SEMI-SUPERVISED ENSEMBLE TRACKING

Huaping Liu and Fuchun Sun

Department of Computer Science and Technology, Tsinghua University, Beijing, P.R.China
State Key Lab. of Intelligent Technology and Systems, P.R.China

ABSTRACT

In this paper, we propose a semi-supervised ensemble tracking approach under the framework of particle filter. The particle filter is used not only for object searching, but also for unlabelled sample generation. By adopting the semi-supervised learning technology, these unlabelled samples which are generated online are utilized to progressively modify the classifier and make the ensemble tracker to be more robust to environment changing. On the other hand, utilizing semi-supervised learning technology can avoid the drifting phenomenons which are often encountered when using supervised learning. Finally, the performance of the proposed approach is evaluated using real visual tracking examples.

Index Terms— Semi-supervised learning, ensemble tracking, visual tracking

1. INTRODUCTION

Recently many scholars formulated the tracking as a binary classification problem, i.e., the core task of tracking is to separate the object from background in each video. [2] firstly proposed a method to adaptively select color features that best discriminate the object from the current background. Another important work is [1], which used an adaptive ensemble of classifier. [3] designed an on-line boosting classifier that selects features to discriminate the object from the background. [5] incorporated this approach into the framework of particle filter. These "classification-based tracking" approaches attract a lot of attentions. However, we notice that the model is updated in a totally supervised manner. That is to say, the classifier which is trained (or updated) in the previous frame is used in current frame to evaluate possible regions. Then we select the so-called "positive" or "negative" samples for updating the classifier. Note that the "positive" or "negative" samples are not manfully labelled but labelled by the previously trained classifier (This is an important difference between tracking and detection problems). Since tracking may introduce error, the labels may be noisy. Therefore these supervised approaches usually tends to "drift" since the error may be accumulated during the learning and tracking process. In fact, in many tracking problems, the labelled samples are given by an extra detector which only works in the first frame and therefore the number of labelled samples is very small, while the unlabelled samples, which can be selected from any frame, is enormous and easy to get. If we wish to update the classifier online, we should not ignore the unlabelled samples. This motivates us to use the popular semi-supervised learning approach[9].

Semi-supervised learning has received a lot of attentions over the past few years. The main motivation is that labelled samples are difficult to obtain, whereas unlabelled ones are easy. The task of semi-supervised learning algorithms is to utilize labelled samples in conjunction with their relationship to unlabelled data to design a classifier. Currently, different algorithms have been proposed for semi-supervised learning such as EM algorithm, co-training, tri-training, etc. For more details on semi-supervised learning, please see [9].

Though the semi-supervised learning achieves great successes, its application in tracking domain is still very rare. Recently, [8] utilized the co-training SVM approach to design a semi-supervised tracker. A demerit of this approach is that the tracker needs several initial frames to get enough labelled samples. In tracking scenarios, extracting feature from the first frame only is more attractive. In [4], a semi-supervised online boosting approach is used for tracking, which is a straightforward extension of the supervised online boosting approach[3]. In our recent works, the semi-supervised learning is incorporated into the framework of particle filter[6]. However, [6] does not exploit the temporal relation between video frames. In this paper, we will use the idea of [1] to enhance the semi-supervised tracking. The presented tracking algorithm is developed under the framework of particle filter. The particle filter is used not only for object searching, but also for unlabelled sample generation. By adopting the semi-supervised learning technology, these unlabelled samples which are generated online are utilized to progressively modify the classier and make the ensemble tracker to be more robust to environment changing.

2. BRIEF REVIEW OF PARTICLE FILTER

The task of tracking is to use the available measurement information to estimate the hidden state variables. Given the available observations $\mathbf{z}_{1:k-1} = \mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_{k-1}$ up to time instant $k-1$, the prediction stage utilizes the probabilistic

system transition model $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ to predict the posterior as $p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})d\mathbf{x}_{k-1}$. At time instant k, the state can be updated using $Bayes's$ rule $p(\mathbf{x}_k|\mathbf{z}_{1:k}) = p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{z}_{1:k-1})/p(\mathbf{z}_k|\mathbf{z}_{1:k-1})$, where $p(\mathbf{z}_k|\mathbf{x}_k)$ is described by the observation equation. The kernel of particle filter is to recursively approximate the posterior distribution using a finite set of weighted samples. Each sample $\mathbf{x}_k^i$ represents one hypothetical state of the object, with a corresponding discrete sampling probability ω_k^i, which satisfies $\sum_{i=1}^N \omega_k^i = 1$. The posterior $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ then can be approximated as $p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \sum_{i=1}^N \omega_k^i \delta(\mathbf{x}_k - \mathbf{x}_k^i)$, where $\delta(\cdot)$ is Dirac function. Then the estimation of the state $\mathbf{x}_k$ can be obtained as $\hat{\mathbf{x}}_k = \sum_{i=1}^N \omega_k^i \mathbf{x}_k^i$. The candidate samples $\{\mathbf{x}_k^i\}_{i=1,2,\cdots,N}$ are drawn from an importance distribution $q(\mathbf{x}_k|\mathbf{x}_{1:k-1}, \mathbf{z}_{1:k})$ and the weight of the samples are $\omega_k^i = \omega_{k-1}^i \frac{p(\mathbf{z}_k|\mathbf{x}_k^i)p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k|\mathbf{x}_{1:k-1},\mathbf{z}_{1:k})}$. The samples are re-sampled to generate an unweighed particle set according to their importance weights to avoid degeneracy. In many cases, $q(\mathbf{x}_k|\mathbf{x}_{1:k-1}, \mathbf{z}_{1:k})$ is set to be $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ and the weights therefore become proportional to the observation likelihood $p(\mathbf{z}_k|\mathbf{x}_k)$.

3. SEMI-SUPERVISED ENSEMBLE TRACKING

In visual tracking scenarios, the likelihood function is usually time-varying and is difficult to get. A natural approach is to learn it online. There exists a difficulty that during tracking period, it is difficult to collect "positive" samples. Some recently proposed approach use the tracking results to extract positive samples and therefore supervised learning technology can be utilized. However, these positive samples, which are not manually-labelled, are not reliable. Once some false positive samples are used to learn the likelihood function, the tracking will tend to drifting away. Semi-supervised learning technology, which train a classifier from some labelled samples and a lot of unlabelled samples, seems as a reasonable approach to tackle this problem.

In this paper, we call the the recently proposed SemiBoost approach[7] for semi-supervised learning. Consider a dataset $\{\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_n\}$ and the corresponding label $\{y_1, y_2, \cdots, y_n\}$. The label y_i takes value from the set $\{+1, 0, -1\}$, where $+1$, -1 and 0 represent positive, negative and unlabelled labels, respectively. For conveniences, we denote $\mathcal{F}_l$ as the index set of labelled samples, and $\mathcal{F}_u$ as the index set of unlabelled samples, i.e., $\mathcal{F}_l = \{i|y_i \neq 0\}$ and $\mathcal{F}_u = \{i|y_i = 0\}$. More specifically, we denote $\mathcal{F}_l^+ = \{i|y_i = +1\}$ and $\mathcal{F}_l^- = \{i|y_i = -1\}$.

Let $S = [S_{ij}]_{n\times n}$ denote the symmetric similarity matrix, where S_{ij} represents the similarity between samples $\mathbf{f}_i$ and $\mathbf{f}_j$. A natural choice of S_{ij} is

$$S_{ij} = \exp(-||\mathbf{f}_i - \mathbf{f}_j||_2^2/\sigma^2) \tag{1}$$

where $||\cdot||_2$ is the 2-norm of vector and σ is the scale parameter controlling the spread of the radial basis function. The goal of semi-supervised learning is to use the labelled samples, unlabelled samples, and the pairwise similarity S to construct a robust classifier.

To exploit the unlabelled samples, two criteria can be utilized: (1) unlabelled samples with high similarity should share the same label;(2)unlabelled samples which are highly similarly to some labelled sample should share its label. To this end, we resort the problem as searching a classifier $H(\cdot)$ to solve the following constrained optimization problem

$$\begin{aligned} \min \quad & \sum_{i\in\mathcal{F}_l}\sum_{j\in\mathcal{F}_u} S_{ij}e^{-2y_iH(\mathbf{f}_j)} + C\sum_{i,j\in\mathcal{F}_u} S_{ij}e^{H(\mathbf{f}_i)-H(\mathbf{f}_j)} \\ \text{s.t.} \quad & H(\mathbf{f}_i) = y_i \quad \text{for } i \in \mathcal{F}_l \end{aligned} \tag{2}$$

where C is the ratio of the number of labelled samples to the number of unlabelled samples. In our setting, $H(\cdot)$ is an ensemble classifier which can be specifically represented as

$$H(\mathbf{f}_i) = \sum_{t=1}^{T} \alpha_t h_t(\mathbf{f}_i) \tag{3}$$

where $h_t(\cdot)$ is a binary weak classifier which takes value from $\{$+1,-1$\}$, α_t is the corresponding weight of classifier, and T is the iteration number. For any sample $\mathbf{f}$, the practical classifier output is $\text{sign}(H(\mathbf{f}))$, and the likelihood function can be determined as $1/(1 + e^{-2H(\mathbf{f})})$.

In this paper, RGB and Edge Orientation (EO) histograms are used for feature representation. RGB color distributions are used as object models as they achieve robustness against non-rigidity, rotation and partial occlusion and EO information provides more robust feature under the complex environment. In our experiments, RGB histogram is typically calculated in the RGB space using $8 \times 8 \times 8 = 512$ bins; and EO histogram is divided into 20 bins. The two histograms are concatenated into a $512 + 20 = 532$ dimensional feature vector, where the first 512 dimensions correspond to RGB feature and the last 20 dimensions correspond to EO feature.

Given sample vector $\mathbf{f}_i$, the t-th weak classifier is designed according to stump decision,

$$h_t(\mathbf{f}_i) = \begin{cases} 1 & \text{if } s_t f_{i,l_t} < s_t\theta_t \\ -1 & \text{otherwise} \end{cases} \tag{4}$$

where l_t is the selected dimension for $h_t(\cdot)$ (i.e., f_{i,l_t} denotes the l_t-th component of vector $\mathbf{f}_i$. Obviously, l_t is an integer between 1 and 532), $s_t \in \{+1, -1\}$ is the polarity which controls the direction of inequality and θ_t is the threshold. In summary, the weak classifier $h_t(\cdot)$ can be characterized by l_t, s_t and θ_t and therefore we can denote it as $h_t(\mathbf{f}_i; l_t, s_t, \theta_t)$. In some cases, we can remove the subscript t to get $h(\mathbf{f}_i; l, s, \theta)$ for simplified representation.

During the iteration, the goal is to find a new weak classifier $h(\cdot)$ and the corresponding weight α that can efficiently minimize the objective function (2). This leads to the following alternative optimization problem.

$$\min_{h(\cdot),\alpha} \sum_{i\in\mathcal{F}_l}\sum_{j\in\mathcal{F}_u} S_{ij}e^{-2y_i(H(\mathbf{f}_j)+\alpha h(\mathbf{f}_j))} + C\sum_{i,j\in\mathcal{F}_u} S_{ij}e^{H(\mathbf{f}_i)-H(\mathbf{f}_j)}e^{\alpha(h(\mathbf{f}_i)-h(\mathbf{f}_j))} \quad (5)$$

$$\text{s.t.} \quad h(\mathbf{f}_i) = y_i \quad \text{for } i \in \mathcal{F}_l.$$

This expression involves products of variables α and $h(\cdot)$, making it nonlinear and hence difficult to optimize. To tackle this problem, [7] proposed an approach to optimize the upper bound of the objective function. The core is to estimate the confidence of unlabelled sample $\mathbf{f}_i$ to be classified as positive as

$$p_i = \sum_{j\in\mathcal{F}_l^+} S_{ij}e^{-2H(\mathbf{f}_i)} + \frac{C}{2}\sum_{j\in\mathcal{F}_u} S_{ij}e^{H(\mathbf{f}_j)-H(\mathbf{f}_i)} \quad (6)$$

and estimate the confidence of unlabelled sample $\mathbf{f}_i$ to be classified as negative as

$$q_i = \sum_{j\in\mathcal{F}_l^-} S_{ij}e^{2H(\mathbf{f}_i)} + \frac{C}{2}\sum_{j\in\mathcal{F}_u} S_{ij}e^{H(\mathbf{f}_i)-H(\mathbf{f}_j)}. \quad (7)$$

Then the pseudo label of unlabelled sample $\mathbf{f}_i$ can be estimated as $z_i = \text{sign}(p_i - q_i)$ and the weight of the sample is $|p_i - q_i|$. After obtaining the pseudo label and the weight of sample, we can use Adaboost algorithm to get the weight of the selected weak classifier. Note that at the first frame, we can extract positive samples from the detection results, and the negative samples from around the detection results. The unlabelled samples are extracted from the particle generation process, which is very straightforward.

The original SemiBoost approach presented in [7] and the tracking approach presented in [6] do not utilize the temporal information. That is to say, at any time instant, the classifier should be totally re-designed. This is not expected in tracking domain. Motivate by the works in [1], we borrow the ensemble tracking idea to design the sequential classifier. Assume that we have a classifier at time instant $k-1$: $H_{k-1}(\mathbf{f}) = \sum_{t=1}^{T}\alpha_{k-1,t}h_{k-1,t}(\mathbf{f})$. When we design the classifier at time instant k, the temporal coherence of video is exploited. We keep the K best weak classifiers, discard the remaining $T-K$ weak classifiers, train $T-K$ new weak classifiers on the newly available data, and reconstructed the strong weak classifier.

An important fact is that we can only get labelled samples from the initial frame($k=0$). For other frames, we can only get unlabelled samples and therefore (6) and (7) reduce to

$$p_i = \sum_{j\in\mathcal{F}_u} S_{ij}e^{H(\mathbf{f}_j)-H(\mathbf{f}_i)}, q_i = \sum_{j\in\mathcal{F}_u} S_{ij}e^{H(\mathbf{f}_i)-H(\mathbf{f}_j)}, \quad (8)$$

respectively.

Though there does not exist labelled samples in any instant for $k > 0$, the proposed approach belongs to semi-supervised learning, but not un-supervised learning. The reason is that the manually-labelled information is kept in previous classifier, which can be used as an initial guess of current classifier. The whole algorithm is summarized in Algorithm 1. Note that in the initial frame, some modifications are needed. First, the previous classifier is not required and the so-called "initial classifier" in (9) is modified as $H_k(\mathbf{f}) = 0$. Secondly, when computing p_i and q_i, we should utilize (6-7), but not (8), since there exist labelled samples.

Algorithm 1 On-line semi-supervised learning algorithm

Given: Samples $\{\mathbf{f}_i\}_{i=1}^n$; Previous classifier $H_{k-1}(\mathbf{f}) = \sum_{t=1}^{T}\alpha_{k-1,t}h_{k-1,t}(\mathbf{f})$.
OUTPUT: $H_k(\mathbf{f}) = \sum_{t=1}^{T}\alpha_{k,t}h_{k,t}(\mathbf{f})$

Compute pairwise similarity S_{ij} between any two samples according to (1).
Initialization:
– Sort the $\{\alpha_{k-1,t}\}_{t=1}^n$ in descending order to get $\{\alpha_{k-1,t'}\}_{t'=1}^n$.
– Obtain the initial classifier as

$$H_k(\mathbf{f}) = \sum_{t'=1}^{K}\alpha_{k-1,t'}h_{k-1,t'}(\mathbf{f}). \quad (9)$$

FOR $t = K+1, K+2, \cdots, T$
– Compute p_i and q_i according to (8)
– Compute the pseudo label $z_i = \text{sign}(p_i - q_i)$
– Compute the normalized weight $w_i \propto |p_i - q_i|$.
– Select the best weak classifier with respect to the weight error

$$\epsilon_t = \min_{l,s,\theta}\sum_{i=1}^{n} w_i|h(\mathbf{f}_i; l, s, \theta) - z_i|$$

– Define $h_t(\mathbf{f}) = h(\mathbf{f}, l_t, s_t, \theta_t)$ where l_t, s_t, θ_t are the minimizers of ϵ_t.
– Compute α_t as

$$\alpha_t = \frac{1}{4}\ln\frac{\sum_{i\in\mathcal{F}_u}\{p_i\delta(h_t(\mathbf{f}_i)=1) + q_i\delta(h_t(\mathbf{f}_i)=-1)\}}{\sum_{i\in\mathcal{F}_u}\{p_i\delta(h_t(\mathbf{f}_i)=-1) + q_i\delta(h_t(\mathbf{f}_i)=1)\}}$$

– Update the classifier as $H_k(\mathbf{f}) = H_k(\mathbf{f}) + \alpha_t h_t(\mathbf{f})$.

4. EXPERIMENTAL RESULTS

The proposed approach is tested on color video sequence from OTCBVS dataset collection (http://www.cse.ohio-state.edu/OTCBVS-BENCH). For all of the experiments, the states of the particle filter are defined as $\mathbf{x}_k = [x_k, y_k, s_k]$, where x_k, y_k indicates the locations of the object; s_k is the corre-

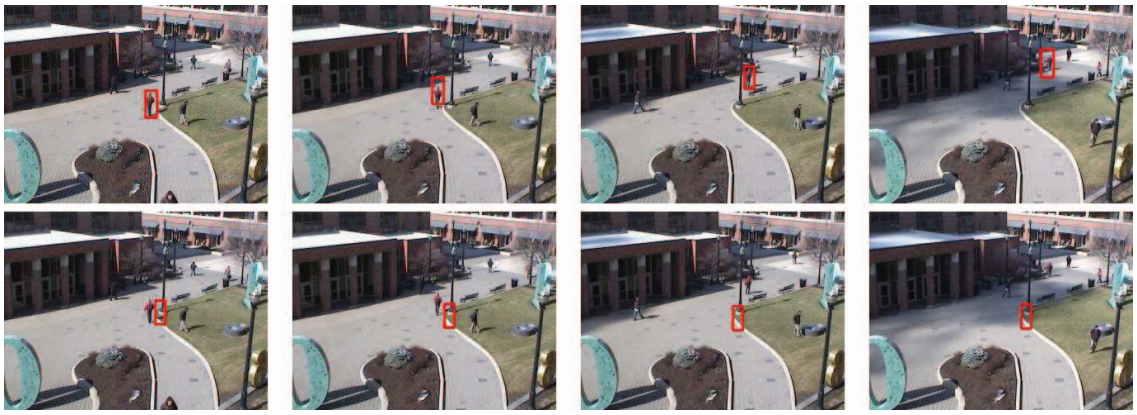

Fig. 1. Frames 69, 123, 277 and 435. Top row: The results of proposed approach; Bottom row:The results of ensemble tracking

sponding scale. The dynamics of the object is represented as $\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{v}_k$, where $\mathbf{v}_k$ is a multivariate zero-mean Gaussian random variables. The variance is set by $[\sigma_x, \sigma_y, \sigma_s] = [2, 2, 0.01]$. The scaling parameter in (1) is set to be $\sigma = 0.1$. In any new frame, we keep 10 best old features and produce 10 new features. The particle filter is assigned to 50 particles. In this experiment, we attempt to track a man through occlusion. In order to examine how can the proposed approach improve the tracking performance, we compare it with the tracking results of ensemble tracking[1]. For fair comparison, both trackers for the sequence are started with same initial detection results. Fig.1 gives some representative tracking results for both algorithms. From the tracking results we can see that original ensemble tracking approach rapidly fails and never recovers from then on, while the proposed approach continues tracking till end.

5. CONCLUSIONS

The main contribution of this paper is semi-supervised learning technology is incorporated into the framework of ensemble tracking. The classifier is online updated by using unlabelled samples which are generated by particle filter. By using semi-supervised technology, we construct efficient online learning algorithm for object tracking and avoid the drifting problems.

6. ACKNOWLEDGEMENTS

This work was jointly supported by the National Science Fund for Distinguished Young Scholars (Grant No. 60625304), the National Natural Science Foundation of China (Grants No. 60621062, 90716021), the National Key Project for Basic Research of China (Grants No. G2007CB311003, 2009CB724002), and National High-tech Research and Development Program (2007AA04Z232).

7. REFERENCES

[1] S. Avidan, Ensemble tracking, *PAMI*, vol.29, no.2, 2007, pp.261-271

[2] R. Collins, Y. Liu, and M. Leordeanu, Online selection of discriminative tracking features, *PAMI*, vol.27, no.10, 2005, pp.1631-1643

[3] H. Grabner, and H. Bischof, On-line boosting and vision, *CVPR*, 2006, pp.260-267

[4] H. Grabner, C. Leistner, and H. Bischof, Semi-supervised online boosting for robust tracking, *ECCV*, 2008, pp.234-247

[5] L. Xu, T. Yamashita, S. Lao, M. Kawade, and F. Qi, Online real Boosting for object tracking under severe appearance changes and occlusion, *ICASSP*, 2007, pp.925-928

[6] H. Liu, and F. Sun, Semi-supervised particle filter for visual tracking, *ICRA*, 2009, (In press)

[7] P. K. Mallapragada, R. Jin, A. K. Jain, and Y. Liu, Semiboost: Boosting for semisupervised learning, *PAMI*, (In press)

[8] F. Tang, S. Brennan, Q. Zhao, and H. Tao, Co-tracking using semi-supervised support vector machines, *ICCV*, 2007, pp.1-8

[9] X. Zhu, Semi-supervised learning literature survey, *Technical Report*, Computer Sciences, University of Wisconsin-Madison, 2005