Non Parametric Classification of Human Interaction

Scott Blunsden, Ernesto Andrade and Robert Fisher

Institute of Perception Action and Behaviour, School of Informatics, University of Edinburgh, UK

Abstract. This paper presents a non parametric method for classifying interactions between two people as taken from a video camera. A nearest neighbour classifier that uses trajectory matching in feature space is introduced and shown to be superior to regular nearest neighbour classification for this problem.

1 Introduction

Many previous attempts have been made at identifying individual human activity, however only recently has the question of identification of interactions been addressed [9, 3, 8, 5, 6]. The classification of multi party interactions is necessary as there are many situations which can only be understood by considering the relationships between persons. For example the idea of 'meeting' cannot be sufficiently expressed or recognised when one only considers a single person in isolation. By considering interactions it is possible to build upon previous work on human motion understanding [4, 1] to build a richer picture of what is going on in a scene.

Within this paper a non parametric approach is taken which can work with few examples of a particular interaction. The classification method is described and then results as applied to interacting pedestrians are presented. It was found that if temporal dependencies are taken into account a relatively simple classification method can improve the classification performance. First a brief review of previous work in the area is undertaken.

2 Previous Work

Previous work upon the identification of interaction has been undertaken, most notably by Oliver et al. [5] who trained coupled hidden Markov models to recognise six different types of interaction between people. The models were also capable of being 'primed' with synthetic interactions to give improved performance upon real interactions. Xiang and Gong [9] also used coupled hidden Markov models to automatically build relationship models between vehicles on an airport runway. The graphical model approach was also taken by Intille [3] who used a hand crafted Bayesian network to identify pre-defined plays within the game of American football.

More recently Sato and Aggarwal [8] have also tackled this problem from the two person case. In order to classify interaction types only cases where people were within close proximity were considered for classification. The nearest mean method was then used for classification of the interaction with good results. Recent work by Park and Aggarwal [6] has also focused on two person interactions where the people can be segmented into a parts. A hierarchical Bayesian network is then used for classification of interactions.

For multi person interactions a hidden Markov model with multiple inputs was discussed in [2]. A role variable was introduced to take account of permutations of the roles people may play in an interaction. This method was shown to work successfully with three person interactions.

3 Classification

Within this paper we take a non-parametric view of modelling and classification. Such an approach has the benefits of not requiring large amounts of data in order to obtain the parameters of a model. The sparsity of certain types of interaction along with the difficulty in obtaining and processing such video was a reason for choosing such an approach.

3.1 Feature Extraction

Throughout each video sequence every moving person was tracked and their bounding box was established. This gave the 2D position of the person in the image plane. This position was projected into ground plane co-ordinates using a homography. Ground plane coordinates were used as they help to normalise distances and speed with respect to the distance from the camera, thus enabling a fairer comparison throughout all image positions. Here $\mathbf{x}_i^t$ is the 2D vector which contains the ground plane coordinates for person i at time t.

From this point-set several features are calculated. The speed s_θ^t of each person is calculated as shown in equation 1. The reason for the w term is due to the high frame rates many surveillance cameras are capable of, typically around 25 fps. This high frame rate means that there is often very little movement between subsequent frames with a high proportion of this movement being a result of noise from the tracking process. Throughout all experiments w was set to 25 (about 1 second).

The normalised direction is calculated as shown in equation 2. This measure is not used directly in the output feature vector but is used to calculate the alignment ($al_{i,j}^t$) between person i and j, as given in equation 3. Alignment is calculated as the dot product between the two normalised directions $\hat{\mathbf{v}}_i^t$ and $\hat{\mathbf{v}}_j^t$).

$$s_\theta^t = \left\| \mathbf{x}_\theta^t - \mathbf{x}_\theta^{t-w} \right\|, \theta \in i, j \tag{1}$$

$$\hat{\mathbf{v}}_\theta^t = \frac{\mathbf{x}_\theta^t - \mathbf{x}_\theta^{t-w}}{\left\| \mathbf{x}_\theta^t - \mathbf{x}_\theta^{t-w} \right\|}, \theta \in i, j \tag{2}$$

$$al_{i,j}^t = \hat{\mathbf{v}}_i^t . \hat{\mathbf{v}}_j^t \tag{3}$$

$$d_{i,j}^t = \mathbf{x}_i^t - \mathbf{x}_j^t \tag{4}$$

$$d\_dif_{i,j}^t = d_{i,j}^{t-w} - d_{i,j}^t \tag{5}$$

$$d\_sp_{i,j}^t = \left\| s_i^t - s_j^t \right\| \tag{6}$$

$$of_\theta^t = \left\| \mathbf{x}_\theta^t - \mathbf{x}_\theta^{start} \right\|, \theta \in \{i, j\} \tag{7}$$

Difference in position ($d_{i,j}^t$, equation 4) is also used as a feature along with the change in distance at two separate time steps as given in equation 6. The offset (of_θ^t) from a starting position, given in equation 7 was calculated for both persons i and j. The difference in speed $d\_sp_{i,j}^t$ between the two persons is given by equation 6.

This gives an eight dimensional final feature vector at time t between people i and j, shown in equation 8. The features have a degree of invariance in that they do not depend upon the absolute direction or position of a person within the scene. Each feature was also scaled to have zero mean and unit variance. The mean and variance were obtained from the training set only.

$$\mathbf{f}_{i,j}^t = \left[s_i^t, s_j^t, al_{i,j}^t, d_{i,j}^t, d\_dif_{i,j}^t, d\_sp_{i,j}^t, of_i^t, of_j^t \right] \tag{8}$$

3.2 Classifier

Once the trajectories had been obtained sequences were manually labelled as containing an interaction or non interaction. The type of interaction was also manually assigned from a restricted vocabulary (see section 4). Sequences ranged in length from a few seconds to several minutes. For every frame of these labelled interactions the features (described in section 3.1) were calculated. Using the feature vector described in the previous section a nearest neighbour classifier is used with a neighbourhood whose size was empirically set to 5. Data was partitioned into a training and testing set with a 50/50 split. The test data was not used in any way until evaluating the performance of the classifier.

It is important that these training and test sequences are complete sequences and as such are completely separate from one another. If only points are taken at random (rather than the complete sequence) then there is a high similarity between the two sets and they are in fact temporally dependent upon one another. The problem may then simply reduce to a simple interpolation procedure.

A simple strategy to classifying a point in such a sequence would be to classify a novel test point based upon its proximity to a training point as measured through some distance metric. However, as illustrated in figure 2 there is a temporal dependency between points. It is visible that the classes create a trajectory in feature space. Point by point matching such as nearest neighbour or clustering will miss this dependency. In order to take account of the trajectory 'shape' of the data the Hausdorff distance (as given in equation 10), was used to compare training and testing samples over a temporal window.

The feature vector for each person at time t is made up of the extracted features given in (8). In order to classify the activities of the tracked person at a

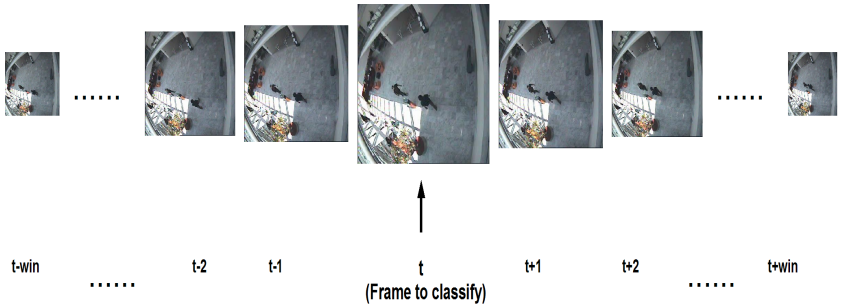


Fig. 1. Extraction of frames to be used within the classification process

given time a temporal window of size *win* around the current frame is taken. In these experiments the size of *win* is set to 25, meaning that the total size of each sample used for matching is 51 time steps in length. This process is illustrated in figure 1. With a video rate of 25 frames per second achievable on many surveillance cameras this is equivalent to watching one second worth of video either side of the frame. This step is taken as when comparing interactions such as two people meeting compared to simply walking past another it is necessary to watch a few seconds to determine what is happening. This is evident in figure (2) where many points are overlapping in feature space. By watching a few seconds of video it is possible to distinguish between interaction types. Results for a measure which only takes into account distance from a single point in time (and not a temporal window) are given as a comparison in the results section.

For classification a k=5 nearest neighbour classifier is used. Three distance measures are compared, the standard squared distance over a window (given in equation 9), the Hausdorff distance (equation 10) and a single point distance cost function.

$$d\left(A,B,win\right) = \sum_{i=1}^{2win+1} \left\|A_i - B_i\right\|^2 \tag{9}$$

$$h(A,B) = \max_i \left\{ \min_j \left\{ \|A_i - B_j\|^2 \right\} \right\} \tag{10}$$

The matrix A is from the training database where each column contains a feature vector as given in 8, centred around the training frame. Matrix B is the novel test point and again its feature points are stored column-wise centred around the current frame. In both cases i and j refer to the whole column vector containing the calculated features as given in 8. The size of this matrix is determined by the choice of *win*. For instances where no window is considered then *win* can be set to 0.

Equation 9 is the sum of the squared distance between all points in the two matrices. The Hausdorff distance is given in equation 10.

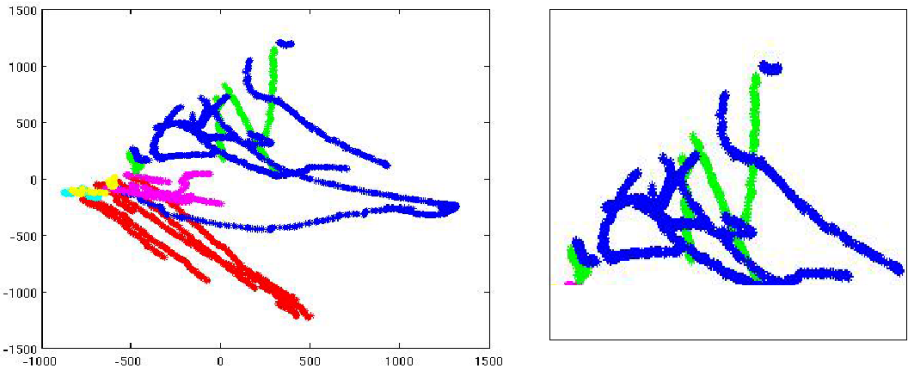


Fig. 2. Plot of first two principle components of the data. The colours refer to the class of data with walking together - red, approach - green, ignoring - blue, meeting - cyan, split - magenta and fight being yellow. The zoomed area on the right shows the ignoring and approach interactions

4 Results

The results of the nearest neighbour classification scheme are now presented. The data was generated from the publicly available CAVIAR project [7]. The interaction classes along with the number of samples are given in the table below. It can be seen that the distribution of interactions is not uniform. As well as having an uneven prior distribution in a real world case such as in surveillance it is also likely that the number of ignore classes would be much higher.

We include the ignore class here as for any practical application of the method would have to distinguish when people are not interacting. For every frame in each sequence the features given in section 3 were calculated, thus giving a 9 dimensional feature vector at each time step. Sequence length ranged from a few seconds to several minutes.

For the first experiment classification of individual points was undertaken. We are in effect asking the computer to "tell me what you think is happening in every frame". In total there where 2230 test points (with a temporal window size of 51 frames). Results of classification by this method are given in table 4. It can be seen that Hausdorff distance, which takes into account the shape of the temporal window in feature space performs better than those that don't. It is also visible to see that both distance measures that use a temporal window perform better than when a single point is used.

Class	Num. Samples	Dist Measure		
		$d(A,B,win)$	$d(A,B,0)$	$h(A,B)$
Walk together	700	100	99.9	100
Approach	145	36.6	26.9	46.9
Ignore	835	80.7	73.9	85.1
Meet	382	100	61.5	100
Split	147	100	87.1	100
Fight	21	61.9	61.9	57.1
Total	2230	88.2	77.62	90.8

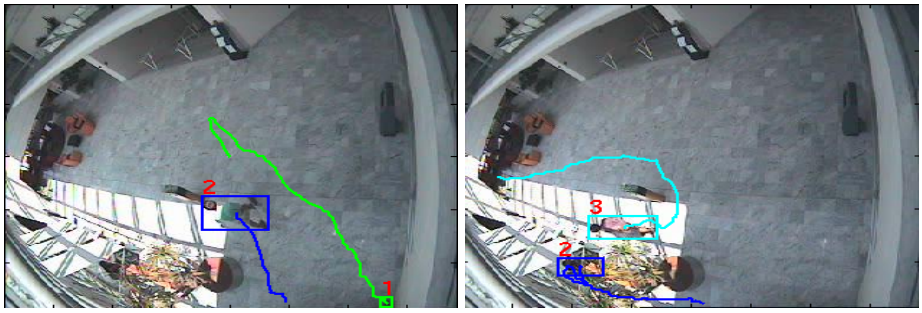
4.1 Complete Sequences

A second experiment was also conducted to test performance of the classifier when a contiguous video stream of pre-segmented data was given to it. The question we are asking here is "If I show you a video clip of arbitrary length tell me what the clip was about". Sequences were manually pre-segmented, continuous and contained only one type of interaction throughout their duration. The number of samples are given in the table below. In order to classify a complete sequence each point in the sequence was classified as described in the previous section. The most frequent class label was then assigned to be the class of the complete sequence. The idea of this test was to see how well the algorithm would perform in situations where longer sequences needed to be classified such as in annotating surveillance data. It is also a good test of how predictable single frame classifications are over longer sequences.

Class	Num. Samples	Dist Measure		
		$d(A,B,win)$	$d(A,B,0)$	$h(A,B)$
Walk together	7	100	100	100
Approach	4	25	0	50
Ignore	6	83.3	100	100
Meet	1	100	100	100
Split	2	100	100	100
Fight	1	100	100	100
Total	21	80.9	80.9	90.4

4.2 Classification Summary

For classifying both complete sequences and individual points the Hausdorff method proves the best distance measure. However certain classes, such as approach which had an accuracy rate of 46.9%, prove difficult to classify and are frequently confused with ignore. This is understandable as many times when two people ignore one another they may get closer as they move through the scene but do not actually interact in any way. Situations like this could be differentiated by using longer term observations and delaying the decision until the classifier is more certian of an interaction. Situations such as these are illustrated in figure 3.



(a) (b)

Fig. 3. Plots of trajectory information for (a) Confusing ignoring with approaching. (b) Difficulties when obstacles are in the scene causing mis-classification. Here person 3 has to go round the obstacle to approach person 2. This also illustrates how you may have to watch a sequence for longer to figure out the actual intention of a person. Lines show tracked points from previous timesteps.

There is also the problem of obstacles within real scenes. Such obstacles can lead to misclassification as the trajectory and the resulting features can seem to veer 'off course' from what one would expect. For example in figure 3(b) the person modifies the approach of another due to an obstacle being in the way.

There are still some problems where two classes physically look like one another (such as approach and ignore) and would even fool a human observer given this limited information. Another problem is when there are very few examples, such as in the case of fighting. Even though fighting looks different to other activities there are too few examples to make accurate classifications.

5 Conclusion and Future Work

The method presented here is shown to work for real interactions as captured on video cameras. The approach of interpreting a trajectory in feature space as a complete shape rather than a collection of points leads to an improvement in classification. Such an approach exploits the temporal dependencies and shape of the longer term temporal dependencies in feature space in an efficient way. By using parametric models of the actual data we can somewhat avoid the problem of having to generate apriori knowledge (in the form of scripts or rules) about what interactions look like. This enables new interaction classes to be incorporated within the same framework with relative ease.

For problems where it is hard to generate good parametric models to represent trajectories with a collection of cluster centres (such as a Gaussian Mixture model) matching the data shape in feature space proves a simple and effective alternative. This problem is compounded when a mixture of Gaussians is used as an observation model for something like a hidden Markov model, as much of the novel inputs generate very low input probabilities even if modelled well.

Here we do not assume that such a larger corpus of data is available to enable learning of complex parametric models such as those used by [5, 9]. Neither do we assume that it it necessary to pre-define the actions which are of interest by using templates as in [3]. Template approaches may indeed be useful in real surveillance applications where explanations may be required by system operators. Future work will compare our method with theirs.

This simplicity of modelling is also an advantage when there is limited data as approaches such as hidden Markov models do not learn well when given few examples in a high dimensional space. However some of the benefits of using a probabilistic model are lost. At present there is no way to tell how certain the model is about the prediction it is giving. There is also the question of how long one should observe something before feeling confident about making a prediction which is currently left un-addressed. Both of these problems will be addressed in future work as will the identification of larger group activity.

Acknowledgments This work was supported by the BEHAVE project funded by EPSRC, project GR/S98146.

References

1. J. W. Davis and A. F. Bobick. The representation and recognition of action using temporal templates. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 23, pages 257–267. IEEE Computer Society, 2001.
2. Y. Du, F. Chen, W. Xu, and Y. Li. Recognizing interaction activities using dynamic bayesian network. In *International Conference on Pattern Recognition (ICPR)*, page 1, Aug 2006.
3. S. S. Intille and A. F. Bobick. A framework for recognizing multi-agent action from visual evidence. MIT Media Laboratory, 1999.
4. D. Makris and T. J. Ellis. Spatial and probabilistic modelling of pedestrian behaviour. In *British Machine Conference*, volume 1, pages 557–566, September 2002.
5. N. M. Oliver, B. Rosario, and A. P. Pentland. A bayesian computer vision system for modelling human interactions. *IEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8), August 2000.
6. S. Park and J. K. Aggarwal. A hierarchical bayesian network for event recognition of human actions and interactions. *Association For Computing Machinery Multimedia Systems Journal*, 2004.
7. EC Funded CAVIAR project/IST 2001 37540. found at url: http://homepages.inf.ed.ac.uk/rbf/caviar/, 2004.
8. K. Sato and J. K. Aggarwal. Recognizing two-person interactions in outdoor image sequences. In *2001 IEEE Workshop on Multi-Object Tracking*. IEEE, 2001.
9. T. Xiang and S. Gong. Recognition of group activities using a dynamic probabilistic network. In *IEEE International Conference on Computer Vision*, pages 742–749, October 2003.