Classifying Surveillance Events from Attributes and Behaviour

P. Remagnino and G.A. Jones
Digital Imaging Research Centre
School of Computing and Information Systems,
Kingston University, Kingston upon Thames, KT1 2EE, UK
{p.remagnino,g.jones}@kingston.ac.uk
www.kingston.ac.uk/dirc/

Abstract

In order to develop a high-level description of events unfolding in a typical surveillance scenario, each successfully tracked event must be classified into *type* and *behaviour*. In common with a number of approaches this paper employs a Bayesian classifier to determine type from event attribute such as height, width and velocity. The classifier, however, is extended to integrate all available evidence from the entire track. A not untypical Hidden Markov Model approach has been employed to model the common event behaviours typical of a car-park environment. Both techniques have been probabilistically integrated to generate accurate type and *behaviour* classifications.

1 Introduction

The VIGILANT project aims to track in real-time all events within a typical surveillance video stream from a car-park scene, and store the associated pixel data in a highly efficient manner. This online process is complemented by an offline process scheduled for quieter periods of activity, which generates a classification of type and behaviour, a colour history, and a semantic 3D-trajectory description of the event. Both tracking and annotation processes ought to be achievable on a single typical single processor high-specification PC. These annotations are designed to support a video retrieval engine enabling retrospective human-oriented queries for forensic scenarios. The work described in this paper concerns the generation of accurate *type* and *behaviour* classifications from tracked events represented as a sequence of bounding boxes. The *type* classification is based on a simple Bayesian decision procedure extended to support the temporal integration of evidence. The *behavioural* classification employs the *hidden Markov model* technique to first build the required models of event activity and classify each new event trajectory. Crucially, integrating both approaches significantly enhances the classification accuracy of each technique. The interpretation of surveillance scenes typically entails the identification of moving regions of interest in the field of view of the camera used to monitor the environment. Only over the last ten years many researchers have developed tracking algorithms [6, 7, 1, 12].

BMVC 2001 doi:10.5244/C.15.70

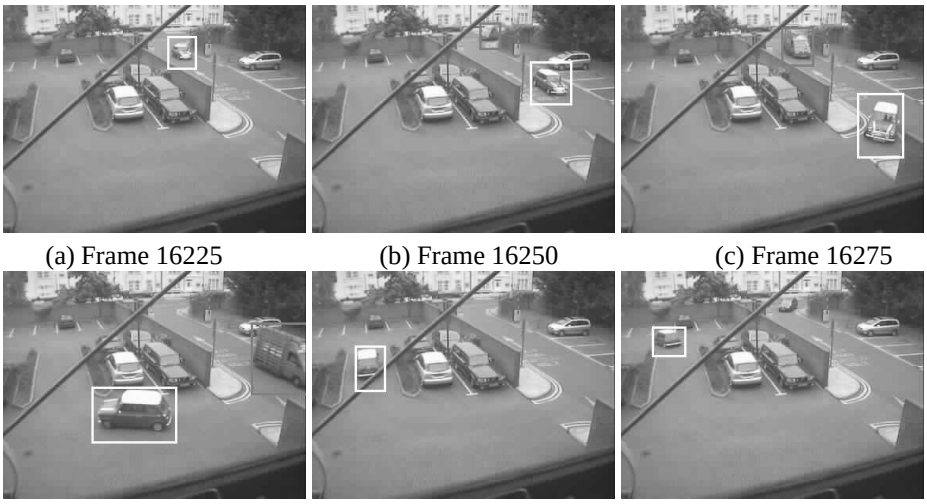
Machine Learning techniques such as the hidden Markov model have recently gained large success in the Computer Vision community. A model of the scene is far too complex to be precompiled, but it can always be learned, as long as sufficient data are available. A hidden Markov model (HMM) is doubly stochastic process, synthesizing both the underlying and observed phenomenon with a set of states and the transitions between them [5]. HMMs are generative models and can be used to recognise or classify new instances of the modelled phenomenon. Such characteristics perfectly match the requirements of scene interpretation. In vision, the HMM algorithm has been used with near [2, 4] and far field image sequences[8]. Exemplar applications using near field imagery include learning partial body models for American sign language[11], the generation of models for computer graphics animation[10], and the modelling of office dynamics against a vocabulary of typical actions [2]. Far field sequences have been used to build models of road traffic and people dynamics in well defined environments such as car park scenes[8]. The coupling of Markov models have also been studied with the purpose of building models of interacting events, such as encounters between pedestrians[9]. The standard HMM technique provides a set of algorithms to build a state space of recurrent variations within the stochastic process, but also means to update the model incorporating new acquired data, and to reproduce the process in all its variations[5].

Our contribution has been organised as follows. After a brief introduction to the application environment, section 2 describes and evaluates this initial object-type classification scheme that employs a relatively simple Bayesian classifier to integrate the event attribute information from the whole track. Section 3 introduces the HMM classifier, describing how the behavioural models are built from the Training data. In section 4, the classification results from this HMM technique are analysed. In addition, a simple method of integrating the results of the two techniques is described and the subsequent results assessed. Section 5 presents a critical appraisal of the presented work.

2 Object Classification

A surveillance test-bed has been installed overlooking a University car park. The pan, tilt and zoom cameras are pre-set with default positions monitoring the entrances with wide fields of view. In order to evaluate both the object classification and the behaviour classification algorithm described in this section and section 3.2 respectively, a large data set of 320,000 video frames was captured during busy arrival and departure periods over four days. This data set contains approximately 400 *Person* and 200 *Vehicle* events all entering, originating within or leaving the car park. A typical image sequence of these events is shown figure 1. In addition to these common events the data set contains roughly 50 *Other* less clear-cut events such as cyclists and large vehicles. This dataset is split into two equal sized *Training* and *Testing* data sets.

Once instantiated, each event must be classified into its object type and specific object behaviour from the image width and height of an object and its visual trajectory. This knowledge is derived from the *camera tracker*[6]. Examples of tracked vehicle objects are shown as bounding boxes in Figure 1. Classification and behavioural analysis is performed by the following algorithms.



(a) Frame 16225 (b) Frame 16250 (c) Frame 16275

(d) Frame 16300 (e) Frame 16325 (f) Frame 16350

Figure 1: Example of vehicle entering and manoeuvring through a car park. This ten-second event generated nearly 200 frames at a frame rate of 20 frames/second.

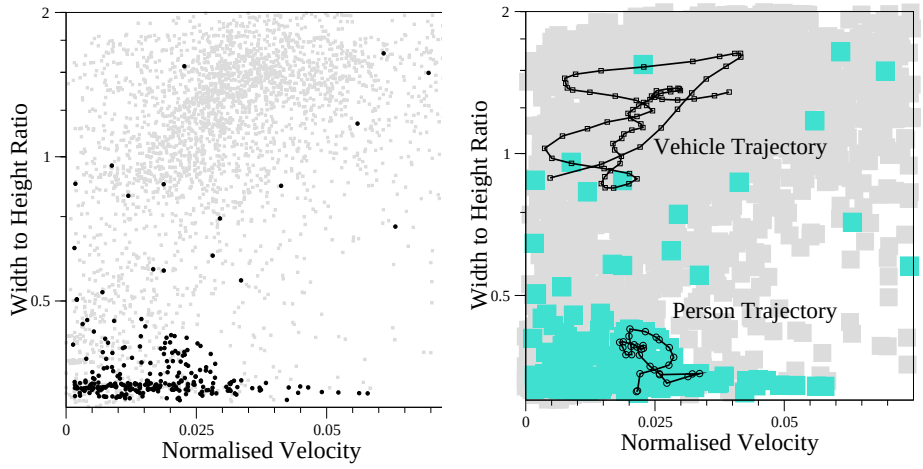
2.1 Object Classification

People and vehicles enjoy distinct velocity width-to-height-ratio characteristics. These are illustrated in Figure 2(a) by plotting the projected width-to-height-ratio of tracker observations against their estimated image velocity. The velocity estimates need to be normalised by the vertical image position of the observation to compensate for the fact that objects closer to the camera have approximately linearly larger visual velocities. These two class conditional probability density functions for the vehicle and people classes $p(\mathbf{a}|\omega); \omega \in \{Person, Vehicle\}$ are extracted from the training data as Normal distributions where $\mathbf{a}_t = (v_t, w_t)$ is the velocity v and width-to-height ratio w of an event at time t. The prior probabilities $P(\omega)$ capture the frequency of each event type.

Since to some extent these distributions are overlapping, it is necessary to integrate velocity and width-to-height observations over the history of the object to reduce the likelihood of false classification. This is illustrated in 2(b) by overlaying the object class PDFs with *trajectories* of a typical person and vehicle event. A simple *maximum a posteriori* decision rule is employed to update the probability of a classification given each new observation $\mathbf{a}_t$

$$\omega^* = \arg\max_{\omega \in \Omega} P\left(\omega|\mathbf{a}_t, \ldots, \mathbf{a}_{t_0}\right) \quad (1)$$

where W is the set of possible classifications $\Omega = person, vehicle$, and t_0 is the time at which the event started. Assuming each new observation $\mathbf{a}_t$ is independent of previous observations, the posterior probability $P\left(\omega|\mathbf{a}_t, \ldots, \mathbf{a}_{t_0}\right)$ may be expressed recursively



(a) Scatter Plot (b) Classification 'Trajectories'

Figure 2: (a) Scatter plots in the Width-to-Height Ratio versus Normalized Velocity classification space for Person (black) and Vehicle (grey) training data. Note separate though overlapping distributions. (b) 'Trajectories' in Width-to-Height Ratio versus Normalized Velocity classification space for two typical person and vehicle events.

$$\begin{aligned} P\left(\omega|\mathbf{a}_t,\ldots,\mathbf{a}_{t_0}\right) &\propto p(\mathbf{a}_t|\omega)P\left(\omega|\mathbf{a}_{t-1},\ldots,\mathbf{a}_{t_0}\right) \\ P\left(\omega|\mathbf{a}_{t_0}\right) &\propto p(\mathbf{a}_{t_0}|\omega)P\left(\omega\right) \end{aligned} \tag{2}$$

In addition to these two common classes, a number of atypical (in our dataset) event types exist including *cyclists* and *trucks*. Indeed car, van and truck events are not easily separable. Currently, the training data (for example Figure 1) has been manually separated into vehicle (cars and vans) and person classes, with all other events collectively represented as *Other*, and the classification set Ω extended to include this new label i.e. $\Omega = \{Person, Vehicle, Other\}$. To account for this other class in the classification equations, a uniform PDF has been assumed. Its prior $P(\omega)$is derived from the training data, while the constant $p(\mathbf{a}|\omega = Other)$ of the uniform PDF is determined empirically as that value yielding the best classification results on the unseen data set. From the training set $P(Vehicle) = 0.61$, $P(Person) = 0.31$ and $P(Other) = 0.08$.

2.2 Evaluating Object Classification

To evaluate the effectiveness of the classification algorithm, events extracted from the *Testing* training set are classified and compared with the correct manually determined classification. The results are presented in the *scatter matrix* in Table 1.

These results indicate that for *Vehicle* events, approximately nine-tenths of the events are correctly classified. The remaining incorrectly classified *Vehicle* events are as likely to

be classified *Person* as *Other*. The classification of *Person* and *Other* events is somewhat less successful with roughly four-fifths and two-thirds respectively correctly classified. In both cases, the incorrect classification is most likely to be *Vehicle*. Nonetheless, the 84% correct classification of the *Testing* dataset is significantly better than the 61% that would result using the largest prior probabilities alone. Moreover, the next section describes a behavioural classifier in which models are constructed for each event type *e.g. Person Entering, Vehicle Exiting*. Despite the imperfect results, we will show in section 4.2 that the above object classification algorithm will have a major impact on the accuracy of the later behaviour classification algorithm.

Scatter Matrix	Classification		
	Vehicle	Person	Other
Vehicle Event	89%	6%	5%
Person Event	17%	79%	4%
Other Event	28%	9%	63%

Table 1: Object Classification Results (Rows refer to the manually derived event classifications, while columns refer to the computed event classifications. Thus the top leftmost cell indicates that 89% of the *Vehicle* events have been correctly classified as *Vehicle*, while the top rightmost cell indicates that 5% of the *Vehicle* events have been incorrectly classified as *Other*).

3 Behaviour Classification

The Markov model is an ideal probabilistic technique for learning and matching activity patterns. Each type of activity for people or vehicle events may be characterised by a family of event trajectories passing through the image. Each family can be represented as a *hidden Markov model* in which states represent regions in the image, the prior probabilities measure the likelihood of an event starting in a particular region, and the transitional probabilities capture the likelihood of progression from one state to another across the image. Extracting clusters from the positional information of extracted event trajectories is the simplest way to build a set of Markov states. The choice of number of states generally depends on the type of scene. The larger the number of states the higher the danger of making the model too specific. The smaller the number of states the higher the danger of making one model indistinguishable from any other learned model. An *expectation-maximisation* (EM) algorithm [3] is employed to fit a number of Gaussian probability distributions (the states) to an *activity landscape* created from the set of all trajectory positions in the *Training* dataset. This learning phase is essentially automatic, requiring no user intervention other than the collection of training data over a period of time which includes all typical types of event and event behaviour e.g. a typical day.

3.1 Extracting Behaviour Dynamics

A behaviour HMM representation is composed of *states* (regions in the image), *prior probabilities* measuring the likelihood of an event starting in a particular region; the *transitional probabilities* capturing the likelihood of trajectory progressing from one region

to another across the image; and the *probability density function* of each state. During the training phase, these following object dynamics are computed from the same training data trajectories used to extract the set of N states $S_i, i \in [1, N]$.

Prior Probabilities The *prior probabilities* $\pi_i, i \in [1, N]$ for each state S_i represent the probability that a particular region S_i is the starting point for a trajectory. These probabilities are derived from the initial trajectory positions for each extracted event in the *Training* data set. In the case of the car-park scenario the image periphery is more likely to experience the beginning of an event, while the central region contains clusters indicating image regions where a driver people may leave their vehicle.

Transitional Probabilities The *transitional probabilities* a_{ij} capture the probability that a trajectory moves from one state S_i to another S_j given all possible transitions from that region. In the car-park scenario, for instance the transitions will mainly coincide with the main trajectories of vehicles and pedestrians. Absorbing states would indicate those events normally terminating in specific areas of the scene, typically either in the periphery of the image, or where vehicles are parked.

State Probability Density Function The probability distribution function (PDF) $b_j(\mathbf{o})$ represents the conditional probability of an position observation $\mathbf{o}$ of an event in state S_j. Currently the set of states for the hidden Markov models are extracted from the training set by clustering observations using the EM algorithm. This algorithm models these clusters as a Gaussian probability density function, and hence automatically generates the *state PDF i.e.* $\mathbf{o} \in N(\mu_j, \Sigma_j)$ where μ_j and Σ_j are the position mean and covariance of state S_j.

3.2 Behavioural Classification

Once the hidden Markov models for all required behaviour have been constructed they can be used to describe the dynamic evolution of the scene. We have constructed two behaviours for each object type i.e. vehicle-entering, person-entering, vehicle-exiting and *person-exiting*. For each new object detected within the scene, behavioural model selection can be performed by finding the behaviour $\lambda \in \Lambda$ from the set of possible behaviours Λ which yields the highest *a posterior likelihood* $P(\lambda|\mathbf{O})$ given a sequence of T trajectory observations of the event where $\mathbf{O} = (\mathbf{o_1}, \ldots, \mathbf{o}_T)$ *i.e.*

$$\lambda' = \arg\max_{\lambda \in \Lambda} P(\mathbf{O}|\lambda) P(\lambda) \tag{3}$$

Following Rainier[5], an HMM evaluation procedure for computing the model likelihood can be derived by introducing a random variable, $\mathbf{q}$, which represents a possible sequence of states explaining the observations $\mathbf{O}$ where $\mathbf{q} = (q_1, \ldots, q_T)$ represents the indices of the temporally ordered sequence of T states. Summing over all possible sequences (*i.e.* $\forall \mathbf{q}$) enables the conditional probability of the trajectory to be expressed as

$$P(\mathbf{O}|\lambda) = \sum_{\forall \mathbf{q}} [P(\mathbf{O}|\mathbf{q}, \lambda) p(\mathbf{q}|\lambda)] \tag{4}$$

The first term $P(\mathbf{O}|\mathbf{q},\lambda)$ measures the likelihood of the observations, $\mathbf{O}$, given both this explanatory sequence and the model λ. This probability may be estimated as the product of HMM positional likelihood terms for each of the observations $\mathbf{o}_1,\ldots,\mathbf{o}_T$.

$$P\left(\mathbf{O}|\mathbf{q},\lambda\right) = b_{q_1}\left(\mathbf{o}_1\right)\cdots b_{q_T}\left(\mathbf{o}_T\right) \tag{5}$$

The second term $P(\mathbf{q}|\lambda)$ of equation 4 measures the likelihood that the explanatory sequence $\mathbf{q}$ actually belongs to behaviour λ, and can then be easily calculated as the product the probabilities of all state transitions and the prior of starting in the initial state of the hypothesis S_{q_1} as follows

$$P\left(\mathbf{q}|\lambda\right) = \pi_{q_1} a_{q_1 q_2}\cdots a_{q_{T-1}q_T} \tag{6}$$

The most likely model is calculated using the classical forward iterative procedure provided by the HMM framework[5].

4 Results

In this car-park scenario, two specific types of event object are considered - *Person* and *Vehicle*. For both of the two specific classes, two basic behaviours are explored: *entering* and *exiting*. To construct the models for these, the *Training* dataset is partitioned into four sets of events to create the four corresponding HMMs - *vehicle-entering*, *person-entering*, *vehicle-exiting* and *person-exiting*. Both models associated with each object type will share the same set of states. In section 4.1 below, the behaviour models constructed from the *Training* datasets are evaluated against the *Testing* datasets. In addition, the appropriate number of states for this imagery is explored using the classification evaluation procedure. In section 4.2, the effect of integrating the object classification procedure described in section 2.1 into the behaviour classification is explored. Moreover, the behavioural analysis results are used to determine the object classification, and compared with the results of section 2.2.

4.1 Behaviour Classification

A key parameter when creating any HMM is determining the appropriate number of states. In many clustering applications, the optimal number of clusters would be determined by locating the mixture of Gaussians model that generated the best description of the modelled population. As the number of clusters increases, the higher the danger of modelling the specific training dataset. Too small a number of states and the higher the danger of modelling the underlying probability density function accurately. In this application, however, the population of trajectory position does not actually form clusters but rather manifolds around the visual trajectories of the principal vehicle and pedestrian thoroughfares on the image. Thus the choice of number of states generally depends on the type of scene and the distribution of events and trajectories in the field of view of the camera i.e. varies from image to image. Consequently, an additional training procedure is required.

To illustrate the effectiveness of the classification process described by equations 3 to 6, the models were tested against a set of test trajectories for the four HMM models each built with 5, 10, 15 and 20 states. The optimum number of states may be determined by

	Number of States			
	5	10	15	20
Percentage of Correctly Identified Behaviours	55%	64%	74%	75%

Table 2: Classification accuracy as function of number of HMM states.

	Estimated Behaviours			
	Vehicle		Person	
Ground Truth Behaviours	Entering	Exiting	Entering	Exiting
Vehicle-entering	76%	1%	23%	0%
Vehicle-exiting	4%	80%	1%	15%
Person-entering	27%	0%	73%	0%
Person-exiting	2%	26%	5%	68%

Table 3: Behaviour Accuracy

inspecting the classification accuracy as illustrated in Table 2 where each entry details the percentage of correctly identified events for all types of behaviour.

As the number of states used to model the activity increases, the classification accuracy rises. For the 5-state model, the EM algorithm has poorly modelled the activity resulting in the essentially random classification. Accuracy can be significantly improved by including greater numbers of states. Negligible gains are achieved as the number increases beyond 20 states. Indeed at this point there is an increasing likelihood of *over-training* in which the HMM no longer generalises but rather begins to model the specific training set. The ideal model, therefore, will have 15 states representing a trade-off between accuracy and computational cost of evaluation. This procedure for determining the number of model states may be refined to allow the optimum to vary for each model. A break down of the behaviour accuracy per type of activity is given in as a *scatter matrix* in table 3 for the 15-state model. Note that while the models are good at distinguishing between *entering* and *exiting* behaviours, there is a significant level of *cross-talk* between the *Vehicle* and *Person* classes.

4.2 Integrating Event and Behaviour Classification

Rather than relying on the prior probabilities $P(\lambda)$, the HMM classification procedure described by equation 3 can achieve greater behavioural classification accuracy by using the previously computed event classification probability $P\left(\omega|\mathbf{a}_t,\ldots,\mathbf{a}_{t_0}\right)$ derived in equation 2 of section 2.1 which enables the classification procedure to directly influence the selection of the appropriate behavioural model as follows

$$p(\lambda|\mathbf{O},\mathbf{a}_t,\ldots,\mathbf{a}_{t_0}) \propto p(\mathbf{O}|\lambda)p(\lambda|\omega)P\left(\omega|\mathbf{a}_t,\ldots,\mathbf{a}_{t_0}\right) \tag{7}$$

where $p(\lambda|\omega)$ is the conditional probability of a particular behaviour λ given the classification ω of the event. These probabilities are again derived from frequency analysis of behaviours and objects in the *Training* dataset. Table 4 shows the classification *scatter matrix* representing a breakdown of the behaviour accuracy per type of activity for the 15-state model. Note that in comparison to table 3, the use of the attribute evidence

$p(\lambda|\omega)P\left(\omega|\mathbf{a}_t,\ldots,\mathbf{a}_{t_0}\right)$ rather than the prior $p(\lambda)$ dramatically improves the *Behaviour* classification accuracy.

	Estimated Behaviours			
	Vehicle		Person	
Ground Truth Behaviours	Entering	Exiting	Entering	Exiting
Vehicle-entering	98%	0%	2%	0%
Vehicle-exiting	1%	95%	0%	4%
Person-entering	6%	0%	94%	0%
Person-exiting	0%	9%	2%	89%

Table 4: Improved Behaviour Accuracy

The event type (*i.e. Vehicle* or *Person*) associated with the selected behavioural model can be used to finally determined the event type. Table 5 compares the event classification results for each of these techniques. While the cross-talk of the HMM behavioural analysis is significant - see Table 5 column (b) - once combined with the more accurate results of attribute-based classification (column (a)), the final algorithm classifies an impressive 95% of the events correctly - column (c).

	(a) Classification from Attributes (section 2)		(b) Classification from Behaviour (section 3.2)		(c) Combined Classification	
Ground Truth	Vehicle	Person	Vehicle	Person	Vehicle	Person
Vehicle	91%	9%	78%	22%	97%	3%
Person	19%	81%	29%	71%	9%	91%

Table 5: Comparison of Event Classification Techniques

5 Conclusions

The VIGILANT project aims to provide real-time storage and annotation of surveillance video-streams, and image retrieval based on human language oriented queries for untrained security operators. Crucial to this goal is the classification of TYPE and BEHAVIOUR of events within the video stream. Currently in this car-park scenario, we have restricted the principal types of event to Person and Vehicle classifications, and the behaviour models to Entering and Exiting activities. This paper investigates a number of solutions to this problem. First, in section 2.1, a MAP based type classification scheme is described based on the temporal integration of the width, height and velocity attributes of each tracked event. Second, in section 3.2, the classification of event behaviours is tackled using the Hidden Markov Model approach: a tool ideally suited to the modelling of complex temporally extended events. Finally in section 4.2, the two techniques are integrated to improve both TYPE and BEHAVIOUR classification - an effective approach clearly demonstrated by the results presented in Table 4 and Table 5.

No actual comparative work with other techniques has yet been undertaken partially due to the lack of adequately reported work which adopts a similar approach of temporally integrating evidence from tracked events. A more fundamental problem with the approach

- particularly in the context of the VIGILANT project goal of eliminating specialists from the installation process - is the difficulties involved in building the behavioural models, which currently require a large amount of manually classified tracked events. A second major weakness is the rather crude TYPE and BEHAVIOUR classes currently modelled i.e. Person, Vehicle, Entering and Exiting. To be effective, a much richer range of classifications is required. Nonetheless, the now validated approach of integrating attribute and trajectory information is expected to underpin future developments of this work.

References

[1] M. Bogaert, N. Chleq, P. Cornez, C.S. Regazzoni, A. Teschioni, and M. Thonnat. "The PASSWORDS Project". In *Proceedings of International Conference on Image Processing*, pages 675–678, 1996.

[2] M. Brand. "Learning concise models of human activity from ambient video". Technical Report 97-25, Mitsubishi Electric Research Labs, 1997.

[3] V. Cadez, S. Gaffney, and P. Smyth. "A General Probabilistic Framework for Clustering Individuals and Objects". *Proceedings of ACM*, August 2000.

[4] S. Gong, S. McKenna, and A. Psarrou. *"Dynamic Vision: From Images to Face Recognition"*. Imperial College Press, 2000.

[5] L.Rabiner and B-H. Juang. *"Fundamentals of Speech Recognition"*. Prentice-Hall, 1993.

[6] J. Orwell, P. Remagnino, and G.A. Jones. "From Connected Components to Object Sequences". In *First IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, pages 72–79, 2000.

[7] C.S. Regazzoni and A. Teschioni. "Real Time Tracking of non rigid bodies for Surveillance applications". In *ISATA Conference*, Firenze, 1997.

[8] P. Remagnino, J. Orwell, and G.A. Jones. "Visual Interpretation of People and Vehicle Behaviours using a Society of Agents". In *Congress of the Italian Association on Artificial Intelligence*, pages 333–342, Bologna, 1999.

[9] B. Rosario, N. Oliver, and A. Pentland. "A Synthetic Agent System for Bayesian Modeling of Human Interactions". In *Proceedings of Conference on Autonomous Agents*, pages 342–343, 1999.

[10] A.D. Wilson. "Luxomatic: Computer Vision for Puppeteering". Technical Report 512, MIT Media Laboratory Perceptual Computing Section, 1997.

[11] A.D. Wilson and A.F. Bobick. "Parametric Hidden Markov Models for Gesture Recognition". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(9):884–900, 1999.

[12] C.R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland. "Pfinder: Real-time Tracking of the Human Body". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780–785, July 1997.