2010 International Conference on Pattern Recognition

Tracking Objects with Adaptive Feature Patches for PTZ Camera Visual Surveillance

Yi Xie[1,2], Liang Lin[2,3,*], Yunde Jia[1]

[1]*Beijing Lab of Intelligent Information, School of Computer Science and Technology*
Beijing Institute of Technology, Beijing 100081, P.R.China
[2]*Lotus Hill Research Institute, P.R.China*
[3]*School of Software, Sun Yat-Sen University, Guangzhou 510275, P.R. China*
Email: linliang@ieee.org, yxie.lhi@gmail.com, jiayunde@bit.edu.cn

***Abstract*—Compared to the traditional tracking with fixed cameras, the PTZ-camera-based tracking is more challenging due to (i) lacking of reliable background modeling and subtraction; (ii) the appearance and scale of target changing suddenly and drastically. Tackling these problems, this paper proposes a novel tracking algorithm using patch-based object models and demonstrates its advantages with the PTZ-camera in the application of visual surveillance. In our method, the target model is learned and represented by a set of feature patches whose discriminative power is higher than others. The target model is matched and evaluated by both appearance and motion consistency measurements. The homography between frames is also calculated for scale adaptation. The experiment on several surveillance videos shows that our method outperforms the state-of-arts approaches.**

***Keywords*-PTZ based tracking; patch based object models; feature pursuit;**

I. Introduction

Tracking object with the PTZ camera plays important roles in a wide range of key applications. There are two main issues in PTZ tracking task. One is that the unavailability of the background modeling increases the difficulty of the location prediction for moving object. The other one is the challenge of the target scale adaptation corresponding to the pan and zoom operations of the PTZ camera.

In many literatures, the learning-based method are proposed for object tracking without background modeling. Avidan [1] trained an ensemble of weak classifiers online to distinguish between the object and background, which was combined into a strong classifier using AdaBoost. Lu et al. [2] proposd a method which treated target localization and segmentation as an online binary classification problem using dynamic foreground/background appearance models. Michael et al. [3] used an on-line boosting technique for learning descriptions of detected keypoints lying within the region of interest. Helmut et al. [4] further extended the on-line boosting to an on-line semi-supervised version which allowed to limit the drifting problem while still staying adaptive to appearance changes. However, these methods may fail in two cases. (i) The global or marginal model trained by learning-based methods is not adept at handling partial occlusion. (ii) The variance of color descriptors caused by volatile illumination will severely impact the accuracy of tracking.

Addressing these two problems, we propose a different tracking algorithm inspired by feature pursuit in object recognition [5]. Our algorithm is different from other patch-based tracking algorithms such as the one introduced in [6], which detects the target from sampled single patches but from flexible patch template. In our approach, the target model is constituted by a set of selected feature patches described by the histogram of orientation gradient in the transformed color space. The spatial locations of the patches encode the structure information explicitly. Together with an online feature pursuit algorithm, the target models are adaptively learned and updated with respect to the background surrounding. Intuitively, the distinctive and discriminative image patches are selected as features. In the tracking process, the target localization and correspondence is defined by a matching function integrating both appearance and motion consistency; the homography between frames is also calculated online to propose the underlying target scale change.

Compared with the previous tracking methods, the contributions of this paper are (i) A patch-based target model is proposed for integrating object appearance and structure information, (ii) The adaptively model learning and updating effectively reduce the model drift problem. On the public LHI dataset [7], our approach is applied on several videos from PTZ-cameras and outperforms the state-of-the-art methods.

II. Learning model with feature patches

We assume that the initial target is designated accurately by a bounding box selected manually. Then, the initial target model is learned from the target image Λ_F cropped from the bounding box. Λ_F is composed by a set of image primitives $\{J_i\}_{i=1}^{N}$, each of which occupies a certain space

*Corresponding author:
Email address: linliang@ieee.org (Liang Lin)

1051-4651/10 $26.00 © 2010 IEEE
DOI 10.1109/ICPR.2010.430

IEEE computer society

with only little overlap with each other. In our approach, we adopt the simplest type of image primitives — image patches. We quantize the feature of the image patch by HoG descriptor [8] which can characterise the local appearance rather well in the surveillance video with low resolution and poor chromatic quality.

To improve the accuracy of matching in surveillance environment with illumination changes, the image patch is normalized as (1) before calculating HoG descriptor. The HoG descriptor calculated on the normalized image patch is scale-invariant, shift-invariant and invariant to light color change and shift [9]. We denote the descriptor of image patch J as $F(J)$.

$$(R', G', B') = (\frac{R-\mu_R}{\sigma_R}, \frac{G-\mu_G}{\sigma_G}, \frac{B-\mu_B}{\sigma_B}) \quad (1)$$

The descriptors of the target image patches form a candidate descriptor pool $\{F(J_1), F(J_2), \ldots, F(J_N)\}$, where $\{J_i\}_{i=1}^N$ satisfy $\bigcup_{i=1}^N J_i = \Lambda_F$. A subset of the candidate descriptor pool forms the template T of a target. All the descriptors in the T are called template descriptors. The image patch whose descriptor is in T is called template image patch. Each template descriptor characters a discriminative feature of the target image against the background. In other words, each template descriptor reduce the uncertainty of whether a image region is the target image being tracked. To reduce the uncertainty as much as possible, we adopt a step-wise feature pursuit method [5] to select the most discriminative descriptors as Alg. 1. From the algorithm, we can find that template descriptors are selected by maximizing the ratio between foreground and background distributions

$$\frac{p(T|\Lambda_F)}{q(T|\ \overline{\Lambda_F}\ )} = \sum_{i=1}^{M} \kappa(p(F(J_i)|\Lambda_F)||q(F(J_i)|\ \overline{\Lambda_F}\ )) \quad (2)$$

where $p(\cdot|\Lambda_F)$ and $q(\cdot|\ \overline{\Lambda_F}\ )$ are the distribution in the foreground and background area, seperately. The M descriptors which can maximize the (2) are selected from candidate descriptor pool. This model encourages the choice of the most discriminative descriptors with respect to the their distribution in the background. In our implementation, $q(F(J_i)|\ \overline{\Lambda_F}\ )$ is collected from the image patches around the target bounding box. A template descriptor has less opportunity to have a match in the background area in the next frame so as to serve as a better clue for tracking.

Algorithm 1: Step-wise feature pursuit algorithm

$p(T|\Lambda_F) = q(T|\ \overline{\Lambda_F}\ )$;
for $i \leftarrow 1$ **to** M **do**
  choose a new $F(J_i)$ which maximize KL divergence $\kappa(p(F(J_i)|\Lambda_F)||q(F(J_i)|\ \overline{\Lambda_F}\ ))$;
  $p(T|\Lambda_F) = p(T|\Lambda_F)\frac{p(F(J_i)|\Lambda_F)}{q(F(J_i)|\ \overline{\Lambda_F}\ )}$;

III. Tracking algorithm

After the target template is learned, we can track the target by the clue. The template matching procedure can be viewed as maximizing the objective probability function

$$p(\hat{T}, \Lambda_F^{(t+1)}|T) = p(\hat{T}|\Lambda_F^{(t+1)}, T) \cdot p(\Lambda_F^{(t+1)}|T) \quad (3)$$

where ${\Lambda_F}^{(t+1)}$ is the new target image matched by the template. $\hat{T}$ is the set of descriptors matched by template descriptors within ${\Lambda_F}^{(t+1)}$. The likelihood part evaluates the motion consistency of the matched patches and is calculated by $MC(T, \hat{T})$, which will be introduced later. The prior part measures the identicalness in appearance. Assuming the template descriptors are mutual independent, we can specify the prior part as

$$p(\Lambda_F^{(t+1)}|T) = \prod_{i=1}^{M} S(F(J_i^{(t+1)}), F(J_i)) \quad (4)$$

where $J_i^{(t+1)}$ is the match of J_i in the new frame. $S(\cdot,\cdot)$ is the similarity measurement between two image descriptors. The similarity is measured by

$$S(F(J_i^{(t+1)}), F(J_i)) = \exp\left\{-\chi^2(F(J_i^{(t+1)}), F(J_i))\right\} \quad (5)$$

where $\chi^2(\cdot,\cdot)$ is the chi-square distance. The $MC(T, \hat{T})$ is measured by a simple consensus method which is a simple version of RANSAC. There exists a deviation between the template patch and its match. The inner product of the deviation vectors of every two template patch is calculated and add to the inconsistency accumulator of both template patch. The variation of the relative position between the template patches and their matches is inevitable. So, we only punish the template patches having inconsistencies above the average to reduce the effect of the less credible matches. Based on the above considerations, the consistency between the template and its match is given by

$$MC(T, \hat{T}) = \prod_{i=1}^{M} MC(J_i) \quad (6)$$

where $MC(\cdot)$ is defined as

$$MC(J_i) = \begin{cases} 0.95, & \text{if inconsistency of } J_i \text{ is above average} \\ 1, & \text{otherwise} \end{cases} \quad (7)$$

The mode of the objective function is found by a local maximum searching scheme inspired by Mean-Shift, which is achieved by weighted sum of the deviation of the template image patches. The weight of every image patch is its similarity multiplied by consistency.

IV. Model Adaptation

To adapt the template to lastest appearance of the target, the template is updated once a good match is achieved. Our definition of a good match is that 60% of patches has a match whose similarity is over 0.6 and over 60% of patches has a inconsistency below the average inconsistency.

When the old template find its good match, we hypothesis that the matched target image Λ_F is relatively precise. Therefore, we could regenerate a new template in the current foreground area to adapt to the new appearance of the target. The M descriptors which can maximize (8) are selected as new template descriptors.

$$\frac{p(T'|\Lambda_F)}{q(T'|\ \overline{\Lambda_F}\ )} \cdot L(T', T) \tag{8}$$

where the T' is the new generated target template. $L(\cdot,\cdot)$ is a measurement of resemblance between the new generated descriptor and the old template descriptors. The resemblance between the new generated template T' and the old template T is defined as

$$L(T', T) = \prod_{m=1}^{M} \sum_{n=1}^{M} S(F(J'_m), F(J_n)) \tag{9}$$

The resemblance is calculated to ensure that majority of the new generated descriptors are still extracted from the target image primitives.

Meanwhile, the scale of the target image Λ_F is adapted according to the zoom rate from homography. The SIFT [10] correspondences between neighboring frames are established and the homography is estimated by RANSAC [11]. Because, the zoom rate estimated between frames is jittering around 1, we don't apply the zoom rate on the target bounding box, directly. The zoom rate is multiplied together till the accumulation is over 1.2 or below 0.8. Then, we extended or shrinked the bounding box following the zoom rate. If the area of the target being tracked is kept from being the major part of the whole image, the bounding box is adapted precisely.

V. Experiments

We test our method on the public LHI dataset [7], which contains 27 video clips captured by hand-held video cameras. Each clip has around 1000 frames and includes scale changing, (i.e. the camera zooms in suddenly), heavy occlusion, and complex background clutter. For each clip, the target (a pedestrian) is annotated manually over frames as the ground truth. The target location at the beginning frame is given from ground truth for the algorithm as initialization. To quantitatively evaluate the tracking performance, we use two types of measurements: (i) the object-level and (ii) the trajectory-level. For comparison, we also perform 3 other state-of-the-art tracking algorithms, Mean-Shift [12], particle filtering [13] and optical flow [14].

Firstly, we count the correct number of correctly tracking frames for each video clip, as reported in Table I. The tracking result is counted as correct only if the ratio of the overlap between the detecting and annotated box to the area of the latter one is over 0.5.

Table I
THE COMPARISON OF OBJECT-LEVEL MEASUREMENT.

	Ours	MS	PF	OF
1	**0.7662**	0.7475	0.7799	0.7537
2	**0.7839**	0.8711	0.8750	0.3034
3	**0.8080**	0.7416	0.7616	0.8091
4	**0.8009**	0.3086	0.5818	0.3735
5	**0.9167**	0.7934	0.7840	0.7629
6	**0.7490**	0.3201	0.9035	0.6839
7	**0.6726**	0.2847	0.4296	0.5040
8	**0.7867**	0.6699	0.6750	0.4833
AVG.	**0.7855**	0.5921	0.7238	0.5842

Secondly, to evaluate the consistency of the tracking trajectories, we calculate the variety (i.e. size and location) of the tracked box inner frames and compare with the ground truth. Note this evaluation is confined to count at the correct tracking frames. As shown in Fig. 1, we plot the curves with this evaluation. Intuitively, the more drastic fluctuation indicates more instability of tracking and the smaller error means greater accuracy.

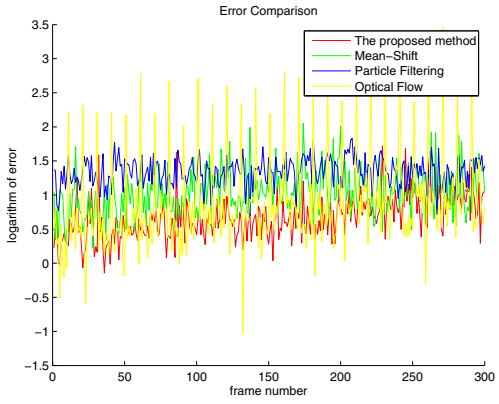

Figure 1. The comparison of trajectory-level measurement.

The result of our method in handling scale changes are shown in Fig. 2. Examples of our method in handling partial occlusion are shown in Fig. 3.

VI. Conclusion

We present a novel tracking method that adaptively learns and updates the target model with a set of selected feature patches. This algorithm can be applied to the application of PTZ camera visual surveillance and can run at real-time speed around 20 frames per second. The quantitative experiments on the public dataset is also proposed.

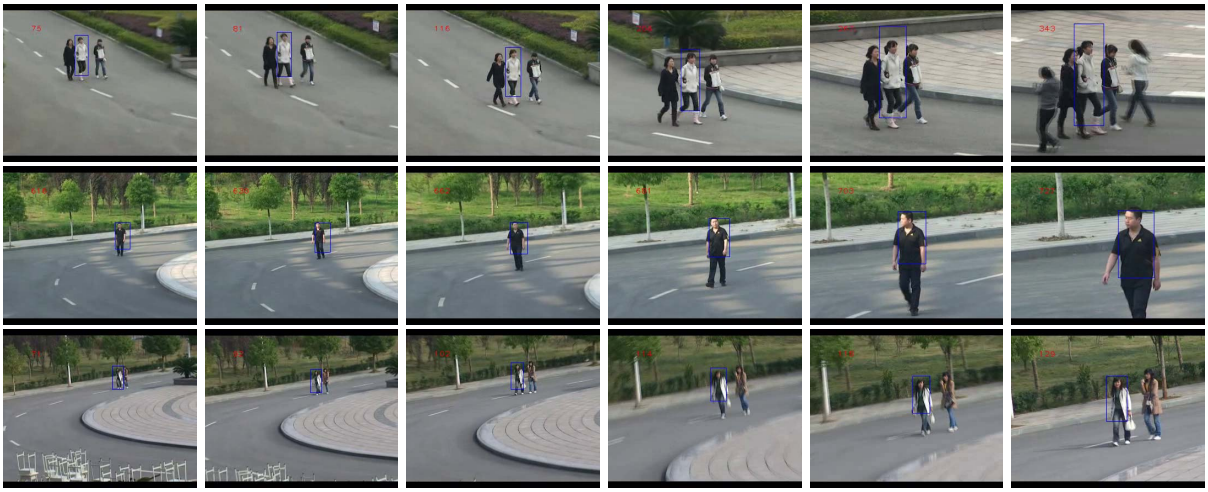

Figure 2. Examples of the proposed tracking method in handling scale changes.

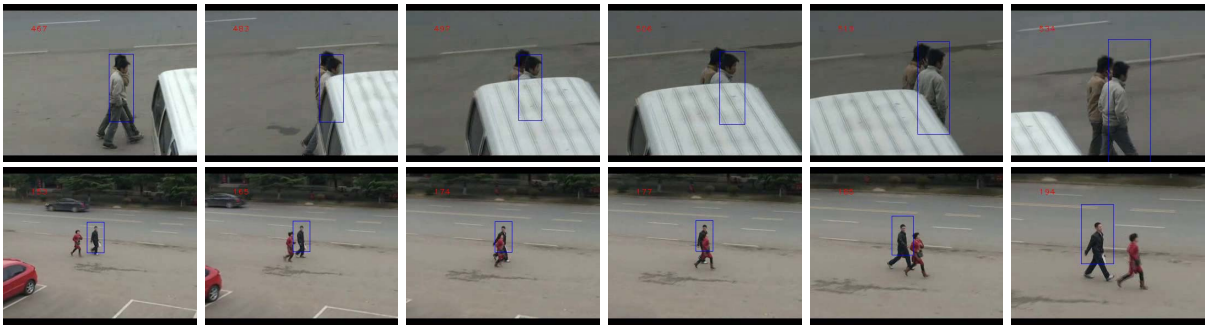

Figure 3. Examples of the proposed tracking method in handling partial occlusion.

ACKNOWLEDGMENT

This work was partially supported by the Chinese High-Tech Program under Grant No.2008AA01Z126 and No.2009AA01Z331, and the Natural Science Foundation of China under Grant No.90920009 and No.60970156.

REFERENCES

[1] S. Avidan, "Ensemble tracking," *CVPR*, June 2005.

[2] L. Lu and G. D. Hager, "A nonparametric treatment for location/segmentation based visual tracking," *CVPR*, 2007.

[3] M. Grabner, H. Grabner, and H. Bischof, "Learning features for tracking," *CVPR*, 2007.

[4] H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised on-line boosting for robust tracking," 2008.

[5] Z. Si, H. Gong, Y. Wu, and S. Zhu, "Learning mixed templates for object recognition," *CVPR*, 2009.

[6] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," *CVPR*, 2009.

[7] B. Yao, X. Yang, and S. Zhu, "Introduction to a large scale general purpose groundtruth dataset: methodology, annotation tool, and benchmark," *EMMCVPR, Springer LNCS*, vol. 4697, 2007.

[8] N. Dalal, "Finding people in images and videos," 2006.

[9] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek, "Evaluation of color descriptors for object and scene recognition," *CVPR*, 2008.

[10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, 2004.

[11] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395.

[12] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Trans, PAMI*, vol. 17, no. 8, pp. 790–799, August 1995.

[13] K. Nummiaro, E. Koller-Meier, and L. V. Gool, "An adaptive color-based particle filter," *IVC*, 2003.

[14] J. Y. Bouguet, "Pyramidal implementation of the lucas kanade feature tracker description of the algorithm."