Understanding Background Mixture Models for Foreground Segmentation

P. Wayne Power Johann A. Schoonees

Industrial Research Limited, PO Box 2225, Auckland, New Zealand
email: {w.power, j.schoonees}@irl.cri.nz

Proceedings Image and Vision Computing New Zealand 2002
University of Auckland, Auckland, New Zealand
26–28th November 2002

Understanding Background Mixture Models for Foreground Segmentation

P. Wayne Power* Johann A. Schoonees†

Industrial Research Limited, PO Box 2225, Auckland, New Zealand

Abstract

The seminal video surveillance papers on moving object segmentation through adaptive Gaussian mixture models of the background image do not provide adequate information for easy replication of the work. They also do not explicitly base their algorithms on the underlying statistical theory and sometimes even suffer from errors of derivation. This tutorial paper describes a practical implementation of the Stauffer-Grimson algorithm and provides values for all model parameters. It also shows what approximations to the theory were made and how to improve the standard algorithm by redefining those approximations.

Keywords: background subtraction, video surveillance, mixture models

1 Introduction

This tutorial paper revisits a family of recent background subtraction techniques which have not been well expounded in the published literature in terms of their theoretical foundation. As a result it is not clear from the seminal papers what approximations to the theoretically optimal methods have been made, nor obvious how the algorithms can be modified for either better segmentation or faster processing.

We shall assume (1) foreground segmentation by exception to a background model (rather than by directly modelling the foreground colour, texture, or edges); (2) per-pixel processing (rather than region based); and (3) per-frame decisions (not making use of decision feedback or tracking information). These assumptions represent the line of least resistance in defining a self-contained problem in an environment where computational resources are the most serious practical limitation.

Given the assumptions, the most obvious approach is to maintain a background image as a cumulative average of the video stream and to segment moving objects by thresholding a per-pixel distance between the current frame and the background image. This method is the foundation of a collection of techniques generally known as *background subtraction* [McIvor 2000].

Simple background subtraction has the advantage of computational speed but fails in uncontrolled environments. The most common problems involve changing illumination levels and temporal background clutter as often found in outdoor scenes. These two problems are usually addressed by making the background model adaptive so that its parameters can track changing illumination and by making the model more complex so that it can more accurately represent multimodal backgrounds.

The algorithm by Stauffer and Grimson [1999] is representative of an adaptive method which uses a mixture of normal distributions to model a multimodal background image sequence. For each pixel, each normal distribution in its background mixture corresponds to the probability of observing a particular intensity or colour in the pixel. This algorithm will be used as the baseline for comparison through the rest of this paper.

*email: w.power@irl.cri.nz

†email: j.schoonees@irl.cri.nz

The Stauffer-Grimson algorithm relies on assumptions that the background is visible more frequently than any foregrounds and that it has modes with relatively narrow variance. These assumptions are consistent with scenes in which the background clutter is generated by more than one surface appearing in the pixel view. Each surface is represented by a normal distribution having a mean equal to the surface intensity or colour and a variance due to surface texture, illumination fluctuations, or camera noise.

2 Theoretical derivation

2.1 Problem statement

Each surface (or uniform object) which comes into the view of a given pixel is represented by one of a set of states $k \in \{1,2,\ldots,K\}$ where the number of surfaces K is an assumed constant (usually between 3 and 7). Some of the K states correspond to background objects and the rest are deemed to be foreground. The process $\mathbf{k}$ which generates the state at each frame time $t = 1,2,\ldots$ is simply modelled by a set of K parameters $\omega_k = P(k),\ k = 1,2,\ldots,K$, each representing the *a priori* probability of surface k appearing in the pixel view, and $\sum_{k=1}^{K} \omega_k = 1$.

The surface process $\mathbf{k}$ is hidden (unpredicted scene activity) and only indirectly observed through the associated pixel value X. Even if it was known which surface k was in view, the pixel value would still have some distribution $f(X|k)$ due to small illumination changes, camera noise, or surface texture[1]. The pixel values are therefore samples of some random variable $\mathbf{X}$ which includes the behaviour of $\mathbf{k}$. $\mathbf{X}$ may be one-dimensional (monochrome intensity), 2D (normalized colour space or intensity-plus-range), 3D (colour), or n-dimensional in general (represented as column vectors).

The most general solution to the foreground segmentation problem is at each sample time t to estimate the most likely state k from a set of observations sampled from $\mathbf{X}$, along with a procedure for demarcating the foreground states from the background states.

The pixel value process $\mathbf{X}$ is assumed to be modelled by a mixture of K Gaussian densities with parameter sets θ_k, one for each state k:

$$f_{\mathbf{X}|k}(X|k,\theta_k) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma_k|^{\frac{1}{2}}} e^{-\frac{1}{2}(X-\mu_k)^T\Sigma_k^{-1}(X-\mu_k)} \qquad (1)$$

where μ_k is the mean and Σ_k is the covariance matrix of the kth density.

A further assumption is usually made that the dimensions of $\mathbf{X}$ are independent so that Σ_k is diagonal—more easily invertible—and may be represented by the n-dimensional variance σ_k^2. Stauffer and Grimson [1999] go even further in assuming that the n variances are identical, implying for example that deviations in the red, green, and blue dimensions of a colour space have the same statistics. A single scalar σ_k may be a reasonable approximation in such a linear colour space, but it may be an excessive simplification in

[1] It is convenient to use the single notation k to represent the random variable $\mathbf{k}$, its values k, and the sampling event $\mathbf{k} = k$.

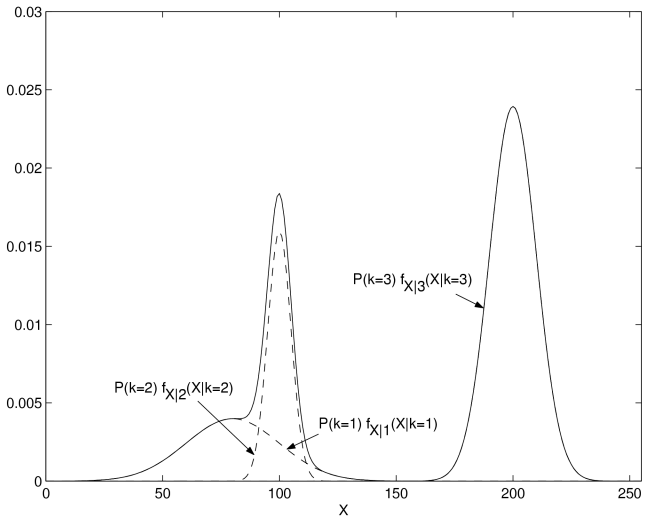

Figure 1: The pixel value probability $f_{\mathbf{X}}(X|\Phi)$ of (2) illustrated for 1D pixel values $X \in \{0,1,\ldots,255\}$, $K = 3$, $\omega_k = \{0.2, 0.2, 0.6\}$, $\mu_k = \{80, 100, 200\}$, and $\sigma_k = \{20, 5, 10\}$.

other applications. Care should be taken with non-linear colour spaces like hue, saturation, and value and especially with spaces combining unlike quantities such as intensity and range, where each dimension is likely to have a peculiar distribution.

The density parameter set is defined as $\theta_k = \{\mu_k, \sigma_k\}$ for a given k and the total set of parameters becomes $\Phi = \{\omega_1, \ldots, \omega_K, \theta_1, \ldots, \theta_K\}$.

Because the events k are disjoint, the distribution of $\mathbf{X}$ may be modelled as a sum-of-Gaussians mixture (see Figure 1)

$$f_{\mathbf{X}}(X|\Phi) = \sum_{k=1}^{K} P(k)\, f_{\mathbf{X}|k}(X|k,\theta_k) \tag{2}$$

where $P(k) = \omega_k$. All the parameters Φ need to be estimated from observations of $\mathbf{X}$ in parallel with the estimation of the hidden state k.

The Stauffer-Grimson [1999] algorithm solves this problem—maximum likelihood parameter estimation from incomplete data—with an approximate formulation of the expectation-maximization (EM) algorithm [Dempster et al. 1977]. The EM algorithm works by iterating two steps: (*E-step*) finding the expected value with respect to the hidden data of the likelihood function of the complete data (observed and hidden) using the observed data and current estimates of the parameters; and (*M-step*) calculating maximum likelihood estimates of the parameters using the observed data and current estimates of the hidden data.

2.2 Estimating current state

The first step is to estimate which of the K distributions most likely gave rise to the current sample $\mathbf{X} = X$. The posterior probability $P(k|X,\Phi)$ is the likelihood that this pixel value was generated in state k, given by Bayes's theorem (see Figure 2):

$$P(k|X,\Phi) = \frac{P(k)\, f_{\mathbf{X}|k}(X|k,\theta_k)}{f_{\mathbf{X}}(X|\Phi)}. \tag{3}$$

The k which maximizes $P(k|X,\Phi)$ (called the *match* in [Stauffer and Grimson 1999]) is the maximum *a posteriori* (MAP) estimate

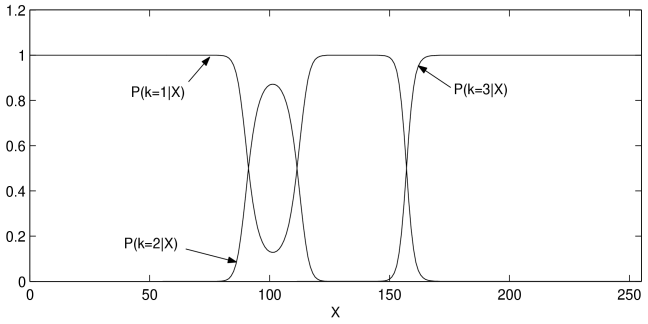

Figure 2: The *a posteriori* probabilities $P(k|X,\Phi)$ plotted as functions of X for each $k = 1,2,3$, using the same parameters as in Figure 1.

$$\begin{aligned} \hat{k} &= \operatorname*{argmax}_k P(k|X,\Phi) \\ &= \operatorname*{argmax}_k \omega_k\, f_{\mathbf{X}|k}(X|k,\theta_k) \end{aligned} \tag{4}$$

where the second equality follows because $f_{\mathbf{X}}(X|\Phi)$ in (3) is independent of k.

Equation (4) is correct as far as it goes but it fails to convey an important practical point: the current input may not have been generated by any of the K surfaces. This typically happens when a previously unseen foreground object appears in view of the pixel. Equations (3) and (4) are therefore incomplete in that they do not take into account the possibility that the input is not captured by the model. For example, if $X \approx 150$ in Figure 1 then it is more likely to have been caused by a new surface than by state $k = 1$ as suggested by Figure 2.

One may accommodate a previously unseen input by assigning a corresponding small but non-zero *a priori* probability to such an event. This could be done by adding a flat default $(K+1)$th distribution to the K Gaussians. It, too, will have a prior ω_{K+1}, but an undefined mean and infinite variance. The effect will be to add a level threshold representing the $(K+1)$th distribution to Figure 1. (4) will then select the default $(K+1)$th distribution whenever the observed pixel value is not captured by any of the model's K Gaussians.

2.3 Segmenting the foreground

The mixture model (2) models both foreground and background surfaces without distinction. This is why a total of $K = 3$ Gaussians may be considered a practical minimum to model two background surfaces and one foreground surface in each pixel. (With fewer than two background modes the algorithm is unnecessarily complex and it would be easier to use simple subtraction of an averaged background image. At a minimum the algorithm can work with only one foreground Gaussian because it can be used roughly to model any foreground.) Up to $K = 7$ has been reported in practical applications but it is likely that not much improvement is obtained beyond $K = 5$ distributions. Once the current state k is estimated, a determination has to be made as to whether it represents a foreground or a background surface.

The Stauffer-Grimson [1999] procedure for demarcation starts by ranking the K states by a criterion ω_k/σ_k which is proportional to the peak amplitude of the weighted distribution $\omega_k f_{\mathbf{X}|k}(X|k,\theta_k)$. The original algorithm uses a scalar σ_k (as discussed in Section 2.1).

If σ_k is n-dimensional (as recommended for non-linear or composite input spaces) the ranking has to be done with $\omega_k/\|\sigma_k\|$ or $\omega_k^2/\|\sigma_k\|^2$. If the dimensions of σ_k are of widely different numerical sizes, care is needed to avoid the larger dimensions dominating $\|\sigma_k\|^2$.

A surface is deemed to be background with higher probability (lower subscript k) if it occurs frequently (high ω_k) and does not vary much (low σ_k). To demarcate the background they provide an overall prior probability T of anything in view being background. The first B of the ranked states whose accumulated probability accounts for T are deemed to be background,

$$B = \underset{b}{\operatorname{argmin}} \left(\sum_{k=1}^{b} \omega_k > T \right), \tag{5}$$

and the rest of the states are by default foreground. The algorithm largely succeeds or fails to the extent that these assumptions are true or false.

2.4 Estimating the parameters

The complete-data likelihood function (including k) is given by

$$P(X_1, X_2, \ldots, X_N, k|\Phi) = \prod_{t=1}^{N} \omega_k f_{\mathbf{X}|k}(X_t|k, \theta_k) \tag{6}$$

where the notation has changed slightly to show X_t explicitly as the pixel value at time t and assuming that there are a total of N samples of $\mathbf{X}$. Renewed estimates of the parameters Φ are obtained by maximizing the expected value of (6) with respect to k. The derivation is too long-winded to repeat here but may be found in, for example, [Bilmes 1998]. The results are, for $k = 1, 2, \ldots, K$:

$$\hat{\omega}_k = \frac{1}{N} \sum_{t=1}^{N} P(k|X_t, \Phi) \tag{7}$$

$$\hat{\mu}_k = \frac{\sum_{t=1}^{N} X_t P(k|X_t, \Phi)}{\sum_{t=1}^{N} P(k|X_t, \Phi)} \tag{8}$$

$$\hat{\sigma}_k^2 = \frac{\sum_{t=1}^{N} ((X_t - \hat{\mu}_k) \circ (X_t - \hat{\mu}_k)) P(k|X_t, \Phi)}{\sum_{t=1}^{N} P(k|X_t, \Phi)} \tag{9}$$

where $\circ$ is the element-wise (Hadamard) multiplication operator and $P(k|X_t, \Phi)$ is given in each case by (3).

Equations (7) to (9) assume stationary processes $\mathbf{k}$ and $\mathbf{X}$, and also a fixed number of observations N. A practical implementation that is capable of foreground segmentation of each frame as it is acquired has to re-estimate the current surface k and all the parameters incrementally from each new sample $\mathbf{X} = X_t$ as well as adapt to changing scene statistics.

3 A practical algorithm

3.1 Generating on-line averages

The three parameter updating equations (7) to (9) are all averages of some derived observations weighted by $P(k|X_t, \Phi)$. A simple way of converting (7) to an on-line cumulative average is to define a time-varying gain $\alpha_t = 1/t$ and write, for $k = 1, 2, \ldots, K$ and $t = 1, 2, \ldots$,

$$\hat{\omega}_{k,t} = (1 - \alpha_t)\omega_{k,t} + \alpha_t P(k|X_t, \Phi) \tag{10}$$

where the time subscript has now been added to the parameters too. For each k and at any time t, $\omega_{k,t}$ is a scalar variable.

It would be desirable to accommodate some non-stationarity in $\mathbf{X}$: the model should be capable of adapting to changing illumination by emphasizing more recent samples of $\mathbf{X}$ over older samples. (10) as it stands integrates from time $t = 1$ to infinity and becomes increasingly insensitive to new input statistics. One pragmatic solution is to make it *leaky* by setting a lower bound $\alpha_t = \alpha$. When the lower bound (typically a small fraction) is reached the integrator effectively computes (7) with an exponentially decaying emphasis (time constant $1/\alpha$) on the most recent samples of $\mathbf{X}$.

Continuing to on-line estimations of the other two parameters, substitute $N\hat{\omega}_{k,t}$ from (7) into the denominators of (8) and (9) to obtain

$$\hat{\mu}_{k,t} = (1 - \rho_{k,t})\mu_{k,t} + \rho_{k,t} X_t \tag{11}$$

and

$$\hat{\sigma}_{k,t}^2 = (1 - \rho_{k,t})\sigma_{k,t}^2 + \rho_{k,t}((X_t - \hat{\mu}_{k,t}) \circ (X_t - \hat{\mu}_{k,t})) \tag{12}$$

where

$$\rho_{k,t} = \frac{\alpha_t P(k|X_t, \Phi)}{\hat{\omega}_{k,t}}. \tag{13}$$

α_t is as defined before with or without a lower bound. For each k and at any time t, $\mu_{k,t}$ and $\sigma_{k,t}$ are n-dimensional variables, the same as X_t. These equations differ somewhat from [Stauffer and Grimson 1999] in that (10) to (13) are implemented there with a fixed α for all time—which leads to problems with initialization [KaewTraKulPong and Bowden 2001]— and (13) differs by a factor of $f_{\mathbf{X}}(X_t|k, \theta_k)$—which leads to impractical values for their $\rho_{k,t}$ if implemented directly.

At the cost of calculating $P(k|X_t, \Phi)$ at each frame, the equations (3) to (5), and (10) to (13) may be used as a reasonable implementation of a foreground segmenting algorithm. On each iteration of the algorithm (one iteration per frame), the newly estimated parameters replace the previous estimates: $\hat{\omega}_{k,t} \rightarrow \omega_{k,t+1}$.

3.2 Approximating $P(k|X_t, \Phi)$

The posterior probability $P(k|X_t, \Phi)$ features prominently in the state estimation (3) and all the parameter estimations as the essential quantity that needs to be calculated first. The most significant contribution of [Stauffer and Grimson 1999] to the state of the art may be said to be their fast approximation of $P(k|X_t, \Phi)$.

Instead of calculating $P(k|X_t, \Phi)$ as in (3), they define a *match* as a pixel value falling within $\lambda = 2.5$ standard deviations of the mean of one of the Gaussian distributions. Performance is is not sensitive to the exact number of standard deviations λ. Because [Stauffer and Grimson 1999] uses a single scalar $\sigma_{k,t}$ for all dimensions of $\mathbf{X}$, a slight elaboration is needed here to define a match for an n-dimensional $\sigma_{k,t}$ in terms of squared distance:

$$\begin{aligned} d_{k,t}^T d_{k,t} &< \lambda^2, \\ d_{k,t} &= (\sigma_{k,t}\mathbf{I})^{-1}(X_t - \mu_{k,t}) \end{aligned} \tag{14}$$

The essential approximation to $P(k|X_t, \Phi)$ is then given by $M_{k,t}$:

$$\begin{aligned} M_{k,t} &= \begin{cases} 1 & \text{match} \\ 0 & \text{otherwise} \end{cases} \\ &\approx P(k|X_t, \Phi) \end{aligned} \tag{15}$$

This is motivated by the observation that $P(k|X_t, \Phi)$ is 0 or 1 for most X_t and is near 1 for only one choice of k at a time, as can be seen from Figure 2. It in effect implements a winner-takes-all tactic. In the case of more that one match, the one with the highest peaking distribution (largest $\omega_{k,t}/\sigma_{k,t}$) is selected. Substituting (15) into

(10) to (13) approximately gives the form of algorithm described in [Stauffer and Grimson 1999].

The approximation (15) does accommodate previously unseen pixel values because it allows for the case of no match being found. When there is no match, the lowest-peaking distribution is replaced with a new wide Gaussian centred on the new pixel value. This is the essential mechanism whereby newly stationary objects have the opportunity gradually to get absorbed into the background model. If the new value is transient, the new distribution will fade away in time and be replaced when the next new object appears. There is therefore a lower bound on the useful values of K: at least one spare distribution is needed to model a foreground process.

For (15) to be practically useful, (14) must be significantly faster than calculating $P(k|X_t,\Phi)$ in (3). Unfortunately the full computational benefit of the approximation is not obtained in [Stauffer and Grimson 1999] because they do not define $\rho_{k,t}$ in terms of $P(k|X_t,\Phi)$ in the estimation of $\mu_{k,t}$ and $\sigma_{k,t}$. The following discussion follows through on the logic of the approximation with a significant speed improvement as a result.

4 Alternative approximation

The definition of $\rho_{k,t}$ in (13) differs from the corresponding definition in [Stauffer and Grimson 1999] by a factor of $f_{\mathbf{X}}(X_t|k,\theta_k)$. Substituting (15) in (13) should give

$$\rho_{k,t} \approx \begin{cases} \frac{\alpha_t}{\hat{\omega}_{k,t}} & \text{match} \\ 0 & \text{otherwise} \end{cases} \qquad (16)$$

which entirely avoids calculation of $P(k|X_t,\Phi)$ at the cost of a division by $\omega_{k,t}$. This is a faster and more logical application of the fundamental approximation (15) than that used in [Stauffer and Grimson 1999].

For surfaces occurring with low probability, it is possible that (16) results in a $\rho_{k,t} \geq 1$, which happens when $\omega_{k,t} \leq \alpha_t$. A value of $\rho_{k,t} = 1$ results in an estimated variance of zero in (12). Stauffer and Grimson [1999] use a more computationally demanding definition of $\rho_{k,t}$ in their equation (8) which does not have this problem but which excessively favours frequently occurring distributions compared with rarer ones.

A fast practical implementation of (16) may additionally be able to avoid the division by $\omega_{k,t}$ through, for example, setting $\rho_{k,t} = \alpha_t$, remembering the last matching X_t (zero-order hold) for each distribution, and updating the parameters with (11) and (12) at every frame using the remembered X_t whether it currently matches or not. This simple and computationally efficient procedure accurately implements (11) and (12) assuming the approximation (15) for $P(k|X_t,\Phi)$.

Implementation speed can further be optimized through careful coding of the algorithm to avoid the K square root operations per frame in calculating standard deviations from (12). The match decision (14) and the distribution ranking mentioned in Section 2.3 can both be performed using the variance directly with all other related quantities squared as appropriate.

5 Necessary extensions

Mixture-of-Gaussian algorithms for moving object segmentation are generally not practically usable in their pure form. Their parameters need to be initialized at the outset and re-initialized when illumination changes drastically. (A separate illumination monitor may be needed to detect events like lights switching on or off.) The basic algorithms also produce false detections in the form of noise-generated speckles which have to be removed. Finally, the detected blobs often need post-processing to fill internal holes and gaps, and to be presented in a form useful to the application.

Symbol	Value	Assumption
K	3	
λ	2.5	
α	0.005	5 frames/second
T	0.7	$K = 3$
ω_{init}	0.05	$K = 3$
σ_{init}	30	$X \in [0,255]$

Table 1: Values of parameters in a typical indoor application. The right-most column shows assumptions which affect the parameter value.

The first question to arise during implementation of the algorithm is how best to initialize the Gaussian distributions. One problem is that it is not always possible to obtain an empty background scene. Gutchess [2001] describes a median-filtering solution.

Another difficulty with the original version of the algorithm is that its parameters stabilize too slowly. KaewTraKulPong [2001] addresses this by allowing the update parameter to be initially higher and by omitting to multiply it by the prior probability. The latter step also increases computation speed.

Previous authors have not specified the initial settings of the mean variances and amplitudes of the Gaussians and were also vague about what the variances and the amplitudes of dynamically reset Gaussians should be initialized to. Our tests indicated that the choice was important. Table 1 suggests practical values of the essential parameters for an indoor office surveillance application.

When the algorithm processing is complete, it is essential that the resulting binary segmented image be further refined by standard morphological functions. Because a statistically significant portion of the input samples will lie in the tails of the distributions, the output image will inevitably contain small, noise-generated blobs. These have to be eliminated through simple area thresholding, whilst shape-based filtering can readily remove other false blobs. The remaining blobs can then be cleaned with a hole-filling algorithm.

6 Optional extensions

We added a new feature to the algorithm by borrowing the concept of hysteresis thresholding from its direct use on grey level images. Whereas Stauffer and Grimson used a single threshold for the number (λ) of standard deviations away from the mean for a match, we observed that tokens resulting from this contained holes and gaps, the incidence of which was significantly reduced by dual thresholding. Thus an upper (λ_+) and a lower ($\lambda\_$) value of λ are set and these operate as follows:

If the normalized pixel deviation is less than $\lambda\_$, the foreground is set at 'yes'. If the normalized pixel deviation is greater than λ_+ the foreground is set at 'no' (i.e. background). In-between pixels are given a preliminary 'maybe' status. A morphological function then allows those 'maybe' pixels to be deemed foreground if it finds that they are 8-connected to a 'yes' pixel. Otherwise they are deemed to be background.

As our video footage (in the CD-ROM version of this paper) indicates, further morphological processing was tested which attempts to detect the tops of the heads of human subjects, but this is outside the scope of this paper.

7 Conclusion

This paper showed how an implementation of a background mixture model for video segmentation can be assembled from an understanding of the underlying theory. It listed all the essential model parameters and their typical values as well as the extensions that are necessary for practical use of the algorithm.

The value of this work is in providing theoretical tools with which to modify or adapt the original algorithms for either better performance or higher speed, and in giving the bridging information needed for rapid implementation.

References

BILMES, J. 1998. A gentle tutorial on the EM algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. Tech. Rep. ICSI-TR-97-021, University of California Berkeley.

DEMPSTER, A., LAIRD, N., AND RUBIN, D. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B 39*, l–38.

GUTCHESS, D., M, T., COHEN-SOLAL, E., LYONS, D., AND JAIN, A. K. 2001. A background model initialization algorithm for video surveillance. In *Eighth International Conference on Computer Vision*.

KAEWTRAKULPONG, P., AND BOWDEN, R. 2001. An improved adaptive background mixture model for real time tracking with shadow dectection. In *Proc. 2nd European Workshop on Advanced Video-Based Surveillance Systems*.

MCIVOR, A. 2000. Background subtraction techniques. In *Proceedings of Image & Vision Computing New Zealand 2000 IVCNZ'00*, Reveal Limited, Auckland, New Zealand.

STAUFFER, C., AND GRIMSON, W. 1999. Adaptive background mixture models for real time tracking. In *Computer Vision and Pattern Recognition*.