A Novel Approach to Shadow Detection in Video-based Virtual Reality Interaction

Haipeng Cai
State Key Lab of CAD&CG
Zhejiang University
Hangzhou, 310027, P.R.C
chapering@gmail.com

Zhigeng Pan
State Key Lab of CAD&CG
Zhejiang University
Hangzhou, 310027, P.R.C
zhigengpan@gmail.com

Jiali Yao
State Key Lab of CAD&CG
Zhejiang University
Hangzhou, 310027, P.R.C
yjl424@hotmail.com

ABSTRACT
When detecting moving shadow for video-based virtual reality interaction (VBVRI), we need not concern the original moving object that casts the shadow. It is based on this feature of application that a novel algorithm of moving shadow detection is proposed in this paper, which primarily introduces a two-step shadow discriminator and an improvement upon the classical Gaussian Mixture Model (GMM). The new algorithm markedly enhances the real-time performance of shadow detection opposed to that of GMM, and exhibits an outstanding resistance against the disturbance that arises from abrupt lighting changes.

Categories and Subject Descriptors
I.4.8 [**Image Processing and Computer Vision**]: Scene Analysis – *motion, tracking, color.*

General Terms
Algorithms, Performance, Design, Reliability, Experimentation, Human Factors, Verification.

Keywords
Keywords are your own d Shadow detection, Virtual reality interaction, Shadow discriminator, GMM.

1. INTRODUCTION
The aim of either moving targets detection in video streams or motion-based segmentation of video objects is to extract the moving objects from the original motion images, as is the fundamental and crucial part in the motion tracking or the recognition and hence analysis of behavior of the moving targets. The shadow arising from scene lighting and occlusion between objects, however, often impedes the detection, and the shadow detection therefore is always regarded by researchers as the key step within the overall framework of motion detection algorithms.

For the purpose of detecting, suppressing and further eliminate the moving shadows in video images so as to enhance the quality of motion segmentation, many relevant original algorithms and improvements upon them have been proposed. According to the classification in terms of introduction of decision process and reference to uncertainty by Andrea Prati et al.[2], most of these proposed approaches can be categorized into two main classes: statistical approaches, which describe the pixel's class membership by way of probabilistic function, and deterministic approaches, which indicates the membership simply by an on/off switch. In their sub-classification, statistical methods are further divided into the parametric and the non-parametric, and deterministic methods the model based and the non-model based. Friedman and Russel present a model based parametric approach in [3] through which parameters are learned by certain samples and each pixel is thus modeled using Gaussian mixture, with the parameters being updated by use of the membership probability of the pixel. This approach is also exploited for detecting shadows in traffic video streams by Mikic et al. in [4]. Concentrating on shadows in static scenes, T. Horprasert et al.[5] put into use an non-parametric approach, utilizing the brightness distortion and chromaticity distortion to construct a computational color model, to implement background subtraction and shadow extraction. Another method is employed in [6], which makes use of the height information of moving objects against the road plane to remove the shadow areas in the road scene under the assistance of two cameras, with the image inversely projected by one camera to the road plane being transformed to the view from the other, and this method belongs to the model based deterministic category[1]. Besides, recently more and more approaches solve shadow detection and elimination by use of the chromatic feature of shadow opposed to the corresponding background. Xiao et al.[7] use Canny operator to extract the edge of real moving foreground objects so as to remove shadow implicitly. In [8], the HSV color information of pixel is exploited to improve the shadow suppression. As an example of yet another approach, the method presented in [9] achieves shadow detection by utilizing the similarity between little textured patches.

In the VBVRI control, shadow detection serves not to reduce or eliminate the side effects of shadows but rather to help locate the shadow-projecting objects and thus provides critical information like positions and paths of those interactive objects for the interaction control. Since only the shadows rather than the interactive objects themselves lie within the video frame, this

ISBN: 978-1-902560-21-2 © 2008 GDTW

information can be acquired indirectly by shadow detection. In [12], a type of VBVRI application offering CSCW features is discussed from an engineering point of view; it is actually implemented by use of shadow detection. In a sense, the fact that the motion images include not the projecting objects but merely the shadows they cast make it reasonable to view the shadow detection simply as special motion detection. The straightforward exploitation of motion detection algorithms, such as the most popular GMM approach, however, proves invalid to serve the shadow detection. Firstly, per-pixel modeling and hence pixel-level model updating of GMM both entails expensive computation, and then greatly undermines the real-time performance of the detection. On the other hand, GMM is fairly weak in adapting to lighting changes, especially abrupt changes, which is rightly awfully pervasive in the VBVRI applications.

In this paper, we extend motion detection under the background subtraction framework by introducing a two-step shadow discriminator and improving the model initialization and adaptation in order to enhance the overall real-time performance and robustness of shadow detection. On the basis of this extension and improvement, we propose a new shadow detection algorithm for VBVRI application (called MSDVRI), and our experimental results substantiate the effectiveness and efficiency of the novel algorithm.

2. THE BASIC GMM AND SHADOW DISCRIMINATOR

2.1 GMM Approach

The basic GMM method[10] describes the RGB value (r_i, g_i, b_i) of pixel $x_t = (x_i, y_i)^t$ within the video frame I as a random variable $X = (x_r, x_g, x_b)$ using a probability mixture consisting of K Gaussian components, where K is closely related to spacial and temporal complexity of the algorithm based upon it and is usually within [3,5], and t indicates the time variance of the video sequence. Letting $\mu_{i,t}$, $\sum_{i,t}$, $\eta(X_t, \mu_{i,t}, \sum_{i,t})$ symbolize the mean, co-variance matrix and probability density function (PDF) of the i^{th} component at time t respectively, the K-Gaussian Mixture of X_t is formulated as

$$P(X_t) = \sum_{i=1}^{K} \omega_{i,t} * \eta(X_t, \mu_{i,t}, \sum_{i,t}) \qquad ()$$

where $\omega_{i,t}$ is the component weight, and the PDF of the i^{th} component is defined as

$$\eta(X_t, \mu_{i,t}, \sum_{i,t}) = \frac{1}{(2\pi)^{n/2} \left|\sum_{i,t}\right|^{1/2}} e^{-\frac{1}{2}(X_t - \mu_{i,t})^T (\sum_{i,t})^{-1} (X_t - \mu_{i,t})} \qquad ()$$
$$i = 1, 2, ..., k$$

with n being the dimension of vector X_t .Under the assumption of independency between the three chromatic channels, the 3D normal random variable $X_t = (x_{r,t}, x_{g,t}, x_{b,t})$ can be regarded as the union of three independent variables of same distribution, therefore the co-variance matrix will be simplified as $\sum_{i,t} = \delta_{i,t}{}^2 I$, $i = 1, 2, ..., k$. During the detection process, model match is judged by whether the difference between pixel's value and component's mean is no more than the standard variance of the same component times 2.5. The model adaptation is defined, introducing $\alpha (0 \le \alpha \le 1)$ as user-defined model parameter learning rate and $\rho = \alpha \eta(x_t \mid \mu_{i,t}, \sigma_{i,t})$ as the parametric learning rate, for the i^{th} component using following equations

$$\omega_{i,t} = (1-\alpha)\omega_{i,t-1} + \alpha M_{i,t} \qquad ()$$

$$\mu_{i,t} = (1-\rho)\mu_{i,t-1} + \rho x_t \qquad ()$$

$$\sigma_{i,t}{}^2 = (1-\rho)\sigma_{i,t-1}{}^2 + \rho (x_t - \mu_{i,t})^T (x_t - \mu_{i,t}) \qquad ()$$

wherein the match tag $M_{i,t}$ is evaluated as 1 when the corresponding component matches x_t and 0 otherwise. In the model updating, components not matching x_t will hold their means and variances unchanged with their weights being decreased to some extent. For those pixels that match none of their components, we need to resume their components with the least weight to hold a larger variance, lower weight and same mean opposed to the latest updated mixture, and leave all other components unchanged in terms of $\mu_{i,t}$ and $\sigma_{i,t}$, while updating their weights as $\omega_{i,t} = (1-\alpha)\omega_{i,t-1}$. Then the normalized $\omega_{i,t}$ s will be used to sort all components of each pixel by descending order in terms of the value of $\omega_{i,t} / \sigma_{i,t}$. After B predominant components, as together signify the background model, are selected from the ordered mixture according to $B = \arg\min_b (\sum_{k=1}^{b} w_k > T_{sw})$, whether the pixel belongs to the background or foreground can be determined simply by judging the math tag using components of the background model: if at least one of these components matches x_t , the pixel will be categorized into background, otherwise it is viewed as part of foreground. In this equation, T_{sw} is the weight sum threshold that is supposed to be evaluated as a constant.

2.2 Shadow Discriminator

The Gaussian background model adopts a per-pixel modeling and model updating, and thus incurs a relatively high computational complexity. While detecting moving shadows in terms of their motions using GMM, we can utilize the appreciable chromatic difference between pixels in background and that in foreground to preliminarily determine pixels that are possibly within the shadow areas, and merely take those possible shadow pixels as the input of later motion detection algorithmic procedure. This approach will not only enhance the real-time performance of the detection, and also conduces to overcoming disturbances stemming from either the abrupt lighting changes or disruptive repetitive motions in the background.

In the RGB color space, when the value of a pixel is changed with the ratio *R:G:B* being left unchanged, the pixel's brightness will deviate but its chrominance will hold. Usually there is a definite ratio between shadow pixels and pixels of background, which is formulated as[12]

$$\begin{aligned} Gray_s &= \alpha R_s + \beta G_s + \gamma B_s \\ &= \alpha k R_b + \beta k G_b + \gamma k B_b = k Gray_b \end{aligned} \qquad ()$$

wherein $R_s, G_s, B_s, Gray_s$ indicates the channel value and gray of shadow pixel and $R_b, G_b, B_b, Gray_b$ is the value and gray of the corresponding background pixel. In addition, the candidate shadow pixel has also smaller brightness than the related background pixel in each channel. In accordance with this quantitative relation, we may precede the shadow motion detection an initial filter to exclude pixels that are unlikely to be within shadow areas. Symbolizing the three channel values of background pixel corresponding to x_t with b_r, b_g, b_b, and the gray range with T_s, T_l respectively, the formula representation of this filter is as follows

$$SP_1(x_t)=\begin{cases}1, & (x_r<b_r \wedge x_g<b_g \wedge x_b<b_b) \\ & \wedge (T_s \le Gray(x_t) \le T_l) \\ 0, & otherwise\end{cases} \quad ()$$

After shadow pixel filtering with SP_1, there will still left lots of specious shadow pixels that are in practice not in shadow areas, especially for the occasion under which the detected scene contains various non-shadow regions of lower gray, or the scene is itself of lower gray due to weather conditions like night and cloudy days. Therefore, we can go further to adopt shadow sifting to save computation to a higher measure. Considering the fact that a pixel's brightness merely changes little when it is cast by certain shadow, although its saturation might decrease appreciably, we refer to the conclusion presented in [13,2] about pixel's HSV color characteristics to introduce the second step of the proposed shadow discriminator as

$$SP_2(x_t)=\begin{cases}1, & (\alpha \le \frac{I^V(xt)}{B^V(xt)} \le \beta)\wedge \\ & (I^S(xt)-B^S(xt) \le TS)\wedge \\ & (\left|I^H(xt)-B^H(xt)\right| \le TH) \\ 0, & otherwise\end{cases} \quad ()$$

where $B^H(x_t), B^S(x_t), B^V(x_t)$ and $I^H(x_t), I^S(x_t), I^V(x_t)$ are the HSV channel values of x_t in the background and that of corresponding pixel assumed to be in shadow areas respectively. Additionally, α should be evaluated with proportion to the intensity of solar light, β is introduced to help avoid classifying the background pixels that are impacted by noises falsely as shadow pixels, and thresholds T_S, T_H can be selected and adjusted according to practical needs, and especially, T_S should never be of positive values.

3. THE MSDVRI ALGORITHM

On the basis of motion detection by way of GMM background modeling and real-time performance enhancement through the two-step shadow discriminator to cut away redundant computation, the following presents the overall algorithmic framework of the novel MSDVRI approach.

3.1 Model Initialization Through Parameters Learning

In order the model parameters to correctly describe the dynamic background model; we adopt the learning policy by samples training of the basic GMM algorithms to initialize all components in the Gaussian mixture for each pixel. Firstly all components can be initialized on the part of their means with the RGB channel values of each pixel in the first sample frame, then for the later incoming samples we just perform the same procedure as that for the formal shadow detection, i.e. the model match and adaptation. Letting the sample volume is *V*, the algorithm will stay in the learning period until all pixels successfully match the background model for exactly *V-1* frames on end, and then the overall background model is regarded as having been initialized steadily and correctly. Otherwise, the learning procedure should be resumed.

3.2 Match The Background Model

In the GMM implementation described in [11,15], when the difference between current values of a pixel and the mean of one of its components is no more than *D* times of the variance of the component, the component is viewed as a match for the pixel. In our approach, all computations concerning variance are omitted in order to simplify the model match and other parts of the algorithm as far as possible. Introducing a constant threshold T_σ, the model match tag $M_{i,t}$ can be defined as

$$M_{i,t}=\begin{cases}1, & \left|x_t-\mu_{i,t}\right| \le T_\sigma \\ 0, & otherwise\end{cases} \quad ()$$

Where $M_{i,t}$ will equal 1 only if x_t matches the i^{th} component, and T_σ needs to be adjusted in experiment and application by reference to an initial real variance acquired through certain number of samples. Since we no longer concern model variances, the components sorting will also be eliminated. For the shadow pixel judgment, we instead adopt a more simplified approach, in which

we first, by formula-(), figure out all the matched components, and then sum up the weights of these components, and shadow pixel judgment will then be made in terms of whether the weight sum surpasses a constant threshold that is also empirically evaluated.

$$SJ(x_t)=\begin{cases}1, & \sum_{i\in\{k \mid M_{k,t}=1\}} \omega_{i,t} > T_{sw} \\ 0, & otherwise\end{cases} \quad ()$$

3.3 Model Adaptation

Model updating should be performed according to model match tag for each pixel. Meanwhile, for the variance-related computation has been dismissed, model adaptation will not concern any variances. If the i^{th} component matches pixel x_t, its mean $\mu_{i,t}$ will be updated by formula-(), or the mean keeps unchanged. As to weight updating, if there exists component matching x_t, all weights of the pixel's components will be updated by formula-(), otherwise we need find the component with the least weight and re-initialize it, assigning to it a lesser weight and current color value as new mean, and for other components updating is in accordance with equation $\omega_{i,t}=(1-\alpha)\omega_{i,t-1}$. In addition, we calculate the parametric learning rate involved in the updating in a simplified way as $\rho_{i,t}=\alpha/\omega_{i,t}$ instead.

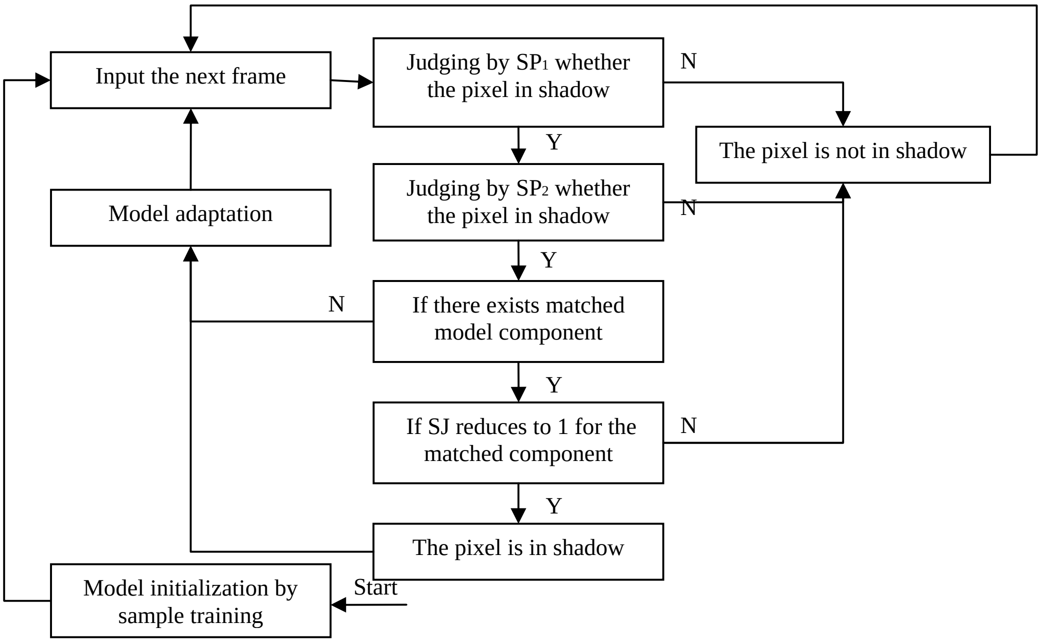


Figure 1. The overall algorithmic flow of MSDVRI

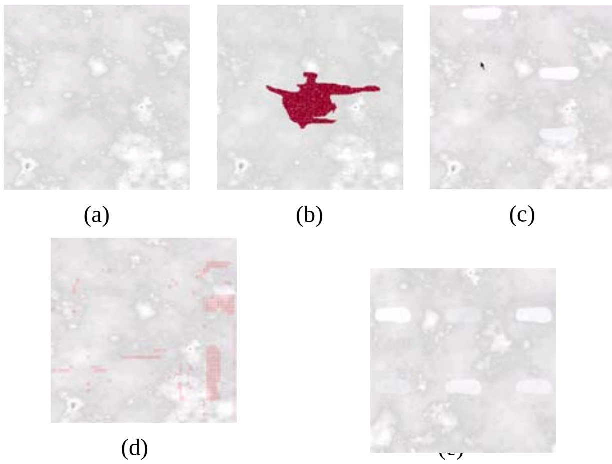
3.4 Overall Algorithmic Flow

With the improved GMM algorithm being as overall framework, and the shadow discriminator preceding the model match and parameter updating, foregoing discussions lead to the novel MSDVRI procedure. The parameter learning policy through a group of training samples will be employed to steadily initialize the background model in the first place. Then, by figuring out the two-step shadow filter upon each pixel, we only input those pixels that are preliminarily considered being in shadow area into the improved GMM motion detection flow and thus finally determine whether the pixel belongs to shadow area or not. Figure-1 shows the detailed flow of the integral MSDVRI algorithm.

4. EXPERIMENTAL RESULTS

The proposed novel approach is tested on a Pentium®4 2.4GHz processor with 512MB DDR, with all the video frames being read in a real-time manner from a video capture card connected to a camera, which snaps motion images of a predefined interactive region on the ground. The algorithm is programmed on VC 7.0 under the assistance of SDK provided along with the video card, and it handles with 352 × 288 frames at rate of 25fps. Main variables and thresholds used in our algorithm are evaluated as follows: V = 100, K = 3, α = 0.005, T_{σ} =10, T_{sw} = 0.8. Among the following results, the comparisons between GMM and MSDVRI, in terms of the motion detection and interaction control based on it, corroborate the essentiality of introducing the shadow discriminator and simplifying certain steps in the original GMM procedures.

Figure 2 displays a group of results of shadow detection and the VBVRI effects implemented using information it provides. 2(a) is the static snow background without any lighting and interactive



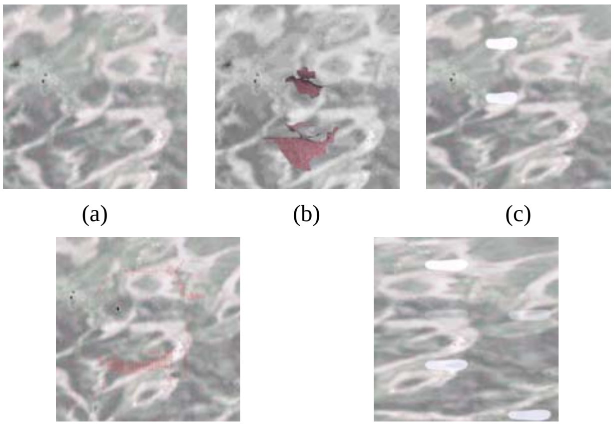
objects, and in 3(b) the light is switched on and a single person is coming into the interactive region that is snapped by the camera, this picture shows the shadow cast by the person. 2(c) is one of the frame sequence in which the virtual interactive footprints have been added. 2(d) and 2(e) illustrate the flaws of GMM in detection error due to jerky lighting changes and in detection

Figure 2. Shadow detection and interactive control based on it on the snow ground ((a) original snow background; (b) shadow detected of a single interacting person; (c) virtual footprint produced based on detected shadow; (d) detection error of GMM arising from abrupt lighting changes; (e) detection delay of GMM arising) .

Figure 3. Shadow detection and the virtual interaction based on it on the undulating water surface ((a) original water background; (b) shadow detected of two interacting persons; (c) the virtual interactive effects produced based on detected shadow; (d) detection error of GMM arising from lighting changes; (e) detection delay of GMM for the poor real-time

Table 1 contrasts of error and delay of shadow detection between GMM and MSDVRI

Algorithms	Number of shadow points	Average number of error shadow points			Average miss detection rate	Average delay(ms)
		Δt =2s	Δt =5s	Δt =10s		
GMM	1000	34	95	35	5.5%	1526
	4000	262	271	279	6.7%	1337
	8000	973	994	964	12.2%	1719
	Overall average				**8.13%**	**1527**
MSDVRI	1000	29	31	27	2.9%	203
	4000	37	49	62	1.2%	223
	8000	88	97	102	1.1%	359
	Overall average				**1.73%**	**261**



(a) (b) (c)

(d) (e)

delay because of the poorer real-time performance, compared to MSDVRI, respectively. In terms of the detection delay, when using GMM, it takes on average 1.5s to detect a moving interactive object, while using MSDVRI, the time is merely about 0.25s on average. The group in figure 3 illustrates a set of similar results with only a different background, an undulating sea water surface.

As to the further detailed contrast of effectiveness and real-time performance between GMM and MSDVRI, table 1 lists a set of quantitative test results concerning the capability of resisting lighting disturbance and the temporal efficiency of the whole shadow detection of the two algorithms. In order to obtain test data, we artificially add periodical blocks of shadow points to the original frames before they enter into the detection procedure, and designate their positions and moving paths as well, so that we have a definitely correct control to analyze and evaluate our novel approach. For the contrast of real-time performance between the two algorithms, a timer is employed in the test program to assess the time expense from the entrance of the interactive objects into the projecting area to the completion of virtual effects rendering. The variable Δt denoted in the table indicates time intervals between two times of additions of artificial shadow points. Compared to the basic GMM, according to data in this table, it can be figured out that the MSDVRI algorithm proposed decreases the average miss rate of detection by 78.72%, and lowers the average detection delay by as much as 82.9%.

In addition, certain computational simplification adopted in the discussion above, the omission of component's variance in the shadow judgment, say, should be carefully considered for the purpose of some practical uses. These simplifications help reduce considerably the overall algorithmic complexity without salient losses of effectiveness and reliability though, raise novel problems as well, such as the difficulty in choosing proper value of T_σ for the highly dynamic video scene, which, under this occasion, might cause appreciable loss of algorithmic reliability. For these cases, the original model initialization and adaptation can be employed: an initial variance for each pixel to initialize each component of its Gaussian mixture can be calculated through real computation upon a set of sample frames, an then the variance is to be updated during the model adaptation, and the match judgment will also refer to the on-line variance. Yet another consideration will involve the size of the Gaussian mixture. In our experiments and application, 3-Gaussian mixture is enough for modeling the video scene background. It is reasonable to use as less as possible components in practice, since the larger the mixture, the higher the computational complexity involved, and thus the poorer the overall real-time performance of the algorithm based on this mixture. For those video scenes of complex background, however, a relatively larger mixture will be required to model the background more correctly and robustly.

5. CONCLUSION

The approach to shadow detection within the integral framework of modern motion detection algorithm can illuminate what shall be exploited and improved about related algorithm to overcome the vital shortcomings of some classical methods, such as the classical GMM, and thus better serve applications like that of the VBVRI. We propose a novel method of detecting shadow specific for these applications, in which the improved global GMM procedure, mainly the computational simplification on the part of background modeling and model adaptation, and the effective two-step shadow discriminator combine to enhance the resistance against detrimental lighting changes and real-time performance of the shadow detection. Our experimental results and test data analysis adequately substantiate that the novel MSDVRI approach can effectively facilitate the reliable and efficient implementation of the VBVRI applications, and it will reasonably provide strong support for other similar applications as well.

6. ACKNOWLEDGMENTS

This research is co-sponsored by the National fund project of natural science (No. 60533080), and the Open Project of the State Key Lab of VR in Beihang University, Beijing China.

7. REFERENCES

[1] Prati, A., Cucchiara,R., Mikic, I. and Trivedi, M.M. 2001. Analysis and Detection of Shadows in Video Streams:A Comparative Evaluation. In the Third Workshop on Empirical Evaluation Methods in Computer Vision - IEEE Int’l Conf. on Computer Vision and Pattern Recognition.

[2] Prati, A., Mikic, I., Trivedi, M.M. and Cucchiara. R. 2003. Detecting Moving Shadows:Algorithms and Evaluation.IEEE Trans. Pattern Analysis and Machine Intelligence, 25, 7, 918-923.

[3] Friedman, N. and Russell. S. 1997. Image segmentation in video sequences: a probabilistic approach. In Proceedings of the 13th Conference on Uncertainty in Artificial Intelligence.

[4] Mikic, I., Cosman, P., Kogut, G. and Trivedi, M.M. 2000. Moving shadow and object detection in traffic scenes. In Proceedings of Int’l Conference on Pattern Recognition, 1, 321-324.

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

[6] Onoguchi, K. 1998. Shadow elimination method for moving object detection. In Proceedings of Int’l Conference on Pattern Recognition. 1, 583-587.

[7] Mei Xiao, Chongzhao Han. 2006. Edged-based shadow removal algorithm for indoor video sequence.PR&AI, 19, 5, 640-644.

[8] Cucchiara, R., Grana, C., Piccardi, M., Prati, A. and Sirotti, S. 2001. Improving Shadow Suppression in Moving Object Detection with HSV Color Information. Proc. 4th IEEE Conference on Intelligent Transportation Systems, 334-339.

[9] Leone, A., Distante, C. and Buccolieri, F. 2005. A texture-based approach for shadow detection. In Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance. 371-376.

[10] Stauffer, C. and Grimson, W. Learning patterns of activity using real-time tracking. 2000. IEEE Transactions on Pattern Analysis and Machine Intelligence. 22, 8, 747-757.

[11] Stauffer, C. and Grimson, W. 1999. Adaptive background mixture models for real-time tracking.in Proceedings CVPR, 246-252.

[12] Mark Apperley, Laurie McLeod, Masood Masoodian, Lance Paine, Malcolm Phillips, Bill Rogers and Kirsten Thomson. 2003. Use of Video Shadow for Small Group Interaction Awareness on a Large Interactive Display Surface. at the Fourth Australasian User Interface Conference (AUIC2003), Conferences in Research and Practice in Information Technology, vol.18.

[13] Cucchiara, R., Grana, C., Neri, G., Piccardi, M. and Prati, A. 2001. The Sakbot system for moving object detection and tracking.in Video-based Surveillance Systems - Computer Vision and Distributed Processing, 145-157.

[14] Amamoto, N. and Fujii, A. 1999. Detecting obstructions and tracking moving objects by image processing technique. Electronics and Communications in Japan, 82, 11, 28-37.

[15] Jehan-Besson, S., Barlaud, M. and Aubert, G. 2001. Region-based active contours for video object segmentation with camera compensation.in Proceedings of IEEE Int’l Conference on Image Processing, 2, 61-64.