A Probabilistic Framework for Recognizing Similar Actions using Spatio-Temporal Features

Alonso Patron-Perez, Ian Reid
Department of Engineering Science, University of Oxford
OX1 3PJ, Oxford, UK
{alonso,ian}@robots.ox.ac.uk

Abstract

One of the challenges found in recent methods for action recognition has been to classify ambiguous actions successfully . In the case of methods that use spatio-temporal features this phenomenon is observed when two actions generate similar feature types. Ideally, a probabilistic classification method would be based on a model of the *full joint* distribution of features, but this is computationally intractable. In this paper we propose using an approximation of the full joint via first order dependencies between feature types using so-called Chow-Liu trees. We obtain promising results and achieve an improvement in the classification accuracy over naive Bayes and other simple classifiers. Our implementation of the method makes use of a binary descriptor for a video analogous to one previously used in location recognition for mobile robots. Because of the simplicity of the algorithm, once the offline learning phase is over, real-time action recognition is possible and we present an adaptation of this method that works in real-time.

1 Introduction

The interpretation of video sequences is a topic that has been growing in importance in recent years. It involves not only computer vision techniques for obtaining information from images but also machine learning algorithms because of the ultimate need for high level interpretation. Many things are involved in the analysis of a video sequence such as object recognition, target localization, motion detection, physical contraints and human behaviour. In some applications, like visual surveillance, the role played by humans is the main priority, so several methods for modeling human activity have been proposed. However the majority of these approches fall short of modelling "behaviour", which, we posit, refers to a more complex combination of factors like intentionality and therefore belongs to a higher level of understanding.

A first step to achieving behaviour undestanding is almost certainly the recognition of actions. In Efros *et al.* [8] a target motion descriptor was developed for classifying frames using a *k*-nearest neighbour classifier. Robertson [18] used Efros' descriptor within a probabilistic framework, fusing the low-level motion descriptors with location and direction cues via a Bayes' net. That work further made an attempt at behaviour recognition via

hidden Markov representations of action sequences which they proposed could be interpreted as behaviours.

Another kind of descriptor for action classification was employed by Davis [6], obtained as histograms of the motion orientation information contained in what he called "Motion History Images" (MHI). Carlsson and Sullivan [3] utilized keyframes to represent actions: given a new frame a voting matrix is created to compare keyframes with the new frame, with the matrix updated if the topology between corresponding points of a keyframe and the new frame is the same. An interesting approach was followed by Boiman and Irani [2], although their main objective was to detect irregularities in videos and not to classify actions. They define irregularities as fragments of videos that can't be composed using previous observed data. The method apparently works well but is computationally expensive.

Other methods developed lately are based directly on image features. Image features are a common tool in computer vision methods, very popular in areas such as pattern and object recognition. In recent years a new kind of feature, found not only in space but also in time, have been used for action recognition. Laptev and Lindeberg [12] presented an approach that is an extension in time of the well know Harris-Stephens [9] corner detector, achieving good results in general although for subtle movements (like a wheel spinning) the method doesn't give a good response and the number of features obtained is very small. Oikonomopoulos *et al.* [17] also extended another kind of spatial features to the temporal case, in this case the saliency approach of Kadir and Brady [10]. The saliency approach is based on localizing points of local maximum entropy. A method that is not an extension of a previously know feature detector was presented by Dollár *et al.* [7], based primarily on a convolution with a Gabor filter in time. The features obtained in this way have also been used for action recognition by Niebles *et al.* [16] in a very interesting application of probabilistic Latent Semantic Analysis.

In this paper we adopt the spatio-temporal features proposed by Dollár *et al.*. In constrast to their work, however, our classification method is solidly grounded in a probabilistic framework. More precisely we are interested in evaluating $p(z_1, \ldots, z_n|A)$, the joint distribution over feature observations z_i given a particular action A. This joint distribution, comprising n dimensions (where n is typically very large) is intractable to compute. Our main contribution in this paper is to show how this joint distribtion can be effectively approximated using only first-order dependencies [4]. This idea was recently employed in mobile robotics for location recognition [5] and here we apply it to action recognition for the first time.

The remainder of the paper is divided as follows: in section 2 we review the theory behind spatio-temporal features and explain why similar actions are more dificult to identify following this approach. In section 3 our probabilistic method is explained in detail, and in section 4 we describe a real-time implementation of this method. Section 5 shows the results obtained and the last section deals with the conclusions and future directions of research.

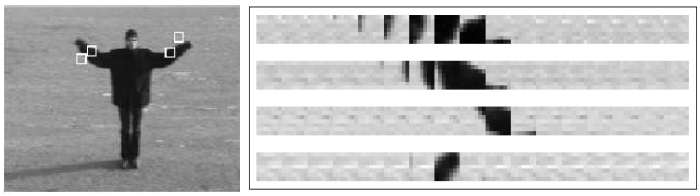
2 Spatio-temporal features

We will use the spatio-temporal features as described in [7], because they have shown good results even using the most simple methods of classification. These features are

obtained by convolving a set of video frames with a Gaussian $g(x,y;\sigma)$ in the spatial dimension and with 1D Gabor filters in the temporal dimension. The 1D Gabor filters are defined as $h_{ev}(t;\tau,\omega) = -\cos(2\pi t\omega)\exp^{-t^2/\tau^2}$ and $h_{od}(t;\tau,\omega) = -\sin(2\pi t\omega)\exp^{-t^2/\tau^2}$. In this quadrature pair each filter is a harmonic function with a Gaussian envelope. The parameter τ in this specific case controls the frequency of the harmonic function and the variance of the Gaussian. The whole response function is then expressed as:

$$R = (I * g * h_{ev})^2 + (I * g * h_{od})^2 \tag{1}$$

Using a threshold and non-maximal suppression, the features with highest response are selected. Usually these high response locations correspond to local regions containing periodic motions but regions with spatio-temporal corners [12] are also detected. Once the spatial and temporal location of a feature is obtained, the set of pixels which surround it in a given size of spatio-temporal window is extracted. [7] call this spatio-temporal window a *cuboid*. The side length of this temporal window is set to six times the scale at which the feature was detected. An example of the features detected by this method and their corresponding cuboids is shown in Figure 1.

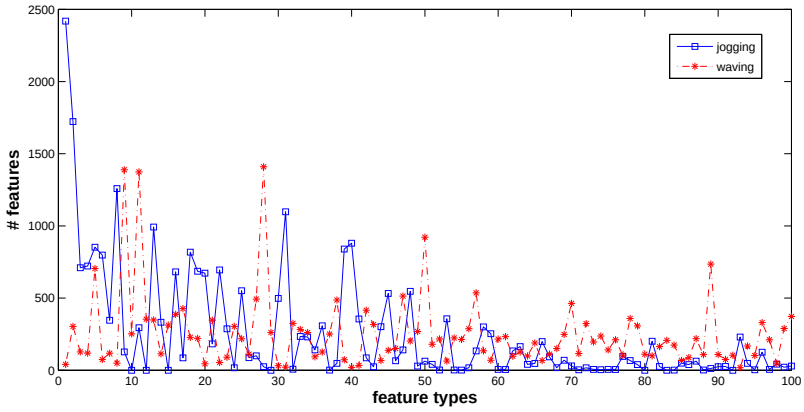


(a) (b)

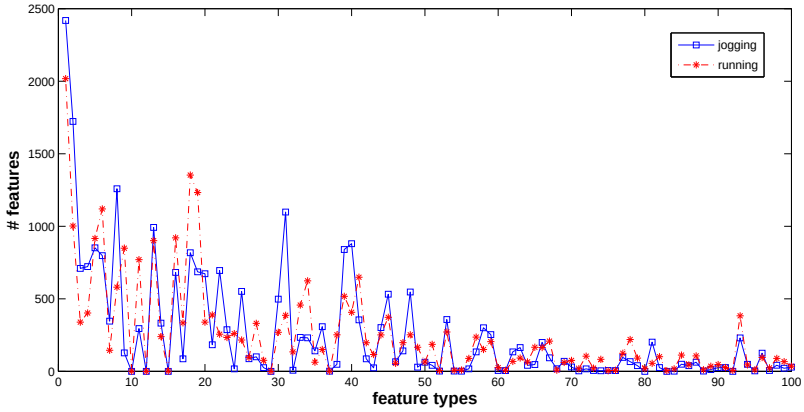
Figure 1: (a) Detected features. (b) Cuboids (flattened representation, time runs from left to right).

As shown in [14] the selection of a good descriptor for spatial features has proven to play an important role in improving the results of any algorithm involving image features. Regarding spatio-temporal features, [7] experiments with some of the classic descriptors including SIFT [13], and show that apparently a simple descriptor such as the concatenated brightness values of the cuboids gives results as good as or better than other descriptors. We have not conducted our own evaluation and like [7] we use this simple descriptor in our work. The features are clustered using *k*-means with Euclidean distance and a fixed number of cluster centres. The clusters centres obtained in this process define the set of feature types. After the clustering process each feature is tagged as belonging to one of these clusters. The overall process involves the selection of four parameters: the scales in space and time (σ,τ), the feature threshold, and the number of cluster centres. The latter determines the complexity of our model.

Using feature types for classifying similar actions has its problems because similar actions generate similar features as can be observed in Figure 2. This is what motivates the development of our method that can deal to some extent with this problem.



(a)



(b)

Figure 2: Graphics of feature ocurrence for (a) Jogging and Waving. (b) Jogging and Running. As can be seen similar actions, like those in (b), produced similar kinds of features.

3 Probabilistic formulation

Different methods have been used for the classification of actions. In [7] histograms of feature types using the Euclidean and χ^2 distances are compared. Laptev and Lindeberg [12] define models for each action and select the model that best fits the features. Probabilistic methods are used in [17] and [16]. The former uses Relevance Vector Machines for obtaining posterior probabilities of actions, and in the latter a probabilistic Latent Semantic Analysis is implemented.

Our implementation follows the spirit of these latter two approaches, recognizing that probabilistic approaches are more robust and give more relevant information that can be

used in higher levels of understanding. The approach described here is based on the method proposed by Cummins and Newman [5]. In their work the aim was to recognise a previously visited location for the specific case of loop closing in robot navigation. It is a relatively simple probalistic approach, but as we will show gives remarkably good results.

Our aim is to classify a *video* as one of the set of previously known actions. Here a video can refer to any sequence of frames, but in the case of our training data (the KTH dataset [11]) each video is in fact a single file, and we have one action per file. In our real-time implementation in section 4 we employ a sliding temporal window with the "video" being the sequence of frames within the window. We begin defining an observation vector as $Z = \{z_1, z_2, ..., z_n\}$ where each z_i is a binary variable that is set to 1 if a feature type is observed in a video and zero otherwise. Defining a set $A = \{A_1, ..., A_k\}$ of actions we want to obtain the conditional probability $p(A_i|Z)$ for each action. Using Bayes rule this is:

$$p(A_i|Z) = \frac{p(Z|A_i)p(A_i)}{p(Z)} \tag{2}$$

where $p(Z)$ is just a normalizing factor. Given the prior that each action is equally likely to occur, the important term in (2) becomes the likelihood: $p(Z|A_i)$. This conditional distribution contains the information on the relationships of the feature types and is therefore the one that we want to learn. Because the complexity of the calculation of the joint distribution increases exponentialy with the number of variables (feature types), computing of the full distribution quickly becomes untractable. We believe that capturing at some extent these relationships between feature types is a key factor for differentiating similar actions, therefore a way of approximating this distribution is of paramount importance. In the next sections we describe an approximation given by Chow and Liu [4] and also we briefly describe the naive Bayes approach as a standard method for comparison.

3.1 Naïve Bayes

One of the simplest ways of approximating a joint distribution is to assume independence between all the variables; this is called naïve Bayes and is expressed mathematically as $p(x_1, x_2, ..., x_n) = \prod_i^n p(x_i)$, in our case this will be:

$$p(Z|A_j) = p(z_1, z_2, ..., z_n|A_j) = \prod_i^n p(z_i|A_j) \tag{3}$$

This is a very simple way of approximating the joint distribution and very easy to implement. We simply have to learn the marginal probability of each feature type, and this can be done as explained in [15] by setting $p(z_i|Aj) = n(z_i)/N_j$, where $n(z_i)$ is the number of times that feature type z_i was observed and N_j is the total number of videos corresponding to action j. This kind of estimation has one drawback: if a feature type is not observed during training then its marginal probability becomes zero and therefore the whole joint becomes zero. To avoid this extreme case we apply a simple way of smoothing namely $p(z_i|Aj) = (n(z_i) + s)/(N + s * v)$ where s is the smoothing factor and v is the number of values that the random variable z can take. If $s = 1$ this is known as Laplacian smoothing.

3.2 Chow - Liu approximation

Chow and Liu [4] presented a first order approximation to the full joint distribution (i.e. the joint is represented as the product of terms $p(z_i|z_j,A)$, so that each observation variable z_i is conditioned on at most one other z_j). The distribution obtained in this way is proved to be the optimal approximation of the full distribution in the sense of minimizing the Kullback-Leibler divergence.

In order to determine the approximation the mutual information I between pairs of variables is calculated

$$I(z_i,z_j) = \sum_{z_i,z_j} p(z_i,z_j) \log \frac{p(z_i,z_j)}{p(z_i)p(z_j)} \tag{4}$$

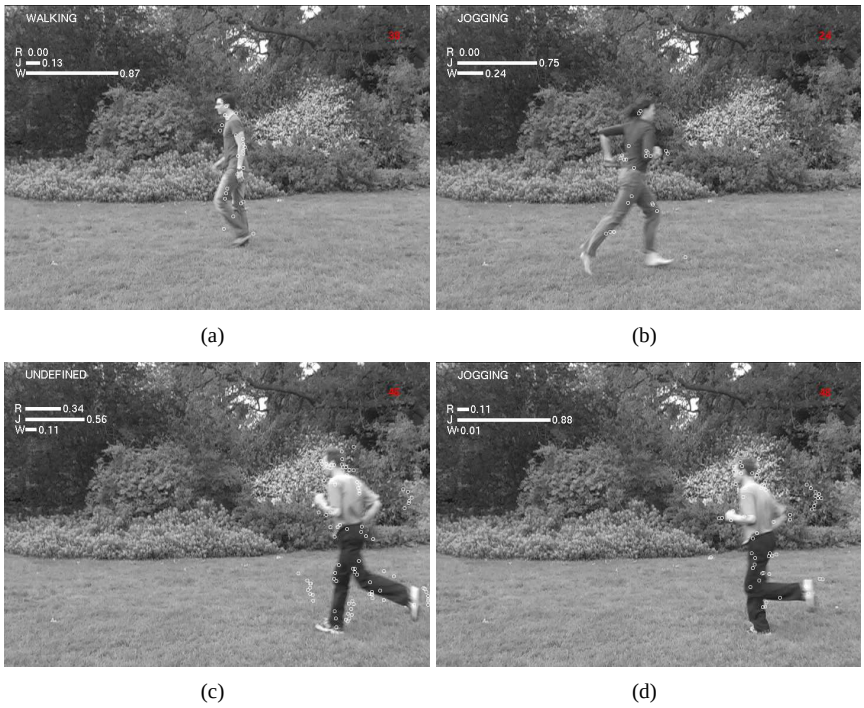
where the sum is made for all combination of values that the variables z_i and z_j can take. The joint distribution $p(z_i,z_j)$ and the marginals $p(z_i)$, $p(z_j)$ can be learnt from data. The next step is to construct a weighted undirected graph whose nodes are the random varibles z_i and the weight of each edge is given by the mutual information between its two nodes. The final step involves finding the maximum spanning tree in this graph, as task common in graph theory. The maximal spaning tree provides the structure of the dependencies between the variables: each variable is dependent on only one other variable (except for the root of the tree). The approximation of the full distribution is then

$$p(Z|A_j) = p(z_r|A_j) \prod_{q\in\Omega} p(z_q|z_{p_q},A_j) \tag{5}$$

where z_r is the root node, Ω is the set of all nodes excluding the root node and z_{p_q} is the parent of node z_q. As in the case of naïve Bayes we have to apply some kind of smoothing to avoid probabilities going to zero.

4 Real-Time implementation

Having obtained action descriptors in an offline learning phase, the recognition task is extremely fast. For our real-time implementation we maintain a FIFO buffer of video frames in memory. The size of this buffer of course depends on the temporal scale chosen and this induces a fixed latency of half the buffer length on the results. Each image received is first smoothed with a Gaussian, and then inserted into the buffer. Once the buffer is full we need only perform the convolution in time, extract the features of the central frame in the buffer, and compute a binary frame descriptor for that frame in an analogous fashion to the complete video in section 3. We then estimate the posterior probability for each one of our actions given this new data. If the estimated probability for the maximum likelihood action is above a threshold then we identify the current action, and if not we read another frame and add its new frame descriptor to our previous one and compute the probabilities again, this continues until the probability of one of the actions reaches the threshold. Frames older than a fixed lag are removed from consideration. Images of the real-time system working are shown in Figure 3.



(a) (b)

(c) (d)

Figure 3: Snap-shots of the real-time implementation. (a)-(b) Correct classification of the action. (c) In this frame there is confusion between jogging and running so the action remains undefined until more information is acquired. Two frames later, in image (d), the action is correctly classified.

5 Experiments

For our experimental evaluation we have used the action database made by Laptev and that is available online at KTH website [11]. This database contains videos of six different actions namely: hand clapping, hand waving, walking, jogging, running and boxing, performed by 25 different persons and has been used in various papers as their training and test data. The total database contains 599 videos. Because our main objective is to compare the results obtained by the Chow-Liu approximation when trying to classify similar actions against the simple naive Bayes approach we first tested with three actions: walking, jogging and running. A set of images extracted from the videos corresponding to these three actions can be seen in Figure 4.

The methodology chosen for the experiments is the one used in [16]. The database was divided in 25 sets, one for each person, we used the features extracted from the videos of 3 randomly selected persons for clustering (and obtain our feature types). With the remaining 22 persons we performed a leave-one-out cross validation test. Because the clustering phase introduces a random component, the experiments were repeated 25 times for each number of cluster centres and the results presented here are an average of these runs. The parameters used for the feature detection were $\sigma = 1$ (spatial scale), $\tau = 2.5$

Figure 4: Some images taken from the videos of the KTH database.

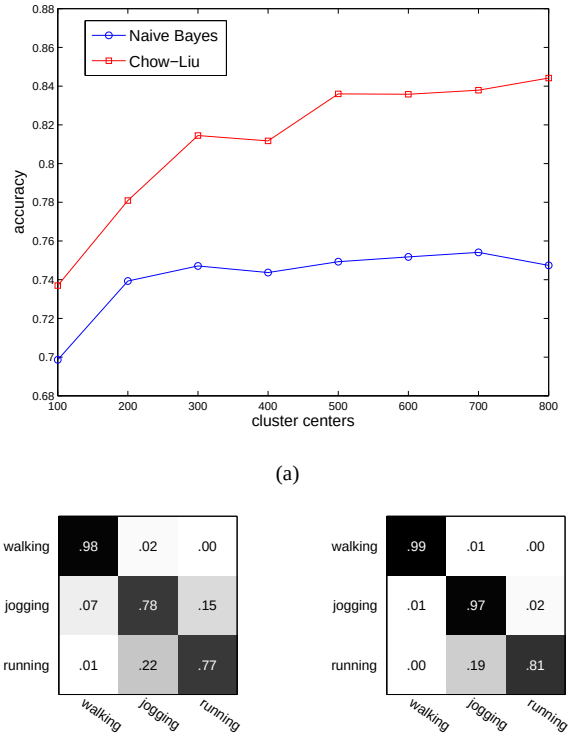
and a threshold of 0.01. The scales were fixed during the experiments and we only used the first 200 frames of each video for extracting features. The results of both the naïve Bayes approach and the Chow-Liu approximation for different cluster centres, can be seen in Figure 5a. Figure 5b shows the confusion matrix obtained with the Chow-Liu algorithm and 800 cluster centres. It should be noted that in measuring the performance of discrimination between actions (especially jogging and running) we take the labels in the KTH dataset as ground truth, but these labels are very much open to individual interpretation. In order to asses this ambiguity in the "ground-truth" we asked several people to classify 300 videos of walking, jogging and running. The confusion matrix of the average results obtained in this test is shown in Figure 5c.

In order to compare our classification scheme with the results presented in other papers we also made experiments using the whole database. Figure 6 show the results of both methods when training and testing with the six actions using 500 cluster centres.

6 Conclusions

We have presented an action classification method that exploits the relationships between different spatio-temporal feature types and we have shown it to improve the classification of similar actions. The results are promising despite some ambiguity of the data set ground truth. We also showed how this method can be implemented in a real-time application. It is also worth remarking that even the simple naïve Bayes approach gives very good results when classifying disimilar actions, as can be seen in the results shown in Figure 6, indeed the accuracy for these actions is sometimes better that the one obtained using more complex models.

Although in this paper we have used a supervised learning approach, this can be easily extended to an usupervised form in the case of naïve Bayes by means of Mixtures of Bernoulli distributions and EM as described in [1]. The case of an unsupervised Chow-Liu algorithm is more difficult and the subject of our ongoing research. In common with the related work of [7], currently our overall action descriptor – the binary occurrence vector – takes no account spatial or temporal ordering, factors which we believe may be



(a)

(b)

(c)

Figure 5: Results obtained. (a) Accuracy of both methods using different cluster centres. (b) Confusion matrix obtained with the Chow-Liu algorithm and 800 cluster centres. (c) Confusion matrix as a result of human classification.

important to incorporate into our descriptor.

7 Acknowledgements

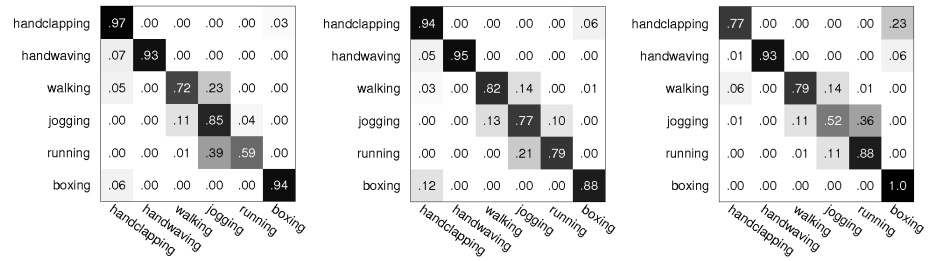
We would like to thank Piotr Dollár for allowing us the use of his code. This work was supported by CONACYT and the EU FP6 project *HERMES*.

References

[1] C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[2] O. Boiman and M. Irani. Detecting irregularities in images and in video. In *International Conference on Computer Vision*, volume 1, pages 462–469, 2005.

[3] S. Carlsson and J. Sullivan. Action Recognition by Shape Matching to Key Frames. In *Workshop on Models versus Exemplars in Computer Vision*, 2001.



(a) (b) (c)

Figure 6: Confusion matrices for the six actions obtained using 500 cluster centres: (a) naïve Bayes. (b) Chow-Liu. (c) Results obtained by Niebles et al. [16] (unsupervised learning).

[4] C.K. Chow and C.N. Liu. Approximating Discrete Probability Distributions with Dependence Trees. *IEEE Transactions on Information Theory*, 14(3):462–467, 1968.

[5] M. Cummins and P. Newman. Probabilistic Appearance Based Navigation and Loop Closing. In *International Conference on Robotics and Automation*, 2007.

[6] J. Davis. Recognizing Movement using Motion Histograms. Technical Report 487, MIT Media Lab Perceptual Computing Group, MIT, 1999.

[7] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *VS-PETS*, pages 65–72, 2005.

[8] A. Efros, A. Berg, G. Mori, and J. Malik. Recognizing Action at a Distance. In *International Conference on Computer Vision*, Nice, France, 2003.

[9] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, pages 147–151, 1998.

[10] T. Kadir and M. Brady. Scale Saliency: A Novel Approach to Salient Feature and Scale Selection. In *International Conference on Visual Information Engineering*, pages 25–28, 2003.

[11] I. Laptev and B. Caputo. Recognition of human actions. Action Database. http://www.nada.kth.se/cvap/actions/, 2007.

[12] I. Laptev and T. Lindeberg. Space-time Interest Points. In *International Conference on Computer Vision*, 2003.

[13] D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 2004.

[14] K. Mikolajczyk and C. Schimd. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[15] T. Mitchell. *Machine Learning*. McGraw Hill, 1997.

[16] J. Niebles, H. Wang, and L. Fei-Fei. Unsupervised Learning of Human Action Categories Using Spatial-Temporal Words. In *British Machine Vision Conference*, 2006.

[17] A. Oikonomopoulos, P. Ioannis, and M. Pantic. Spatio-Temporal Salient Points for Visual Recognition of Human Actions. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, 36(3):710–719, 2006.

[18] N. Robertson and I. Reid. Behaviour understanding in video: a combined method. In *International Conference on Computer Vision*, 2005.