Improving Shadow Suppression in Moving Object Detection with HSV Color Information

Rita Cucchiara, Costantino Grana, Massimo Piccardi, Andrea Prati, Stefano Sirotti

Abstract— **Video-surveillance and traffic analysis systems can be heavily improved using vision-based techniques able to extract, manage and track objects in the scene. However, problems arise due to shadows. In particular, moving shadows can affect the correct localization, measurements and detection of moving objects. This work aims to present a technique for shadow detection and suppression used in a system for moving visual object detection and tracking. The major novelty of the shadow detection technique is the analysis carried out in the HSV color space to improve the accuracy in detecting shadows. Signal processing and optic motivations of the approach proposed are described. The integration and exploitation of the shadow detection module into the system are outlined and experimental results are shown and evaluated.**

Keywords— **Shadow detection, HSV color space, background suppression, motion detection**

I. Introduction

TRAFFIC surveillance and traffic control systems are often equipped with computer vision systems capable of extracting and processing visual information of the traffic scene. The aim is to detect significant objects (e.g. vehicles, people, moving infrastructure), computing object features related with their motion and appearance (shape, color, texture, centroid, area, etc.) and eventually assessing the traffic situation on the basis of the object behaviour (trajectory, motion variation, etc.). While the final part of the process is dependent on the specific application, the initial step of (moving) objects detection and their identification must be very robust and, if possible, application-independent.

In particular, in traffic surveillance the focus of image processing task does not address the good detection of single object details but is more oriented to a robust shape detection and trajectory computation of the moving objects present in the scene. *Shadow suppression* helps to achieve these goals.

The shadow points and the object points share two important visual features: motion model and detectability. In particular, since the most common techniques for foreground object detection in dynamic scene are inter-frame difference or background suppression, all the moving points of both objects and shadows are detected at the same time. Moreover, shadow points are usually adjacent to object points and with the more commonly used segmentation techniques shadows and objects are merged in a single blob. These aspects cause two important drawbacks: the former is that the object shape is falsified by shadows and all the measured geometrical properties are affected by an error (that varies during the day and when the luminance changes). This affects both the classification and the assessment of moving object position (normally given by the shape centroid), as, for instance, in traffic control systems that must evaluate the trajectories of vehicles and people on a road.

The second problem is that the shadows of two or more objects can create a false adjacency between different objects, thus detecting them as merged in a single one. This affects many higher level surveillance tasks such as counting and classifying individual objects in the scene.

In order to avoid the drawbacks due to shadows, we have defined an approach of shadow detection and suppression based on HSV color space. We have implemented it within an image processing system, called **Sakbot** - Statistical and Knowledge-Based Object Tracker - for traffic control and surveillance purposes [1].

In this paper we present the algorithm for shadow suppression and its integration in Sakbot; it is exploited at two levels: first for improving segmentation, and second for improving background update. In the next Section we underline the shadow detection problem with some related works, in Section III we present the shadow detection technique whose major novelty is exploitation of the HSV color space. In Section IV its integration in Sakbot is described and Section V reports some experimental results. Conclusions follow.

II. Related Work

Shadows are due to the occlusion of light source by an object in the scene. In particular, that part of the object not illuminated is called *self-shadow*, while the area projected on the scene by the object is called *cast shadow* [2]. This last one is more properly called *moving cast shadow* if the object is moving.

In literature, many works have been published on shadow detection topic. Jiang and Ward [2] extract both self-shadows and cast shadows from a static image. They use a three level processes approach: the low level process extracts dark regions by thresholding input image; the middle level process detects features in dark regions, such as the vertexes and the gradient of the outline of the dark regions and uses them to further classify the region as penumbra (part of the shadow where the direct light is only partially blocked by the object), self-shadow or cast shadow; the high level process integrates these features and confirms the consistency along the light directions estimated from

Rita Cucchiara, Costantino Grana, Andrea Prati and Stefano Sirotti are with the Dipartimento di Scienze dell'Ingegneria, Universitá di Modena e Reggio Emilia, Italy

Massimo Piccardi is with Dipartimento di Ingegneria, Universitá di Ferrara, Italy

the lower levels.

Since our work addresses the problem of segmentation of moving objects, we aim to define an approach for detecting moving cast shadows on the background, without computing static shadows (due to static objects).

In [3], the authors detail the shadow handling system using signal processing theory. Thus, the appearance of a point belonging to a cast shadow can be described as:

$$s_k(x,y) = E_k(x,y)\rho_k(x,y) \tag{1}$$

where s_k is the image *luminance* of the point of coordinate (x,y) at time instant k. $E_k(x,y)$ is the *irradiance* and it is computed as follows:

$$E_k(x,y) = \begin{cases} c_A + c_P \cos\angle(\mathbf{N}(x,y),\mathbf{L}) & \textit{illuminate} \\ c_A & \textit{shadowed} \end{cases} \tag{2}$$

where c_A and c_P are the intensity of the ambient light and of the light source, respectively, $\mathbf{L}$ the direction of the light source and $\mathbf{N}$(x,y) the object surface normal. $\rho_k(x,y)$ is the *reflectance* of the object surface.

In [3], some hypotheses on the environment are outlined:
1. strong light source
2. static background (and camera)
3. planar background

Most of the papers take implicitly into account these hypotheses. In fact, typically the first step computed for shadow detection is the difference between the current frame and a reference image, that can be the previous frame, as in [3], or a reference frame, typically named *background model* [4][5][6][1].

Using eq. 1, we can write this difference $D_k(x,y)$ as:

$$D_k(x,y) = s_{k+1}(x,y) - s_k(x,y) \tag{3}$$

Let us consider that a previously illuminated point is covered by a cast shadow at frame $k+1$. According to the hypothesis 2 of a static background, reflectance $\rho_k(x,y)$ of the background does not change with time, thus we can assume that

$$\rho_{k+1}(x,y) = \rho_k(x,y) = \rho(x,y) \tag{4}$$

Then, eq. 3 can be rewritten (using eqs. 1,2 and 4) as [3]:

$$D_k(x,y) = \rho(x,y)c_P \cos\angle(\mathbf{N}(x,y),\mathbf{L}) \tag{5}$$

Thus, if hypothesis 1 holds, c_P in eq. 5 is high. Summarizing, if hypotheses 1 and 2 hold, difference in eq. 3 is high in presence of cast shadows covering a static background. This implies (as assumed in many papers) that shadow points can be obtained by thresholding the frame difference image.

Eq. 5 detects not only shadows, but also foreground points. The papers in literature mainly differ in the way they distinguish between those points.

In [4] Kilger uses a background suppression technique to find the moving objects and moving cast shadows in the scene. Then, for each object, it exploits the information on date, time and heading of the road computed by its system to choose whether to look for vertical or horizontal edges to separate shadows from objects.

In [7], a the statistical a-posteriori estimation of the pixel probabilities of membership to the class of background, foreground or shadow points. The authors use three sources of information: *local*, based on the assumption that the appearance of a shadowed pixel can be approximated using a linear transformation of the underlying pixel appearance, according with the fact that the difference of eq. 5 should be positive; *spatial*, which iterates the local computation by recomputing the a-priori probabilities using the a-posteriori probabilities of the neighborhood; *temporal*, which predicts the position of shadows and objects from previous frames, therefore adapting the a-priori probabilities.

The approach in [3] exploits the local appearance change due to shadow by computing the ratio $R_k(x,y)$ between the appearance of the pixel in the actual frame and the appearance in a reference frame:

$$R_k(x,y) = \frac{s_{k+1}(x,y)}{s_k(x,y)} \tag{6}$$

that can be rewritten as ratio between irradiance and reflectance by using eqs. 1 and 4 :

$$R_k(x,y) = \frac{E_{k+1}(x,y)}{E_k(x,y)} \tag{7}$$

If a static background point is covered by a shadow, we have:

$$R_k(x,y) = \frac{c_A}{c_A + c_P \cos\angle(\mathbf{N}(x,y),\mathbf{L})} \tag{8}$$

This ratio is less than one. In fact, the angle between $\mathbf{N}(\mathbf{x},\mathbf{y})$ and $\mathbf{L}$ is in the range between $\frac{-\pi}{2}$ and $\frac{\pi}{2}$, therefore the *cos* function is always positive.

Moreover, due to hypothesis 3, we can assume $\mathbf{N}(\mathbf{x},\mathbf{y})$ as spatially constant in a neighborhood of the point, because the background is supposed planar in a neighborhood.

In [3], authors exploit the spatial constancy of $\mathbf{N}$ to detect shadows by computing the variance in a neighborhood of the pixel of the ratio $R_k(x,y)$: a low variance means that assumption 3 holds, then they mark that pixel as "possible shadow". Moreover, authors use a lot of other techniques in order to exploit all the four assumptions (such as edge detection and gradient calculation).

Eq. 8 can be seen as the ratio between the luminance after and before shadow appears. In a similar way, Davis et al. ([5][8]) define a local assumption on the ratio between shadow and shadowed point luminance. This is based on the hypothesis that shadows darken the covered point,as eq. 8 and the considerations above confirm. This approach has been improved in [6] where the authors state that shadow has similar chromaticity but lower brightness than that of the same pixel in the background image. They base this statement on the notion of the shadow as a semi-transparent region in the image, which retains a representation of the underlying surface pattern, texture or color value. They work in the RGB space; we exploit a similar concept working in the HSV color space.

III. Shadow detection

The shadow detection algorithm we have defined in Sakbot aims to prevent moving cast shadows being misclassified as moving objects (or parts of them), thus reducing the undersegmentation problem and improving background update. In Fig. 1 an undersegmentation example is reported. Fig. 2 shows how shadow suppression allows the correct identification of all the objects in the scene.

Fig. 1. Object detection w/o shadow suppression

Fig. 2. Object detection with shadow suppression

The major problem is how to distinguish between moving cast shadows and moving object points. In fact, also points belonging to a moving object are detected by background suppression because the difference $D_k(x,y)$ computed on them is greater than a threshold and the ratio $R_k(x,y)$ between their luminance and the luminance of the corresponding reference image point could have a value similar to that of the ratio between shadows and the reference image.

To solve this problem, Sakbot uses the Hue-Saturation-Value (HSV) color space. The HSV color space corresponds closely to the human perception of color [9] and it has proven more accurate in distinguish shadows than the RGB space. Thus, Sakbot tries to estimate how the occlusion due to shadow changes the value of H, S and V.

We analyze only points belonging to possible moving objects, i.e. that are detected with a high difference according to eq. 3. Then, according with assumption of Section II, the ratio in eq. 6 must be less than one. In fact, a cast shadow point darkens the background point, whereas an object point could darken it or not, depending on the object color texture. We approximate in eq. 6 the luminance $s_k(x,y)$ with $I_k^V(x,y)$, where $I_k^V(x,y)$ is the intensity value for the component V of the HSV pixel at coordinates (x,y) in the frame k (neglecting strong camera noise, usually a good approximation).

Thus, we define a shadow mask SP_k for each (x,y) points (previously detected as moving one) with three conditions as follows:

$$SP_k(x,y) = \begin{cases} 1 & if \quad \alpha \leq \frac{I_k^V(x,y)}{B_K^V(x,y)} \leq \beta \\ & \wedge \, (I_k^S(x,y) - B_k^S(x,y)) \leq \tau_S \\ & \wedge \, |I_k^H(x,y) - B_k^H(x,y)| \leq \tau_H \\ 0 & otherwise \end{cases} \tag{9}$$

The first condition works on the luminance (the V component). The use of β (less than one) allows us to avoid identification of the points where the background was slightly changed by noise as shadows, whereas α takes into account how strong the light source is, i.e. accounts for the c_P, the c_A and the angle defined in the ratio in eq. 8. Thus, stronger and higher the sun (in outdoor scenes), the lower will be that ratio, and lower value of α must be chosen. In fact, c_P will raise and the cos function will tend to 1.

On component S a threshold on the difference is performed. Shadows lower saturation of points and, according with many experimental tests, the difference in saturation between image and reference is usually negative for shadow points. On the component H a threshold on the absolute difference turns out better results. However, the choice of the parameters τ_H and τ_S is less straightforward and, for now, is done empirically with the assumption that the chrominance of shadowed and non-shadowed points even if could vary, does not vary too much.

Fig. 3. Shadow detection with luminance only

The color information improves the discrimination between object and shadow drastically. In Fig. 3 an example of the effect of the three conditions are reported: black pixels are those classified as belonging to the background model, dark gray pixels those classified as foreground, light gray ones would be identified as shadows by means of only the luminance information, while white pixels are shadow points detecting using also the chrominance information. Removing light gray pixels from the shadow mask improves the accuracy by avoiding the misclassification as shadow of pixel belonging to the car. In particular in the lest vehicle in Fig. 3 a large part of the car will be removed as shadow if only luminance is considered; this do not happens with the mask of eq. 9, having the car a different color than the background.

IV. Shadow suppression in Sakbot

Sakbot is the system we have developed for moving object detection and tracking; it is currently tested for video-surveillance applications and for a possible inclusion in a traffic-light control systems[1]. The Sakbot acronym derives from the model we use for background update. It is based on the concurrent use of results of a *statistical* process on previously sampled frames and the *knowledge-based* feedback from previously detected moving objects. For detail on the moving object detection in Sakbot , refer to [1].

The basic steps are:

1. background suppression
2. moving blob detection
3. moving object identification
4. background update.

Background suppression is done by thresholding a difference as in eq. 3, but working on the three color components and not only in the luminance component. The extracted foreground points are candidate moving points and after some morphology operations (opening and closing) are grouped into blobs. Due to shadow objects visually easy to be disjoint are merged into a single blob. On all blob points the shadow mask defined in 9 is applied. Shadow points are not discarded but are grouped into blobs classified as *moving shadow* (MS), while the remaining are grouped into blobs again. All blobs are then processed by image analysis tasks able to extract some features, such as the area, centroid and external extent, and a motion value, in particular the *average optical flow* [10].

If the area is large enough (according with the scene) and the motion measurement is sufficiently high, the blob is classified as a real *moving visual object* (MVO). Not all moving points belong to MSs or MVOs: some blob could have a low average optical flow and thus resulting not in motion. This is due to possible errors in background reference images causing false positive (sometime called *ghost*). Consider for instance that a stopped car, thus included in the background, starts its motion. In the current frame the car will be detected as a MVO, but also the ghost of the car (the points where the car was stopped) will be detected, but not classified as a MVO. For the car's shadow the same reasoning can be done.

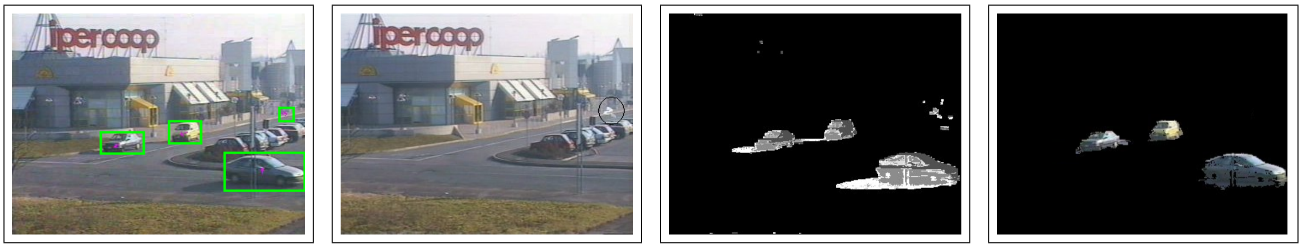
[1]The Sakbot project is partially granted by the "Finanziamento dipartimento aree urbane del Min LL.PP: - progetto supervisore della citta' di Bologna".

Then, the most peculiar Sakbot process must be provided: the background update. In [1] we detailed as the new background is computed by means of a temporal median function of previously sampled frames in a finite time window. This step is corrected by an adaptiveness factor, i.e. the previously computed background is added in the median function, with an adequate weight. However, not all the image points are processed, but only those points that do not belong neither to MVOs nor to MSs. The use of a temporal median function with adaptiveness assure a good responsiveness to changes in a limited observation windows (few samples are used with a fast execution); the selectivity based on the knowledge of real moving objects and shape prevents from false positives, since it allows not to include in the background the points of moving object [1]. This is similar to what is called *selective update* in [5], although in that work, single moving points are used without the further validation to belong to moving objects.

In Fig. 4 an example of urban traffic scenes is reported. Fig. 4(b) shows the background model computed with Sakbot: the object on the left (enclosed in an ellipse) is a typical example of erroneous background due to ghost. Even if Sakbot detects it, it is not included in the MVOs (Fig. 4(d)) because its average optical flow is near to zero. Thus, the system works properly and the ghost is rapidly removed from the background model.

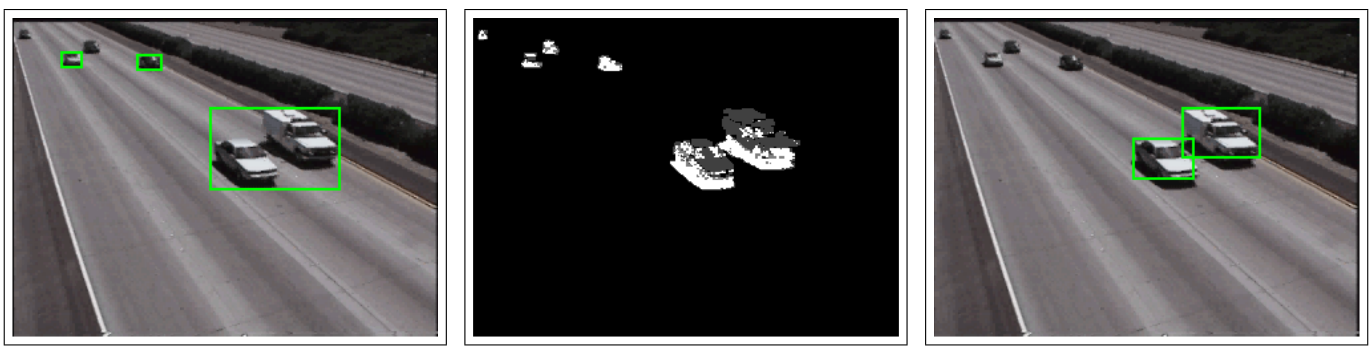
V. Experimental Results

Sakbot is currently tested in different environment, in scenes of Italian and USA highways, urban traffic at intersection and other outdoor and indoor scenes with people. In Fig. 5 two results of the shadow detector of Sakbot are presented; the former example derives from tests executed at University of San Diego, while the latter has evaluated at University of Modena. Even if the two sequences consider totally different environment, light condition and object classes, the shadows are correctly removed (even in Fig. 5(d) where the shadow is difficult to be seen by human eyes). However, also small objects are removed from the scene. This is due to the fact that they are dark cars (thus detected partially as shadow, Fig. 5(b)) or too small to be detected after shadow suppression. However, they are immediately detected as they grow in size or by changing the threshold on the minimum area. The use of chrominance information reduces the number of candidate shadow points classified as shadow. In the graph of Fig. 6 we report the amount of pixels classified as shadow and classified as object due to V,H and S conditions. These tests refer to a video of 1000 frames as the one reported in Fig. 2. On average the shadow detection allows the system to classify the 26.88% of points detected as moving points as shadow points. Without the consideration on the H and S components, i.e. using luminance only, the 43% of points would be classified as shadow, resulting in a drastic erosion of the real object shape. These numbers indicate the percentage of points detected as shadow w.r.t. object points. A more precise measurement of shadow detection *accuracy* accounts for good detection and good discrimi-



(a) Input image (b) S&KB background (c) Shadow detection (d) MVOs extracted

Fig. 4. Examples of Sakbot system in urban traffic scene



(a) Detection w/o shadow suppression (b) Shadow detection (c) Detection with shadow suppression

(d) Detection w/o shadow suppression (e) Shadow detection (f) Detection with shadow suppression

Fig. 5. Examples of shadow detection

nation with respect to a reference optimum segmentation. The process, in fact, could provide some *false negatives*, i.e. the shadow points classified as background/foreground, and some *false positives*, i.e. the foreground/background points detected as shadows. In Table I, the system performance is detailed by showing these measurements with different parameter values. This evaluation has been done by manually segmenting some frames of a video sequence in order to identify shadows, foreground and background regions. In the test set different situations have been considered (dark/ligth cars, multiple cars or single car, occlusions or not). In Table I, the first row reports the results of shadow detection by using only the information on the luminance, while the other rows report results by changing independently the four parameters. The first two columns are the average total number of pixels detected as object and shadow, respectively. The false positives and false negatives (in percentage w.r.t. the area of the MVO) are also reported in the last two columns. Without chrominance many false positives arise. The second, third and fourth rows differ from an increasing α value: if it grows, as it is believable, the FP decreases while the FN number increases. Same consideration can be done, varying the β parameter. *In media stat virtus* and therefore parameters adopted for α and β are 0.4 and 0.6, respectively. The last four rows show a similar behavior achieved by varying the chrominance threshold.

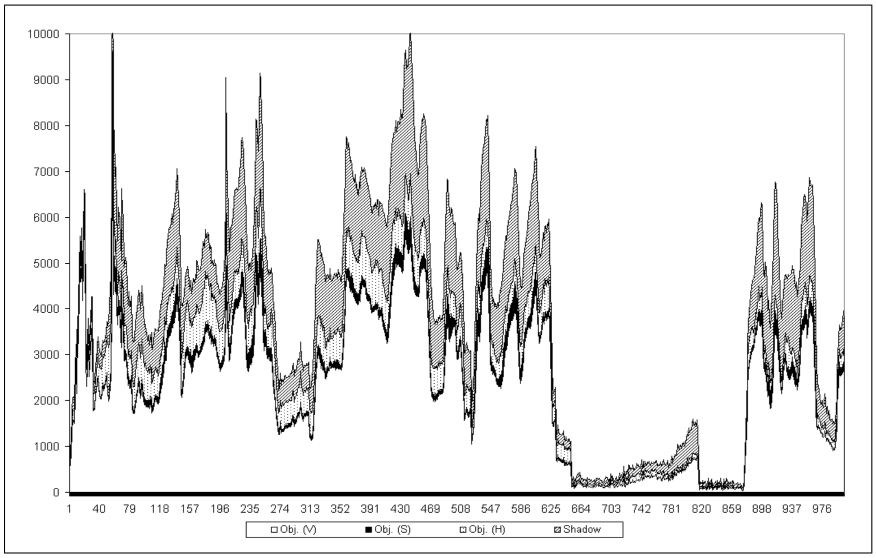


Fig. 6. Shadow and object points

α	β	$\tau_{\mathbf{H}}$	$\tau_{\mathbf{S}}$	**Detected Object points**	**Detected Shadow points**	**FP%**	**FN%**
0,4	0,60	N/A	N/A	5210	3413	14,37%	9,62%
0,3	0,60	0,5	0,1	5657	3676	10,53%	5,96%
0,4	**0,60**	**0,5**	**0,1**	**6130**	**3156**	**7,30%**	**11,09%**
0,5	0,60	0,5	0,1	6597	1102	3,62%	26,28%
0,4	0,50	0,5	0,1	6528	1971	3,58%	14.35%
0,4	0,90	0,5	0,1	5230	4098	20.22%	7,00%
0,4	0,60	0,9	0,1	5565	3306	11.43%	9.79%
0,4	0,60	0,1	0,1	6841	921	1,80%	26,64%
0,4	0,60	0,5	0,5	5838	3273	9,24%	10,35%
0,4	0,60	0,5	0,0	6583	1755	3,46%	16.73%

TABLE I
EXPERIMENTAL MEASUREMENTS OF ACCURACY

VI. Conclusions

In the paper a robust shadow detection approach based on HSV color space has been presented and discussed. We proved that shadow detection and suppression improve object segmentation and object feature computation, that are very critical tasks in video-surveillance and vision-based traffic control systems. Moreover, we proved that the adoption of chrominance improves shadow detection considerably. Finally, analyzing only candidate moving points allows us to detect real moving cast shadow, distinguishing them from apparent shadow blobs (due to errors in the reference image). Static shadows due to static objects are intentionally excluded from detection: points belonging to static shadows are, instead, included in the background reference image and their changes due to luminance condition variations and the daytime are taken into account in the knowledge based background update.

References

[1] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Statistical and knowledge-based moving object detection in traffic scene," in *Proceedings of IEEE Int'l Conference on Intelligent Transportation Systems*, Oct. 2000, pp. 27–32.

[2] C. Jiang and M.O. Ward, "Shadow identification," *Proceedings of IEEE Int'l Conference on Computer Vision and Pattern Recognition*, pp. 606–612, 1992.

[3] J. Stauder, R. Mech, and J. Ostermann, "Detection of moving cast shadows for object segmentation," *IEEE Transactions on Multimedia*, vol. 1, no. 1, pp. 65–76, Mar. 1999.

[4] M. Kilger, "A shadow handler in a video-based real-time traffic monitoring system," *Proceedings of IEEE Workshop on Applications of Computer Vision*, pp. 11–18, 1992.

[5] A. Elgammal, D. Harwood, and L.S. Davis, "Non-parametric model for background subtraction," in *Proceedings of IEEE ICCV'99 FRAME-RATE Workshop*, 1999.

[6] T. Horprasert, D. Harwood, and L.S. Davis, "A statistical approach for real-time robust background subtraction and shadow detection," in *Proceedings of IEEE ICCV'99 FRAME-RATE Workshop*, 1999.

[7] I. Mikic, P. Cosman, G. Kogut, and M.M. Trivedi, "Moving shadow and object detection in traffic scenes," in *Proceedings of Int'l Conference on Pattern Recognition*, Sept. 2000.

[8] I. Haritaoglu, D. Harwood, and L.S. Davis, "W4: Real-time surveillance of people and their activities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 809–830, Aug. 2000.

[9] N. Herodotou, K.N. Plataniotis, and A.N. Venetsanopoulos, "A color segmentation scheme for object-based video coding," in *Proceedings of the IEEE Symposium on Advances in Digital Filtering and Signal Processing*, 1998, pp. 25–29.

[10] B.K.P. Horn and B.G. Schunck, "Determing optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.