High Five: Recognising human interactions in TV shows

Alonso Patron-Perez
alonso@robots.ox.ac.uk
Marcin Marszalek
marcin@robots.ox.ac.uk
Andrew Zisserman
az@robots.ox.ac.uk
Ian Reid
ian@robots.ox.ac.uk

Department of Engineering Science
University of Oxford
Oxford, UK

Abstract

In this paper we address the problem of recognising interactions between two people in realistic scenarios for video retrieval purposes. We develop a per-person descriptor that uses attention (head orientation) and the local spatial and temporal context in a neighbourhood of each detected person. Using head orientation mitigates camera view ambiguities, while the local context, comprised of histograms of gradients and motion, aims to capture cues such as hand and arm movement. We also employ structured learning to capture spatial relationships between interacting individuals.

We train an initial set of one-vs-the-rest linear SVM classifiers, one for each interaction, using this descriptor. Noting that people generally face each other while interacting, we learn a structured SVM that combines head orientation and the relative location of people in a frame to improve upon the initial classification obtained with our descriptor. To test the efficacy of our method, we have created a new dataset of realistic human interactions comprised of clips extracted from TV shows, which represents a very difficult challenge. Our experiments show that using structured learning improves the retrieval results compared to using the interaction classifiers independently.

1 Introduction

The aim of this paper is the recognition of *interactions* between two people in videos in the context of video retrieval. In particular we focus on four interactions: hand shakes, high fives, hugs and kisses. Recognising human interactions can be considered an extension of single-person action recognition and can provide a different criteria for content-based video retrieval. Two-person interactions can also be used directly or as a building block to create complex systems in applications like surveillance, video games and human-computer interaction.

Previous work in two-person interaction recognition is scarce compared to closely related areas such as single-person action recognition [7, 10, 12, 22], group action recognition [14, 24] and human-object interaction recognition [16, 23]. Closer to our work are [4,

© 2010. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.



Figure 1: Dataset snapshots. Note the variation in the actors, scale and camera views.

[17, 19], where interactions are generally recognised in a hierarchical manner putting special attention on higher level descriptions and using very constrained data. These approaches rely heavily upon many low level image pre-processing steps like background subtraction and segmentation of body parts which are, by themselves, very difficult problems to solve when working with more complex scenarios. In contrast, recent publications on single-action recognition have shown a natural move from simplified and constrained datasets to more realistic ones [11, 12, 13, 21, 22]. One of the contributions of this paper is the compilation of a realistic human interaction dataset extracted from a collection of TV shows (Section 2). Working with realistic datasets introduces a new set of challenges that have to be addressed in order to achieve successful recognition: background clutter, a varying number of people in the scene, camera motion and changes of camera viewpoints, to name a few.

Our approach is to introduce a person-centred descriptor that uses a combination of simple features to deal in a systematic way with these challenges. An upper body detector [6] is first used to find people in every frame of the video (Section 3). The detections are then clustered to form tracks. A *track* is defined as a set of upper body bounding boxes, in consecutive frames, corresponding to the same person. The aim of this first step is to reduce the search space for interactions to a linear search along each track in an analogous way as [9]. We then calculate descriptors along these tracks and use them to learn a Support Vector Machine (SVM) classifier for each interaction. Then interaction scores are computed for each bounding box of each track. We also use the head orientation of people detected in two novel ways: first to achieve a weak view invariance in the descriptor (see Section 3), and second to learn interaction-based spatial relations between people (Section 4). The latter is based on our assumption that people generally face each other while interacting. This assumption is used to learn a structured SVM [20] that is trained to obtain the best joint classification of a group of people in a frame. We show that using structured learning (SL) can improve the retrieval results obtained by independently classifying each track. An additional characteristic of our structured formulation is that it provides information about which people are interacting. In Section 4.2 we show the retrieval results obtained by the individual and structured track classification. Section 5 presents our conclusions and future work.

2 Dataset

We have compiled a dataset of 300 video clips extracted from 23 different TV shows[1]. Each of the clips contains one of four interactions: hand shake, high five, hug and kiss (each

[1] http://www.robots.ox.ac.uk/∼vgg/data/tv_human_interactions

appearing in 50 videos). Negative examples (clips that don't contain any of the interactions) make up the remaining 100 videos. The length of the video clips ranges from 30 to 600 frames. The interactions are not temporally aligned (i.e. a clip containing a hand shake might start with people walking towards each other or directly at the moment of the hand shake). There is a great degree of variation between different clips and also in several cases within the same clip (Figure 1). Such variation includes the number of actors in each scene, their scales and the camera angle, including abrupt viewpoint changes (shot boundaries).

To have a ground truth for the evaluation of the methods developed in this paper, we have annotated every frame of each video with the following: the upper body, discrete head orientation and interaction label of all persons present whose upper body size is within a certain range. This range goes from far shots that show the whole body to medium shots where only the upper body is visible and is equivalent to 50-350 pixels in our videos. We have also annotated which persons are interacting, if any, in each frame. For the purposes of training and testing, the dataset has been split into two groups, each containing videos of mutually exclusive TV shows. The experiments shown in the following sections were performed using one set for training, the other for testing and vice versa.

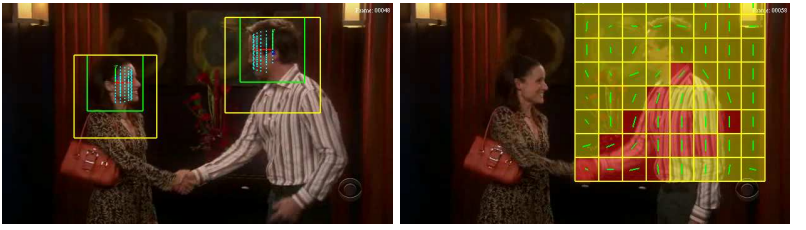
3 Modeling human activity

Because of the complexity and variability of the videos in our dataset, finding relevant and distinctive features becomes increasingly difficult. The descriptor has to be simultaneously (i) relatively coarse to deal with variation, and (ii) to some extent focused to avoid learning background noise when codifying the interaction. We address these points by making our descriptor person-centred, and by further organising the data based on head orientation.

The person-centred descriptor focuses on the area around the upper body of a single person, enabling us to localise regions of potential interest and to learn relevant information inside them. Our descriptor does this by coarsely quantifying appearance and motion inside this region. This is in contrast to other approaches in single-action recognition [7, 11, 12, 15, 22], where features are estimated in the whole frame or video and then clustered to localise where the action is happening. Another advantage for implementing a person-centred descriptor is that, depending on the camera angle, both persons are not always visible in a given frame, and we would like to be able to provide a classification in these instances. For the moment, we assume that we know the location and scale of people in each frame and leave the detection method for section 3.2.

3.1 Person-centred descriptor

The following describes the process for obtaining a descriptor given an upper body location, which is repeated for each person detected in a frame. Our descriptor superimposes an 8×8 grid around an upper body detection. The size of the grid, being dependent on the detection size, deals with changes of scale. We then calculate histograms of gradients and optical flow in each of its cells. An example of this can be seen in Figure 2b. This technique of using histograms of gradients and flow is a coarse analog to the descriptor used in [3, 11]. Gradients are discretised into five bins: horizontal, vertical, two diagonal orientations and a no-gradient bin. Optical flow is also discretised into five bins: no-motion, left, right, up and down. The histograms are independently normalised and concatenated to create an initial grid descriptor $\mathbf{g}$ (Note on notation: whenever a vector is used in this paper is considered to be in row format by default). We also experimented with several variants of the grid



(a) (b)

Figure 2: (a) Upper body detections and estimated discrete head orientation. (b) Grid showing dominant cell gradient and significant motion (red cells) for a hand shake.

descriptor: using only motion, only gradients, only information of the cells outside the upper body detection as well as different normalisations. The experiments described in Section 3.3 show the results obtained by selecting different parameters.

To obtain the final descriptor $\mathbf{d}$, we take into account the head orientation, discretised into one of five orientations: profile-left, front-left, front-right, profile-right and backwards. Perfect frontal views are very rare and they are included in either of the two frontal categories. Effectively, we want to create a compact and automatic representation from which we can learn a different classifier for each discrete head orientation. To do this, the discrete head orientation, θ, is used to perform the following operation:

$$\mathbf{g}^{+} = \mathbf{g} \otimes \boldsymbol{\delta}_{\theta}, \quad \mathbf{d} = [\mathbf{g}^{+} \quad \mathbf{g}] \tag{1}$$

where $\otimes$ is the Kronecker product, $\boldsymbol{\delta}_{\theta}$ is an indicator vector with five elements (corresponding to the discrete head orientations) having a one at position θ and zero everywhere else. By using the head orientation, we are aiming to capture information correlated with it. Assuming that an interaction occurs in the direction a person is facing (Figure2a) this can provide us with a weak kind of view invariance. We add an extra copy of $\mathbf{g}$ at the end of the descriptor $\mathbf{d}$ to account for any information that is independent of the head orientation and to help in cases where the automatic estimation of the head orientation is wrong. We can duplicate the amount of examples used for training by horizontally flipping the video frames resulting in oposite head orientations (i.e. profile-left becomes profile-right).

The descriptor $\mathbf{d}$ is used as a data vector for training a linear SVM classifier. An illustrative example of the results that we obtain, Figure 3, shows the motion regions (outside the upper body detection) learnt by a linear SVM classifier trained to discriminate between hand shakes and high fives. As expected, important motion regions are correlated with the head orientation and occur in lower locations for hand shakes and higher ones for high fives.

3.2 Localising humans and estimating head orientation

To be able to use the descriptor proposed above, we need to pre-process our video clips. The pre-processing follows the same steps as in [8], and we briefly explain them here for completeness. First we run an upper body detector in each frame. This detector is trained using a standard Histogram of Oriented Gradients (HOG) descriptor [7] and a simple linear SVM classifier. We train two such detectors at a different initial scale (to improve the detection rate). Next, we cluster these detections using clique partitioning to form tracks. Very short tracks and tracks with low average SVM scores are eliminated, and those that remain are used in the experiments. As in [1, 13] we learn a classifier for discrete head orientations,

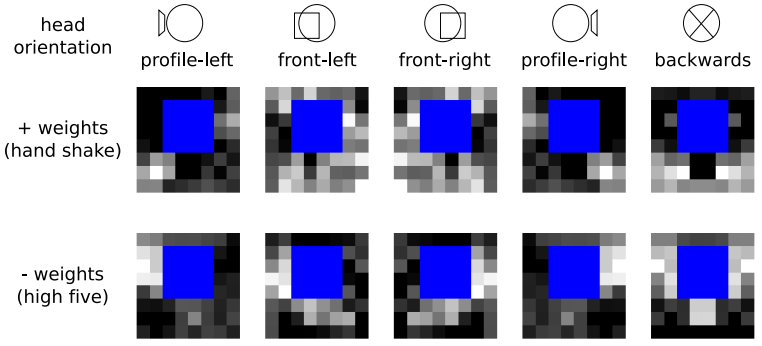


Figure 3: Motion weights outside upper body detection (blue square) learnt by a linear SVM classifier trained to discriminate between hand shakes and high fives. Higher weights are indicated by lighter areas. As expected, the more important motion regions are in lower locations for hand shakes and in higher ones for high fives. These also follow the direction of the face.

however we simply train a one-vs-the-rest linear SVM using HOG descriptors. Once the classifier is learnt, we estimate the head location in each bounding box of each track and obtain a discrete head orientation classification.

3.3 Experiments

Given that people's tracks have been calculated in every video as previously described, we want to evaluate the accuracy of our descriptor when classifying interactions. We have designed a set of experiments to show the effect of: *(i)* not using head orientation information *vs* adding it either by manual annotation or by automatic classification; *(ii)* changing descriptor information: using only motion, only gradients or both; *(iii)* adding weak temporal information by concatenating descriptors of consecutive frames to form a single descriptor. The term *n-frame* descriptor refers to a concatenation of *n* descriptors from consecutive frames.

To be able to compare the results obtained, all of the experiments follow the next steps. We manually select from each clip five consecutive frames that are inside the temporal region where the interaction is happening. From these frames we extract descriptors from a track of one of the people performing the interaction (again we manually select the track). The same process is applied to the negative videos. As described in Section 2, the dataset is divided into two sets for training and testing. We use in turn the descriptors of each set to train a one-vs-the-rest linear SVM classifier for each interaction in a supervised way. The classification of a clip is done by adding the SVM classification scores of each one of the descriptors extracted from its five selected frames.

Figure 4 provides a visual representation of the results. Column-wise we observe accuracy results obtained using different *n-frame* descriptors. Row-wise represents the average accuracy when choosing different information to include in the descriptor: only motion, only gradients and both. Each row is an average over tests using full or external cells and different normalisations (L1, L2 or no-norm). The table itself is an average of the results obtained when testing on both sets.

Several things can be concluded from this representation. First, we can readily observe that the use of head orientation improves the classification accuracy when correctly esti-

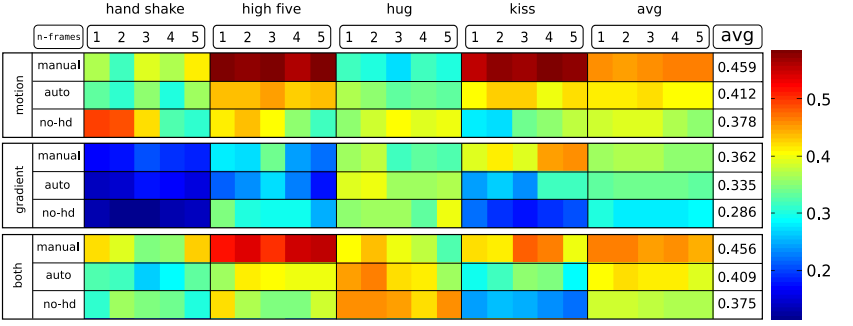


Figure 4: Average classification accuracy results with different parameter combinations. No consistent improvement is noticed by using higher *n-frame* descriptors. Motion information is a more discriminative feature than gradients in three of the four interactions. On average using head information improves the accuracy. (Best viewed in color).

mated, but errors when automatically classifying the head orientation reduce it. Taking the best combination of parameters for each interaction (using *1-frame* descriptors), the average accuracy when using manually annotated head orientation is 59.4%, for automatic head orientation 52.2% and for no head orientation 48.8%. We noted that the concatenation of descriptors did not consistently improve the classification results.

Another easily distinguishable characteristic is that the use of motion features alone has better performance when classifying high fives and kisses, while a combination of both works better for hugs. This is very intuitive because hugs contain minimal motion in contrast to the other actions. The bad performance of using only gradients could be explained by the coarseness of our descriptor, which results in learning gradients that are too general to be distinctive. We tried to improve these results by increasing the number of cells. The resulting increased size of the descriptor combined with a reduced number of training examples led to worse classification results.

4 Learning human interactions

As mentioned before, sometimes only one of the two people performing an interaction appears in the video clip. However, when the location of two or more people is available in a specific frame, we should use this to improve our classification. The assumption we make is that people face each other while interacting. Thus we want to learn relative locations of people given both their head orientation and an interaction label. We propose to do this by using a structured learning (SL) framework similar to the one described in [5]. The goal is to simultaneously estimate the best joint classification for a set of detections in a video frame rather than classifying each detection independently. In contrast to [5], where SL is used to learn spatial relations between object classes, we want to learn spatial relations between people given their interaction class and head orientation.

4.1 Structured learning

We pose the SL problem in the following terms: in each frame we have a set of upper body detections $\mathbf{X} = [\mathbf{x}_1 \dots \mathbf{x}_M]$. Each detection $\mathbf{x}_i = [l_x \;\; l_y \;\; s \;\; \theta \;\; \mathbf{v}]$, has information about its upper left corner location (l_x, l_y), scale (s), discrete head orientation (θ), and SVM classification

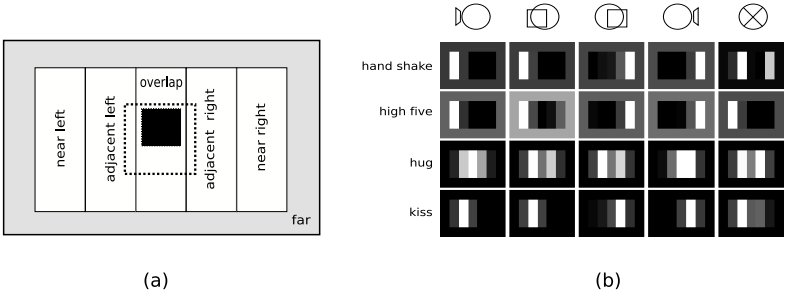


Figure 5: (a) Spatial relations (δ_{ij}) used in our structured learning method. The black square at the centre represents the head location inside an upper body detection. (b) Weights ($\boldsymbol{\beta}$) learnt for each interaction class and head orientation combination. Lighter intensity indicates a higher weight.

scores ($\mathbf{v}$) obtained by classifying the descriptor associated with this detection using the interaction classifiers previously learnt. Associated with each frame is a label $\mathbf{Y} = [y_1 \dots y_M\, y_c]$. This label is formed by a class label $y_i \in \{0,..,K\}$ for each detection (where K is the number of interaction classes, with 0 representing the no-interaction class) and a configuration label y_c that serves as an index for one of the valid pairings of detections. For example, for three detections there are four valid configurations: {(1,0) , (2,0), (3,0)}, {(1,0) , (2,3)} , {(1,3) , (2,0)} and {(1,2) (3,0)}, where (i, j) indicates that detection i is interacting with detection j and the 0 index means there is no interaction. We measure the match between an input X and a labeling Y by the following cost function:

$$S(\mathbf{X},\mathbf{Y}) = \sum_{i}^{M} \alpha^0_{y_i\theta_i} v_{y_i} + \sum_{i}^{M} \alpha^1_{y_i\theta_i} + \sum_{(i,j)\in P_{y_c}} (\boldsymbol{\delta}_{ij}\boldsymbol{\beta}^T_{y_i\theta_i} + \boldsymbol{\delta}_{ji}\boldsymbol{\beta}^T_{y_j\theta_j}) \tag{2}$$

where v_{y_i} is the SVM classification score for class y_i of detection i, P_{y_c} is the set of valid pairs defined by configuration index y_c , $\boldsymbol{\delta}_{ij}$ and $\boldsymbol{\delta}_{ji}$ are indicator vectors codifying the relative location of detection j with respect to detection i (and vice versa) into one of $R = 6$ spatial relations shown in Figure 5a. $\alpha^0_{y_i\theta_i}$ and $\alpha^1_{y_i\theta_i}$ are scalar weighting and bias parameters that measure the confidence that we have in the SVM score of class y_i when the head discrete orientation is $\theta_i \in \{1,\dots,D\}$. $\boldsymbol{\beta}_{y_i\theta_i}$ is a vector that weights each spatial configuration given a class label and discrete head orientation. Once the weights are learnt, we can find the label that maximises the cost function by exhaustive search, which is possible given the small number of interaction classes and number of people in each frame.

Learning. We use the SVM^{struct} package [8] to learn the weights α and $\boldsymbol{\beta}$ described previously. To do this, we must first re-arrange equation 2 to define a single weight vector and encapsulate the X and Y components into a potential function Ψ (see [20]), and second we need to define a suitable loss function. We start by defining: $\boldsymbol{\delta}^+_{ij} = \boldsymbol{\delta}_{ij} \otimes \boldsymbol{\delta}_{y_i\theta_i}$ and $\boldsymbol{\delta}^+_{ji} = \boldsymbol{\delta}_{ji} \otimes \boldsymbol{\delta}_{y_j\theta_j}$, where $\otimes$ means the Kronecker product and $\boldsymbol{\delta}_{y_i\theta_i}$ is an indicator vector of size KD having a one at position $y_i * K + \theta_i$ and zeros everywhere else. Also, let $\boldsymbol{\alpha}^0_* = [\alpha^0_{01} \dots \alpha^0_{KD}]$, $\boldsymbol{\alpha}^1_* = [\alpha^1_{01} \dots \alpha^1_{KD}]$ and $\boldsymbol{\beta}_* = [\boldsymbol{\beta}_{01} \dots \boldsymbol{\beta}_{KD}]$. By substituting into equation 2 we obtain:

$$S(\mathbf{X},\mathbf{Y}) = \underbrace{[\boldsymbol{\alpha}^0_* \quad \boldsymbol{\alpha}^1_* \quad \boldsymbol{\beta}_*]}_{\mathbf{w}} \underbrace{[\sum_{i}^{M} v_{y_i}\boldsymbol{\delta}_{y_i\theta_i} \quad \sum_{i}^{M} \boldsymbol{\delta}_{y_i\theta_i} \quad \sum_{(i,j)\in P_{y_c}} (\boldsymbol{\delta}^+_{ij} + \boldsymbol{\delta}^+_{ji})\ ]^T}_{\Psi} \tag{3}$$

A key element of a SL framework is to define an adequate loss function for the problem in consideration. Here we would like the loss function not only to penalise wrong assignments of interaction labels but configuration labels as well. We also want additionally to penalise a label mismatch between detections that are labeled as interacting. Taking these elements into consideration, we define our loss function as:

$$\Delta(\mathbf{Y},\hat{\mathbf{Y}}) = \sum_{i}^{M} \Delta_{01}(y_i,\hat{y}_i) + \sum_{(i,j)\in P_{y_c}} \Delta_c(i,j) \tag{4}$$

$$\Delta_c(i,j) = \begin{cases} 1 & \text{if } (i,j) \notin P_{\hat{y}_c} \\ 1 & \text{if } (i,j) \in P_{\hat{y}_c} \text{ and } \hat{y}_i \neq \hat{y}_j \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

were Δ_{01} is the zero-one loss, $\mathbf{Y}$ is the ground truth labeling and $\hat{\mathbf{Y}}$ is a labeling hypothesis. Consider a frame with three people, two of them interacting. A candidate label that assigns an incorrect interaction label to a person that is not interacting will result in a loss of 1 from Δ_{01}. If instead this error occurs in one of the people that are interacting the loss will be 2 (1 for the incorrect label in Δ_{01} plus 1 for assigning different labels to interacting people in Δ_c). Errors in the configuration label (y_c) tend to increase the loss significantly depending on the number of actors present. An example of the spatial weights learned using this method can be seen in Figure 5b.

4.2 Experiments

In this section we compare the retrieval results obtained by individual classification and by SL. As indicated in Section 3.3, the concatenation of descriptors did not consistently improve the classification accuracy. Therefore, we selected a simple *1*-frame descriptor that uses both motion and gradients with L1 normalisation and all cells in the grid. The classifiers were trained to discriminate between five classes: the four interactions and a no-interaction class. For a retrieval task we need to define a scoring function for a video clip. We propose a score based on the classification of each track extracted from the clip. In each frame a detection belonging to a track is classified either independently using the classifiers learned in Section 3.3 or using the SL framework. The score of each interaction in a track is simply the percentage of its detections that were classified as that interaction. The overall interaction scores of a clip are the average of the track scores. The average is calculated over the tracks where at least one frame was classified as an interaction. This is to avoid assigning low

Method	HS	HF	HG	KS	AVG
M + ID	0.5433	0.4300	0.4846	0.5349	0.5032
M + SL	0.5783	0.5108	0.7116	0.7654	**0.6415**
M + ID + N	0.4069	0.3348	0.3952	0.5003	0.4093
M + SL + N	0.4530	0.4507	0.6200	0.7058	**0.5574**
A + ID	0.4765	0.3194	0.4184	0.3153	0.3824
A + SL	0.4423	0.3255	0.4462	0.3592	**0.3933**
A + ID + N	0.3981	0.2745	0.3267	0.2613	0.3151
A + SL + N	0.3517	0.2569	0.3769	0.3250	**0.3276**

Table 1: Average precision results for the video retrieval task, when using manual (M) or automatic (A) annotations, independent (ID) or structured (SL) classification and when including the negative (N) videos as part of the retrieval task. In every case, the use of structured learning improves the average results.



Figure 6: Highest ranked true and false positives for each interaction obtained using the automatic method. The red square indicates negative videos.

interaction scores to videos with many actors (most of whom are not interacting). The score for no-interaction is an average over all tracks. The same process is used for scoring the negative videos and evaluate the effect that including these clips has on the overall ranking.

Average precision (AP) results obtained using this ranking measure are shown in Table 1. We tested the influence of using SL when we have manually labeled upper body detections and head orientations, and when we use the automatic method described in Section 3.2. Considering the substantial challenges of the task, our results fall within those obtained by state-of-the-art methods in single-action recognition that use similar datasets [7, 10, 12, 22], although a direct comparison is not possible.

In every case the mean AP is improved by the use of SL. This improvement is more obvious in the manually labeled case. When using the automatic method, there are many factors that can account for the smaller degree of improvement when using SL, namely: the inability to always detect both people performing the interaction (SL, as we have employed it, can't improve the results in this case), the appearance of false positives and the incorrect automatic classification of head orientation. In the last two cases, the input to the SL method is corrupted, and attempts to derive a joint classification will most likely produce incorrect results. To give an insight into the difficulty of this task Figure 6 shows the best ranked true and false positives when generating tracks automatically and using the full dataset including negative videos (complete average precision results for this setup are shown in the last two rows of Table 1). We observed that hand shakes tend to be detected where no interaction is happening, this could be because the natural motion of the arms (when walking or talking) resembles the motion pattern of a hand shake in some frames.

5 Conclusion and future work

In this paper we have proposed a new descriptor for human interactions that captures information in a region around a person and uses head orientation to focus attention on specific places inside this region. We have also introduced a new dataset of realistic interactions extracted from TV shows, and have shown good classification and retrieval results using our descriptor. Furthermore, we have shown that using SL to incorporate spatial relationships between detected people in the scene improves the retrieval results obtained by independently classifying each detection.

Several ideas for future work are readily available by analysing the results obtained in Sections 3.3 and 4.2. It's clear that an improvement in the automatic head orientation classification and the automatic generation of video tracks will have a positive effect on the classification and retrieval results. Although concatenating descriptors of consecutive frames didn't improve the classification scores in a consistent way, this may be due to the fact that there wasn't much temporal variance to be captured in the five frames of an interaction that these experiments considered. It is likely that capturing motion and appearance information in longer periods of time could give us a better classification.

Acknowledgements. We are grateful for financial support from CONACYT and ERC grant VisRec no. 228180.

References

[1] B. Benfold and I. Reid. Guiding visual surveillance by tracking human attention. In *British Machine Vision Conference*, 2009.

[2] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *Conference on Computer Vision and Pattern Recognition*, 2005.

[3] N. Dalal, B. Triggs, and C. Schimd. Human Detection Using Oriented Histograms of Flow and Appearance. In *European Conference on Computer Vision*, 2006.

[4] A. Datta, M. Shah, and N. Da Vitoria Lobo. Person-on-Person Violence Detection in Video Data. In *International Conference on Pattern Recognition*, 2002.

[5] C. Desai, D. Ramanan, and C. Fowlkes. Discriminative models for multi-class object layout. In *International Conference on Computer Vision*, 2009.

[6] V. Ferrari, M. Marin-Jimenez, and A. Zisserman. Pose Search: retrieving people using their pose. In *Conference on Computer Vision and Pattern Recognition*, 2009.

[7] A. Gilbert, J. Illingworth, and R. Bowden. Fast Realistic Multi-Action Recognition using Mined Dense Spatio-temporal Features. In *International Conference on Computer Vision*, 2009.

[8] T. Joachims, T. Finley, and C. Yu. Cutting plane training of structural SVMs. *Machine Learning*, 77(1):27–59, 2009.

[9] A. Kläser, M. Marszalek, C. Schmid, and A. Zisserman. Human Focused Action Localization in Video. In *SGA*, 2010.

[10] I. Laptev and P. Perez. Retrieving Actions in Movies. In *International Conference on Computer Vision*, 2007.

[11] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *Conference on Computer Vision and Pattern Recognition*, 2008.

[12] J. Liu, J. Luo, and M. Shah. Recognizing Realistic Actions from Videos "in the Wild". In *Conference on Computer Vision and Pattern Recognition*, 2009.

[13] M Marszalek, I. Laptev, and C. Schmid. Actions in Context. In *Conference on Computer Vision and Pattern Recognition*, 2009.

[14] B. Ni, S. Yan, and A. Kassim. Recognizing Human Group Activities with Localized Causalities. In *Conference on Computer Vision and Pattern Recognition*, 2009.

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

[16] K. Ogawara, Y. Tanabe, R. Kurazume, and T. Hasegawa. Learning Meaninful Interactions from Repetitious Motion Patterns. In *International Conference on Intelligent Robots and Systems*, 2008.

[17] S. Park and J.K. Aggarwal. Simultaneous tracking of multiple body parts of interacting persons. *Computer Vision and Image Understanding*, 102(1):1–21, 2006.

[18] N. Robertson and I. Reid. Estimating gaze direction from low-resolution faces in video. In *European Conference on Computer Vision*, 2006.

[19] M. S. Ryoo and J. K. Aggarwal. Spatio-Temporal Relationship Match: Video Structure Comparison for Recognition of Complex Human Activities. In *International Conference on Computer Vision*, 2009.

[20] I. Tsochantaridis, T. Hofman, T. Joachims, and Y. Altun. Support Vector Machine Learning for Interdependent and Structured Output Spaces. In *International Conference on Machine Learning*, 2004.

[21] G. Willems, J. H. Becker, T. Tuytelaars, and L. Van Gool. Exemplar-based Action Recognition in Video. In *British Machine Vision Conference*, 2009.

[22] X. Wu, C. W. Ngo, J. Li, and Y. Zhang. Localizing Volumetric Motion for Action Recognition in Realistic Videos. In *ACM international conference on Multimedia*, 2009.

[23] B. Yao and L. Fei-Fei. Grouplet: a Structured Image Representation for Recognizing Human and Object Interactions. In *Conference on Computer Vision and Pattern Recognition*, 2010.

[24] W. Zhang, F. Chen, W. Xu, and Y. Du. Hierarchical group process representation in multi-agent activity recognition. *Signal Processing: Image Communication*, 23(10): 739–753, 2008.