Statistical Background Modelling For Tracking With A Virtual Camera

Simon Rowe and Andrew Blake
Department of Engineering Science, University of Oxford
Parks Road, OX1 3PJ

Abstract

A method of robust feature-detection is proposed for visual tracking with a pan-tilt head. Even with good foreground models, the tracking process is liable to be disrupted by strong features in the background. Previous researchers have shown that the disruption can be somewhat suppressed by the use of image-subtraction. Building on this idea, a more powerful statistical model of background intensity is proposed in which a Gaussian mixture distribution is fitted to each of the pixels on a "virtual" image plane. A fitting algorithm of the "Expectation-Maximisation" type proves to be particularly effective here. Practical tests with contour tracking show marked improvement over image subtraction methods. Since the burden of computation is off-line, the online tracking process can run in real-time, at video field-rate.

1 Introduction

This paper presents a statistical treatment of background modelling for use in visual curve trackers. The new methods are tested using a real-time tracker based on snakes deforming over time [15, 9, 3], represented by B-spline curves [18, 7]. The tracker runs at video field rate (50Hz). The background modelling technique described here is not restricted to curves; it could also be applied to real-time trackers based on polygons or other geometrical representations [23, 14, 17]. Some tracking applications, surveillance for instance, call for a panoramic field of view which can be achieved by a pan-tilt head [8, 22, 21, 6]. Such a head is used in the experiments reported here.

A major problem in achieving robust curve tracking is the distracting effect of background objects—clutter. Strong features in the background compete for the attention of the tracked curve and may eventually succeed in pulling it away from the foreground object. This effect is clearly visible in figures 1(a)–(d). Immunity to distraction can be enhanced by both by modelling of the foreground and of the background. A foreground model may include a template, object dynamics [25, 11] and intensity profiles for certain object features [26, 10]. This paper deals with modelling the background. It develops a statistical model of the distribution of intensities at *each* point in the background, which can then be used to discriminate the foreground object from the background. The model is applied to an image stream taken from a video camera mounted on a pan-tilt head—a situation where the normal technique of image differencing has problems. These techniques only

BMVC 1995 doi:10.5244/C.9.42

apply to rotating cameras on a static mounting–they will not work when the camera is mounted on a moving vehicle.

It should be noted that there are slight differences between the image sequences used to compare results in this paper, this is due to the tests being performed on *live* data, as it is currently impossibly to record sequences when the camera is mounted on a rotating head. However the sequences are representative examples showing typical behaviours.

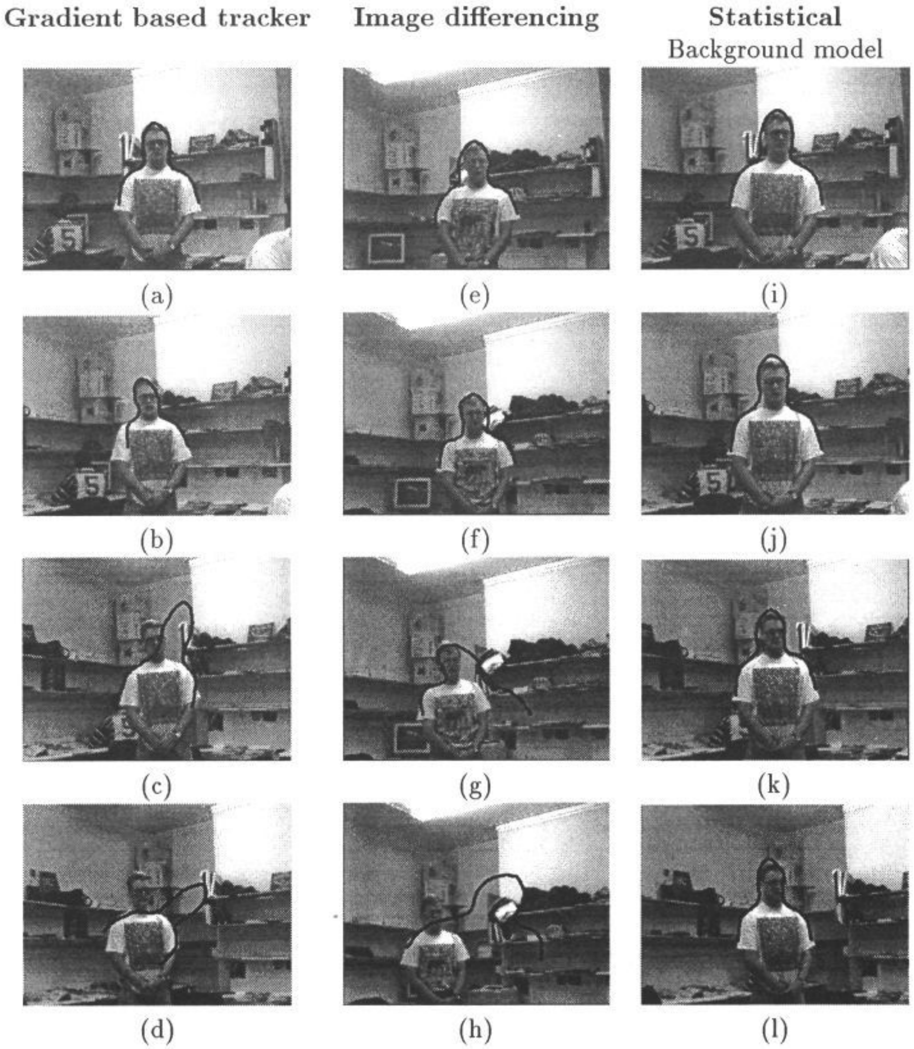
Our aim here is to analyse the background for an image stream taken from a video camera mounted on a moving pan-tilt head. Murray and Basu [20] have shown that image differencing can be applied to the case of a moving camera if suitable compensation is applied to allow for inter-frame head-motion. Of course motion compensation cannot be exact because the depth of background objects is unknown, and since the head does not rotate precisely about the camera's optical centre, parallax errors are introduced. Then successive images can be subtracted, though additional morphological filtering is required to suppress parallax error. Even so, some errors remain, either spurious features belonging to the background or lost foreground features.

This paper presents an alternative method of background modelling for a totally static background. It shifts the burden of processing from on-line filtering to off-line analysis, so that the entire tracking process continues to be feasible with the processing power of a modest desktop workstation (SUN SPARC 2). The entire background is observed in repeated sweeps of the camera head, over an extended period. Intensity values are mapped from the physical camera plane onto a *virtual* plane where they are not merely averaged — histograms of intensities values are accumulated from which, in due course, simple Gaussian mixture models are estimated for the probability distribution of intensities at *each* pixel in the virtual image. For computational feasibility a modest degree of subsampling may be necessary. The estimated probability distributions then encompass intensity variability arising for each of a number of reasons, for instance: parallax errors, errors due to sensor noise, miscalibration and subsampling, and errors arising at a higher level from the uncompensated residue of illuminant variations and even those from cast shadows. Once such compensation is incorporated, our curve tracker becomes markedly more immune to distraction as figure 1 shows, and performance surpasses considerably what is attainable by simple background subtraction.

2 The tracking process

Our test task of curve tracking follows the method of Blake *et al* [5] and consists of a quadratic B-spline curve curve $(x(s), y(s))$ stabilised by a template curve $(\overline{x}(s), \overline{y}(s))$. Limited shape deformations of $(x(s), y(s))$ are allowed relative to the template and 2D Euclidean transformations are allowed to occur over time relatively freely. These dynamical constraints are used in the predictor of a Kalman filter [13] which constitutes the curve tracker.

The tracker is driven by a measurement process in which normal vectors to the tracked curve are constructed. The tradational tracker, performs one dimensional edge-searches along normals attempt to locate contrast features on the foreground object. When a candidate feature possessing plausible contrast is found, its posi-



Gradient based tracker **Image differencing** **Statistical** Background model

(a) (e) (i)

(b) (f) (j)

(c) (g) (k)

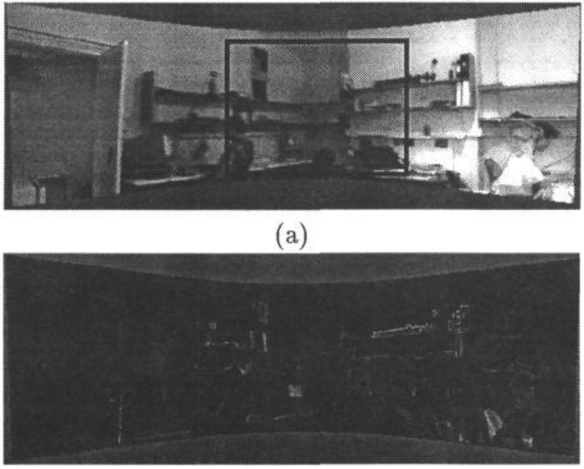
(d) (h) (l)

Figure 1: *In the left-hand sequence ((a) - (d)), a gradient-based feature detector is used to track a target as it moves across a room. The camera is mounted on a pan-tilt head. Because the foreground is fixated it appears stationary, but note how the background moves relative to the target. As the target passes some strong clutter the contour is distracted ((c) and (d)) and loses track of the target. The middle column((e)-(h)) shows a similar tracking sequence, using image differencing. The contour is again distracted by the edges of objects in the background and loses track of the target (h). Finally, in the right hand column, the background has been modelled statistically on the virtual image plane. Edges in the background are ignored (k) enabling tracking to continue past the clutter (l).*

tion is added into the curve's estimated position and shape. It is at this juncture that background features may accidentally achieve a match and distract the tracking curve. This paper suggests an approach whereby the edge-detector is replaced by a statistical test to determine if each point on each searchline is foreground or background. The boundary between the foreground region and the background is then used as the *feature* in the measurement process.

2.1 The virtual camera

Rather than use the image on the moving camera directly, the active contour works from the image on a static, virtual camera. This camera is a single mathematical plane fixed in the world-frame onto which a physical image can be projected, in a manner somewhat akin to the recently developed technique of "image mosaicing" [24]. Ideally the centre of projection of the camera should coincide with the centre of rotation of the camera-head. In that case, for a given pan/tilt position, image pixels are projected along rays passing through the centre of projection, from the physical image onto the virtual one. Note that a single virtual plane is sufficient where the union of all physical fields of view is contained within a hemisphere (otherwise several planes are required, forming a chart for the sphere). In practice there is some small misalignment of the two centres so that the projection process involves parallax errors, typically of a few mrad. The result is a panoramic image on the virtual image plane in which the parallax errors appear as blur, and this is shown in figure 2. The crucial point is that the (mean) image is accompanied by an overlaid probability distribution. In the simplest case this is a map of the *variance* of intensity, as shown in figure 2b.



(a)

(b)

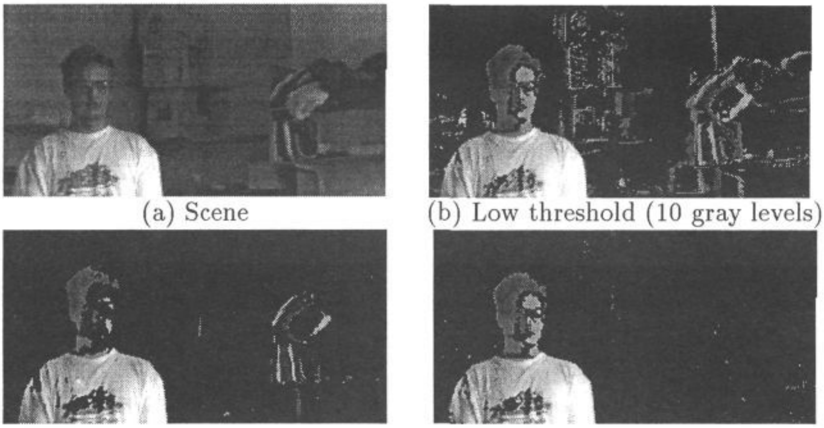
Figure 2: *The virtual image plane. The image in (a) was obtained from a physical camera mounted on a pan and tilt head, mapping its image onto the virtual image plane as it is swept round the room. The instantaneous field of view of the physical camera is shown as the black rectangle in the image. Calibration errors in the system mean that the image is slightly blurred. Although the apparent effect of blur is small, it is significant for background modelling because of the consequent variability of intensity I. The variance (b) of intensity over the virtual image is particularly great where ∇I is large (i.e. at edges).*

The curve tracker now runs on the virtual rather than the physical image and this allows tracking to continue as if on a static camera with a very wide field of view, but with the advantage of high resolution. Working in virtual camera coordinates means that the tracking process is quite decoupled from the effects of pan and tilt. In fact the controller for the position of the pan-tilt head can be quite slow and inaccurate, provided it is just agile enough to retain the foreground object within the field of view of the physical camera. A standard "Proportional-Integral-Derivative" (PID) controller [2] is quite sufficient. The head itself may then have substantial tracking lag, but this has no effect whatever on the curve tracker because the mapping from the physical to the virtual plane is computed using positional feedback signals directly from encoders on the motor shafts. Of course these encoders must be sufficiently fast and accurate but in practice such devices are routinely available. (Note that the physical camera must be calibrated, at least approximately, relative to the head.) This arrangement parallels the situation in animal vision in which slow head movements can be compensated by good proprioception, via the "vestibulo-ocular reflex" [1].

3 Background intensity variations

A number of researchers have used "image-differencing" to increase the robustness of tracking [19, 4, 16]. This uses a simple model of the background in which its mean intensity is represented as an image. Off-line estimation of this mean can be made robust to occasional moving objects by using a suitable filter — the median filter for instance. Once the mean image is obtained it is stored for repeated on-line subtraction from images in the incoming stream. This tends to cancel out background features, leaving features on moving foreground objects prominently exposed. A global threshold is applied to this differenced image to determine whether a point is part of a foreground object, or part of the background. Unfortunately, simple image differencing and thresholding has a somewhat limited power for rejection of spurious background clutter. This limitation is even more severe when there is additional variation introduced by viewing from a rotating camera. The limitations of a simple scheme like this are shown below in figure 3. In order to develop a system to discriminate foreground from background by using a model of the background, it is useful to think about the sources of variability in intensities of the background points. These sources include: sensor noise, shadows and inter-reflection, parallax errors in mapping between the physical and virtual image planes and mapping errors due to the sub-sampling of the virtual plane needed to reduce physical memory requirements.

In some cases, illumination variation for example, partial compensation for error is possible (via AGC or other means), leaving only a residual uncompensated error to be modelled statistically. In other cases such as parallax error, the entire error is accounted for by the statistical model. It is not assumed that the errors are small—indeed errors due to shadow-casting, for instance, may be considerable. Given that the system is going to be exposed to, and must tolerate, such gross errors, the pressure is removed for accurate camera calibration of the head/camera. Approximate calibration is sufficient since any residual error will be absorbed into the statistical model. The intensity error due to mis-calibration and non-central mounting of the head will be approximately $\underline{a}.\nabla I$ where $\underline{a}$ is a gradient coefficient related to the current position of the head and the camera offset, and I is the



(a) Scene (b) Low threshold (10 gray levels)

(c) High threshold (27 gray levels) (d) Statistical based threshold

Figure 3: *This figure shows the result of using various thresholds to compare a scene with a head and shoulders target (a) to a reference scene without the target. The intensities from (a) are used to show areas differing by more than the threshold from the reference image. (b) shows the result of image differencing using a low threshold, note how edges are segmented along with the target. (c) shows the result of using a higher threshold, note how some edges still remain, even though quite a lot of the target has now been classified as background. Finally (d) shows the result of using individual, statistical based thresholds. Note how the target is better segmented with this approach than with the image differencing approach.*

intensity field in the scene. The effect of this error will only be seen in areas of the scene where ∇I is high—such as the edges of objects in the background, and this is effect is clear in figure 2b.

4 Fitting to a Normal distribution

The simplest reasonable model of the intensity variation at a single pixel is a univariate normal distribution. It can be obtained by estimation of mean and variance in the usual way. Given a training set, consisting of a set of N readings $\mathbf{z} = [z_1, z_2 ... z_N]$. The mean μ and variances σ^2 are given by $\mu = \frac{1}{N}\sum_{i=1}^{N} z_i$ and $\sigma^2 = \frac{1}{N-1}\sum_{i=1}^{N}(z_i - \mu)^2$.

In many cases the univariate normal is an adequate model. Unfortunately, in practice the data tends can be disrupted by foreground objects moving during data-collection and this calls for a fitting method that is robust to outliers. However, such intensity histograms can be modelled by a single, dominant Gaussian together with a contaminating distribution. The centre of the Gaussian can be located as the mode of the histogram and an initial estimate of the standard deviation of it obtained as the width at half the modal frequency. With that initial estimate of σ, the relative proportions α,and $1 - \alpha$ of contaminant lying to the left and right of the Gaussian peak can be estimated. A variable proportion η of the data can then be trimmed, $\alpha\eta$ from the left tail and $(1 - \alpha)\eta$ from the right tail. As the trim-level varies, a χ^2 test detects when the remaining data is an uncontaminated Gaussian.

Unfortunately, the trimming removes not only the contaminating dataset, but also the *tails* of the Gaussian. This will mean that the σ^2 calculated above will actually underestimate the Gaussian's variance slightly. A solution to this problem is to use an Expectation-Maximisation , or EM, algorithm—the estimates obtained by the above method are used as the starting point for the EM estimation.

Expectation-Maximisation [12] is a technique for obtaining a maximum likelihood estimate (MLE) of a family of model parameters given some incomplete (or trimmed) observed data. It is essentially an iterative two stage technique. In stage one, the **E**xpectation step, sufficient statistics are estimated based on the observed data. In stage two, the **M**aximisation step, takes this estimate of the sufficient statistics and estimates the model parameters by maximum likelihood as though complete data were observed. A more complete explanation of the general EM algorithm is given by Dempster [12].

The derivation of an EM estimation scheme for fitting a single Gaussian, based on trimmed data, is not presented here due to space restrictions, however it leads to the update equations: $\mu_{i+1} = \frac{1}{N(1+q)}\left(\sum_{n=1}^{N} x_n + Nq\mu_i\right)$ and $\sigma^2_{i+1} = \frac{1}{N(1+q)}\left(\sum_{n=1}^{N}(x_n - \mu_{i+1})^2 \ + Nq(\sigma_i^2 + \mu_{i+1}^2)\right)$ where μ_i is the ith estimate for the mean of the distribution, and σ_i^2 for its variance. The dataset consists of N measurements of intensity $x_1 ... x_N$, and q is a scale factor related to the area of the trimmed tails of the distribution.

The iterative application of these equations will converge [12] onto an unbiased, MLE of μ and σ for the gray level distribution for a point. In order to have fast convergence to the correct answer in the presence of clutter, it is essential to have a good initial estimate of both μ and σ. Such an estimate could be obtained by using the repetitive trimming technique described earlier.

4.1 Limitations of the single Gaussian model

The single Gaussian fails to do justice to the underlying distribution near high-contrast edges, as figure 4 shows, but a two-Gaussian mixture would appear to be adequate. Either the trimming technique or the single Gaussian EM algorithm might be expected, at best, to converge to one of the Gaussian's. The remaining un-modelled Gaussian will generate false foreground features and cause the tracker to stick on the background clutter.

The single Gaussian model is also inadequate when the foreground and background interact with each other — when the target casts a shadow on the background for instance. In this case points in the background can be expected to have two intensity distributions associated with them - one for direct illumination and one from the ambient illumination. This means that the PDF for the point will again comprise two separate Gaussian's.

5 Fitting a two-Gaussian mixture

The modelling problem is now expanded to fit the two Gaussian mixture to the intensity data. This data may still have been contaminated by a moving forground object while it was collected, and the fitting technique must be robust against this. For this reason we would still like to trim the dataset collected by $\pm\lambda\sigma$. Both approaches mentioned in above in section 4 can be applied to this problem with

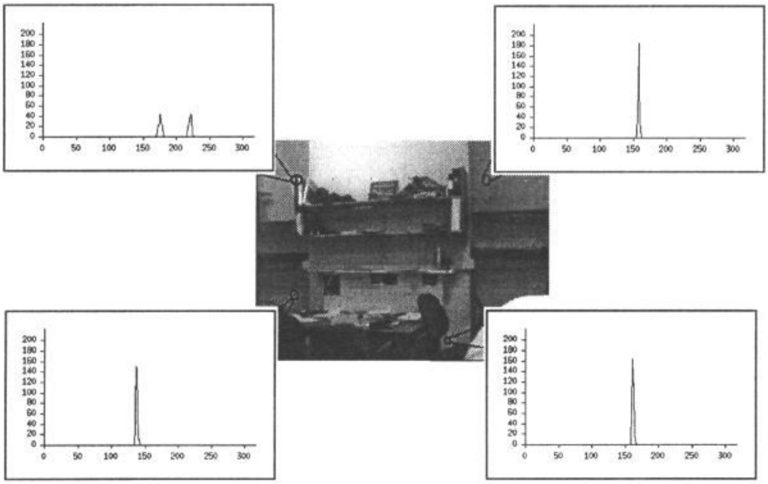


Figure 4: *Intensity histograms for points on the image plane. The graphs show the frequency of occurrence for a particular gray level for a particular pixel over 500 frames. The image has been sub-sampled by mapping each* 2×2 *pixel block onto a single point. Near an edge a two Gaussian mixture will be necessary to model the intensity as can clearly be seen from the upper left graph. Note also how the widths of the distributions are different in different parts of the image.*

only slight modification.

The *Trim and Fit* algorithm is applied by first finding the largest Gaussian in the data as a single Gaussian model, exactly as in section 4. Then the training data can be altered to compensate for this distribution, and the estimate-trim-fitting algorithm re-applied to the altered dataset. The compensation algorithm involves subtracting the expected intensity histogram of the fitted gaussian from the measured intensity histogram. The second normal distribution in the measured dataset can then be estimated from this compensated dataset in the same way as the first one.

Of course this Trim and Fit algorithm still suffers from the same problems when applied recursively to the two Gaussian case as it does when applied to the single Gaussian case, namely in underestimating the variance of the distributions. This approach may also suffer from problems when two Gaussians overlap, since it is not taking their interactions into account properly. Both these problems can be eliminated by using an properly formulated EM algorithm.

An EM algorithm can also be formulated for the case of a two Gaussian mixture. Again, the derivation is not shown here for brevity, however it can be sucessfully applied to this problem. An example showing the result of applying the two Gaussian EM algorithm is given in figure 5.

5.1 The use of a two Gaussian mixture model

It can be shown that wrongly fitting a single Gaussian model to a two Gaussian distribution (without trimming) results in the fitted Gaussian covering a much larger area than the two component Gaussians. This means that in situations

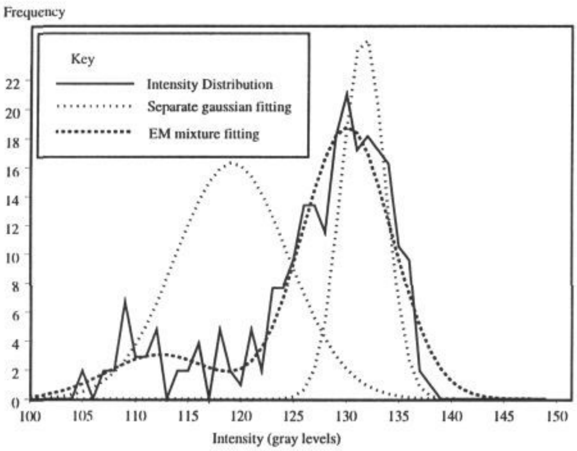


Figure 5: *Use of EM for fitting a two Gaussian mixture. Note how the distribution fitted by the algorithm is much closer to the underlying distribution that that obtained by using the trim and fit algorithm (indeed the trim and fit algorithm was only able to find one correct Gaussian in this case, the other one displayed is the closest approximation to a Gaussian that it could find).*

where the underlying intensity distribution for a large proportion of the image is a two component Gaussian mixture, a tracker which utilises this correct two Gaussian mixture model will be much more sensitive and track significantly better than one based around a one Gaussian distribution.

Unfortunately, correctly fitting a two Gaussian model to points in the image takes a long time — the current implementation on a Sun Sparc IPX, takes of the order of 1 second for each point. In a typical static image sub-sampled by a factor or 16, there are still of the order of 25000 points, meaning that it will take approximately 7 hours to fit a two Gaussian model to them. Fortunately however, not all points need a two Gaussian mixture to represent their intensity distribution, in typical examples less than 8000 of the points require the more complex model, meaning that this model can be learnt in under 2.5hours (contrasting with about 1 minute for the single gaussian model). The points requiring the more complex model can be automatically detected by examining the variance that they have when fitted by the simpler model—points where the simple model fails tend to have large variances. There may however be situations when this long start-up time is perfectly acceptable, such as a security camera looking down a corridor night after night. Certainly as the computational speed of computers increases, this time will become negligible for more and more cases.

A further problem when attempting to fit a background distribution both in direct lighting and in shadow is that in normal situations the shadow may only be present for a small, but highly significant, proportion of the time. This can make collecting representative background data difficult unless it is done by deliberately casting shadows onto the background without allowing the foreground object to appear in the image too often. The result is that the background modeller is forced to model the intensity variability due to shadows, but the foreground object appears only as a contaminant and is not modelled.

6 Discussion

Extensions have been proposed to improve and extend the tracking ability of active contours so that they can successfully robustly track in a wider range of applications. Use of a virtual image-plane has been proposed, enabling an active contour tracker designed for a static camera to operate transparently with a pan/tilt head. Results have been shown for a hard tracking sequences which demonstrate the improvements in tracking performance possible by statistically modelling the distributions of points in the background.

Future work will address more efficient ways to fit the background model to the intensity distribution. An interesting possibility, worthy of investigation, is to extend the statistical modelling the background beyond modelling intensity to include also the gradients of the intensity field — both in space and in time.

References

[1] D.B. Arnold and D.A.. Robinson. *Phil. Trans. R. Soc*, 337:327–330, 1992.

[2] K. J. Astrom and B. Wittenmark. Addison Wesley, 1984.

[3] N. Ayache, I. Cohen, and I. Herlin. In A. Blake and A. Yuille, editors, *Active Vision*, pages 285–302. MIT, 1992.

[4] Adam Baumberg and David Hogg. In Jan-Olof Eklundh, editor, *Computer Vision - ECCV '94*, volume Volume I, pages 299 – 308. Springer-Verlag, 1994.

[5] A.Blake R.Curwen and A. Zisserman. *Int. Journal of Computer Vision* 11(2), 1993.

[6] C.M. Brown, D. Coombs, and J. Soong. In A. Blake and A. Yuille, editors, *Active Vision*, pages 123–136. MIT, 1992.

[7] R. Cipolla and A. Blake. In *Proc. 3rd Int. Conf. on Computer Vision*, pages 616–625, 1990.

[8] J.J. Clark and N.J. Ferrier. In *Proc. 2nd Int. Conf. on Computer Vision*, pages 514–522, 1988.

[9] Laurent D. Cohen and Isaac Cohen. In *Proc. 3rd Int. Conf. on Computer Vision*, pages 587–591. IEEE Computer Society Conference, December 1990. Osaka, Japan.

[10] T.F. Cootes, C.J. Taylor, A. Lanitis, D.H. Cooper, and J. Graham. In *Proc. 4th Int. Conf. on Computer Vision*, pages 242–246, 1993.

[11] R. Curwen, A. Blake, and A. Zisserman. In *Computer Vision - ECCV '92*, pages 879–883, 1992.

[12] A.P. Dempster, M.N. Laird, and D.B. Rubin. *J. R. Stat. Soc.*, B 39:1–38, 1977.

[13] Arthur Gelb, editor. MIT Press, Cambridge, MA, 1974.

[14] C. Harris. In A. Blake and A. Yuille, editors, *Active Vision*, pages 59–74. MIT, 1992.

[15] M. Kass, A. Witkin, and D. Terzopoulos. In *Proc. 1st Int. Conf. on Computer Vision*, pages 259–268, 1987.

[16] Dieter Koller, Joseph Weber, and Jitendra Malik. In Jan-Olof Eklundh, editor, *eccv94*, volume Volume I, pages 189 – 196. Springer-Verlag, 1994.

[17] D.G. Lowe. *Int. Journal of Computer Vision*, 8(2):113–122, 1992.

[18] S. Menet, P. Saint-Marc, and G. Medioni. In *Proceedings DARPA*, pages 720–726, 1990.

[19] M.Kilger. In *IEE 4th International Conference on Image Processing and its applications*, 1992.

[20] Don Murray and Anup Basu. *IEEE Trans. Pattern Analysis and Machine Intell.*, 16(5):449–459, May 1994.

[21] D.W. Murray, F. Du, P.F. McLauchlin, I.D. Reid, P.M. Sharkey, and M. Brady. In A. Blake and A. Yuille, editors, *Active Vision*, pages 303–336. MIT, 1992.

[22] K Pahlavan and J-O Eklundh. Technical Report CVAP-80, Dept of Numerical Analysis and Computing Science, Royal Inst of Tech, Stockholm, 1991.

[23] G.D. Sullivan. *Phil. Trans. R. Soc. Lond. B.*, 337:109–118, 1992.

[24] R. Szeliski. Technical report, Digital Equipment Corporation, Cambridge, USA, 1994.

[25] D. Terzopoulos and R. Szeliski. In A. Blake and A. Yuille, editors, *Active Vision*, pages 3–20. MIT, 1992.

[26] A. Yuille and P. Hallinan. In A. Blake and A. Yuille, editors, *Active Vision*, pages 20–38. MIT, 1992.