2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

Moving Shadow Detection with Low- and Mid-Level Reasoning

Ajay J. Joshi, Stefan Atev, Osama Masoud, and Nikolaos Papanikolopoulos
Dept. of Computer Science and Engineering, University of Minnesota – Twin Cities
{ajay,atev,masoud,npapas}@cs.umn.edu

Abstract— **In this paper, we propose a multi-level shadow identification scheme which is generally applicable without restrictions on the number of light sources, illumination conditions, surface orientations, and object sizes. In the first level, we use a background segmentation technique to identify foreground regions which include moving shadows. In the second step, pixel-based decisions are made by comparing the current frame with the background model to distinguish between shadows and actual foreground. In the third step, this result is improved using blob-level reasoning which works on geometric constraints of identified shadow and foreground blobs. Results on various indoor and outdoor sequences under different illumination conditions show the success of the proposed approach.**

I. Introduction

The problem of moving shadow detection has had great interest in the computer vision community because of its relevance to visual tracking, object recognition, and many other important applications. One way to define the problem is to cast it as a classification problem in which image regions are classified as either foreground objects, background, or shadows cast by foreground objects. Despite many attempts, the problem remains largely unsolved due to several inherent challenges: (i) Dark regions are not necessarily shadow regions since foreground objects can be dark too; (ii) Self-shadows should not be classified as cast shadows since they are part of the foreground object; and (iii) A commonly used assumption that these shadows fall only on the ground plane is not valid to general scenes. In this paper, we address these challenges by proposing a shadow detection method which does not put any restrictions on the scene in terms of illumination conditions, geometry of the objects, and size and position of shadows. Results obtained using the proposed approach, in varied conditions, are very promising.

II. Related Work

There has been significant work done recently that deals with the problem of moving cast shadows. Reference [1] presents a comprehensive survey of most of the known methods that deal with moving shadow identification. It details the requirements of shadow detection methods, identifies important related issues and makes a quantitative and qualitative comparison between different approaches in the literature. In [2], shadow detection is done using heuristic evaluation rules based on many parameters which are pixel-based as well as geometry-based. The authors assume a planar and textured background on which shadows are cast. They also assume that the light source is not a point source as they use the presence of penumbra for detection. Our objective was not to impose any such restrictions of planar or textured background, nor to assume a specific light source. Some methods use multiple cameras for shadow detection [3]. Shadows are separated based on the fact that they are on the ground plane, whereas foreground objects are not.

There has been an interest in using different color spaces to detect shadows. Reference [4], for example, uses normalized values of the R, G, and B channels, and shows that they produce better results than the raw color values. Their system relies on color and illumination changes to detect shadow regions. In [5], shadow detection is done in the HSV color space and automatic parameter selection is used to reduce prior scene-specific information necessary to detect shadows. A technique which uses color and brightness information to do background segmentation and shadow removal is outlined in [6]. In [7], the authors describe a technique which uses color and brightness values to deal with the problem of ghosts - regions which are detected as foreground but not associated with any moving object. However, the techniques based only on color and illumination are not effective when foreground color closely matches that of the background. In our work, we go a step further in pixel-based decisions by using edge magnitude and edge gradient direction cues along with color and brightness to separate foreground from background. The method in [8] models the values of shadowed pixels using a transformation matrix which indicates the amount of brightness change a pixel undergoes in the presence of a cast shadow. Another method which uses geometry to find shadowed regions is outlined in [9]. It produces height estimates of objects using their shadow positions and size by applying geometric reasoning. However, shadows need to be on the same plane so that the height estimates be valid. A sophisticated approach which works on multiple levels in a hierarchy is shown in [10]. Low-level processing is done at the first level and as we go higher up, processing controls parameters which change slower. This hierarchical approach takes care of both fast- and slow-changing factors in the scene. However, geometric constraints which are applied make the algorithm applicable only to traffic sequences.

III. Approach

In this work, we implement a three-stage shadow detection scheme. In the first step, background segmentation is done using the mixture of Gaussians technique from [11], as modified in [12]. The next step presents a parametric approach in which four parameters are computed for each pixel and a pixel-wise decision process separates shadows

and foreground. In order to improve the results obtained in this step, further processing is done in the next step which works on the blob level to identify and correct misclassified regions. Both pixel-based and geometric cues are eventually used to make a shadow/ foreground decision. However, the geometric cues applied are general enough to allow their application to all commonly occurring scenes.

A. Step 1

This step implements a background update procedure and maintains a foreground mask, which the next two steps process. The learning rate of the background update is tuned depending on the type of sequence and the expected speed of motion. This foreground mask contains moving objects and their moving shadows. Static objects and static shadows are automatically separated and are not considered any further in the processing. This approach differs from a few other approaches which use statistical measures to do background segmentation and shadow removal simultaneously, i.e., the same measures are used to differentiate between background/foreground and foreground/shadow. An example of that is [6], which uses color and intensity measures for background segmentation. An advantage of our approach is that it uses a sophisticated background maintenance technique which takes care of static objects. Secondly, there are more parameters that differentiate between shadows and foreground than shadows and background. Once we have a foreground mask, these parameters can be used effectively, without worrying about similarities between the measure for shadows and background, since background pixels can never be mistaken for shadow after this step.

B. Step 2

For effective shadow removal, we need to use features in which shadows differ from foreground. Such differences can lead to preliminary estimates for shadow/foreground separation. We experimented using different measures such as intensity, color, edge information, texture, and feature points, and found the following subset to be the most effective in shadow detection. Thus, for each pixel the following four parameters are computed by comparing the current frame with a constantly updated background model: a) Edge magnitude error (E_{mag}), b) Edge gradient direction error (E_{dir}), c) Intensity ratio (I_r) [6], and d) Color error (C_e) [6]. Edge magnitude error is the absolute difference between the current frame edge magnitude and background frame edge magnitude at each pixel. If the edge magnitudes at a pixel in the current frame and the background frame are m_1 and m_2 respectively, we then have

$$E_{mag} = |m_1 - m_2|. \tag{1}$$

Edge gradient direction images represent edge gradient direction (angle) for each pixel scaled between values of 0 and 255. Edge gradient direction error is the difference between gradient directions of current frame and background frame for each pixel. If d_1 and d_2 denote the gradient direction values for a pixel in current frame and background frame respectively, we then obtain

$$E_{dir} = \min(|d_1 - d_2|,\ 255 - |d_1 - d_2|). \tag{2}$$

This gives the scaled angle between the edge gradient directions in the current frame and the background frame. If there are changes in edge magnitude, or if a new edge occurs at a pixel where there was no edge in the background model, or if an edge disappears, it is highly likely that the pixel belongs to a foreground object. Edge detection is carried out using simple forward difference in the horizontal and the vertical directions. Our experiments show that for our purposes, this works better than using any other edge detection operators like Sobel, Prewitt or others. Shadows do not significantly modify the edge gradient direction at any pixel. On the other hand, the presence of a foreground object will generally substantially modify the edge gradient direction at a pixel in the frame. These two edge measures are important cues in shadow detection. They work best along the edges of the foreground object while the other two measures of intensity and color work well in the central region of the object. These edge measures also work extremely well where foreground and background have significant difference in texture. For regions in which the foreground color matches that of the background, edge cues are the most reliable ones for shadow detection.

The Intensity ratio I_r can be easily explained using the color model in [6]. Given a pixel in the current frame, we project the point in RGB color space that represents the pixel's color on the line that connects the RGB space origin and the point representing the pixel background color according to the background model. The intensity ratio is calculated as the ratio of two distances: (a) the distance from the origin to the point projection, and (b) the distance from the origin to the background color point. Color error C_e is calculated as the angle between the line described above and the line that connects the origin and the point representing the current pixel color. Shadows show a lower intensity than background, while maintaining the background color. On the other hand, color change generally indicates the presence of a foreground object.

Probability Mapping

We need to find the probability of a pixel being a shadow pixel based on the four parameters described above. Let the events that the pixel is a shadow pixel and foreground pixel be represented by S and F respectively. Our goal is to make a decision whether a pixel is shadow or foreground. We use a maximum likelihood approach to make this decision by comparing the conditional probabilities of feature values given shadow and foreground. Assuming that these four parameters are independent, we get

$$P(E_{mag}, E_{dir}, I_r, C_e|S) = P(E_{mag}|S) \ldots P(C_e|S). \tag{3}$$

We used a number of video frames along with their ground truth data and inspected the histograms of the edge magnitude

error and the edge gradient direction error for shadow pixels and foreground pixels. E_{dir} histogram of shadow pixels, as in Fig. 1, shows exponential behavior with significant peaks corresponding to values of $0°$, $45°$, $90°$, and $180°$. The exponential curve decays fast initially, but has a long tail. The peaks are aberrations caused by dark regions, where edge gradient directions are highly quantized. Since we use the edge magnitude error, E_{mag}, as another measure apart from edge gradient direction error, the errors due to peaks do not lead to erroneous results. Our inspection also showed that E_{mag} exhibits similar statistical properties as E_{dir}, but without the peaks. In the case of foreground pixels, the histograms for E_{dir} and E_{mag} resembled a uniform distribution with E_{dir} showing similar peaks as mentioned above. The difference in the distributions is the basis for differentiating between shadows and foreground using these two features. Both these distributions are intuitively expected.

To model $P(E_{mag}|S)$ and $P(E_{dir}|S)$, we use the exponential functions in Equations (4) and (5). The variances of these exponentials (λ_1,λ_2) are parameters that can be tuned. In the equations, ω_1 and ω_2 are used for appropriate scaling.

The histograms computed for the intensity ratio measure (I_r) in shadow regions have sigmoid function-like shape as shown in Fig. 2. Color error histograms show similar behavior except that shadow pixel frequency is high for small error values. These behaviors are modeled by Equations (6) and (7). β_1 and β_2 provide necessary shift in these equations. δ_1 and δ_2 are scaling factors. Histograms of I_r and C_e were found to have a close to uniform distribution for foreground pixels.

$$P(E_{mag}|S) = (\omega_1/\lambda_1) \cdot \exp(-E_{mag}/\lambda_1) \quad (4)$$

$$P(E_{dir}|S) = (\omega_2/\lambda_2) \cdot \exp(-E_{dir}/\lambda_2) \quad (5)$$

$$P(I_r|S) = \delta_1/(1+\exp(-(I_r-\beta_1)/\sigma_1)) \quad (6)$$

$$P(C_e|S) = \delta_2 - \delta_2/(1+\exp(-(C_e-\beta_2)/\sigma_2)) \quad (7)$$

Once we have probability maps using the above mapping for each parameter for every pixel, we blur the log-probability maps so as to account for the information of nearby pixels. This blurring is carried out for a box of 3x3 pixels around the given pixel. In case the conditional probabilities are around 0.5, blurring helps as it brings in neighborhood information to aid the decision making process. The probability maps are then multiplied together as in Equation (3). A final decision is made based on comparing the conditional shadow probability and conditional foreground probability thus obtained. Table I reports typical parameter values and issues for the above mapping.

C. Step 3

The second step is restricted to local pixels to make a decision as to whether a pixel is shadow or foreground. The farthest it goes is a 3x3 square of pixels around, which happens when log-probability maps are blurred. For good foreground object recovery, this is not enough. There are many regions of misclassification which make the object

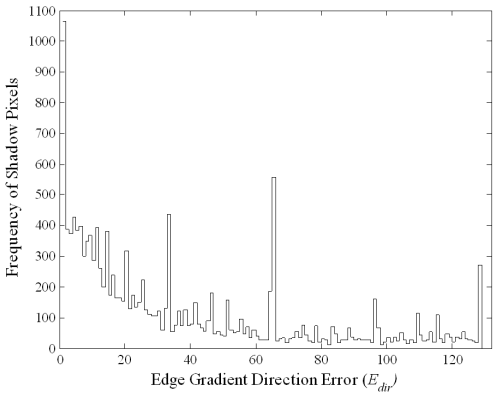


Fig. 1. Histogram of E_{dir} values for shadow pixels. Note the significant peaks corresponding to $0°$, $45°$, $90°$, and $180°$- aberrations caused due to dark regions of the image.

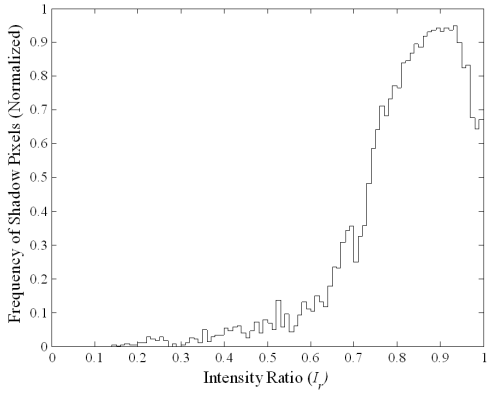


Fig. 2. Normalized histogram of I_r values for shadow pixels. This can be reasonably modeled by the sigmoid function in Equation (6).

shape estimate erroneous. Thus, we need some technique to recover shape from frames obtained after step 2. Blob-level reasoning is a step further towards this objective. The processing is done at a higher level in order to identify connected components in an image and provide a better estimate of shadow positions.

At this stage, we have an image with shadow and foreground blobs marked by the previous step. Blob labeling is done along with other computations like blob area, perimeter, and the number of perimeter pixels that are neighbors to pixels of a blob of another type (shadow pixels neighboring foreground pixels and vice versa). In order to improve shadow detection accuracy, we propose to reason out misclassified blobs (e.g., flip shadow blob to foreground blob) based on the heuristic and metrics as follows: 1) Blob area - the smaller the area, the easier it is to flip. 2) The ratio of the length in contact with another type blob to the length touching background - if this ratio is large for a blob, it is likely that the blob has been misclassified. 3) Whether flipping that blob connects two previously unconnected blobs - in case it does, it is less likely to have been misclassified.

TABLE I

TYPICAL PARAMETER VALUES FOR STEP 2

Parameter	Notes	Typical Values
λ_1	Lower for darker scenes –	50 - 80
λ_2	E_{mag} and E_{dir} decay fast for dark scenes	30 - 50
σ_1	Lower for steeper rise of sigmoid – when shadow and foreground intensity and color	0.10 - 0.15
σ_2	show distinct separation	4-6
β_1	Control shift of the sigmoid functions. Significant tuning	0.4 - 0.6
β_2	parameters based on shadow strength	0.85-0.9

TABLE II

PARAMETER DESCRIPTION AND RELATIONS FOR STEP 3

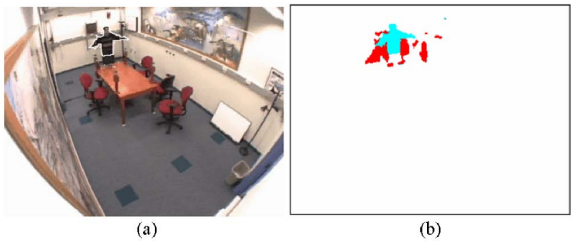
Parameter	Description	Notes
BA	Blob Area	In Pixels
P_{fg}, P_{bg}	Percent of perimeter of blob which is foreground, background respectively	Ratio P_{fg}/P_{bg} has an upper threshold T_h
T_h	Upper Threshold for Ratio P_{fg}/P_{bg}	Depends on C_{td} and whether $BA < T$ or $BA \geq T$
C_{td}	True if flipping connects two unconnected blobs, false otherwise	Influences T_h – higher T_h if C_{td} is true
T	Threshold for BA	Influences T_h – higher T_h if $BA \geq T$

IV. RESULTS

The video clips Highway I, Highway II, Intelligent Room, Laboratory, Campus, and associated ground truth data for the sequence Intelligent Room are courtesy of the Computer Vision and Robotics Research Laboratory of UCSD.

An important result of using edge-based measures in shadow detection is the improvement in performance at locations where the scene has highly reflective surfaces. Color change occurs on reflective surfaces even if they are under shadow and makes our premise on C_e invalid. Color-based and intensity-based measures cannot effectively differentiate between such reflective surfaces under shadow and foreground. However, since the surface contains similar edge gradients even under shadow, it is possible to classify it as a shadow rather than foreground. In Fig. 3 below, blue regions indicate foreground, red indicates shadow, and white indicates background as detected by our algorithm. The figure shows that a substantial part of the reflection on the table is detected as shadow and not foreground even though color changes occur.

As mentioned earlier, the removal of holes results in cleaner images and better foreground object contour calculation. Fig. 4 and Fig. 5 show such a case. The figures on the left show the results of processing using only the first two steps of the algorithm. The ones on the right are the output of the final stage. In Fig. 4(a), a number of foreground pixels are marked as shadow due to two reasons: the color similarity between the shirt of the person and the color of background, and the shadow of the person's hand on his shirt, which

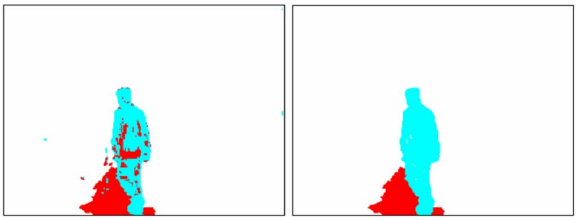


(a) (b)

Fig. 3. Result on a frame of the Intelligent Room sequence. Note the shadow on table (reflective surface).

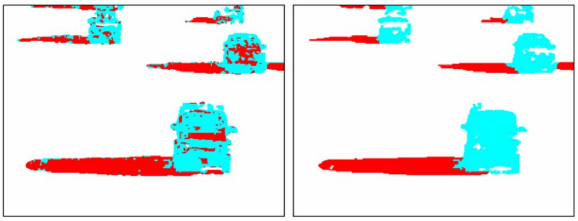
make the intensity similarity significant. Problems caused by indirect cast shadows are also reduced by the technique, as shown in Fig. 4(b).

Windshields cause a significant problem in detecting the foreground correctly. Since windshields generally occur as dark regions, they are often misclassified as shadows. All the four parameters mentioned above cannot consistently classify windshields as foreground. Fig. 5 shows such a case. Fig. 5(b) shows how blob-level reasoning substantially reduces that problem.



(a) (b)

Fig. 4. Results on a frame of sequence Laboratory. Note how step three of the algorithm reduces problems due to intensity and color matching.

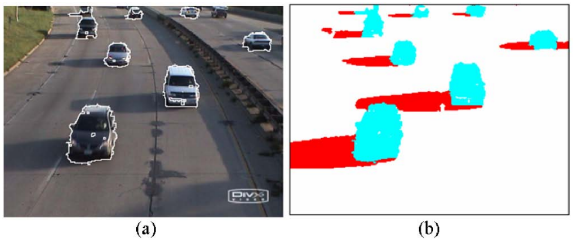


(a) (b)

Fig. 5. Results on a frame of sequence Highway I. Note how step three of the algorithm reduces problems due to misclassified windshields of cars.

In the following, we present results on test sequences for which ground truth data was available. For each sequence, we used the shadow discrimination accuracy [1] as the measure to quantify performance. Figs. 6-11 show the original frames with foreground contours overlaid in the figure on the left along with the detected shadow, foreground and background regions in the figure on the right. To show how the performance of our method varies, we have shown two frames per sequence. The first shows one of the best results using our method, in terms of discrimination accuracy and the second shows one of the worst.

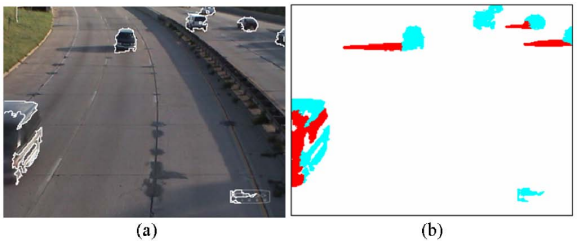
In Fig. 7(a), road surface has a strip of dark color which



(a) (b)

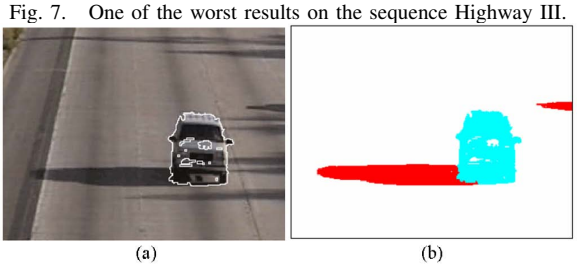
Fig. 6. One of the best results on the sequence Highway III. Note how contours fit the foreground closely.

is coincidentally aligned perfectly with the car windshield which makes distinction difficult and results in low discrimination accuracy. In Fig. 10, we can see that the shadows are present on multiple planes of the background. Our algorithm does not put any restriction on the planes on which the shadow is cast. A strip in the background matches very closely in color with the persons shirt in Fig. 10. The edges are also oriented along the same direction. A small strip is therefore classified as shadow after step two. However, it is further flipped in step 3 as it is completely surrounded by foreground regions.



(a) (b)

Fig. 7. One of the worst results on the sequence Highway III.



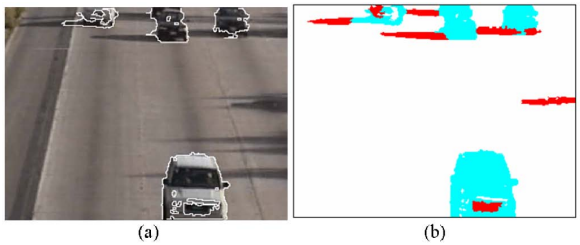
(a) (b)

Fig. 8. Best results on sequence Highway I.

Fig. 11 shows a frame in which our method performs one of its worst. Small parts of the shirt are misclassified and a tiny part of the foot is classified as shadow. This worst result also produced a very good discrimination accuracy which shows how well our method performs on the given sequence.

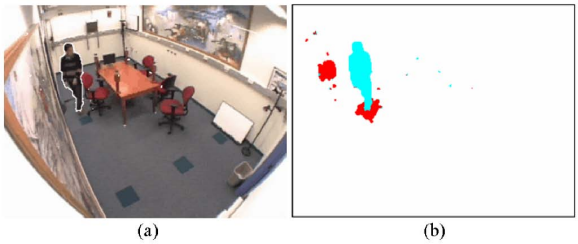
Challenges

Detecting shadows is challenging, especially when shadows are dark, which make them similar in intensity to dark colored vehicles. Also, in dark regions, edge gradient values are unreliable since they are highly quantized. Fig. 12 shows results in such a case in which the method does reasonably well. Another problem in this sequence is that vehicles are



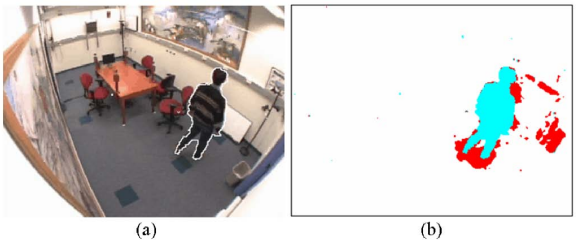
(a) (b)

Fig. 9. Worst results on sequence Highway I.



(a) (b)

Fig. 10. Best results on sequence Intelligent Room. Note the shadows cast on different planes in the background.



(a) (b)

Fig. 11. One of the worst results on the sequence Intelligent Room.

small relative to frame size. This makes our resolution lower and distinction tougher. Fig. 13 further demonstrates this. Vehicles in the image are even smaller; also the shadow of one falls on another vehicle in the frame shown. This is the problem of indirect cast shadows mentioned in [1]. Fig. 13(b) shows that both vehicles are still separated correctly. As we go farther away from the camera, resolution becomes too coarse to make any distinction possible and shadows cannot be separated.

In order to quantify results, we use shadow discrimination accuracy and detection accuracy [1]. Shadow detection accuracy (ξ) is the ratio of true positive shadow pixels detected to the sum of true positives and shadow pixels detected as foreground. Discrimination accuracy (η) is the ratio of total ground truth foreground pixels minus number of foreground pixels detected as shadow to the total ground truth foreground pixels. We manually segmented about 70 frames randomly from the sequence Highway I and about 50 frames (1 in every 5) from sequence Highway III. Results are mentioned in Table III. Table III also shows the results for Highway I with step 3 parameters tuned to Highway III, and vice versa. The respective accuracies show an expected decline; however, this decline is very small. This shows that even though step 3 requires fine tuning, performance without specific tuning also maintains good accuracy values.

(a) (b)

Fig. 12. Results on sequence Highway II. Note the similarity in the appearance of shadows and dark vehicles.

(a) (b)

Fig. 13. Results on the sequence Highway IV. Note the small size of vehicles and indirect cast shadows.

TABLE III

DETECTION AND DISCRIMINATION ACCURACY ON VARIOUS SEQUENCES

Video Sequence	η(%)	ξ(%)	Number of pixels under test
Intelligent Room	87.99	97.08	246048
Highway I	84.11	96.69	903567
Highway I - Step 3 parameters tuned to *Highway III*	83.14	96.28	903567
Highway III	88.57	93.78	394524
Highway III - Step 3 parameters tuned to *Highway I*	88.39	92.58	394524

In Table IV, we provide best results out of those reported in [1] on the Intelligent Room sequence. The ground truth for only this sequence was obtained from UCSD. We compare results on this sequence only to keep the comparison fair. From Table IV, it can be seen that our method performs better in terms of both ξ and η. An important point is that best results in [1] for ξ and η are produced by different algorithms as mentioned in Table IV. Both ξ and η are conflicting in the sense that increasing one usually decreases the other. Our method outperforms both methods in both parameters together.

Table III indicates that our proposed method does very well on different sequences with varying illumination conditions, sizes of shadow, noise levels and whether the sequence is indoor or outdoor.

V. CONCLUSION

It is important for any shadow removal technique to be independent of factors which are specific to a scene. In our approach, we use a background segmentation method which is generally applicable and global in nature. The edge, color, and intensity cues span all the common cases. Step three, which uses geometric relationships between blobs is somewhat application dependent in the sense that it needs parameter tuning for each specific video. However, once tuned, it gives a substantial improvement in performance as compared to step two. Also, for highway traffic monitoring, the principal motivation behind this work, the parameters need to be varied only in a small range. In conclusion, the method can be considered to be scene independent for all practical purposes.

TABLE IV

COMPARISON WITH THE BEST RESULTS REPORTED IN [1]

Sequence	η (%)	ξ (%)
Intelligent Room	78.61 (DNM1)	93.89 (DNM2)
Intelligent Room	87.99 (our approach)	97.08 (our approach)

ACKNOWLEDGMENT

This work has been supported in part by the National Science Foundation through grants #IIS-0219863 and #IIP-0443945, the Minnesota Department of Transportation, and the ITS Institute at the University of Minnesota.

REFERENCES

[1] A. Prati, I. Mikic, M. M. Trivedi, and R. Cucchiara, "Detecting Moving Shadows: Algorithms and Evaluation", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 7, pp. 918–923, July 2003.

[2] J. Stauder, R. Mech, and J. Ostermann, "Detecting of Moving Cast Shadows for Object Segmentation", *IEEE Trans. Multimedia*, vol. 1, no. 1, pp. 65–76, March 1999.

[3] K. Onoguchi, "Shadow Elimination Method for Moving Object Detection", *Proc. Int'l Conf. Pattern Recognition*, vol. 1, pp. 583–587, 1998.

[4] L. Elgammel, D. Harwood, and L. S. Davis, "Non-Parametric Model for Background Subtraction", *IEEE Int'l Conf. Computer Vision Frame-Rate Workshop* 1999.

[5] S. Tattersall and K. Dawson-Howe, "Adaptive Shadow Identification through Automatic Parameter Estimation in Video Sequences", *Proc. Irish Image Processing and Machine Conf.*, pp. 57–64, September 2003.

[6] T. Horprasert, D. Harwood, and L. S. Davis, "A Statistical Approach for Real-Time Robust Background Subtraction and Shadow Detection", *Proc. IEEE Int'l Conf. Computer Vision Frame-Rate Workshop*, 1999.

[7] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting Objects, Shadows, and Ghosts in Video Streams by Exploiting Color and Motion Information ", *Proc. Eleventh Int'l. Conf. Image Analysis and Processing*, pp. 360–365, September 2001.

[8] I. Mikic, P. Cosman, G. Kogut, and M. M. Trivedi, "Moving Shadow and Object Detection in Traffic Scenes", *Proc. Int'l Conf. Pattern Recognition*, vol. 1, pp. 321–324, September 2000.

[9] Y. Sonoda and T. Ogata, "Separation of Moving Objects and their Shadows, and their Application to Tracking on the Loci in the Monitoring Images ", *Proc. Fourth Int'l Conf. Signal Processing* vol. 2, pp. 1261–1264, 1998.

[10] M. Kilger, "A Shadow-Handler in a Video-Based Real-Time Traffic Monitoring System", *Proc. IEEE Workshop on Applications of Computer Vision*, pp. 11–18, 1992.

[11] C. Stauffer and W. E. L. Grimson, "Adaptive Background Mixture Models for Real-Time Tracking", *Proc. Int'l Conf. Computer Vision and Pattern Recognition*, vol. 2, 1999.

[12] S. Atev, O. Masoud, and N. Papanikolopoulos, "Practical Mixtures of Gaussians with Brightness Monitoring", *Proc. IEEE Seventh Int'l Conf. Intelligent Tranportation Systems*, pp. 423–428, October 2004.