To appear in Proc. IEEE Workshop on Applications of Computer Vision, January 2005.

Evaluation of Matching Metrics for Trajectory-based Indexing and Retrieval of Video Clips

Shehzad Khalid, Andrew Naftel
School of Informatics, University of Manchester, UK
s.khalid-2@postgrad.manchester.ac.uk, a.naftel@manchester.ac.uk

Abstract

This paper describes a comparative evaluation of three different similarity metrics for trajectory-based indexing and retrieval of video motion clips. The motion paths are generated using a low-level tracking algorithm incorporating first-order Kalman filter and colour appearance models.

For simple motion paths, a RANSAC approach can be used to generate smooth trajectories for each tracked object described by low–order polynomials. This allows us to obtain a representative trajectory model even in the case of high numbers of outlier points caused by target mis-detection and multiple occlusions.

We show that more complex trajectories including stop-start motions, can be modelled as time series using high order Chebyshev polynomials. Similarity metrics based on coefficient descriptors are shown to have comparable performance to a Hausdorff distance measure when retrieving trajectory-based motion clips but at substantially reduced computational cost. Experimental results are presented to illustrate the comparative performance of different matching metrics on real-world trajectory data collected by a retail store CCTV installation.

Index Terms—motion trajectory, video indexing and retrieval, object tracking, similarity metric

1. Introduction

Intelligent surveillance systems are assuming an increasingly important role in crime detection and prevention. This is evidenced by the growing number of installed camera networks. One of the most important tasks for the next generation of commercial CCTV surveillance systems is to automate the process of robustly tracking objects in complex and crowded environments.

Over the last two decades, the tracking problem has received extensive attention by the computer vision community [1]-[6]. 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 [7] where the system attempts to recognise user-predefined events such as certain types of potential criminal activity. An alternative is to analyse object motion paths [8]-[10] in order to learn and predict patterns of behaviour, or to allow users to create queries about the content of surveillance scenes [11]-[13], e.g. trajectory, colour, type of object, etc. and thereby retrieve useful information.

Object-based motion trajectory descriptors are known to be useful candidates for content-based video indexing and retrieval schemes [14]-[16]. The importance of selecting an appropriate trajectory model and similarity matching metric for trajectory-based querying of motion clips has received relatively scant attention.

The work presented in this paper aims to address this issue and further develops ideas propounded in [11],[21]. The application domain addressed is retail store surveillance which presents a number of challenging problems when attempting to search a large database of video motion clips. These include crowded scenes leading to multiple target tracking, static and dynamic occlusions, highly constrained (and thus many indistinguishable) motion paths and short or broken trajectories.

The remainder of the paper is organised as follows. We briefly outline the low-level tracking algorithm used to generate the raw trajectory data in section 2. The modelling process for obtaining smooth motion paths in described in section 3. In section 4, we derive a novel similarity metric for comparing polynomial-based trajectory models and review some existing matching metrics. Experimental results of a simulation study to compare the performance of our proposed metrics are presented in section 5. The matching matching metrics are then used to query a database of real-world trajectory-based motion clips collected by a retail store surveillance system. The paper concludes with a discussion and summary in section 6.

2. Acquisition of trajectory data

2.1. Foreground detection and object segmentation

The overall performance of the tracking algorithm largely depends on a robust and accurate foreground detection and object segmentation stage. We have found the adaptive background modelling technique based on [3] has provided reliable results. 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. It also adapts to any changes in the background even after the background model has been established.

Foreground detection is combined with a SAKBOT shadow detection model [17]. Shadows are detected by assuming that they reduce the intensity of the underlying pixel without having a significant effect on its colour.

2.2. Tracking via motion and appearance models

We deploy a motion model based on first-order Kalman Filter and a colour appearance model [18] using histogram intersection and backprojection [19]. The advantages of this approach are its speed and simplicity of representation.

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.

The overall structure of the tracking algorithm is illustrated in Fig. 1. For each frame in the video sequence:

1. Predict new position of tracked objects using motion model.
2. Calculate the most likely position of the object based on the prediction and the actual measurement associated with the object.
3. Use histogram backprojection technique to identify the location of the object centroid based on colour model. Use the additional information obtained to validate and adjust the object location.
4. Update the object state variable based on the object's most likely position.
5. Update the colour model for the object if it is not subject to static or dynamic occlusion.

Dynamic object occlusions are handled by an extra processing stage, further details of which can be found in [18].

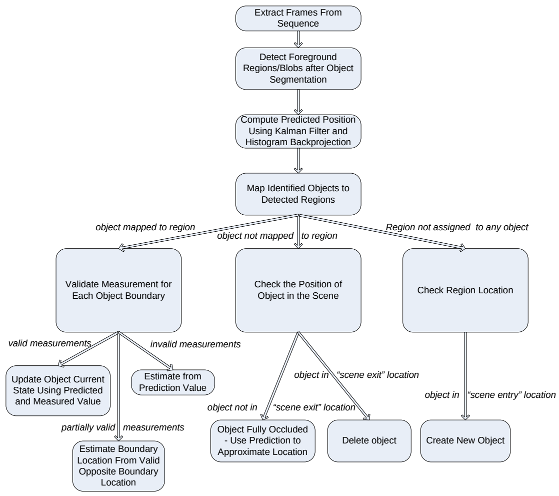


Figure 1. Overview of the low-level tracking algorithm

3. Motion trajectory modelling

3.1. Low degree polynomials using RANSAC

The output of the tracking algorithm is a set of noisy 2-D points representing each object's motion path over a sequence of frames. Initially, we propose to model the overall shape and speed of the object trajectory using a low degree polynomial. More complex trajectories are dealt with in section 3.2.

In the case of retail store surveillance scenes, the motion paths are highly constrained by the store layout, e.g. placing of shelves and racks, although individual customer behaviour can be erratic resulting in complex trajectory paths. At the simplest level of representation, it is found that most trajectories are adequately modelled by polynomials of degree 3 or less. Typically, the motion path of a tracked object is usually constrained so that one of the coordinates is either predominantly monotonically increasing or decreasing. This will be termed the *predominant coordinate*.

Given a set of n data points (x_i, y_i) (i = 1,2,..., n) the motion trajectory can be represented by a polynomial $P_m(x)$ of degree $m < n$ as

$$P_m(x) = a_0 + a_1 x + \ldots + a_m x^m \tag{1}$$

where x is the predominant coordinate. The unknown m+1 coefficients $\{a_i\}$ can be determined in a least squares (LS) sense by minimising the function E with respect to a_0, a_1, ...

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

The roles of x and y can be reversed in (2), where the y-coordinate is predominant.

It is well known that the least squares smoothing technique is highly sensitive to gross errors. Low-level tracking algorithms tend to produce noisy motion paths with a high degree of outliers resulting from mis-detected object points and multiple occlusions. To reduce the effect of noise contamination on motion path modelling, we have used a RANSAC technique [20]

RANSAC 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 standard smoothing technique such as LS to compute an improved estimate for the fit. The results of applying RANSAC to the trajectory data are shown in section 5.

The object trajectory speed is also modelled in a similar manner using low degree polynomials. We have found it more convenient to model the x, y components of velocity (V_x ,V_y) separately.

3.2. Chebyshev polynomials

For more complex trajectories, it is necessary to use a different basis function. In recent work, spatio-temporal trajectories have been successfully modelled using high order Chebyshev approximations [21], although there are other possibilities, e.g. radial basis functions, which are just as easy to compute and offer equally compact representations.

Given a set of time-ordered points (x_i, y_i, t_i) (i = 1,2,..., n), the motion path can be modelled as a spatio-temporal trajectory, i.e. as a 2-D time series. In the 1-D case, it can be represented by a function $f(t)$ expressed as a weighted sum of Chebyshev polynomials $P_k(t)$ up to degree m, defined as:

$$f(t) \approx \sum_{k=0}^{m} c_k P_k \tag{3}$$

where $P_k(t) = \cos(k \cos^{-1}(t))$ and

$$c_0 = \frac{1}{m} \sum_{k=1}^{m} f(t_k)\,,\ c_i = \frac{2}{m} \sum_{k=1}^{m} f(t_k) P_i(t_k) \tag{4}$$

for $t \in$ [-1,1] and i = 1,..., m. Further implementation details can be found in [18],[22]. In the next section, we show how to incorporate both shape and spatio-temporal descriptors into a coefficient-based similarity metric.

4. Similarity metrics for trajectory-based motion clip retrieval

As we wish to search and retrieve similar motion trajectories to an example (or sketch) query, it makes sense to index the video object motion clips in a database using our model-based descriptors. Each tracked and labelled video object is therefore indexed by a set of shape $\{a_i\}$ and $\{c_i\}$, and velocity coefficients $\{b_{xi}, b_{yi}\}$ representing the interpolated trajectory path. When a user invokes a query (motion path) which could be a free-hand sketch, a set of trend points marked on a representative background scene or a stored example, the coefficients are generated and compared to each of those in the database of clips. The closest matches are then retrieved in rank order according to some pre-defined measure of similarity. The speed with which the sketch query is drawn on a digital pad, or the proximity of points can be used to construct the $\{b_i\}$ coefficients. This should be normalised with respect to a typical time duration of object motion clips in the database.

We compare the performance of three different matching metrics; Coefficient Differencing [11], Root Mean Square (RMS) Integral Difference and Hausdorff Distance [10].

4.1. Coefficient differencing (CD)

This metric simply evaluates the Euclidean distance $d(M_q,M_k)$ between the coefficients of the query M_q and stored trajectory position models M_k as

$$d(M_q, M_k) = \left\{ \sum_{i=0}^{m} (a_{iq} - a_{ik})^2 \right\}^{1/2} \qquad (5)$$

where $M_q = \{a_{iq}\}$, $M_k = \{a_{ik}\}$ ($i = 0,...,m$) denote the coefficient sets for the query and stored models respectively. A similar expression for the velocity component gives

$$d(V_q, V_k) = \left\{ \sum_{i=1}^{m} (b_{x_{iq}} - b_{x_{ik}})^2 + (b_{y_{iq}} - b_{y_{ik}})^2 \right\}^{1/2} \qquad (6)$$

where $V_q = \{b_{iq}\}$ and $V_k = \{b_{ik}\}$ ($i = 0,...,m$) denote the coefficient set for the query and stored motion path speeds, and x, y subscripts represent seperate horizontal and vertical components of velocity.

The two terms are then combined to form the *CD* distance function d_{CD} as

$$d_{CD} = \alpha\, d(M_q, M_k) + \beta\, d(V_q, V_k) \qquad (7)$$

where α, β are weight parameters chosen according to the query type.

4.2. Root mean square integral differencing (ID)

A modified form of the *CD* similarity metric can be obtained using a RMS integral function. Given polynomial expressions for query and stored trajectory models, $P_q(.)$ and $P_k(.)$, the *ID* distance function d_{ID} can be defined as

$$d_{ID} = \sqrt{\frac{1}{l_2 - l_1} \int_{l_1}^{l_2} (P_q(.) - P_k(.))^2 \, dl} \qquad (8)$$

This can be evaluated as a closed form expression, e.g. for polynomials of degree $m = 3$, as

$$d_{ID} = \sqrt{\frac{1}{l_2 - l_1} \sum_{i=1}^{7} T_i} \qquad (9)$$

where $[l_1, l_2]$ is the interval over which the trajectory query is defined, and

$$T_1 = \frac{k_3^2}{7}(l_2^7 - l_1^7), \quad T_2 = \frac{k_3 k_2}{3}(l_2^6 - l_1^6)$$

$$T_3 = \frac{(2k_3 k_1 + k_2^2)}{5}(l_2^5 - l_1^5),$$

$$T_4 = \frac{(k_3 k_0 + k_2 k_1)}{2}(l_2^4 - l_1^4),$$

$$T_5 = \frac{(2k_2 k_0 + k_2^2)}{3}(l_2^3 - l_1^3),$$

$$T_6 = (k_1 k_0 + k_2^2)(l_2^2 - l_1^2), \quad T_7 = k_0^2 (l_2 - l_1)$$

$$k_i = (a_{iq} - a_{ik}) \qquad (i = 0,...,3)$$

In the case of complex trajectories when using high order Chebyshev approximations (typically $m \geq 25$), it is still possible to derive an equivalent expression to (9) using a computer algebra system such as *MAPLE*.

The *ID* metric represents a quantity related to the area between the trajectory curves normalised with respect to the length of the shorter curve. We can develop a similar expression to (9) using the coefficients of the velocity functions.

4.3. Hausdorff distance (HD)

The *HD* metric works directly with the raw pojnt data obtained from the output of the tracking process. Given two trajectories S and T, where $S = \{s_1, s_2,..., s_m\}$ and $T = \{t_1, t_2,..., t_n\}$ are discrete trajectory point sets, we define the *HD* distance metric d_{HD} as

$$d_{HD} = \min(h(S,T), h(T,S)) \qquad (10)$$

where
$$h(S,T) = \max_{s \in S} \min_{t \in T} d(s,t)$$
$$h(T,S) = \max_{t \in T} \min_{s \in S} d(t,s)$$

and $d(s,t)$ is the Euclidean distance from the position of point s in one trajectory to point t in the other trajectory. A similar expression can be derived to match velocity data sets for different trajectories.

The advantage of using *HD* in calculating the similarity between query and stored trajectories is that one can use the object-tracked points directly and it can cope with complex trajectory paths which cannot be adequately modelled with low degree polynomials. On the other hand, HD is expensive to compute and requires $O(mn)$ operations, where $m = |S|$ and $n = |T|$. It is also very sensitive to outlier points which can tend to dominate the distance calculation.

5. Results

5.1. Trajectory modelling using RANSAC

Typical output from the tracking algorithm and trajectory modelling process (up to a dynamic occlusion) is shown in Fig. 2. The difference between LS and RANSAC modelling is not apparent for smooth trajectories with few noisy points.

The advantage of using RANSAC over LS is demonstrated in Fig. 3 where the RANSAC result provides a more faithful representation of the motion curve. RANSAC implementation is represented by lighter grey curves whereas the black curve shows the curve fitting using simple LS.

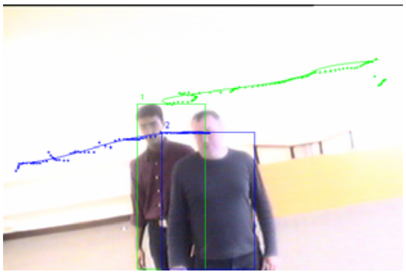


Figure 2. Motion trajectory modelling using LS polynomial fitting of degree 3.

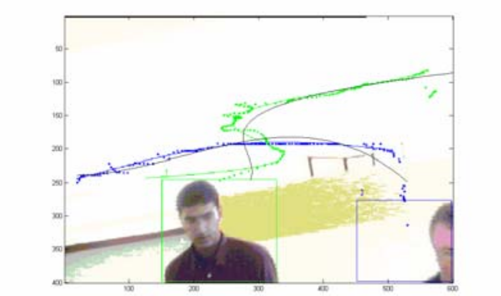


Figure 3. Comparison between LS and RANSAC modelling for trajectory paths through occluding frames.

5.2. Comparative evaluation of retrieval metrics: Simulation study

In order to compare the performance of the proposed similarity metrics *CD*, *ID* and *HD* on simple motion paths, we have generated 1000 simulated trajectories from an original set of 6 real tracking datasets. This ensures that each trajectory can be assigned to one of six categories for testbed evaluation using Precision (P) and Recall (R) measures. We use the standard definitions of precision and recall proposed in the information retrieval literature. The testbed allows us to establish the ground truth for determining the relevant results for each query.

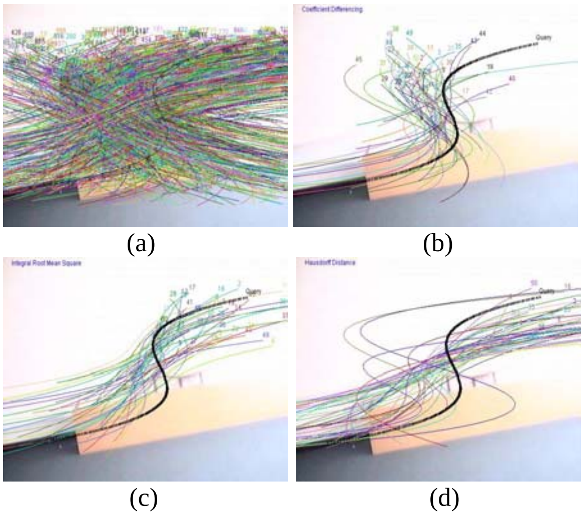


Figure 4. Comparison of results retrieved for an example query. (a) 1000 simulated trajectories. The top 5% of trajectories ranked closest to the example query using matching metrics: (b) CD, (c) ID and (d) HD.

Fig. 4 shows the results retrieved for the top 5% of trajectories ranked closest to an example query using each of the proposed matching metrics. In all simulation experiments we neglect the velocity term in (5), i.e. set $\beta = 0$. The example query is highlighted in black.

From a qualitative point of view, the apparent similarity of the retrieved results depend on the shape of the query example, although the *ID* metric appears to produce more consistent results over a number of examples. A more objective evaluation can be gained by plotting precision-recall (P-R) curves for a wider range of queries. The P-R curve is obtained by varying the size of the retrieved list and calculating the precision and recall values for each list and then averaging over all queries. The results are shown in Fig. 5.

The P-R curves suggest better performance for *ID* and *HD* matching metrics than for *CD*. The precision value for *CD* decreases more sharply than either *ID* or *HD* as the size of the retrieved list is increased. The *HD* metric slghtly outperforms ID, at the expense of increased computational cost.

Although computational burden is significantly reduced using CD and ID metrics, since they only require processing of stored coefficient values, there are obvious drawbacks. For low degree polynomial models, matching can only take place when the predominant coordinate is consistent for all trajectories. Clearly, this is not always guaranteed in practice. For more complex trajectories, it is not obvious how to extend the CD and ID metrics to segmented piecewise defined motion paths. However, this is not a problem for the model-free *HD* matching metric.

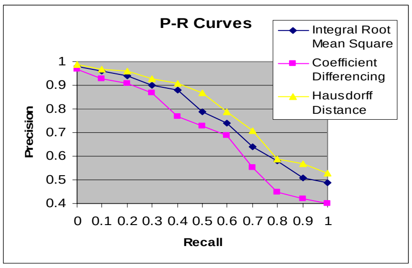


Figure 5. Precision-Recall curves for different matching metrics.

In the next section, we present some preliminary results of matching more complex spatial-temporal trajectories (as time series using high order Chebyshev approximations) to a query and qualitatively compare the retrieval performance using metrics based on coefficient descriptors and Hausdorff distance.

5.3. Retrieval results with real-world trajectory data

The motion sequences were obtained using a digital video recorder which was part of a retail store CCTV installation. A fixed CCTV camera was used to record 15 mins of video footage at 15 frames/sec and having resolution 352 by 288 pixels. A partial or complete spatio-temporal trajectory dataset was stored for each successfully tracked object in the sequence using the technique described in section 2. A sample subset of trajectories are shown in Fig. 6. When tracking has been lost due to static or dynamic occlusion after a short time interval, or when multiple overlapping objects cannot be resolved, the trajectory has been excluded from the database or stored only as a partial trajectory.

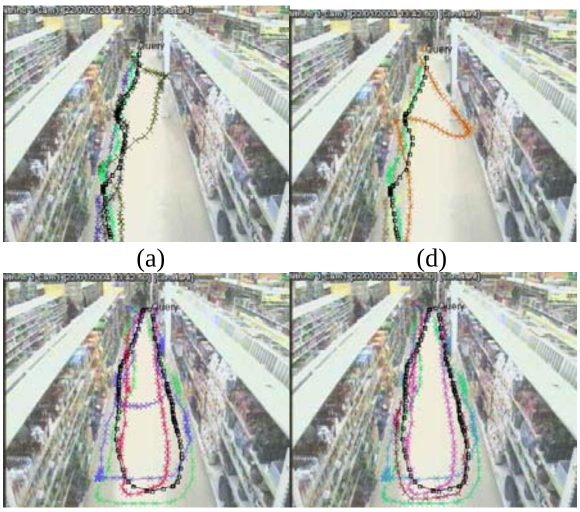
The motion clips stored in the database (size = 30) are then indexed using the coefficient descriptors derived from the Chebyshev approximation using m = 30 (see section 3.2).

Figs. 7(a)-(f) show the motion paths retrieved for various user-specified queries. The queries all contain stop/start motions observed in typical customer behaviour. The results indicate only those trajectories whose d_{CD} and d_{HD} values lie within a certain tolerance. The query curve is shown in black and the proximity of points denotes the object velocity, i.e. points that are closer together show the object moving slowly or stopping in the case of repeated points.

Figure 6. Sample subset of stored spatio-temporal trajectories.

The CD metric, although simple to compute, appears to give qualitatively similar results to HD, although these are dependent on the query type. In most cases, the top 3-4 retrieved results are the same for both metrics. Even taking into account the time taken to compute the polynomial approximation, the search time is drastically reduced when using CD rather than HD as a similarity metric.

These preliminary results show that it is possible to distinguish between those customers who are browsing products along the aisle (at specific locations) and those who are mainly using it as a thoroughfare.



(a) (d)

(b) (e)

(c) (f)

Figure 7. Retrieval results for 3 different user-specified spatio-temporal queries. (a)-(c) Highest-ranked matches based on CD metric (using Chebyshev approximation), (d)-(f) Highest-ranked matches based on HD metric.

6. Conclusions

A polynomial modelling approach is proposed for smoothing motion trajectories and using coefficient descriptors as a basis for content-based indexing and retrieval of video clips. Simple motion trajectories are modelled by least squares polynomial fitting incorporating a RANSAC technique for ensuring resistance to outliers. More complex spatio-temporal trajectories are described using high order Chebyshev polynomial approximations. In both instances, the coefficient descriptors are shown to be a useful index key into a database of video surveillance clips representing tracked objects. A user-defined query can be sketched as a means of searching this database and results are retrieved in rank-order of proximity to the query trajectory according to a matching metric.

The performance of three different matching metrics has been compared: Coefficient Differencing (CD), Hausdorff distance (HD) and a novel distance measure known as Root Mean Square Integral Difference (ID). Both position and speed can be combined in a unified way when formulating these distance metrics. On the basis of a simulation study, HD gives marginally better results than ID but at greater computational cost. Precision-recall results indicate that ID metric is an acceptable alternative to HD. The advantages are that the ID metric is simple and quick to compute, as the evaluation depends only on stored coefficient descriptors and query defined end-points. The disadvantage of using coefficient descriptors is that the monotonicity of the predominant coordinate has to be consistent for all trajectories in the database. This is difficult to achieve when modelling complex motion paths that are not adequately described by low degree polynomials.

In the case of real-world complex spatial-temporal trajectories, we have found that retrieval results obtained using similarity metrics based on coefficient descriptors (derived from high order Chebyshev polynomial approximations) give qualitatively similar rankings to Hausdorff distance metrics used previously.

We have compared two different similarity metrics (CD and HD) when querying a video surveillance database of motion clips derived from observing customers shopping in a busy retail store. Encouraging results have been obtained which suggest that users are able to search for motion clips that distinguish between customers' trajectories through the store.

In further work, we intend to evaluate the performance of these and other matching metrics on a larger, more comprehensive database of spatio-temporal trajectories involving more complex motion paths and tracking under severe occlusions. Other polynomial models, e.g. radial basis functions, offer the potential of improving compactness of representation when combined with the detection of path critical points and this will be explored in future.

7. 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, 2003, pp. 585-601

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

[4] 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

[5] 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.

[6] 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.

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

[8] D. Makris and T. Ellis, "Path detection in video surveillance," *Image and Vision Computing*, 20 (2002) 895-903

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

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

[11] 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.

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

[13] A. J. Lipton, J. Clark, P. Brewer, A. Chosak, P. Venetianer, "Object video forensics: Activity-Based Video Indexing and Retrieval for Physical security," in *Proc Intelligent Distributed Surveillance Systems* (IDSS-04), February 23, 2004, London, UK, pp. 56-60.

[14] C. Shim, J. Chang, "Content-based Retrieval Using Trajectories of Moving Objects in Video Databases", In *Proc. IEEE. 7th Intl. Conf. on Database Systems for Advanced Applications*, pp. 169-170, 2001.

[15] F. Bashir, A. Khokhar, D. Schonfeld, "Segmented Trajectory-based Indexing and Retrieval of Video Data", In *Proc. IEEE Intl. Conf. on Image Processing* (ICIP 2003), Barcelona, Spain, 2003.

[16] S. Jeannin and A. Divakaran, "MPEG-7 Visual Motion Descriptors," IEEE Trans. Circuits & Systems for Video Technology, vol. 11, no. 6, 2001, pp. 720-724.

[17] 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.

[18] A. Naftel and S. Khalid, **"**Video Sequence Indexing Through Recovery of Object Based Motion Trajectories**"**, *Proc. Irish Machine Vision and Image Processing Conference*, Dublin, Sept 1-3, 2004, pp.232-239.

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

[20] 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.

[21] Y. Cui and R. Ng, "Indexing Spatio-temporal Trajectories with Chebyshev Polynomials", *Proc. ACM SIGMOD*, Paris, France, June, 2004, pp.599-610.

[22] W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling, *Numerical Recipes in FORTRAN: The Art of Scientific Computing, 2nd ed.* Cambridge, England: Cambridge University Press, 1992.