An Adaptive Visual System for Tracking Low Resolution Colour Targets

P KaewTraKulPong[1,2] and R Bowden[1]
[1]Vision and Virtual Reality group, Department of Systems Engineering, Brunel University, Middlesex, UB8 3PH, UK.
[2]King Mongkut's University of Technology Thonburi, Bangmod, Toong-kru, Bangkok, 10140 Thailand.

Abstract

This paper addresses the problem of using appearance and motion models in classifying and tracking objects when detailed information of the object's appearance is not available. The approach relies upon motion, shape cues and colour information to help in associating objects temporally within a video stream. Unlike previous applications of colour in object tracking, where relatively large-size targets are tracked, our method is designed to track small colour targets. Our approach uses a robust background model based around Expectation Maximisation to segment moving objects with very low false detection rates. The system also incorporates a shadow detection algorithm which helps alleviate standard environmental problems associated with such approaches. A colour transformation derived from anthropological studies to model colour distributions of low-resolution targets is used along with a probabilistic method of combining colour and motion information. This provides a robust visual tracking system which is capable of performing accurately and consistently within a real world visual surveillance arena. This paper shows the system successfully tracking multiple people moving independently and the ability of the approach to recover lost tracks due to occlusions and background clutter.

1 Introduction

This paper addresses the problem of using appearance and motion models in classifying and tracking low-resolution colour objects. Instead of relying upon a motion model and maintaining multiple hypotheses, simple shape and colour information can be useful in data association so fewer hypotheses need be maintained. However, the number of pixels constituting an object can be too small to be able to build a reliable appearance model in complex shape or colour tracking systems. An example of targets used by most authors is shown in figure 1a while our method monitors targets in the scene similar to figure 1b. In such cases, the colour distribution learning from the scene is deemed unreliable due to limited number of examples obtained from the scene. The number of pixels supporting the object is too few to train a complex shape or colour model. As the model becomes more complex, the number of required training samples increases exponentially. This is called the '*curse of dimensionality*' in the field of pattern recognition [1]. This makes it unreliable to classify or track objects by their shape or colour alone. Under these circumstances, most systems treat the colour

BMVC 2001 doi:10.5244/C.15.26

information as unreliable and use motion cues and simple shape features alone to track objects [2-6]. This paper utilises both simple shape and motion along with a colour model based on transformations derived from psychophysical studies [7, 8].

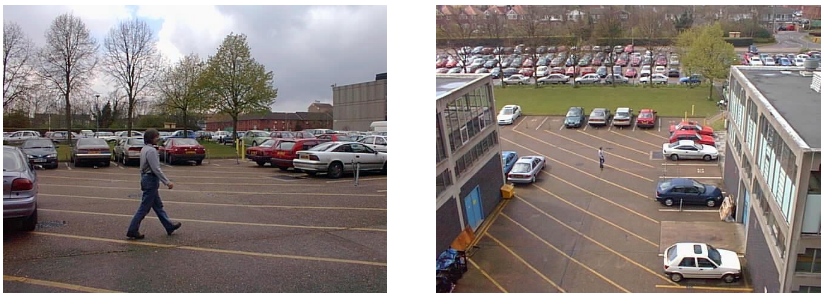

Figure 1: (a) scene with a high-resolution target and (b) scene with a low-resolution target

1.1 Related Work

The work of tracking objects by appearance has been performed using various image cues. Colour, motion, shape, depth and so on are the common appearance modalities used in such work. This information constitutes attributes of the object being tracked.

Birchfield uses histogram and histogram intersection introduced by Swain [9] to model and track human heads [10]. The colour model consists of a histogram with eight bins for chrominance elements, (*G*-*R*) and (*B*-*G*) and four bins for the luminance (*R*+*G*+*B*). Bradski tracks a person's face to form part of a perceptual user interface [11]. The model is built by sampling skin-coloured pixels, converting to HSV colour space and making a histogram of their hues if their saturations and values are greater than a predefined threshold. The Pfinder system [12] models parts of a person's body with Gaussian distributions called *blobs*. The coordinate of the blob centroid and YUV colour components are encoded in this model. McKenna *et al* [13-16] use a mixture of Gaussian distributions to model and track a multi-colour object. At each time step, the model is fitted by semi-parametric learning based on a cross-validation method. The training and validation sets are sampled hues and saturations of pixels with values within a predefined band. The current estimate is calculated from an *L* most recent window of the results from *L* individual runs. W4 [17] uses relatively complex shape and intensity to track people, identify poses and separate individuals during occlusion. Koller *et al* [18] use contour and camera placement assumptions to track objects during occlusion. The tracking process in these systems is done by proximity or simple 2-D motion models and a similarity measure of object appearance. These systems require a high-quality representation or a reasonably large number of pixels on the targets, for example, at least 250 pixels per blob are required W4. On the other hand, systems like [2, 4, 5] and [6] employ simple shape information and a motion models to classify and track multiple objects. Moving objects are classified into people, cars and other objects. Since, the camera has a wide-angle lens and is mounted at a great distance from most targets, the extracted objects in this situation have low resolution and a small number of supporting pixels.

This paper is organised as follows: Section 2 describes the system components. Section 3 outlines the object detection module. Section 4 illustrates the object tracking module. Experimental results are presented in Section 5, followed by a conclusion in Section 6.

2 System Overview

The system consists of two parts, object detection and object tracking modules as show in figure 2. The object detection module deals with detecting moving objects from a stationary scene, eliminating shadows and removing spurious objects. The object tracking module takes detected objects from the current frame and matches them to those at the last time step. It also maintains appearance and motion models of the objects to facilitate the matching process as well as eliminate spurious trajectories. The outputs of this module are object trajectories that exhibit temporal and spatial consistency.

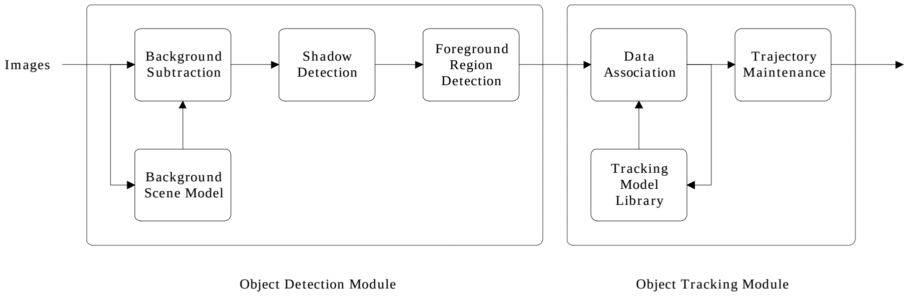

Figure 2: Outline architecture of our system.

3 Object Detection Module

The module for object detection consists of three parts. First, each pixel in the input image is segmented into moving regions by a background subtraction method. The background subtraction uses a per pixel mixture of Gaussians to compare with the current image. The outcome is fed into shadow detection module to eliminate shadows from moving objects. The resulting binary image is then grouped into different objects. The false detection from the backgroud subtraction are also removed in this step.

3.1 Background Modelling

This section discusses the background model used by our method. The model is similar to the one introduced by Grimson and Stauffer [4, 5]. The difference lies on the update equation of the model parameters. In previous work we have demonstrated the superior performance of update equations derived from *sufficient statistics* and *L*-recent window formula over other approaches [19, 20]. This provides a system which learns background scene faster and more accurately than that of Grimson *et al*. Each pixel in the scene is modelled 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. 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 wider clusters due to different reflecting surfaces during the movement. The measure of this is called the *fitness* value. To allow the model to adapt to changes in illumination and run in real-time, an update scheme is applied. Every new pixel value is checked against existing model components in order of fitness. The first matched model component will be updated with the new observation. If it finds no match, a new Gaussian component will be added with the mean at that point, a large covariance matrix and a small value of weighting parameter. To maintain a fixed number of mixture components, the component with least fitness is replaced by this new component if there is no free component available.

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

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

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

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

where $\boldsymbol{\mu}_k$ is the mean and $\boldsymbol{\Sigma}_k = \sigma_k^2 \mathbf{I}$ is the spherical covariance of the k$^{\text{th}}$ component.
The K distributions are ordered based on the fitness value w_k/σ_k and the first B distributions are considered models of the background scene at that point. B is found by

$$B = \arg\min_b \left(\sum_{j=1}^{b} w_j > T \right) \tag{3}$$

The threshold T is the minimum fraction of the background scene. 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,

$$\hat{w}_k^{N+1} = \hat{w}_k^N + \alpha^{N+1}\left(\hat{p}(\omega_k \mid \mathbf{x}^{N+1}) - \hat{w}_k^N\right) \tag{4}$$

$$\hat{\boldsymbol{\mu}}_k^{N+1} = \hat{\boldsymbol{\mu}}_k^N + \alpha^{N+1}\left(\mathbf{x}^{N+1} - \hat{\boldsymbol{\mu}}_k^N\right) \tag{5}$$

$$\hat{\boldsymbol{\Sigma}}_k^{N+1} = \hat{\boldsymbol{\Sigma}}_k^N + \alpha^{N+1}\left((\mathbf{x}^{N+1} - \hat{\boldsymbol{\mu}}_k)(\mathbf{x}^{N+1} - \hat{\boldsymbol{\mu}}_k)^T - \hat{\boldsymbol{\Sigma}}_k^N\right) \tag{6}$$

where $\hat{p}(\omega_k \mid \mathbf{x}^{N+1}) = \begin{cases} 1 & ;\text{if } \omega_k \text{ is the first match Gaussian component} \\ 0 & ;\text{otherwise} \end{cases}$

and $\alpha^{N+1} = \max(1/(N+1), 1/L)$.

ω_k is the k$^{\text{th}}$ Gaussian component and $1/L$ defines the time constant which determines change.

3.2 Shadow Detection

As a requirement to identify moving shadows, we need to consider a colour model that can separate chromatic and brightness components. It should also be compatible and make use of our mixture model. This can be done by comparing non-background pixels against the current background components. If the difference in both chromatic and brightness components are within some threshold, the pixel is considered as a moving shadow. We use an effective computational colour model similar to the one proposed by Horprasert *et al* [21] 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 $a = \arg\min_{z} (I - zE)^2$ 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 a 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 $\tau < c < 1$.

3.3 Foreground Region Detection

Although we have maintained a good model for the background scene, problems of camera jitter, swaying tree, camouflage and occlusions can not be eliminated by background subtraction alone. By analysing the characteristics of these events, we can lessen the problems.

- Camera noise normally appears to be regions with small areas. These can be eliminated by applying morphological operations or simply discarding objects of a small number of supporting pixels.
- The multi-colour background model can reduce errors resulting from Camera jitter, swaying tree and spectacularity. However, if some part of the background (a tree branch for example) moves to occupy a new pixel, but there was not part of the model for that pixel, then false detection occurs. These problems share the same characteristic that is the detected objects have long narrow or hollow shapes. They can be eliminated by ruling out objects with small ratio of area to the boundary.
- Camouflage and occlusion from stationary objects give similar results. Parts of object will not be detected. These problems are handled by the trajectory maintenance module.

4 Object Tracking Module

4.1 Motion Model

In our system, Kalman filters are used to track multiple objects based on measurement of their position, motion, simple shape features and colour contents. The coordinates of the object's centroid and the minimum bounding box of the object are modelled by a discrete-time kinematic model. The centroid is modelled by a white noise acceleration model while the height and width of the bounding box are modelled by a white noise velocity model. This is to correspond with the assumptions that objects move with a constant velocity and the bounding box does not change extensively. The object maneuvering ability and the change in shape are encoded in the process covariance matrix as will be described below. The state equation for x-coordinate centroid derived from the piecewise-constant white acceleration model [22] is

$$\mathbf{x}(k+1) = \mathbf{F}(k)\mathbf{x}(k) + \mathbf{M}(k)v(k) \tag{7}$$

where

$$\mathbf{x}(k) = \begin{bmatrix} x(k) \\ \dot{x}(k) \end{bmatrix}, \mathbf{F}(k) = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}, \mathbf{M}(k) = \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \end{bmatrix}.$$

$\mathbf{x}(k)$, $\mathbf{F}(k)$ and $\mathbf{M}(k)$ are state, state transition matrix and vector gain multiplying scalar process noise of x coordinate at time k. the process noise $v(k)$ a zero-mean acceleration sequence and Δt represents the time difference between frame k+1 and k.
The measurement equation for this coordinate is

$$\mathbf{z}(k+1) = \mathbf{H}(k+1)\mathbf{x}(k+1\,|\,k) + \mathbf{w}(k+1) \tag{8}$$

with the measurement matrix, $\mathbf{H}(k+1) = [1 \quad 0]$ and $\mathbf{w}(k+1)$ a zero-mean measurement noise sequence.
The process noise covariance and measurement variance are as follows

$$\mathbf{Q}(k) = \begin{bmatrix} \frac{1}{4}\Delta t^4 & \frac{1}{2}\Delta t^3 \\ \frac{1}{2}\Delta t^3 & \Delta t^2 \end{bmatrix}\sigma_p^2 \text{ and } \mathbf{R}(k+1) = \sigma_m^2$$

where σ_p^2 and σ_m^2 represent the uncertainties of (probably maneuvering) acceleration and measurement respectively.
The y coordinate centroid is modelled with the same equations. The height, h and width w are modelled in the similar way; however, the piecewise-constant white velocity models are used in stead. This yields the following equations for h (and similarly for w):

$$h(k+1) = h(k) + v_h(k)\Delta t \tag{9}$$

where $v_h(k+1)$ is a zero-mean noise sequence of the rate of change in h.

$$z_h(k+1) = h(k+1) + w_h(k+1) \tag{10}$$

where $w_h(k+1)$ is a zero-mean measurement noise sequence of h.
These variables are assumed uncorrelated and the final equations can be obtained by augmenting the vectors block-wisely and the matrices block-diagonally. The model is initialised by the two-point differencing method [22].

4.2 Appearance Model

Most colour tracking systems model multi-coloured objects using a colour histogram or a mixture of Gaussian distributions. If only a small set of samples is available, most of the bins in the histogram are empty and the solution of fitting data to a mixture model tends to converge to a singular solution. One answer to this is to increase the bin size or limit the number of Gaussian components in the mixture. This leads to another difficulty of how large the bin size should be or how many components in the mixture that is sufficient to represent the colour distribution of the object. Too large a bin size or too few mixture components can result in no discernible difference between objects. Too small bin size or too many mixture components may not only cause singular solutions but also not be able to cope with change when objects move pass different regions in the scene. Our colour model utilises a colour transformation obtained from consensus colours in Munsell colour space in converting obtained colours into eleven basic colours [23]. This consensus colour was experimentally developed by Sturges *et al* [7]. Figure 3 and Table 1 display the Sturges' consensus areas and focal points of chromatic and achromatic colours respectively. The focal points are the samples which were, on average, named the fastest. The ill-defined regions of the colour space shown in figure 3 depict the colours which were ambiguously interpreted by the human subjects. The colour conversion for this region is obtained by a nearest neighbour from the colour point to the edge of the consensus areas. The colour histogram therefore contains eleven normalised bins. The *L*-recent instantaneous histograms will be maintained in the model by

$$\hat{B}_i(k+1) = \hat{B}_i(k) + \beta[B_i(k+1) + \hat{B}_i(k)] \quad ;k = 1,2,\ldots \tag{11}$$

where $\beta = \max(\frac{1}{L}, \frac{1}{k+1})$.

$\hat{B}_i(k+1)$ is the i^th^ bin of the filtered histogram at frame *k*+1. $B_i(k+1)$ is the i^th^ bin of the instantaneous histogram at frame *k*+1.

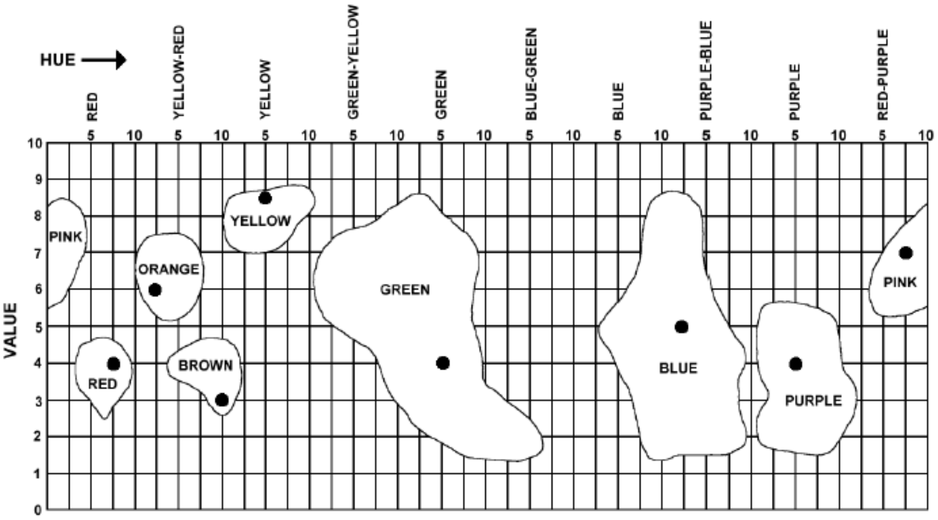

Figure 3: Location of consensus areas and focal points on two dimensional representation of the Munsell space identified by Sturges *et al*.

Colour	Focal Point	Consensus Range
White	9.5	V ≥ 9
Grey	5.5	4 ≤ V ≤ 7
Black	0.5	V ≤ 1

Table 1: Location of consensus areas and focal points of achromatic colours.

4.3 Data association module

This module is based on both motion prediction and appearance similarity. Each tracking model includes motion and appearance models. It gives an ellipsoidal search area called *gate* [24]. In our system, this area is represented by a Mahalanobis distance of less than or equal 3 from the predicted measurement with a covariance matrix being the Kalman innovation covariance matrix, **S**. Figure 4 shows an example of locations of gates and measurements.

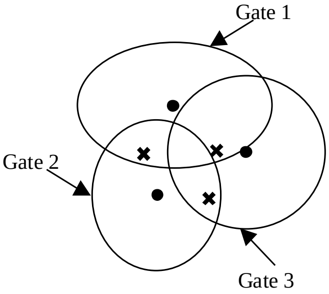

Figure 4: Example of Gates (measurements are displayed by x).

The observations which are within this gate, will have its geometric score calculated by

$$GS_{ij} = \begin{cases} \dfrac{D_{ij}}{\sum\limits_{\mathbf{z}_l \in G_j} D_{lj}} & ; \mathbf{z}_i \in G_j \\ 0 & ; otherwise \end{cases} \tag{12}$$

where

G_j is the gate of model j and $D_{ij} = \left|\mathbf{S}_j\right|^{-\frac{1}{2}} e^{-\frac{1}{2}(\mathbf{z}_i - \hat{\mathbf{z}}_j(k+1|k))\mathbf{S}_j^{-1}(\mathbf{z}_i - \hat{\mathbf{z}}_j(k+1|k))}$

An appearance similarity score, *AS* will be calculated by histogram intersection

$$AS_{ij} = \begin{cases} \dfrac{A_{ij}}{\sum\limits_{\forall l} A_{lj}} & ; (\sum\limits_{\forall l} A_{lj} \neq 0) \\ 0 & ; otherwise \end{cases} \tag{13}$$

where $A_{ij} = \begin{cases} HI_{ij} & ; HI_{ij} \geq th_a \\ 0 & ; otherwise \end{cases}$ and $HI_{ij} = \sum\limits_{k=1}^{11} \min(B_{ik}, \hat{B}_{jk})$

This allows the object's appearance to change within a predefined threshold. In our experiment, th_a was set at 0.6. The elements of the assignment matrix, **T** are then computed by

$$T_{ij} = GS_{ij} + AS_{ij} \tag{14}$$

The search for the best solution is done by recursively matching the best match pair in the assignment matrix. The matched candidates are taken out the matrix then the process iterates until no further match can be found. The matched models are updated by the corresponding object if their shape does not change significantly. The acceleration noise variance of the unmatched models will be reduced to decrease the chance of stealing another object's trajectory. New models will be created for all unmatched objects.

4.4 Trajectory Maintenance Module

Since the system has no explicit model of occlusion. Situations like objects which move slowly into the scene or are occluded by other moving objects / stationary objects generate many spurious trajectories. The trajectory maintenance module is designed to dynamically eliminate the trajectories which looses their tracks for a relatively long period. This module eliminates spurious trajectories resulted from noise, small repetitive motion and occlusions.

5 Experimental Results

Figure 5 shows a sequence of busy outdoor scene containing people walking in a public pathway. The sequence shown here is 192x144 images. The sequence includes strong sunshine, large shaded area, tree, reflections from glasses and long moving shadows. Targets typically contain 40 to 60 pixels. By applying our method, we can detect and track moving objects successfully even after occlusions. It is evidence that many spurious trajectories/hypotheses are constructed during occlusion and sometimes during normal tracking. This is due to background clutter, imperfect shadow or noise elimination. However, these short-lived tracks are eliminated by the trajectory maintenance module.

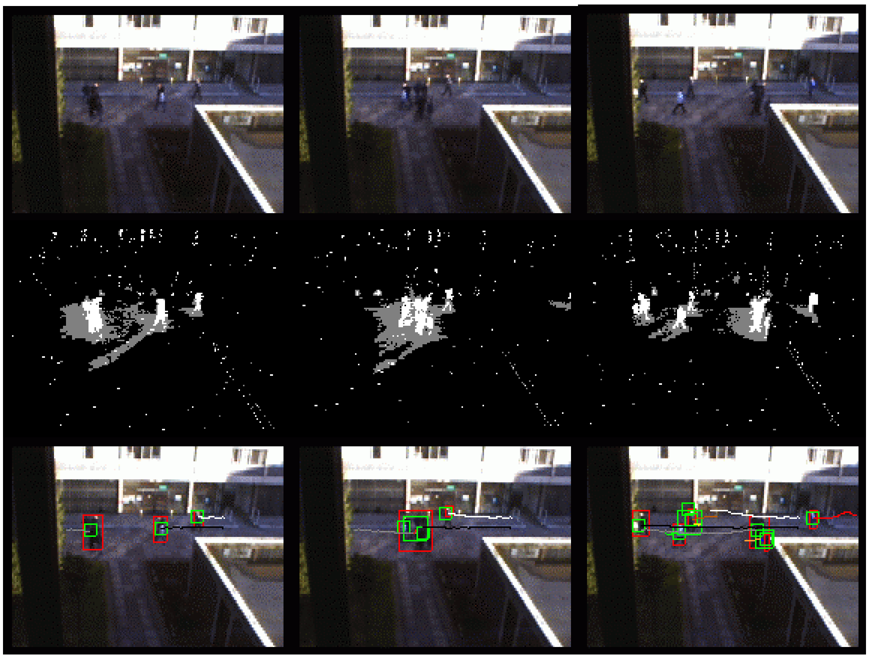

Figure 5: The top row displays the original sequence at frames 220, 268 and 326 respectively. The second row shows the results from background subtraction and moving shadow elimination. The shadows are shown in grey. The last row illustrates tracking process. The red boxes are the minimum bounding box of the objects while gate of the object centroid of each Kalman filter is the maximum inscribed ellipse of the green box. The trajectories are shown with different colours.

6 Conclusions

We have presented a method to track low-resolution moving objects in the image, using colour, simple shape and motion information. The key strength of this method is the use of robust background modelling and colour mapping obtained from anthropological study to model low-resolution colour targets. We also introduced a probabilistic method for combining this information to obtain a robust tracking system. Due to the robust background modelling, few false detections are picked up even in hostile environments. Incorporating the tracking and association module, the system has been successfully applied to track multiple people walking in a surveillance area. It also shows ability to recover lost tracks due to occlusions and background clutter.

References

1. Bishop, C. M., *Neural networks for pattern recognition*. 1995, Oxford, New York: Clarendon Press ; Oxford University Press. xvii, 482.
2. Grimson, W.E.L, Stauffer, C., Romano, R., and 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. IEEE Comput. Soc. 1998*. 1998.
3. Stauffer, C. *Automatic hierarchical classification using time-based co-occurrences*. in *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Comput. Soc. Part Vol. 2, 1999*. 1999.
4. Stauffer, C. and 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). IEEE Comput. Soc. Part Vol. 2, 1999*. 1999.
5. Stauffer, C. and 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.
6. Lipton, A., Fujiyoshi, H., and Patil, R. *Moving Target Classification and Tracking from Real-time Video*. in *Proc. of the 1998 DARPA Image Understanding Workshop (IUW'98)*. 1998.
7. Sturges, J. and Whitfield, T. W. A., *Locating basic colours in the Munsell space*. Color Research and Application, 1995. **20**(6): p. 364-376.
8. Seaborn, M., Hepplewhite, L., and Stonham, J. *Fuzzy colour category map for content based image retrieval*. in *BMVC99. Proceedings of the 10th British Machine Vision Conference. Univ. Nottingham. Part vol.1, 1999*. 1999.
9. Swain, M. J. and Ballard, D. H., *Color indexing*. International Journal of Computer Vision, 1991. **7**(1): p. 11-32.
10. Birchfield, S. *Elliptical head tracking using intensity gradients and color histograms*. in *Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Comput. Soc. 1998*. 1998.
11. Bradski, G. R., *Computer Vision Face Tracking For Use In A Perceptual User Interface*. Intel Technology Journal, 1998(2).
12. Wren, Cr, Azarbayejani, A., Darrell, T., and Pentland, A. P., *Pfinder: real-time tracking of the human body*. IEEE Transactions on Pattern Analysis & Machine Intelligence, 1997. **19**(7): p. 780-5.
13. McKenna, S. J., Raja, Y., and Gong, Shaogang, *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.
14. McKenna, S. J. and Gong, S. *Tracking faces*. in *Proceedings of the Second International Conference on Automatic Face and Gesture Recognition. IEEE Comput. Soc. Press. 1996*. 1996.
15. Raja, Y., McKenna, S. J., and 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.
16. Raja, Y., McKenna, S. J., and Gong, Shaogang, *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.
17. Haritaoglu, I., Harwood, D., and Davis, L. S., *W/sup 4/: real-time surveillance of people and their activities*. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2000. **22**(8): p. 809-30.
18. Koller, D., Weber, J., and Malik, J. *Robust multiple car tracking with occlusion reasoning*. in *Computer Vision - ECCV'94. Third European Conference on Computer Vision. Proceedings. Vol.I. Springer-Verlag. 1994*. 1994.
19. KaewTraKulPong, P. and R., Bowden. *An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection*. in *2nd European Workshop on Advanced Video-based Surveillance Systems*. 2001. Kingston upon Thames.
20. KaewTraKulPong, P. and R., Bowden, *Comments on: "Learning Patterns of Activity Using Real-Time Tracking". Submitted to* IEEE Transactions on Pattern Analysis & Machine Intelligence, 2001.
21. Horprasert, T., Harwood, D., and Davis, L.S. *a statistical approach for real-time robust background subtraction and shadow detection*. in *IEEE ICCV'99 FRAME-RATE WORKSHOP*. 1999.
22. Bar-Shalom, Y. and Li, X.-R., *Estimation and tracking : principles, techniques and software*. 1993, Boston ; London: Artech House.
23. Berlin, B. and Kay, P., *Basic color terms : their universality and evolution*. Paperback ed. 1991, Berkeley ; Oxford: University of California P. 196p.
24. Blackman, S. and Robert, P., *Design and analysis of modern tracking systems*. Artech House radar library. 1999, Boston, Mass.. London.: Artech House. xxxi,1230p. ill. 24cm.