Disparity Statistics for Pedestrian Detection: Combining Appearance, Motion and Stereo

Stefan Walk[1], Konrad Schindler[1,2], and Bernt Schiele[1,3]

[1] Computer Science Department, TU Darmstadt
[2] Photogrammetry and Remote Sensing Group, ETH Zürich
[3] MPI Informatics, Saarbrücken

Abstract. Pedestrian detection is an important problem in computer vision due to its importance for applications such as visual surveillance, robotics, and automotive safety. This paper pushes the state-of-the-art of pedestrian detection in two ways. First, we propose a simple yet highly effective novel feature based on binocular disparity, outperforming previously proposed stereo features. Second, we show that the combination of different classifiers often improves performance even when classifiers are based on the same feature or feature combination. These two extensions result in significantly improved performance over the state-of-the-art on two challenging datasets.

1 Introduction

Pedestrian detection has been an active research area and significant progress has been reported over the years. An important lesson from previous research is that combining complementary cues is vital to improve state-of-the-art performance. Gavrila&Munder [1] and Ess et al. [2] combine appearance with stereo cues to detect pedestrians from moving vehicles, with the stereo components as modules for candidate generation and post-verification. Dalal et al. [3] and Wojek et al. [4] combine appearance and motion features in a sliding window framework, significantly improving performance. Despite impressive advances reported in the literature, state-of-the-art detectors seldom satisfy application requirements and leave ample room for improvement.

This paper advances pedestrian detection in two ways: first, we contribute a novel feature for pedestrian detection in stereo images, which we use in combination with standard appearance and motion cues. Despite its simplicity, the new feature yields significant improvements in detection performance. Second, we explore the potential of classifier combination for pedestrian detection. While the combination of different features [1,2,3,4] has been key to recent progress, the combination of different classifiers for the *same* feature has not been explored in the context of pedestrian detection to the best of our knowledge. The benefit of both contributions is analyzed and discussed in detail using two different recent pedestrian datasets.

K. Daniilidis, P. Maragos, N. Paragios (Eds.): ECCV 2010, Part VI, LNCS 6316, pp. 182–195, 2010.

2 Related Work

Early work on pedestrian detection by Papageorgiou and Poggio [5] and Viola et al. [6] used wavelet features. Viola et al. used a cascade of boosted classifiers, while Papageorgiou and Poggio use an SVM with a quadratic kernel. In [6] temporal information is included by taking intensity differences to adjacent frames (shifted to multiple directions).

Many techniques have been published since then, greatly improving performance – the datasets used to evaluate the early works are essentially solved now. Enzweiler&Gavrila [7] and Dollár et al. [8] recently published surveys on monocular pedestrian detection. For datasets with strong pose variations, such as sport scenes, articulated models like [9] provide best performance. For "standard" pedestrians, which are in an upright pose (as they are when standing or walking), monolithic global descriptors applied in a sliding window framework are still state of the art [8,4].

A pedestrian detector usually consists of candidate generation, followed by feature extraction for the candidate windows, classification of the feature vector, and then non-maximum suppression to prevent multiple detections on a single pedestrian. The most popular method of generating candidate windows is the sliding-window framework, where the scale/position space is sampled with fixed strides. Other work (e.g. [1]) utilizes some method of region-of-interest generation in order to reduce the number of candidate windows and filter out negative samples at an early stage.

The dominant appearance features are variants of the HOG descriptor [10,11,12] and different flavors of generalized Haar wavelets [6,8]. To encode motion information, [6] encodes wavelets on temporal intensity differences. [10,13] encode differences of optical flow into local histograms, similar to HOG.

Stereo information is commonly used in separate modules of pedestrian detection and tracking systems [1,2]. [1] use stereo information in two ways: first, they identify regions of interests in the disparity maps; after the pedestrian detection step, hypotheses are verified by cross correlation between the two images – if there is no object at the estimated disparity level, the correlation measure is low and the hypothesis is rejected. In the model used by [2], the disparity map is used for ground plane estimation, to ensure that detections have a reasonable size (using a prior on human height), and to verify that a pedestrian detection has consistent depth. Rohrbach et al. [14] use the depth field generated by a dense stereo matcher as input for the HOG descriptor to build HOG-like histograms on the depth gradient. Rapus et al. [15] utilize a low-resolution 3D camera (time-of-flight principle), and extract multiple features, including gradients and Fourier coefficients, from intensity and depth to detect pedestrians.

The most wide-spread classifiers are statistical learning techniques to separate positive and negative instances in the feature space. Popular algorithms are support vector machines [16,10,17,18] and variants of boosting [6,19,20,4]. Duin&Tax [21] perform experiments regarding the combination of multiple classifiers on a digit recognition dataset. They found that, while combining complementary features provides the largest gain, combining different classifiers trained

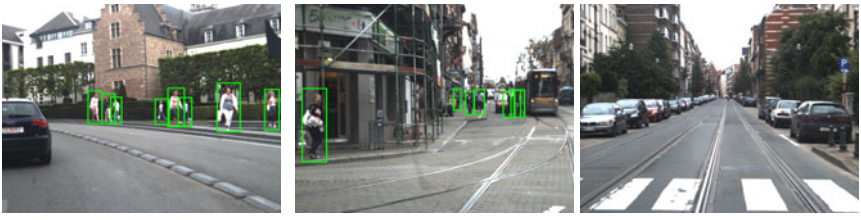


Fig. 1. Sample images from the new auxiliary training set. The last image is from the negative set.

on the *same* features can also help. We show that this also holds in the pedestrian detection setting.

3 Datasets

We use two different challenging datasets for our tests. Both databases have been recorded from a moving car in scenarios with many pedestrians: *ETH-Loewenplatz* [2,22,23] and *TUD-Brussels* [4]. Since we want to build a detector that utilizes both motion and stereo information, we are constrained in our choice of training data. We use two datasets: *TUD-MotionPairs* [4] and a new, auxiliary dataset to train the stereo-based component of our detector.

ETH-Loewenplatz. Our first test set consists of a video sequence of 800 consecutive stereo frames taken from a moving car, with annotations every 4 frames. In total it contains 2631 annotations, however we scan only for pedestrians bigger or equal to 48 pixels in size, which leaves us with 1431 annotations for evaluation.

TUD-Brussels. The second test set has 508 annotated frames recorded from a moving car. It originally had 1326 pedestrian annotations, but there were some small pedestrians missing. We supplemented those, resulting in a total of 1498 pedestrian annotations, with 1235 of them at least 48 pixels high. The dataset allows for optic flow estimation, but there is no published stereo information. However the authors kindly provided us with stereo pairs for this dataset.

TUD-MotionPairs. This dataset is used for training and contains 1776 pedestrian annotations in 1092 images, including the following frame for each annotated frame (to compute optical flow). The images are recorded in a pedestrian zone from a handheld camera, with pedestrians seen from multiple viewpoints. 192 image pairs without pedestrians, partly taken from a handheld camera and partly from a moving car, serve as negative set.

Auxiliary Training set. As *TUD-MotionPairs* does not contain stereo information, we have created a new dataset to train our stereo classifiers. The new dataset contains 2570 annotations in 824 frames in the positive set, with stereo and motion information available. However, most of the pedestrians in this set are small (2033 of them are smaller than the detection window,resulting in suboptimal quality). The negative set contains 321 frames, again with motion and

stereo information. The images have a resolution of 640x480 pixels and were recorded from a moving car. Sample images are shown in Figure 1.

4 Baseline Features and Classifiers

The set of features and classifiers we use as baselines includes HOG [10] and HOF [3] as features, and SVMs and MPLBoost [24,25] as classifiers. The same features and classifiers were used recently in [4].

HOG. Dalal&Triggs proposed using histograms of oriented gradients in [10]. In HOG, every pixel votes for its gradient orientation into a grid of histograms using trilinear (spatial and orientation) interpolation. Local normalization is employed to make the feature robust against changes in illumination. Interpolation and histogramming makes the feature robust with regard to small changes in pose.

HOF. Histograms of Flow were introduced in [3] to encode motion information from optical flow. We use a reduced variant of the original IMHcd scheme with 2x2 blocks. Our version is on par with the original HOF in terms of performance. Flow fields are estimated with the publicly available optical flow implementation by Werlberger et al. [26].

SVM. Support Vector Machines are currently the standard for binary classification in computer vision. Linear SVMs learn a hyperplane that optimally separates negative and positive samples in high-dimensional feature space. Kernel SVMs are also possible, however their high computation time makes them intractable for sliding-window detection with high-dimensional feature vectors. An exception to this are histogram intersection kernels (HIKSVMs), for which an approximation can be evaluated in constant time [27].

MPLBoost. MPLBoost is an extension to AdaBoost[6], where K strong classifiers are learnt jointly, with each strong classifier focusing on a subset of the feature space. The final confidence is the maximum over the K classifiers, so only one of them needs to correctly identify a positive sample. Unless noted otherwise, we use $K = 4$ strong classifiers.

For training, negative samples are first randomly drawn from the negative training set to create an initial classifier. With this classifier the negative training images are scanned for *hard negatives* that get misclassified. These are added to the negative set and the classifier is retrained. We repeat this *bootstrapping* step twice to ensure that the result is minimally influenced by the random choice of the initial negative set.

The feature/classifier *components* we are using throughout the paper were previously studied in our paper [4]. Due to optimizations and changes in training procedure, there are some differences. Figure 5(b) compares the implementations. The three dotted lines compare the "old" HOG-detector (red dotted line) and HOG+Haar-detector (green dotted line) with our HOG-implementation (blue dotted line). Similarly the "new" HOG+HOF-feature (blue solid line) performs similar to or better than the previous HOG+HOF+Haar-feature (green

(a) LinSVM (b) MPLBoost (c) LinSVM+MPLBoost

(d) LinSVM (e) MPLBoost (f) LinSVM+MPLBoost

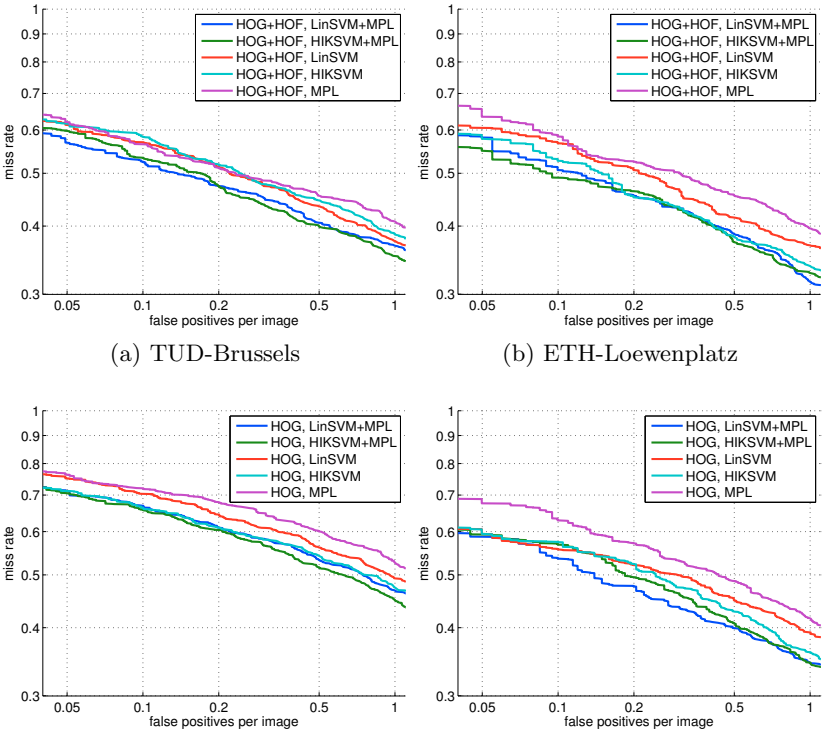
Fig. 2. MPLBoost and SVMs perform well but tend to have different false positives (a,b,d,e – red boxes correspond to false positives). By combining both classifiers the false positive rate can be reduced (c,f).

solid line) and HOG+HOF-feature (red solid line). Note that we do not use Haar features as in [4] we found them not to be beneficial in all cases.

5 Combination of Classifiers

It is well-established that utilizing a combination of complementary cues significantly boosts detection performance. E.g. Gavrila et al. [1] use shape, texture, and stereo cues to build a detection system while Wojek et al. [4] use multiple features (including appearance and motion information) to boost detection performance. Rohrbach et al. [14] fuse classifiers separately trained on intensity and depth. In these cases, the complementarity of the classifiers results from the cues being from different sources (such as stereo and motion information) or from the sources being encoded into different features. However, those are not the only sources of complementary information.

In [4], we noticed that MPLBoost and SVMs, while both giving good performance, tend to produce different false positives using the *same* feature set. For true positives, different classifiers are likely to give a positive answer, while for false positives the classifiers do not necessarily agree. See figure 2 for examples where LinSVM and MPLBoost (for the feature set HOG+HOF) produce different false positives (2(a,d) and (b,e) respectively). This gives a strong hint that by combining SVM and an MPLBoost classifiers, one can reduce the false positive rate. See figure 2(c,f) where such a combination eliminated false positives. This combination is described in the following.



(a) TUD-Brussels (b) ETH-Loewenplatz

(c) TUD-Brussels (d) ETH-Loewenplatz

Fig. 3. Results using classifier combination on *TUD-Brussels* and *ETH-Loewenplatz* with HOG+HOF and HOG alone as features. The single-component detectors are on par with the best published ones on *TUD-Brussels* from [4] (figure 5(b)), combining multiple classifiers yields a noticeable improvement.

Starting from the above observation, this paper explores the possibility to combine classifiers not only for different features but also for the same feature. The combination of classifiers for the same features is especially interesting as it is "cheap": Feature extraction is computationally expensive and often the bottleneck in today's systems. When combining classifiers on the same feature space, the feature vector has to be computed only once.

Classifiers are already combined at the training stage, which influences the bootstrapping phase: a window gets registered as a hard sample if it's hard for the *combined* classifier, enabling the classifiers to focus on data that is problematic for the final detector. This results in slightly better performance than training them separately. The combinations that we study in this section are linear SVM+MPLBoost and HIKSVM+MPLBoost, both trained on the same feature space, HOG+HOF. Combining a linear SVM with an HIKSVM did not show any improvement and thus is not reported here.

As noted before, one can expect classifier combination to improve classification if the combined classifiers have complementary characteristics. A (confidence-rated) classifier is a mapping from the feature vector space to a score. For an imperfect (but better than chance) classifier, the *probability density functions* (*pdf*s) of the positive and negative classes are overlapping. Under the reasonable assumption that the mean of the positive pdf is higher than the mean of the negative pdf, we can – without loss of generality – rescale the mapping so that the means of the positive and negative pdfs are at +1 and -1, respectively. Classification errors (caused by the overlap of the pdfs) can then be expected to decrease when the variance decreases. The variance σ^2_{x+y} of a weighted sum $\alpha x + \beta y$ of classifiers x and y ($\alpha + \beta = 1$) for a given class is $\sigma^2_{x+y} = \alpha^2\sigma^2_x + 2\alpha\beta\sigma^2_{xy} + \beta^2\sigma^2_y$ with σ^2_{xy} being the covariance. If this is lower than σ^2_x and σ^2_y, the combination can be expected to be beneficial.

Results are shown in figure 3 for the two test sets. For comparison, results for individual classifiers are shown as well. For *TUD-Brussels* and the feature combination HOG+HOF (Fig. 3(a)) the two combined classifiers (blue and green curves) clearly improve performance over the individual classifiers (red, cyan, violet curves). For *ETH-Loewenplatz* (Fig. 3(b)) the improvement of the combinations (blue, green curves) over the individual classifiers is also visible.

At 0.1 false positives per image the best combined classifier for HOG+HOF (Linear SVM + MPLBoost) has 4.2% more recall than the best single component classifier on *TUD-Brussels*, and 3.7% more recall on *ETH-Loewenplatz*. Using only HOG as feature, a smaller improvement can be observed over the best individual classifier for *TUD-Brussels* (see Fig. 3(c)) while on *ETH-Loewenplatz* the improvement is substantial at higher false positive rates: 5% improvement at 0.2 fppi (see Fig. 3(d)).

The results reported so far have been obtained by averaging classifier scores as a confidence measure of the combined classifier. This gives both components equal weight. To see if performance improves when the weights are learned instead, we employ a linear SVM as a top-level classifier with the lower level classifier confidences as inputs. Here, 5-fold cross validation on the training set is used to train the top-level classifier without overfitting: we train on 80% of the training data and evaluate the component classifiers on the remaining 20%, with the cross-validation scores being the feature vectors for the top-level classifier. The final component classifiers are then trained using the whole training set. However, there is no significant improvement over equal weights, which is not surprising, as the classifiers work about equally well. As training takes significantly longer with this approach (≈ 6 times), we do not use it in the rest of the paper. In the context of combining SVM kernels, [28] found that if the kernels are comparable in performance, averaging works well, while learning the combination is important when there are uninformative components, which agrees with our experience.

6 Utilizing Stereo Information

In the previous section, we showed that different classifiers on the same feature set can be combined to form a better classifier. However, the combination of

different kinds of features promises a greater possible gain in information and consequently also in performance. One prominent source of information that is complementary to appearance and motion is binocular vision. Using a stereo image pair, we can extract disparity and depth information, which turns out to improve performance considerably.

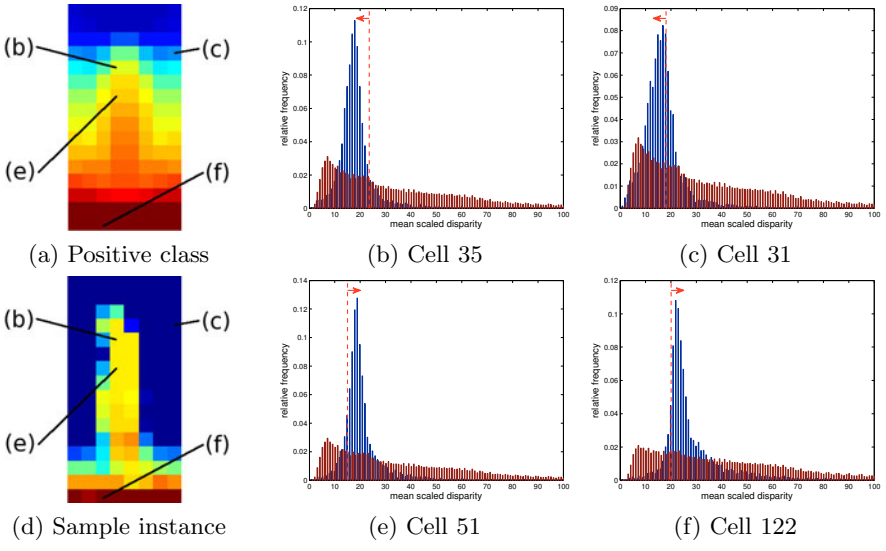
HOS-feature. As a first stereo feature, we use a HOG/HOF-like feature. In [14], Rohrbach et al. computed the HOG descriptor on the depth field, which is inversely proportional to the disparity field, because its gradients are – in theory – invariant to the position of the pedestrian in the world. The gradients in the disparity image are not invariant (they are nonlinearly scaled). However, HOG is designed to provide invariance against scale changes in "intensity" (in this case, disparity). This becomes problematic only for very small disparities, where the nonlinearity are noticeable. On the other hand, using the depth also has its problems: since $Z \propto \frac{1}{d}$, small errors in disparity result in large errors of the depth map; moreover, pixels with disparity 0 have infinite depth and require special handling when building the descriptor, otherwise a single pixel can cause an infinite entry in the histogram. If we directly compute gradients on the disparity map, no special handling is required.

We have experimented with standard HOG descriptors (encoding small-range gradients in depth or disparity) and also with a variant of HOF on the disparity field, where we treat the disparity field like a vector field with the disparity as the x-coordinate and the y coordinate set to 0. The only relevant orientation bins here are the left and right bins: For every pixel, it is encoded if the pixels that are 8 pixels (in the L_∞ norm) away in horizontal, vertical or diagonal direction have a smaller or greater distance to the camera, weighted by the difference. This scheme in principle encodes less information than the full HOG descriptor, however stereo algorithms are not that accurate on a small scale, so long-range differences are more stable. Experimentally we did not observe any significant difference between the performance of this encoding and the encoding proposed by [14]. Therefore, in the following we use the HOF-like descriptor on the disparity field (termed HOS in the following) with a linear SVM as the classifier.

Disparity statistics (DispStat) feature. The disparity field has an interesting invariant property: in the pinhole camera model, the disparity d at a given point is $d = \frac{fB}{Z} \propto \frac{1}{Z}$ with the focal length f, the baseline B, and the depth Z. The observed height h of an object of height H is $h = \frac{fH}{Z} \propto \frac{1}{Z}$

This means that the ratio of disparity and observed height is inversely proportional to the 3D object height; for objects of fixed size that ratio is constant. The heights of pedestrians are not identical, but very similar for most pedestrians. We can therefore, during sliding window search, divide the disparity values by the appropriate scale level determined by the layer of the image pyramid – e.g. for a reference height of 96 pixels and a scaled detection window of 64 pixels, disparities will be multiplied by 1.5. The scaled disparities of positive (pedestrian) samples will then follow a narrow distribution.[1]

[1] If the camera setup is different between the training and test images, the ratio between height and disparity has to be adapted accordingly.



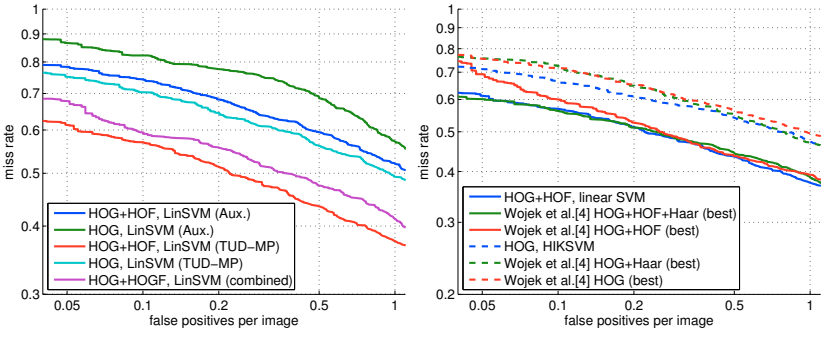
(a) Positive class (b) Cell 35 (c) Cell 31

(d) Sample instance (e) Cell 51 (f) Cell 122

Fig. 4. Visualization of the Disparity Statistics feature. (a) is a color map of the median of the feature values over all positive samples (symmetric because training images get mirrored), (d) of an example training instance. Warmer color corresponds to bigger disparity/nearer points. Clearly, the feature is able to encode information like the pedestrian standing on the ground plane and the area around the upper body being more likely to be behind the pedestrian.

This observation enables us to design a very simple and surprisingly effective feature. We divide the detection window into 8x8 pixel cells (the same as the HOG cell size, for computational efficiency). For each cell, the mean of the scaled disparities is computed. The concatenation of all 8×16 mean values from the 64×128 pixel window is the feature vector. For this feature, we use MPLBoost as classifier with $K = 2$ (more clusters did not help) and 100 boosting rounds.

Figure 4 visualizes the feature. In figure 4(a), the cell-wise median of all positive training samples is shown, 4(d) shows one particular positive training sample. One can immediately see different pieces of information captured by the new descriptor: the surrounding background is typically further away than the person, and the person usually stands on an approximately horizontal ground plane. In figure 4(b,c,e,f) statistics from example cells are shown along with weak classifier boundaries from the MPLBoost classifier. Displayed are the relative per-class frequencies of the disparity values. For the positive class, all 5140 training instances (including mirrored samples) are plotted, to plot the negative class 5 images were sampled densely, with the same parameters as in the sliding window search, resulting in 721900 samples (training of course uses all 321 images of the negative set). The dashed red line shows the weak classifier threshold, with arrows to the right signaling a lower bound, and arrows to the left an upper bound. Note that they are *weak* classifiers – they are only required to work better



(a) Different training sets (b) Comparison with [4]

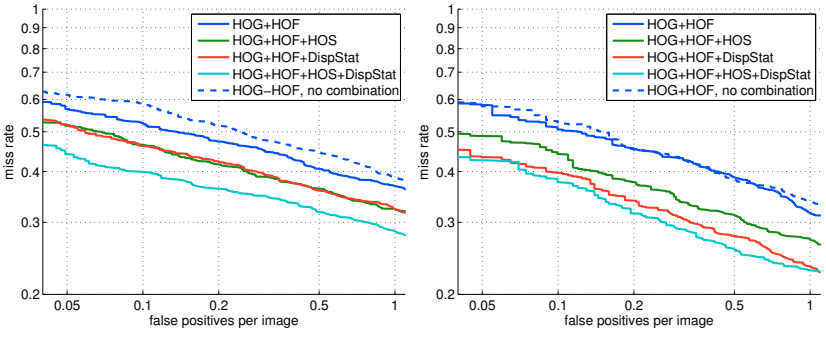
Fig. 5. (a) *TUD-MotionPairs (TUD-MP)* is a better training set than the auxiliary training set *(Aux.)* for appearance and motion information, however it contains no stereo information. Even combining *TUD-MotionPairs* with the auxiliary training set results in inferior performance for our detector when using appearance and motion as cues. In (b), one can see that our components are as least as good as the ones shown in [4].

than chance, so it does not matter if they miss-classify a portion of the training set. Even though the distributions overlap, making learning a non-trivial task, it is obvious that the class distributions are different and something can be learned from this data.

In figure 4(b) and (e), the disparity range for the upper body is evaluated by the weak classifiers, meaning the classifiers learn the size of a pedestrian (since the observed height is fixed – the height of the bounding box under evaluation – the *scaled* disparity relates inversely proportional to a height in 3D).

In 4(c), one weak classifier learned that the area to the right of the pedestrian usually is not closer to the camera than the pedestrian itself (note that the maximum of the distribution is at a lower disparity than the maxima of the distributions for (b) and (e)). However, the distribution here is not as narrow, because it is not uncommon that pedestrians stand next to other objects in a similar depth range. Figure 4(f) visualizes a weak classifier testing that the pedestrian stands on a ground plane, meaning that the cell under the pedestrian is closer to the camera than the pedestrian itself. Note that learning the pedestrian size and the ground plane assumption is completely data-driven.

Combining classifiers for different cues. Finding a dataset to train a detector using depth, motion, and appearance is not trivial: The public designated training sets we are aware of don't have both stereo and motion information available. Our new training set, the *auxiliary training set*, has this, however it is not as good as *TUD-MotionPairs* for appearance and motion, as can be seen in figure 5(a). The detector using HOG+HOF with a linear SVM has over 15% less recall when trained on this set (compare blue and red curves). Even joining the datasets for training results in inferior performance (violet curve).



(a) TUD-Brussels (b) ETH-Loewenplatz

Fig. 6. Results using stereo information on *TUD-Brussels* and *ETH-Loewenplatz*

To address this problem, we train different components on different datasets, and combine the components with an additional classifier stacked on top, which operates on the outputs of the components. In this section, we take the best combined classifier for appearance and motion (linear SVM + MPLBoost on HOG+HOF trained on *TUD-MotionPairs*) as one component. To combine the appearance/motion with the stereo components, a linear SVM is trained on top of the component outputs to provide the final score. The top-level SVM and the stereo-based classifiers are trained jointly using 5-fold cross validation on the auxiliary training set. To generate dense disparity maps, we used the algorithm of Zach et al. [29].

Results. As can be seen in figure 6, our new feature/classifier combination improves performance significantly. Best results from figure 3 are reproduced for reference: the dotted blue lines are the best performing individual classifier (HOG+HOF); the solid blue lines are the best performing combined classifier. On *TUD-Brussels* (Fig. 6(a)), the new disparity statistics feature combined with our HOG+HOF-classifier (red curve) performs as good as the HOS feature combined with HOG+HOF (green curve), resulting in an improvement of 6.4% recall at 0.1 fppi over the detector using HOG+HOF alone (blue curve). Combining both stereo features (cyan curve), the improvement is 12.6% over the HOG+HOF detector (solid blue curve), and more than 18% better than HOG+HOF with a linear SVM (dashed blue curve), which in turn is slightly better than the best reported result in the literature for this dataset [4] (c.f. figure 5(b)). The improvements are consistent over a wide range of false positive rates.

On *ETH-Loewenplatz* (Fig. 6(b)), adding HOS (green curve) results in an improvement of 6.6% at 0.1 fppi over HOG+HOF (blue curve). Using DispStat in addition to HOG+HOF (red curve) results in a higher improvement than HOS resulting in 11% improvement at 0.1 fppi. Further combining DispStat with HOS (cyan curve) in addition to HOG+HOF improves recall by another 2%. These results clearly show that DispStat is the stronger feature than HOS

Fig. 7. Sample results using stereo information

for this dataset. Compared to the best single-classifier detector with HOG+HOF as features (dashed blue), the overall improvement is 15%.

Comparing to state-of-the-art performance by [23] (they use a complete system integrating stereo, ground-plane estimation and tracking) our combined detector outperforms their best performance. In their evaluation scheme (pedestrians larger than 60 pixels) we outperform their system by about 5% at 0.1 fppi. This clearly underlines the power of the contributions of this paper to improve the state-of-the-art in pedestrian detection.

Figure 7 show sample results using stereo information. In every pair, the upper image shows the HOG+HOF detector with HIKSVM+MPLBoost, the lower the full detector including HOS and the DispStat feature. Both detectors are shown at the point where they reach 70% recall, so differences are to be seen in the amount of false positives. The stereo features are especially good at eliminating false positives at the wrong scale, or not standing on the ground plane. “Typical”

false positives, like car wheels (top left) and body parts (top right, bottom left) are easily filtered out, as well as detections having moving pedestrian as "legs" (bottom left). False positives on objects that are similar in 3d to a pedestrian are still an issue, for example the trash can with a traffic sign in the middle image in the lower row. Since the disparity field suffers from artifacts and missing information at the image border, some pedestrians (e.g. at the left border of the upper left image pair) are missed, however it detects others that the monocular detector misses (as both are tuned to get 70% recall). Also note that in the lower left image the HOG+HOF detector overestimates the size of the pedestrian at the right image border, causing a false positive and a missed detection, while the detector using stereo features correctly estimates the size and position of the pedestrian.

7 Conclusion

This paper consists of two contributions for pedestrian detection. First, we show that combining different classifiers trained on the same feature space can perform better than using a single classifier. Second, we introduce a new feature, called DispStat, for stereo, enabling the classifier to learn scene geometry information (like pedestrian height and the ground plane assumption) completely data-driven, without any prior knowledge. Combining those two contributions, we outperform the best published result on *TUD-Brussels* by over 12%, in combination with an adaptation of HOG for disparity fields similar to [14], this increases to over 18%. We verified these results on a second challenging dataset, *ETH-Loewenplatz*, where the performance of DispStat is even better, outperforming the HOS feature.

Acknowledgments. The authors thank Christopher Zach for providing his implementation of [29] and Christian Wojek for code and valuable discussion.

References

1. Gavrila, D.M., Munder, S.: Multi-cue pedestrian detection and tracking from a moving vehicle. IJCV 73, 41–59 (2007)
2. Ess, A., Leibe, B., Schindler, K., van Gool, L.: A mobile vision system for robust multi-person tracking. In: CVPR (2008)
3. 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)
4. Wojek, C., Walk, S., Schiele, B.: Multi-cue onboard pedestrian detection. In: CVPR (2009)
5. Papageorgiou, C., Poggio, T.: A trainable system for object detection. IJCV 38, 15–33 (2000)
6. Viola, P., Jones, M., Snow, D.: Detecting pedestrians using patterns of motion and appearance. In: ICCV (2003)
7. Enzweiler, M., Gavrila, D.M.: Monocular pedestrian detection: Survey and experiments. In: PAMI (2009)

8. Dollár, P., Wojek, C., Schiele, B., Perona, P.: Pedestrian detection: A benchmark. In: CVPR (2009)
9. Andriluka, M., Roth, S., Schiele, B.: Pictorial structures revisited: People detection and articulated pose estimation. In: CVPR (2009)
10. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR (2005)
11. Felzenszwalb, P., McAllester, D., Ramanan, D.: A discriminatively trained, multiscale, deformable part model. In: CVPR (2008)
12. Wang, X., Han, T.X., Yan, S.: A HOG-LBP human detector with partial occlusion handling. In: ICCV (2009)
13. Dalal, N.: Finding People in Images and Videos. PhD thesis, Institut National Polytechnique de Grenoble (2006)
14. Rohrbach, M., Enzweiler, M., Gavrila, D.M.: High-level fusion of depth and intensity for pedestrian classification. In: Denzler, J., Notni, G., Süße, H. (eds.) DAGM 2009. LNCS, vol. 5748, pp. 101–110. Springer, Heidelberg (2009)
15. Rapus, M., Munder, S., Baratoff, G., Denzler, J.: Pedestrian recognition using combined low-resolution depth and intensity images. In: IEEE Intelligent Vehicles Symposium (2008)
16. Shashua, A., Gdalyahu, Y., Hayun, G.: Pedestrian detection for driving assistance systems: Single-frame classification and system level performance. In: IVS (2004)
17. Sabzmeydani, P., Mori, G.: Detecting pedestrians by learning shapelet features. In: CVPR (2007)
18. Lin, Z., Davis, L.S.: A pose-invariant descriptor for human detection and segmentation. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 423–436. Springer, Heidelberg (2008)
19. Zhu, Q., Yeh, M.C., Cheng, K.T., Avidan, S.: Fast human detection using a cascade of histograms of oriented gradients. In: CVPR (2006)
20. Wu, B., Nevatia, R.: Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet part detectors. IJCV 75, 247–266 (2007)
21. Duin, R.P.W., Tax, D.M.J.: Experiments with classifier combining rules. In: Kittler, J., Roli, F. (eds.) MCS 2000. LNCS, vol. 1857, p. 16. Springer, Heidelberg (2000)
22. Ess, A., Leibe, B., Schindler, K., van Gool, L.: Moving obstacle detection in highly dynamic scenes. In: ICRA (2009)
23. Ess, A., Leibe, B., Schindler, K., Van Gool, L.: Robust multi-person tracking from a mobile platform. PAMI 31(10), 1831–1846 (2009)
24. Babenko, B., Dollár, P., Tu, Z., Belongie, S.: Simultaneous learning and alignment: Multi-instance and multi-pose learning. In: ECCV workshop on Faces in Real-Life Images (2008)
25. Kim, T.K., Cipolla, R.: MCBoost: Multiple classifier boosting for perceptual coclustering of images and visual features. In: NIPS (2008)
26. Werlberger, M., Trobin, W., Pock, T., Wedel, A., Cremers, D., Bischof, H.: Anisotropic Huber-L1 optical flow. In: BMVC (2009)
27. Maji, S., Berg, A.C., Malik, J.: Classification using intersection kernel support vector machines is efficient. In: CVPR (2008)
28. Gehler, P.V., Nowozin, S.: On feature combination for multiclass object classification. In: ICCV (2009)
29. Zach, C., Frahm, J.M., Niethammer, M.: Continuous maximal flows and Wulff shapes: Application to MRFs. In: CVPR (2009)