REAL-TIME MOVING OBJECT DETECTION IN VIDEO SEQUENCES USING SPATIO-TEMPORAL ADAPTIVE GAUSSIAN MIXTURE MODELS

Katharina Quast, Matthias Obermann and André Kaup
Multimedia Communications and Signal Processing, University of Erlangen-Nuremberg
Cauerstr. 7, 91058 Erlangen, Germany

Keywords: Object detection, Background modeling.

Abstract: In this paper we present a background subtraction method for moving object detection based on Gaussian mixture models which performs in real-time. Our method improves the traditional Gaussian mixture model (GMM) technique in several ways. It takes into account spatial and temporal dependencies, as well as a limitation of the standard deviation leading to a faster update of the model and a smoother object mask. A shadow detection method which is able to remove the umbra as well as the penumbra in one single processing step is further used to get a mask that fits the object outline even better. Using the computational power of parallel computing we further speed up the object detection process.

1 INTRODUCTION

The detection of moving objects in video sequences is an important and challenging task in multimedia technologies. Most detection methods follow the principle of background subtraction. To segment moving foreground objects from the background a pure background image has to be estimated. This reference background image is then subtracted from each frame and binary masks with the moving foreground objects are obtained by thresholding the resulting difference images.

In (Stauffer and Grimson, 1999; Power and Schoonees, 2002) the values of a particular pixel over time are modeled as a mixture of Gaussian distributions. Thus, the background can be modeled by a Gaussian mixture model (GMM). Once the pixel-wise GMM likelihood is obtained, the final binary mask is either generated by thresholding (Stauffer and Grimson, 1999; Power and Schoonees, 2002; KaewTraKulPong and Bowden, 2001) or according to more sophisticated decision rules (Carminati and Benois-Pineau, 2005; Li et al., 2004; Yang and Hsu, 2006). Although the Gaussian mixture model technique is quite successful the obtained binary masks are often noisy and irregular. A main reason for this is that spatial and temporal dependencies are neglected in most approaches. In (Li et al., 2004) a Bayesian framework for object detection is proposed that incorporates spectral, spatial, and temporal features. But the spatial dependency is only deployed during post processing mainly by applying morphological operations which leads to poor object contours.

We improve the standard GMM method by regarding spatial and temporal dependencies and integrating a limitation of the standard deviation into the traditional method. Combining this improved method with our fast shadow removal technique, which is inspired by the technique of (Porikli and Tuzel, 2003), leads to good binary masks without adding any complex and computational expensive extensions to the method. Thus, better masks are obtained while the computational speed of the standard GMM method is kept and further post processing can be omitted. Through parallelization of the algorithm we even achieve an enormous performance speedup.

In the follwing, an overview of the GMM method is given in Section 2. In Section 3 the proposed method is first described explaining the use of spatial and temporal dependencies, the limitation of the standard deviation, and the shadow removal technique. Experimental results and implementation issues are discussed in Section 4. Finally conclusions are drawn in Section 5.

Quast K., Obermann M. and Kaup A. (2010).
REAL-TIME MOVING OBJECT DETECTION IN VIDEO SEQUENCES USING SPATIO-TEMPORAL ADAPTIVE GAUSSIAN MIXTURE MODELS.
In *Proceedings of the International Conference on Computer Vision Theory and Applications*, pages 413-418
DOI: 10.5220/0002816904130418

2 GMM OVERVIEW

As proposed in (Stauffer and Grimson, 1999) each pixel in a scene can be modelled by a mixture of K Gaussians. The modelling is based on the estimation of the probability density of the color value for each pixel. It is assumed that the color value of a given pixel is determined by the surface of an object which is in the view of the concerned pixel. In non-static scenes up to K different objects $k = 1...K$ might come into the view of a pixel. Therefore, in a monochromatic video sequence the probability density of the color value c of a pixel $\mathbf{x}$ caused by an object k can be expressed as a Gaussian function with mean μ_k and standard deviation σ_k

$$\eta(c, \mu_k, \sigma_k) = \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{1}{2}(\frac{c-\mu_k}{\sigma_k})^2} \quad (1)$$

In case of more than one color channel the probability density of the color value of a pixel is

$$\eta(\mathbf{c}, \boldsymbol{\mu}_k, \Sigma_k) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma_k|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{c}-\boldsymbol{\mu}_k)^T \Sigma_k^{-1} (\mathbf{c}-\boldsymbol{\mu}_k)} \quad (2)$$

where $\mathbf{c}$ is the color vector and Σ is a n-by-n covariance matrix of the form $\Sigma_k = \sigma_k^2 \mathbf{I}$, because it is assumed that the RGB color channels have the same standard deviation and are independent from each other. While the latter is certainly not the case, by this assumption a costly matrix inversion can be avoided at the expense of some accuracy.

The probability of a certain pixel $\mathbf{x}$ in frame t having the color value $\mathbf{c}$ is the weighted mixture of the probability densities of the $k = 1...K$ objects

$$P(\mathbf{c}_t) = \sum_{k=1}^{K} \omega_{k,t} \cdot \eta(\mathbf{c}_t, \boldsymbol{\mu}_{k,t}, \Sigma_{k,t}) \quad (3)$$

with ω_k as the weight for the respective Gaussian distribution. For practical reasons K is limited to a small number from 3 to 5. For each new frame of the video sequence the existing model has to be updated. After that a background image is estimated based on the model and the image can be segmented into foreground and background. To update the model it is checked if the new pixel color matches one of the existing K Gaussian distributions. A pixel $\mathbf{x}$ with color $\mathbf{c}$ matches a Gaussian k if

$$|\mathbf{c} - \boldsymbol{\mu}_k| < d \cdot \sigma_k \quad (4)$$

where d is a user defined parameter. If $\mathbf{c}$ matches a distribution the model parameters are adjusted as follows:

$$\omega_{k,t} = (1-\alpha)\omega_{k,t-1} + \alpha \quad (5)$$

$$\boldsymbol{\mu}_{k,t} = (1-\rho_{k,t})\boldsymbol{\mu}_{k,t-1} + \rho_{k,t}\mathbf{c}_t \quad (6)$$

$$\sigma_{k,t} = \sqrt{(1-\rho_{k,t})\sigma_{k,t-1}^2 + \rho_{k,t}(\|\mathbf{c}_t - \boldsymbol{\mu}_{k,t}\|)^2} \quad (7)$$

where $\rho_{k,t} = \alpha/\omega_{k,t}$ according to (Power and Schoonees, 2002). For unmatched distributions only a new $\omega_{k,t}$ has to be computed following equation (17).

$$\omega_{k,t} = (1-\alpha)\omega_{k,t-1} \quad (8)$$

The other parameters remain the same. The Gaussians are now ordered by the value of the reliability measure $\omega_{k,t}/\sigma_{k,t}$ in such a way that with increasing subscript k the reliability decreases. If a pixel matches more than one Gaussian distribution the one with the most reliability is choosen. If the constraint in equation (4) is not complied and a color value can not be assigned to any of the K distributions, the least probable distribution is replaced by a distribution with the current value as its mean value, a low prior weight and an initally high standard deviation and $\omega_{k,t}$ is rescaled.

A color value is regarded to be background with higher probability (lower k) if it occurs frequently (high ω_k) and does not vary much (low σ_k). To determine the B background distributions a user defined prior probability T is used

$$B = \underset{b}{\operatorname{argmin}}\Big(\sum_{k=1}^{b} w_k > T\Big). \quad (9)$$

The rest $K - B$ distributions are foreground.

3 PROPOSED METHOD

3.1 Temporal Dependency

The traditional method takes into account only the mean temporal frequency of the color values of the sequence. The more often a pixel has a certain color value, the greater is the probability of occurrence of the corresponding Gaussian distribution. But the direct temporal dependency is not taken into account.

To detect the static background regions and to enhance adaption of the model to these regions a parameter u is introduced to measure the number of cases where the color of a certain pixel was matched to the same distribution in subsequent frames

$$u_t = \begin{cases} u_{t-1} + 1, & \text{if } k_t = k_{t-1} \\ 0 & \text{else} \end{cases} \quad (10)$$

where k_{t-1} is the distribution which matched the pixel color in the previous frame and k_t is the current Gaussian distribution. If u exceeds a threshold u_{min} the factor α is multiplied by a constant $s > 1$

$$\alpha_t = \begin{cases} \alpha_0 \cdot s, & \text{if } u_t > u_{min} \\ \alpha_0 & \text{else} \end{cases} \quad (11)$$

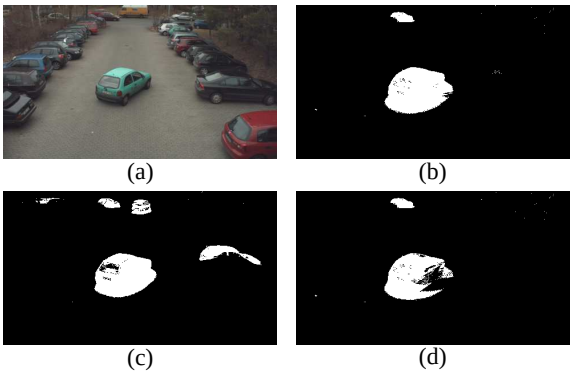

Figure 1: A frame of sequence *Parking* and the corresponding detection results of the proposed method compared to the traditional method. First row: original frame (a) and background estimated by the proposed method with temporal dependency ($\alpha_0 = 0.001$, $s = 10$, $u_{min} = 15$) (b). Bottom row: standard method with $\alpha = 0.001$ (c) and $\alpha = 0.01$ (d).

The factor α_t is now temporal dependent and α_0 is the initial user defined α. In regions with static image content the model is now faster updated as background. Since the method doesn't depend on the parameters σ and ω, the detection is also ensured in uncovered regions. In the top row of Figure 1 the original frame of sequence *Parking* and the corresponding background estimated using GMMs combined with the proposed temporal dependency approach is shown. The detection results of the standard GMM method with different values of α are shown in the bottom row of Figure 1. While the standard method either detects a lot of false positives or false negatives, the method considering temporal dependency obtains quite a good mask.

3.2 Spatial Dependency

In the standard GMM method each pixel is treated separately and spatial dependency between adjacent pixels is not considered. Therefore, false positives caused by noise based exceedance of $d \cdot \sigma_k$ in equation (4) or slight lighting changes are obtained. Since the false positives of the first type are small and isolated image regions the ones of the second type cover larger adjacent regions as they mostly appear at the border of shadows, the so called penumbra. Through spatial dependency both kinds of false positives can be eliminated.

Since in the case of false positives the color value $\mathbf{c}$ of $\mathbf{x}$ is very close to the mean of one of the B distributions, at least for one distribution $k \in [1...B]$ a small value is obtained for $|\mathbf{c} - \boldsymbol{\mu}_k|$. In general this is not the case for true foreground pixels. Instead of generating a binary mask we create a mask M with weighted foreground pixels. For each pixel $\mathbf{x} = (x, y)$

Figure 2: Detection result of the proposed method with temporal dependency (left) compared to the proposed method with temporal and spatial dependencies (right) for sequence *Parking*.

its weighted mask value is estimated according to the following equation

$$M(\mathbf{x}) = \begin{cases} 0, & \text{if } k(\mathbf{x}) \in [1...B] \\ \min\limits_{k=[1...B]}(|\mathbf{c} - \boldsymbol{\mu}_k|) & \text{else} \end{cases} \quad (12)$$

The background pixels are still weighted with zero, while the foreground pixels are weighted according to the minimum distance between the pixel and the mean of the background distributions. Thus, foreground pixels with a larger distance to the background distributions get a higher weight. To use the spatial dependency as in (Aach and Kaup, 1995), where the neighborhood of each pixel is considered, the sum of the weights in a square window W is computed. By using a threshold M_{min} the number of false positives is reduced and a binary mask BM is estimated from the weighted mask M according to

$$BM(\mathbf{x}) = \begin{cases} 1, & \text{if } \sum\limits_{W} M(\mathbf{x}) > M_{min} \\ 0 & \text{else} \end{cases} \quad (13)$$

In Figure 2 (right) part of a binary mask for sequence *Parking* obtained by GMM method considering temporal as well as spatial dependency is shown.

3.3 Avoiding Typical Detection Artefacts

If a pixel in a new frame is not described very well by the current model, the standard deviation of a Gaussian distribution modelling the foreground might increase enourmously. This happens most notably when the pixel's color value deviates tremendously from the mean of the distribution and large values of $\mathbf{c} - \boldsymbol{\mu}_k$ are obtained during the model update. The larger σ_k gets the more color values can be matched to the Gaussian distribution. Again this increases the probability of large values of $\mathbf{c} - \boldsymbol{\mu}_k$.

Figure 3 illustrates the changes of the standard deviation over time for the first 150 frames of sequence *Street* modeled by 3 Gaussians. The minimum, mean and maximum standard deviations of all Gaussian distributions for all pixels are shown (dashed lines). The

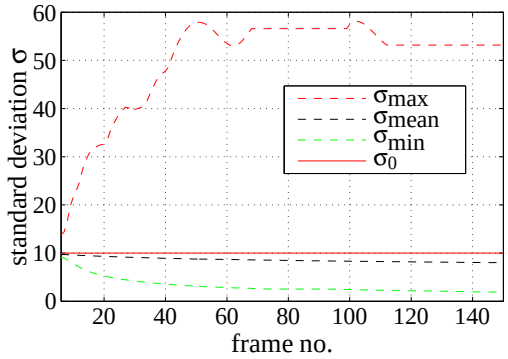

Figure 3: Maximum, mean and minimum standard deviation of all Gaussian distribution of all pixels for the first 150 frames of sequence *Street*.

maximum standard deviation increases over time and reaches high values. Hence, all pixels which are not assigned to one of the other two distributions will be matched to the distribution with the large σ value. The probability of occurrence increases and the distribution k will be concidered as a background distribution. Therefore, even foreground colors are easily but falsely identified as background colors. Thus, we suggest to limit the standard deviation to the initial standard deviation value σ_0 as demonstrated in Figure 3 by the continuous red line. In Figure 4 the traditional method (left background) is compared to the one where the standard deviation is restricted to the initial value σ_0 (right background). By examining the two backgrounds it is clearly visible that the limitation of the standard deviation improves the quality of the background model, as the dark dots and regions in the left background are not contained in the right background.

Figure 4: Background estimated for frame 97 of sequence *Street* without (left) and with limited standard deviation (right). Ellipse marks region, where detection artefacts are very likely to occur.

3.4 Single Step Shadow Removal

Even though the consideration of spatial dependency can avert the detection of most penumbra pixels, the pixels of the deepest shadow, the so called umbra, might still be detected as foreground objects. Thus, we combined our detection method with a fast shadow removal scheme inspired by the method of (Porikli and Tuzel, 2003).

Since a shadow has no affect on the hue, but changes the saturation and decreases the luminance, possible shadow pixels can be determined as follows. To find the true shadow pixels, the luminance change is computed in the RGB space by projecting the color vector $\mathbf{c}$ onto the background color value $\mathbf{b}$

$$h = \frac{\langle \mathbf{c}, \mathbf{b} \rangle}{|\mathbf{b}|} \tag{14}$$

A luminance ratio is defined as $r = |\mathbf{b}|/h$ to measure the luminance difference between $\mathbf{b}$ and $\mathbf{c}$ while the angle $\phi = \arccos(h/\mathbf{c})$ between the color vector $\mathbf{c}$ and the background color value $\mathbf{b}$ measures the saturation difference. Each foreground pixel is classified as a shadow pixel if the following two terms are both statisfied

$$r_1 < r < r_2, \qquad \phi < \phi_1 \tag{15}$$

where r_1 is the maximum allowed darkness, r_2 is the maximum allowed brightness and ϕ_1 is the maximum allowed angle separation. Since umbra pixels are considerably darker than penumbra pixels the conditions for penumbra and umbra suppression can not be satisfied simultaneously. Thus, the shadow removal scheme described in (Porikli and Tuzel, 2003) has to be run twice with different values for r_1, r_2 and ϕ_1 to remove either penumbra or umbra.

In the ϕ-r-plane the area were shadow is removed is represented by two rectangles as shown in the left graph of Figure 5. To reduce the processing time we introduce a second angle ϕ_2 and the angle constraint of equation (15) is replaced by

$$\phi < \frac{\phi_2 - \phi_1}{r_2 - r_1} \cdot (r - r_1) + \phi_1. \tag{16}$$

a new shadow detection area is defined in the ϕ-r-plane as can be seen in the right graph of Figure 5. Thus, umbra and penumbra can be removed reasonably well in one single step instead of two separated ones. To clearly show the performance, both shadow removal approaches were applied on the results of Subsection 3.2. The obtained masks are shown in Figure 6.

4 IMPLEMENTATION AND EXPERIMENTAL RESULTS

The proposed algorithm has been tested on several video sequences. After parameter testing we obtained good detection results for the sequences applying the

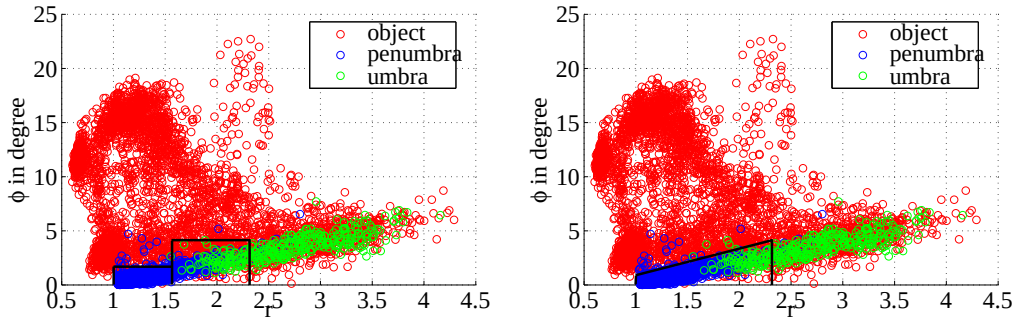

Figure 5: Detection areas (black rectangles) for umbra and penumbra removal in ϕ-r-plane using the two-step method (left) and detection area (black triangle) for shadow removal (umbra and penumbra) in ϕ-r-plane using the one-step method (right).

Figure 6: The proposed one-step shadow removal technique achieves the same result(left) as the two-step method (right), while it needs only half the processing time.

Table 1: Average detection rate R_D, average false positives rate R_{FP} and average false negatives rate R_{FN} using the traditional GMM method.

sequence	no. of frames	R_D(%)	R_{FP}(%)	R_{FN}(%)
Parking	20	98.72	1.11	0.17
Shopping	20	95.65	3.40	0.95
Airport	20	95.35	2.56	2.09

GMM method with $K = 3$, $T = 0.7$, $\alpha_0 = 0.002$, $d = 2.5$ and $\sigma_0 = 10$, while setting the parameters for temporal dependency $u_{min} = 15$ and $s = 10$ and the parameters for spatial dependency to $M_{min} = 180$ and $W = 3 \times 3$. One-step shadow removal was run with $r_1 = 1$, $r_2 = 1.7$, $\phi_1 = 4$ and $\phi_2 = 6$. For sequences *Shopping* and *Airport* the binary masks of the proposed method are compared with the results of the traditional GMM method (Stauffer and Grimson, 1999) and the statistical background modeling method of (Li et al., 2004) in Figure 7. The visual study of the masks shows that the proposed method generates reasonably good detection results which can even outperform methods with more complicated detection routines. To further evaluate the detection performance a detection rate R_D and a false alarm rate for the false positives R_{FP} and the false negatives R_{FN} were calculated for each frame and then averaged over the whole sequence. For computing R_D, R_{FP} and R_{FN} the mask is compared with a ground truth. False positives are defined as the number of background pixels that are misdetected as foreground while false negatives

Table 2: Average detection rate R_D, average false positives rate R_{FP} and average false negatives rate R_{FN} using the proposed method.

sequence	no. of frames	R_D(%)	R_{FP}(%)	R_{FN}(%)
Parking	20	99.29	0.50	0.21
Shopping	20	97.56	1.43	1.01
Airport	20	95.88	2.07	2.06

Table 3: F_1 scores of the traditional GMM method and the proposed method.

sequence	traditional GMM	proposed method
Parking	0.76	0.85
Shopping	0.70	0.81
Airport	0.65	0.68

are the number of missing foreground pixels. For sequences *Shopping* and *Airport* the ground truths from (Li et al., 2004) were used while the ground truths for sequence *Parking* were manually labeled.

By comparing the detection rates it is obvious that the proposed method (Table 2) outperforms the traditional method (Table 1). We further calculated the F_1 measure (Table 3) for sequence *Airport*:

$$F_1 = 2 \cdot \frac{Recall \cdot Precision}{Recall + Precision} \tag{17}$$

where *Precision* is the number of detected foreground pixels divided by the number of all detected pixels and *Recall* is the number of detected foreground pixels divided by the number of foreground pixels in the ground truth.

The performance of the proposed algorithm without using parallel computing is about 29 fps for 480x270 image resolution on a 2.83GHz Intel Core2 Q9550. Thus, the algorithm already performs at least as fast as the traditional GMM method while obtaining better results and is clearly faster than background subtraction methods with complex and computation-

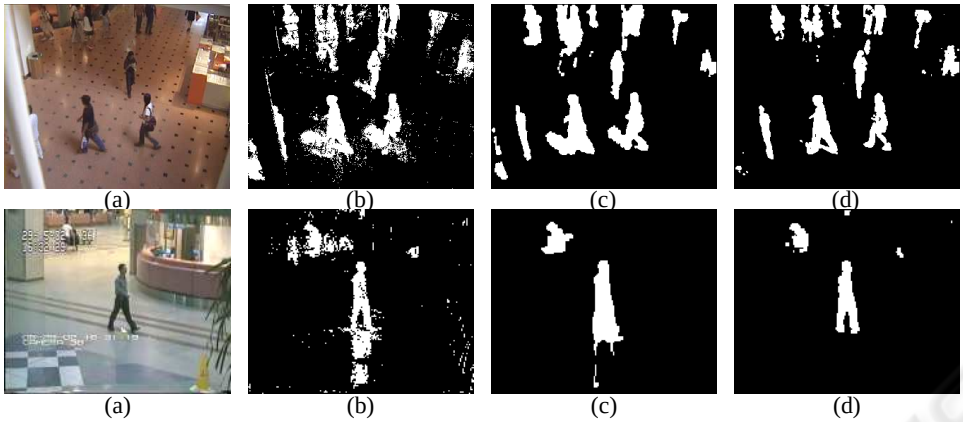

Figure 7: Original frame (a) and the corresponding detection results of the traditional method (b), the statistical background modeling method (Li et al., 2004) (c) and the proposed method (d) for sequences *Shopping* (top) and *Airport* (bottom).

Table 4: Average detection frame rate for sequence *Parking* using different numbers of threads.

threads	1	2	4	8	16	32
fps	29.20	48.54	60.16	73.21	75.43	72.76

ally expensive routines such as (Yang and Hsu, 2006). Since the GMM estimation is done independently for each pixel, parallel computing using multithreading can further speed up the object detection process. Of course it would not be practical to use a single thread for each pixel. Thus, we devide each frame into n slices. The slices are then parallel processed. By using multithreading we increased the frame rate as shown in Table 4. For each number of threads the algorithm was run 100 times and the obtained frame rates were then averaged.

5 CONCLUSIONS

A moving object detection method based on spatio-temporal adaptive GMMs is proposed. The proposed method significantly increases the quality of the detection results without increasing the needed processing time. Through parallelization of the algorithm we further achieve a speedup factor of up to 2.5 compared to a single thread implementation.

ACKNOWLEDGEMENTS

The authors would like to thank Christoph Seeger for his valuable assistance with the implementation of the algorithm.

This work has been supported by the Gesellschaft für Informatik, Automatisierung und Datenverarbeitung (iAd) and the Bundesministerium für Wirtschaft und Technologie (BMWi), funding ID 20V0801I.

REFERENCES

Aach, T. and Kaup, A. (1995). Bayesian algorithms for change detection in image sequences using Markov random fields. *Signal Processing: Image Communication*, 7(2):147–160.

Carminati, L. and Benois-Pineau, J. (2005). Gaussian mixture classification for moving object detection in video surveillance environment. In *Proc. IEEE International Conference on Image Processing*, volume 3.

KaewTraKulPong, P. and Bowden, R. (2001). An improved adaptive background mixture model for real-time tracking with shadow detection. In *Proc. 2nd European Workshop Advanced Video Based Surveillance Systems*, volume 1.

Li, L., Huang, W., Gu, I., and Tian, Q. (2004). Statistical modeling of complex backgrounds for foreground object detection. *IEEE Transactions on Image Processing*, 13(11):1459–1472.

Porikli, F. and Tuzel, O. (2003). Human body tracking by adaptive background models and mean-shift analysis. In *Proc. IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*.

Power, P. W. and Schoonees, J. A. (2002). Understanding background mixture models for foreground segmentation. In *Proc. Image and Vision Computing*, pages 267–271.

Stauffer, C. and Grimson, W. E. L. (1999). Adaptive background mixture models for real-time tracking. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2.

Yang, S. and Hsu, C. (2006). Background modeling from gmm likelihood combined with spatial and color coherency. In *Proc. IEEE International Conference on Image Processing*.