J. Cent. South Univ. (2014) 21: 1643−1653
DOI: 10.1007/s11771-014-2106-1

Springer

A robust system for real-time pedestrian detection and tracking

LI Qi(李琦), SHAO Chun-fu(邵春福), ZHAO Yi(赵熠)

Key Laboratory for Urban Transportation Complex Systems Theory and Technology of Ministry of Education (Beijing Jiaotong University), Beijing 100000, China

Abstract: A real-time pedestrian detection and tracking system using a single video camera was developed to monitor pedestrians. This system contained six modules: video flow capture, pre-processing, movement detection, shadow removal, tracking, and object classification. The Gaussian mixture model was utilized to extract the moving object from an image sequence segmented by the mean-shift technique in the pre-processing module. Shadow removal was used to alleviate the negative impact of the shadow to the detected objects. A model-free method was adopted to identify pedestrians. The maximum and minimum integration methods were developed to integrate multiple cues into the mean-shift algorithm and the initial tracking iteration with the competent integrated probability distribution map for object tracking. A simple but effective algorithm was proposed to handle full occlusion cases. The system was tested using real traffic videos from different sites. The results of the test confirm that the system is reliable and has an overall accuracy of over 85%.

Key words: image processing technique; pedestrian detection; tracking; video camera

1 Introduction

Pedestrians are main participants in the transportation system. More and more traffic engineers have realized the importance of pedestrians and have tried to improve the conditions of pedestrian travel. HUGHES et al [1] addressed these issues. However, deficiencies in data on pedestrians hamper research endeavors. Researchers have spent efforts on expanding current ITS application into pedestrian traffic to collect qualified data, but results are unsatisfactory thus far [2]. Commonly used traffic sensors, such as loop, sonar, and microwave, cannot monitor pedestrians automatically because pedestrians generally have poorer textures and are not restricted by lane discipline as vehicles. Another traffic sensor, the video camera, has been increasingly deployed due to its benefits of competitive cost, easy installation, easy operation and maintenance, and capacity to monitor wider areas. However, the use of video cameras as traffic sensors has been focused on vehicles and is rarely applied to pedestrian transportation modes. Thus, current algorithms could not be directly used to monitor pedestrians. Thus, a pedestrian detection and tracking (PDT) system based on video image techniques is aimed to develop, which integrates a simple but effective algorithm.

The potential of video image processing techniques for traffic data collection has been explored since the 1970s. The current commercial system available for monitoring traffic can be divided into two categories based on the measure method. The first category works with an operation principle that is similar to that of loop detectors. This category detects the existence of vehicles but does not track them. Thus, the information acquired by these systems is limited. A few systems under this category include AUTOSCOPE, CCATS, TAS, IMPACTS and TraffiCam [3−4]. To obtain more comprehensive information on traffic movements, the second category tracks vehicles, identifies individual vehicles, and follows their movements. The typical systems are CMS Mobilizer, Eliop EVA, PEEK VideoTrak, Nestor TrafficVision, Autocolor, and Sumitomo IDET. These systems allow a measure of density of the road during occupancy at a single point through yielded vehicle trajectories over a length of roadway. However, their performances are challenged by situations where vehicles occlude each other [5−6].

Although various commercial traffic data collection systems have been introduced in the market, most of these systems concentrate on motor monitoring. As attention for pedestrian traffic continues to increase,

Foundation item: Project(50778015) supported by the National Natural Science Foundation of China; Project(2012CB725403) supported by the Major State Basic Research Development Program of China
Received date: 2012−07−27; **Accepted date:** 2013−11−28
Corresponding author: SHAO Chun-fu, Professor, PhD; Tel: +86−10−51683665; E-mail: cfshao@bjtu.edu.cn

researchers have begun to focus on pedestrian detection [7−9]. The approaches used to monitor pedestrians normally involve three steps, object detection, tracking, and classification.

Object detection is the basic step. At present, several major approaches are available, including frame difference, optical flow, statistical learning, and background subtraction [10−11]. Although frame difference has low computation cost, it is sensitive to noise and not robust in a dynamic background. Optical flow can handle non-static background, but its computation is expensive. Statistical learning can detect objects either from a single image or from video sequences, but it normally requires artificial initialization and is computationally expensive. Background subtraction is the most popular approach due to its characteristics of fast computation and accuracy [12]. It can be classified into two broad categories: non-regression recursive and regression recursive. The first category uses and stores a fixed number of frames in a buffer as the recent observation data to conduct background modeling, whereas the second category does not need to maintain a buffer of history frames. Instead, it recursively updates the background model with the incoming frame. Thus, it requires less storage compared with previous techniques. Recursive techniques have been studied extensively [13−15]. The Gaussian mixture model (GMM) is a representative of these techniques [16]. The reason for its success is essentially due to its capacity to obtain an adaptive background and robustness even for the lights that change or the trees that wave in the environment.

Many algorithms have been attempted for tracking, and these can be classified into two types. The first type is the probabilistic method, which views tracking as a dynamic state estimation problem under the Bayesian framework [18]. Representative methods include the Kalman filter [19−20], condensation [21], and particle filtering [22−25], among others. The deterministic method is the second type, and it involves a comparison of a model with a current frame and then determining the most promising matching region. The mean-shift (MS) technique is a typical example. Deterministic methods suffer from a severe occlusion problem because their tracking is based on previous tracking results. However, they are usually more accurate and quicker than the probabilistic multi-hypothesis tracking methods [26].

Object classification is used to identify pedestrians from other detected objects. Under this method, model-based techniques are used commonly. However, discerning and complete libraries are necessary, and efficient scaling algorithms must be present in operating these techniques [27−28]. In addition, the manual establishment and maintenance of the library is also a problem. Model-free methods are much more appealing because of their easy maintenance and flexibility. They classify the objects by several criteria, which were related to length, width, area, and perimeter of the objects [29]. For fixed cameras, pedestrians are mostly different in shape parameters from vehicles. Thus, model-free methods are effective and fit most scenes [9].

At present, despite efforts to study traffic data collection, no single algorithm or system for pedestrian monitoring has gained wide acceptance. The various methods mentioned above inspired this work. In this work, a system is developed to detect and track pedestrians from a low-resolution video camera. The MS segmentation technique, combined with GMM, is used to enhance the foreground detection reliability in noise and illumination changes. Shadow removal is utilized to alleviate the negative impact of the shadow on the detection mechanism. A multi-cue integrated tracking algorithm is present to track objects and then obtain their trajectories. A model-free method is used to identify pedestrians. With the occlusion handler, not only can the severe occlusion problem with two pedestrians be handled, but also the full occluding situation when more pedestrians participate. The system was implemented by using Microsoft Visual C# and tested through live video images from several locations. As shown in the experimental results, the proposed system is robust for pedestrian monitoring and can effectively handle total occlusion problems.

2 New detection and tracking system

2.1 System overview

The proposed system contains five steps: image preprocessing, moving object detection, shadow removal, object tracking, and object classification. First, the MS segmentation algorithm is utilized to preprocess the image and enhance the image quality in the video sequence. Then, foreground objects are extracted from the background by the GMM. Shadow detection and removal are applied to each detected object to refine the detection results. In the challenging tracking module, trajectories are yielded from the multi-cue integrated MS method, and the total occlusion problem is addressed by an effective algorithm that serves as an occlusion handler. Next, a model-free method is used to classify pedestrians from other objects. Finally, identified pedestrians are assigned ID tags. Figure 1 shows the flowchart of the proposed system.

2.2 Pre-processing

Image segment techniques can divide an image into nearly homogeneous regions. Using this method to process images can smooth image noise and diminish

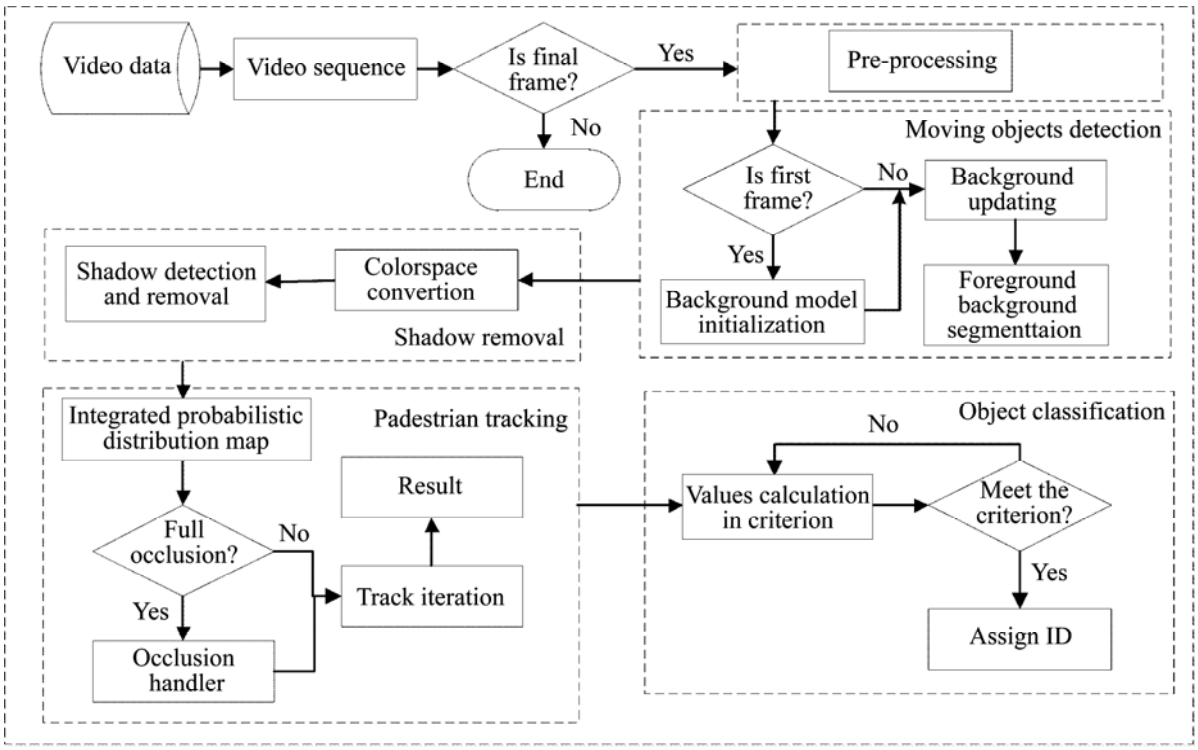

Fig. 1 Flowchart of entire system

scattered pixels. Hence, this method can help obtain integrated objects that are detected in the detection step, and it can enhance the quality of the following video analysis. In this work, MS segmentation is employed for image preprocessing because of its efficiency and flexibility [30].

In MS segmentation, the first step is discontinuity preserving filtering that aims to identify dimensional convergence points through the MS procedure, which can be expressed as

$$\boldsymbol{M}_h(x)=\frac{1}{k}\sum_{x_i}(x_i-x) \tag{1}$$

where $\{x_i\}_{i=1,2,\cdots,n}$ is the n sample points in dimensional space, $\boldsymbol{S}_h(x)$ is a window with the radius h, and $\boldsymbol{M}_h(x)$ is the MS vector that has the direction of the gradient of the density estimate at x. $\boldsymbol{M}_h(x)$ always points toward the increasing direction of the maximum density. A translation of the window $\boldsymbol{S}_h(x)$ by $\boldsymbol{M}_h(x)$ guarantees the convergence of the MS procedure. When in convergence, the center of the window $\boldsymbol{S}_h(x)$ is the convergence point.

The next step is MS clustering, whereby the clusters of convergence points are identified. All points closer than 0.5 unit from each other in the joint domain are linked together. Then, the image is segmented with connecting points labeled by the clusters. Figure 2 shows the resulting image of the pre-processing. Figure 2(b) has less noise than the former image (Fig. 2(a)), especially for the ground part in Fig. 2(b), due to MS segmentation technique.

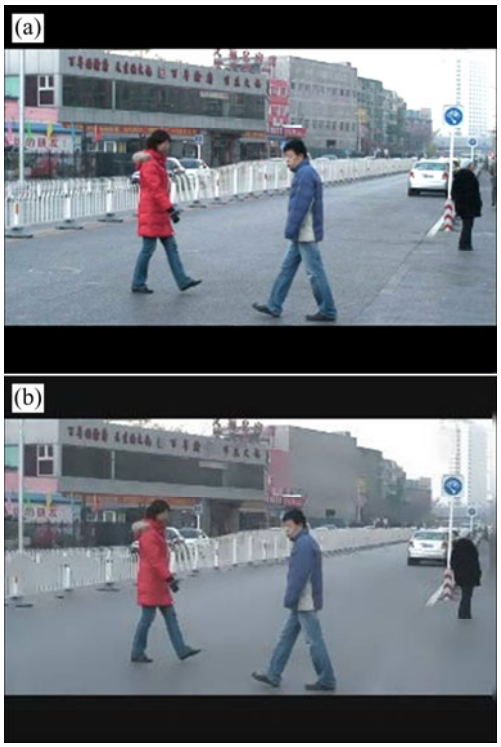

Fig. 2 Pre-processing result through MS segmentation: (a) Original frame; (b) Frame after pre-processing

2.3 Object detection

In the PDT system, object detection segments image

pixels that belong to the foreground from the background. The essential work of object detection is to obtain an adaptive background from image sequences. In this work, the GMM is employed for background extraction. In the model, each input pixel is estimated by the mixture of k Gaussian distributions. The probability of observing an input pixel X at frame t is as follows:

$$P(\boldsymbol{X}_t)=\sum_{i=1}^{K} w_{i,t}\times\eta(\boldsymbol{X}_t,\boldsymbol{\mu}_{i,t},\boldsymbol{\Sigma}_{i,t}) \tag{2}$$

where $w_{i,t}$ is the weight of the i-th Gaussian distribution at time t; $\boldsymbol{\mu}_{i,t}$ and $\boldsymbol{\Sigma}_{i,t}$ are the mean value and covariance matrix of the i-th Gaussian distribution in the mixture at time t, respectively; $\eta(X_t,\boldsymbol{\mu}_{i,t},\boldsymbol{\Sigma}_{i,t})$ is the i-th Gaussian probability density function.

$$\eta(\boldsymbol{X}_t,\boldsymbol{\mu}_{i,t},\boldsymbol{\Sigma}_{i,t})=\frac{1}{(2\pi)^{\frac{n}{2}}\,|\,\boldsymbol{\Sigma}_{i,t}\,|^{\frac{1}{2}}}\mathrm{e}^{-\frac{1}{2}(\boldsymbol{X}_t-\boldsymbol{\mu}_{i,t})^{\mathrm{T}}\boldsymbol{\Sigma}_{i,t}^{-1}(\boldsymbol{X}_t-\boldsymbol{\mu}_{i,t})} \tag{3}$$

Then, the background distributions B are chosen as the background model,

$$B=\mathrm{argmin}_b\left(\sum_{k=1}^{b} w_k>T\right) \tag{4}$$

where T is a measure of the minimum portion of the recent data that should be accounted for in obtaining the "best" distributions. In GMM, each distribution represents a different color.

An update scheme is adopted to adapt to the changes of illumination and moving objects. For each input pixel, if its color matches a Gaussian distribution in B, the corresponding distribution is updated:

$$\begin{cases} w_{k,t}=(1-\alpha)w_{k,t}+\alpha(M_{k,t}) \\ \boldsymbol{\mu}_{k,t}=(1-\rho)\boldsymbol{\mu}_{k,t}+\rho\boldsymbol{X}_t \\ \boldsymbol{\Sigma}_{k,t}=(1-\rho)\boldsymbol{\Sigma}_{k,t-1}+\rho(\boldsymbol{X}_t-\boldsymbol{\mu}_{k,t})^{\mathrm{T}}(\boldsymbol{X}_t-\boldsymbol{\mu}_{k,t}) \\ \rho=\alpha\eta(\boldsymbol{X}_t,\boldsymbol{\mu}_{k,t},\boldsymbol{\Sigma}_{k,t}) \end{cases} \tag{5}$$

where α is the learning rate, $M_{i,t}$ is 1 for the matched model and 0 for the remaining models, and ρ is the learning rate for parameters. If none of the Gaussian distributions matches the input pixel value, the least probable distribution is replaced by a fresh distribution. Additionally, the new distribution will be initialized with the current pixel value as the mean value with low weight and a high variance.

2.4 Shadow removal

Shadow removal significantly affects both object tracking and object classification. The detected object area and shape can be falsified by shadows that can lead to track failure. The merging or overlapping of shadows can also cause misclassification. Thus, shadow removal is required in the system.

The shadow removal technique mentioned in Ref. [31] is employed in this work. Hue-Saturation-Value (HSV) color space is used. In HSV, shadow points can be detected because their hue and luminance (the V component in HSV) are slightly changed, and the saturation is generally lower than that of the background points. Hence, the criterion to define a shadow point is as follows:

$$P_{\mathrm{S}}(x,y)=\begin{cases} 1,\ \text{if } \alpha\le\dfrac{I^{\mathrm{V}}(x,y)}{B^{\mathrm{V}}(x,y)}\le\beta \\ \qquad\wedge(I^{\mathrm{S}}(x,y)-B^{\mathrm{S}}(x,y))\le\tau_{\mathrm{S}} \\ \qquad\wedge\left|I^{\mathrm{H}}(x,y)-B^{\mathrm{H}}(x,y))\right|\le\tau_{\mathrm{H}} \\ 0,\ \text{otherwise} \end{cases} \tag{6}$$

where $I(x, y)$ and $B(x, y)$ are the pixel values at coordinate (x, y) in the original and background images, respectively. α, β, τ_{S} and τ_{H} are the thresholds. $P_{\mathrm{S}}(x, y)=1$ indicates that the pixel at coordinate (x, y) belongs to the shadow. The shadow can be defined according to the criterion, and the foreground image without shadow can be obtained.

2.5 Tracking

Once the object areas are determined in each frame, the tracking algorithm traces the objects from each frame by using the MS algorithm [31]. The motion-cue and color-cue are fused to ensure tracking performance. The occlusion handler is also used to address the total occlusion condition.

2.5.1 Multi-cue fusion

MS tracking employs only color information, which is computed easily, but this approach causes the tracking results to be vulnerable to distraction from some background areas with similar colors. Furthermore, tracked objects with low saturation would be lost easily because of heavy noise. A multi-cue integration method is developed to enhance MS tracking robustness against these problems. The method, which combines motion cue with color cue, can compute an integrated probabilistic distribution map (PDM).

In MS tracking iterations, the fused PDM that indicates the object being tracked should be calculated first.

The PDM M_1, based on color-cue, is calculated via histogram back projection using two steps. First, the color histogram of the object is calculated and kept in a look-up table. Second, when a new frame is added as input, the table is checked for the color of each pixel, and

a probability value is given to each pixel. Thus, a color PDM M_1 is obtained.

The motion PDM M_2 can be obtained as follows. First, the difference image is computed by using background subtraction between the background image extracted by GMM beforehand and the incoming frame. Then, using the threshold approach, the difference image is converted into a binary image. The binarized image can be viewed as the motion PDM M_2. The likelihood of motion pixel is 1, whereas the probability of the background pixel is 0.

A fusion method, called maximum and minimum integration (MMI), is developed to integrate multi-cues. In this method, the probability of each pixel is obtained by integrating the color PDM and motion PDM. The combined likelihood of a certain pixel at location x_i can be calculated as

$$p(x_i,t)=\max\begin{Bmatrix}C^*\times\max\{p_{\text{color}}(x_i\mid M_1,t),p_{\text{motion}}(x_i\mid M_2,t)\}\\C^*\times\min\{p_{\text{color}}(x_i\mid M_1,t),p_{\text{motion}}(x_i\mid M_2,t)\}\end{Bmatrix} \quad (7)$$

where $p(x_i, t)$, $p_{\text{color}}(x_i|M_1,t)$, and $p_{\text{motion}}(x_i|M_2,t)$ are the observation likelihoods at pixel position x_i in the combined PDM, PDM M_1 and PDM M_2, respectively. C^* is the binary parameter, which can be determined by

$$C^*=\begin{cases}1, & x_i\in A\\0, & \text{otherelse}\end{cases} \quad (8)$$

where A is the area enclosed by the outline of the tracking object. If the pixel at x_i belongs to region A, C^* will be assigned as 1. Otherwise, C^* will be 0. The MMI method is open, and more cues can be integrated into it. Through the method, the scattered noise on the PDM of the single cue can be eliminated. In addition, the image region of the tracking object in each cue can be enhanced. Thus, MMI-based MS tracking is robust even with scattered noise and distraction coming from similar background colors. Figure 3 shows the stepwise approach for generating the combined PDM.

After obtaining the combined PDM, the object can be located by seeking its peak value. **Algorithm 1** illustrates the MMI based MS tracking algorithm.

Algorithm 1. MS algorithm based on multi-cue integration.

Calculate color PDM $p_{\text{color}}(x_i,t)$ through histogram back project.

Calculate motion PDM $p_{\text{motion}}(x_i,t)$ through background subtraction and image binarization.

Integrate two maps using MMI method with formulas (8−10).

Initialize MS tracking iteration: Assign initial values to location l_0 and scale s_0 of tracking window, respectively, on combined PDM.

MS tracking iteration: Compute moments $M_{00}M_{01}$, new tracking window location $\hat{l}$, and the scale $\hat{s}$ until algorithm reaches convergence.

$$M_{00}=\sum_i p(x_i,t)\quad M_{01}=\sum_i x_i p(x_i,t)$$

$$L=\frac{M_{01}}{M_{00}}\quad S=k\sqrt{M_{00}}$$

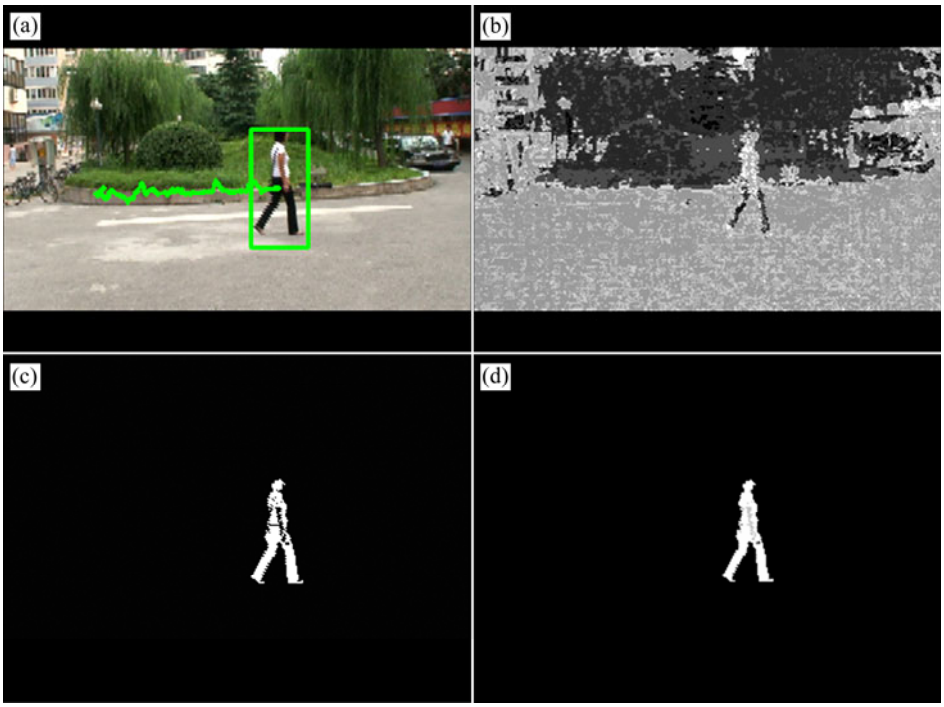

Fig. 3 Stepwise illustration of multi-cue integration: (a) Tracking result based on MMI probabilistic map; (b) Color probabilistic map; (c) Motion probabilistic map; (d) MMI probabilistic map

2.5.2 Occlusion handler

A significant problem arises in the case of total occlusion, whereby one tracking object is completely hidden by another object. As total occlusion is a major challenge for tracking performance, addressing this problem is essential for succeeding detection systems. Based on the MS algorithm, an occlusion handling approach is proposed to detect full occlusion cases to elucidate the occlusion (to understand whether an object is occluded and then to identify the occluder) and to reinitialize the tracking automatically when the hidden object reappears.

When the occlusion involves two objects (Fig. 4(a)), the objects compete for the points of the same image region. The image points that are compatible with the appearance of the occluder (P1) will be assigned to it, and no significant changes in its area will be observed. However, the area of the occluded object (P2) will appear to shrink gradually and obviously as fewer image points will be assigned to it (Figs. 4(b) and (c) in PDM)) Thus, occlusion ratio $R_{C,i}$ of the object O_i is defined as

$$R_{C,i}=\frac{A_i}{\dot{A}_i} \tag{9}$$

where A_i is the current area of the object O_i, and $\dot{A}_i$ is the area of the object O_i which remains isolated in the last frame. $R_{C,i}$ is used to measure the degree of the object O_i that shares the points of a region with other objects; it can indicate the degree of the occluded significance of the object O_i.

However, using only $R_{C,i}$ to characterize an occlusion quantitatively is inadequate because some inevitable noises may affect the area size of the object O_i in a certain frame. Therefore, the average digression degree $D_{A,i}$ is used, which can indicate the progressively decreasing trend of object area:

$$D_{A,i}=\frac{\sum_{k=0}^{n} M_{i,t-k}}{n} \tag{10}$$

where $M_{i,t-k}$ is the binary parameter that is used to decide whether the area of the object O_i at frame $t-k$ is reduced compared to the area in the last frame. $M_{i,t-k}$ can be calculated by Eq. (11), and n is the number of frames for calculation.

$$M_{i,t-k}=\begin{cases}1, & \dfrac{A_{i,t-k}}{A_{i,t-1-k}}\le T_i\\ 0, & \text{otherwise}\end{cases} \tag{11}$$

where $A_{i,t-k}$ and $A_{i,t-1-k}$ are the areas of the object O_i at frames $t-k$ and $t-1-k$, respectively. T_i is a presetting threshold, which is less than 1. The average digression degree $D_{A,i}$ is measured for the decreasing trend of the area size of the object O_i in the continuous frames.

The object O_i will be reported as disappeared due to a complete occlusion when the following condition is met:

$$R_{C,i}<T_{CRi}\wedge T_{ADi}<D_{A,i}\le 1$$

In a full occlusion case, the declared occluded object is assumed to be behind its occluder and moves with it until the occluded object reappears ($R_{C,i}>T'_{CRi}$ and $D_{A,i}<T'_{ADi}$). When a previously hidden object does reappear in the vicinity of its occluder, some other large regions will be found in the PDM. The position of the largest one can be used to reinitialize the location of the tracking window.

In actual practice, occlusions with more than two objects do occur. Thus, the case of full occlusions that involve must be considered in the PDT system. By introducing a relational matrix (RA), the proposed method can form and then record the relationships between occluded objects and their occluders.

For a group of objects in an overlapped occlusions relation, there will always be a foremost occluder and some blocked ones behind it. For the hidden objects, all the other objects will be reported as potential occluders.

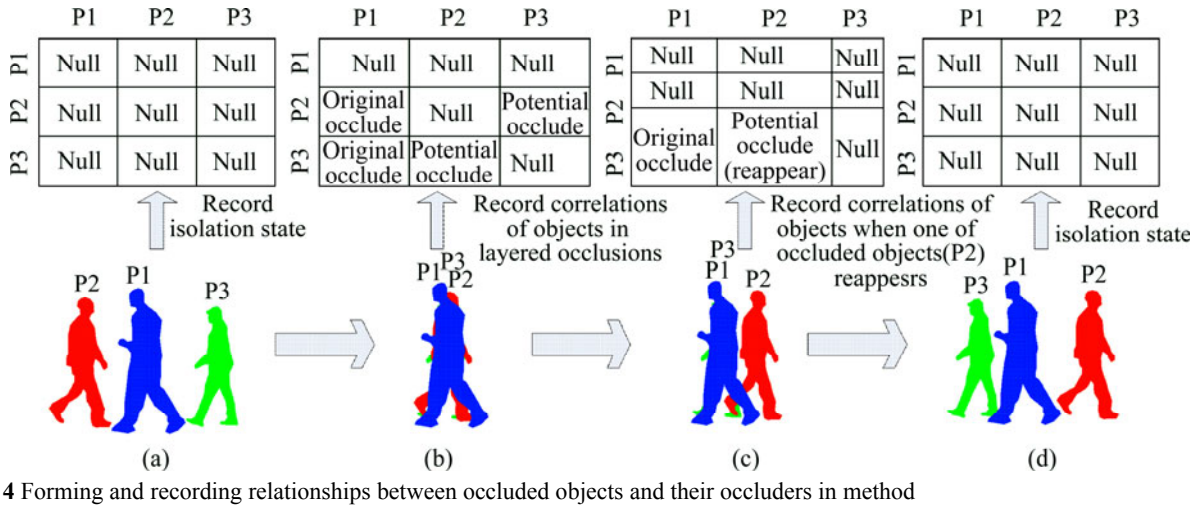

Fig. 4 Forming and recording relationships between occluded objects and their occluders in method

These relations can be maintained in an RA as shown in Fig. 4(b). The records of row 2, column 1 and row 3, column 1 in RA indicate that object P1 moves in front of P2 and P3 and occludes them, whereas the records in row 2, column 3 and in row 3, column 2 indicate that P3 and P2 potentially occlude each other.

As for the reappearance of these blocked objects, the labels of the occluders (including both potential and original occluders) should be removed from the RA correspondingly. The search for the remaining occluded objects should be done for both the nearby original occluder and reappeared object. As shown in Fig. 4(c), in RA, the contents in row 2 (row P2) were cleared, and the approximation of the objects in row 3 (row P3) is searched from objects related to the records "Original Occlude" or "Potential Occlude (Reappear)" for P3.

Figure 4 shows an example of full occlusions with three participating objects. The method can handle more complex cases of occlusions, which involve more objects.

2.6 Classification

Object classification is an essential procedure in identifying correct object types. In this system, objects are discriminated based on their inherent properties. These properties include

1) Length-width ratio (R_{wh}) of the bounding box (W/H)

2) Area of the region (A)

3) Ratio between the area and perimeter (P_r) square of the region. The ratio depicts the degree of object compactness (C).

$$C = \frac{A}{P_r^2} \tag{12}$$

An object O_i will be recognized as a pedestrian if and only if the conditions below are met:

$$\begin{cases} A_i > A_{T,\min} \wedge A_i < A_{T,\max} \\ R_{whi} > R_{whT,\min} \wedge R_{whi} < R_{whT,\max} \\ C_i > C_{T,\min} \wedge C_i < C_{T,\max} \end{cases} \tag{13}$$

where $A_{T,min}$, $A_{T,max}$, $R_{whT,min}$, $R_{whT,max}$, $C_{T,min}$, and $C_{T,max}$ are the thresholds of the minimum and maximum values of the length-width ratio, area, and object compactness, respectively.

For a given pedestrian, its ID will remain unchanged in the video frames, including when the tagged pedestrian is occluded.

3 Application and performance test

The PDT system is implemented by C# and runs on a 1.83 GHz Intel(R) Core(TM) 2 CPU computer. Actual videos were tested to verify its performance. The videos for the experiment were captured in Beijing, China using a SONY Handycam video camera, and the size of each frame was 320×240. Considering the potential uses of this system in entrances and exits, walkways, sidewalks and crosswalks, three test sites were selected: entrance to a residential building, walkway in a public facility, and a sidewalk and crosswalk at an intersection. Test video 1 was taken at Jiaotongdaxue Road. Test video 2 was taken at a campus road at Beijing Jiaotong University. Test video 3 was captured at the intersection of Xueyuannan Road and Jiaodadong Road. The snapshots for each of the three test sites are shown in Fig. 5.

One challenging aspect of monitoring systems for

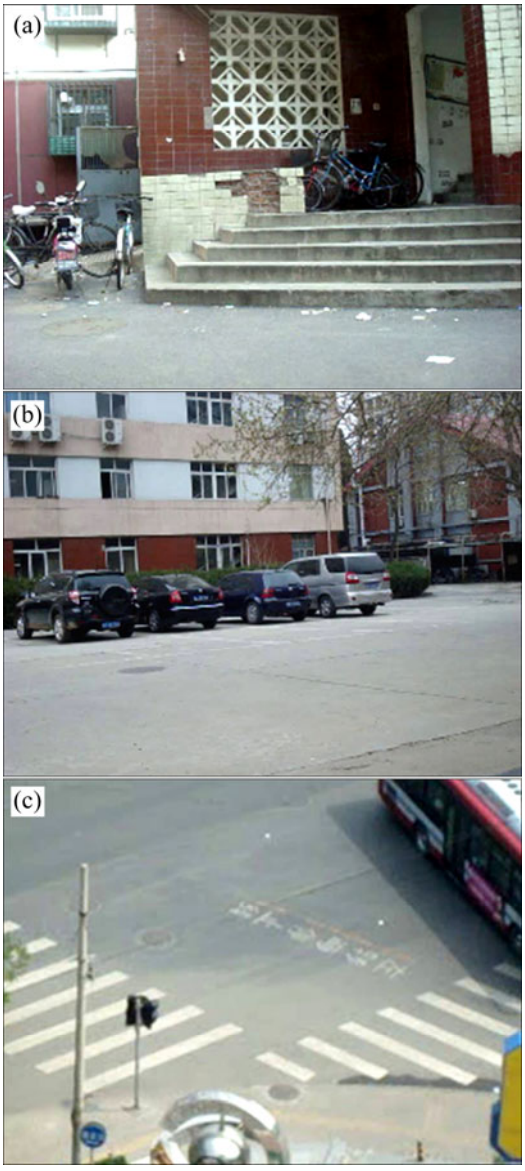

Fig. 5 Snapshots of test sites: (a) Test site one (Jiaodadong Road); (b) Test site two (a campus road of Beijing Jiaotong University); (c) Test site three (intersection of Xueyuannan Road and Jiaodadong Road)

real environments is the occurrence of occlusion. Figure 6 shows an example of occlusion with two pedestrians and the corresponding curve chart of an occlusion ratio $R_{C,i}$ and average depression degree $D_{A,i}$ during occlusion at test site 1. In Fig. 6, the pedestrians were labeled with numbers 0 and 1, respectively. Pedestrian 0(P0) walked in a direction opposite to pedestrian 1(P1). When they encountered each other, an occlusion occurred, and both of the object models of P0 and P1 were not updated. As soon as occlusion started, the area of P1 decreased gradually, until P1 disappeared because of the full cover by P0, as shown in Fig. 6(b). Later, P1 reappeared and moved away from P0. Finally, P1 and P0 moved apart, as shown in Fig. 6(c), and their object models were reinitialized and then updated. Figure 6(d) shows that the curve variations for occluded object P1 and occluder P0 were obviously different during occlusion. During the reappearance of P1, occlusion ratio $R_{C,i}$ increased, whereas the average digression degree $D_{A,i}$ significantly reduced. T_{CRi} and T_{ADi} were equal to 0.6 and 0.3, respectively. Thus, the PDT was effective for detecting the occlusion, reasoning the occlusion, and determining the reappearance of the hidden object.

Figure 7 shows a more complex example of occlusion with more pedestrians than the previous example. In the 1160th frame, from right to left of the image, pedestrian objects were tagged as P11, P10, P13, P12, respectively. The colors of their search windows were cyan, red, yellow, and green. In frame 1218, pedestrian targets P10 and P13 were totally occluded by P11. From frame 1218, P10 and P13 remained behind P11, moved with it, and were searched around it all the way until frame 1224. In frame 1224, as soon as P13 reappeared, the search for P13 stopped, whereas that for P10 near P13 and P11 began. In frame 1233, target P10 matched the image region surrounding P13 and reappeared. The occlusion handler is shown to work successfully, and the PDT system is proven to be robust to total occlusion with multiple pedestrians.

Table 1 shows the results of system evaluation for the tests at these three sites, including manually observed results (ground-truth data), system operation results, and comparisons between the two sets of results. In general, the PDT system performed well. The detection rate was over 85% for different test sites involving 180 pedestrians in total. The accuracy was slightly varied at the test sites. Test sites one and two were set in an ideal flow situation, where the average pedestrian rate was only 110 pedestrian/h. The appearances of people caught in videos in these sites were clear, and their mean size was about 550 pixel×pixel. Hence, detection errors in both sites were less than 8%. Test site three was set at an intersection where the average pedestrian flow rate was 500 pedestrian/h. The relatively heavy volume of traffic caused more occlusion to occur at this site and was the major reason for tracking failure. The test results show

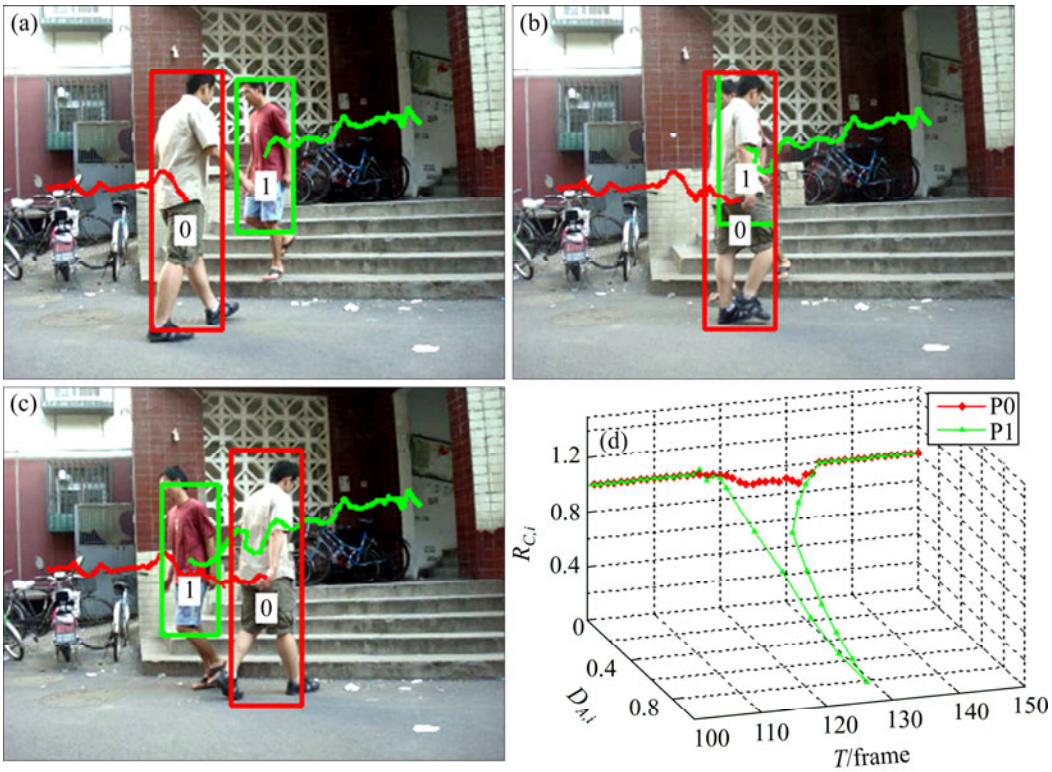

Fig. 6 Tracking multiple pedestrians in an occlusion: (a) Two pedestrians P0 and P1; (b) P1 covered by P0; (c) P1 and P0 moving apart; (d) Curves for $D_{A,i}$ and $R_{C,i}$

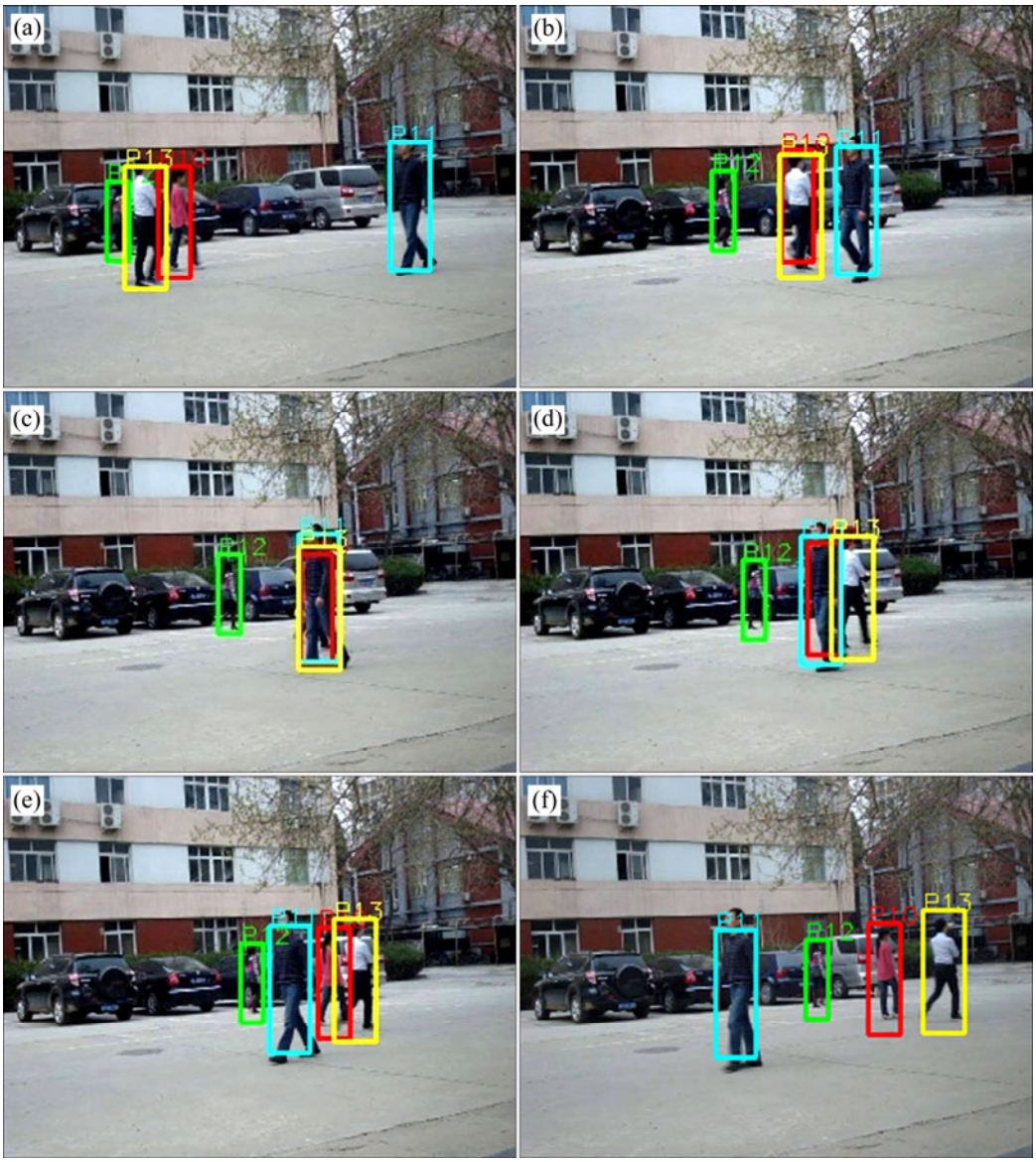

Fig. 7 Total occulsion with multi-objects: (a) Frame 1160; (b) Frame 1206; (c) Frame 1218; (d) Frame 1224; (e) Frame 1233; (f) Frame 1264

Table 1 Summary of test results

Test site	Ground-truth	System detected	Comparison error/%
1	28	26	7.14
2	18	19	5.56
3	134	116	13.43
Total	180	161	10.55

that the pedestrian count error was 13.43%. The system could not perform as well in this site as in the other two sites. However, considering that the test condition is more complicated and the volume is larger, the accuracy level achieved in this test is still acceptable.

In comparison with the ground-truth data, a major issue of the PDT system is failure in case of occlusion. Figure 8 illustrates one failure case. In Fig. 8(a), two pedestrians were recognized as one pedestrian and surrounded with a yellow search window because these two pedestrians entered the scene together. In the successive frames, they were not separate in Figs. 8(b) and (c), and left the scene in Fig. 8(d), and thus, one count was missing in the tracking process.

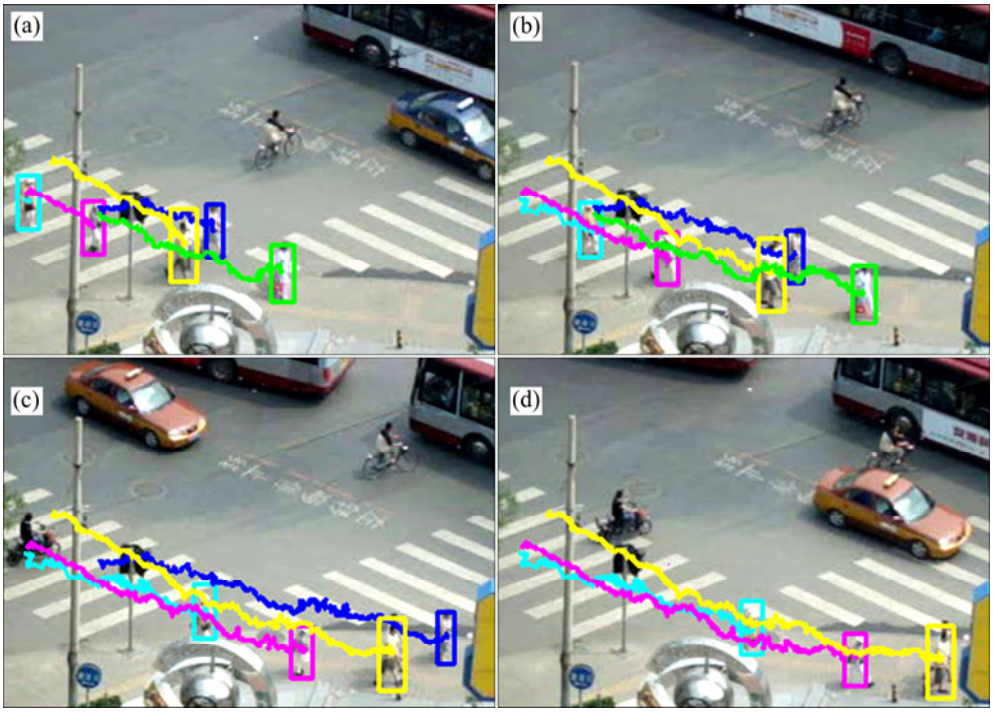

Fig. 8 Example of tracking failure: (a) Frame 2143; (b) Frame 2198; (c) Frame 2285; (d) Frame 2324

4 Conclusions

1) A robust system for real-time pedestrian detection and tracking from a low-resolution video camera to collect pedestrian data automatically and maximize the utility of existing video infrastructure is proposed.

2) This system integrates several modules into image preprocessing, motion detection, tracking, and classification, as well as proposes an effective algorithm to handle complete occlusion. With the occlusion handler, not only can the severe occlusion problem with two pedestrians be handled, but also the full occluding situation when more pedestrians participate. The system is tested using actual videos. The results obtained are encouraging, with the accuracy at approximately 90% for pedestrians for all three test sites. The loss in accuracy is due to two objects sticking together. Objects entering and leaving the scene together result in missed counting.

3) Some problems need to be solved although the results show that this system is effective and feasible in the tested traffic conditions. The PDT system must be improved to make it more effective for practical application. In future studies, more robust algorithms should be explored to deal with the counting of objects that stick together.

References

[1] HUGHES R, HUANG H, ZEGGEER C, CYNECKI M. Automated pedestrian detection used in conjunction with standard pedestrian push buttons at signalized intersections [J]. Transportation Research Record, 2000, 1705: 32−39.

[2] COTTRELL W D, PAL D. Evaluation of pedestrian data needs and collection efforts [J]. Transportation Research Record, 2003, 1828: 12−19.

[3] COIFMAN B, BEYMER D, MCLAUCHLAN P, MALIK J. A real-time computer vision system for vehicle tracking and traffic surveillance [J]. Transportation Research: Part C, 1998, 6(4): 71−288.

[4] Image sensing systems inc. Applications of autoscope [EB/OL]. [2011−07−06]. http://autoscope.com/applications/.

[5] ZHANG X, FORSHAW M R B. A parallel algorithm to extract information about the motion of road traffic using image analysis [J]. Transportation Research: Part C, 1997, 5(2): 141−152.

[6] LIU X, XU X, DAI B. Vision-based long-distance lane perception and front vehicle location for full autonomous vehicles on highway roads [J]. Journal of Central South University, 2012, 19(5): 1454−1465.

[7] KILAMBI P, RIBNICK E, JOSHI A J, MASOUD O, PAPANIKOLOPOULOS N. Estimating pedestrian counts in groups [J]. Computer Vision and Image Understanding, 2008, 110(1): 43−59.

[8] LI J, SHAO C F, XU W T, LI J. A real-time system for tracking and classification of pedestrian and bicycles [J]. Transportation Research Record, 2010, 2198: 83−92.

[9] MALINOVSKIY Y, ZHENG J Y, WANG Y H. Simple and model-free algorithm for real-time pedestrian detection and tracking [C]// The 86th Annual Meeting of the Transportation Research Board. Washington, D.C.: TRB, 2007.

[10] Li Q, SHAO C F, YUE H. Real-time foreground-background segmentation based on improved codebook model [C]// Image and Signal Processing (CISP). Yantai, China: IEEE Press, 2010, 1: 269−273.

[11] CHRISTODOULOU L, KASPARIS T, MARQUES O. Advanced statistical and adaptive threshold techniques for moving object

detection and segmentation [C]// Digital Signal Processing (DSP). Corfu: IEEE Press, 2011: 1−6.

[12] CHEN Y, CHEN C, HUANG C, HUNG Y. Efficient hierarchical method for background subtraction [J]. Pattern Recognition, 2007(40): 2706−2715.

[13] PARUCHURI J K, SATHIYAMOORTHY E P, CHEUNG S S, CHEN C H. Spatially adaptive illumination modeling for background subtraction [C]// IEEE Computer Vision workshops. Kentucky, USA: IEEE Press, 2011: 1745−1752.

[14] ZAHARESCU A, JAMIESON M. Multi-scale multi-feature codebook-based background subtraction [C]// IEEE Computer Vision Workshops, ON, Canada: IEEE Press, 2011: 1753−1760

[15] KIM K, THANARAT H C, HARWOOD D, DAVIS L. Real-time foreground-background segmentation using codebook model [J]. Real-Time Imaging, 2005, 11(3): 172−185.

[16] STAUFFER C, GRIMSON W E L. Adaptive background mixture models for real-time tracking [C]// Computer Vision and Pattern Recognition. Fort Collins, Colorado: IEEE Press, 1999: 246−252.

[17] LI Z D, WANG W H, WANG Y, CHENG F, WANG Y. Visual tracking by proto-objects [J]. Pattern Recognition, 2013, 46(8): 2187−2201.

[18] PAPADOURAKIS V, ARGYROS A. Multiple objects tracking in the presence of long-term occlusions [J]. Computer Vision and Image Understanding, 2010, 114(7): 835−846.

[19] FU Z X, HAN Y. Centroid weighted Kalman filter for visual object tracking [J]. Measurement, 2012, 45(4): 650−655.

[20] MEUTER M, IURGEL U, PARK S B, KUMMERT A. The unscented Kalman filter for pedestrian tracking from a moving host [C]// IEEE Intelligent Vehicles Symposium. Eindhoven, Holland: IEEE Press, 2008: 37−42.

[21] RATSCH M, BLUMER C,VETTER T, TESCHKE G. Efficient object tracking by condentional and cascaded image sensing [J]. Computer Standards and Interfaces, 2012, 34(6): 549−557.

[22] HU Z T, PAN Q, YANG F, CHENG Y M. An improved particle filtering algorithm based on observation inversion optimal sampling [J]. Journal of Central South University of Technology, 2009, 16(5): 815−820.

[23] HOTTA K. Adaptive weighting of local classifiers by particle filters for robust tracking [J]. Pattern Recognition, 2009, 42 (5): 619−628.

[24] YAO A B, LIN X G, WANG G J, YU S. A compact association of particle filtering and kernel based object tracking [J]. Pattern Recognition, 2012, 45(7): 2584−2597.

[25] WANG Z W, YANG X K, YI X, YU S Y. CamShift guided particle filter for visual tracking [J]. Pattern Recognition Letters, 2009, 30(4): 407−413.

[26] LIU H, YU Z, ZHA H B, ZOU Y X, ZHANG L. Robust human tracking based on multi-cue integration and mean-shift [J]. Pattern Recognition Letters, 2009, 30(9): 827−837.

[27] COLLINS R T. Mean-shift blob tracking through scale space [C]// IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Pittsburgh, USA: IEEE Press, 2003, 2: 234−240.

[28] ZHANG Z X, HUANG K Q, WANG Y H, LI M. View independent object classification by exploring scene consistency information for traffic scene surveillance [J]. Neurocomputing, 2013, 99(1): 250−260.

[29] OWENS J, HUNTER A, FLETCHER E. A fast model-free morphology-based object tracking algorithm [C]// British Machine Vision Conference. Cardiff, UK, 2002: 767−776.

[30] COMANICIU D, MEER P. Mean-shift a robust approach toward feature space analysis [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24: 603−619.

[31] CUCCHIARA R, GRANA C, PICCARDI M, PRATI A, SIROTTI S. Improving shadow suppression in moving object detection with HSV color information [C]// IEEE Intelligent Transportation Systems Conference. Oakland, CA: IEEE Press, 2001: 334−339.

[32] COMANICIU D, RAMESH V, MEER P. Real-time tracking of non-rigid objects using mean shift [C]// IEEE Computer Vision Pattern Recognition. Hilton Head Island, SC: IEEE Press, 2000, 2: 142−149.

(Edited by DENG Lü-xiang)