In Proc. 2nd European Workshop on Advanced Video Based Surveillance Systems, AVBS01. Sept 2001.
VIDEO BASED SURVEILLANCE SYSTEMS: Computer Vision and Distributed Processing,
Kluwer Academic Publishers

An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection

P. KaewTraKulPong and R. Bowden
Vision and Virtual Reality group,
Department of Systems Engineering,
Brunel University, Middlesex, UB8 3PH, UK.
Email: Pakorn.Kaewtrakulpong@brunel.ac.uk
Email: Richard.Bowden@brunel.ac.uk

Abstract

Real-time segmentation of moving regions in image sequences is a fundamental step in many vision systems including automated visual surveillance, human-machine interface, and very low-bandwidth telecommunications. A typical method is background subtraction. Many background models have been introduced to deal with different problems. One of the successful solutions to these problems is to use a multi-colour background model per pixel proposed by Grimson *et al* [1,2,3]. However, the method suffers from slow learning at the beginning, especially in busy environments. In addition, it can not distinguish between moving shadows and moving objects. This paper presents a method which improves this adaptive background mixture model. By reinvestigating the update equations, we utilise different equations at different phases. This allows our system learn faster and more accurately as well as adapt effectively to changing environments. A shadow detection scheme is also introduced in this paper. It is based on a computational colour space that makes use of our background model. A comparison has been made between the two algorithms. The results show the speed of learning and the accuracy of the model using our update algorithm over the Grimson *et al*'s tracker. When incorporate with the shadow detection, our method results in far better segmentation than that of Grimson *et al*.

1 Introduction

Background subtraction involves calculating a reference image, subtracting each new frame from this image and thresholding the result. What results is a binary segmentation of the image which highlights regions of non-stationary objects. The simplest form of the reference image is a time-averaged background image. This method suffers from many problems and requires a training period absent of foreground objects. The motion of background objects after the training period and foreground objects motionless during the training period would be considered as permanent foreground objects. In addition, the approach cannot cope with gradual illumination changes in the scene. These problems lead to the requirement that any solution must constantly reestimate the background model. Many adaptive background-modelling methods have been proposed to deal with these slowly-changing stationary signals. Friedman and Russell modelled each pixel in a camera scene by an adaptive parametric mixture model of three Gaussian distributions [4]. They also provide some brief discussion on the online update equations based on *sufficient statistics*. Koller *et al* used a Kalman filter to track the changes in background illumination for every pixel [5]. They applied a selective update scheme to include only the probable background values into the estimate of the background. The methods can cope well with the illumination changes; however, can not handle the problem of objects being introduced or removed from the scene. One solution is to use a multiple-colour background model per pixel. Grimson *et al* employed an adaptive nonparametric Gaussian mixture model to solve these problems [1,2,3]. Their model can also lessen the effect of small repetitive motions; for example, moving vegetation like trees and bushes as well as small camera displacement. Elgammal *et al* used a *kernel* estimator for each pixel [6]. Kernel exemplars were taken from a moving window. They also introduced a method to reduce the result of small motions by employing a *spatial coherence*. This was done by comparing simply connected components to the background model of its circular neighbourhood. Although the authors presented a number of speed-up routines, the approach was still of high computational complexity. Other techniques using high level processing to assist the background modelling have been proposed; for instance, the Wallflower tracker [7] which circumvents some of these problems using high level processing rather than tackling the inadequacies of the background model. Our method is based on Grimson et al's framework [1,2,3], the differences lie in the update equations, initialisation method and the introduction of a shadow detection algorithm.

A common optimisation scheme used to fit a Gaussian mixture model is the *Expectation Maximisation* (EM) algorithm. The EM algorithm is an iterative method that guarantees to converge to a local maximum in a search space. Due to the space-time requirements in modelling each pixel for the background image, an online EM algorithm is required. Many online EM algorithms have been introduced. They can be classified into two groups. The first group was in the realm of parametric estimation of *probability density functions* (pdf's). In other words, to use new data in updating the previous estimate without modifying the structure of the previous model. The procedure was introduced by Nowlan [8] and explained in terms of the results by Neal and Hinton [9]. Traven derived an *N* most recent window version of the procedure [10]. McKenna *et al* [11,12,13] extended the result of Traven [10] to an *L* most recent window of the results from *L* batch EM runs and used it for tracking a multi-colour foreground object. This parametric estimation approach can not run effectively without a good initial estimate (normally found by running the batch EM algorithm). The second group is that of non-parametric approaches. Priebe et al introduced an adaptive mixture model with stochastic thresholding for generating new Gaussian kernels to the existing mixture model [14,15]. Grimson and Stauffer [2,3], however, applied the same scheme with deterministic thresholding.

In addition to Grimson *et al*, many other authors have applied mixture models to model every pixel in camera scenes. Rowe and Blake applied the batch EM algorithm for off-line training in their *virtual* image plane [16]. However, the model does not update with time and therefore leads to failure for external environments where the scene lighting changes with time. Friedman and Russell modelled road, shadow and vehicle distribution for each pixel using an adaptive mixture of three Gaussian distributions [4]. The classification was based on a heuristic method of relative distances in the intensity space. They reported a good segmentation using the expected sufficient statistics formulas. However, this still requires a preprocessing initialisation to learn an initial model using batch EM algorithm.

We explain the background model by Grimson and Stauffer [2,3] and its deficiencies in section 2.1. Our proposed solution to the problem is presented in section 2.2. Section 2.3 explains our shadow detection. Results from each method are shown and compared in section 3 and concluded in section 4.

2 Background Modelling

In this section, we discuss the work of Grimson and Stauffer [2,3] and its shortcomings. The authors introduces a method to model each background pixel by a mixture of *K* Gaussian distributions (*K* is a small number from 3 to 5). Different Gaussians are assumed to represent different colours. The weight parameters of the mixture represent the time proportions that those colours stay in the scene. Unlike Friedman et al's work, the background components are determined by assuming that the background contains *B* highest probable colours. The probable background colours are the ones which stay longer and more static. Static single-colour objects trend to form tight clusters in the colour space while moving ones form widen clusters due to different reflecting surfaces during the movement. The measure of this was called the *fitness* value in their papers. To allow the model to adapt to changes in illumination and run in real-time, an update scheme was applied. It is based upon selective updating. Every new pixel value is checked against existing model components in order of fitness. The first matched model component will be updated. If it finds no match, a new Gaussian component will be added with the mean at that point and a large covariance matrix and a small value of weighting parameter.

2.1 Adaptive Gaussian Mixture Model

Each pixel in the scene is modelled by a mixture of *K* Gaussian distributions. The probability that a certain pixel has a value of $\mathbf{x}_N$ at time *N* can be written as

$$p(\mathbf{x}_N) = \sum_{j=1}^{K} w_j \eta(\mathbf{x}_N; \boldsymbol{\theta}_j)$$

xác suất pixel có giá trị x_N tại thời điểm N

where w_k is the weight parameter of the k[th] Gaussian component. $\eta(\mathbf{x}; \boldsymbol{\theta}_k)$ is the Normal distribution of k[th] component represented by

$$\eta(\mathbf{x}; \boldsymbol{\theta}_k) = \eta(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{(2\pi)^{\frac{D}{2}} |\boldsymbol{\Sigma}_k|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}-\boldsymbol{\mu}_k)}.$$

where $\boldsymbol{\mu}_k$ is the mean and $\boldsymbol{\Sigma}_k = \sigma_k^2 \mathbf{I}$ is the covariance of the k[th] component.

The K distributions are ordered based on the fitness value w_k/σ_k and the first B distributions are used as a model of the background of the scene where B is estimated as

$$B = \arg\min_b \left(\sum_{j=1}^{b} w_j > T \right)$$

The threshold T is the minimum fraction of the background model. In other words, it is the minimum prior probability that the background is in the scene. Background subtraction is performed by marking a foreground pixel any pixel that is more than 2.5 standard deviations away from any of the B distributions. The first Gaussian component that matches the test value will be updated by the following update equations,

$$\hat{w}_k^{N+1} = (1-\alpha)\hat{w}_k^N + \alpha\hat{p}(\omega_k \mid \mathbf{x}_{N+1})$$
$$\hat{\boldsymbol{\mu}}_k^{N+1} = (1-\alpha)\hat{\boldsymbol{\mu}}_k^N + \rho\mathbf{x}_{N+1}$$
$$\hat{\boldsymbol{\Sigma}}_k^{N+1} = (1-\alpha)\hat{\boldsymbol{\Sigma}}_k^N + \rho(\mathbf{x}_{N+1} - \hat{\boldsymbol{\mu}}_k^{N+1})(\mathbf{x}_{N+1} - \hat{\boldsymbol{\mu}}_k^{N+1})^T$$
$$\rho = \alpha\eta(\mathbf{x}_{N+1}; \hat{\boldsymbol{\mu}}_k^N, \hat{\boldsymbol{\Sigma}}_k^N)$$
$$\hat{p}(\omega_k \mid \mathbf{x}_{N+1}) = \begin{cases} 1 & ;\text{if } \omega_k \text{ is the first match Gaussian component} \\ 0 & ;\text{otherwise} \end{cases}$$

anpha: learning rate

where ω_k is the k[th] Gaussian component. $1/\alpha$ defines the time constant which determines change. If none of the K distributions match that pixel value, the least probable component is replaced by a distribution with the current value as its mean, an initially high variance, and a low weight parameter. According to their papers [1,2,3], only two parameters, α and T, needed to be set for the system.

The details of its robustness were explained in their papers [1,2,3]; however, with a simple discussion, we can see its incapability. Firstly, if the first value of a given pixel is a foreground object, there is only one Gaussian where its weight equals unity. With only one-colour subsequent background values, it will take $\log_{(1-\alpha)}(T)$ frames until the genuine background can be considered as a background and $\log_{(1-\alpha)}(0.5)$ frames until it will be the dominant background component. For example, if we assume that at least 60% of the time the background is present and α is 0.002 (500 recent frames), it would take 255 frames and 346 frames for the component to be included as part of the background and the dominant background component, respectively. The situation can be worse in busy environments where a clean background is rare. This paper presents a solution to the problem in the next section. Secondly, ρ is too small due to the likelihood factor. This leads to too slow adaptations in the means and the covariance matrices, therefore the tracker can fail within a few seconds after initialisation. One solution to this is to simply cut out the likelihood term from ρ.

2.2 Online EM Algorithms

We begin our estimating of the Gaussian mixture model by expected sufficient statistics update equations then switch to L-recent window version when the first L samples are processed. The expected sufficient statistics update equations provide a good estimate at the beginning before all L samples can be collected. This initial estimate improves the accuracy of the estimate and also the performance of the tracker allowing fast convergence on a stable background model. The L-recent window update equations gives priority over recent data therefore the tracker can adapt to changes in the environment.

The online EM algorithms by expected sufficient statistics are shown in the left column while the by L-recent window version in the right.

$$\hat{w}_k^{N+1} = \hat{w}_k^N + \frac{1}{N+1}\left(\hat{p}(\omega_k \mid \mathbf{x}_{N+1}) - \hat{w}_k^N\right)$$

$$\hat{w}_k^{N+1} = \hat{w}_k^N + \frac{1}{L}\left(\hat{p}(\omega_k \mid \mathbf{x}_{N+1}) - \hat{w}_k^N\right)$$

$$\hat{\boldsymbol{\mu}}_k^{N+1} = \hat{\boldsymbol{\mu}}_k^N + \frac{\hat{p}(\omega_k \mid \mathbf{x}_{N+1})}{\sum_{i=1}^{N+1}\hat{p}(\omega_k \mid \mathbf{x}_i)}\left(\mathbf{x}_{N+1} - \hat{\boldsymbol{\mu}}_k^N\right)$$

$$\hat{\boldsymbol{\mu}}_k^{N+1} = \hat{\boldsymbol{\mu}}_k^N + \frac{1}{L}\left(\frac{\hat{p}(\omega_k \mid \mathbf{x}_{N+1})\mathbf{x}_{N+1}}{\hat{w}_k^{N+1}} - \hat{\boldsymbol{\mu}}_k^N\right)$$

$$\hat{\boldsymbol{\Sigma}}_k^{N+1} = \hat{\boldsymbol{\Sigma}}_k^N + \frac{\hat{p}(\omega_k \mid \mathbf{x}_{N+1})}{\sum_{i=1}^{N+1}\hat{p}(\omega_k \mid \mathbf{x}_i)}\left((\mathbf{x}_{N+1} - \hat{\boldsymbol{\mu}}_k^N)(\mathbf{x}_{N+1} - \hat{\boldsymbol{\mu}}_k^N)^T - \hat{\boldsymbol{\Sigma}}_k^N\right)$$

$$\hat{\boldsymbol{\Sigma}}_k^{N+1} = \hat{\boldsymbol{\Sigma}}_k^N + \frac{1}{L}\left(\frac{\hat{p}(\omega_k \mid \mathbf{x}_{N+1})(\mathbf{x}_{N+1} - \hat{\boldsymbol{\mu}}_k^N)(\mathbf{x}_{N+1} - \hat{\boldsymbol{\mu}}_k^N)^T}{\hat{w}_k^{N+1}} - \hat{\boldsymbol{\Sigma}}_k^N\right)$$

2.3 Shadow Detection and Colour Model

As it is evidence in their papers [1,2,3], Grimson *et al*'s tracker can not identify moving shadows from the objects casting them. The reason behind this is that no heuristic exists to label Gaussian components as moving shadows. One solution is to use a chromatic colour space representation which reduces susceptibility. As many colour spaces can separate chromatic and illumination components, maintaining a chromatic model regardless of the brightness can lead to an unstable model especially for very bright or dark objects. This conversion also requires computational resources particularly in large images. The idea of preserving intensity components and saving computational costs lead us back to the RGB space. As the requirement to identify moving shadows, we need to consider a colour model that can separate chromatic and brightness components. It should be compatible and make use of our mixture model. This is done by comparing a non-background pixel against the current background components. If the difference in both chromatic and brightness components are within some thresholds, the pixel is considered as a shadow. We use an effective computational colour model similar to the one proposed by Horprasert et al [17] to fulfil these needs. It consists of a position vector at the RGB mean of the pixel background, E, an *expected chromaticity line*, $\|E\|$, a *chromatic distortion*, d, and a brightness threshold, τ. For a given observed pixel value, I, a brightness distortion, a, and a colour distortion, c, from the background model can be calculated as

$$a = \arg\min_{z}(I - zE)^2 \text{ and}$$

$$c = \|I - aE\|$$

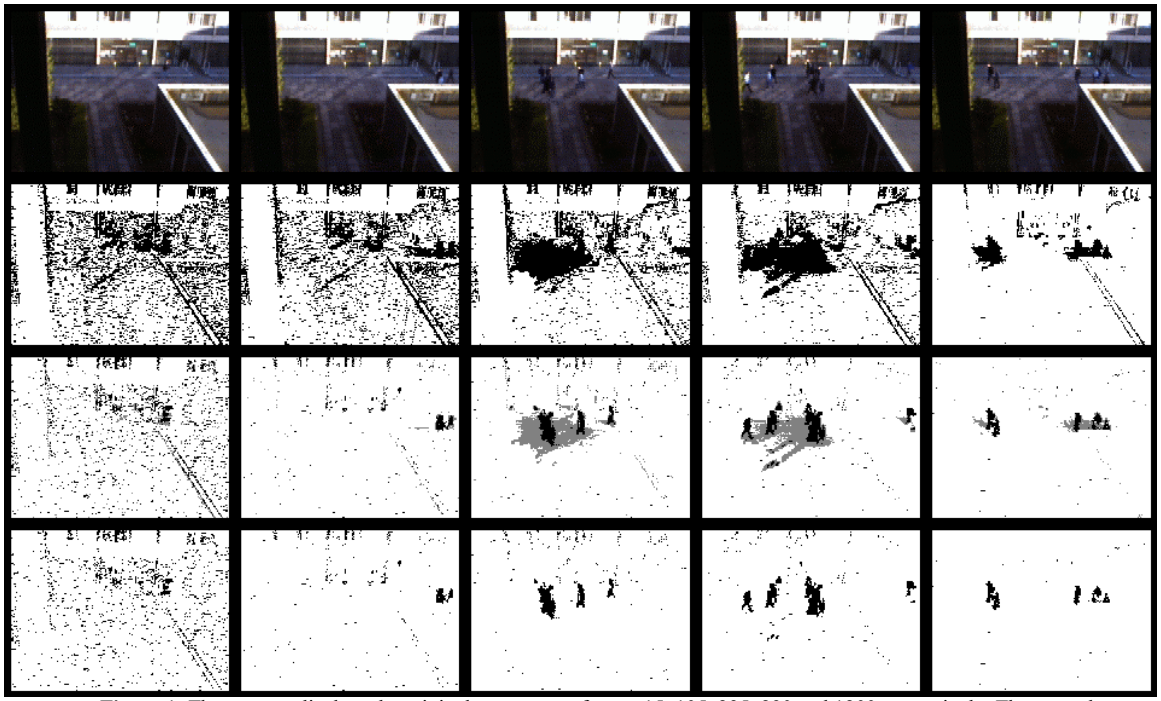
With the assumption of spherical Gaussian distribution in each mixture component, the standard deviation of the k[th] component σ_k can be set equal to d. The calculation of a and c are trivial using vector dot product. A non-background observed sample is considered a moving shadow if a is within, in our case, 2.5 standard deviations and $\tau < c < 1$.

3 Experiment

This section demonstrates the performance of the Grimson model [2,3] and our proposed algorithms on an image sequence. The sequence shown here is 192x144 images. We used an adaptive mixture of five Gaussian components. The L was set at 500 frames (α=0.002 in Grimson et al's) and the threshold T was set at 0.6. In the shadow detection module, the brightness threshold, τ of 0.7 was used. To show the performance of the background models, higher level processes such as noise cleaning or connected component analysis algorithms were not introduced to the results of background subtractions. Figure 1 shows a sequence of busy outdoor scene containing people walking in a public pathway. The sequence includes strong sunshine, large shaded area, tree, reflections from windows and long moving shadows. We have presented images of the initial phase, busy scene and a long run. Because of no clean images at the beginning, an artefact of the initial image left in Grimson et al's tracker lasted for over a hundred frames. Better segmentation can be seen from our method. The performance enhances dramatically with the shadow detection module.

4 Conclusion

We have presented new update algorithms for learning adaptive mixture models of background scene for the real-time tracking of moving objects. The algorithm run under the framework of the real-time robust tracker proposed by Grimson et al. A comparison has been made between the two algorithms. The results show the speed of learning and the accuracy of the model using our update algorithm over the Grimson et al's tracker. We have proposed a method to detect moving shadows using our existing mixture model. This significantly reduces additional computational burdens. Shadow detection need only be performed upon pixels labelled as foreground and therefore with negligible computational overheads the moving shadows can be detected successfully. The shadow detection also reduces the effect of small repetitive motions in the background scene.



Figrue 1: The top row displays the original sequence at frames 15, 105, 235, 290 and 1200 respectively. The second row shows the results from Grimson et al's. The last two rows are the results of our proposed method with and without moving shadows displayed in the images. The shadows are shown in grey.

References

[1] Grimson Wel, Stauffer C. Romano R. Lee L. *Using adaptive tracking to classify and monitor activities in a site*. in *Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No.98CB36231). IEEE Comput. Soc. 1998*. 1998.

[2] Stauffer C, Grimson W. E. L. *Adaptive background mixture models for real-time tracking*. in *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149). IEEE Comput. Soc. Part Vol. 2, 1999*.

[3] Stauffer C, Grimson W. E. L., *Learning patterns of activity using real-time tracking.* IEEE Transactions on Pattern Analysis & Machine Intelligence, 2000. **22**(8): p. 747-57.

[4] Friedman N., Russell S. *Image Segmentation in Video Sequences: A Probabilistic Approach*. in *The Thirteenth Conference on Uncertainty in Artificial Intelligence*. 1997. Brown University, Providence, Rhode Island, USA: Morgan Kaufmann Publishers, Inc., San Francisco, 1997.

[5] Koller D, Weber J. Huang T. Malik J. Ogasawara G. Rao B. Russell S. *Towards robust automatic traffic scene analysis in real-time*. in *Proceedings of the 33rd IEEE Conference on Decision and Control (Cat. No.94CH3460-3). IEEE. Part vol.4, 1994*. 1994.

[6] Elgammal A., Harwood D., Davis L. *non-parametric model for background subtraction*. in *IEEE ICCV'99 FRAME-RATE WORKSHOP*. 1999.

[7] Toyama K, Krumm J. Brumitt B. Meyers B. *Wallflower: principles and practice of background maintenance*. in *Proceedings of the Seventh IEEE International Conference on Computer IEEE Comput. Soc. Part vol.1, 1999*. 1999.

[8] Nowlan, S. J., *Soft Competitive Adaptation: Neural Network Learning Algorithms based on Fitting Statistical Mixtures*, in *School of Computer Science*. 1991, Carnegie Mellon University: Pittsburgh, PA.

[9] Neal, R. M., Hinton, G. E., *A view of the EM algorithm that justifies incremental, sparse, and other variants*, in *Learning in Graphical Models*, M. I. Jordan, Editor. 1998, Dordrecht: Kluwer Academic Publishers. p. 355-368.

[10] Traven, H. G. C., *A neural network approach to statistical pattern classification by 'semiparametric' estimation of probability density functions.* IEEE Transactions on Neural Networks, 1991. **2**(3): p. 366-77.

[11] McKenna Sj, Raja Y. Shaogang Gong, *Object tracking using adaptive colour mixture models.* Computer Vision - ACCV '98. Third Asian Conference on Computer Vision. Proceedings. Springer-Verlag. Part vol.1, 1997, 1998: p. 615-22 vol.

[12] Raja Y, McKenna S. J. Gong S., *Color model selection and adaptation in dynamic scenes.* Computer Vision - ECCV'98. 5th European Conference on Computer Vision. Proceedings. Springer-Verlag. Part vol.1, 1998, 1998: p. 460-74 vol.

[13] Raja Y, McKenna S. J. Shaogang Gong, *Segmentation and tracking using colour mixture models.* Computer Vision - ACCV '98. Third Asian Conference on Computer Vision. Proceedings. Springer-Verlag. Part vol.1, 1997, 1998: p. 607-14 vol.

[14] Priebe Ce, Marchette D. J., *Adaptive mixtures: recursive nonparametric pattern recognition.* Pattern Recognition, 1991. **24**(12): p. 1197-209.

[15] Priebe Ce, Marchette D. J., *Adaptive mixture density estimation.* Pattern Recognition, 1993. **26**(5): p. 771-85.

[16] Rowe S., Blake A. *Statistical background modelling for tracking with a virtual camera*. in *BMVC `95 Proceedings of the 6th British Machine Vision Conference. BMVA Press. Part vol.2, 1995*. 1995.

[17] Horprasert T., Harwood D., Davis L.S. *a statistical approach for real-time robust background subtraction and shadow detection*. in *IEEE ICCV'99 FRAME-RATE WORKSHOP*. 1999.