REALTIME OBJECT-OF-INTEREST TRACKING BY LEARNING COMPOSITE PATCH-BASED TEMPLATES

Yuanlu Xu, Hongfei Zhou, Qing Wang, Liang Lin*

Sun Yat-Sen University, Guangzhou, China

ABSTRACT

In this paper, we propose a patch-based object tracking algorithm which provides both good enough robustness and computational efficiency. Our algorithm learns and maintains Composite Patch-based Templates (CPT) of the tracking target. Each composite template employs HOG, CS-LBP, and color histogram to represent the local statistics of edges, texture and flatness. The CPT model is initially established by maximizing the discriminability of the composite templates given the first frame, and automatically updated on-line by adding new effective composite patches and deleting old invalid ones. The inference of the target location is achieved by matching each composite template across frames. By this means the proposed algorithm can effectively track targets with partial occlusions or significant appearance variations. Experimental results demonstrate that the proposed algorithm outperforms both MIL and Ensemble Tracking algorithms.

Index Terms— Object Tracking, Composite Template, On-line Learning

1. INTRODUCTION

The problem of object tracking in surveillance applications has been long inquired in the last decades, however there are still many unresolved issues:

- Tracking target with occlusions: the tracking target is partially/fully blocked by trees, buildings, other moving objects, etc.
- Similar object interference: indistinctive appearances between the tracking target and the background.
- Tracking target with appearance variations: posture changes, and movement of the tracking targets.

Many tracking algorithms have been proposed against these issues. According to Yilmaz et al.'s [1] summary, tracking approaches can be categorized as key-point-based [2, 3], contour-based [4, 5], and kernel-based [6, 7]. Nevertheless, in recent years, new categories of tracking approaches, namely tracking-by-detection and patch-based tracking, have been extensively studied.

In 2005, Avidan [8] consider tracking as a binary classification problem. The pixels of the tracking target is detected by an on-line Adaboost classifier. In 2009, Babenko et al. [9] introduce multiple instance learning (MIL) to further ameliorate Grabner's framework. They present learning samples in sets (bags) and suppose sets labeled as positives at least contain one positive instance. Therefore the problem of tracking target is converted to the problem of finding the most 'correct' bag.

The tracking-by-detection suffers from an unavoidable model-drifting caused by updating with false positive samples. Specifically, when the tracking target is with significant occlusions, positive training samples are severely contaminated by the false positives and thus the tracking model invalidates.

The patch-based model is originally employed in object recognition [10]. In 2010, Xie et al. [11] introduce the patch-based model to the tracking problem. They compose the tracking model with the distinguishable image patches of the tracking target. In 2011, considering the inadequacy of only one type of feature, Liu et al. [12] employ hybrid templates (image patches with hybrid features) and propose a unified measurement for comparison among hybrid templates. A main drawback of above methods lies in no explicit replacement of the invalid patches during model maintenance.

In this paper, we propose a novel model maintenance approach for patch-based tracking, which adds new patches from the inferred target location of the current frame. By substituting out-dated composite patches with new valid ones, our approach increases the robustness for tracking challenging targets. Moreover, considering the limitation of LBP descriptor, we employ a more concise and effective feature: CS-LBP descriptor to represent the textural statistics of the image patch.

The rest of this paper is organized as follows. we present representation of the tracking model in Section 3.2, and discuss the tracking algorithm in Section 3. Experimental details and results are illustrated in Section 4 and finally a conclusion is reached in Section 5.

2. MODEL REPRESENTATION

As illustrated in Fig. 1, the CPT model is constructed by first extracting and selecting image patches from the tracking win-

*This work was supported by the National Natural Science Foundation of China (Grant No. 61173082), the Guangdong Natural Science Foundation (Grant no. S2011010001378), and the Science and Technology Planning Project of Guangdong Province, China (Grant No. 2010A040307003). Corresponding author: Qing Wang (wangq79@mail.sysu.edu.cn)

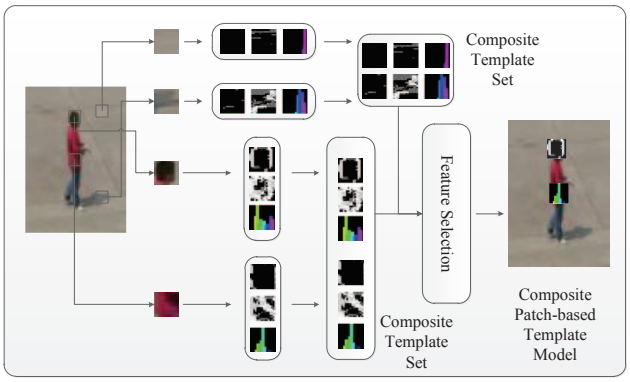

Fig. 1. With an initial frame and a tracking window, image patches with composite features are extracted in the tracking window and the background. CPT model is composed of templates with the most discriminative feature.

dow, by maximizing the difference with the background.

In this paper, we extract image patches with size $s \cdot s$ from the tracking window and the up, down, left and right 4 subregions around the tracking window, forming the patch set of the tracking target $\mathbf{B^t} = \{b_i^t, i = 1, \ldots, N_t\}$ and the patch set of the background $\mathbf{B^b} = \{b_i^b, i = 1, \ldots, N_b\}$, respectively.

For each image patch, three types of features are applied to capture the local statistics of edge, texture and flatness.

1) Edge: we describe the edges with a simplified version of histogram of gradient (HOG) [13]. Given an image patch b_i, without using block and cell overlapping, we split an image patch into 4 cells and then compute a 8-bin gradient histogram for each cell. Finally, an $8 \cdot 4 = 32$d edge feature vector is concatenated by the 4 gradient histograms.

2) Texture: we use center-symmetric local binary patterns (CS-LBP), a modified version of LBP, to characterize the local texture information. CS-LBP operator compares the intensities in center-symmetric direction, as shown in Fig. 2.

For a pixel p_c, CS-LBP is encoded as

$$\text{CS-LBP}(p_c) = \sum_{i=0}^{\frac{M}{2}-1} S(|p_i - p_{i+\frac{M}{2}}|) \cdot 2^i \qquad (1)$$

where the function $S(x)$ is defined as

$$S(x) = \begin{cases} 1, & \text{if } x > T, \\ 0, & \text{else}, \end{cases} \qquad (2)$$

where T is the encoding threshold. Following the same segmentation of an image patch in 1), we compute binary code histograms for all cells and extend them to a $16 \cdot 4 = 64$d feature vector.

3) Flatness: we represent the color probability distribution with a simple color histogram. In each color channel of an image patch, a 8-bin histogram is counted and all 3 histograms are stretched into a 24d feature vector.

Therefore, given $\mathbf{B^t}$ and $\mathbf{B^b}$, we obtain the composite template set of the tracking target $\mathbf{F^t} = \{f_{i,j}^t, i = 1, \ldots, N_t; j = 1, \ldots, 3\}$ and of the background $\mathbf{F^b} = \{f_{i,j}^b, i = 1, \ldots, N_b; j = 1, \ldots, 3\}$, where $f_{i,j}$ denotes the jth type of feature of the ith image patch.

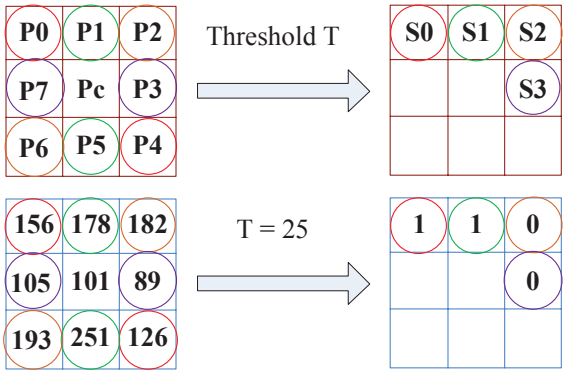

Fig. 2. Illustration of CS-LBP descriptor.

We assume the statistics in each image patch can only be well described by one certain type of feature, and thus drop the feature type notation j for simplicity in the following discussion. To formulate CPT model from the two sets $\mathbf{F^t}$ and $\mathbf{F^b}$, we select N_m composite patches from $\mathbf{F^t}$, which maximizing the probability ratio of the feature distributions within the tracking target against those within the local background regions, that is

$$\begin{aligned} \bar{\mathbf{M}} &= \arg\max_{\mathbf{M}} \frac{p(\mathbf{M})}{q(\mathbf{M})} = \arg\max_{\mathbf{M}} \sum_{i=1}^{N_m} \frac{p(f_i^t)}{q(f_i^t)}, \\ &= \arg\max_{\mathbf{M}} \sum_{i=1}^{N_m} \sum_{j=1}^{N_b} \text{KL}\,(f_i^t, f_j^b), \end{aligned} \qquad (3)$$

where $\mathbf{M} = \{f_i^t, i = 1, \ldots, N_m\}$ indicates CPT model, $p(\mathbf{M})$ the probability distribution of CPT model within the tracking target, $q(\mathbf{M})$ the probability distribution of the tracking model within the background, $p(f_i^t)$, $q(f_i^t)$ the probability distribution of a composite patch f_i^t within the tracking target and the background, respectively. KL distance is defined as

$$\text{KL}(x, y) = \sum_{i=1}^{d} x_i \cdot \log \frac{x_i}{y_i}, \qquad (4)$$

where the subscript i denotes the ith dimension.

3. TRACKING ALGORITHM

In this section, we first introduce the inference algorithm, which closely follows the hybrid templates approach [12], and then propose our maintenance mechanism.

3.1. Tracking Target Location Inference

When a new frame I_t appears, we employ KL distance to measure the similarity between two composite patches and

traverse peripheral regions around every composite patch in $\mathbf{M}$ to find the matching patch. To guarantee efficiency, the searching region is limited to a $2s \cdot 2s$ rectangle (centered by the patch) and the searching step is fixed as r_s. Afterwards, inferring the tracking target location is equivalent to obtaining a matching set $\mathbf{F^d} = \{f_i^d, i = 1, \ldots, N_m\}$, which maximizes the posterior probability $p(\mathbf{I_t}, \mathbf{F^d}|\mathbf{M})$. According to Bayesian rules, the posterior probability can be estimated as

$$p(I_t, \mathbf{F^d}|\mathbf{M}) \propto p(\mathbf{F^d}|\mathbf{M}) \cdot p(I_t|\mathbf{F^d}, \mathbf{M}), \quad (5)$$

where $p(\mathbf{F^d}|\mathbf{M})$ indicates the prior, and $p(\mathbf{I_t}|\mathbf{F^d}, \mathbf{M})$ is the likelihood between $\mathbf{M}$ and $\mathbf{F^d}$.

Supposing every patch is independent with each other, the prior $p(\mathbf{F^d}|\mathbf{M})$ is simplified as

$$p(\mathbf{F^d}|\mathbf{M}) \propto \prod_{i=1}^{N_m} p(f_i^d|f_i^t). \quad (6)$$

The probability $p(f_i^d|f_i^t)$ is acquired from the offset $\vec{\delta}_i$ of in before and after the two successive frames.

Since we assume each feature is independent and every patch is represented with one single feature, the likelihood $p(\mathbf{I_t}|\mathbf{F^d}, \mathbf{M})$ is calculated by

$$p(I_t|\mathbf{F^d}, \mathbf{M}) \propto \prod_{i=1}^{N_m} p(I_t|f_i^d, f_i^t). \quad (7)$$

To determine $p(I_t|f_i^d, f_i^t)$, we calculate three probability: probability of similarity $p^s(f_i^d, f_i^t)$ from the KL distance between f_i^d and f_i^t, probability of motion consistency $p^c(f_i^d, f_i^t)$ by the Euclidian distance between $\vec{\delta}_i$ and other offsets, probability of discriminability $p^d(f_i^d)$ by the KL between f_i^d and composite patches in $\mathbf{F^b}$. Therefore, $p(I_t|f_i^d, f_i^t)$ is estimated as

$$p(I_t|f_i^d, f_i^t) = \alpha p^s(f_i^d, f_i^t) + \beta p^c(f_i^d, f_i^t) + \gamma p^d(f_i^d), \quad (8)$$

where α, β, γ are pre-defined coefficients.

Eventually, we infer the tracking target location L_t by

$$L_t = L_{t-1} + \sum_{i=1}^{N_m} w_i \cdot \vec{\delta}_i, \quad (9)$$

where $w_i \propto p(I_t|f_i^d, f_i^t)$.

3.2. On-line Model Maintenance

Since sometimes certain composite patches are no longer effective for distinguishing the tracking target, an on-line maintenance approach which only replaces the previous patches with the matching patches in the new frame is not robust enough. Therefore, we propose a new maintenance algorithm by picking new patches and fusing them into CPT model.

The on-line maintenance mechanism is divided into two steps: firstly, we construct a candidate set $\mathbf{M_c}$ from the new frame I_t; secondly, we pick N_m composite patches from both $\mathbf{M_c}$ and the matching patch set by maximizing the discriminability of CPT model.

Following the methods proposed in Section , with a new frame I_t and the inferred target location, we obtain two feature sets $\mathbf{F^t}$, $\mathbf{F^b}$ from the inferred tracking window and the background, respectively. By maximizing Equ.(3), a candidate composite template set $\mathbf{M_c}$ is established.

Next, we fuse the tracking model $\mathbf{M}$ and the candidate set $\mathbf{M_c}$ into one excess tracking model $\mathbf{M}^+$, and select N_m composite patches from $\mathbf{M}^+$ by maximizing the probability ratio of CPT model within regions of the tracking target against those within regions of the local background. That is,

$$\begin{aligned} \bar{\mathbf{M}} = \arg\max_{\mathbf{M}} \frac{p(\mathbf{M}^+)}{q(\mathbf{M}^+)} &= \arg\max_{\mathbf{M}} \frac{p(\mathbf{M_c} \cup \mathbf{M})}{q(\mathbf{M_c} \cup \mathbf{M})}, \\ &= \arg\max_{\mathbf{M}} \sum_{i=1}^{N_m} \frac{p(f_i^+)}{q(f_i^+)}. \end{aligned} \quad (10)$$

Applying the above mechanism, we summarize our tracking algorithm in Algorithm 1.

Algorithm 1: Tracking by Learning Composite Patch-based Templates

Input: Image sequence $I = \{I_1, I_2, \ldots, I_T\}$, initial tracking window

Output: tracking target locations $L = \{L_1, L_2, \ldots, L_T\}$

```
Extract patches from I_1 and compute composite template set F^t, F^b;
Construct tracking model M = {f_i^t, i = 1, ..., N_m} given in (3)-(4);
foreach Image sequence I_t ∈ {I_2, I_3, ..., I_T} do
    Infer target location L_t by (5)-(9);
    Extract patches from I_t and compute F^t, F^b;
    Update tracking model M given in (10);
end
```

4. EXPERIMENTS

We implement our tracking algorithm using C++ language and integrate it with a surveillance system. Our algorithm achieves $15 - 30$ frames per second on an Intel Core2 Duo E5400 Computer with 2.7GHZ CPU and 2GB RAM.

All the parameters are fixed in the experiments, including the size of an image patch $s = 16$, the number of patch in tracking target set $N_t = 150$, in background set $N_b = 150$, in CPT model $N_m = 45$, and the search step $r_s = 4$, which obtains a pretty balance between the performance and the computational efficiency. The coefficients to compute the likelihood are defined as $\alpha = 0.55$, $\beta = 0.15$, $\gamma = 0.15$.

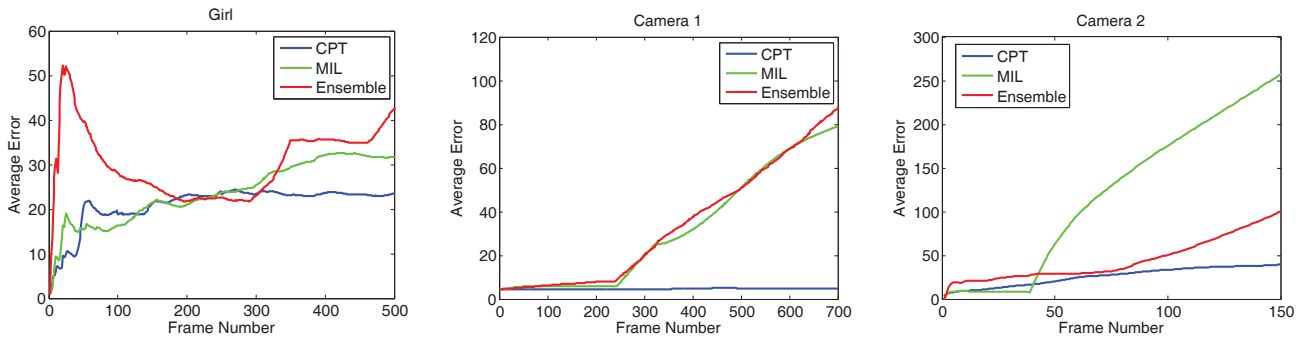

Fig. 3. Quantitative comparisons of our approach, MIL [9] and Ensemble Tracking [8] on 3 datasets

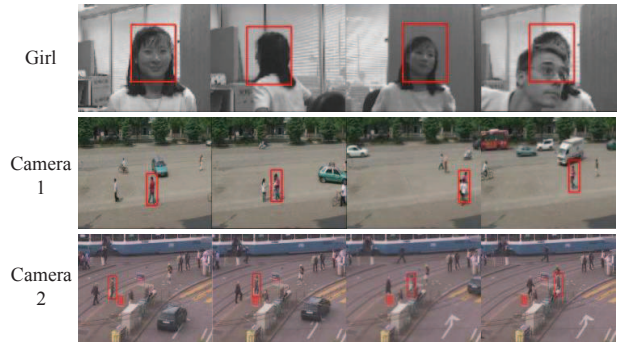

Fig. 4. Sampled results of our tracking methods. Severe body variations and large-scale occlusions take place in above datasets.

We collect 3 test videos to verify our approach, one video of the human face from MIL [9] and two surveillance videos from the internet. A number of sampled tracking results are shown in Fig. 4.

To quantitatively evaluate the performance of our method, we compute the average tracking error with manually labeled groundtruth of the tracking target location. We compare the proposed method (CPT) with two state-of-the-art algorithms: MIL [9] and Ensemble Tracking [8], as shown in Fig. 3. From the results, we can find our approach outperforms the other 2 algorithms.

5. CONCLUSION

This paper proposes an improved patch-based tracking algorithm with a novel on-line maintenance approach and the dimension-reduced textural feature: CS-LBP descriptor. This novel maintenance approach selects effective composite templates from the fusion of the matching templates and the candidate set, which outperforms other state-of-art algorithms in tracking targets with various challenges.

6. REFERENCES

[1] A. Yilmaz, O. Javed and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.*, vol. 38(4), 2006.

[2] C.J. Veenman, M.J.T. Reinders, E. Backer, "Resolving motion correspondence for densely moving points," *T-PAMI*, vol. 23(1), pp. 54–72, 2001.

[3] D. Serby et al, "Probabilistic object tracking using multiple features," *ICPR*, pp. 184–187, 2004.

[4] A. Yilmaz, X. Li and M. Shah, "Contour based object tracking with occlusion handling in video acquired using mobile cameras," *TPAMI*, pp. 1531–1536, 2004.

[5] K. Sato, J.K. Aggarwal, "Temporal spatio-velocity transform and its application to tracking and interaction," *Computer Vision and Image Understanding*, vol. 96(2), pp. 100–128, 2004.

[6] S. Avidan, "Support vector tracking," *CVPR*, pp. 184–191, 2001.

[7] D. Comaniciu, V. Ramesh, P. Meer, "Kernel-based object tracking," *TPAMI*, vol. 25(5), pp. 564–577, 2003.

[8] S. Avidan, "Ensemble tracking," *CVPR*, pp. 494–503, 2005.

[9] B. Babenko et al, "Visual tracking with online multiple instance learning," *CVPR*, pp. 938–990, 2009.

[10] X. Liu et al, "Representing and recognizing objects with massive local image patches," *Pattern Recognition*, vol. 45(1), pp. 231–240, 2012.

[11] Y. Xie, L. Lin, and Y. Jia, "Tracking objects with adaptive feature patches for ptz camera visual surveillance," *ICPR*, pp. 1739–1742, 2010.

[12] X. Liu, L. Lin, S. Yan, H. Jin, and W. Jiang, "Adaptive object tracking by learning hybrid template on-line," *TCSVT*, vol. 21(11), pp. 1588–1599, 2011.

[13] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *CVPR*, pp. 886–893, 2005.