# VIDEO SEQUENCE INDEXING THROUGH RECOVERY OF OBJECT-BASED MOTION TRAJECTORIES

**A. Naftel**
Department of Computation
UMIST
Sackville Street
Manchester M60 1QD, UK
andrew.naftel@co.umist.ac.uk

**S. Khalid**
Department of Computation
UMIST
Sackville Street
Manchester M60 1QD, UK
s.khalid-2@postgrad.umist.ac.uk

**Abstract**
In this paper, we present a hybrid approach for tracking multiple objects through occlusion observed by a stationary camera. This tracking data can then be used to generate accurate object motion trajectories that provide an index key into a database of motion sequences. The system is intended for tracking people and objects in crowded environments such as supermarkets or shopping malls. The output of the system can be used for intelligent behavioural analysis or activity-based video indexing and retrieval for security management. The approach starts with a robust foreground object detection and SAKBOT-based shadow suppression stage. It is shown how both static and dynamic occlusions are handled using a first-order Kalman Filter combined with a simple colour model incorporating histogram intersection and back-projection for each tracked object. In the next step we use a RANSAC-type approach to generate smooth motion trajectories for each object modelled with *m*-degree polynomials. Preliminary results are presented to show how our method produces robust trajectory paths insensitive to outliers containing high numbers of mis-detected points. A similarity metric is then defined using polynomial coefficients. This enables a user to construct a motion trajectory query which can be used to index into a database of surveillance clips and retrieve similar results..

**Keywords**: object tracking, shadow detection, motion trajectory, occlusion handling.

## 1 Introduction

Intelligent surveillance systems are assuming an increasingly important role in crime detection and prevention as the number of installed camera networks can attest. One of the most important tasks for the next generation of commercial CCTV surveillance systems is to automate the process of tracking people, objects and their interactions in complex and crowded environments. The tracking problem (i.e. establishing inter-frame correspondence for individual objects over a video sequence) has been extensively studied in the computer vision literature [1][2]. However, the issue of how to curate the vast quantities of tracking data collected has only recently been addressed by researchers. One approach is through semantic video interpretation [3] where the system attempts to recognise user-predefined events such as certain types of possible criminal activity. An alternative is to analyse object motion paths [4][5][6] in order to learn and predict patterns of behaviour, or to allow users to create queries about the content of surveillance scenes [7][8][9], e.g. trajectory, colour, type of object, etc. and thereby retrieve useful information.

Our work most closely relates to [7][8], since the aim of the project is to develop a system for indexing and retrieval of relevant video sequences based on object motion paths. The specific application domain addressed is indoor retail store surveillance which offers a number of challenging problems when attempting to automate scene analysis. These are as follows:

- **Static/dynamic occlusion**: Indoor environments such as retail stores and shopping malls are often crowded and hence are full of static and dynamic objects that may partially or totally occlude the target object. This results in rapid appearance and shape changes which must be dealt with carefully if identified objects are not to be mis-classified. This is an inherent problem when attempting to analyse crowded scenes.

- **Shadows:** Often strong artificial illumination from multiple light sources used in indoor scenes introduces problems in effective foreground detection due to the generation of shadows of varying intensities in different parts of the scene. When the object moves close to the light source, the intensity of the shadow increases rapidly.
- **Background changes:** Previously moving objects that suddenly become stationary in the scene for long periods or sharp changes in lighting conditions cause instability in the background model. As the first step in reliable object segmentation is normally background subtraction, the system must react and adapt to the background changes by frequently updating the model.

### 1.1 Related work

There have been numerous efforts to robustly track objects in crowded scenes. This work can be categorised into single or multiple cameras, static or moving cameras, colour or grey scale, and single or multiple person tracking systems. The $W^4$ system [10] tracks people in grey scale video obtained from a static camera. The foreground object is detected using a statistical background model, and a set of features (such as silhouette calculation, body parts localisation) for each object and group is computed. These features are then used to track moving objects through various types of occlusion. An enhancement to the system known as W4S, which integrates real-time stereo computation in order to suppress shadows (previously detected as separate blob or new foreground objects), is described in [11]. Other noteworthy tracking systems working with fixed cameras have been reported [12][13][14].

In [15], tracking is accomplished by decoupling the problem into two parts. Firstly, the object appearance is defined using a colour-based object representation and it then models 2D and 3D velocities of the object. An appearance-based description of moving objects is used for measuring similarity among detected moving objects whereas Kalman Filtering is used for 2D/3D modelling of tracked objects. The SAKBOT [16] approach enables effective tracking of objects, even in the presence of heavy shadows. It uses HSV colour space to improve the accuracy in detecting shadows by exploiting the general effects of shadows on the HSV component of the pixel on which it falls. This effect includes a lowering of brightness value for the pixel (caused by darkening) greater than expected with little effect on the S and V (colour) component.

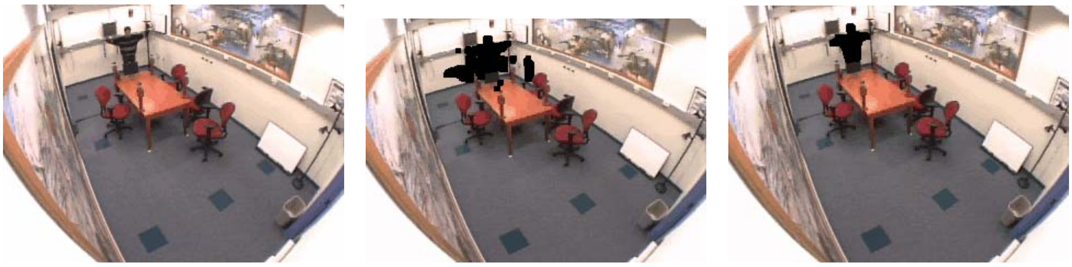
Here, we propose a simple and effective solution for tracking multiple objects in a busy environment such as a shopping centre or retail store. The solution combines various existing techniques with some modifications.

The remainder of the paper is organised as follows. In section 2, an algorithm to detect foreground objects using an adaptive statistical background model is discussed. A technique to avoid moving shadows being classified as part of a moving object is then described. Section 3 specifies the algorithm used to track objects in various possible scenarios. These scenarios include tracking of multiple objects through both partial and complete static or dynamic occlusions. The algorithm uses a hybrid model comprising a spatial prediction component based on the output of a first order Kalman Filter and appearance model component based on technique of colour indexing proposed by Swain and Ballard [17]. In section 4, we describe the procedure for modelling the motion path generated by the object tracker points. This method is shown to be insensitive to gross outliers in the data, instabilities inherent in the tracking algorithm and is particularly suited to smoothing through occluded sequences. We then show how to use the output for indexing motion histories. Preliminary results are presented in section 5, concluding with a discussion and summary in section 6.

## 2 Foreground Object Detection

We adopt the adaptive background modelling technique based on [10] which has proved reliable. Before background subtraction can be applied, an initial background model should be learned based on frames with a majority of the background visible. However, the algorithm can create an initial background model even if there are small localised visible objects moving in the scene. A set of masks can be used to neglect the moving objects and we can select only the valid regions to be used to update the background model. It also caters for any object that moves into the scene (then identified as a foreground object) and remains stationary for a long time period. Similarly, any object which is initially assumed to be the part of the background but then starts to move, can cause false background changes for a short period but settles down in a reasonable number of frames.

This is combined with shadow detection based on SAKBOT model [16]. Shadows are detected by assuming that they reduce the intensity of the underlying pixel without having a significant effect on its colour. As background subtraction only takes into account the brightness component of pixel, we need to model hue and saturation pixel components separately for shadow removal. The result of applying background subtraction with shadow suppression is shown in Fig 1.



(a) (b) (c)

**Fig. 1. (a) Current frame. (b) Foreground object detected after background subtraction. (c) Foreground object after background subtraction with shadow detection and removal.**

# 3 Tracking via Motion and Appearance Models

This section describes the techniques employed to track labelled moving objects through frames. We deploy a simple motion model based on first order Kalman Filter and an appearance model using colour histogram intersection and backprojection [17]. The advantage is this approach is its speed and simplicity of representation.

The motion model for the object is specified as follows. The bounding box and centroid coordinates of the identified object are used as the state and measurement variables in the Kalman Filter such that:

$$\mathbf{m} = [x_1 \quad y_1 \quad x_2 \quad y_2] \tag{1}$$

$$\mathbf{S} = [x_1 \quad y_1 \quad x_2 \quad y_2 \quad \Delta x_1 \quad \Delta y_1 \quad \Delta x_2 \quad \Delta y_2 \quad w \quad h] \tag{2}$$

where **m** and **S** are measurement and state models and ($x_1$, $y_1$, $x_2$, $y_2$) represents the left, top, right, bottom boundaries of the bounding box. ($\Delta x_1$, $\Delta y_1$, $\Delta x_2$, $\Delta y_2$) represent the corresponding change in the values of boundaries in recent frame and ($w$, $h$) specify the overall width and height of the bounding box. These variables are used in the case where one bounding edge of a target is observable and its opposite boundary just becomes occluded. The occluded boundary can then be approximated by adding/subtracting the $w$ or $h$ state variable. Since we assume the object moves through image space at constant velocity, the new position at time $t + 1$ is predicted from the position at $t$ by the equation:

$$\begin{pmatrix} x_1^{t+1} \\ y_1^{t+1} \\ x_2^{t+1} \\ y_2^{t+1} \\ \Delta x_1^{t+1} \\ \Delta y_1^{t+1} \\ \Delta x_2^{t+1} \\ \Delta y_2^{t+1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1^t \\ y_1^t \\ x_2^t \\ y_2^t \\ \Delta x_1^t \\ \Delta y_1^t \\ \Delta x_2^t \\ \Delta y_2^t \end{pmatrix} + \begin{pmatrix} n_1^t \\ n_2^t \\ n_3^t \\ n_4^t \\ n_5^t \\ n_6^t \\ n_7^t \\ n_8^t \end{pmatrix} \tag{3}$$

The appearance model for the object is constructed as soon as the foreground blob is identified as a valid moving object. The object model is obtained by creating a colour histogram for the pixels considered part of the object. Each component of the colour model is quantised using a variable number of bits. Here, we use 5 bits (32 bins) for each colour component (H and S) and 4 bits for brightness (V) component. A smaller number of bins for V component is consistent with the fact that as the object moves, the brightness of the illumination varies according to its distance from the light source. The overall structure of the tracking algorithm is illustrated in Fig. 2. The tracking algorithm works as follows. For each frame in the video sequence:

Step 1. Predict the new position of each tracked object using eq.(3).

Step 2. Calculate the most likely position of the object based on the prediction and the actual measurement associated with the object. If the measurement obtained varies significantly from the predicted position (e.g. in the case of static occlusion), use the predicted position.

Step 3. Use histogram backprojection technique[17] to identify the location of the object centroid based on colour model. Use the additional information obtained to validate and adjust the object location.

Step 4. Update the object state variable based on the object's most likely position.

Step 5. Update the colour model for the object if it is not subject to static or dynamic occlusion.

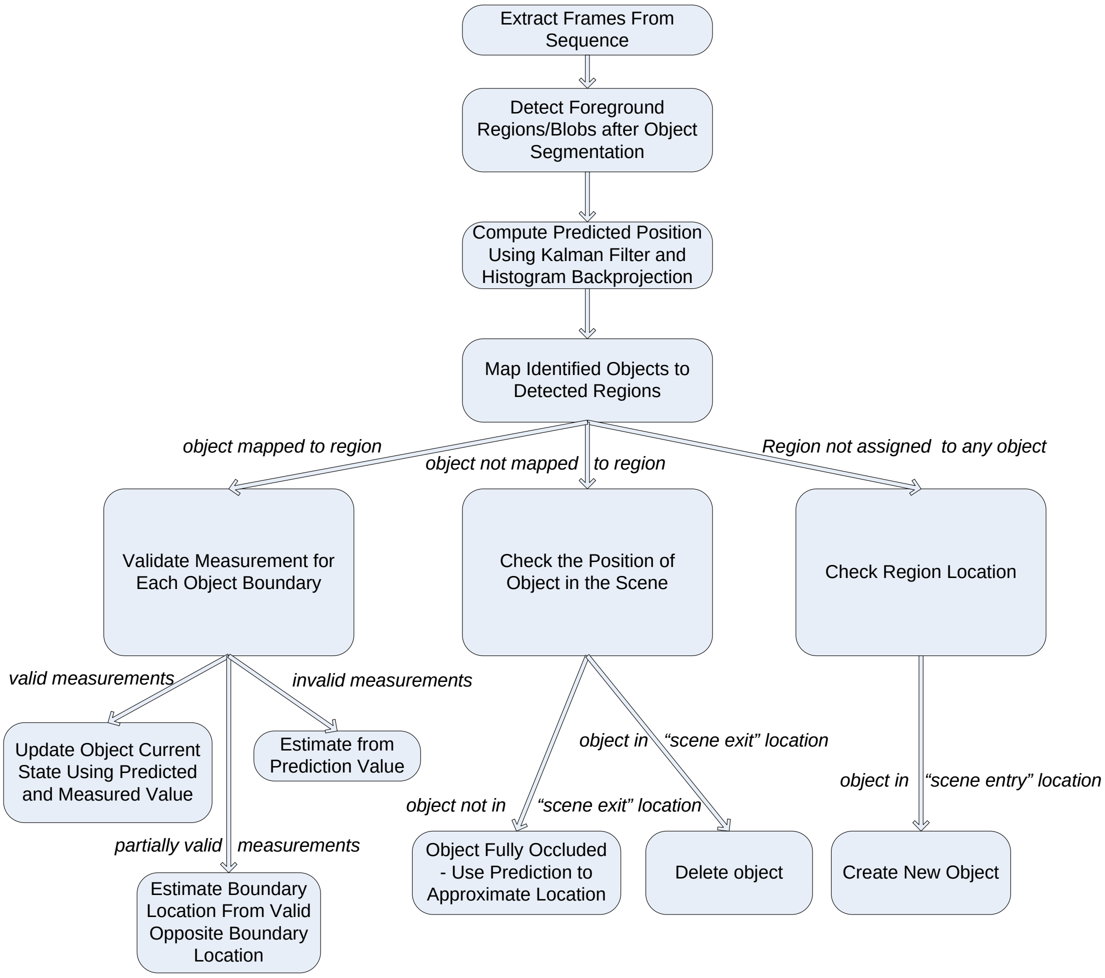

Extract Frames From Sequence
Detect Foreground Regions/Blobs after Object Segmentation
Compute Predicted Position Using Kalman Filter and Histogram Backprojection
Map Identified Objects to Detected Regions
object mapped to region
object not mapped to region
Region not assigned to any object
Validate Measurement for Each Object Boundary
Check the Position of Object in the Scene
Check Region Location
valid measurements
invalid measurements
Update Object Current State Using Predicted and Measured Value
Estimate from Prediction Value
partially valid measurements
Estimate Boundary Location From Valid Opposite Boundary Location
object in "scene exit" location
object not in "scene exit" location
Object Fully Occluded - Use Prediction to Approximate Location
Delete object
object in "scene entry" location
Create New Object

**Fig. 2. Block diagram of the tracking algorithm**

For objects that are dynamically occluded, extra processing is needed. In this type of situation, an object may partially or fully occlude the other objects. When different objects start to overlap in the scene and appear to move together, then all the constituent objects of the group are tracked as one large blob. Within the blob, the location of the object is approximated by the colour model using a histogram backprojection technique. This is accomplished as follows.

Assuming a pair of multi-dimensional histograms $G$ and $H$ each containing $n$ bins, where $G$ represents the target object model and $H$ the 'background' image, we define a ratio histogram Ω between object and image as

$$\Omega_i = \min\left(\frac{G_i}{H_i}, 1\right) \tag{4}$$

Image values are then replaced by the values of $\Omega_i$ which they index. The backprojected image is then convolved with a mask which is approximately the size of the object's bounding box. The index with maximum value in the convolved image is the approximate location of the object. This additional information provides a cue for the object location within the larger blob (representing multiple objects with dynamic occlusion). When the objects separate from each other, the unique identity of each object is verified by intersecting the separated region model (i.e. histogram) with the histogram of all the objects (originally part of the dynamically occluded blob) in the current frame. The histogram intersection measure $\Psi$ is defined as

$$\Psi = \sum_{i=1}^{n} \min(R_i, G_i) \tag{5}$$

where $\Psi$ represents the number of pixels with the same colour in the two reference histograms, $R$ is the colour histogram of the region, and $G$ is the target object histogram. The object with the maximum value for $\Psi$ is assigned to the region that is separated from the dynamically occluded group.

# 4 Modelling the Motion Path

## 4.1 Model Fitting

As for most tracking algorithms, the output is a set of (usually noisy) 2-D points representing the frame-to-frame reference location of an object tracked through the image space. We propose to model the overall shape of the resulting tracked points using a low degree polynomial. For more complex motions, the representation could be piecewise but we do not consider that here. The advantages are that a model representation will result in significant compression of the tracked data and it can also be used to index stored video sequences where the generic motion path of an object is of interest, e.g. to a CCTV operator.

We consider a RANSAC implementation [18] for the least squares (LS) approximation of a set of $n$ data points ($x_i$, $y_i$) ($i$ = 1,2,...,$n$) by a polynomial $p_m(x)$ of degree $m < n$. The unknown $m$+1 coefficients $a_k$ ($k$ = 0,1,...,$m$) can be determined by minimising the function E with respect to $a_0$, $a_1$, ...

$$E(a_0, a_1, \ldots, a_m) = \sum_{i=1}^{n} \left[ y_i - \left( a_0 + a_1 x + \ldots + a_m x_i^m \right) \right]^2 \tag{6}$$

This is suitable in the case where $x$ coordinate values are monotonically increasing. Where the values are monotonically increasing in $y$, we reverse the roles of $x$ and $y$ in eq.(6).

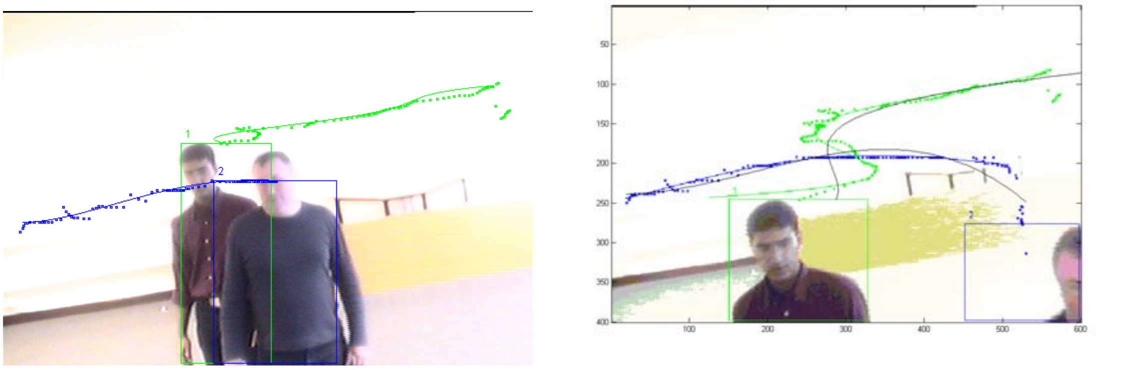
It is well known that least squares is a smoothing technique that is highly sensitive to gross errors. These outliers commonly arise in the tracking process due to object miss-classification and measurement error. RANSAC, on the other hand, is particularly suited to model fitting where the data is highly contaminated by outliers. Instead of using all the points to fit the curve (as in LS), it initialises the model with as small a data set as possible and then enlarges this set with consistent data where possible. When there are sufficient mutually consistent points, RANSAC then employs a smoothing technique such as LS to compute an improved estimate for the fit. This is demonstrated in Fig. 3 where the RANSAC result provides a more faithful representation of the motion path data. The intersection of the curves indicate the position at which object occlusion occurs. In most cases, $m$ = 3 provides an adequate representation of the modelled trajectory.

## 4.2 Similarity Metric for Retrieval of Motion Paths

Since we wish to search and retrieve similar trajectories for tracked objects, it makes sense to index the video motion clips in a database using a model-based descriptor. Each tracked and labelled video object is therefore represented by the set of coefficients $\{a_i\}$ of the interpolated curve through its motion path. When a user invokes a query (motion path) which could be a free-hand sketch or set of trend points marked on a representative background scene, the coefficients are generated and compared to each of those in the database of clips using a Euclidean distance metric. The best matches are then retrieved in order of similarity. The similarity metric $d$ is defined as

$$d(M_q, M_k) = \left\{ \sum_{i=1}^{m} (a_{iq} - a_{ik})^2 \right\}^{1/2} \tag{7}$$

where $M_q = \{a_{iq}\}$ and $M_k = \{a_{ik}\}$ ($i = 1,..., m$) denote the coefficient set for the query and stored motion path models respectively.



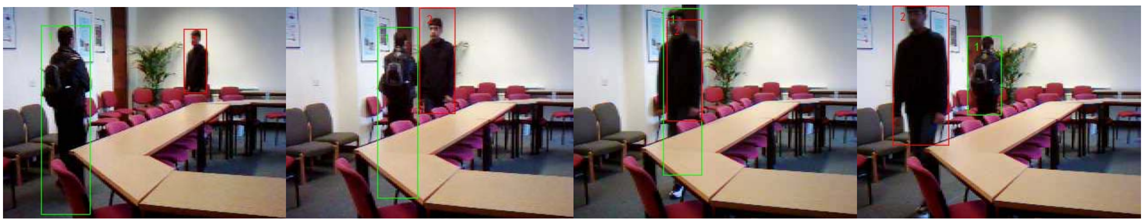
(a) (b)

**Fig. 3. Fitting polynomials of degree 3 to motion paths. (a) Tracking up to occluding frames. (b) Comparison of LS and RANSAC model fitting. RANSAC produces tighter fitting curves.**

# 5 Experimental Results

In this section, we present some results to indicate the effectiveness of the proposed techniques for tracking people through static and dynamic occlusions. We then generate motion path models and demonstrate how these can be used for object-based video indexing and retrieval.

The results shown in Fig. 4 demonstrate object tracking and interaction in the presence of static and dynamic occlusion. In Fig. 4(a), objects are tracked independently in the presence of static occlusion (represented by the table). Objects move towards each other and come into contact, thus merging into a single blob, but are still identified as separate objects shown in Fig. 4(b). Object 1 moves behind object 2 and is completely occluded as shown in Fig. 4(c). Two objects then separate and are identified and tracked with the correct label as shown in Fig. 4(d). The results demonstrate usefulness of appearance model since both objects are of similar colour distributions and object 1 is completely occluded by object 2 for some time as shown in Fig. 4(c).

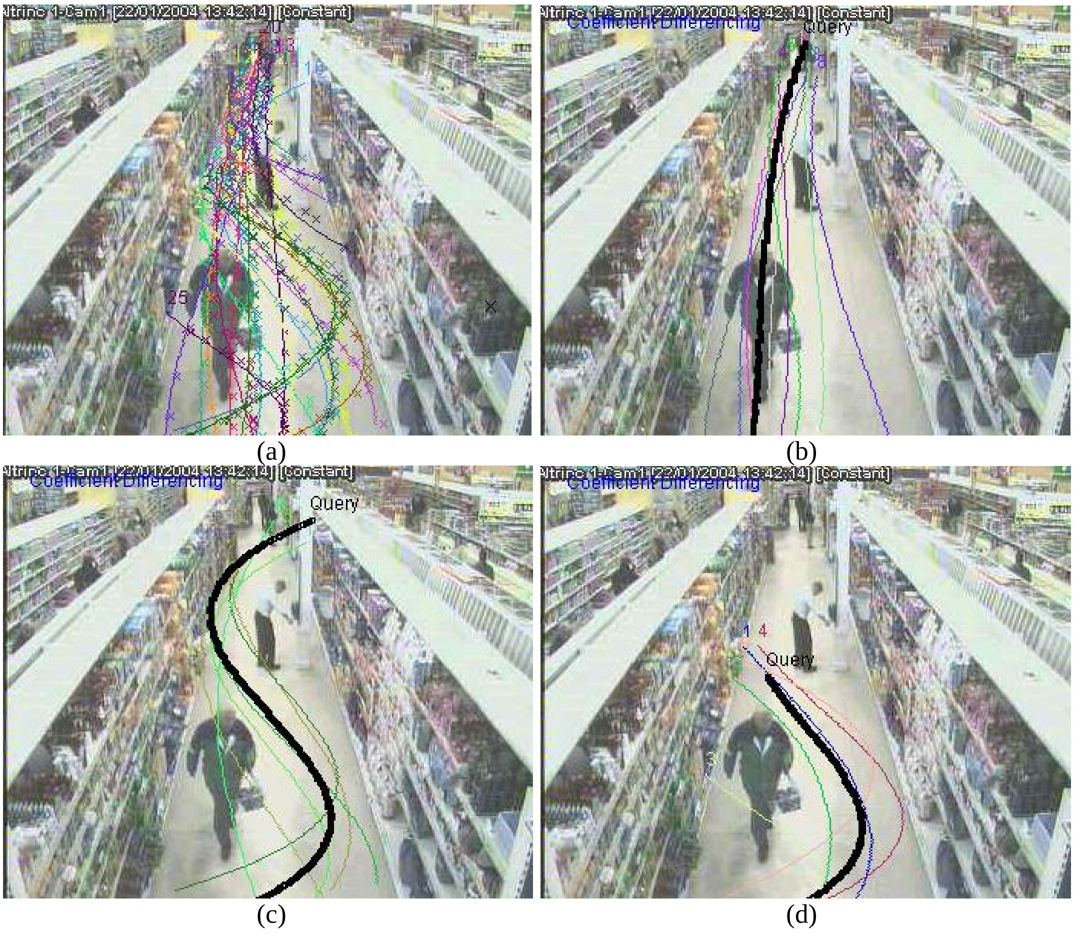


(a) (b) (c) (d)

**Fig. 4. Tracking of multiple objects through occlusions. (a) Objects are tracked independently (b) Objects come in contact with each other (c) Object 1 is fully occluded by object 2 (d) Tracking continues with correct labels on the objects.**

Fig. 5 illustrates the results of using eq.(7) to search and retrieve object motion paths, similar to a user-defined query, from a surveillance database of motion clips. A partial or complete trajectory has been recovered for each successfully tracked object in the motion clip using the method described in section 4.1. A sample set of trajectories are shown in Fig. 5(a). Where the motion path is more complex and cannot be adequately modelled using a low-order polynomial, either this has been excluded from the database or stored only as a partial trajectory. The same is true of object paths where tracking has been lost due to complete occlusion.

Figs. 5(b)-(d) show the object motions retrieved for various user-specified queries. The stored trajectories (and hence motion clips) are ranked according to their degree of similarity to the query and the results indicate those inter-trajectory coefficient distances lying within a certain tolerance $\tau$, where $d(M_q,M_k) < \tau$. The coefficient distance metric, though simple to compute, appears to give plausible results even in the case of a partial trajectory query, shown in Fig. 5(d). In future work, we intend to compare the performance of several different similarity metrics including Hausdorff distance measures (HDM). HDMs [6] are expensive to compute but have the advantage of working with point sets that are more suited to the case of complex trajectory shapes. We also intend to investigate the addition of a velocity difference term to the metric since this important information is currently neglected.



(a) (b)

(c) (d)

**Fig. 5. Using a user-sketched query to retrieve similar motion paths. (a) Database of stored motion paths. (b)-(d) Highest ranked results for various queries based on closest distances in coefficient space-only those trajectories lying within a certain tolerance are displayed.**

# 6 Conclusion

We have presented a simple but effective approach for tracking multiple objects through static and dynamic occlusions. It requires colour images as this is vital for shadow detection and maintaining an object-based appearance model used for disambiguating merged regions during occluding frames. If the predicted object position varies significantly from the measured position based on the current frame, Kalman Filtering is used to estimate the new location. The prediction is then adjusted after performing histogram intersection of the object in the current frame.

Motion trajectories are then modelled via polynomial interpolation adopting a RANSAC approach for ensuring the generated motion paths are resistant to outliers. The coefficient descriptors prove to be a useful index key into a database of video clips representing object motions. A user-defined query can be sketched as a means of retrieving similar motion events which makes this a useful tool for surveillance-based intelligent behaviour analysis.

# Acknowledgements

Mr Shehzad Khalid would like to thank the Higher Education Commission of Pakistan for financial support of his postgraduate studies.

# References

[1] J. Aggarwal and Q. Cai, "Human motion analysis: A review," Computer Vision and Image Understanding, vol. 73, pp. 428--440, 1999.

[2] L. Wang, W. Hu and T. Tan, Recent developments in human motion analysis, Pattern Recognition, Volume 36, Issue 3, March 2003, Pages 585-601

[3] B. Georgis, M. Maziere, F. Bremond, M Thonnat, "A video interpretation platform applied to bank agency monitoring, Proc. IEE Intelligent Distributed Surveillance Systems (IDSS-04), February 23, 2004, London, UK, pp. 46-50.

[4] D. Makris and T. Ellis, Path detection in video surveillance. Image & Vision computing, 20 (2002) 895-903

[5] N. Johnson and D. Hogg "Learning the distribution of object trajectories for event recognition", Image & Vision Computing, 14 (1996) 609-615.

[6] J. Lou, Q. Liu, T. Tan, Weiming Hu, Semantic Interpretation of Object Activities in a Surveillance System. 16th International Conference on Pattern Recognition (ICPR'02) Volume 3 August 11 - 15, 2002

[7] Y.Jung, K. Lee, Y. Ho, Content-Based event retrieval using semantic Scene interpretation for automated traffic surveillance, IEEE Trans. Intell. Transport. Syst. 2, 151-163, 2001.

[8] W. P. Berriss, W. G. Price and M.Z. Bober, “The Use of MPEG-7 for Intelligent Analysis and Retrieval in Video Surveillance” Proc. IEE Intelligent Distributed Surveillance Systems Symposium (IDSS-03), pp. 8/1 – 8/5, London, February 25, 2003.

[9] A. J. Lipton, J. Clark, P. Brewer, A. Chosak, P. Venetianer, Object video forensics: Activity-Based Video Indexing and Retrieval for Physical security, IDSS-04, February 23, 2004, London, UK, pp. 56-60.

[10] Haritaoglu, D. Harwood, and L.S. Davis. W4: Real-Time Surveillance of People and Their Activities. IEEE Tras. On Pattern Analysis and Machine Intelligence, 22(8):809-830, August 2000.

[11] Haritaoglu, I., D Harwood & L. Davis (1998). W4S: A Real Time System for Detecting and Tracking People in 2.5D. European Conference on Computer Vision, 1998

[12] C. Wren, A. Azarbavejani, T. Darrell, and A. Pentland, “Pfinder: Real-Time Tracking of the Human Body”, IEEE Trans. Pattern Analysis and Machine Intelligence vol.19, no. 7, July 1997.

[13] Robert T. Collins, Alan J. Lipton, Hironobu Fujiyoshi and Takeo Kanade, "Algorithms for Cooperative Multisensor Surveillance," Proceedings of the IEEE, Vol. 89(10), Oct 2001, pp.1456-1477.

[14] Robert T. Collins, Alan J. Lipton, Takeo Kanade, Hironobu Fujiyoshi, David Duggins, Yanghai Tsin,David Tolliver, Nobuyoshi Enomoto, Osamu Hasegawa, Peter Burt1 and Lambert Wixson1. “A System for Video Surveillance and Monitoring”. The Robotics Institute , Carnegie Mellon University. CMU-RI-TR-00-12

[15] J. Kang, I. Cohen and G. Medioni, “Tracking Objects from Multiple Stationary and Moving Cameras”, Proc. IEE Intelligent Distributed Surveillance Systems (IDSS-04), February 23, 2004, London, UK, pp. 31-35.

[16] R. Cucchiara, C. Grana, M. Piccardi, A. Prati, and S. Sirotti, “Improving shadow suppression in moving object detection with HSV colour information ”, in Proc. of the 4th International IEEE Conference on Intelligent Transportation Systems, August 25-29, 2001, Oakland, CA, USA, pp.334-339.

[17] M. J. Swain and D. H. Ballard. Color Indexing. International Journal of Computer Vision, 7(1):11-32, 1991.

[18] M. A. Fischler, R. C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Comm. of the ACM, Vol 24, pp 381-395, 1981.