International Journal of Automation and Computing 04(1), January 2007, 38-46
DOI: 10.1007/s11633-007-0038-z

Moving Shadow Detection and Removal for Traffic Sequences

Mei Xiao[1*] Chong-Zhao Han[1] Lei Zhang[2]

[1]School of Electronics and Information Engineering, Xi′an Jiaotong University, Xi′an 710049, PRC

[2]Taibai Campus, Chang'an University, Xi′an 710064, PRC

Abstract: Segmentation of moving objects in a video sequence is a basic task for application of computer vision. However, shadows extracted along with the objects can result in large errors in object localization and recognition. In this paper, we propose a method of moving shadow detection based on edge information, which can effectively detect the cast shadow of a moving vehicle in a traffic scene. Having confirmed shadows existing in a figure, we execute the shadow removal algorithm proposed in this paper to segment the shadow from the foreground. The shadow eliminating algorithm removes the boundary of the cast shadow and preserves object edges firstly; secondly, it reconstructs coarse object shapes based on the edge information of objects; and finally, it extracts the cast shadow by subtracting the moving object from the change detection mask and performs further processing. The proposed method has been further tested on images taken under different shadow orientations, vehicle colors and vehicle sizes, and the results have revealed that shadows can be successfully eliminated and thus good video segmentation can be obtained.

Keywords: Moving shadow detection, moving shadow removal, edge information, traffic scene.

1 Introduction

Identifying moving objects from a video scene is a fundamental and critical task in video surveillance, traffic monitoring and analysis. Background subtraction is a common approach in extracting the concerned objects accurately from the image sequence. However, shadow detection is a very different issue since shadows and objects have two important features. First, points in shadow area are detectable as foreground points because the intensity difference arising from shadow is significant. Second, shadows have the same motion as the objects casting them. Usually, shadows are extracted along with the objects in most of the moving segmentation algorithms. When the detected objects contain shadow, it may cause large errors in object localization, recognition, and tracking. Thus, the algorithm that uses segmentation result as its basic measurement may fail. For example, Shadows can in fact change the shape and colour of objects and therefore affect scene analysis and interpretation systems in many applications. On the other hand, objects together with their shadows or several objects connected through shadows can cause problems for algorithms which aim to track or recognize objects in the scene. So shadow elimination is a critical issue for systems aiming at extracting, tracking or recognizing objects in a given scene.

Some shadow elimination techniques have been proposed in recent years. We classify shadow detection techniques in the literature into two groups: model-based and property-based. Model-based techniques[1,2] are based on matching sets of geometric features such as edges, lines or corners to 3D object models, and rely on models representing the a priori knowledge of the geometry of the scene, the objects, and the illumination. Model-based techniques can be successful when shadow conforms to this hypothesis. However, they are generally only applicable to the specific application and designed for specific applications. Because of these limitations, there has been a trend towards using the properties such as the geometry, brightness, colour or edge of shadows to obtain better result.

Property-based shadow elimination algorithms are described as follows: Based on the hypothesis that shadow areas were not textured, Fung *et al.* [3] proposed a moving shadow removal algorithm which worked for applications like traffic surveillance. A combined shadow confidence score was derived based on the luminance, chrominance and gradient density proprieties, and then the confidence score value and geometry information were used to separate a cast shadow from the object. Cucchiara *et al.* [4] exploited hue, saturation and motion information to detect moving objects, shadows and ghosts in video sequences. The detection of shadows was based on the observation that shadows changed significantly the lightness of an area without greatly modifying the colour information. Based on the linear influence of shadow to the YUV values, Schreer *et al.* [5] proposed a shadow detection for video conference applications. Bevilacqua and Roffilli[6] exploited spatial information, transparency and homogeneity to remove shadow. Two steps based on the shadow proprieties were used to find out the quite large shadows and small shadows respectively, shadow area was obtained by OR'ed the outcome of the two steps. Sohail *et al.* [7] presented a moving cast shadows removal method for outdoor environment. The approach was based on a new spatio-temporal albedo test and dichromatic reflection model and accounted for both the sun and the sky illuminations. Branca[8] proposed a approach for shadow removal in outdoor scenes. The approach detected shadow points that were characterized by a lower photometric gain than the reference background. Shadow removal techniques[3−8] mainly exploited the colours in RGB, HSV or YUV space. However, the color information are not al-

Manuscript received November 22, 2005; revised March 6, 2006.
The work was supported by the National Natural Science Foundation of PRC (No. 60574033) and the National Key Fundamental Research & Development Programs (973) of PRC (No. 2001CB309403)
*Corresponding author. *E-mail address*: xiaomeijx@163.com

ways steady with the effect of noise, illumination conditions, different camera, *etc.* Clearly these cases have a negative influence on shadow removal results.

Wang *et al.* [9] put forward a new shadow removal method for traffic scenes. The method estimated locations and orientations of shadows at first, secondly a number of points were sampled from the shadow candidates, thirdly the attributes of shadow were computed by sampled points, finally object shapes was recovered progressively instead of directly removing shadows. Wu *et al.* [10] proposed a method to separate shadow from vehicles using knowledge including results of edges detection, road direction, *etc.* In [11] a method to locate and subsequently remove shadows was presented. A camera calibration stage was needed to generate a 1-D illumination invariant shadow-free image, which was used together with the original image to locate shadow edges. Thresholding out image edges that were due to shadows and re-integrating them allowed to obtain full colour shadow-free images. We also can find that [10,11] must rely on the knowledge such as camera location, road geometry, illumination direction, *etc.* However, the information can not always be obtained reliably. A colour ratios approach to distinguish shadow from foreground was presented in [12]. This method took into account the colour ratios which were usually neglected by other approaches. Mikic *et al.* [13] presented a statistical model for separating moving objects from their cast shadows. The method ran in real-time. However, the parameters of the method were difficult to choose and it was not fit for general traffic sequence. Onoguchi[14] presented a method for eliminating shadow in outdoor scenes, which assumed shadows were on the road plane and two cameras were set at any locations. The method got rid of shadow areas in common visual fields by the way of height information. Xu D *et al.* [15] used the edge information to remove insignificant moving cast shadows in indoor scenes. Shadows could be removed successfully when shadows presented blur edges, due to the illumination source far away from the objects. In this paper, we only pay attention to moving shadow removal algorithms for traffic sequence.

The paper is organized as follows. Section 2 describes the procedure of shadow detection. In Section 3 the shadow removal algorithm based on edge information is presented. The simulation results and comparisons are then analyzed and shown in Section 4. Finally, Section 5 concludes the paper.

2 Shadow detection procedure

The algorithm links up shadow detection to decide whether there are shadows present in an image. The system will not perform the shadow processing algorithm if no shadow occurring in the image by shadow detection. The shadow detection procedure is composed of two parts: shooting-time estimate and illumination assessment. Shadow detection begins with shooting-time estimate procedure and is followed by illumination assessment. Details of each step are described in the following subsections.

2.1 Shooting-time estimation

Image shooting-time, which consists of the date and the time of each image, is stored in order to determine coarsely the plausibility of shadow occurence. The knowledge of image shooting-time can be obtained in every PC-system. In the shooting-time estimate step, if image shooting-time excludes the possibility of shadow occurring (e.g. during night or in a rainy day), then no shadow occurring is supposed and the system will not process the shadow algorithm. Otherwise, we take the assumption that the plausibility of shadow occurring is great. Then the illumination assessment is processed at the next step to decide accurately whether there are shadows in the image or not.

The selection of shadow occurring period relies on seasons. The shadow occurring period, which is a constant, is set differently based on season. For example, the period of shadow occurring in summer is longer than that in winter.

2.2 Illumination assessment

Wang *et al.* [9] extended the illumination evaluating algorithm which was proposed by Wixson *et al.* [16,17] to detect shadow. In this paper, illumination assessment in [9] is used to decide accurately whether there are shadows in an image or not. Rather than in the entire input image, illumination assessment is processed in the area of a selected foreground figure, therefore the processing time of the algorithm is saved. Two criteria are employed for the decision: energy values and relative size of the dark.

Brightness energy A scene is regarded as being in high level of brightness when objects in the scene are clearly visible. Daytime with sunshine is a typical example. On the other hand, a scene is regarded as being in a low level of brightness when objects are hardly to recognize, which typically occurs at night or rainy/misty daytimes. Since objects are usually brighter than shadows in a scene, brightness energy E_b is use to reflect the degree of visibility of objects and the level of brightness of scene. Therefore, large brightness energy indicates a high possibility that shadows exist.

Let S_d and S_b be the sets of dark and bright pixels in the selected figure, respectively. The brightness energy values E_b of S_b and darkness energy values E_d of S_d are

$$E_j = \frac{\sum_{i \in s_j} e_i}{n_j},\ j \in \{b, d\} \tag{1}$$

where n_j is the number of pixels in set S_j and e_i is the energy value of pixel i defined as

$$e_i = \frac{\sum_{j \in N_i} |I_i - I_j|}{n} \tag{2}$$

in which I_i, N_i, and n are the intensity value, neighbor set, and number of neighbors of pixel i, respectively.

Relative size of the dark P_d is also introduced, a parameter to indicate the relative size of the dark area to the bright area in a figure; P_d is defined as

$$P_d = \frac{n_d}{n_b}. \tag{3}$$

A small E_b indicates a low level of brightness. A small P_d value indicates a relative small size of shadows with respect

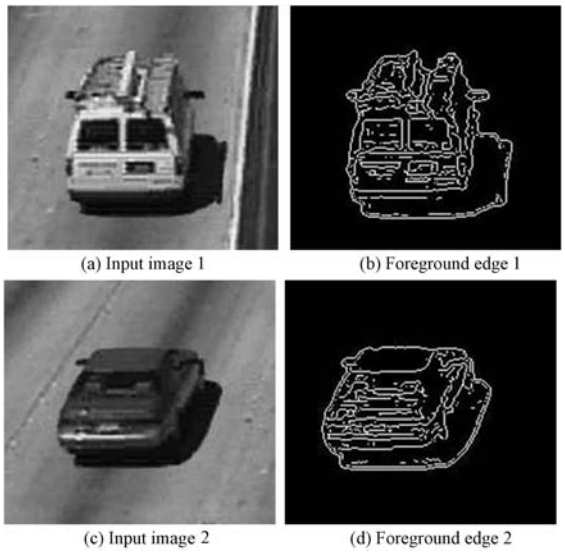


Fig. 1 Input image and foreground edge

to the scene. Hence, it can be concluded that no shadow exits whenever image has a small E_b value. However, even though an image has a large E_b value, we may not apply shadow removal to the image either, if it has a small P_d value. The detail of illumination assessment has been described in [9,16,17].

The algorithm continues for several frames in order to increase the plausibility of shadow occurring. Once a shadow has been detected, shadow removal algorithm is used only in a selected foreground region to eliminate shadow.

3 Edge-based moving shadow removal algorithm

The paper assumes the algorithm works for traffic surveillance and the foreground has been extracted. The foreground objects can be detected manually or automatically via other techniques[18,19]. In the simulation experiments, we adapted the moving detection method proposed in [18] to extract the foregrounds. We can observe that the edges of cast shadow in traffic sequence have three properties: 1) The cast shadows present sharp edges because the illumination source is far from the objects; 2) The vehicle has significant edges, however the corresponding shadow is generally edgeless; 3) The edge of cast shadow fastens on the boundary region of the moving foreground mask. Fig. 1 shows the two input images and the foreground edges correspond to them. Thought the algorithm is assumed to work for traffic scene, the proposed shadows removal method is still valid if the shadow areas are not or less textured.

The shadow elimination algorithm removes the boundary of the cast shadow and preserves object edges firstly; secondly, it reconstructs coarse object shapes based on the edge information of objects; finally, it extracts the cast shadow by subtracting the moving object from the change detection mask and performs further processing. The steps of moving shadow removal algorithm are described as follows.

3.1 Edge detection for foreground region

In our algorithm, the edge information plays an important role for shadow detection. We apply Sobel edge detection to a selected foreground region rather than apply to the entire input image. The illustration of the moving shadow removal algorithm is shown in Fig. 2. Let C_t denote the selected foreground mask, which is shown in Fig. 2(b). Edge detection for C_t marked as E_t is shown in Fig. 2(c). Subscript t is the time.

3.2 Compute initial interior edge of foreground

The purpose of this section is to remove the boundary of the cast shadow and preserve object edges. The steps of compute initial interior edge of foreground are described as follows:

Step 1. Boundary extraction for foreground mask

Computing the boundary of foreground mask is the first step to remove edge of shadow. It is easy to obtain foreground edge image EC_t because C_t is a binary image with values 0 and 1. EC_t can be obtained by first eroding C_t by B_1 and then performing the set difference between C_t and its erosion. That is,

$$EC_t = C_t - (C_t \ominus B_1) \tag{4}$$

where B_1 is a suitable structuring element, and in this paper, 3×3 pixels are used; $\ominus$ denotes erosion.

In order to remove the edge of the E_t, dilated boundary of change mask DB_t which is more than one pixel thick should be acquired. The dilated boundary of change mask DB_t is given by

$$DB_t(x,y) = \begin{cases} 1, & DE_t(x,y) = 1 \text{ and } C_t(x,y) = 1 \\ 0, & \text{otherwise} \end{cases}. \tag{5}$$

Step 2. Calculated dilated boundary of foreground mask DB_t is shown in Fig. 2 (d). And DE_t is given by

$$DE_t = EC_t \oplus B_2 \tag{6}$$

where, DE_t denotes the dilated EC_t, B_2 is the dilated structure elements, usually, $5 \times 5 \sim 9 \times 9$ pixels are used.

Note that: EC_t is on the boundary of the change mask, part of the structuring elements may be outside the image. The normal treatment of this condition is to assume that values outside the borders of the image are 0. So it is clear that using 7×7 structuring elements of 1's would result in a boundary between 3 and 4 pixels thick.

Step 3. Compute initial interior edge

The edge points of foreground mask outside DB_t are considered as initial interior edge IE_t, that is,

$$IE_t(x,y) = \begin{cases} 1, & E_t(x,y) = 1 \text{ and } DB_t(x,y) = 1 \\ 0, & \text{otherwise} \end{cases}. \tag{7}$$

IE_t is shown in Fig. 2 (e).

3.3 Refine the interior edge of moving object

After the operator of Section 3.2, almost all the edge points in cast shadow area are eliminated. However, a litter of false edges due to the noise, which is far away from

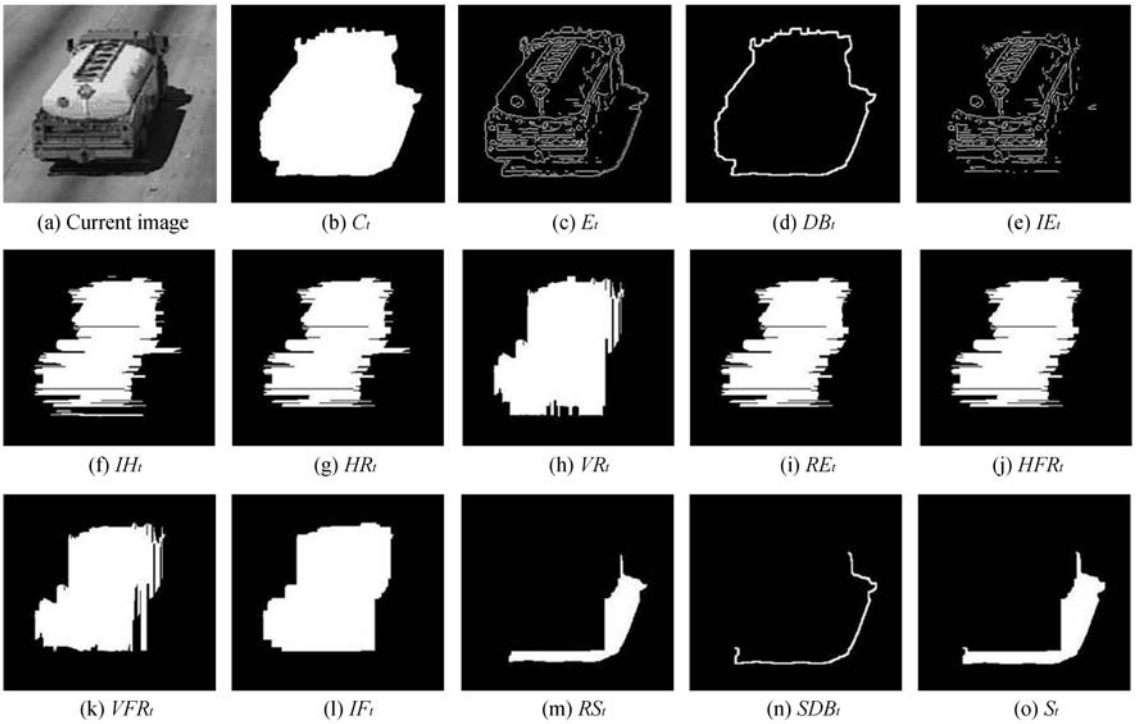


Fig. 2 The illustration of moving shadow removal algorithm

the foreground and near the boundary of the cast shadow, should be removed because it has a great effect on shadow detection. The refining process consists of horizontal operation and vertical operation. HR_t and VR_t are the results of horizontal and vertical operations on IE_t, respectively. The horizontal candidates HR_t are obtained as follows:

First, the region inside the first and last edge points (here means the points with value 1) in each row is set to 1, the resulting mask IH_t is shown in Fig. 2 (f).

Second, remove the small noise region of IH_t. The noise region tends to be smaller than the area of the object, our approach to eliminate the noise region is: To begin with the classic connected components algorithm on the IH_t. Then, the area of each region is calculated. Finally, among all connected regions, only those whose areas are smaller than 5 percent of total pixels in IH_t are considered as the noise regions. Here, the area of one region means the number of pixels within that region. VR_t is for each column, which is very similar to generation of HR_t. HR_t and VR_t are shown in Figs. 2 (g) and (h) respectively.

After horizontal operation and vertical operation, the edges of foreground region E_t in both of HR_t and VR_t are the interior edge of moving object after refining operation, which is marked as RE_t:

$$RE_t(x,y) = \begin{cases} 1, & HR_t(x,y) = 1 \text{ and} \\ & VR_t(x,y) = 1 \text{ and} \\ & E_t(x,y) = 1 \\ 0, & \text{otherwise} \end{cases}. \tag{8}$$

RE_t is shown in Fig. 2 (i).

3.4 Foreground region classification

If a foreground region contains more than one foreground, the shadows from different vehicles will possibly connect to each other in this region. In order to efficiently remove the unwanted shadows from the background, it must be determined that which edge points in RE_t belong to the same foreground. Foreground edge classification may be summarized as the following steps:

Step 1. The horizontal operation and vertical operation, which were described in Section 3.3, are applied to RE_t. HRE_t and VRE_t are the results of horizontal and vertical operation on RE_t, respectively.

Step 2. Label connected components in HRE_t and VRE_t with 8-connected labeling. Any adjacent blobs are regarded as the same foreground, if the distance between two connected blobs is less than or equal to some value T_d, and re-label the two blobs with the same number. When no further merging of any blobs is possible, the procedure is terminated. The step is necessary if an object is erroneously split in the horizontal and vertical operations. The distance between two connected blobs in HRE_t and VRE_t is defined as number of rows and columns, respectively. T_d is given as

$$T_d = \alpha \cdot \min\{\min_{i=1,2,\cdots,M_1}(HL^i), \min_{j=1,2,\cdots,N_1}(VC^j)\}$$
$$\alpha \in (0,1] \tag{9}$$

where HL^i and VC^j are the number of rows of blob labeled i in HRE_t and the number of columns of blob labeled j in VRE_t, respectively, M_1 and N_1 are the total numbers of blobs of both labeled images.

$HREM_t$ and $VREM_t$ denote the results of merging operations on HRE_t and VRE_t, respectively.

Step 3. Blobs classification. The Blobs classification algorithm can be applied to acquire the correct classification of foreground based on $HREM_t$ and $VREM_t$. Let $O_H = \{Oh^1, Oh^2, \cdots, Oh^M\}$ be the blobs of the labeled image from $HREM_t$, and let $O_V = \{Ov^1, Ov^2, \cdots, Ov^N\}$ be the blobs of the labeled image from $VREM_t$, M_1 and N_1 be the numbers of blobs of both labeled images. The final classification of foreground is denoted as OM_t. The blobs classification operation is described as follows:

```
For i = 1, 2, ··· , M
    For j = 1, 2, ··· , N
        If Oh^i ∩ Ov^j ≠ ∅
            Om^k = Oh^i ∩ Ov^j;
            k = k + 1;
        End
    End
End
```

3.5 Construct the moving object

Assume q foregrounds are contained in a moving object mask, and let $IFE_t^k(k = 1, 2, \cdots, q)$ denote the edge of k^{th} moving object, i.e. the edge pixels labeled k in Section 3.4. We begin with the operation of constructing the k^{th} moving object based on its edge IFE_t^k. q foregrounds construction process is as follows.

Step 1. Horizontal operation and vertical operation are executed for the edge in k^{th} moving object. HFR_t^k and VFR_t^k denote the results of the horizontal operation and vertical operation, respectively.

Step 2. After finding both of HFR_t^k and VFR_t^k candidates, the union of HFR_t^k and VFR_t^k, which is marked as $FR_t^k(k = 1, 2, \cdots, q)$, is marked as absolute interior part of k^{th} foreground through logical OR operation.

$$FR_t^k(x, y) = \begin{cases} 1, & HFR_t^k(x, y) = 1 \text{ or} \\ & VFR_t^k(x, y) = 1 \\ 0, & \text{otherwise} \end{cases}. \quad (10)$$

Step 3. Interior regions of q foregrounds are shown as:

$$IF_t(x, y) = \begin{cases} 1, & FR_t^k(x, y) = 1, \ \forall k = 1, 2, \cdots, q \\ 0, & \text{otherwise} \end{cases}. \quad (11)$$

HFR_t^k, VFR_t^k (k is 1 in Fig. 2) and IF_t are shown in Figs. 2 (j∼l).

3.6 Compute seed region of moving cast shadow

With C_t, DB_t, and IF_t, the initial seed region IS_t is acquired by

$$IS_t = C_t - DB_t - IF_t. \quad (12)$$

After obtaining initial seed region IS_t, an algorithm that uses label technology as the basis for finding real seed region of shadow RS_t is as follows:

First, the connected components algorithm is applied to IS_t to mark each isolated region. Then, the size of each region is calculated. Among all connected seed points, only the first largest q regions will be considered as the correct seed regions. Here the size of one region means the number of pixels within it. RS_t is shown in Fig. 2 (m).

3.7 Find real moving shadow region

From RS_t we see the detected shadow seed region is almost the region of the shadow. However, RS_t does not include the shadow part in dilated boundary DB_t, so further processing can be applied to recover shadow region in dilated boundary area. Let S_t denote the final moving shadow using our shadow algorithm. Further processing of k^{th} moving shadow is illustrated as follows.

Step 1. Label connected components in RS_t, and let $RS_t^k(k = 1, 2, \cdots, q)$ denote the region labeled k.

Step 2. The blob $RS_t^{k\prime}s$ containing rectangle CS^k is defined as the minimal rectangle that covers the blob pixels.

Step 3. The upper, lower, right and left sides of CS^k is translated for several pixel along upper, lower, right and left directions, respectively, i.e. we want to enlarge the rectangle by several pixels. The width of pixels is the same as the size of structuring element B_1. For example, if B_1 is 7×7 in this paper, then the four sides containing rectangle is translated 3 or 4 pixels along its direction.

Step 4. The locus of parts of dilated boundary DB_t within this rectangle are classified as the $k's$ shadow blob, which is marked as SDB_t^k. SDB_t^k and S_t are shown in Figs. 2 (n) and (o).

4 Simulation results and comparisons

The proposed method has been implemented on a personal computer with an INTEL Pentium IV 1.0-GHZ CPU using Matlab 6.5. In order to analyze the robustness and effectiveness of the proposed method, seven experiments under different conditions are demonstrated in this paper. For comparisons, multi-gradient (MG)[9] and HSV[10] were implemented. In the experiment, α, B_1 and B_2 are $0.5, 3 \times 3$ and 7×7 structuring elements, respectively.

The first experiment (see Fig. 3) shows the capability of the proposed algorithm to eliminate unwanted shadows when a foreground region contains only one foreground. Figs. 3 (a) and (e) are original images, Figs. 3 (b) and (f) are extracted foregrounds, Figs. 3 (c) and (g) are extracted shadows by our shadow algorithm, Figs. 3 (d) and (h) are foregrounds after shadow elimination process. Despite multiple vehicles with different color in original image and the vehicle widows are as dark as the shadows, the shadow removal performs well and the vehicles are extracted successfully.

Fig. 4 indicates two examples when the original images show multiple vehicles moving along a road and the shadow connects to the other vehicles. Figs. 4 (a) and (e) are original images, Figs. 4 (b) and (f) are extracted foregrounds, Figs. 4 (c) and (g) are extracted shadows by our shadow algorithm, Figs. 4 (d) and (h) are foregrounds after shadow elimination process. All the shadows are clearly removed with the proposed method. Note that, the shadow regions under bodywork were not detected.

Fig. 5 indicates two examples when only the shadow appeared in a foreground region. Figs. 5 (a) and (e) are original images, Figs. 5 (b) and (f) are extracted foregrounds, Figs. 5 (c) and (g) are extracted shadows by our shadow algorithm, Figs. 5 (d) and (h) are foregrounds after shadow elimination process. Though a vehicle did not appear in the

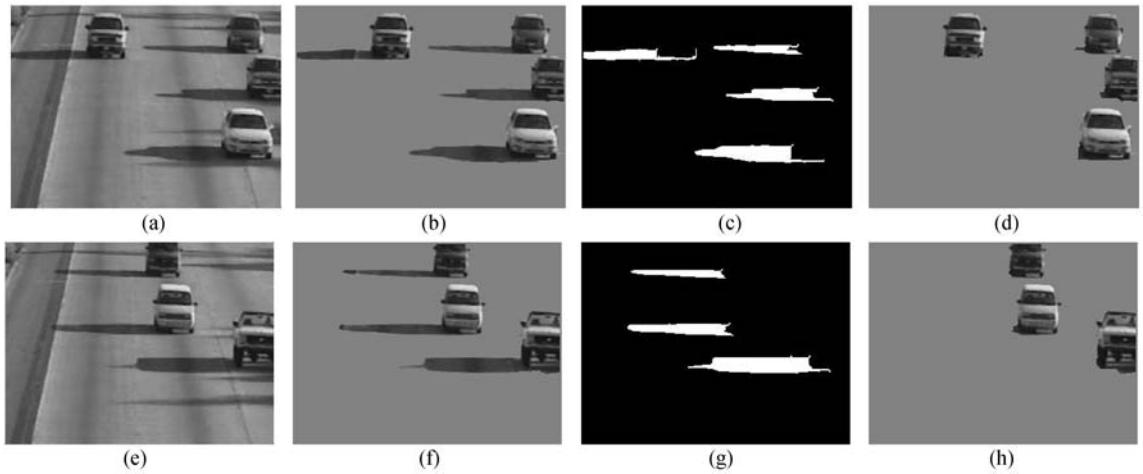


Fig. 3 Results of shadow elimination when a foreground region contains only one foreground

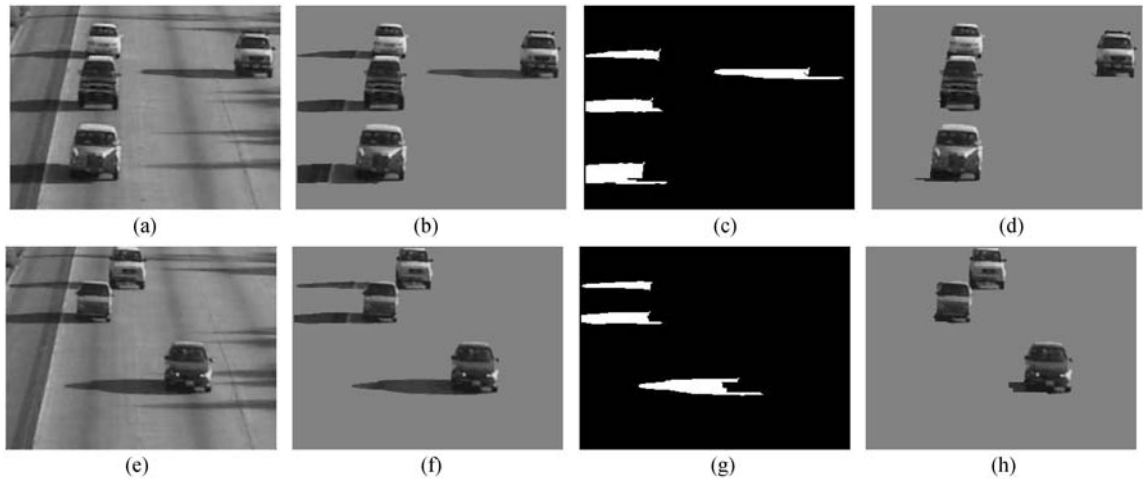


Fig. 4 Results of shadow elimination when a foreground region contains more than one foreground

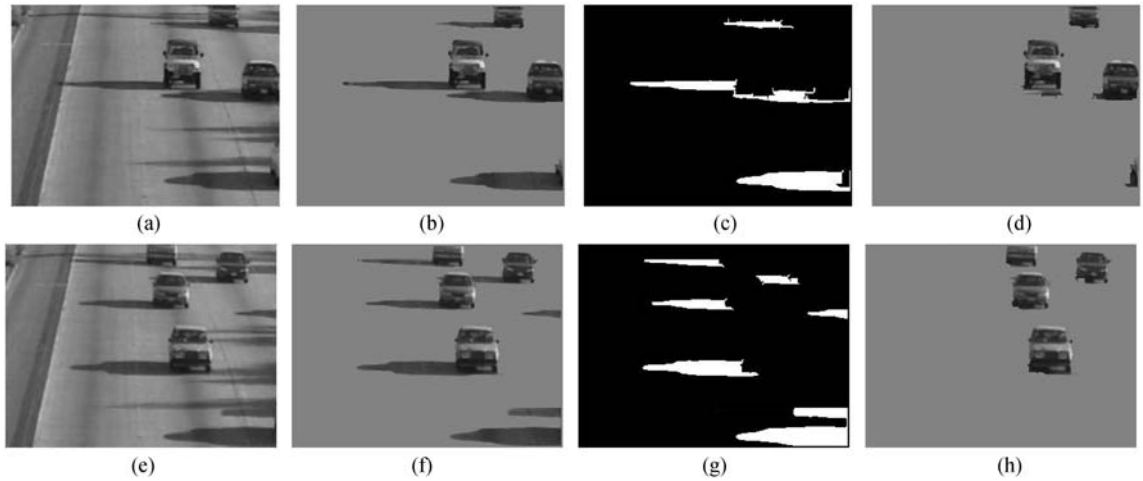


Fig. 5 Results of shadow elimination when only the shadow appeared in a foreground region

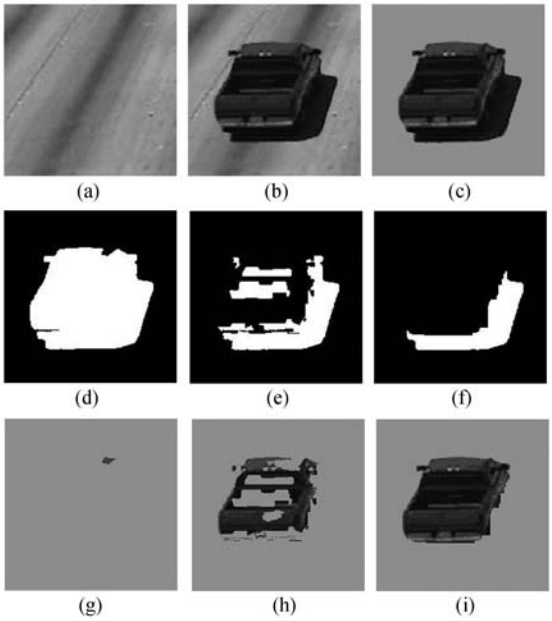


Fig. 6 Results of shadow elimination for a red vehicle

original together with its shadow, the proposed method still eliminated effectively shadow from extracted foreground mask. The conclusion can be made that the proposed method indeed works robustly regardless of changes in shadow appearance.

An image of a red car is shown in Fig. 6 (b). In Fig. 6 (a), the background color image was generated by the background estimation. After subtracting the background frame from the input frame, the extracted foreground image is shown in Fig. 6 (c), where the gray region is the background region. In Figs. 6 (d∼f), the results of MG, HSV and our method are shown respectively. Figs. 6 (g∼i) are final foregrounds after shadow removal correspond to Figs. 6 (d∼f), respectively.

In Fig. 6 (d), the car is classified as shadow. Clearly, the result of (e) is better than (d). In Fig. 6 (e) based on the chrominance value, the red part of the car is clearly classified as non-shadow because its chrominance value is different from the shadow. However, some parts of the car such as window and the black parts are recognized as shadow since they have similar luminance and colour level as the shadow image. Obviously, the result of Fig. 6 (f) is much better than those in Fig. 6 (d) and Fig. 6 (e), all shadow pixels have been removed.

Fig. 7 shows the case when the shadow is from a black vehicle. Fig. 7 (a) is background image; Fig. 7 (b) is original image; Fig. 7 (c) is extracted foreground image, where the gray region is the background region; Figs. 7 (d∼f) are the extracted shadow by MG, HSV and our method, respectively; Figs. 7 (g∼i) are final foregrounds correspond to Figs. 7 (d∼f), respectively. In Figs. 7 (d) and (e), the shadows are classified wrongly since shadows have similar luminance and colour level as the foreground. From Fig. 7 (f), the shadow part of the car is clearly classified as shadow by our shadow algorithm based on edge information though its luminance and colour value are similar to the black car.

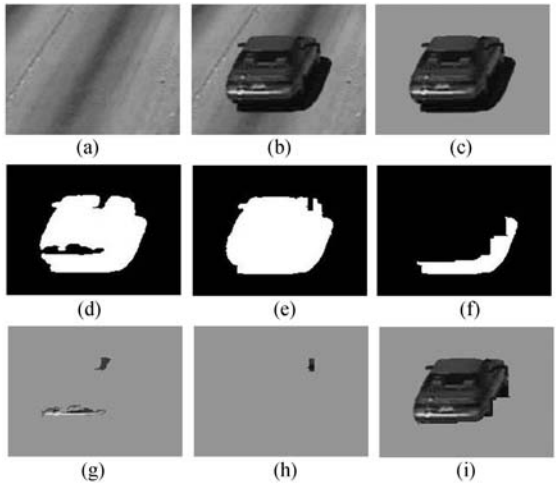


Fig. 7 Results of shadow elimination for a black vehicle

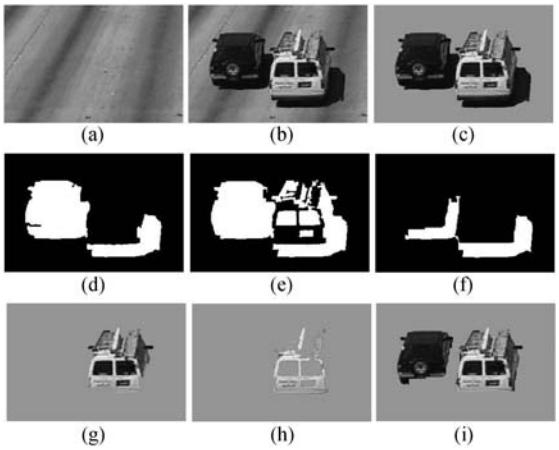


Fig. 8 Results of shadow elimination for two vehicles

Fig. 8 shows two vehicles with different colour and the right shadow connects to the right white car. Fig. 8 (a) is background image; Fig. 8 (b) is original image; Fig. 8 (c) is extracted foreground image, where the gray region is the background region; Figs. 8 (d∼f) are the extracted shadow by MG, HSV and our method respectively; Figs. 8 (g∼i) are final foregrounds correspond to Figs. 8 (d∼f), respectively. In Fig. 8 (d) and (e), only the bright parts of foreground are classified as non-shadow, while the dark parts are classified as shadow by error. From Fig. 8 (f) we can see all the pixels are accurately expect for a few pixels by our shadow removal algorithm.

Fig. 9 shows shadow elimination results of original image with multiple different size and color vehicles. Fig 9 (a) is background image; Fig. 9 (b) is original image; Fig. 9 (c) is foreground image; Figs. 9 (d∼f) are the extracted shadows by MG, HSV and our method respectively; Figs. 9 (g∼i) are final foreground correspond to Figs. 9 (d∼f), respectively. From Figs. 9 (d) and (e), we can see that the shadow methods of MG and HSV are only effective for bright foreground. In Fig. 9 (d), our approach performs well compared

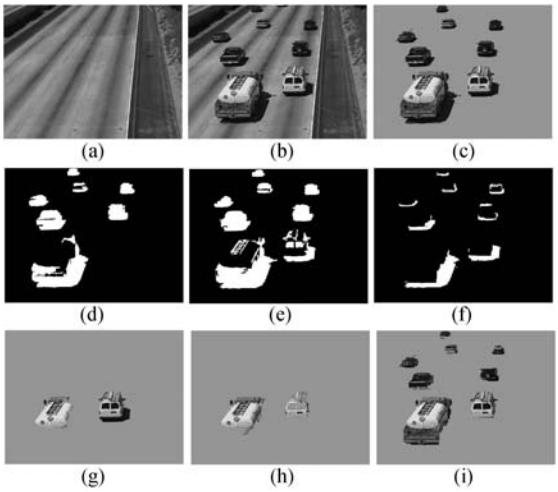


Fig. 9 Results of shadow elimimation for multi-vehicles

to methods of MG and HSV. Note that the shadow elimination algorithm performs better for cases where both objects and their shadows are closer to the camera.

5 Conclusions

In this paper, we have proposed innovative solutions for shadow elimination from extracted foreground based on edge information. The contributions of this paper are: 1) Without exploiting any a prior knowledge of traffic scene such as the direction of illumination. 2) The algorithm is robust to widely different shadow orientations, shadow appearance and foreground materials such as color, size. 3) The parameter can be chosen in a wide range. 4) No matter how many shadows appear in the video sequence, all these shadows can be precisely eliminated from the background with the proposed method. However, edge information, which is extracted by Sobel edge detection, has a large effect on algorithm, so appropriate threshold value is important to the robustness of our shadow removal method. In the future, an implementation in complex environment will be explored, for example, surface texture (e.g. grass, brick, tiles, road, *etc*), types and shapes of foreground, working environment (outdoor or indoor), *etc*..

References

[1] Y. Sonoda, T. Ogata. Separation of Moving Objects and Their Shadows, and Application to Tracking of Loci in the Monitoring Images. In *Proceedings of 1998 Fourth International Conference on Signal Processing*, Beijing, China, vol. 2, pp. 1261−1264, 1998.

[2] K. Onoguchi. Shadow Elimination Method for Moving Object Detection. In *Proceedings of Fourteenth International Conference on Pattern Recognition*, Los Alamitos, California, vol. 1, pp. 583−587, 1998.

[3] G. S. K. Fung, N. H. C. Yung, G. K. H. Pang, A. H. S. Lai. Effective Moving Cast Shadows Detection for Monocular Color Image Sequences. In *Proceedings of 11th International Conference on Image Analysis and Processing*, Los Alamitos, California, pp. 404−409, 2001.

[4] R. Cucchiara, C. Grana, M. Piccardi, A. Prati. Detecting Objects, Shadows and Ghosts in Video Streams by Exploiting Color and Motion Information. In *Proceedings of 11th International Conference on Image Analysis and Processing*, Los Alamitos, California, pp. 360−365, 2001.

[5] O. Schreer, I. Feldmann, U. Goelz, P. Kauff. Fast and Robust Shadow Detection in Videoconference Applications. In *Proceedings of International Symposium on Video/Image Processing and Multimedia Communications 2002*, Zadar, Croatia, pp. 371−375, 2002.

[6] A. Bevilacqua, M. Roffilli. Robust Denoising and Moving Shadows Detection in Traffic Scenes. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Kauai Marriott, Hawaii, pp. 1−4, 2001.

[7] S. Nadimi, B. Bhanu. Physical Models for Moving Shadow and Object Detection in Video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 8, pp. 1079−1087, 2004.

[8] A. Branca, G. Attolico, A. Distante. Cast Shadow Removing in Foreground Segmentation. In *Proceedings of 16th International Conference on Pattern Recognition*, Los Alamitos, California, pp. 214−217, 2002.

[9] J. M. Wang, Y. C. Chung, C. L. Chang, S. W. Chen. Shadow Detection and Removal for Traffic Images. In *Proceedings of the 2004 IEEE International Conference on Networking, Sensing and Control*, Piscataway, United States, vol. 1, pp. 649−654, 2004.

[10] Y. M. Wu, X. Q. Ye, W. K. Gu. A Shadow Handler in Traffic Monitoring System. In *Proceedings of IEEE Vehicular Technology Conference*, Piscataway, NJ, vol. 1, pp. 303−307, 2002.

[11] G. D. Finlayson, S. D. Hordley, M. S. Drew. Removing Shadows from Images Using Retinex. In *Proceedings of the 10th Color Imaging Conference: Color Science, Systems, and Applications*, Scottsdale, AZ, United States, pp. 73−79, 2002.

[12] K. Barnard, G. Finlayson. Shadow Identification Using Colour Ratios. In *Proceedings of the 8th Color Imaging Conference: Color Science, Systems, and Applications*, Scottsdale, AZ, United States, pp. 97−101, 2000.

[13] I. Mikic, P. C. Cosman, G. T. Kogut, M. M. Trivedi. Moving Shadow and Object Detection in Traffic Scenes. In *Proceedings of IEEE International Conference on Pattern Recognition*, Los Alamitos, California, vol. 1, pp. 321−324, 2000.

[14] K. Onoguchi. Shadow Elimination Method for Moving Object Detection. In *Proceedings of Fourteenth International Conference on Pattern Recognition*, Brisbane, Australia, vol. 1, pp. 583−587, 1998.

[15] D. Xu, X. L. Li, Z. K. Liu, Y. Yuan. Cast Shadow Detection in Video Segmentation. *Pattern Recognition Letters*, vol. 26, no. 1, pp. 91−99, 2005.

[16] L. Wixson. Illumination Assessment for Vision-based Traffic Monitoring. In *Proceedings of 13th International Conference on Pattern Recognition*, Los Alamitos, California, vol. 3, pp. 56−62, 1996.

[17] L. Wixson, K. Hanna, D. Mishra. Improved Illumination Assessment for Vision-based Traffic Monitoring. In *Proceedings of IEEE Workshop on Visual Sfurveillance*, Los Alamitos, California, pp. 34−41, 1998.

[18] Z. Q. Hou, C. Z. Han. A Background Reconstruction Algorithm Based on Pixel Intensity Classification in Remote Video Surveillance System. In *Proceedings of 7th International Conference on Information Fusion*, Stockholm, Sweden, pp. 754−759, 2004.

[19] C. Kim, N. Hwang. Fast and Automatic Video Object Segmentation and Tracking for Content-based Applications. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 2, pp. 122−128, 2002.

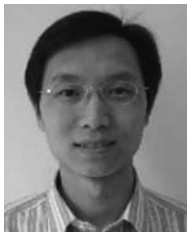
Mei Xiao received her B. Sc. degree from Chang'an University, China, in 2000, and the M. Sc degree from Chang'an University in 2003, China. She is currently a Ph. D. student of Xi′an Jiaotong University, China.

Her research interests include computer vision, image processing and information fusion.

Chong-Zhao Han received his B. Sc. degree in automation from Xi′an JiaoTong University, China, in 1968, and the M. Sc degree from Chinese Academy of Sciences, China, in 1981. He is currently a professor in the School of Electronic and Information Engineering at Xi'an JiaoTong University, China.

He has over 40 years of experience in industrial applications and academic research. He has published 7 books and more than 140 journal and conference papers. His research interests include stochastic system analysis, estimation theory, nonlinear analysis and information fusion.

Prof. Han is a member of board of directors in Chinese Association of Automation, and also the vice chair of Shanxi Provincial Association of Automation.



Lei Zhang received his B. Sc. degree from Chang'an University, China, in 1998, and the M. Sc degree from Chang'an University, China, in 2005. He is currently a Ph.D. student of Chang'an University, China.

His research interests include computer vision, image processing and intelligent vehicle.