Multi-Cue Onboard Pedestrian Detection

Christian Wojek Stefan Walk Bernt Schiele
Computer Science Department
TU Darmstadt, Germany
{wojek, walk, schiele}@cs.tu-darmstadt.de

Abstract

Various powerful people detection methods exist. Surprisingly, most approaches rely on static image features only despite the obvious potential of motion information for people detection. This paper systematically evaluates different features and classifiers in a sliding-window framework. First, our experiments indicate that incorporating motion information improves detection performance significantly. Second, the combination of multiple and complementary feature types can also help improve performance. And third, the choice of the classifier-feature combination and several implementation details are crucial to reach best performance. In contrast to many recent papers experimental results are reported for four different datasets rather than using a single one. Three of them are taken from the literature allowing for direct comparison. The fourth dataset is newly recorded using an onboard camera driving through urban environment. Consequently this dataset is more realistic and more challenging than any currently available dataset.

1. Introduction

Detecting pedestrians using an onboard camera is a challenging problem but an important component e.g. for robotics and automotive safety applications. While psychologists and neuroscientists argue that motion is an important cue for human perception [17] only few computer vision object detectors (e.g. [30, 6]) exploit this fact. Interestingly, [30] showed improved detection performance but for static cameras only. It is unclear how to transfer their results to onboard sequences. In contrast, [6] proposed motion features that are – at least in principle – applicable to onboard sequences. While [6] showed improved performance using the FPPW evaluation criterion (False Positives per Window) they were unable to outperform their own static HOG feature [5] in a complete detector setting [4].

The second avenue we follow in this paper is to incorporate multiple and complementary features for detection.

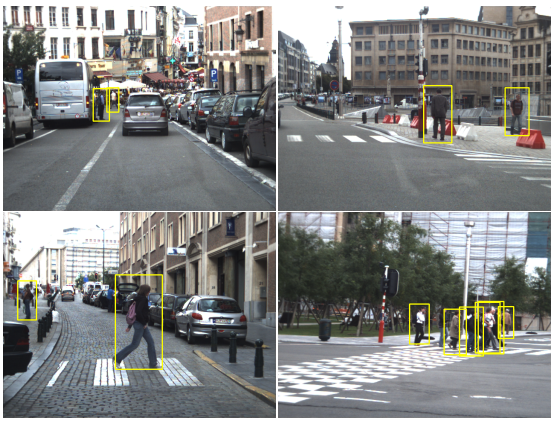


Figure 1: Detections obtained with our detector in an urban environment

While [29] convincingly showed that multiple features improve performance for image classification, for detection only few approaches exploit this fact [34, 16, 32].

The third avenue of this paper is related to the classifier choice. Popular classifiers are SVMs [26, 5, 13, 20, 19] or boosting [31, 7, 33, 23]. However, the large intra-class variability of pedestrians seems to require a more careful design of the classifier framework. Several authors have argued that e.g. viewpoint variation requires a different classifier design. Wu&Nevatia [33] remedy this issue by learning a tree structured classifier, Lin&Davis [19] use a handcrafted hierarchy, while Seemann et al. [25] propose multi-articulation learning. Gavrila [15] proposes a tree-structured Bayesian approach that builds on offline clustering of pedestrian shapes. What is common to these approaches is that they treat the problem of data partitioning and classifier learning separately. In this paper however we address this problem in a more principled way by using the MPLBoost classifier [2] that simultaneously learns the data partitions and a strong classifier for each partition.

The main focus of this work is to advance the state-of-the-art in pedestrian detection for realistic and challenging onboard datasets. For this we experimentally evaluate combinations of features and classifiers and address the problem of learning a multi-viewpoint pedestrian detector.

Our contribution is threefold. Firstly, we show that mo-

978-1-4244-3991-1/09/$25.00 ©2009 IEEE

tion cues provide a valuable feature, also for detection from a moving platform. Secondly, we show that MPLBoost and histogram intersection kernel SVMs can successfully learn a multi-viewpoint pedestrian detector and often outperform linear SVMs. Thirdly, a new realistic and publicly available onboard dataset (*TUD-Brussels*) containing multi-viewpoint data is introduced. It is accompanied by one of the first training datasets (*TUD-MotionPairs*) containing image pairs which allow to extract and train from motion features. These two datasets will enable comparison of different approaches based on motion. Besides these contributions we discuss several important algorithmic details that prove important and that are often neglected and overlooked.

The paper is structured as follows. Sec. 2 reviews related work. Sec. 3 introduces features and classifiers and Sec. 4 discusses several important technical details. Sec. 5 introduces datasets while Sec. 6 discusses the experimental results and Sec. 7 concludes.

2. Related Work

Within the last years a number of systems and detectors have been presented to tackle the problem of detecting pedestrians from a moving platform such as a driving car or a robot. This reflects the growing interest in applications such as automotive safety and robotics scenarios.

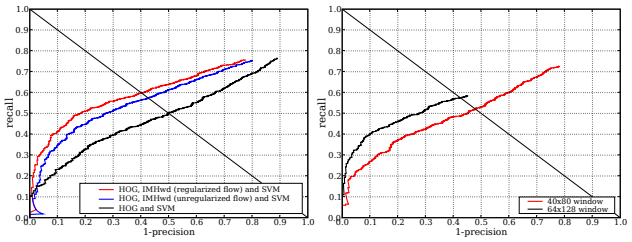
Early work in pedestrian detection started with Papageorgiou&Poggio [22] who employ Haar features in combination with a polynomial SVM in order to detect pedestrians. Sashua et al. [26] use parts and employ histograms of gradients as features. Similarly, Dalal&Triggs [5] train SVMs on histograms of oriented gradients features (HOG) and achieve good performance. An extension by Felzenszwalb et al. [13] adds a flexible part model where the position of the parts is considered as latent variable for the SVM learning algorithm. Similarly, Dollár et al. [7] present an approach that automatically learns flexible parts from training data and uses a boosting framework with wavelet features. Also Tuzel et al. [28] employ LogitBoost on Riemannian manifolds to classify windows based on covariance features. Sabzmeydani&Mori [23] learn low level features on gradient responses and use AdaBoost to combine them. Maji et al. [20] approximate the evaluation of histogram intersection kernels and use a kernel SVM in conjunction with a hierarchy of gradient histograms as features. Tran&Forsyth [27] learn a model of human body configurations and use local histograms of gradients and local PCA of gradients as features. In [33] Wu&Nevatia propose a system to automatically construct tree hierarchies for the problem of multi-view pedestrian detection. They use a boosting framework in combination with edgelet features.

Most detectors in this domain as well as ours employ the sliding-window scheme, but notable exceptions exist [21, 24, 1]. These methods are based on keypoint detectors and a probabilistic voting scheme to accumulate evidence. Andriluka et al. [1] additionally model the human walking cycle and infer a consistent movement within the temporal neighborhood.

With the availability of acceptably performing detectors some approaches use them as component in systems and add further reasoning such as tracking and 3D scene geometry in order to improve the initial detections. While Ess et al. [11] extract the ground plane from a depth map and fuse it with detections in a graphical model, Ess et al. [12] add further 3D scene information by integration with Structure-from-Motion. Gavrilla&Munder [16] propose a pipeline of Chamfer matching and several image based verification steps for a stereo camera setup. While they optimize overall system performance we focus on the detector part and improve it by the combination of multiple features.

Even though the combination of multiple features should allow for increased detection performance only few approaches leverage from the complementarity of different object representations. Wu&Nevatia [34] automatically learn the efficiency-discriminance tradeoff in a boosting cascade for HOG, edgelet and covariance features but with a focus on runtime. In particular human motion information can be a rich source as shown by Viola et al. [30]. Their motion features proved to be the most discriminative features. However, their work is restricted to a static camera setting, while we would like to detect people from a moving platform. Dalal et al. [6] enrich their static feature descriptor [5] with internal motion histograms to improve detection. Their database consists of movies and is not publicly available. Movies contain rather little ego-motion, in particular little translation along the optical axis of the camera. Thus, it is unclear whether their results also apply to sequences taken from a car e.g. traveling at the typical inner-city speed of 50 km/h (30 mph). Moreover, their detectors' performance is only shown to improve in terms of FPPW but not in a full image detector setting [4]. We will show that the choice of the non-maximum suppression strategy is crucial to obtain best performance. A further difference to their approach is the choice of optical flow; while they use an unregularized flow method, we found globally regularized flows [35] to work better. Additionally, we show that the combination with additional features (such as Haar wavelets [22]) can allow for further improvements. Enzweiler et al. [10] use motion information in a ROI generation step for an onboard system, while we investigate the use of motion features for the detector.

Several authors have evaluated features and their combinations with different classifiers. However, all of them are limited to static images only and do not include motion based features. For people detection Wojek&Schiele [32]



(a) Performance for different flow algorithms – using the regularized flow algorithm by Zach et al. [35] works best

(b) Performance drops when using a smaller detection window.

Figure 2: Impact of flow algorithm and detection window size

evaluated different static image features in combination with AdaBoost and SVMs as classifiers. Dollár et al. [8] report results for various recent approaches on a new challenging onboard dataset with improved evaluation metrics. Enzweiler&Gavrila [9] evaluate different detectors in combination with an onboard system with focus on performance and runtime.

3. Features and Classifiers

In the following subsections we will discuss the features (Sec. 3.1) and classifiers (Sec. 3.2) which we deploy in a sliding window framework.

3.1. Features

A wide range of features has been proposed for pedestrian detection. Here, we focus on three successful features containing complementary information (see [32] for a wider range of features). While HOG features encode high frequency gradient information, Haar wavelets encode lower frequency changes in the color channels. Oriented Histograms of Flow features exploit optical flow and thus a complementary cue.

HOG Histograms of oriented gradients have originally been proposed by Dalal&Triggs [5]. The bounding box is divided into 8×8 pixel *cells* containing histograms of oriented gradients. 2×2 cells constitute a *block* which is the neighborhood to perform normalization. For people detection L^2-norm with an additional hysteresis performs best.

Haar Haar wavelets have been introduced by Papageorgiou&Poggio [22] for people detection. Those provide an overcomplete representation using features at the scale of 32 and 16 pixels. Similarly to HOG blocks, wavelets overlap by 75%. As proposed we use the absolute responses of horizontal, vertical and diagonal wavelet types.

Oriented Histograms of Flow The motion feature we use throughout this paper is the Internal Motion Histogram wavelet difference (IMHwd) descriptor described by Dalal et al. in [4, 6]. The descriptor combines 9 bins per histogram on 8×8 pixel cells, with interpolation only for histogram bins. It is computed by applying wavelet-like operators on a 3×3 cell grid, letting pixel-wise differences of flow vectors vote into histogram bins. We use IMHwd due to its consistently better performance in previous experiments compared to other proposed descriptors. The flow field is computed using the TV-L^1 algorithm by Zach et al. [35], which provides regularization while allowing for discontinuities in the flow field. Contrary to [4], we compute the optical flow for the training samples on full images instead of crops, which is particularly important for the regularized TV-L^1 flow. We also conducted experiments with the unregularized flow algorithm described in [4], but it resulted in a slight loss of performance compared to the algorithm by Zach et al. [35] (cf. Fig. 2(a)). For further discussion see Sec. 6.

Feature combination In the experiments reported below we analyze various combinations of the above features. To combine features we L^2-normalize each cue-component and concatenate all subvectors.

3.2. Classifiers

The second major component for sliding window based detection systems is the employed classifier. Most popular choices are linear SVMs and AdaBoost. As discussed before these are not perfectly suited because of the high intra-class variability of humans e.g. caused by multiple viewpoints and appearance differences. In this paper we therefore explore the applicability of MPLBoost that learns data clusters and strong classifiers for these clusters simultaneously.

SVM Linear SVMs learn the hyperplane that optimally separates pedestrians from background in a high-dimensional feature space. Extensions to kernel SVMs are possible, allowing to transfer the data to a higher and potentially infinity dimensional representation as for RBF kernels. For detection however, kernel SVMs are rarely used due to higher computational load. One remarkable exception is Maji et al. [20] who approximate the histogram intersection kernel for faster execution. Their proposed approximation is used in our experiments as well.

AdaBoost Contrary to SVMs, boosting algorithms [14] optimize the classification error on the training samples iteratively. Each round a weak classifier is chosen in order to optimally reduce the training error. The weighted sum of all weak classifiers forms the final strong classifier. A typical choice for weak learners, which are required to do better than chance, are decision tree stumps operating on a single dimension of the feature vector. In this work, we use AdaBoost as formulated by Viola and Jones [31].

MPLBoost MPLBoost [2] (or MCBoost [18]) is a recently proposed extension to AdaBoost. While AdaBoost fails to learn a classifier where positive samples appear in multi-

ple clusters arranged in a XOR-like layout, MPLBoost successfully manages this learning problem. This is achieved by simultaneously learning K strong classifiers, while the response to an input pattern is given as the maximum response of all K strong classifiers. Thus, a window is classified as positive if a single strong classifier yields a positive score and negative only if all strong classifiers consider the window as negative. Also the runtime is only linear in the number of weak classifiers. During the learning phase positive samples which are misclassified by all strong classifiers obtain a high weight, while positive samples which are classified correctly by a single strong classifier are assigned a low weight. This enables the learning algorithm to focus on a subpart of misclassified data (up to the current round) with a single strong classifier. Other strong classifiers are not affected and therefore do not loose their discriminative power on their specific clusters learned.

4. Learning and Testing

While features and classifiers are the key components of the detectors several issues need to be taken care of for both learning and testing. Those details are often crucial to obtain best performance, even though they are seldom discussed in literature. The following sections give some detailed insights on our learning (Sec. 4.1) and testing procedure (Sec. 4.2).

4.1. Improved Learning Procedure

Our classifiers are trained in a two-step bootstrapping process. In order to improve the statistics of hard examples for the domain where pedestrians actually appear, the negative test set also contains frames from an onboard camera recorded in an urban area. Those are scanned for hard examples, but detections that are close to a pedestrian in x-y-$scale$-space are considered true positive. The minimal distance is chosen such that detections on body parts are allowed as hard examples.

Often these types of false positives are not well represented in other detectors' training data. Fig. 3 shows highest scoring false positive detections in the bootstrapping phase after removing the full detections, showing that body parts are indeed hard examples for the initial detector.

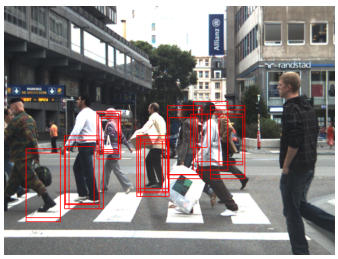


Figure 3: False positive detections with high scores before the bootstrapping stage. Detections close to pedestrians are true positives and not shown here.

Additionally, we found that merging the false positive detections on the negative images by mean shift is beneficial in several ways. First, the variability of false positive detections for the second round of training can be increased and the space of negative samples is covered well, while keeping the memory requirements reasonable. Second, false positive regions with a larger number of false detections are not overcounted since they will only be contained once in the training set and thus have the same weight as regions on which the detectors only fires a few times. This is consistent with the fact that for real-world systems the optimal image-based performance is sought and all false detections should be treated equally.

4.2. Testing

As it is desirable for real-world applications to detect pedestrians as soon as possible we are aiming to detect pedestrians as small as possible. Empirically we found that given appropriate image quality upscaling the input image allows for a better performance gain with respect to small detections than shrinking the detection window (cf. Fig. 2(b)). Therefore, we upscale the input image by a factor of two which allows to detect pedestrians as small as 48 pixels with a 64×128 pixel detection window (the window contains context in addition to the pedestrian). Sliding-window based detection systems usually fire multiple times on true pedestrians on nearby positions in scale and space. These detections need to be merged in order to allow for a per-image based evaluation such as false positive per image (FPPI) or precision and recall (PR). Here, we adopt an adapted bandwidth mean-shift based mode seeking strategy [3] to determine the position in x-y-$scale$-space, but determine the final detection's score to be the maximum of all scores within the mode. While others (e.g. [4]) have used the kernel density to form the final score, we found the maximum to provide more robust results. While most of the time the performance is comparable, in some cases choosing the kernel density leads to a significantly decreased performance in particular for the motion-enhanced detector (cf. Fig. 5(l)). Another important issue is the estimation of the kernel density – in a scale pyramid setting with a constant pixel stride for every scale, detections on larger scales are sparser. Thus, contrary to [4] when computing the kernel density we omit the kernel volume's scale adaption for the normalization factor.

5. New Dataset

To the best of our knowledge the sequences of [11, 12] are currently the only publicly available video sequences for pedestrian detection recorded from a moving platform. While those are realistic for robotics scenarios, they are less realistic for automotive safety applications. This is mainly due to the relatively small ego-motion and the camera's field of view which is focusing on the near range. In order to show results for a more realistic and challenging automotive safety scenario in urban environment, we captured a new onboard dataset (*TUD-Brussels*) from a driving car. Note

Figure 4: Positive sample crops and flow fields of *TUD-MotionPairs*.

that [8] simultaneously introduces a new onboard dataset but evaluates static features only.

At the same time there is no dedicated training set containing temporal image pairs which has sufficient variability to train a discriminative detector based on motion features. Thus, we additionally recorded a new training dataset (*TUD-MotionPairs*) containing pairs of images to compute optical flow. Both new datasets are made publicly available[1].

Training sets Our new positive training set (*TUD-MotionPairs*) consists of 1092 image pairs with 1776 annotated pedestrians (resulting in 3552 positive samples with mirroring), recorded from a hand-held camera at a resolution of 720×576 pixels. The images are recorded in busy pedestrian zones. Some samples are shown in Fig. 4. Note that contrary to [5] our data base is not restricted to upright standing pedestrians but also contains pedestrians from side views which are particularly relevant in applications due to the possibility of crossing the camera's own trajectory.

Our negative training set consists of 192 image pairs. 85 image pairs were recorded in an inner city district, using the same camera as was used for the positive dataset at a resolution of 720×576 pixels, while another 107 image pairs were recorded from a moving car. For finding body parts as hard samples as described in Sec. 4.1 we use an additional set of 26 image pairs, recorded from a moving vehicle containing 183 pedestrian annotations. We use this training set for all experiments throughout this paper.

Test sets The new *TUD-Brussels* dataset is recorded from a driving car in the inner city of Brussels. The set contains 508 image pairs (one pair per second and its successor of the original video) at a resolution of 640×480 with overall 1326 annotated pedestrians. The dataset is challenging due to the fact that pedestrians appear from multiple viewpoints and at very small scales. Additionally, many pedestrians are partially occluded (mostly by cars) and the background is cluttered (e.g. poles, parking cars and buildings and people crowds) as typical for busy city districts. The use of motion information is complicated not only by the fact that the camera is moving, but also by the facts, that the speed is varying and the car is turning. Some sample views are given in Fig. 1.

Additionally we evaluate our detectors on the publicly available ETH-Person [11] dataset. In [11], Ess et al. presented three datasets of 640×480 pixel stereo images recorded in a pedestrian zone from a moving stroller. The camera is moving forward at a moderate speed with only minor rotation. The sets contain 999, 450 and 354 consecutive frames of the left camera and 5193, 2359 and 1828 annotations respectively. As our detector detected many pedestrians below the minimum annotation height in these sets, we complemented the sets with annotations for the smaller pedestrians. Thus, all pedestrians with a height of at least 48 pixels are considered for our evaluation.

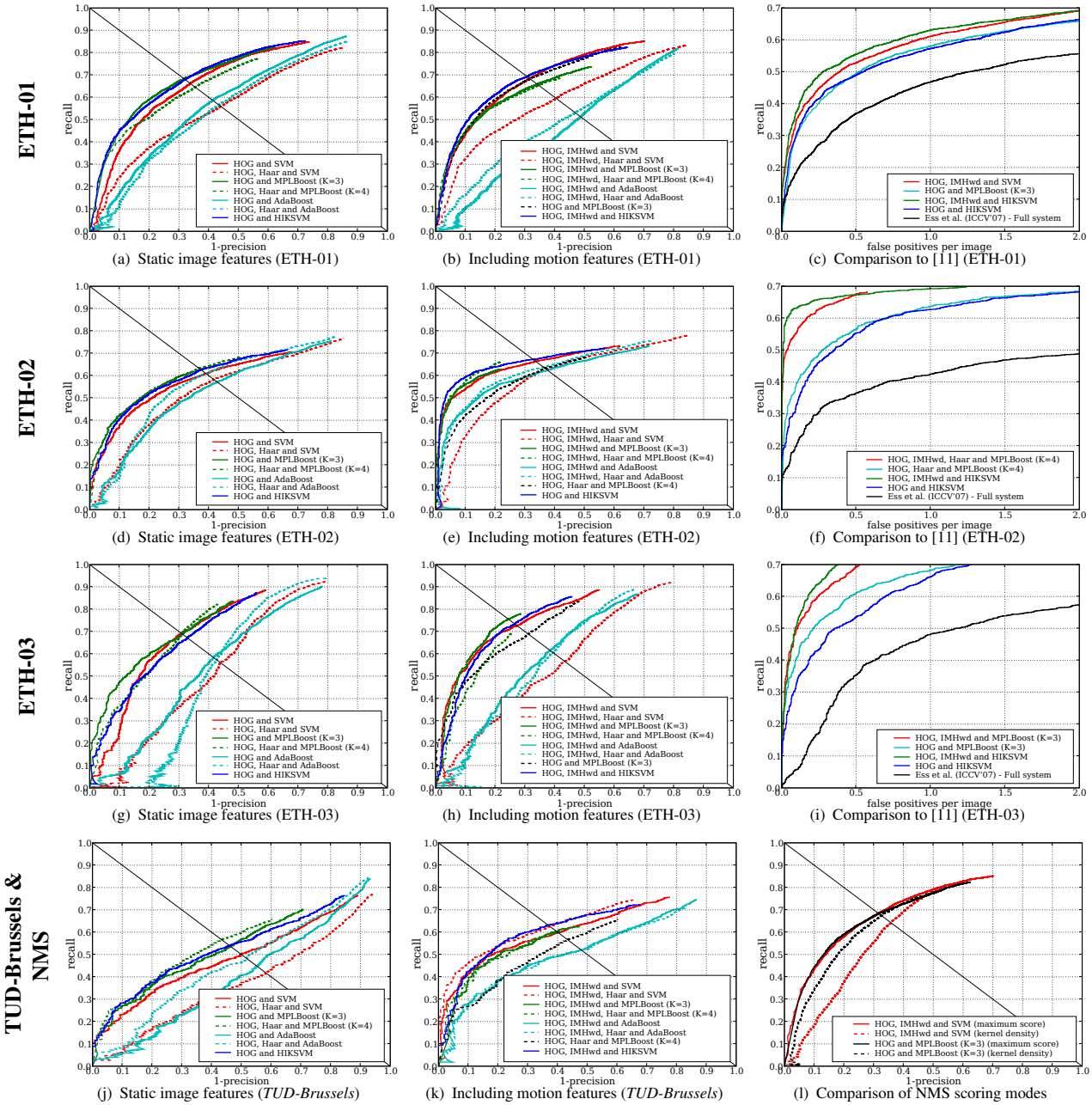
6. Results

Since we are interested in performance on a system level we refrain from evaluation in terms of FPPW but present plots in terms of recall and precision. This allows a better assessment of the detector as the entire detector pipeline is evaluated rather than the feature and classifier in isolation (cf. [8]). As a common reference point we will report the obtained recall at a precision of 90%. We also show plots of false positives per image to compare with previous work (i.e. [11]). We start the discussion of results with the static image descriptors and then discuss the benefit of adding motion features.

Results for the static features are given in the first column of Fig. 5. In combination with the HOG feature MPLBoost significantly outperforms AdaBoost on all tested sequences. In detail the improvement in recall at 90% precision is: 27.7% on ETH-01 (Fig. 5(a)), 24.4% on ETH-02 (Fig. 5(d)), 41.1% on ETH-03 (Fig. 5(g)) and 20.3% on *TUD-Brussels* (Fig. 5(j)). Also it can be observed that HOG features in combination with MPLBoost do better than HOG features in combination with a linear SVM on all four datasets. The gain in detail in recall at 90% precision is: 8.5% on ETH-01 (Fig. 5(a)), 4.9% on ETH-02 (Fig. 5(d)), 22.6% on ETH-03 (Fig. 5(g)) and 2.0% on *TUD-Brussels* (Fig. 5(j)). Compared to a SVM with histogram intersection kernel (HIKSVM) the results are divergent. While HIKSVM outperforms MPLBoost by 1.4% on *TUD-Brussels* (Fig. 5(j)) and by 0.4% on ETH-01 (Fig. 5(a)), on ETH-02 and ETH-03 MPLBoost performs better by 1.9%(Fig. 5(d)) and 12.9%(Fig. 5(g)) respectively.

Next we turn to the results with HOG and Haar features in combination with different classifiers. On the *TUD-Brussels* dataset (Fig. 5(j)) we observe an improvement of 0.3% at 90% precision for MPLBoost, while on equal error rate (EER) the improvement is 4.3%. For the ETH databases we yield equal or slightly worse results compared to the detectors with HOG features only (Fig. 5(a), (d), (g)). Closer inspection revealed minor image quality (cf. Fig. 7) with respect to colors and lighting on the ETH

[1] http://www.mis.informatik.tu-darmstadt.de



(a) Static image features (ETH-01) (b) Including motion features (ETH-01) (c) Comparison to [11] (ETH-01)

(d) Static image features (ETH-02) (e) Including motion features (ETH-02) (f) Comparison to [11] (ETH-02)

(g) Static image features (ETH-03) (h) Including motion features (ETH-03) (i) Comparison to [11] (ETH-03)

(j) Static image features (*TUD-Brussels*) (k) Including motion features (*TUD-Brussels*) (l) Comparison of NMS scoring modes

Figure 5: Results obtained with different combinations of features and classifiers. Rows (1)-(3) show results on ETH-Person [11], Row (4) details the results on the new *TUD-Brussels* onboard dataset. Note that first and second column show details on static and motion features in combination with different classifiers considering all detections larger than 48 pixels with recall and precision as metric. Column three compares our detector to the system of [11] (only pedestrians larger than 70 pixel are regarded, evaluation in FPPI) and shows a comparison of different non-maximum suppression approaches (Fig. 5(l)).

databases to be problematic, impeding a performance improvement (cf. Fig. 5(a), (d), (g)). Haar wavelets computed on color channels are not robust enough to these imaging conditions. Note however, that MPLBoost outperforms linear SVM, HIKSVM and AdaBoost for this feature combination showing its applicability for pedestrian detection. HIKSVM consistently obtained worse results with Haar features for static as well as for motion-enhanced detectors. Hence, these plots are omitted for better readability.

We continue to analyze the performance when IMHwd motion features in combination with HOG features are used for detection. The resulting plots are depicted in the second column of Fig. 5. For HIKSVM we observe a consistent improvement over the best static image detector. In detail the improvement at a precision of 90% precision is: 3.7% on ETH-01 (Fig. 5(b)), 16.9% on ETH-02 (Fig. 5(e)), 2.2% on ETH-03 (Fig. 5(h)) and 14.0% on *TUD-Brussels* (Fig. 5(k)). In contrast to [4] we can clearly show a significant perfor-

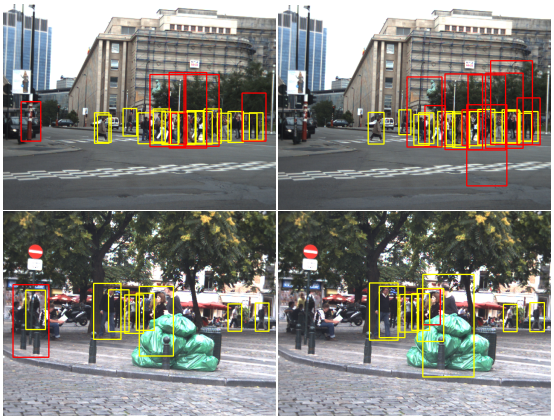


Figure 6: Sample detections on the *TUD-Brussels* onboard dataset at equal error rate for HOG, Haar, IMHwd and MPLBoost(K=4) (left column) and HOG, Haar, IMHwd and SVM (right column). True positives are yellow, false positives red.

mance gain using motion features. The difference in performance however depends on the dataset and the distribution of viewpoints in the test sets. More specifically motion is beneficial mostly for side views but also for 45-degrees views whereas front-back views profit less from the added motion features. This explains the lower performance gain for ETH-01 (Fig. 5(b)) and ETH-03 (Fig. 5(h)) which are dominated by front-back views. We also observe that linear SVMs perform about as good as MPLBoost for this feature combination, while HIKSVM does better than both except for ETH-03. Sample detections for MPLBoost and linear SVMs are shown in Fig. 6. Note that false detections differ between both classifiers. While MPLBoost tends to fire on high frequency background structure, SVMs tend to fire more often on pedestrian-like structures such as poles. We explain the similar overall performance by the fact that motion features allow a good linear separability in particular for side-views. This is consistent with our observation that MPLBoost mainly uses appearance features for the clusters firing on front-back views and more IMHwd features for clusters which fire on side views. Additionally, MPLBoost and SVMs again clearly outperform AdaBoost.

Combining IMHwd and HOG features additionally with Haar features yields similar results as for the static case with only little changes for MPLBoost. Interestingly linear SVMs obtain a better precision on *TUD-Brussels* for this combination, but loose performance on the ETH sequences as discussed for the static detectors. More sophisticated feature combination schemes (e.g. [29]) may allow to improve performance more consistently based on multiple features.

We have also analyzed the viewpoints different MPLBoost classifiers fire on. Fig. 8 depicts the two highest scoring detections on *TUD-Brussels* of the detector using HOG, IMHwd and Haar features for each of the four clusters. Clearly, two clusters predominantly fire on side and 45-degree side views while two clusters mostly detect pedestrians from front-back views.

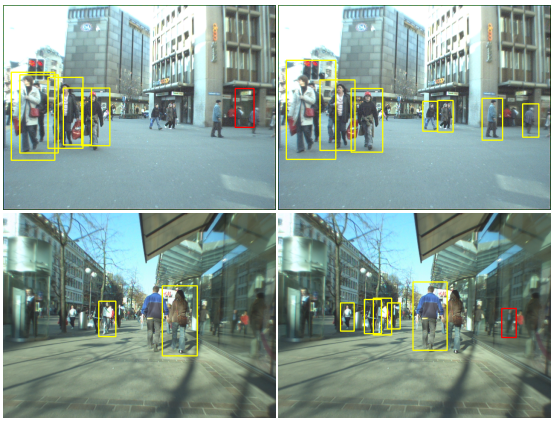


Figure 7: Sample detections at 0.5 FPPI (First column: System of [11], Second column: Our motion-enhanced detector). Rows 1 and 2 correspond to figures 5(f) and 5(i) respectively, however all detections (even those smaller than 70 pixels) are shown. Note the false positive in the lower right image is actually a reflection of a true pedestrian.

Figure 8: Sample detections for the different models learned by MPLBoost (K=4) using HOG, Haar, IMHwd. The models to the left respond more strongly to side/45-degree views, the models to the right to front/back views.

Finally, we compare our detector to the system of Ess et al. [11] (last column of Fig. 5). The original authors kindly provided us with their system's output in order to allow for a fair comparison based on the modified set of annotations. For each sequence we plot the best performance of a static image feature detector and of the best detector including motion features. We consistently outperform Ess et al. [11] on all three sequences without any refinement of detections by the estimation of a ground plane. This refinement could obviously be added and would allow for further improvement. At 0.5 false positives per image we improve recall compared to their system by: 18.6% on ETH-01 (Fig. 5(c)), 32.2% on ETH-02 (Fig. 5(f)) and 37.3% on ETH-03 (Fig. 5(i)). To keep this comparison fair, we only considered pedestrians larger than 70 pixels similar to the original evaluation setting. Also note that HIKSVM with motion features clearly outperforms MPLBoost, while both classifiers are almost on par when all pedestrians as small as 48 pixels are considered. We also outperform Zhang et al. [36] who report 64.3% recall at 1.5 FFPI even though their detector is trained on ETH-02 and ETH-03 whereas our detector is trained on an independent and more general multi-view training set. Sample detections of our detector as well as system results of [11] are shown in Fig. 7. Note that our detector can detect very small pedestrians and achieves better recall throughout all scales

by exploiting motion information.

7. Conclusion

In this work we tackled the challenging task of detecting pedestrians seen from multiple views from a moving car by using multiple appearance features as well as motion features. We show that HIKSVM and MPLBoost achieve superior performance to linear SVM-based detectors for static multi-viewpoint pedestrian detection. Moreover, both significantly outperform AdaBoost on this task. When additional motion features are used, HIKSVMs perform best while MPLBoost performs as good as linear SVMs but in any case better than AdaBoost. In general however, MPLBoost seemed to be the most robust classifier with respect to challenging lighting conditions while being computationally less expensive than SVMs.

Additionally, our careful design of the learning and testing procedures improves detection performance on a per-image measure substantially when the IMHwd motion features of Dalal et al. [6] are used which has been identified as an open problem in [4]. This improvement is observed for pedestrians at all scales but particularly for side views which are of high importance for automotive safety applications, since those pedestrians tend to cross the car's trajectory. Additionally, we show (contrary to [6]) that regularized flows [35], allow to improve detection performance. Adding additional Haar wavelets as features allowed to improve detection performance in some cases, but in general we observe that the feature is quite sensitive to varying cameras and lighting conditions.

For future work, we will further investigate ways of encoding motion information in an ego-motion invariant way. Also we are planning to work on the issue of partial occlusion, which is a prominent drawback of global object descriptors. Moreover, temporal integration by means of tracking over multiple frames will help to bridge missing detections while a more complete scene analysis featuring 3D scene understanding will help to prune false positive detections.

Acknowledgements This work has been funded, in part, by Toyota Motor Europe. Further we thank Christoph Zach and Thomas Pock for publicly releasing OFLib, Andreas Ess for his dataset and results and Piotr Dollár for helpful discussion.

References

[1] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *CVPR*, 2008.

[2] B. Babenko, P. Dollár, Z. Tu, and S. Belongie. Simultaneous learning and alignment: Multi-instance and multi-pose learning. In *ECCV Faces in Real-Life Images*, 2008.

[3] D. Comaniciu. An algorithm for data-driven bandwidth selection. *PAMI*, 25(2):281–288, 2003.

[4] N. Dalal. *Finding People in Images and Videos*. PhD thesis, Institut National Polytechnique de Grenoble, 2006.

[5] N. Dalal and B. Triggs. Histogram of oriented gradient for human detection. In *CVPR*, 2005.

[6] N. Dalal., B. Triggs., and C. Schmid. Human detection using oriented hist. of flow and appearance. In *ECCV*, 2006.

[7] P. Dollár, B. Babenko, S. Belongie, P. Perona, and Z. Tu. Multiple component learning for object detection. In *ECCV*, 2008.

[8] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian Detection: A Benchmark. In *CVPR*, 2009.

[9] M. Enzweiler and D. Gavrila. Monocular pedestrian detection: Survey and experiments. *PAMI*, to appear.

[10] M. Enzweiler, P. Kanter, and D. Gavrila. Monocular pedestrian recognition using motion parallax. In *IV*, 2008.

[11] A. Ess, B. Leibe, and L. V. Gool. Depth and appearance for mobile scene analysis. In *ICCV*, 2007.

[12] A. Ess, B. Leibe, K. Schindler, and L. van Gool. A mobile vision system for robust multi-person tracking. In *CVPR*, 2008.

[13] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, 2008.

[14] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. *The Annals of Statistics*, 38(2):337–374, 2000.

[15] D. M. Gavrila. A bayesian, exemplar-based approach to hierarchical shape matching. *PAMI*, 2007.

[16] D. M. Gavrila and S. Munder. Multi-cue pedestrian detection and tracking from a moving vehicle. *IJCV*, pages 41–59, 2007.

[17] J. J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston, MA, 1979.

[18] T.-K. Kim and R. Cipolla. MCBoost: Multiple classifier boosting for perceptual co-clustering of images and visual features. In *NIPS*, 2008.

[19] Z. Lin and L. S. Davis. A pose-invariant descriptor for human detection and segmentation. In *ECCV*, 2008.

[20] S. Maji, A. Berg, and J. Malik. Classification using intersection kernel SVMs is efficient. In *CVPR*, 2008.

[21] K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a prob. assembly of robust part det. In *ECCV*, 2004.

[22] C. Papageorgiou and T. Poggio. A trainable system for object detection. *IJCV*, 38(1):15–33, 2000.

[23] P. Sabzmeydani and G. Mori. Detecting pedestrians by learning shapelet features. In *CVPR*, 2007.

[24] E. Seemann, M. Fritz, and B. Schiele. Towards robust pedestrian detection in crowded image sequences. In *CVPR*, 2007.

[25] E. Seemann, B. Leibe, and B. Schiele. Multi-aspect detection of articulated objects. In *CVPR*, 2006.

[26] A. Shashua, Y. Gdalyahu, and G. Hayun. Pedestrian detection for driving assistance systems: single-frame classification and system level performance. In *IV*, 2004.

[27] D. Tran and D. Forsyth. Configuration estimates improve pedestrian finding. In *NIPS*, volume 20, 2008.

[28] O. Tuzel, F. M. Porikli, and P. Meer. Pedestrian det. via classification on riemannian manifolds. *PAMI*, 30(10):1713–1727, 2008.

[29] M. Varma and D. Ray. Learning the discriminative power-invariance trade-off. In *ICCV*, 2008.

[30] P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *CVPR*, 2003.

[31] P. A. Viola and M. J. Jones. Robust real-time face detection. *IJCV*, 57(2):137–154, 2004.

[32] C. Wojek and B. Schiele. A performance evaluation of single and multi-feature people detection. In *DAGM*, 2008.

[33] B. Wu and R. Nevatia. Cluster boosted tree classifier for multi-view, multi-pose object detection. In *ICCV*, 2007.

[34] B. Wu and R. Nevatia. Optimizing discrimination-efficiency tradeoff in integrating heterogeneous local features for object detection. In *CVPR*, 2008.

[35] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime $TV - L^1$ optical flow. In *DAGM*, 2007.

[36] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, 2008.