Detecting Anomalous Motion Trajectories in the Coefficient Space

Abstract. This paper proposes a novel technique for clustering and classification of object-based video motion clips using spatio-temporal approximations. A Mahalanobis classifier is then used for the detection of anomalous trajectories. Motion trajectories are treated as time series data and modeled using the leading Fourier coefficients obtained by DFT. Trajectory clustering is then carried out in the coefficient feature space to discover patterns of similar object motion behaviour. The coefficients of the orthogonal basis functions are used as input feature vectors to a Self-Organising Map which can learn similarities between object trajectories in an unsupervised manner. Encoding trajectories in this manner leads to efficiency gains over existing approaches that use discrete point-based flow vectors to represent the whole trajectory. Experiments are performed on three different datasets – synthetic, vehicle and pedestrian tracking - to demonstrate the effectiveness of our approach. Applications to motion data mining in visual surveillance databases are envisaged.

1 Introduction

The current prevalence of video surveillance systems has prompted much research activity aimed at the development of sophisticated content-based visual data management techniques. General purpose tools are now urgently required for video event mining including discovery and grouping of similar motion patterns, detection of anomalous behaviour and object motion prediction.

Much of the earlier research focus has been on high-level object trajectory representation schemes that are able to produce compressed forms of motion data [1-11]. This work presupposes the existence of some low-level tracking scheme for reliably extracting object-based motion trajectories[12, 13]. The literature on trajectory-based motion understanding and pattern discovery is less mature but advances using learning Vector Quantization (LVQ) [14], Self-Organising Maps (SOMs) [15, 16], hidden Markov Models (HMMs) [17], and fuzzy neural networks [18] have all been reported. Most of these techniques attempt to learn high-level motion behaviour patterns from sample trajectories using discrete point-based flow vectors as input to the learning phase. For realistic motion sequences, convergence of these techniques is slow and the learning phase is carried out offline. This is due to the high dimensionality of the input feature space.

Related work within the data mining community on approximation schemes for indexing time series data is highly relevant to the parameterisation of object trajectories. However, computer vision researchers have been slow to adopt this work. For example discrete Fourier transforms (DFT) [19], wavelet transforms (DWT) [20], adaptive piecewise constant approximations (APCA) [21], and Chebyshev polynomials [22] have successfully been used to model and index spatio-temporal trajectories.

In this paper, we aim to apply time series modeling of spatio-temporal data to the problem of trajectory classification and show how to learn motion patterns by projecting the high-dimensional trajectory data into a low-dimensional manifold represented by a suitably chosen parameter feature space. The vector of parameters is used as an input feature vector to a neural network learning algorithm - a SOM - which can learn similarities between object trajectories in an unsupervised manner. It is shown that significant improvements in the accuracy of trajectory classification and recognition result when learning takes place in the coefficient feature space rather than the original trajectory space.

The remainder of the paper is organized as follow. We review some relevant background material in section 2. The system architecture and trajectory learning algorithm is presented in section 3 within the framework of a self-organising map. In section 4, the trajectory classification and anomaly detection procedure is discussed and experimental results for synthetic and object tracking data are reported in section 5. The paper concludes with discussion and proposals for further work.

2 Review of Background Work

Trajectory descriptors are known to be useful candidates for compressed representation of object motion in videos. Previous work has sought to represent moving object trajectories through a wide variety of direction schemes, polynomial models and other function approximations [1-10, 19-22]. The importance of selecting the most appropriate trajectory model has received relatively scant attention [11]. It is surprising to find that many of these candidate indexing schemes have not yet been applied to the problem of motion data mining and trajectory classification. Recent work has either used probabilistic models such as HMMs [17] or discrete point-based trajectory flow vectors [14, 16, 18] as a means of learning patterns of motion activity.

In [25, 26] an agglomerative clustering algorithm is presented, based on the Longest Common Subsequence (LCSS) approach, for grouping similar motion trajectories. Yacoob [27] and Bashir *et al*. [28, 29] have presented a framework for modeling and recognition of human motion based on a trajectory segmentation scheme. Classification is performed using Gaussian Mixture Model (GMMs) and HMMs for trajectory modeling that relies on PCA-based representation of segmented object trajectories. In [30], a semantic event detection technique based on discrete HMMs is applied to snooker videos. Various machine learning techniques used for classifying biological motion trajectories are compared in [31].

The contribution of this paper is to show that a trajectory-encoding scheme based on an input coefficient feature space can be used to learn motion patterns more effectively than previous approaches relying on discrete flow vectors. Clustering, classification and the detection of anomalous trajectories can then be carried out in the parameter space with reduced computational burden.

3 Learning Trajectory Patterns using Self-Organizing Maps

Self-organised maps (SOMs) have been previously used for motion activity classification [15, 16] with trajectories encoded as point-based flow vectors. However, the use of point-based feature vectors results in high dimensionality and reduced system efficiency. In this case, unsupervised learning of motion patterns takes place offline. To achieve dimensionality reduction, we consider object trajectories as motion time series and use a coefficient indexing scheme. A performance comparison of different motion indexing schemes can be found in [11].

An overview of the system architecture used for trajectory modeling and recognition is shown in Fig. 1.

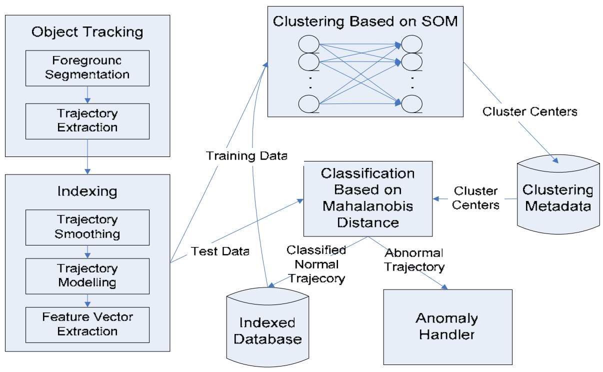

Fig. 1. Overview of system architecture for motion trajectory data mining.

3.1 Network Model

The network topology chosen for the SOM is a layer of input neurons connected directly to a single 1-dimensional output layer. Each input neuron is connected to every output neuron with the connection represented by a weight vector. A similar network model was proposed in [16] for learning vehicle trajectories as a means for accident prediction.

In a SOM network, physically adjacent output nodes encode patterns in the trajectory data that are similar and hence, it is known as a topology-preserving map. Consequently, common object trajectories are mapped to the same output neuron. The number of input neurons is determined by the size of the feature vector which in this case relates to the selected number of coefficients in the model. The degree of the polynomial can be chosen by setting a threshold on the maximum deviation of the approximation from the data or by examining the mean-squared error. The number of output neurons represents the number of distinct patterns in the trajectory data and is selected manually at present.

3.2 Learning Algorithm

The algorithm used to cluster the trajectories differs slightly from the original SOM proposed by Kohonen [23]. The number of output neurons is initially set to a value greater than the desired number of cluster patterns that we wish to produce. After training the network, clusters representing the most similar patterns are merged in an agglomerative manner until the cluster count is reduced to the target number.

Let B be the input feature vector representing the set of trajectory basis function coefficients, and W the weight vector associated to each output neuron. The learning algorithm comprises the following steps:

1. Determine the winning output node k (indexed by c) such that the Euclidean distance between the current input vector B and the weight vector W_k is a minimum amongst all output neurons, given by the condition

$$\| B - W_c(t) \| \leq \| B - W_k(t) \| \qquad \forall k \tag{1}$$

2. Train the network by updating the weights. A subset of the weights constituting a neighbourhood centred around node c are updated using

$$W_k(t+1) = W_k(t) + \alpha(t)\eta(k,c)(B - W_k(t)) \tag{2}$$

   where $\eta(k, c) = \exp(-|r_k - r_c|^2 / 2\sigma^2)$ is a neighbourhood function that has value 1 when k=c and falls off with distance $|r_k - r_c|$ between output nodes k and c, σ is a width parameter that is gradually decreased and t is the training cycle index.
3. Decrease the learning rate $\alpha(t)$ linearly over time.
4. After a pre-determined number of training cycles, decrease the neighbourhood size.
5. At the end of the training phase, merge the most similar cluster pairs until the desired number of groupings is achieved. Assuming W_a and W_b are the weight vectors associated with output neurons representing the most similar clusters, and m, n are the number of sample trajectories mapped to these neurons respectively, a new weight value W_{ab} for the merged cluster can be calculated as

$$W_{ab} = \frac{mW_a + nW_b}{m+n} \tag{3}$$

4 Trajectory Classification and Anomaly Detection

The SOM algorithm can be used to learn a set of motion patterns for the trajectory training dataset. The resulting labelled classes can then be used to classify unseen trajectory data as normal or abnormal. We use a simple k-NN classifier with the optimum value of k chosen by leave-one-out analysis. Classification results are presented in the following section using hand-labelled trajectories as ground truth. Visualisation of the clusters in the coefficient feature space shows that it is a reasonable assumptionto represent class conditional probability density functions as multivariate normal. Anomalous trajectories can be detected through analysis of the covariance structure of a pattern at each output node. Hotelling's T^2 test is used to determine if the Mahalanobis distance of a sample trajectory to its nearest class centre makes it abnormal. The test is now described in more detail.

Assuming that sample x belongs to pattern class Γ_i, where $\#\{\Gamma_i\}$ denotes the number of sample vectors x assigned to class Γ_i and $i = 1,..,K$. The class mean is denoted by and the covariance estimate is thus

$$\Sigma_i = \Sigma_{x \in \Gamma_i} (x - \mu_i)(x - \mu_i)^T / (\#\{\Gamma_i\} - 1) \quad \textbf{(4)}$$

where μ_i and Σ_i are calculated during training. The T^2 statistic based on the Mahalanobis distance can be calculated as

$$T^2 = \frac{n}{n+1} (x - \mu_i)^T \Sigma^{-1} (x - \mu_i) \quad \textbf{(5)}$$

where $n = \#\{ \Gamma_i \}$ and μ_i is the class centre to which the sample vector is closest. A hypothesis test can be conducted to determine whether x is 'too far' from μ_i and hence anomalous. Here p is the size of the input feature vector in the coefficient space and we have that

$$\alpha = P\left(T^2 > \frac{(n-1)p}{(n-p)} F_{p,n-p} \right) \quad \textbf{(6)}$$

where $F_{p,n-p}$ is a random variable with an F-distribution and p, n-p degrees of freedom. $F_{p,n-p}(\alpha)$ is the upper $(100\alpha)^{\text{th}}$ percentile of the $F_{p,n-p}$ distribution.

5 Experimental Results

We now present some results to indicate the effectiveness of the proposed clustering, classification and anomaly detection techniques performed in the coefficient feature space. The experiments have been performed on 3 different datasets – synthetic, vehicle and pedestrian object tracking data.

The synthetic dataset is known as the Cylinder-Bell-Funnel (C-B-F) data and has been widely used to benchmark time series data mining algorithms [33]. Examples from each of the 3 classes are shown in Fig. 2. We performed two tests on this dataset. First we compared the classification error using Euclidean distance similarity measure firstly on the original data points and then on the Fourier coefficient feature vector using DFT. Each time series was modeled with 9 DFT coefficients. The classification accuracy was determined using 1-NN and leave-one-out cross-validation. For each class, we generated n = 50 and 120 instances and averaged the result over 50 runs. The results presented in Table 1 illustrate that improved classification results from using DFT-coefficient indexing of time series over the raw point data vectors.

The classification errors resulting from unsupervised learning using *K*-means clustering to attach labels to 'unknown' classes are also shown for comparison.

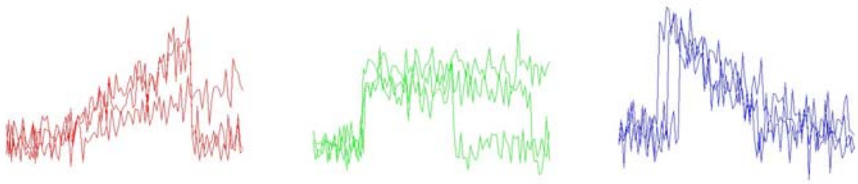

Fig. 2. Examples of the synthetic dataset of which there are 3 classes; bell (left), cylinder (centre), and funnel (right).

Table 1. Classification error based on Euclidean distance similarity measure using 1-NN leave-one-out analysis on synthetic time series dataset represented as point sequence and DFT coefficient feature vectors.

Model	% Error
Point sequence (n = 50)	3.2
DFT coefficients (n = 50)	2.0
Point sequence (n = 120)	2.5
DFT coefficients (n = 120)	0.8
Unsupervised DFT (n=100)	21.2
Unsupervised Point (n=100)	35.4

In the second experiment we tested the anomaly detection algorithm using a Fourier coefficient representation. It was not possible to use the point data as the covariance matrices for each class were near-singular and hence non-invertible. We generated a set of 100 sample coefficient vectors from each class and tested each instance against all the other classes seperately to determine the average number of detected anomalies. The mean and covariances of each class were generated from a sample size of 100 instances of each class and stored for each run. Again the reported results were averaged over 50 runs. We would expect that few instances drawn from class C would be recorded as anomalous when tested against the same class, whereas nearly all instances would be detected as anomalous when tested against a different class. The results demonstrating this fact are shown in Table 2. The results of best tests for the simulated dataset were promising and encouraged us to try out real motion trajectory data.

We now evaluate the performance of the SOM machine learning algorithm using the CAVIAR visual tracking database [24]. The database consisted of hand annotated video sequences of moving and stationary people and are intended to provide a test-bed for benchmarking vision understanding algorithms. The dataset is shown in Fig. 3.

Table 2. Percentage of instance vectors detected as anomalous. If the classes were completely separable the diagonal table entries would be zero and the off-diagonal entries would be 100.

Sample\Test class	Cylinder	Bell	Funnel
Cylinder	0.8	90.1	7.3
Bell	99.0	1.5	98.8
Funnel	98.9	100.0	0.3

The trajectories are indexed using DFT-based feature vectors with 9 coefficients for each spatial coordinate. We initially train a SOM network with 50 output neurons and then reduce these to 9 using the agglomerative clustering method described in section 3.2. Sample trajectories from the test set are then classified using classification technique described in section 4. The resulting trajectory cluster pattern are shown in Fig. 4. Visual inspection confirms that qualitatively similar motion trajectories have been clustered together quite successfully. Motions across the shopping mall corridor from left-to-right and right-to-left are grouped into separate clusters as expected. Sample trajectories that satisfy eq. (6) are identified as abnormal. These are shown highlighted in Fig. 5.

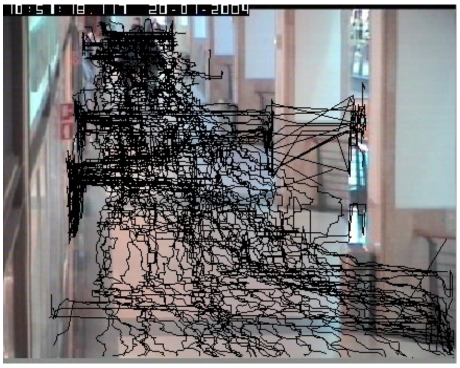

Fig. 3. Background scene containing database of hand-labeled object trajectories.

In order to visualise the effects of trajectory clustering in the transformed feature space, we perform Principal Component Analysis (PCA) on the DFT coefficient vectors. The first 3 PCs account for 94% of the total variability. The results are shown in Fig. 6. Each point represents an instance trajectory and these are colour/marker coded to highlight the separate cluster groups each trajectory is allocated to. These plots show a good degree of cluster separation in the low-dimensional PCA subspace.

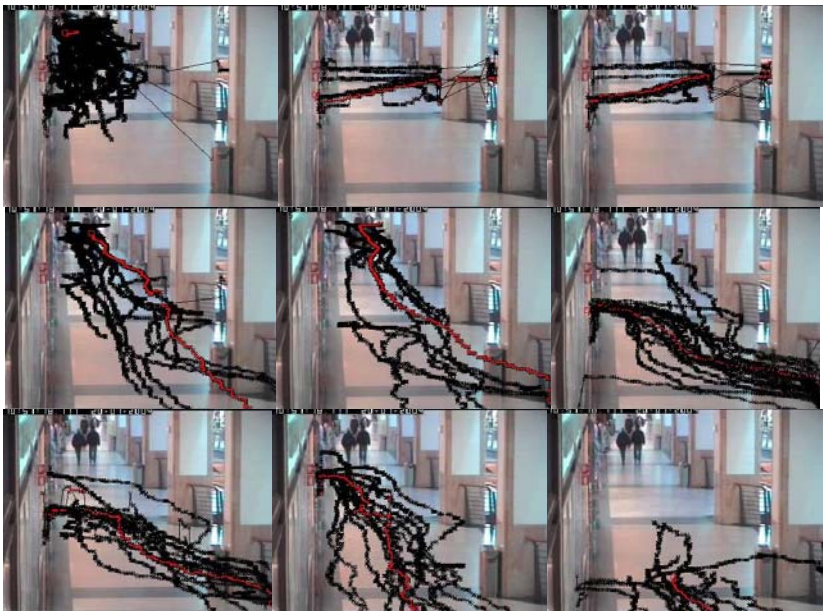

Fig. 4. Clustering of motion trajectories in CAVIAR dataset using SOM with DFT-based coefficient input feature vectors.

Fig. 5. Sample motion trajectories detected as anomalous using Hotelling's test with $P < 0.01$.

To investigate the effectiveness of clustering in the reduced coefficient feature space compared to clustering in the original space using trajectories encoded as discrete point-based flow (PBF) vectors, we performed some classification tests. The class labels of the motion trajectory patterns were learned using the SOM and K-means unsupervised techniques on the CAVIAR dataset. The dataset S was then randomly partitioned into training and test sets of equal sizes for cross validation purposes. We used a 1-NN classifier to classify all instance trajectories from the test set and generated the overall classification accuracy. To avoid bias, we repeated the random partitioning 500 times and averaged the classification errors over the test set. The results summarised in Table 3 demonstrate the superiority of learning trajectory patterns in the coefficient space. The classification accuracy obtained using coefficient feature space learning is higher than that of discrete PBF vector encoding for both SOM and *K*-means algorithms.

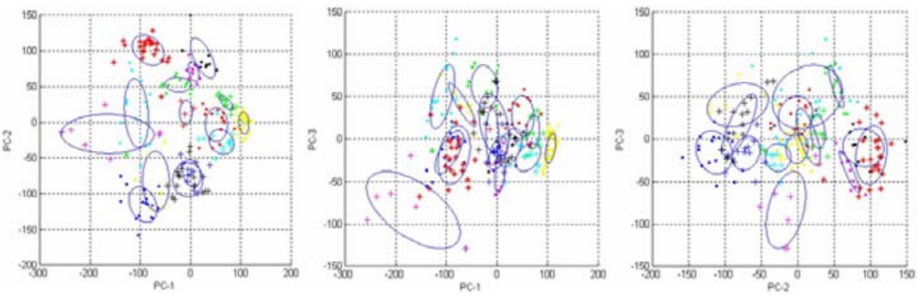

Fig. 6. Clustering visualised in the PCA subspace of DFT coefficient vectors. The plots represent PC_1 vs PC_2 (left), PC_1 vs PC_3 (center) and PC_2 vs PC_3 (right). Error ellipses (1-sigma contours) for the covariance matrix are shown for each cluster pattern. This data corresponds to that of Fig. 4.

Table 3. Comparison of mean overall classification accuracy for 2 different clustering techniques (SOM and *K*-means) and 2 different trajectory encodings (coefficient subspace and discrete PBF vectors). #classes : #trajectories = 9 : 111.

Method type	% Accuracy
SOM: coefficient subspace	91.9
SOM: PBF vectors	79.9
K-Means: coefficient subspace	88.8
K-Means: PBF vectors	85.6

In the next experiment we compare the performance of all 4 methods in trajectory classification and prediction. From the original set *S*, we define a set of partial trajectories S_P by sampling a subset of points from the original trajectories from 10% of the original length up to 100% in steps of 10. The classification is performed based on the input vectors consisting of coefficients or PBF vectors. This is repeated for the SOM and *K*-Means defined set of codebook vectors. Mahalanobis Distance is calculated between the input vector x representing the partial trajectory and cluster mean μ_i associated with *i*th output neuron. The input sample is classified to class *i* if:

$$\min_i[\log(\det(\Sigma_i)) + (x-\mu_i)^T \Sigma^{-1}(x-\mu_i)] \tag{7}$$

where Σ is covariance estimate and is calculated using eq.(4). A partial trajectory is said to be misclassified if it is not assigned to the original class based on the full trajectory set *S*. As evidenced from Fig. 7, the classifier derived from SOM in the coefficient subspace again outperforms *K*-Means. Furthermore, parameterized models prove more effective than point-based flow vectors in the trajectory prediction and classification task. These results give further impetus to the development of alternative dimensionality reduction techniques for learning and prediction of motion activity patterns.

As a final example, we applied the motion clustering and anomaly detection technique to vehicle motion tracking data. The results shown in Fig. 8 illustrate the utility of coefficient space methods for identifying clusters and outliers in the underlying data.

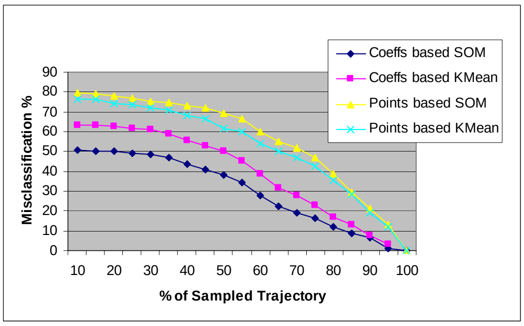

Fig. 7. Comparison of mean overall classification accuracy in trajectory prediction using 2 different clustering techniques (SOM and *K*-means) and 2 different trajectory encodings (coefficient subspace and discrete PBF vectors). #classes : #trajectories = 9 : 111.

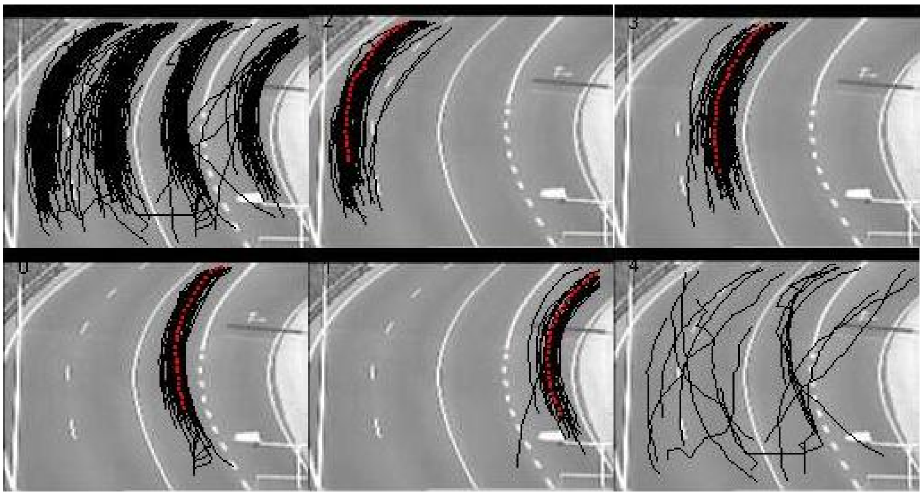

Fig. 8. Detection of abnormal vehicle trajectories using region-based tracker. Trajectoy indexing uses cubic polynomials of the form $x = P(y)$ modeled in the spatial domain only.

6 Discussion and Conclusion

This paper presents a neural network learning algorithm for classifying spatiotemporal trajectories. Global features of motion trajectories are represented well by polynomial approximations and this is apparent in the cluster visualizations. Using coefficients of basis functions as input feature vectors to a neural network learning algorithm offers an efficient alternative to the use of flow vectors for trajectory classification and anomaly detection. An efficient technique to differentiate between normal and abnormal time series data is proposed.

References

1. S-F. Chang, W. Chen, H.J. Meng, H. Sundaram, and D. Zhong, "A fully automated content-based video search engine supporting spatiotemporal queries," *IEEE Trans. Circuits Syst. Video Technol.,* vol. 8, no. 5, pp. 602-615, Sept. 1998.
2. S. Jeannin and A. Divakaran, "MPEG-7 visual motion descriptors," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 6, pp. 720-724, June 2001.
3. S. Dagtas, W. Ali-Khatib, A. Ghafor, and R.L. Kashyap, "Models for motion-based video indexing and retrieval," *IEEE Trans. Image Proc.*, vol. 9, no. 1, pp. 88-101, Jan 2000.
4. Z. Aghbari, K. Kaneko, and A. Makinouchi, "Content-trajectory approach for searching video databases," *IEEE Trans. Multimedia*, vol. 5, no. 4, pp. 516-531, Dec. 2003.
5. F. Bashir, A. Khokhar, and D. Schonfeld, "Segmented trajectory-based indexing and retrieval of video data," in *Proc. IEEE Int. Conf. Image Processing*, Spain, 2003, pp. 623-626.
6. C-T. Hsu and S-J. Teng, "Motion trajectory based video indexing and retrieval," in *Proc. IEEE Int. Conf. Image Processing*, pt 1, 2002, pp. 605-608.
7. F. Bashir, A. Khokhar, and D. Schonfeld, "A hybrid system for affine-invariant trajectory retrieval," in *Proc. MIR'04*, 2004, pp. 235-242.
8. C. Shim and J. Chang, "Content-based retrieval using trajectories of moving objects in video databases," in *Proc. IEEE. 7th Int. Conf. Database Systems for Advanced Applications*, 2001, pp. 169-170.
9. C. Shim and J. Chang, "Trajectory-based video retrieval for multimedia information systems," in *Proc. ADVIS*, LNCS 3261, 2004, pp. 372-382.
10. Y. Jin and F. Mokhtarian, "Efficient video retrieval by motion trajectory", in *Proc. BMVC'04*, 2004.
11. Anon. (withheld to preserve identity of authors)
12. L. Wang, W. Hu, and T. Tan, "Recent developments in human motion analysis," *Pattern Recognition*, vol. 36, no. 3, pp. 585-601, 2003.
13. W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Trans. Systems, Man & Cybernetic*, Part C, vol.34, no.3, pp. 334-352, August 2004.
14. N. Johnson and D. Hogg, "Learning the distribution of object trajectories for event recognition," *Image Vis. Comput.*, vol. 14, no. 8, pp. 609-615, 1996.
15. J. Owens and A. Hunter, "Application of the self-organising map to trajectory classification," in *Proc. IEEE Int. Workshop Visual Surveillance*, pp. 77-83, 2000.
16. W. Hu, X. Xiao, D. Xie, T. Tan, and S. Maybank, "Traffic accident prediction using 3-D model-based vehicle tracking," *IEEE Trans. Vehicular Tech*., vol. 53, no. 3, pp. 677-694, May 2004.
17. J. Alon, S. Sclaroff, G. Kollios, and V. Pavlovic, "Discovering clusters in motion time-series data," in *Proc. IEEE CVPR*, June 2004.
18. W. Hu, D. Xie, T. Tan, and S. Maybank, "Learning activity patterns using fuzzy self-organizing neural networks," *IEEE Trans. Systems, Man & Cybernetic*, Pt. B, vol. 34, no. 3, pp. 1618-1626, June 2004.

19. C. Faloutsas, M. Ranganathan, and Y. Manolopoulos, "Fast subsequence matching in time-series databases," in *Proc. ACM SIGMOD Conf.*, 1994, pp. 419-429.
20. K. Chan and A. Fu., "Efficient time series matching by wavelets," in *Proc. Int. Conf. Data Engineering*, Sydney, March 1999, pp. 126-133.
21. E. Keogh, K. Chakrabarti, M. Pazzani, and S. Mehrota, "Locally adaptive dimensionality reduction for indexing large time series databases," in *Proc. ACM SIGMOD Conf.*, 2001, pp. 151-162.
22. Y. Cui and R. Ng, "Indexing spatio-temporal trajectories with Chebyshev polynomials", in *Proc. ACM SIGMOD Conf.*, Paris, June 2004, pp. 599-610.
23. T. Kohonen, *Self-Organizing Maps,* 2nd ed. New York: Springer-Verlag, 1997, vol. 30.
24. CAVIAR: Context aware vision using image-based active recognition. (2004, Jan. 10). [Online]. Available: http://homepages.inf.ed.ac.uk/rbf/CAVIAR
25. Buzan D., Sclaroff S., Kollios G., "Extraction and Clustering of Motion Trajectories in Video," *InternationalConference on Pattern Recognition,* 2004.
26. Vlachos M., Kollios G., Gunopulos D., "Discovering Similar Multidimensional Trajectories," *Proceedings of the International Conference on Data Engineering,* p. 673, 2002.
27. Yacoob Y., Black M. J., "Parameterized Modeling and Recognition of Activities," *Computer Vision and Image Understanding,* Vol. 73 (2), pp. 232-247, Feb. 1999.
28. Faisal Bashir, Ashfaq Khokhar, Dan Schonfeld, "HMM-Based Motion Recognition System using Segmented PCA", *IEEE International Conference on Image Processing (ICIP 2005)*, Sept. 11 - Sept. 14, 2005. Genova, Italy.
29. Faisal I. Bashir, Ashfaq A. Khokhar, Dan Schonfeld, "Object Trajectory-Based Motion Modeling and Classification using Hidden Markov Models", *submitted to IEEE Transactions on Image Processing*.
31. Rea N., Dahyot R., Kokaram A., "Semantic Event Detection in Sports through motion understanding," *Proceedings of Conference on Image and Video Retrieval*, Dublin, Ireland, July 21-23, 2004.
32. Ivo F. Sbalzarinii, Julie Theriot, "Machine learning for biological classification applications", *Center for Turbulence Research, Proceedings of the Summer Program* 2002.