13

Targets Tracking in the Crowd

Cheng-Chang Lien
Department of Computer Science and Information Engineering,
Chung Hua University,
Taiwan

1. Introduction

Conventional video surveillance systems often have several shortcomings. First, target detection can't be accurate under the light variation environment or clustering backgrounds. Second, multiple targets tracking become difficult on a crowd scene because the split/merge and occlusions among the tracked targets occur frequently and irregularly. Third, it is difficult to the partition the tracked targets from a merged image blob and then the target tracking may be inaccurate. In this chapter, the methods for targets detection and tracking in the crowd are addressed. In general, the methods for targets tracking in the crowd can be categorized into the blob-based and point-based methods. The blob-based methods detect and track the targets based on the appearance models; while the point-based methods detect and track the moving targets with the reliable feature points.

2. Blob-based methods

In the conventional target detection systems, some typical methods are applied to extract the moving objects, e.g., background subtraction [1], and pixel-wise temporal difference analysis [2]. However, these methods are extremely sensitive to the variation of lighting or the dynamic background changing. Applying pixel-wise temporal differencing [4] may reduce the influence of the dynamic illumination change, but the regions of the moving objects are extracted incompletely when the background variation occurs. All the above mentioned methods do not utilize the motion information including the object and camera motions [3]. By applying the method of optical flow [2], the moving objects may be detected even in the presence of camera motion. However, the high computation complexity makes the real-time object detection difficult. In this study, we apply the pixel-wise temporal statistical model [5], voting rule for the Y, C_r, and C_b Bayesian classifiers, and foreground verification with the dynamic texture modeling to detect the targets on the crowd scene accurately.

In most target tracking systems, central point on the target is used as the reference location to predict the position at the next frame. However, central point can be influenced easily by the inaccurate foreground detection. Here, the methods of principle-axis detection [9] is applied to extract the ground point of each target to serve as the reference point in the target tracking algorithms, e.g., Kalman filter [6-7] and Particle filter [8]. Fig. 1 illustrates the block diagram of the proposed system.

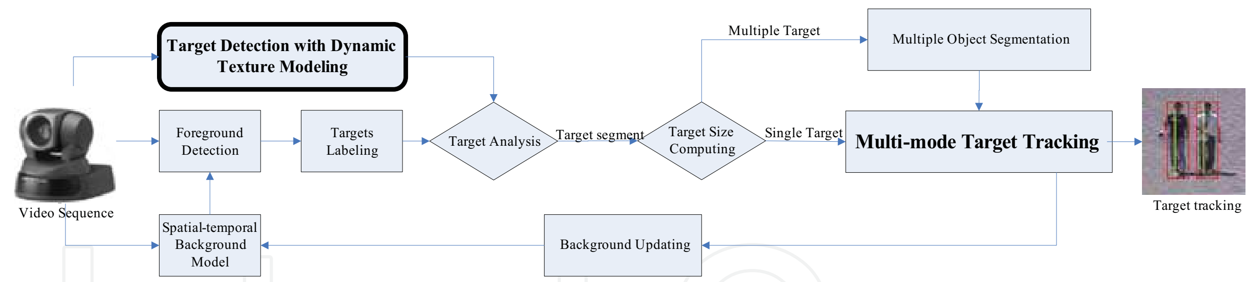


Fig. 1. The block diagram of the proposed system.

2.1 Temporal probability background model

In this section, the background model fusing with temporal and texture models is established to segment the foreground and background on a light variant or clustering background. In an image sequence, the intensity variation within a time period for each pixel can be modeled by the Gaussian distribution function [4]. The pixel-based MOG function is defined as:

$$p(I) = p(I \mid B)P(B) + \sum_{j=1}^{c-1} p(I \mid \omega_j)p(\omega_j)\,, \tag{1}$$

where, I is the intensity value, B denotes the background, ω_j denotes the moving object and c denotes the number of Gaussians. The intensity distribution of the background pixel at a certain position x_b can be expressed as:

$$p(x_b \mid B) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(I - \bar{I}(x_b))^2}{2\sigma^2}\right), \tag{2}$$

where $\bar{I}(x_b)$ and σ are the mean and standard deviation of the pixel intensity at x_b. According to the Bayesian decision rule [9], whether the pixel belongs to the background or the foreground (the moving objects) can be determined by the following likelihood inequality.

$$\frac{p(I \mid B_{x_b})}{p(I \mid T_{x_b})} \geq \frac{P(T)}{P(B)} = \lambda\,, \tag{3}$$

where $P(T)$ and $P(B)$ are the prior probabilities for the background and moving objects respectively. By replacing $p(I \mid B_{xb})$ with Eq. (2), the likelihood ratio can be further simplified as:

$$\left| I - \bar{I}(x_b) \right| \leq k\sigma\,, \tag{4}$$

where $k = \sqrt{-2\ln(\sqrt{2\pi}\,\sigma\lambda/L)}$. If $| I - \bar{I}(x_b) | \leq k\sigma$, then the pixel is categorized as the background, otherwise, the pixel is categorized as the foreground.

2.2 Foreground detection rule

It can be very difficult to detect the moving objects when the intensity distribution is closed to the background model. The fusion of likelihood ratios of three color components (RGB or YC_rC_b) are then proposed to overcome this problem. In general, linear combination and voting rules are applied to detect the foreground, which are described in the following.

Linear combination rule:
If $w_y\, p_Y(u_y|B_x)+w_{cr}\, p_{Cr}(u_{Cr}|B_x)+w_{cb}\, p_{Cb}(u_{Cb}\ |B_x) > T$
pixel u is classified as background,
otherwise,
pixel u is classified as foreground,
where, w_y , w_{cr} , w_{cb} are weighting factors and sum of the weighting factors is equal to one.

Voting rule:
Given $p_Y(u_y|B_x)$, $p_{Cr}(u_{Cr}|B_x)$, $p_{Cb}(u_{Cb}\ |B_x)$, If a pixel is classified as background with more than two components' background models,
Pixel u is classified as background,
otherwise,
Pixel u is classified as foreground.

By comparing the above two fusion rules, we apply the voting rule to cope with the illumination variation problem. If a pixel is classified as background with more than two components' background models, then this pixel is classified as the background, otherwise, it is classified as the foreground. Fig. 2 illustrates the foreground detection using the voting rule. It is obvious that the foreground detection using voting rule outperform the one using the linear combination rule. Hence, we apply the voting rule to detect the objects on the outdoor crowd scene to cope with the illumination variation problem.

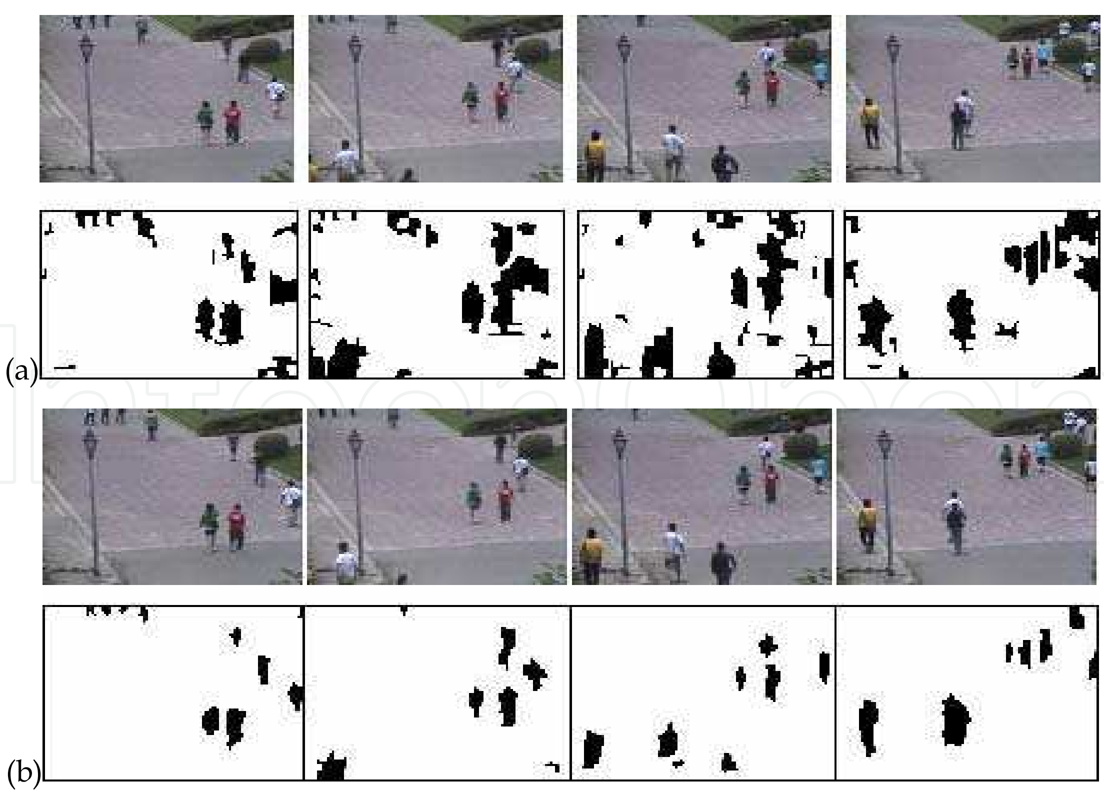


Fig. 2. (a) Foreground detection using the linear combination rule. (b) Foreground detection using the voting rule.

2.3 Foreground verification using texture modeling

Many environmental dynamic textures such as leaves, fire, smoke, and sea waves may reduce the accuracy of target detection. Here, the dynamic texture will be modeled by using the modified local binary pattern (LBP)[11] and then the target can be detected without the influence of dynamic textures in the crowd scene. Here, a local texture pattern T [11] centering the pixel g_c and having P neighboring pixels is defined as:

$$T \approx t(s(g_0 - g_c), s(g_1 - g_c), \ldots, s(g_{P-1} - g_c)) , \tag{5}$$

where

$$s(x) = \begin{cases} 1, |x| \geq threshold \\ 0, |x| < threshold \end{cases} . \tag{6}$$

Then, we transform the modified LBP in (5) to an integer value with the formula in Eq. (7).

$$LBP_{PR} = \sum_{0}^{P-1} s(g_p - g_c) 2^P . \tag{7}$$

Finally, the modified LBP is utilized to model the dynamic texture background and remove the false foregrounds. In the LBP-based foreground detection, the bit difference η between the captured scene and LBP-based background model can be used to separate the foreground from the background. The LBP-based foreground detection rule is defined as:

$$P^{frame(t+1)}(\eta) = \begin{cases} foreground, & if \quad \eta \geq \eta\_th \\ background, & if \quad \eta < \eta\_th \end{cases} . \tag{8}$$

The bit difference η is calculated as:

$$\eta = \sum_{p=0}^{8} (LBP_p^{frame(t+1)} \ XOR \ LBP_p^{frame(t)}) \tag{9}$$

where, p is the index of the pixel on the circular chain. In this study, both the pixel-wise temporal probability model and LBP texture model are constructed to detect the foreground, but how to integrate both background models to reduce the false detection is a very important issue. Based on the careful observation of foreground detections, the foreground detection rule is then designed as:

If $R(O_c) \in$ **foreground**
  count $R(F_c^{LBP} | O_c)$,
  If $N(R(F_c^{LBP} | O_c)) > N_{th}$,
    $O_c \in$ **True foreground**,
    update $R(O_c)$,
  else
    $O_c \in$ **False foreground** ,
    clear $R(O_c)$,
Endif

where, $R(O_c)$ denotes the region of a detected object using the pixel-wise temporal probability model on the current frame c, $R(F^c_{LBP} | O_c)$ denotes the region of the foreground detected by pixel-wise LBP texture model on the current frame c around the region of O_c. In order to remove the false detections, we propose the update/clear method as follow:

$$\mathbf{R'(O_c)} = \begin{cases} R(O_c) \ \cup \ \mathbf{R(F_c^{LBP} | O_c)} \ \mathbf{, \text{ if Update}}_{foreground} \ \ \mathbf{\text{is choosed.}} \\ \quad Null \qquad\qquad\quad \mathbf{, \text{ if Clear}}_{foreground} \ \ \mathbf{\text{is choosed.}} \end{cases} \tag{10}$$

Consequently, we can not only detect the object regions precisely, but also can remove the false foregrounds.

2.4 Multi-mode target tracking

The targets tracking become complex on a crowd scene because the split and merging or occlusion conditions among the tracked targets occur frequently. In addition, the targets appear on the scene at the first time may be a single target or a merged multiple targets. In this study, the bottom-up tracking scheme is applied to develop the multi-modes tracking scheme. Each detected image blob is classified into single or multiple targets according to its area and then the tracked targets are examined whether they belong to the targets appeared on previous frames. Based on status classifications of target tracking on a crowd scene, there are six target tracking modes [13] shown in Fig. 3 are described as follows.

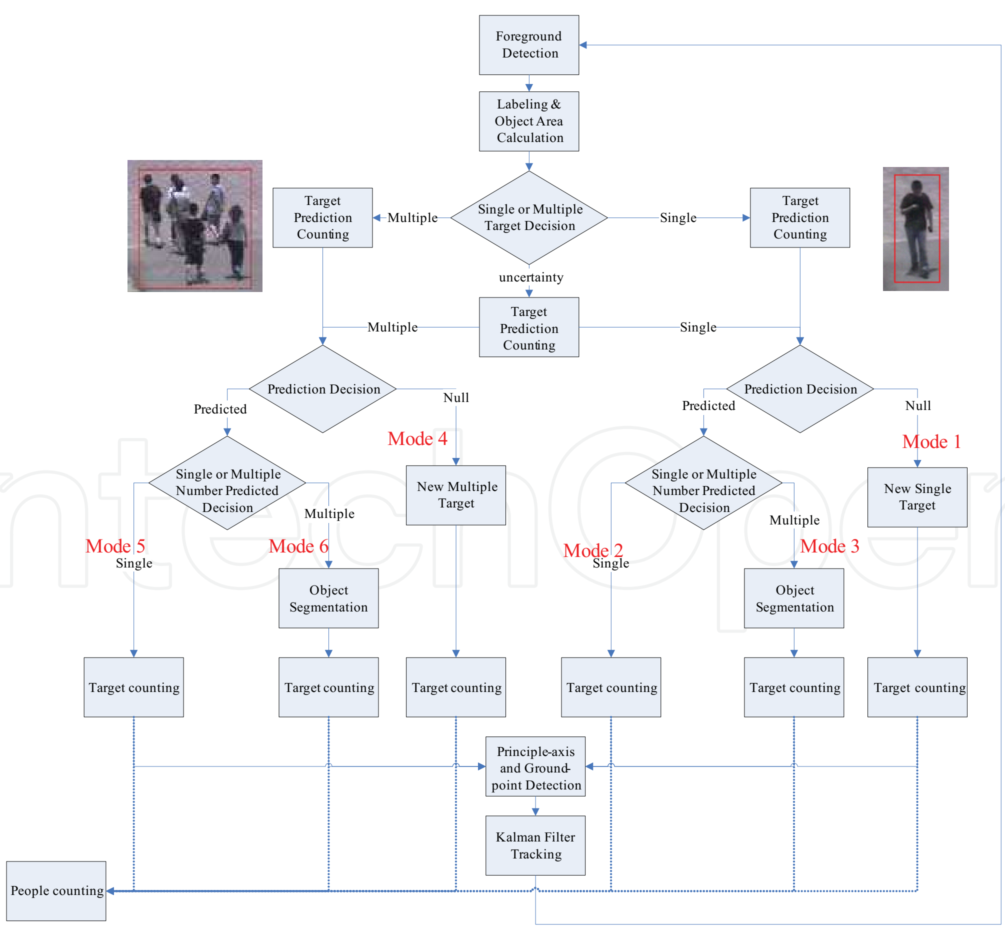


Fig. 3. Flowchart of the multi-mode multi targets tracking scheme.

Mode 1: An image blob is detected as a single target and its location is not predicted by other tracked targets from previous frames, i.e., the target is appeared on the scene at first time.
Mode 2: An image blob is detected as a single target and its location is predicted by one of the tracked targets from previous frames.
Mode 3: An image blob is detected as a single target and its location is predicted by a multiple target from previous frames, i.e., the object occlusion is occurred.
Mode 4: An image blob is detected as merged multiple targets and its location is not predicted by other tracked targets from previous frames, i.e., the merged multiple targets is appeared on the scene at first time.
Mode 5: An image blob is detected as a merged multiple targets and its location is predicted by one of the tracked merged multiple targets from previous frames.
Mode 6: An image blob is detected as a merged multiple targets and its location is predicted by a single target from previous frames, i.e., the objects' separation is occurred.
The detailed description of the multi-mode target tracking can be seen in [13].
When the tracked targets are slightly occluded it is possible to separate these targets. Then, the separated target can be tracked according to rules of modes 1 or 2. In general, color feature is effective to separate the merger targets. However, to develop robust target segmentation we apply the color-based difference projection to separate the targets from the merged targets. By observing the color difference projection histogram, we found the peak is not distinct for each color feature. To overcome this problem, the correlation for the color feature is used to find the segmentation line. In Fig. 4, the position of the prominent correlation peak is obtained to extract the segmentation line.

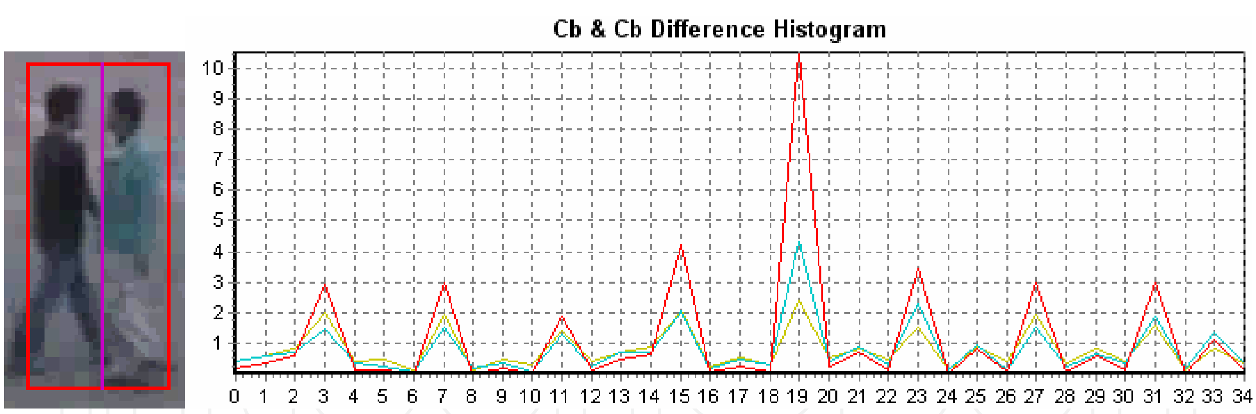
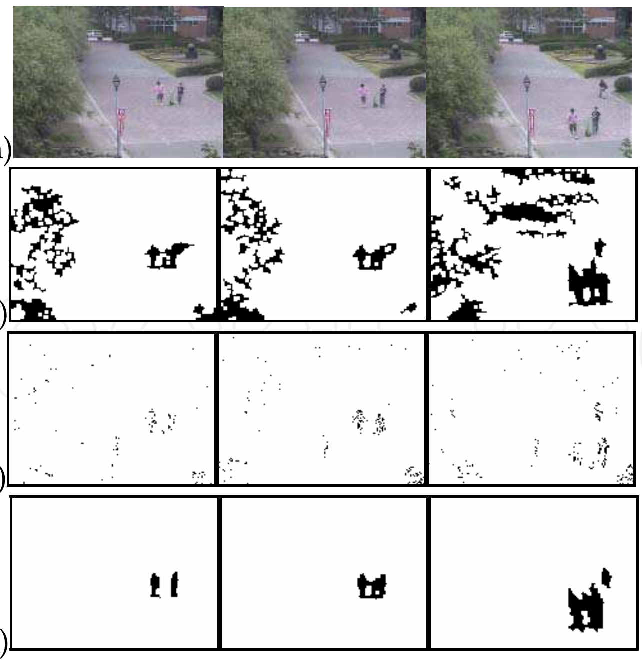


Fig. 4. The correlation for the color feature is used to find the segmentation line

2.5 Simulation results

2.5.1 Moving object filtering using the dynamic texture model

In Fig. 5-(a), it shows an outdoor scene. Fig. 5-(b) represents the foreground with the pixel-wise temporal probability model, and the dynamic texture detection model is described in Fig. 5-(c). By using the dynamic detection model, the targets will be separated into the truly foreground target and the constant texture object. If the object with too many constant textures, we will define the target as the noise, and it then will be removed, i.e. in Fig. 5-(d). Finally, we can improve the accuracy about the detected target.



(a) (b) (c) (d)

Fig. 5. (a) Outdoor scene. (b) The objects are detected by using pixel-wise temporal probability model. (c) Foreground detection using the dynamic texture model. (d) The extracted objects after the texture noise removing process.

2.5.2 Multi-mode target tacking scheme

In Fig. 6, the multi-mode target tracking on a crowd outdoor scene is illustrated. The split, merge, and occlusion among the targets occur repeatedly. The merged multiple-target is labeled "M". Meanwhile, some important features, e.g., color, weight, height, ground point position, are record as the tracking measurements for each target. The multi–mode multi targets detection on a crowd outdoor scene is shown in Fig. 7 and each target will be tracked with the mode according to the situation of target occlusion [13].

(a) (b)

Fig. 6. Multi-mode tracking on an outdoor crowd scene.

Fig. 7. Multi-mode tracking on an outdoor crowd scene.

3. Point-based method

In the conventional blob-based object detection systems, some typical methods are applied to extract the moving objects, e.g., background subtraction [15], optical flow [16-18], frame difference analyses [19], and codebook model [20]. In [15], the regions of moving objects may be acquired precisely by using the method of background subtraction but it is extremely sensitive to the illumination variation and the dynamic background changing. In [16-18], the optical flow method is used to independently track each low-level object feature. Applying frame differencing method [19] may be adaptive to the illumination changes, but the moving objects are extracted incompletely when the objects move slowly. In [20], the codebook method can overcome the problems of changing backgrounds or illumination variations, but this method is difficult to detect and track the targets in the crowded scene.

In the novel video surveillance application, one of the most challenging problems is the target tracking in the crowded scene shown in Fig. 8. Generally, the serious occlusions make the conventional blob-based target detection/tracking methods failed. For example, to track a target in a crowded rail station or square, the tracked person can be partially occluded by other persons and only part of region can be served as a clue to track. Hence, to track the individual target in the dense crowds may face two major problems: 1) the target size can be small when we are monitoring a large space where the crowds move; 2) frequent partial occlusions make the target segmentation very difficult. Therefore, the point-based features are considered to be applied to tackle the problems of tracking in the crowd. Even in the crowded scenes, there may still have some feature points on the partial target regions that are not occluded. In Fig. 9-(b), the example of object extraction using the blob-based method is illustrated. It is obvious that the foreground region formed by merging several targets is difficult to segment each target. On the contrary, Fig. 9-(c) shows the feasibility for applying the point-based method to detect and track each individual target in the crowded scene.

With the same concept of feature point tracking, Brostow and Cipolla [21] proposed a Bayesian clustering algorithm that can segment each individual entity within the crowd with the space-time proximity and trajectory coherence. However, the efficiency is unsatisfied. In this study, we propose a novel method to detect and track the individual target in the crowd.

Fig. 8. (a) Crowded scene in tunnel. (b) Crowded scene in Chu-Hua University. (c) Occlusion occurs in the scene.

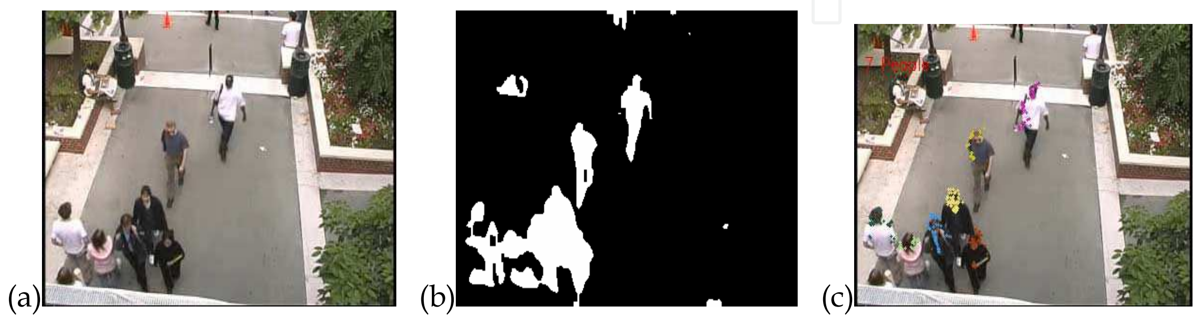


Fig. 9. (a) Original image. (b) The foreground detection with the method of blob-based background subtraction. (c) Target detection and tracking using the point-based feature and each color represents each individual target.

3.1 Point-based target tracking

In general, the conventional object detection in the crowd may have several problems in the individual segmentation process. First, it's difficult to find accurate boundaries by using background subtraction methods [23] in the crowded scenes. Second, the supervised learning or any subject-specific model [20] needs more computation cost to train. Third, the moving subjects may have different moving directions but merge together. In order to tackle these problems, we propose a coarse-to-fine approach based on the corner points extraction and tracking processes, in which the C-means algorithm is used to extract the coarse clusters and the spatial-temporal shortest spanning tree is proposed to segment each individual subject.

In the proposed system framework, we firstly detect the low-level feature points with the Shi-Tomasi-Kanade detector [16]. In the Kanade's algorithm, once the corner points are extracted, each feature point can be tracked by the Kanade-Lucas-Tomasi(KLT) optical flow [22] between two successive frames. Each trace of the tracked corner point can be represented as:

$$\{\{(x_t^i, y_t^i), t = T_{init}^i, \ldots, T_{final}^i\}, i = 1, \ldots, N\},$$

where (x_t^i, y_t^i) denotes the image coordinate of the corner point i at frame t, N denotes the total number of feature point tracks, and T denotes the frame number. By the careful observation, the corner points on each individual object can have high spatial-temporal correlation. The spatial correlation measures the geometrical relationship among the corner points that belong to an object; while the temporal correlation measures the trajectory consistency for the corner points that belong to an object. Here, the C-means algorithm is used to roughly cluster the dynamic feature points based on the spatial correlation measure.

3.1.1 Individual segmentation with spatial-temporal shortest spanning tree

Based on the rough segmentations from the C-means clustering, the corner points located on a subject should be close on both the geometrical distribution and moving trajectory. To this end, the cluster refining process implemented with spatial shortest spanning tree is applied to the rough segmented clusters to partition the individual objects. The algorithm of individual segmentation with the spatial shortest spanning tree [25] is described as follows.

1. Construct a spatial shortest spanning tree in each cluster. Let set $C^i = \{p^1, p^2\ldots, p^N\}$ represent the point set within a cluster where N denotes the total number of cluster members.
2. Calculate the weighted distance according to the following formula.

$$d_{ij} = \left\{ \sqrt{\beta_x \left(p_x^i - p_x^j\right)^2 + \beta_y \left(p_y^i - p_y^j\right)^2} \;, i = 1,2,\ldots,N \;, j = 1,2,\ldots,N \right\},$$

   where β_x, β_y are weights on x and y directions separately.

1. According to the distance calculation in step 2, construct the spatial shortest spanning tree sequentially for each cluster.
2. Retain the weighting table and the linking order in the spatial shortest spanning tree for constructing the spatial-temporal shortest spanning tree.

In the following, we will modify this algorithm by combining the trajectory correlation to segment the individual more precisely.

Given two tracks T_u and T_v generated from two feature points, if the track variance defined in Eq. (11) for the two tracks is small, then the two feature points are likely to belong to same target.

$$Correlation\left(T_u \;, T_v\right) \;=\; \frac{1}{1 \;+ Variance\left(T_u \;, T_v\right)}, \tag{11}$$

where $Variance(T_u \,,T_v)$ = $Variance(DistanceEucl(T_u \,,T_v))$ within N frames. Finally, we can establish a spatial-temporal conformance measure as:

$$Conformance \;= \frac{\sqrt{\beta_x \left(p_x^i - p_x^j\right)^2 + \beta_y \left(p_y^i - p_y^j\right)^2}}{Correlation\left(T_i \;,\; T_j\right)}, i = 1,2,\ldots,N-1 \;, j = i+1, \tag{12}$$

If the conformance value is larger than a predefined threshold, then it means that the two points don't belong to the same subject. But sometimes we may face a situation that a feature point belong to subject A may be mis-located on subject B when two subject are crossing. Fig. 10 illustrates that a lot points are possible to be mis-located when two person are crossing each other. To overcome the above-mentioned problem, we propose the voting method with temporal information to ensure the trajectory conformance of tracked feature points in a cluster. The voting rule for trajectory conformance is described as follows.

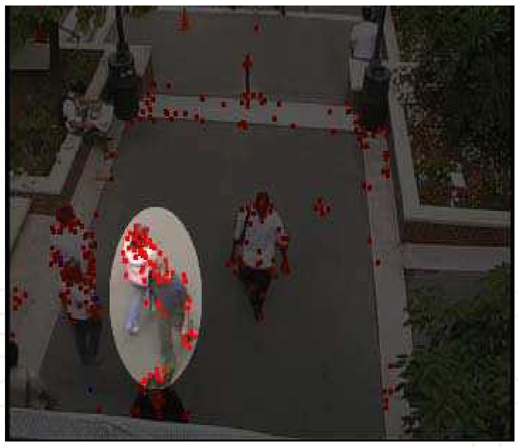


Fig. 10. A lot points are mis-located when two person are crossing each other.

1. The vector $\boldsymbol{d}_{(x,y)}(F_t, F_{t-10})$ acquired from successive 10 frames is defined to record the moving direction for each feature point.
2. Based on the x and y components of $\boldsymbol{d}_{(x,y)} = (d_x, d_y)$, there are four possible sign pairs to denote the rough moving direction shown in Fig. 11. They are defined as (+ , +), (+ , -), (- , +), and (- , -) separately. We record the direction of all dynamic points that belong to each subject.
3. By observing the recorded direction list, there may have a direction with the highest votes. We assume that it is the dominant moving direction in this cluster, and then delete other dynamic points with different directions.
4. Repeat steps 1~3 until all cluster are processed.
5. Once the cluster's direction list is modified, the cluster center must be recalculated.

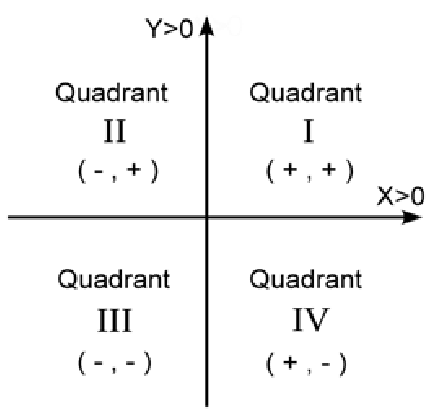


Fig. 11. The possible moving direction for each feature point.

3.1.2 Object tracking

In order to accurately track all targets, to develop a stable and reliable point-based object tracking method is necessary. According to the mechanism of feature tracking, we know the corner feature would be tracked efficiently between two successive frames. Therefore, the characteristic of KLT tracking algorithm are adopted for tracking the existed dynamic feature points with the inheritance of the points' attributes in last frame. On the other hand, the new points can appear while the existed points' tracking fail or vanish. Hence, it's important to link the tracking relation between old and new points.

Point-Based Target Tracking Algorithm

1. If the corner points are tracked successfully and continuously, they will inherit the attribute of the segmented individual cluster. Otherwise, if the corner points are newly generated in the area of ROI, we will classify the feature points into the set $P_{NEW} = \{p^1, p^2 \ldots p^N\}$. Furthermore, the distance between the new points and existed cluster centers C_j are calculated as:

$$Distance = \left\{ \left\{ \sqrt{\alpha_x \left(p_x^i - C_x^j\right)^2 + \alpha_y \left(p_y^i - C_x^j\right)^2} \;, i = 1,2,\ldots(5),N \right\} , j = 1,2,\ldots,N \right\}, \quad (13)$$

where a_x and a_y are weights on x-axis and y-axis that are similar to the C-means clustering process. Then, we add the new points into the closest cluster.

2. After adding the new points to the nearest cluster, the cluster centers may change. In addition, the situation that multiple objects moving together may occur and then the individual segmentation are required to perform again. The individual segment mechanism is based on normal width of human body. First we calculate the width of the cluster as:

$$ClusterWidth \;=\; \mathbf{RightBoundary_y} \mathbf{\ -\ LeftBoundary_y}.$$

If the width of the cluster is larger than the predefined threshold TWIDTH (evaluated by the normal width of human body), then the cluster is required to be segmented. The new segmented cluster will be given a new ID. Fig. 12 illustrates the situation that needs to be re-segmented.

3. If some new points are not classified into any existed clusters, then theses point are classified into the new set $P'_{NEW} = \{p'^1, p'^2 \ldots p'^N\}$ and regarded as new moving objects. Fig. 13 illustrates existed cluster (existed objects) and the new clusters (new objects). For this reason, we use C-means algorithm to classify them but without comparing to the existed clusters.
4. Integrate the new and old clusters' information.
5. Check the consistence of all points in each cluster and execute the process of individual segmentation with the spatial-temporal shortest spanning tree for each cluster. Fig. 14 shows the final results.

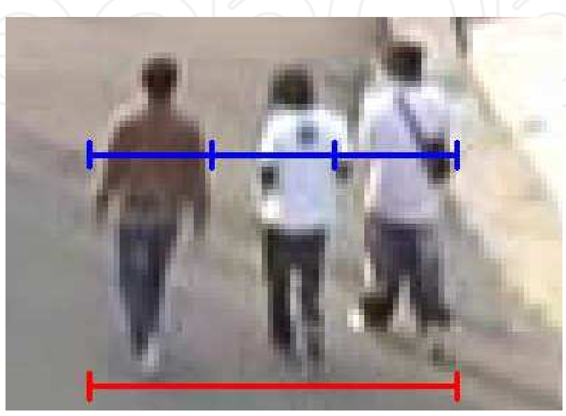


Fig. 12. The red line denotes the cluster width. The blue line denotes the normal width of human body.

Fig. 13. Grouping of new points. (a) Green points are the new generated points that are distant from other existed red clusters. (b) The result of clustering.

Fig. 14. Final result of segmenting the tracked objects

3.2 Simulation results

Two test videos: "Commons01" and "Tunnel-A125" are used to evaluate the performance of individual segmentation and object tracking. The video "Commons01" is obtained from the web site in [24]. Another test video "Tunnel-A125" is provided from professor Brostow [21], which is captured from a tunnel where the people are walking side by side. The object tracking is based on the concept of feature points' inheritance. The proposed point-based object tracking algorithm can make the target tracking more accurate and efficient. Fig. 15 illustrates the experimental results for the point-based object tracking. In Fig. 15, there are 3 people walking closely and the occlusion problem is serious. The proposed method can segment each individual person even the occlusion problem exists.

In order to evaluate the accuracy of the proposed system, we compare the proposed method to other methods proposed in recent years. Table 1 shows the accuracy analysis among the methods in [21, 26] and ours. It can be seen that the performance in detection rate and miss detection rate outperforms the other methods and the false detection rate is close to the other methods. In the accuracy analysis, we select 400 successive frames from the video "tunnel-A125". The efficiency of our system can approach 8 fps and it can be further improved with the adjustment of the amount of feature points. The method of Bayesian detection [21] selected the test video frames from "tunnel-A125" randomly. In order to calculate the likelihoods between many image features, the computational complexity of the method in [21] is higher. They take about 5 seconds to perform the spatial clustering process for each frame. In our experiment, we analyze and count the detection rate in the area between the blue lines. Some results of individual segmentation are shown in Fig. 16.

frame #247 frame #307 frame #637

Fig. 15. Segmentation and tracking for the targets in the crowd.

frame #2762 frame #2981

Fig. 16. Analysis of the detection rate in the area between the two blue lines.

	Brostow & Cipolla	Zhao & Nevatia	Ours
distinct detections	144	8466	1319
correctly detected	136	7881	1254
missed detections	8	585	65
false detections	33	291	56
detection rate	94%	93.09%	95.07%
miss detection rate	22.9%	6.91%	4.92%
false detection rate	5.6%	3.43%	4.25%

Table 1. The accuracy analyses for the methods of Brostow & Cipolla, Zhao & Nevatia, and ours.

4. Conclusions

In this chapter, the methods for targets detection and tracking in the crowd are addressed. In general, the methods for targets tracking in the crowd can be categorized into the blob-based and point-based methods. The blob-based methods detect and track the targets based on the appearance models; while the point-based method detect and track the moving targets with the reliable feature points. For the blob-based method, the spatial-temporal probability background model, multi-mode tracking scheme, color-based difference projection, and ground point detection are proposed to improve the conventional target tracking systems. Experimental results show that the targets in the crowd scene may be tracked with the correct tracking modes and with rate above 15fps. For the point-based method, a novel system for tracking in the crowd is proposed. The spatial-temporal shortest spanning tree and target tracking with point inheritance are proposed to improve the problems for the object tracking in the crowd. The experimental results show that the accuracy of individual segmentation in the crowd can be higher than 90%, and the efficiency of our system can approach 8 fps. The future works in the subsystem of tracking in crowd is focus on refining the accuracy of targets' segmentation and tracking.

5. Reference

[1] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction," in *Proceedings of the 6th European Conference on Computer Vision, 2000,* pp. 751-767.

[2] X. Dai, and S. Khorram, "Performance of optical flow techniques," *Int J Compute Vision 12(1),* pp. 42-77, 1994.

[3] R. Jain, W. Martin and J. Aggarwal, "Segmentation through the detection of changes due to motion," *Compute Graph Image Process 11, 1979,* pp. 13–34.

[4] Y. Ren, C. S. Chua and Y. K. Ho, "Motion detection with nonstationary background," *Machine Vision and Application, Mar. 2003,* Vol. 13, No. 5-6, pp. 332–343.

[5] C. C. Lien and S. C. Hsu, "The target tracking using the spatial-temporal probability model," *IEEE Nonlinear Signal and Image Processing, NSIP 2005, May 2005*, pp. 34-39.

[6] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence, July 1997,* Vol. 19, No. 7, pp. 780-785.

[7] M. Xu, J. Orwell, L. Lowey and D. Thirde, "Architecture and algorithms for tracking football players with multiple cameras" *Image and Signal Processing, IEE Proceedings, 8 April 2005,* Vol. 152, Issue 2, pp. 232-241

[8] K. Nummiaro, E. K. Meier, L. J. V. Gool, "An adaptive color-based particle filter" *Journal: Image Vision Computing,* Vol. 21, Issue. 1, pp. 99-110.

[9] W. Hu, M. Hu, X. Zhou, Tieniu Tan, "Principal axis-based correspondence between multiple cameras for people tracking" *IEEE Transactions on Pattern Analysis and Machine Intelligence, APRIL 2006,* Vol. 28, No. 4, pp. 663-671.

[10] D. Salmond, "Target tracking: introduction and Kalman tracking filters," *IEEE Target Tracking: Algorithms and Applications, 2001,* Vol. 2, pp. 1/1-1/16.

[11] E. Alpaydin, Introduction to Machine Learning, *MIT Press, Cambridge 2004.*

[12] Y. Yang and M. Levine, "The background primal sketch: an approach for tracking moving objects," *Machine Vision and Applications, 1992,* Vol. 5, pp. 17-34.

[13] C. C. Lien, J. C. Wang and Y. M. Jiang, "Multi-mode target tracing on s crowd scene," *IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing 2007 (IIH-MSP 2007),* Nov. 26-28, Kaohsiung, Taiwan.

[14] G. J. Brostow and R. Cipolla, "Unsupervised Bayesian detection of independent motion in crowds," in Proc. of IEEE Conference on Computer Vision and Pattern Recognition, 2006, vol. 1, pp. 594-601.

[15] C. R. Jung, "Efficient Background Subtraction and Shadow Removal for Monochromatic Video Sequences," *IEEE Transactions on Multimedia*, Vol. 11, No. 3, April 2009, pp. 571-577.

[16] J. Shi and C. Tomasi, "Good Features to Track," *Proceedings of the IEEE Conference Computer Vision and Pattern Recognition*, June 1994, pp. 593-600.

[17] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, Vol. 60, No. 2, Jan. 2004, pp. 91-110.

[18] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded-Up Robust Features (SURF)," *Proceedings of the 9th European Conference on Computer Vision*, Vol. 3951, May 2006, pp. 404-417.

[19] S. Wang, X. Wang, and H. Chen, "A Stereo Video Segmentation Algorithm Combining Disparity Map and Frame Difference," *International Conference on Intelligent System and Knowledge Engineering*, Vol. 1, Nov. 2008, pp. 1121-1124.

[20] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis, "Real-Time Foreground–Background Segmentation Using Codebook Model," *Real-Time Imaging*, Vol. 11, No. 3, June 2005, pp. 172-185.

[21] G. Brostow and R. Cipolla, "Unsupervised Bayesian Detection of Independent Motion in Crowds," *Proceedings of the IEEE Conference Computer Vision and Pattern Recognition*, Vol. 1, June 2006, pp. 594-601.

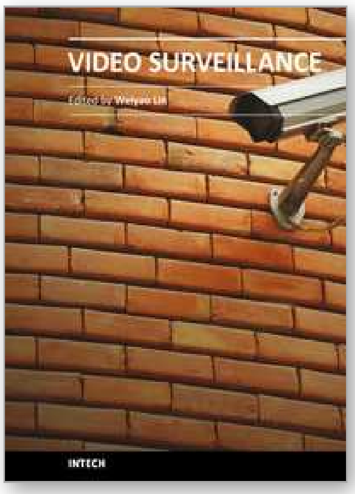
[22] C. Tomasi and T. Kanade, "Detection and Tracking of Point Features," *Carnegie Mellon University Technical Report CMU-CS-91-132*, April 1991.

[23] J. Rosen, "A Cautionary Tale for A New Age of Surveillance," *The New York Times Magazine*, Oct. 2001.

[24] A. M. Cheriyadat and R. J. Radke, "Detecting Dominant Motions in Dense Crowds," *IEEE Journal of Selected Topics in Signal Processing*, Vol. 2, No. 4, Aug. 2008, pp. 568-581.

[25] O. J. Morris, M. J. Lee, and A. G. Constantinides, "Graph theory for image analysis: an approach based on the shortest spanning tree," *IEE Proceedings F. Communications, Radar and Signal Processing*, Vol. 133, No. 2, April 1986, pp. 146-152.

[26] T. Zhao and R. Nevatia, "Tracking Multiple Humans in Complex Situations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 26, No. 9, Sep. 2004, pp. 1208-1221.



Video Surveillance

Edited by Prof. Weiyao Lin

ISBN 978-953-307-436-8
Hard cover, 486 pages
Publisher InTech
Published online 03, February, 2011
Published in print edition February, 2011

This book presents the latest achievements and developments in the field of video surveillance. The chapters selected for this book comprise a cross-section of topics that reflect a variety of perspectives and disciplinary backgrounds. Besides the introduction of new achievements in video surveillance, this book also presents some good overviews of the state-of-the-art technologies as well as some interesting advanced topics related to video surveillance. Summing up the wide range of issues presented in the book, it can be addressed to a quite broad audience, including both academic researchers and practitioners in halls of industries interested in scheduling theory and its applications. I believe this book can provide a clear picture of the current research status in the area of video surveillance and can also encourage the development of new achievements in this field.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821