IEICE TRANS. INF. & SYST., VOL.E100–D, NO.4 APRIL 2017

LETTER

Adaptive Updating Probabilistic Model for Visual Tracking

Kai FANG[†], ***Member***, **Shuoyan LIU**[†a)], **Chunjie XU**[†], ***and*** **Hao XUE**[†], ***Nonmembers***

SUMMARY In this paper, an adaptive updating probabilistic model is proposed to track an object in real-world environment that includes motion blur, illumination changes, pose variations, and occlusions. This model adaptively updates tracker with the searching and updating process. The searching process focuses on how to learn appropriate tracker and updating process aims to correct it as a robust and efficient tracker in unconstrained real-world environments. Specifically, according to various changes in an object's appearance and recent probability matrix (TPM), tracker probability is achieved in Expectation-Maximization (EM) manner. When the tracking in each frame is completed, the estimated object's state is obtained and then fed into update current TPM and tracker probability via running EM in a similar manner. The highest tracker probability denotes the object location in every frame. The experimental result demonstrates that our method tracks targets accurately and robustly in the real-world tracking environments.

key words: *visual tracking, transition probability matrix, expectation-maximization*

1. Introduction

It is a challenging problem to track a target in the real-world environment where different types of variations such as illumination, shape, occlusion, or motion changes occur at the same time [1]. Recently, several tracking methods solved the problem and successfully tracked targets in the real-world environment [2]–[5]. Among them, one of promising methods is the visual tracking decomposition (VTD), which utilizes a set of multiple trackers and runs them simultaneously and interactively [6]–[9]. The method assumes that, given a fixed number of trackers, at least one tracker can deal with target variations at each time. However, this assumption is insufficient to cope with the complicated real-world tracking environment. Since generally the tracking target severely varies from frame to frame, trackers should not be fixed but should be generated dynamically depending on the current tracking environment.

To this end, we propose an adaptive updating probabilistic model to accomplish visual track. Specifically, according to various changes in an object's appearance and recent transition probability matrix (TPM),the tracker probability is achieved in Expectation-Maximization (EM) manner [10]. When the tracking in each frame is completed, the estimated object's state is obtained and then fed into update current TPM and tracker probability via running EM in a similar manner. The highest tracker probability denotes the object location in every frame.

Manuscript received September 8, 2016.
Manuscript revised December 9, 2016.
Manuscript publicized January 6, 2017.

[†]The authors are with the Institute of Computing Technology Department, China Academy of Railway Sciences, Beijing, China.
a) E-mail: 06112062@bjtu.edu.cn

DOI: 10.1587/transinf.2016EDL8188

2. Adaptive Updating Probabilistic Model for Visual Tracking

Given a bounding box defining the object of interest in a first frame, our goal is to automatically determine the object's bounding box or indicate that the object is not visible in every frame that follows. The work flow of our tracking algorithm is summarized in Fig. 1. Since the real-world tracking target varies severely over time, a novel adaptive updating probabilistic (AUP)model discovers the track in the searching and updating process. The searching process focuses on how to learn appropriate track and updating process aims to adapt it to challenging real scenarios.

2.1 Adaptive Updating Probabilistic Model

Adaptive Updating Probabilistic (AUP) Model is the extension of probabilistic Latent Semantic Analysis model (pLSA model) [10]. pLSA model has received considerable interest in the text analysis community as a tool to model documents as a mixture of several semantic-but a-prior unknown, and hence latent-topics. Such a model is of interest to tracking problem since the target is similar to the latent semantic concepts for the each frame.

Given the observation's appearance and recent object's state, the probability $p(x_{t-1}|z_{t-1})$ can be efficiently estimated by the decomposed posterior probabilities as follows:

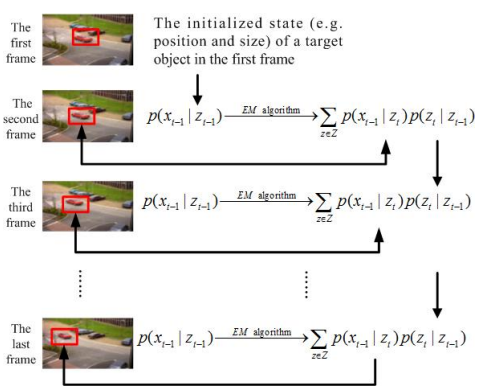


Fig. 1 Work flow of the proposed approach

$$p(x_{t-1}|z_{t-1}) = \sum_{t=1}^{T} p(x_{t-1}|z_t)p(z_t|z_{t-1}) \quad (1)$$

where $p(z_t|z_{t-1})$ is transition probabilistic matrix, representing the transition from the previous state z_{t-1} to the new state z_t. $p(x_{t-1}|z_t)$ defines the observation likelihood that measure similarity between the current state z_{t-1} and the current observation appearance x_{t-1}. The purpose of visual tracking is to estimate object's state by approximately estimating the tracker probability as following:

$$x_t = \arg\max_{z \in Z} p(x_{t-1}|z_t) \quad (2)$$

Since the proposed tracker performs object localization using a sliding-window-search scheme, the highest tracker probability denotes the object location in every frame.

The pLSA model discovers the latent topics with training and testing process. In a similar manner, the AUP model adaptively updates the tracker in the searching and updating process.

2.2 The Searching and Updating Process

In the searching process, the tracker probabilistic $p(x_{t-1}|z_t)$ is calculated with the previous $p(z_t|z_{t-1})$ kept fixed in Expectation-Maximization (EM) manner. When the searching in each frame is completed, the estimated object's state is fed into update the tracker distributions $p(x_{t-1}|z_t)$ and corresponding transition probabilistic matrix $p(z_t|z_{t-1})$ in a similar manner.

The searching process estimates $p(x_{t-1}|z_t)$ with the previous $p(z_t|z_{t-1})$ kept fixed in Expectation-Maximization (EM) manner [10] by maximizing the log-likelihood function:

$$L = \sum_{z \in Z} n(z_{t-1}, x_{t-1}) \log p(z_{t-1}, x_{t-1}) \quad (3)$$

where $p(z_{t-1}, x_{t-1}) = p(z_{t-1})p(x_{t-1}|z_{t-1})$, and $n(z_{t-1}, x_{t-1})$ is the similar frequency, representing every observation regions appearance for the target.

Since the EM algorithm is sensitive to the initialization, an important consideration for EM is that the performance of model is strongly affected by the initialization. We assume that the recent object state is similar to the current object appearance. Hence, the initialization of track probabilistic $p(x_{t-1}|z_t)$ is same as $p(x_{t-1}|z_{t-1})$. And then $p(x_{t-1}|z_t)$ is calculated with the previous $p(z_t|z_{t-1})$ kept fixed.

In E-step, the posterior probabilities for $p(z_t|z_{t-1}, x_{t-1})$ are calculated, and in M-step, $p(x_{t-1}|z_t)$ is updated. Here we list the rules in the EM algorithm as follows:

E-step:
$p(z_t|z_{t-1}, x_{t-1}) \propto p(x_{t-1}|z_t)p(z_t|z_{t-1})$
M-step:
$p(x_{t-1}|z_t) \propto \sum_{z \in Z} n(z_{t-1}, x_{t-1})p(z_t|z_{t-1}, x_{t-1})$
$p(z_t|z_{t-1}) \propto p(z_{t-1}, x_{t-1})p(z_t|z_{t-1}, x_{t-1})$

When the searching is completed, the estimated object's state is obtained and then fed into update current TPM and tracker probability via running EM in a similar manner. We first compute the visual feature similarity between estimated object's state(Eq. (2))and current states according to the histogram intersection:

$$p_v(x_t|z_t) = [S(x_t, z_{t1}), \ldots, S(x_t, z_{tz}), \ldots, S(x_t, z_{tZ})],$$
$$S(x_t, z_{tz}) = \frac{\sum_{i=1}^{N} min(x_t(i), z_{tz}(i))}{\sum_{i=1}^{N} x_t(i)} \quad (4)$$

where $x_t(i)$,$z_{tz}(i)$ define the appearance features of estimated object's state and z^{th} object state, respectively. Since the proposed tracker adopts dense sampling method as searching scheme, $z_t \in \{z_{t1}, \ldots, z_{tz}, \ldots, z_{tZ}\}$ represents the object state of Z sliding-windows. In addition, we normalize the probability $p_v(x_t|z_t)$ according to $\frac{p_v(x_t|z_t)}{\sum_{z=1}^{Z} p_v(x_t|z_t)}$.

To ensure the $p(x_{t-1}|z_t)$ be closer to the visual similarity $p_v(x_t|z_t)$, we then incorporate their difference as a regularization factor into the Q function as Eq. (5). The AUP model tries to learn $p(z_t|x_{t-1})$ and $p(z_t|z_{t-1})$ with an EM algorithm by maximizing the Q function below:

$$Q = \sum_{z \in Z} n(z_{t-1}, x_{t-1}) \log p(z_{t-1}, x_{t-1}) - \rho \sum_{z \in Z} (p(x_{t-1}|z_t) - p_v(x_t|z_t))^2 \quad (5)$$

The parameter ρ is a selecting parameter, whose value is between 0 and 1. In the searching stage, ρ is set as 0, which means that the regularization factor has no influence on the maximum log-likelihood. ρ is set to 1 in the updating process, which means that regularization factor forces the learned tracker be more appropriate for real scenarios. The tracking algorithm is summarized in Table 1.

Table 1 Adaptive updating probabilistic model for visual tracking

Input: image sequences
If frame =1 then 1.Given the initialized state (e.g., position and size) of a target object; 2.$p(z_1|z_0) = 1$ End
For frame t=2:last Searching processing: 1. Calculate $p(x_{t-1}|z_{t-1})$ 2. The initialization of $p(x_{t-1}|z_t) = p(x_{t-1}|z_{t-1})$; The initialization of $p(z_t|z_{t-1}) = p(z_{t-1}|z_{t-2})$ 3. $p(x_{t-1}|z_t)$ is calculated with the previous $p(z_t|z_{t-1})$ kept fixed in Expectation-Maximization (EM) manner. Updating processing: 1. Calculate the visual feature similarity $p_v(x_t|z_t)$ 2. Learn $p(z_t|x_{t-1})$ and $p(z_t|z_{t-1})$ with an EM algorithm by maximizing the Q function according to Eq. (4) 3. The highest tracker probability denotes the object location in every frame. End
Output:the location where the target might be in each frame.

3. Experiment Results

The proposed tracking model aims to accomplish the tracking and surveillance in the passenger railway stations (PRS). We construct a new PRS dataset of videos collected in the different passenger railway stations on the different times. Some examples of the dataset are shown in Fig. 2. The database contains 9100 videos from 140 stations on 65 different times. This property makes video sequences contain diverse events such as occlusion, object pose variation, lighting changes and out-of-plane rotation.

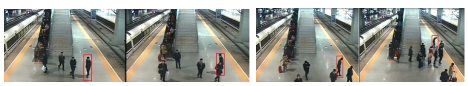
Since the PRS dataset is an unpublished dataset, we propose to categorize the sequences by annotating them with the 11 attributes like [1](IV: Illumination Variation, SV: Scale Variation, OCC: Occlusion, DEF: Deformation, MB: Motion Blur, FM: Fast motion, OV: Out-of-view, BC:Background clutters, LR:Low Resolution). The attribute distribution in our dataset is shown in Fig. 3.

For each sequence, the location of the tracked target is manually labeled in the first frame. The proposed tracker performs object localization using a sliding-window-search scheme with a search radius of 30 pixels. For each image region, we extract the histograms of the oriented gradients (HoG) features. The HoG feature is composed of 2×2 cells, with each cell represented by a 9-dimensional histogram vector, in five spatial block-division models, resulting in a 180-dimensional feature vector. Finally, the adaptive updating probabilistic model follows the target from frame to frame.

In this work, we use the precision and success rate for quantitative analysis. Precision rate is the average center location error, which is defined as the average Euclidean distance between the center locations of the tracked targets and the manually labeled groundtruths. Success rate is measured by the bounding box overlap. Given the tracked bounding box B_T and the ground truth bounding box B_G, the overlap score is defined as $SR = \frac{|B_G \cap B_M|}{|B_G \cup B_M|}$. Hence, the overall performance for the tracker in summarized by the success and precision plots of one-pass evaluation (OPE) as shown in Fig. 4. Specifically, we first run the proposed track model throughout each test sequence with initialization from the ground truth position in the first frame and report the average precision or success rate.

There are four videos, which collected from the ZhengZhou Dong railway station and Taiyuan railway station at the different time. It can be seen that our method tracks targets accurately and robustly in the real-world tracking environments. A closer look at the figure reveals that the lower rate occurs in the Taiyuan station. This is because that Zhengzhou station adopted the high definition cameras while there are standard definition cameras in Taiyuan station. In addition, the performance of video captured on the March is higher than the November. The main reason is that haze easily occurs in this reason.

Figure 5 compares the Adaptive Updating Probabilistic (AUP) model with existing results in the literature[1]. To compare these approaches in a fair manner, we also implement the proposed method in the well-known benchmark datasets[1], which collected and annotated most commonly used tracking sequence. We use the same evaluation measurement(i.e., plots of OPE, SRE(spatial robustness evaluation) and TRE(temporal robustness evaluation)). For each figure, the top 10 trackers [1] and the proposed are presented for clarity and comparison. The AUP model performs well in SRE and TRE, which suggests this model is effective to



(a) Occlusion and Motion blur (b) Pose variation and illumination change

Fig. 2 Example of PRS dataset

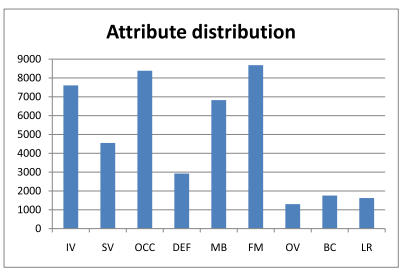


Fig. 3 Attribute distribution of the PRS testset

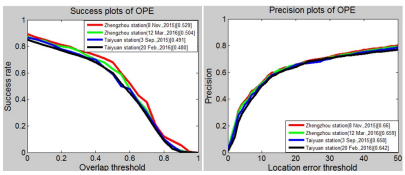


Fig. 4 the overall performance of the proposed tracking algorithm on the four different scenes

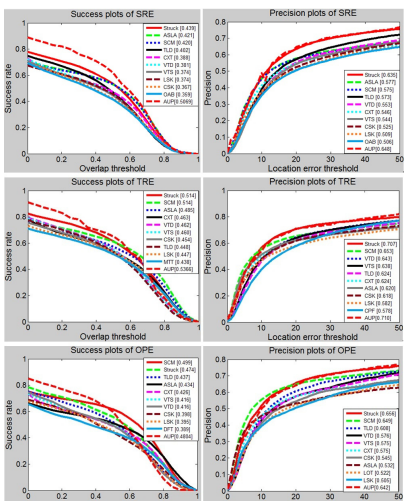


Fig. 5 Comparison tracking results with existing work for the benchmark datasets [1]

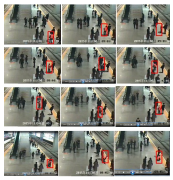


Fig. 6 Visual tracking results for video collected from Zhengzhou Dong Railway station

account for appearance change (e.g., occlusion). The main reason is that AUP model adaptively updates track with the searching and updating process. The searching process focuses on how to learn appropriate track and updating process aims to adapt it to challenging real scenarios. In contrast, the other approaches estimate the location of target depended on the appearance only. The initialization errors tend to cause trackers to update with imprecise appearance information, thereby causing gradual drifts. In addition, the top ranked tracker in OPE outperforms AUP model by 2.6%. This is because AUP model performs well in long sequences while there are numerous short segments.

Video surveillance system in the railway station is the important application for visual tracking model. We apply the proposed visual tracking algorithm for video surveillance system in the railway station as shown in Fig. 6. The average performance of this approach achieves 87.8%. In addition, we now provide a breakdown of the computational cost, since our emphasis in this work is on practicality. We run our code on PC with Intel Core i7 with 3.4 GHz and 8 GB of RAM. The average processing time for the on-line process is 2.8 sec/frame. The experimental results demonstrate the effectiveness of the proposed method for visual tracking at a modest computational cost. This approach is very practical since it has reasonable computational costs.

4. Conclusion

This paper proposes the adaptive updating probabilistic model which tracks target with the searching and updating process. The searching process focuses on how to learn appropriate tracker and updating process aims to correct it as a robust and efficient tracker in unconstrained real-world environments. The experimental results demonstrate that our method tracks targets accurately and robustly in the real-world tracking environments. However, the task of visual tracking still leaves room for improvement. This is promising direction, and we might be able to find much more powerful tracker for visual tracking.

References

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," ACM Computer Survey, vol.38, no.4, Article No. 13, 2006.
[2] B. Babenko, M. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp.215–228, 2009.
[3] Y. Wu, J. Lim, and M.-H. Yang, "Online Object Tracking: A Benchmark," Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp.2411–2418, 2013.
[4] J.H. Yoon, D.Y. Kim, and K.-J. Yoon, "Visual Tracking via Adaptive Tracker Selection with Multiple Features," Proc. IEEE European Conference on Computer Vision, pp.28–41, 2012.
[5] X. Li, C. Shen, A. Dick, and A.V. Hengel, "Learning Compact Binary Codes for Visual Tracking," Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp.2214–2222, 2013.
[6] H. Grabner, J. Matas, L.V. Gool, and P. Cattin, "Tracking the Invisible: Learning Where the object Might be," Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp.4298–4306, 2013.
[7] M. Yang, Y. Wu, and G. Hua, "Context-aware visual tracking," IEEE Trans. Pattern Anal. Mach. Intell., vol.31, no.7, pp.1195–1209, 2009.
[8] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-Detection," IEEE Trans. Pattern Anal. Mach. Intell., vol.34, no.7, pp.1409–1422, 2012.
[9] J. Kwon and K.M. Lee, "Tracking by Sampling Trackers," Proc. IEEE Int. Conf. Computer Vision, pp.102–110, 2011.
[10] T. Hofmann, "Probabilistic latent semantic indexing," Proc. ACM SIGIR Conf. Research and Development in Information Retrieval, pp.50–57, 1999.
[11] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection[C]," Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp.886–893, 2005.