Introducing a Inter-frame Relational Feature Model for Pedestrian Detection

Andreas Zweng* and Martin Kampel

Computer Vision Lab
Vienna University of Technology
Favoritenstr. 9/183-2, 1040 Vienna, Austria
{zweng,kampel}@caa.tuwien.ac.at

Abstract. Pedestrian detection has been used with the help of various local features in still images such as histograms of oriented gradients (HOG), local binary patterns (LBP) and more recently, the histograms of optical flow (HOF). In order to improve the robustness of pedestrian detection, movement of people can be taken into the training process which has been done in the HOF descriptor. Optical flow is used to model the movement of a person and to detect actions in image sequences. For action recognition it is necessary to incorporate movement into models when using feature descriptors such as the HOF descriptor. In this paper we introduce a novel method to train and to detect human movement for pedestrian detection using relational gradient features within multiple consecutive frames. The goal of this descriptor is to detect pedestrians using multiple frames for moving cameras instead of static cameras. The relational features between consecutive frames help to robustly find pedestrians in image sequences due to a flexible detection algorithm. We demonstrate the robustness of the resulting feature model computed for a temporal time window of three frames. In our experiments we show the improvement regarding true positives as well as false positives using our inter-frame HOG (ifHOG) model compared to other feature descriptors.

Keywords: pedestrian detection, local features, relational features, machine learning, histograms of oriented gradients.

1 Introduction

Pedestrian detection in images is known as one of the most difficult problems in object detection due to the variable poses of the human body and the changing illumination conditions in outdoor scenarios. Dalal [1] introduced the Histogram of Oriented Gradients (HOG) feature for people detection which is robust against illumination due to the use of image gradients. A Support Vector Machine (SVM) is used with a linear kernel to train the computed features and classifies image regions using a sliding window through all possible image positions in multiple

* The work was partly supported by the Austrian Research Promotion Agency (FFG), project 830041 (SCIBA) and CogVis Ltd.

J.-K. Kämäräinen and M. Koskela (Eds.): SCIA 2013, LNCS 7944, pp. 225–235, 2013.

image scales. Feature combinations of the HOG features and the local binary pattern (LBP) features are used by Wang et al. in [2] to extend the work of Dalal. Wen introduced the local ternary patterns (LTP) as an extension to the LBP in order to describe regions and classify textures [3]. The LTP descriptor addresses the biggest problem of LBP, which is the sensitivity to noise, by handling small differences of pixel comparisons separately. Felzenszwalb et al. proposed an approach to detect instances of a body such as the legs, arms or the upper body [4]. The body parts are detected separately in order to allow deformable configurations of the human body. A part based approach is also used by Bo [5] in order to handle partial occlusions. They use edgelet features in combination with a three part model as well as a full-body detector. The used parts of the body are head-shoulder, torso and legs. Ronfard et al. propose dedicated detectors learned for all defined body parts using SVMs as well as Relevance Vector Machines (RVM) which give similar results compared to SVMs with many fewer kernels [6]. They developed a body tree parsing algorithm to find the best body part candidates based on geometric constraints and the detected body parts as input.

Recently, proposed detection methods use relations of features in combination with stochastic learning [7–9]. The idea is to compute a co-occurrence matrix with a precomputed set of HOG features which represents salient characteristics of the object structure. Yamauchi et al. proposed a relational HOG feature model which focuses on histogram binarization using the size relationship [10]. They use 8 directional histogram bins in the HOG feature computation to combine the 8 binarized bins to a 8 bit variable for each resulting histogram to reduce the memory requirement. This idea was further developed by Zweng and Kampel by using more accurate comparison functions for relational feature extraction [11]. In order to incorporate motion into a trained model, Viola et al. [12] introduce motion features using two consecutive frames combined with image intensity information using images from a static camera which constrains the use of this algorithm. Dalal et al. [13] improve the idea of using consecutive video frames and motion patterns for object detection by embracing camera movement which makes the detector more robust for dynamic movement in a scene and applicable for realistic sequences such as TV content. Laptev et al. [14] introduce histograms of optical flow (HOF) from space-time volumes in order to characterize motion and appearance of local features. They detect human actions such as kissing, answering a phone and getting out of a car with the HOF descriptor. A detection approach using these features will fail when people stand still, due to the lack of movement, which is needed by the HOF feature descriptor. Klser et al. [15] use a spatio-temporal gradient descriptor to recognize actions. They compute histograms from a subdivided cuboid representing the spatial and temporal dimensions.

In this work we introduce a feature descriptor to detect people using relations between HOG features from consecutive frames. This descriptor aims to resolve problems of the state of the art descriptors such as the limitation to a static camera. With the use of histogram similarity functions, such as the bhattacharyya

distance, histogram intersection, histogram correlation and the χ^2 histogram similarity function, a similarity between two histograms of two different frames is computed. Our main contribution is a new feature model, the inter-frame relational HOG feature descriptor, including a training algorithm and an efficient detection algorithm. The paper is structured as follows. The following section describes the inter-frame relational HOG descriptor, followed by a section which describes the training algorithm and a section which describes the detection algorithm. In Section 5 we show qualitative as well as quantitative results with varying settings of the feature detectors. Section 6 concludes this paper with a discussion and future work.

2 Inter-frame Relational HOG Model

People detection with the HOG descriptor is done using single frames with an overlapping sliding window to cover all areas in the image. In order to achieve a more robust detection, multiple frames can be taken into computation [12–15]. The inter-frame relational HOG model computes relational features using HOG features from [11] between consecutive frames. We have chosen the number of consecutive frames for one detection to three frames, where the relations are computed between the first and the second frame as well as between the second and the third frame. The two resulting feature vectors are combined to one longer feature vector for each sample. We used the best histogram similarity function for relational feature computation between consecutive frames which was determined by Zweng and Kampel [11] by experiments - the (χ^2) histogram similarity measure using a cell size of 12 by 12 pixels.

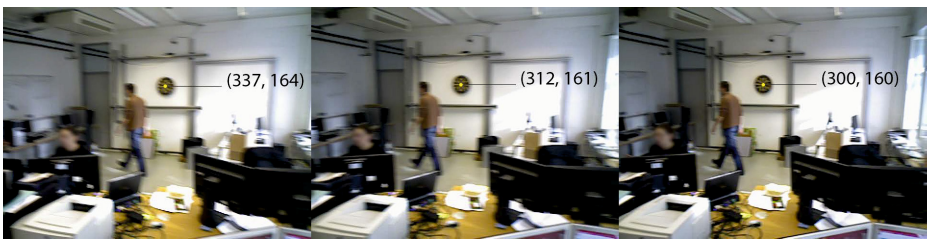

Fig. 1. Three consecutive frames using a moving camera. The coordinates of the centroid of the dartboard are taken to demonstrate the amount of movement between frames.

The goal of this descriptor is to minimize the difficulty of detecting a person for moving cameras. There are two main difficulties for object detection with moving cameras. Motion blur caused by rapid movement of the camera decreases the magnitude of gradients for feature computation and increases the gap of the same objects (i.e. pedestrians) between frames at the same time. Because motion blur

does not effect the classification performance as much as the distances of the same objects between frames, the inter-frame relational feature model concentrates on compensation of large distances between the same objects between frames. An example of three consecutive frames with a moving camera is illustrated in Fig. 1. The idea of the inter-frame relational feature model is illustrated in Fig. 2. The relations using the histogram similarity function (in this work we used the χ^2 similarity function as it was evaluated as the best choice in [11]) are computed between the first frame and the second frame and between the second and the third frame. The results are then concatenated to one final feature vector. For simplification, in Fig. 2 only selected blocks are taken for computation of the relations. The relations of two consecutive histograms can be computed as follows.

$$r_i = f(h_a^i, h_b^i) \tag{1}$$

where r_i is relation i, h_a^i is the i^{th} feature of the precompute HOG feature vector of the search window in frame n, h_b^i is the i^{th} feature of the precompute HOG feature vector of the search window in frame $n+1$ and $f()$ is the chosen similarity function (i.e. χ^2).

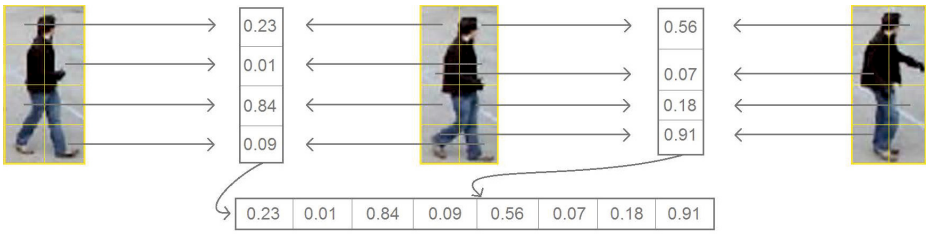

Fig. 2. The idea of the inter-frame relational feature model

3 Training Algorithm

For the training stage, we manually segmented people in consecutive frames for a set of three frames for one positive training sample. The negative samples are combined from three randomly selected negative images. We did experiments, where we also used two negative images and one positive image for one negative sample in order to reduce the number of false positives during detection. The output is a decreasing number of true positves which forced us to use three negative images for each negative sample. Examples of positive samples are illustrated in Fig. 3. The samples include standing people and walking people in all kinds of directions. We used all samples to train only one model with a linear kernel using a support vector machine. For our experiments, the positive training set contains 1303 positive samples and 2120 negative samples.

Fig. 3. Examples of positive training samples

4 Detection Algorithm

The detection algorithm uses three frames at once for the detection in one frame. For each position of the sliding window in frame 1, the nine neighbour windows (including the actual position) of frame 2 are used to compute the relations between the computed HOG features at those positions. For each of the nine positions of frame 2, the nine neighbour windows of frame 3 are used to compute the second vector of relational features. The neighbour windows have a distance of 8 pixels horizontally and/or vertically. The result of this approach are 81 combinations of relational features for each detection window which means, that 81 HOG windows have to be computed, compared to only one HOG computation for a single frame approach. This was the reason, why we chose only three frames for the inter-frame feature model - a four frame inter-frame feature model would already require 729 HOG computations for each window in one frame. The idea of using the neighbour windows in consecutive frames to assemble the relational features is illustrated in Fig. 4.

We implemented the algorithm in Matlab with an initial computation time of three hours for one frame. The computation in one frame includes three scales of the input image in order to detect people of different sizes. Initially we used the support vectors for the detection as it has to be done for RBF (radial basis function) kernels but not for linear kernels. With linear kernels, the weight vector of the support vectors can be used to compute the same result of the obtained features from a search window. The computational difference between the two classification calculations depends on the number of support vectors obtained during training. In our training algorithm, the trained SVM has 400 support vectors and the computation of the weight vector is done at the end of the training stage. Using the weight vector for classification we could reduce the computation time to 11 minutes for each frame. Another computational bottleneck is the fact, that the HOG features are computed several times for each window caused by the approach which computes 81 combinations of relational features using the neighbour windows. The solution to this problem is to precompute the HOG features for each frame and save it in a lookup table. With that tweak, the computation time dropped to 90 seconds per frame. Another significant improvement is to save the precomputed features for the next frames. In frame 1, the algorithm computes all features from frame 1, frame 2 and frame 3 and saves the features from frame 2 and frame 3. In the second frame, the

algorithms sets the features from the previous computation from frame 2 and frame 3 to the current features from frame 1 and frame 2, since the frames are consecutive. Therefore, the HOG feature computation is only done in one frame which drops the computation time to 3 seconds per frame. We also planned to save all relational features for the next frames, but due to the huge amount of data, we could only temporarily save relational features in small regions. We store the features as 32 bit floating point variables and one feature vector has 7560 features. In total, the three scales require 12514 sliding window positions with our parameter settings and the number of combinations for the computation of the relational features is 81 for each window. In total, we would require approximately 96 gigabytes of memory to save all combinations of relational features. With a few minor tweaks we managed to improve the performance to 2 seconds per frame and we believe that the algorithm can be implemented with real time performance using a low level programming language such as C or C++.

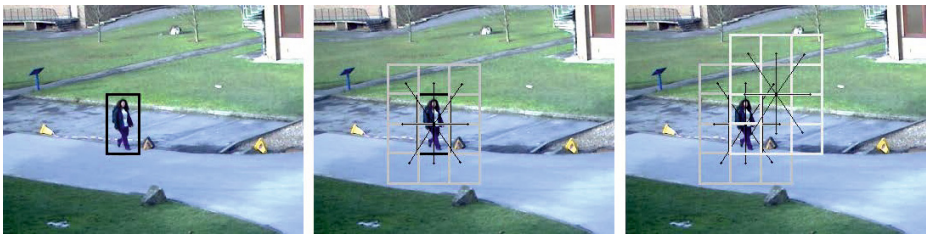

Fig. 4. Possible combinations for detection using inter-frame features. For one detection window in frame 1, it requires 81 combinations of feature vectors when using three frames.

5 Experiments

We have tested our approach on 795 images of the PETS 2013 dataset and two sequences from our laboratory containing 611 and 636 images. For training we used a different sequence of the PETS 2013 dataset, where we manually annotated people. Three consecutive frames of the same person are used as one positive sample in order to compute the inter-frame relational features. Our experiments on descriptors are based on re-implementations and open source projects of the evaluated approaches. We made experiments using different settings of cell sizes for the relational feature model (RFM) computation and different histogram similarity functions. Our implementation is done with Matlab and the built-in SVM. We evaluate all approaches with the same settings for the HOG feature computation and the same amount of scales for the sliding window detection except the cell size. In our classification implementation we have set the step size of the sliding window to 8 pixels and we use three image scales which results in 5120 classification positions for the initial scale, 4118 positions for the second scale and 3276 positions for the third scale, where the images are scaled down to 90% relative to the previous image resolution.

5.1 Qualitative Results

Qualitative experiments on the PETS 2013 dataset show, that the χ^2 histogram similarity function achieves the best results for the RFM compared to the intersection similarity function, the bhattacharyya distance and the pearson product-moment correlation coefficient. We evaluated those functions on several images with two different block normalization methods in the HOG computation. In Fig. 6 results are shown on the different histogram similarity functions using the *L2-hys* block normalization method and the *L2-norm* block normalization method. The red rectangles denote positive responses from the SVM in that particular area and the green, thicker rectangles denote the final detection using a grouping algorithm based on rectangle similarities, overlap and distances to each other. All methods achieve better results in terms of true positives as well as false positives with the *L2-norm* block normalization method compared to the *L2-hys* block normalization method except for the correlation histogram measure, which has the same result for both normalization methods. A significant improvement of the *L2-norm* method compared to *L2-hys* can be seen on the top of the street lamp. All methods have less false positives in this area with an additional improvement in the number of true positives in the image.

We have also compared the results of the standard HOG descriptor, the RFM and our ifHOG implementation in more detail. In Fig. 5 the three approaches are compared to each other. The ifHOG clearly outperforms the other approaches in terms of false positives as well as true positives. An interesting observation is, that the positive detections are closer to the people and the rectangles are smaller as can be seen best on the person in the center of Fig. 5.

5.2 Quantitative Results

Our quantitative evaluation covers evaluation of 795 images of the PETS 2013 dataset using the HOG descriptor, the RFM from [11] using the χ^2 measure combined with a 12 by 12 cell size and different normalization methods, the relational HOG features from [10] and our ifHOG descriptor. Fig. 7 shows the results of those descriptors on the dataset. The RFM is computed using the *L2-hys* and

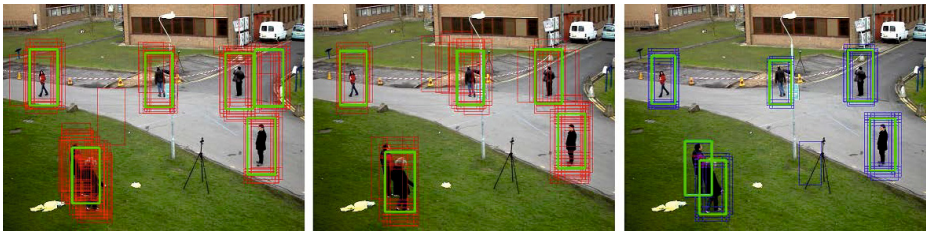

Fig. 5. Detection results using HOG, RFM and ifHOG

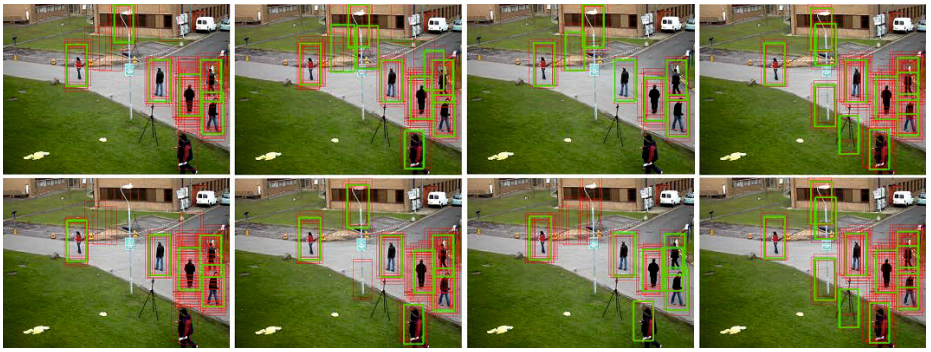

Fig. 6. Left to right: χ^2, intersection, bhattacharyya and correlation measure. Top to bottom: *L2-hys* block normalization and *L2-norm* block normalization.

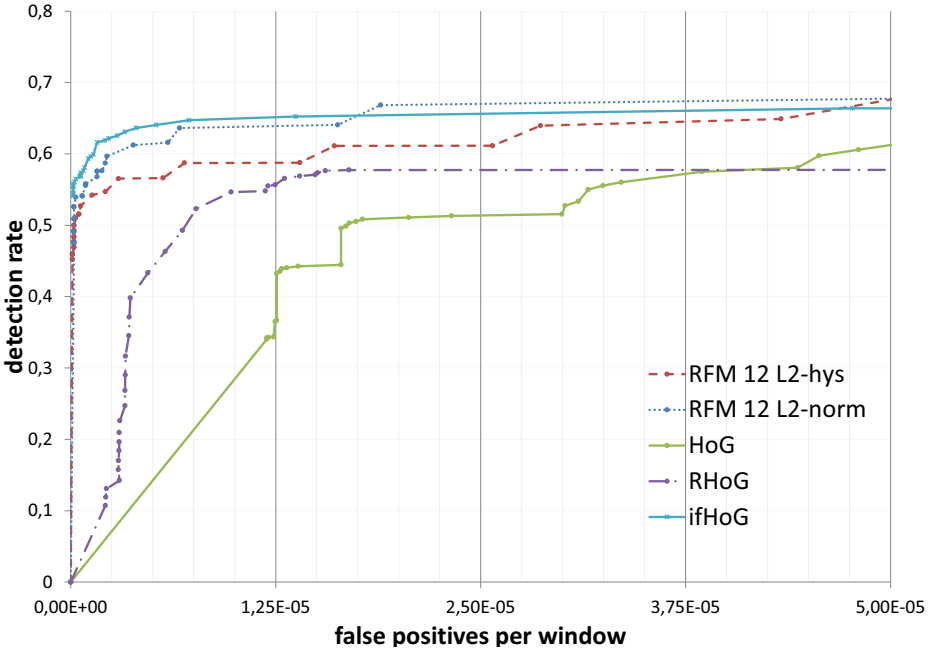

Fig. 7. Evaluation on 795 images of the PETS 2013 dataset with HOG, RHOG, RFM12 with *L2-hys* and *L2-norm* and ifHOG

L2-norm normalization method, where the *L2-norm* method outperforms the *L2-hys* method. Our ifHOG descriptor outperforms all other descriptors in the area of 0 and $5 \cdot 10^{-5}$ false positives per window which can be better seen in a zoom of the graphs in Fig. 9. Our evaluation is based on reimplementations of the mentioned algorithms in order to have an in-depth evaluation on our dataset.

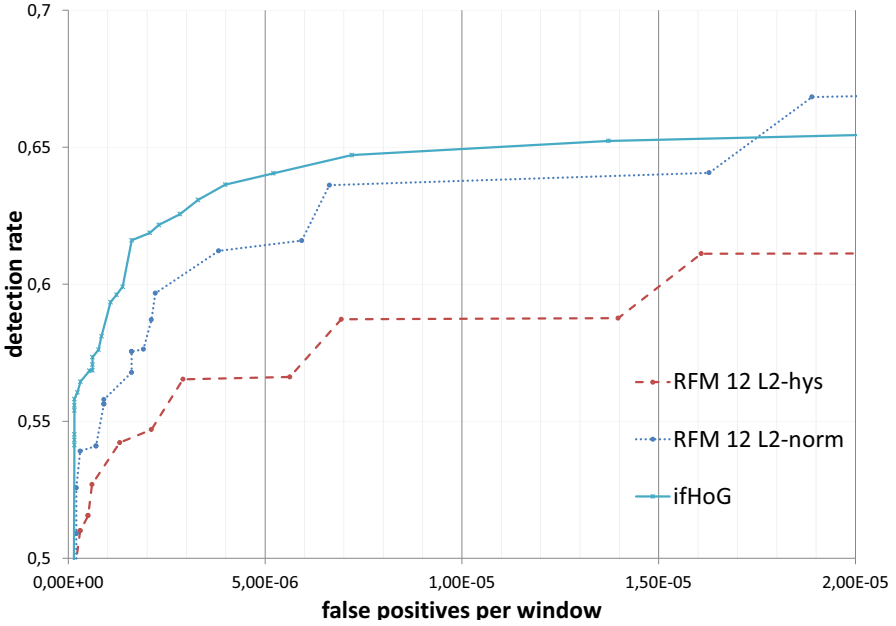

Fig. 8. Performance of RFM12 with *L2-hys* and *L2-norm* and ifHOG in the area of $5 \cdot 10^{-5}$ false positives per window

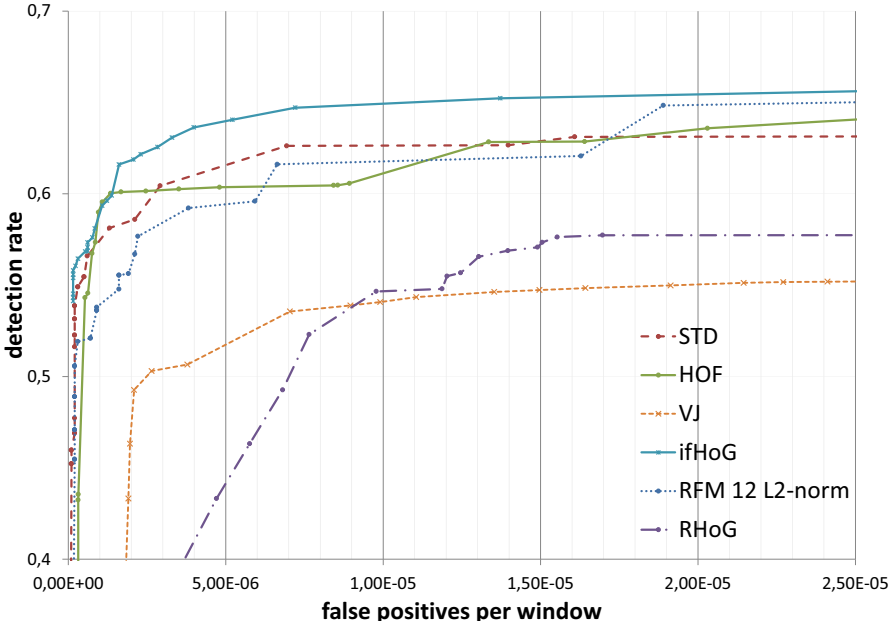

Fig. 9. Detection performance of various feature descriptors on the laboratory dataset

Our second evaluation is done on two image sequences with moving cameras and the tested descriptors are the spatial-temporal descriptor (STD) from [15], the HOG/HOF descriptor from [13], the feature descriptor from Viola et al. (VJ) from [12], our inter-frame relational feature descriptor ($ifHOG$), the improved relational feature model ($RFM12$) from [11] and the relational HOG model ($RHOG$) from [10]. Below the recognition rate of 0.6, the HOF descriptor has a very low false positive rate. This is due to the robust motion features in their work which achieves a very low false positive rate. The spatio-temporal descriptor from Klser et al. performs better than the motion descriptors for those sequences since it uses 3D gradients optimized for action recognition. For a very low false positive rate as well as for higher recognition rates, our descriptor outperforms the other approaches due to the fact, that it combines the advantages of motion descriptors and descriptors such as the spatio-temporal descriptor from Klser et al.

6 Conclusions and Future Work

We have shown the improved robustness of a feature descriptor using relations of features and inter-frame comparisons (ifHOG) compared to single-frame appearance descriptors and motion descriptors. We have also shown, that this approach can be used for real time detection when our optimized algorithms are implemented in a low-level programming language like C or C++. Our future work includes the implementation and evaluation of other local features for the relational feature model. We also want to compare other histogram similarity functions in order to find the best combination of histogram similarity function for computation of the relational features and feature descriptors.

References

1. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2005), vol. 1, pp. 886–893 (June 2005)
2. Wang, X., Han, T.X., Yan, S.: An hog-lbp human detector with partial occlusion handling. In: IEEE 12th International Conference on Computer Vision (ICCV 2009), pp. 32–39 (October 2009)
3. Liao, W.H.: Region description using extended local ternary patterns. In: 20th International Conference on Pattern Recognition (ICPR 2010), pp. 1003–1006 (August 2010)
4. Felzenszwalb, P.F., Huttenlocher, D.P.: Pictorial structures for object recognition. International Journal of Computer Vision (IJCV 2005) 61(1), 55–79 (2005)
5. Wu, B., Nevatia, R.: Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In: 10th IEEE International Conference on Computer Vision (ICCV 2005), vol. 1, pp. 90–97 (October 2005)
6. Ronfard, R., Schmid, C., Triggs, B.: Learning to parse pictures of people. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part IV. LNCS, vol. 2353, pp. 700–714. Springer, Heidelberg (2002)

7. Watanabe, T., Ito, S., Yokoi, K.: Co-occurrence histograms of oriented gradients for pedestrian detection. In: Wada, T., Huang, F., Lin, S. (eds.) PSIVT 2009. LNCS, vol. 5414, pp. 37–47. Springer, Heidelberg (2009)
8. Tuzel, O., Porikli, F., Meer, P.: Human detection via classification on riemannian manifolds. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2007), pp. 1–8 (June 2007)
9. Ren, H., Heng, C.K., Zheng, W., Liang, L., Chen, X.: Fast object detection using boosted co-occurrence histograms of oriented gradients. In: 17th IEEE International Conference on Image Processing (ICIP 2010), pp. 2705–2708 (September 2010)
10. Yamauchi, Y., Matsushima, C., Yamashita, T., Fujiyoshi, H.: Relational hog feature with wild-card for object detection. In: IEEE International Conference on Computer Vision Workshops (ICCV 2011 Workshops), pp. 1785–1792 (November 2011)
11. Zweng, A., Kampel, M.: Improved relational feature model for people detection using histogram similarity functions. In: IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance (AVSS 2012), pp. 422–427 (September 2012)
12. Viola, P., Jones, M.J., Snow, D.: Detecting pedestrians using patterns of motion and appearance. International Journal of Computer Vision (IJCV 2005) 63(2), 153–161 (2005)
13. Dalal, N., Triggs, B., Schmid, C.: Human detection using oriented histograms of flow and appearance. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 428–441. Springer, Heidelberg (2006)
14. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2008), pp. 1–8 (June 2008)
15. Kläser, A., Marszałek, M., Schmid, C.: A spatio-temporal descriptor based on 3d-gradients. In: British Machine Vision Conference, pp. 995–1004 (September 2008)