SPATIOTEMPORAL LATENT SEMANTIC CUES FOR MOVING PEOPLE TRACKING

Peng Zhang and Sabu Emmanuel
School of Computer Engineering
Nanyang Technological University
Singapore, 639798
{zh0036ng,asemmanuel}@ntu.edu.sg

Pradeep K Atrey
Department of Applied Computer Science
The University of Winnipeg
515 Portage Avenue, Winnipeg, Canada
p.atrey@uwinnipeg.ca

Mohan S Kankanhalli
School of Computing
National University of Singapore
Singapore, 117590
mohan@comp.nus.edu.sg

ABSTRACT

Effective and robust visual tracking is one of the most important tasks for the intelligent visual surveillance. In this paper, we proposed a novel method for detecting and tracking moving people using the spatiotemporal latent semantic cues and the incremental eigenspace tracking techniques. During tracking process, the target appearance model is incrementally learned in low dimensional tensor eigenspace by adaptively updating the eigenbasis and sample mean. At the same time, the spatiotemporal latent semantic cues calibrate the estimation of tracking and detect new moving people coming in the same surveillance scene. Experiment results show that with the calibration based on spatiotemporal latent semantic cues, the proposed method can track the moving people automatically and effectively.

***Index Terms*—** Tracking, detection, surveillance, eigenvectors, learning systems

1. INTRODUCTION

Object tracking mainly aims at locating the position of the interesting object frame by frame and marking the regions where the object is in each frame [1]. The main challenge of the object tracking is to deal with the intrinsic and extrinsic appearance variations of the tracking target. Thus, how to effectively model the appearance variations becomes a key to the solution of the object tracking problems.

In recent years, a lot of research works have been done for the object tracking based on target appearance modeling and using the eigenspace analysis for object tracking has been demonstrated to be effective by many works. Black and Jepson [2] proposed an algorithm by utilizing the representation of pre-trained view-based eigenspace and a robust error norm to model the appearance variations. Their algorithm makes the assumption of subspace constancy in motion estimation instead of using the assumption of brightness constancy in optical flow based techniques. However, the robust performance of this algorithm is at the cost of large amount of off-line training images that may cover as much as possible appearance variation (due to viewing angle or illumination) from which to construct the eigenbasis. This requirement can not be always fulfilled under many realistic visual surveillance. A more flexible mixture model via online *EM* to explicitly model the appearance change during tracking was recently proposed by Fleet *et al.* [3]. Although their algorithm has good discriminability between the variations of pose, illumination and expression, its treatment of the pixels in the target region makes it fail when background pixels are modeled other than foreground during tracking. In order to treat the tracking target as an abstract “thing” other than independent pixels, Lim *et al.* proposed their online incremental learning algorithm for robust visual tracking in [4]. Their algorithm is also an online learning approach without training phase because it learns the eigenbasis during the object tracking process. The efficient subspace update mechanism facilitates this algorithm successfully track the object under varying pose and illumination conditions. However, this robust tracking algorithm may also drift from target under some circumstances, such as small target size or strong noise. Based on the work of Lim, Li *et al.* [5] and Zhang *et al.* [6] respectively proposed their algorithm for object tracking.

Since many tracking methods require the object detection before tracking begins, how can we properly incorporate the detection and tracking algorithms into a more robust tracker? Inspired by the work of Ishiguro *et al.* [7], we propose a method to achieve robust moving people tracking by incorporating the incremental eigenspace tracking and spatiotemporal latent semantic analysis for moving people detection. Although there are a lot of algorithms that have been proposed for moving people detection, many of these algorithms often fail when there is camera motion (pan-tilt-zoom) or infrequent background changes. Therefore, utilizing the interest points for the moving people detection becomes more and more popular recently because the interest points have a desirable quality of illumination and camera viewpoint invariance [1]. One of the interest point detection algorithms was proposed by Harris *et al.* [8] to detect the interest points in a still image by using an auto-correlation matrix. Although this spatial algorithm can find a lot of interest points, many of these points do not belong to the moving objects that the

978-1-4244-2354-5/09/$25.00 ©2009 IEEE

user really wants to detect and track in visual surveillance application. Thus, some methods make use of the temporal information obtained from the consecutive video frames to reduce the spatial detection method errors. An extension of the two-dimensional spatial interest point detection to the three-dimension of space and time *STIP* was proposed by Laptev in [9]. However, only using the *STIP* for detection can only detect the spatial interest points which have the temporal changing information in the spatial region, but cannot differentiate whether these points really belong to the moving people or to the other objects. An effective way is to use these points to generate some more discriminative features, which could represent the object or the moving people more effectively. In the proposed work, we employ the (*SIFT*) features [10] trained by *pLSA* [11] to obtain the latent semantic information for moving people detection purpose. The detection process can help the tracking algorithm to determine whether the tracker drifts from the target and whether the new incoming objects are moving people that need to be tracked or not. In section 2, we introduce proposed method, in section 3, we present the experiment results and this paper will be concluded in section 4.

2. PROPOSED METHOD

The proposed method includes the tracking and calibration based on spatiotemporal latent semantic cues of moving people. The incremental eigenspace tracking algorithm will firstly be introduced in general, then the spatiotemporal latent semantic cues discovery of moving people will be presented, finally we will describe how to calibrate the tracking result by using the discovered cues.

2.1. Incremental eigenspace tracking

The algorithm of incremental eigenspace tracking is proposed as an extension of the work sequential Karhunen-Loeve (*SKL*) [12] by using an incremental *PCA* mechanism, which can update the eigenbasis as well as mean. Suppose we have a $d \times n$ data matrix $\mathcal{A} = \{\mathbf{I_1}, \mathbf{I_2}, ..., \mathbf{I_n}\}$, each column $\mathbf{I}$ is an observation, which is a d dimensional vector of image, the SVD of $\mathcal{A}$ can be represented as $\mathcal{A} = U\Sigma V^\top$. If there is a new observation $\mathcal{B} = \{\mathbf{I_1}, \mathbf{I_2}, ..., \mathbf{I_m}\}$, which is a $d \times m$ matrix, let $\mathcal{C} = \left[\begin{array}{cc} \mathcal{A} & \mathcal{B} \end{array}\right]$ to be the concatenation of $\mathcal{A}$ and $\mathcal{B}$, the SVD of $\mathcal{C}$ can be represented as follows:

$$\mathcal{C} = \left[\begin{array}{cc} \mathcal{A} & \mathcal{B} \end{array}\right] = \left(\left[\begin{array}{cc} U & \tilde{\mathcal{B}} \end{array}\right] \tilde{U}\right) \tilde{\Sigma} \left(\tilde{V}^\top \left[\begin{array}{cc} V^\top & 0 \\ 0 & I \end{array}\right]\right) \tag{1}$$

In this equation, $\tilde{\mathcal{B}}$ is the component of $\mathcal{B}$ orthogonal to U, the $\tilde{U}$ and $\tilde{V}^\top$ come from the SVD of matrix $\tilde{U}\tilde{\Sigma}\tilde{V}^\top = \left[\begin{array}{cc} \Sigma & U^\top\mathcal{B} \\ 0 & \tilde{\mathcal{B}}^\top\mathcal{B} \end{array}\right]$. If we denote the means of $\mathcal{A}, \mathcal{B}, \mathcal{C}$ as $\bar{I}_\mathcal{A}, \bar{I}_\mathcal{B}, \bar{I}_\mathcal{C}$ and scatter matrices (defined as the outer product of the centered data matrix) as $\mathcal{S}_\mathcal{A}, \mathcal{S}_\mathcal{B}, \mathcal{S}_\mathcal{C}$, the problem caused by time-varying mean of the new coming data of the *SKL* algorithm can be corrected by using the following equation:

$$\mathcal{S}_\mathcal{C} = \mathcal{S}_\mathcal{A} + \mathcal{S}_\mathcal{B} + \frac{nm}{n+m}(\bar{I}_\mathcal{B} - \bar{I}_\mathcal{A})(\bar{I}_\mathcal{B} - \bar{I}_\mathcal{A})^\top \tag{2}$$

The mean of the $\mathcal{C}$ can be computed as $\bar{I}_\mathcal{C} = \frac{n}{n+m}\bar{I}_\mathcal{A} + \frac{m}{n+m}\bar{I}_\mathcal{B}$. If there is a forgetting factor f, $\bar{I}_\mathcal{C}$ can be modified as $\bar{I}_\mathcal{C} = \frac{fn}{fn+m}\bar{I}_\mathcal{A} + \frac{m}{fn+m}\bar{I}_\mathcal{B}$.

Then the tracking process is controlled by using a variant type of condensation algorithm [13]. Given a set of observed images $\mathcal{I}_t = \{\mathbf{I}_1, \mathbf{I}_2, ..., \mathbf{I}_t\}$, the estimation of the hidden data variable $\mathbf{X}_t$ which describe the affine motion transformation of the target at time t can be computed by

$$p(\mathbf{X}_t|\mathcal{I}_t) \propto p(\mathbf{I}_t|\mathbf{X}_t) \int p(\mathbf{I}_t|\mathbf{X}_{t-1}) p(\mathbf{X}_{t-1}|\mathcal{I}_{t-1}) d\mathbf{X}_{t-1} \tag{3}$$

Here, the affine motion transformation $\mathbf{X}_t$ is composed of six parameters including x, y translation, rotation angle, scale, aspect ratio and skew direction at time t. These parameters are updated continually as they evolve over time. However, at the beginning of tracking, the tracker need motion detection algorithm to detect the original location of the object, and during the tracking process, the detection algorithm calibrates the object location to help the tracker adjust the target's new position and also adds new tracker to the new incoming moving people. Our moving people detection algorithm based on latent semantic analysis is presented next.

2.2. Discovery of spatiotemporal latent semantic cues of moving people

The Fig.1 shows the flowchart of the discovery of spatiotemporal latent semantic cues of moving people. In this flowchart, the rectangle boxes denote the process and the ellipse boxes denote the process results. The whole process begins with detecting the spatial interest points in every video frame of training video by using the Harris [8] spatial interest point detection algorithm. The next work is to find an appropriate interest point descriptor that could represent the features of these points. Our work choose the Scale Invariant Feature Transform *(SIFT)* [10] to generate the features because *(SIFT)* features are scale and illumination invariant and also robust to the camera motion such as pan, tilt and zooming within a loose range. The output of this step is the spatial interest points and the *(SIFT)* feature information of each video frame, which is stored in a table called $SPIP\_FV$. Then all the *(SIFT)* features of the entire training video frames are clustered into several clusters to generate a codebook whose codewords are the clustering centers of all the generated *(SIFT)* features. Each feature is then represented by its codeword, the corresponding feature-codeword information

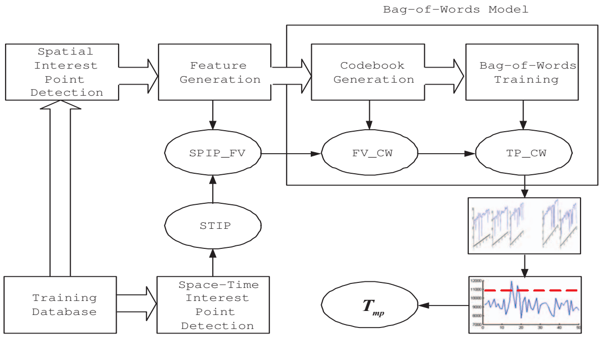

Fig. 1. spatiotemporal latent semantic cues discovery

are stored in a table called FV_CW. We then generate the 'bag-of-words' for every video frame which is the histogram of codewords within each frame.

How to make use of 'bag-of-words' to find the latent cues behind them is the reason why we employ the *pLSA* [11] for training. Let $t \in T = \{t_1, t_2, ..., t_k\}$ be the hidden/latent cues with each occurrence of codeword $c \in C = \{c_1, c_2, ..., c_m\}$ in an image $i \in I = \{i_1, i_2, ..., i_n\}$. The final output of the *pLSA* training process is a $P(t|c)$ conditional probability matrix TP_CW. After discovering the latent cues, the next step is to determine which cues should belong to the moving people. We then find the spatiotemporal interest points from same training images. The x,y position information of the obtained spatiotemporal interest points are used to find the corresponding *(SIFT)* feature in the table of $SPIP\_FV$. Next, the found *(SIFT)* feature indices are used to look up the corresponding codeword in the table FV_CW, and we record these codewords as $\{c_1^{'}, c_2^{'}, ..., c_j^{'}\} \in C_i^{st} \subset C$ and C_i^{st} is the spatiotemporal codeword collection and $i \in \{1, 2, ..., N\}$ is the training image number. Finally, the likelihood of each latent cues $\mathcal{L}^k$ to the moving people can be computed by using equation (4),

$$\mathcal{L}^k = \sum_{i=1}^{N} \mathcal{L}_i^k = -\sum_{i=1}^{N} \sum_{\substack{t_k \in T \\ c_j^{'} \in C_i^{st}}} \log P(t_k|c_j^{'}) \qquad (4)$$

Where the $P(t|c)$ comes from the conditional probability matrix TP_CW, k denotes the cue index. For some cue k, if $\mathcal{L}^k \geq \varepsilon_1$ (ε_1 is a threshold) is true, the cue k will be chosen as the moving people cue, all the discovered cues of moving people are denoted as T_{mp}. Discovery of the moving people cues is done during the training process, which is carried out before the tracking starts. We next present the tracking in detail.

2.3. Tracking with Latent Semantic Cues

The moving people detection and calibration process runs at the initialization and every tracking batch size interval of 5 frames. The process first detects spatial interest points in the frame and generates the *(SIFT)* features for each point. Then the codewords C_is corresponding to each features are picked out by using the Earth Mover's Distance [14]. Finally, whether each point is on the moving people's body or not is determined by $\sum_{k \in T_{mp}} P(t_k|c_i) \geq \varepsilon_2$ is true or not respectively, where ε_2 is a threshold.

Since a video frame can contain several moving people, the points belonging to different people are then separated. For this purpose the pairwise vertical and horizontal distance between the (moving people) points and the aspect ratio of group of points are computed first, which are then compared against the standard metric coming from the moving people templates which contain 406 types of different poses as shown in Fig.2. The measurement using the templates should consider setting the size of the templates proportional to the real size of people in the surveillance scenes. After the points are separated into different parts which denote different moving people, each center position of the people are computed by using the weighted mean of each group of points, the weight $\sum_{k \in T_{mp}} P(t_k|c_i)$ is the likelihood of the point's feature belonging to the moving people. The moving people regions are identified, which are then tracked using the incremental eigenspace tracking algorithm explained in section 2.1. In fact the tracking algorithm can go out of tracking due to accumulative errors. Hence, the tracking needs to be calibrated after every few frame interval of tracking batch size.

The calibration algorithm will check the centers and the scales(width and height) of the tracking region. If the drift distance of the target center accounts ξ portion of its scale, the calibration of the center will be $P_c = P_t \cdot \xi + P_d \cdot (1 - \xi)$, here the P_c is the new center of tracking region, the P_t is the old region center and the P_d is the center computed by calibration algorithm. For the tracking region, if the area difference between the tracking and calibration algorithm is over the ratio ζ, the region computed by calibration will replace the old one. In addition to the calibration, new moving people entering the scene also need to be detected regularly. Therefore, on every starting frame of the batch size interval (5 frames) moving people detection and calibration processes are run and on the intervening frames only the incremental eigenspace tracking algorithm is run.

3. EXPERIMENTS AND RESULTS

To verify the accuracy and robustness of proposed algorithm, we use the *KTH* database for training the cues. For track-

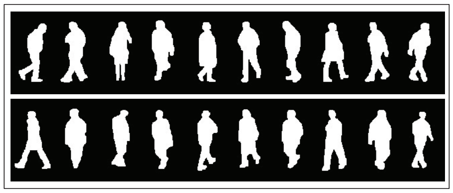

Fig. 2. Binary templates of moving people

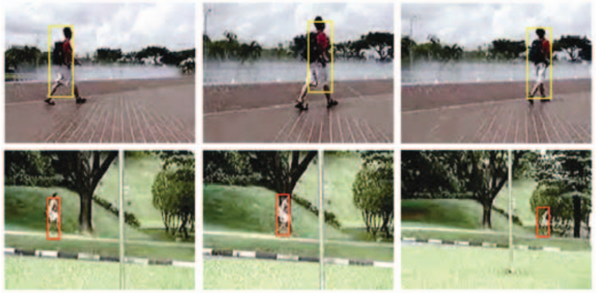

Fig. 3. Single people tracking in outdoor scenes

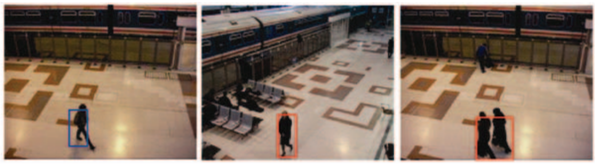

Fig. 4. Tracking results using *PETS2006* database

ing test, we use our own surveillance video for the outdoor test and *PETS2006* database for the indoor test. Fig.3 shows the tracking results of single people in outdoor scenes with zooming camera. As can be seen, the proposed method can track the moving people robustly even under dynamic background (spraying fountain and shaving trees) with zooming camera. Fig.4 shows tracking results in the indoor scenes using *PETS2006* database. The proposed method can also track single or single group of people in the shown scenes robustly. Fig.5 shows the multiple moving people tracking results using *PETS2006* database. The first row in Fig.5 shows the tracking results only using incremental eigenspace tracking algorithm, the second row shows the calibrated tracking results by using the proposed method. The comparison between the tracking results shows that with the calibration by using the latent semantic cues, the tracking region is more accurate than only using the incremental eigenspace tracking algorithm.

4. CONCLUSION

In this paper, we have proposed a method which is based on the latent semantic cues for the moving people tracking. Compared with only using the incremental eigenspace tracking algorithm, the results of experiments verify that with the help of the latent semantic cues, the proposed method can automatically and robustly track the moving people in a complex background for the indoor and outdoor environment.

5. REFERENCES

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.*, vol. 38, no. 4, pp. 13, 2006.

[2] M.J. Black and A.D. Jepson, "Eigentracking: Robust matching and tracking of articulated objects using a view-based representation," *International Journal of Computer Vision*, vol. 26, no. 1, pp. 63–84, 1998.

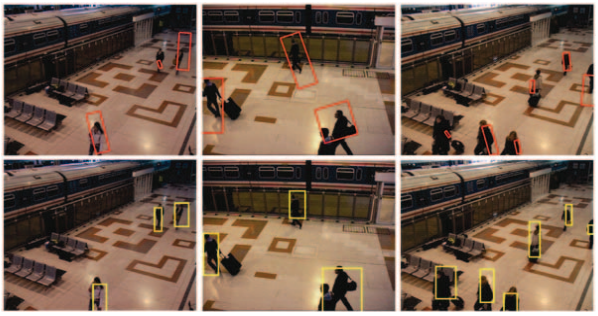

Fig. 5. Comparison between the tracking results using *PETS2006* database

[3] A.D. Jepson, D.J. Fleet, and T.F. El-Maraghi, "Robust online appearance models for visual tracking," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, no. 10, pp. 1296–1311, Oct. 2003.

[4] J. Lim, D.Ross, R.S. Lin, and M.H. Yang, "Incremental learning for visual tracking," *Advances in Neural Information Processing Systems (NIPS) 17*, pp. 793–800, 2005.

[5] Z.F. Zhang X.Q. Zhang X. Li, W.M. Hu and G. Luo, "Robust visual tracking based on incremental tensor subspace learning," *Computer Vision. IEEE International Conference on*, pp. 1–8, Oct. 2007.

[6] X.Q. Zhang, W.M. Hu, S. Maybank, and X. Li, "Graph based discriminative learning for robust and efficient object tracking," *Computer Vision, 2007. IEEE International Conference on*, pp. 1–8, Oct. 2007.

[7] K. Ishiguro, T. Yamada, and N. Ueda, "Simultaneous clustering and tracking unknown number of objects," *Computer Vision and Pattern Recognition. IEEE Conference on*, pp. 1–8, June 2008.

[8] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, 2004.

[9] I. Laptev, "On space-time interest points," *International Journal of Computer Vision*, vol. 64, no. 2-3, pp. 107–123, 2005.

[10] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[11] T. Hofmann, "Probabilistic latent semantic indexing," *Proc. International ACM SIGIR Conference*, pp. 50–57, 1999.

[12] A. Levy and M. Lindenbaum, "Sequential karhunen-loeve basis extraction and its application to images," *Image processing, IEEE Transactions on*, vol. 9, pp. 1371–1374, 2000.

[13] M. Isard and A. Blake, "Condensation – conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.

[14] C.Tomasi Y. Rubner and L.J. Guibas, "The earth mover's distance as a metric for image retrieval," *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99–121, 2000.