively for the *Bear* sequences. As the number of dummy trackers increases, the performance of the proposed tracker is robust to the influence of these dummy trackers.

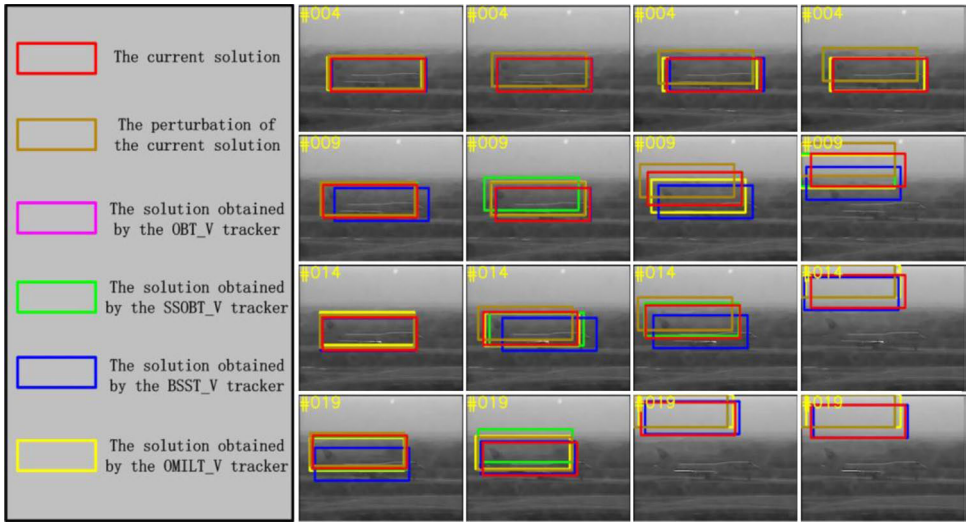

Fig. 15. Results of the perturbation and tracking. The values of the disturbing frame number N and the bias distance d are both varied from 5 to 20 (in increments of 5) from the top row to the bottom row and from the left column to the right column respectively.

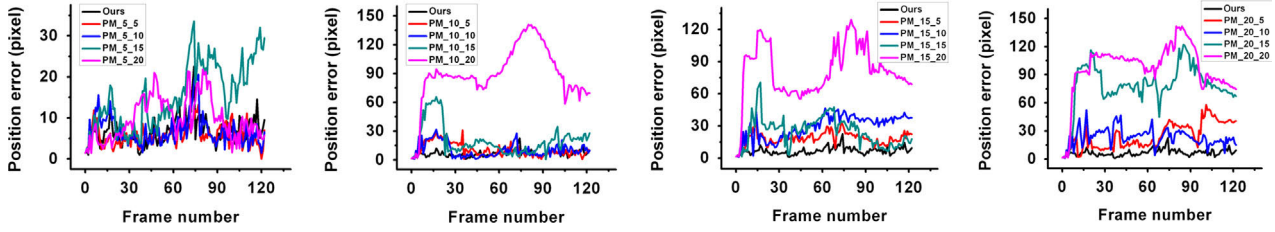

Fig. 16. The position error curves of the proposed tracker and **PM_d_N** for the *Airplane* sequence. **PM_d_N** denotes a tracker in which the tracking positions are artificially permuted with d pixels at the first N frames for the *Airplane* sequence.

proposed tracker can still recover the correct positions within a certain range of perturbation. If the assumptions on the quality of each individual oracle and the diversity between different oracles are invalid, the proposed method would break down.

6. Conclusion

In this paper, we develop a novel visual tracking framework for combining outputs of multiple trackers to improve tracking accuracy. In the proposed framework, visual tracking is considered in the setting of weakly supervised learning where (possibly noisy) labels provided by multiple imperfect trackers can be efficiently used for inference. To construct the proposed weakly supervised tracking framework, we extend the GLAD model [6] to the task of visual tracking and take advantage of sequential data for making real time and accurate inference. An online evaluation strategy is developed to incrementally update the accuracy of each tracker. Meanwhile, the target appearance model of each candidate tracker is updated if it is an appearance-adaptive tracker. We have shown that our tracker can

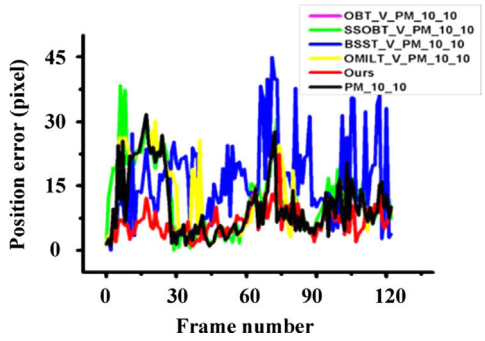

Fig. 17. The position error curves obtained by the proposed tracker, **PM_10_10, OBT_V_PM_10_10, SSOBT_V_PM_10_10**, **BSST_V_PM_10_10** and **OMILT_V_PM_10_10** for the *Airplane* sequences. Please note that the curve obtained by **OBT_V_PM_10_10** is the same as that obtained by **SSOBT_V_PM_10_10** in this example.

work in a wide variety of scenarios, including appearance variations of the object, pose changes, scale changes, illumination changes, cluttered scenes and occlusion. Moreover, extensive experimental results show that the proposed method can obtain more accurate data labels or results than a single tracker. Furthermore, the proposed method is robust to noisy labeling (i.e., dummy trackers) by online evaluation of the trackers. The estimated accuracy of a tracker is shown to be a reliable confidence measure that can be used to infer in the GLAD model to make the inference more accurate and improve the noisy labeling. The proposed method is also demonstrated to be tolerant to some perturbations of the correct solution. In summary, we propose a scalable and off-the-shelf tracking framework, in which the advantages of multiple complementary trackers can be seamlessly combined to achieve robust tracking results by simultaneously inferring both the most likely object position and the accuracy of each tracker in the absence of ground truth, which is usually the case in real-world tracking applications.

From the promising results obtained by the proposed method, we expect that weakly supervised learning from multiple imperfect oracles can lead to competitive solutions to a large variety of problems in computer vision and pattern recognition.

7. Future work

Next, we would like to discuss several general issues to improve our weakly supervised learning-based tracking framework.

(1) One direction for future research is to generate the conditional probability of interest (i.e., Eq. (1)) by using more effective methods. Now, we adopt Eq. (1) as the modeling strategy because it has the intuitive meaning of the relationship between the conditional probability and the expertise of a tracker (or the labeling difficulty of an image).
(2) Another possible direction is to extend the proposed method to take into account trackers whose outputs are confidence values or probabilistic density functions. One strategy is that we could straightforwardly incorporate these values into the formula of image patch difficulty.
(3) The third potentially idea is to generate online candidate trackers by taking into account the current solution and its "difficult" parts (like Adaboost, in which next weak classifier is picked by observing where the current strong classifier fails to make good prediction). The open issues are that how to define the meaning of a "difficult" part, and how to ensure that there is enough diversity in the candidate trackers.
(4) The fourth possible direction is to adopt a CRF model to consider both affinities and dependencies among the trackers to improve the learning of important dependencies among oracles.

Conflict of interest statement

None declared.

Acknowledgment

This work is supported by the Natural Science Foundation of China (Nos. 61202299 and 61170179), the China Postdoctoral Science Foundation (No. 2011M501081), Fundamental Research Funds for the Central Universities (No. JB-ZR1219), Scientific Research Foundation of Huaqiao University (No. 11BS109), and Xiamen Science & Technology Planning Project (No. 3502Z20116005) of China, Natural Science Foundation of Fujian Province (No. 2013J05092).

References

[1] L. Ahn, B. Maurer, C. McMillen, D. Abraham, M. Blum, reCAPTCHA: human-based character recognition via web security measures, Science (2008).
[2] R. Snow, B. Connor, D. Jurafsky, A. Ng, Cheap and Fast-but is it Good? Evaluating Non-expert Annotations for Natural Language Tasks, in: Proceedings of EMNLP, 2008.
[3] P. Donmez, J. Carbonell, Proactive Learning: Cost-Sensitive Active Learning with Multiple Imperfect Oracles, in: Proceedings of CIKM, 2008.
[4] V. Raykar, S. Yu, L. Zhao, A. Jerebko, C. Florin, G. Valadez, L. Bogoni, L. Moy, Supervised Learning from Multiple Experts: Whom to trust when everyone lies a bit, in: Proceedings of ICML, 2009.
[5] O. Dekel, O. Shamir, Good Learners for Evil Teachers, in: Proceedings of ICML, 2009.
[6] J. Whitehill, P. Ruvolo, J. Bergsma, T. Wu, J. Movellan, Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise, in: Proceedings of NIPS, 2009.
[7] Amazon. Mechanical Turk. ⟨http://www.mturk.com⟩.
[8] H. Wu, A. Sankaranarayanan, R. Chellappa, Online Empirical Evaluation of Tracking Algorithms, in: Proceedings of TPAMI, 2010.
[9] Z. Kalal, K. Mikolajczyk, J. Matas, Tracking–Learning–Detection, in: Proceedings of TPAMI, 2012.
[10] B. Zhong, H. Yao, S. Chen, R. Ji, X. Yuan, S. Liu, W. Gao, Visual Tracking via Weakly Supervised Learning from Multiple Imperfect Oracles, in: Proceedings of CVPR, 2010.
[11] J. Lim, D. Ross, R. Lin, M. Yang, Incremental Learning for Visual Tracking, in: Proceedings of NIPS, 2004.
[12] R. Collins, Y. Liu, M. Leordeanu, Online Selection of Discriminative Tracking Features, in: Proceedings of TPAMI, 2005.
[13] B. Han, L. Davis, On-Line Density-Based Appearance Modeling for Object Tracking, in: Proceedings of ICCV, 2005.
[14] H. Grabner, H. Bischof, On-line Boosting and Vision, in: Proceedings of CVPR, 2006.
[15] S. Avidan, Ensemble Tracking, in: Proceedings of TPAMI, 2007.
[16] M. Isard, A. Blake, CONDENSATION-Conditional Density Propagation for Visual Tracking, in: Proceedings of IJCV, 1998.
[17] D. Comaniciu, V. Ramesh, P. Meer., Kernel-Based Object Tracking, in: Proceedings of TPAMI, 2003.
[18] A. Adam, E. Rivlin, I. Shimshoni, Robust Fragments-based Tracking using the Integral Histogram, in: Proceedings of CVPR, 2006.
[19] I. Matthews, T. Ishikawa, S. Baker, The Template Update Problem, in: Proceedings of TPAMI, 2004.
[20] M. Yang, J. Yuan, Y. Wu, Spatial Selection for Attentional Visual Tracking, in: Proceedings of CVPR, 2007.
[21] H. Grabner, C. Leistner, H. Bischof, Semi-Supervised On-line Boosting for Robust Tracking, in: Proceedings of ECCV, 2008.
[22] S. Stalder, H. Grabner, L. Van Gool, Beyond Semi-Supervised Tracking: Tracking Should Be as Simple as Detection, but not Simpler than Recognition, in: Proceedings of ICCV 2009 Workshop on On-line Learning for Computer Vision.
[23] B. Babenko, M. Yang, S. Belongie, Robust Object Tracking with Online Multiple Instance Learning, in: Proceedings of TPAMI, 2011.
[24] F. Tang, S. Brennan, Q. Zhao, H. Tao, Co-Tracking Using Semi-Supervised Support Vector Machines, in: Proceedings of ICCV, 2007.
[25] Q. Yu, T. Dinh, G. Medioni, Online Tracking and Reacquisition Using Co-trained Generative and Discriminative Trackers, in: Proceedings of ECCV, 2008.
[26] H.C. Lu, Q.H. Zhou, D. Wang, R. Xiang, A Co-training Framework for Visual Tracking with Multiple Instance Learning, in: Proceedings of FG, 2011.
[27] X.H. Lou, F.A. Hamprecht, Structured Learning for Cell Tracking, in: Proceedings of NIPS, 2011.
[28] X.H. Lou, F.A. Hamprecht, Structured Learning from Partial Annotations, in: Proceedings of ICML, 2012.
[29] J.G. Wang, E. Sung, W.Y. Yau, Active Learning for Solving the Incomplete Data Problem in Facial Age Classification by the Furthest Nearest-neighbor Criterion, in: Proceedings of TIP, 2011.
[30] C. Vondrick, D. Ramanan, Video Annotation and Tracking with Active Learning, in: Proceedings of NIPS, 2011.

[31] R. Yao, Q.F. Shi, C.H. Shen, Y.N. Zhang, A. Hengel, Robust Tracking with Weighted Online Structured Learning, in: Proceedings of ECCV, 2012.
[32] B. Yang, C. Huang, R. Nevatia, Learning Affinities and Dependencies for Multi-Target Tracking using a CRF Model, in: Proceedings of CVPR, 2011.
[33] D.N. Vizireanu, Generalizations of Binary Morphological Shape Decomposition, J. Electron. Imaging (2007).
[34] D.N. Vizireanu, R.M. Udrea, Visual-oriented morphological foreground content grayscale frames interpolation method, J. Electron. Imaging (2009).
[35] R.M Udrea, D.N. Vizireanu, Iterative Generalization of Morphological Skeleton, J. Electron. Imaging (2007).
[36] X. Ren, J. Malik, Tracking as Repeated Figure/Ground Segmentation, in: Proceedings of CVPR, 2007.
[37] Z. Yin, R. Collins, Shape Constrained Figure-Ground Segmentation and Tracking, in: Proceedings of CVPR, 2009.
[38] J. Fan, X. Shen, Y. Wu, Scribble Tracker: A Matting-based Approach for Robust Tracking, in: Proceedings of TPAMI, 2012.
[39] S. Wang, H.C. Lu, F. Yang, M.H. Yang, Superpixel Tracking, in: Proceedings of ICCV, 2011.
[40] S. Hare, A. Saffari, P. Torr, Efficient Online Structured Output Learning for Keypoint-Based Object Tracking, in: Proceedings of CVPR, 2012.
[41] W. He, T. Yamashita, H. Lu, S. Lao, SURF Tracking, in: Proceedings of ICCV, 2009.
[42] S. Birchfield, Elliptical Head Tracking using Intensity Gradients and Color Histograms, in: Proceedings of CVPR, 1998.
[43] Y. Wu, T.S. Huang, Robust Visual Tracking by Integrating Multiple Cues based on Co-inference Learning, in: Proceedings of IJCV, 2004.
[44] F. Moreno-Noguer, A. Sanfeli, D. Samaras, Dependent Multiple Cue Integration for Robust Tracking, in: Proceedings of TPAMI, 2001.
[45] M. Yang, F. Lv, W. Xu, Y. Gong, Detection Driven Adaptive Multi-cue Integration for Multiple Human Tracking, in: Proceedings of ICCV, 2009.
[46] B. Stenger, T. Woodley, R. Cipolla, Learning to Track with Multiple Observers, in: Proceedings of CVPR, 2009.
[47] J. Santner, C. Leistner, A. Saffari, T. Pock, H. Bischof, PROST: Parallel Robust Online Simple Tracking, in: Proceedings of CVPR, 2010.
[48] J. Kwon, K. Lee, Tracking by Sampling Trackers, in: Proceedings of ICCV, 2011.
[49] Y.R Wang, X.L. Tang, Q. Cui, Dynamic Appearance Model for Particle Filter Based Visual Tracking, in: Proceedings of PR, 2012.
[50] E. Erdem, S. Dubuisson, I. Bloch, Visual Tracking by Fusing Multiple Cues With Context-sensitive Reliabilities, in: Proceedings of PR, 2012.
[51] L. Lu, G. Hager, A Nonparametric Treatment for Location/Segmentation Based Visual Tracking, in: Proceedings of CVPR, 2007.
[52] S. Oron, A.B. Hillel, D. Levi, S. Avidan, Locally Orderless Tracking, in: Proceedings of CVPR, 2012.
[53] N. Jiang, W. Liu, Y. Wu, Learning Adaptive Metric for Robust Visual Tracking, in: Proceedings of TIP, 2011.
[54] T.X. Bai, Y.F. Li, Robust Visual Tracking with Structured Sparse Representation Appearance Model, in: Proceedings of PR, 2012.
[55] X. Mei, H. Ling, Y. Wu, E. Blasch, L. Bai, Minimum Error Bounded Efficient L1 Tracker with Occlusion Detection, in: Proceedings of CVPR, 2011.
[56] K.H. Zhang, L. Zhang, M.H. Yang Real-time, Compressive Tracking, in: Proceedings of ECCV, 2012.
[57] A. Yilmaz, O. Javed, M. Shah, Object tracking: a survey, ACM Comput. Surv. (2006).
[58] ⟨http://www.cs.technion.ac.il/~amita/fragtrack/fragtrack.htm⟩.
[59] ⟨http://www.vision.ee.ethz.ch/boostingTrackers/index.htm⟩.
[60] ⟨http://vision.ucsd.edu/~bbabenko/project_miltrack.shtml⟩.
[61] ⟨http://ice.dlut.edu.cn/lu/publications.html⟩.
[62] ⟨http://www4.comp.polyu.edu.hk/~cslzhang/CT/CT.htm⟩.
[63] J. Fan, Y. Wu, S. Dai, Discriminative Spatial Attention for Robust Tracking, in: Proceedings of ECCV, 2010.

Bineng Zhong received the B.S., M.S., and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 2004, 2006, and 2010, respectively. From 2007 to 2008, he was a Research Fellow with the Institute of Automation and Institute of Computing Technology, Chinese Academy of Science. Currently, he is a Lecturer with the School of Computer Science and Technology, Huaqiao University, Xiamen, China, and he is also a Post-Doc with the School of Information Science and Technology, Xiamen University, Xiamen, China. His current research interests include pattern recognition, machine learning, and computer vision.

Hongxun Yao received the B.S. and M.S. degrees in electronic engineering from Harbin Engineering University, Harbin, China, in 1987 and 1990, respectively, and the Ph.D. degree in computer science from Harbin Institute of Technology, Harbin, China, in 2003. Currently, she is a Professor with the Harbin Institute of Technology. Her current research interests include computer vision image retrieval, and pattern recognition.

Sheng Chen received the B.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 2010. He is currently pursuing the Ph.D. degree from the Oregon State University, USA. His current research interests include pattern recognition, machine learning, computer vision, and intelligence video surveillance.

Rongrong Ji received the B.S. degree in electronic engineering from Harbin Engineering University, Harbin, China, in 2005, and the Ph.D. degree in computer science from Harbin Institute of Technology, Harbin, China, in 2011. Currently, he is a Post-Doc with the Columbia University, USA. His current research interests include computer vision image retrieval, and pattern recognition.

Tat-Jun Chin is a Lecturer at the School of Computer Science, The University of Adelaide. His current research interests include computer vision, object tracking, detection and recognition, robust geometric model fitting, statistical learning techniques in vision, multiple view geometry.

Hanzi Wang is currently a "MinJiang" Distinguished Professor at Xiamen University, China and an Adjunct Professor at the University of Adelaide, Australia. He was a Senior Research Fellow (2008–2010) working with Prof. David Suter, at the University of Adelaide, Australia; an Assistant Research Scientist (2007–2008) and a Postdoctoral Fellow (2006–2007) working with Prof. Gregory D. Hager, at the Johns Hopkins University; and a Research Fellow working with Prof. David Suter, at Monash University, Australia (2004–2006). He received the Ph.D. degree in Computer Vision from Monash University. He was awarded the Douglas Lampard Electrical Engineering Research Prize and Medal for the best PhD thesis in the Department. His research interests are concentrated on computer vision and pattern recognition including visual tracking, robust statistics, object detection, video segmentation, model fitting, optical flow calculation, 3D structure from motion, image segmentation and related fields. He is a Senior Member of the IEEE. He is an Associate Editor for IEEE Transactions on Circuits and Systems for Video Technology (T-CSVT) and he was a Guest Editor of Pattern Recognition Letters (September 2009).