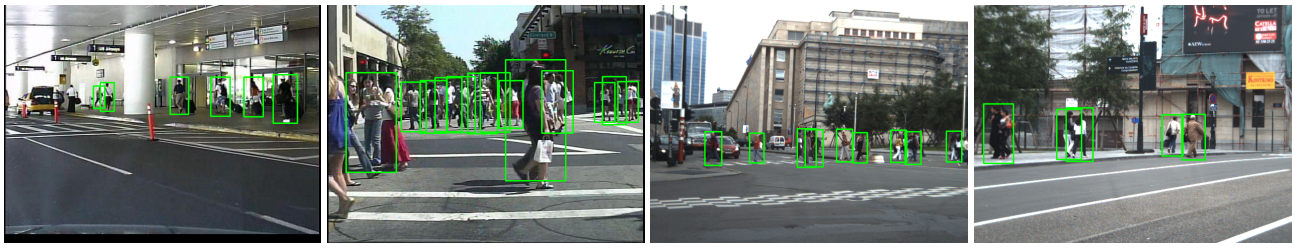
New Features and Insights for Pedestrian Detection

Stefan Walk[1] Nikodem Majer[1] Konrad Schindler[1] Bernt Schiele[1,2]

[1] Computer Science Department, TU Darmstadt [2] MPI Informatics, Saarbrücken



Abstract

Despite impressive progress in people detection the performance on challenging datasets like Caltech Pedestrians *or* TUD-Brussels *is still unsatisfactory. In this work we show that motion features derived from optic flow yield substantial improvements on image sequences, if implemented correctly—even in the case of low-quality video and consequently degraded flow fields. Furthermore, we introduce a new feature,* self-similarity *on color channels, which consistently improves detection performance both for static images and for video sequences, across different datasets. In combination with HOG, these two features outperform the state-of-the-art by up to 20%. Finally, we report two insights concerning detector evaluations, which apply to classifier-based object detection in general. First, we show that a commonly under-estimated detail of training, the number of bootstrapping rounds, has a drastic influence on the relative (and absolute) performance of different feature/classifier combinations. Second, we discuss important intricacies of detector evaluation and show that current benchmarking protocols lack crucial details, which can distort evaluations.*

1. Introduction

Pedestrian detection has been a focus of recent research due to its importance for practical applications such as automotive safety [11, 8] and visual surveillance [23]. The spectacular progress that has been made in detecting pedestrians (i.e. humans in an upright position) is maybe best illustrated by the increasing difficulty of datasets used for benchmarking. The first [16] and second [3] generation of pedestrian databases are essentially saturated, and have been replaced by new more challenging datasets [7, 27, 6]. These recent efforts to record data of realistic complexity have also shown that there is still a gap between what is possible with pedestrian detectors and what would be required for many applications: in [6] the detection rate of the best methods is still $< 60\%$ for one false positive detection per image, even for fully visible people.

The present paper makes three main contributions. First, we introduce a *new feature based on self-similarity of low-level features*, in particular color histograms from different sub-regions within the detector window. This feature, termed CSS, captures pairwise statistics of spatially localized color distributions, thus being independent of the actual color of a specific example. The self-similarity allows to represent properties like "the color distributions on the left and right shoulder usually exhibit high similarity", independent of the actual color distribution, which may vary from person to person depending on their clothing. Adding CSS significantly improves state-of-the-art classification performance for both static images and image sequences. The new feature is particularly powerful for static images, and hence also valuable for applications such as content-based image retrieval. It also yields a consistent improvement on images sequences, in combination with optic flow.

The second main contribution is to establish a standard what pedestrian detection with a global descriptor can achieve at present, including a number of recent advances which we believe should be part of the "best practice", but have not yet been included in systematic evaluations. In evaluations on the two most challenging benchmarks currently available—*Caltech Pedestrians* [6] and *TUD-Brussels* [27]—our detector achieves the *best results to date, outperforming published results by 5 to 20%*.

Our third main contribution are two important insights that apply not only to pedestrian detection, but more generally to classifier-based object detection. The first insight

978-1-4244-6985-7/10/$26.00 ©2010 IEEE

is concerned with the fact that—for all classifiers—correct iterative bootstrapping is crucial. According to our experiments, the number of bootstrapping iterations is more important than the number of initial negative training samples, and too few iterations can even lead to incorrect conclusions about the performance of different feature sets. As a second insight, we point out some issues w.r.t. benchmarking and evaluation procedures, for which we found the existing standards to be insufficient.

Related Work Since the pioneering works [16, 23], many improvements have been proposed, constantly pushing performance further. An important insight of past research is that powerful articulated models, which can adapt to variations in body pose, only help in the presence of strong pose variations, such as in sport scenes [1]. On the contrary, the most sucessful model to date for "normal" pedestrians, who are usually standing or walking upright, is still a monolithic global descriptor for the entire search window.

With such a model, there are three main steps which can be varied to gain performance: feature extraction, classification, and non-maxima suppression. The most common *features* extracted from the raw image data are variants of the HOG framework, i.e. local histograms of gradients and (relative) optic flow [3, 4, 10, 24, 27], and different flavors of generalized Haar wavelets, e.g. [6, 23]. All competitive *classifiers* we know of employ statistical learning techniques to learn the mapping from features to scores (indicating the likelihood of a pedestrian being present)—usually either support vector machines [3, 13, 17, 19, 27] or some variant of boosting [23, 27, 28, 30]. An important detail during learning is iterative bootstrapping to improve the decision boundary with difficult negative examples—see Sec. 5.

We evaluate new features and new combinations of features and classifiers on the *Caltech Pedestrians* dataset. In that sense the evaluation can be seen as an extension of [6]: we also discuss optic flow, the recently proposed HOG-LBP feature [24], and our new color self-similarity.

Instance-specific color information was recently used in the form of implicit local segmentation features [15], encoding gradients of distances w.r.t. two *local* color distribution models ("foreground" and "background"). Only few authors have advocated the use of self-similarity as a feature. Most notably [20] encodes the self-similarity of raw image patches in a log-polar binned descriptor. They demonstrate superior performance over gradient features in a template-matching task. In [12] the authors propose self-similarity descriptors over feature time series for human action recognition, observing good viewpoint invariance of the descriptor. In a different context, [21] proposed a representation similar to ours, where color similarity is computed at the pixel level, assuming a Gaussian conditional color distribution. To the best of our knowledge, self-similarity has not yet been used as a feature for people detection.

2. Datasets

For our evaluation, we focus on two databases, *Caltech Pedestrians* [6] and *TUD-Brussels* [27], which are arguably the most realistic and most challenging available datasets, *Caltech* also being by far the largest. *INRIAPerson* is still a popular dataset, but it contains no motion, and consists mainly of large upright pedestrians with little occlusion.

Caltech Pedestrians contains a vast number of pedestrians—the training set consists of 192k ($= 192000$) pedestrian bounding boxes and the testing set of 155k bounding boxes, with 2300 unique pedestrians on 350k frames. Evaluation happens on every 30th frame. The dataset is difficult for several reasons. On the one hand it contains many small pedestrians and has realistic occlusion frequency. On the other hand the image quality is lacking, including blur as well as visible JPEG artifacts (blocks, ringing, quantization) which induce phantom gradients. These hurt the extraction of both gradient and flow features. For our evaluation we use the model trained on *TUD-MotionPairs*[27] (see below), and test on the *Caltech* training set. Some results for this setting—-train on external data, test on the *Caltech* training set—have been published on the same website[1] as the database, and we got results for additional algorithms directly from Piotr Dollár for comparison. We will show that our enhanced detector using HOG, motion, and CSS outperforms all previously evaluated algorithms by a large margin, often by 10% or more.

TUD-Brussels contains 1326 annotated pedestrians in 508 image pairs of 640×480 pixels recorded from a car moving through an inner city district. It contains pedestrians on various scales and from various viewpoints. It comes with a training set (*TUD-MotionPairs*) of 1776 annotated pedestrians seen from multiple viewpoints taken from a handheld camera in a pedestrian zone, with a negative dataset of 192 images partially taken from the same camera and partially from a moving car. This training set is used for all experiments except for those on INRIAPerson (where the corresponding training set is used).

3. Methods

As mentioned above, both feature and classifier choice strongly influence the performance of any sliding-window based method. In the following we describe the employed features including our proposed new feature based on self-similarity as well as our modifications of the histograms of flow (HOF) feature. This section also describes the classifiers and the training procedure used in the evaluation.

[1] http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/

3.1. Features

Obviously, the choice of features is the most critical decision when designing a detector, and finding good features is still largely an empirical process with few theoretical guidelines. We evaluate different combinations of features, and introduce a new feature based on the similarity of colors in different regions of the detector window, which significantly raises detection performance. The pedestrian region in our detection window is of size 48×96 pixels. As it has been shown to be beneficial to include some context around the person [3] the window itself is larger (64×128 pixels).

HOG Histograms of oriented gradients are a popular feature for object detection, first proposed in [3]. They collect gradient information in local cells into histograms using trilinear interpolation, and normalize overlapping blocks composed of neighbouring cells. Interpolation, local normalization and histogram binning make the representation robust to changes in lighting conditions and small variations in pose. HOG was recently enriched by Local Binary Patterns (LBP), showing a visible improvement over standard HOG on the *INRIA Person* data set [24]. However, while we were able to reproduce their good results on *INRIA Person*, we could not gain anything with LBPs on other datasets. They seem to be affected when imaging conditions change (in our case, we suspect demosaicing artifacts to be the issue), see Fig. 2(a) and 2(b). Hence, we have not included HOG-LBP in further evaluations. In our experiments we compute histograms with 9 bins on cells of 8×8 pixels. Blocksize is 2×2 cells overlapping by one cellsize.

HOF Histograms of flow were initially also proposed by Dalal et al. [4]. We have shown that using them (e.g. in [4]'s `IMHwd` scheme) complementary to HOG can give substantial improvements on realistic datasets with significant egomotion. Here, we introduce a lower-dimensional variant of HOF, `IMHd2`, which encodes motion differences within 2×2 blocks with 4 histograms per block, while matching the performance of `IMHwd` (3×3 blocks with 9 histograms). Fig. 2(d) schematically illustrates the new coding scheme: the 4 squares display the encoding for one histogram each. For the first histogram, the optical flow corresponding to the pixel at the i^{th} row and j^{th} column of the upper left cell is subtracted from the one at the corresponding position of the lower left cell, and the resulting vector votes into a histogram as in the original HOF scheme. `IMHd2` provides a dimensionality reduction of 44% (2520 instead of 4536 values per window), without changing performance significantly. We used the publicly available flow implementation of [26][2]. In this work we show that HOF continues to provide a substantial improvement even for flow fields computed on JPEG images with strong block artifacts (and hence degraded flow fields).

[2]In our previous paper [27] we used the optic flow software of [29], which is a precursor of [26]. We used the updated flow library for purely technical reasons: only the newer library is available for 64 bit Linux. In our experiments we did not experience significant differences in detection performance between the two.

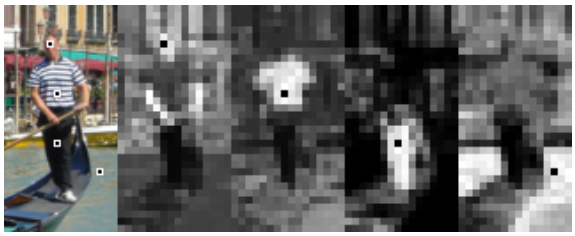
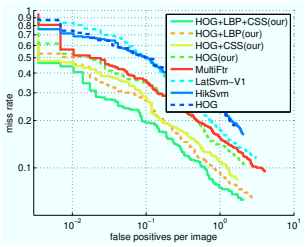


Figure 1: CSS computed at marked cell positions (*HSV*+histogram intersection). Cells with higher similarity are brighter. Note how self-similarity encodes relevant parts like clothing and visible skin regions.

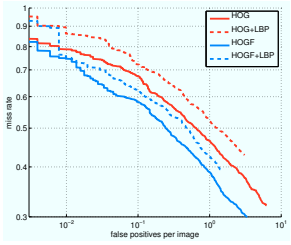
CSS Several authors have reported improvements by combining multiple types of low-level features [5, 18, 27]. Still, it is largely unclear which cues should best be used in addition to the now established combination of gradients and optic flow. Intuitively, additional features should be complementary to the ones already used, capturing a different part of the image statistics. Color information is such a feature enjoying popularity in image classification [22] but is nevertheless rarely used in detection. Furthermore, second order image statistics, especially co-occurrence histograms, are gaining popularity, pushing feature spaces to extremely high dimensions [25, 18].

We propose to combine these two ideas and use second order statistics of colors as additional feature. Color by itself is of limited use, because colors vary across the entire spectrum both for people (respectively their clothing) and for the background, and because of the essentially unsolved color constancy problem. However, people *do* exhibit some structure, in that colors are locally similar—for example (see Fig. 1) the skin color of a specific person is similar on their two arms and face, and the same is true for most people's clothing. Therefore, we encode color *self-similarities* within the descriptor window, i.e. similarities between colors in different sub-regions. To leverage the robustness of local histograms, we compute D local color histograms over 8×8 pixel blocks, using trilinear interpolation as in HOG to minimize aliasing. We experimented with different color spaces, including $3 \times 3 \times 3$ histograms in *RGB*, *HSV*, *HLS* and *CIE Luv* space, and 4×4 histograms in normalized *rg*, *HS* and *uv*, discarding the intensity and only keeping the chrominance. Among these, *HSV* worked best, and is used in the following.

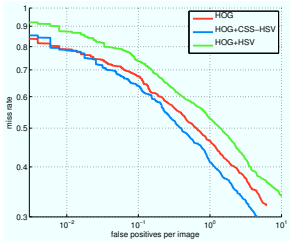
The histograms form the base features between which pairwise similarities are computed. Again there are many possibilities to define similarity between histograms. We experimented with a number of well-known distance functions including the L_1-norm, L_2-norm, χ^2-distance, and histogram intersection. We use histogram intersection as it worked best. Finally, we apply L_2-normalization to the



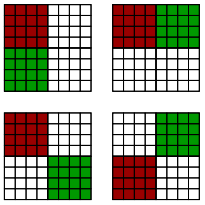
(a) LBP and CSS on INRIAPerson



(b) LBP on TUD-Brussels



(c) CSS vs. raw color (TUD-Brussels)



(d) IMHd2 coding scheme

Figure 2: (a)-(c) Performance comparisons. Detections and labels on (a) are taken from the *Caltech*[6] website, plus ours. The classifier in (b,c) and for our curves in (a) is HIKSVM. (d) IMHd2 coding scheme for the pixelwise differences (4×4 cells are shown for simplicity, actual cell size is 8×8).

$(D\cdot(D-1)/2)$-dimensional vector of similarities. In our implementation with $D = 128$ blocks, CSS has 8128 dimensions. Normalization proved to be crucial in combination with SVM classifiers. Note that CSS circumvents the color-constancy problem by only comparing colors locally. In computation cost, CSS is on the same order of magnitude as HOF.

Fig. 2(c) supports our claim that self-similarity of colors is more appropriate than using the underlying color histograms directly as features. CSS in *HSV* space yields a noticeable improvement. On the contrary adding the color histogram values directly even hurts the performance of HOG. In an ideal world this behavior should not occur, since SVM training would discard un-informative features. Unfortunately this holds only if the feature statistics are identical in the training and test sets. In our setup—and in fact quite often in practice—this is not the case: the training data was recorded with a different camera and in different lighting conditions than the test data, so that the weights learned for color do not generalize from one to the other. A similar observation was made by [27], who found that adding Haar features can sometimes help, but careful normalization is required, if the imaging conditions vary. Note that [5] do sucessfully utilize (raw) color, and in future work we plan to look into ways of incorporating it robustly into our detector (e.g. skin color may in principle be a sensible cue).

Note that self-similarity is not limited to color histograms and directly generalizes to arbitrary localized subfeatures within the detector window. We experimented with self-similarity on HOG blocks (see Fig. 3) as well as flow histograms, but we did not see significant gains.

3.2. Classifiers

We stick with those classifiers which performed best in recent evaluations [6, 27]: support vector machines with linear kernel and histogram intersection kernel (HIK), and MPLBoost [2]. Since AdaBoost did not yield competetive results, we chose not to include it here.

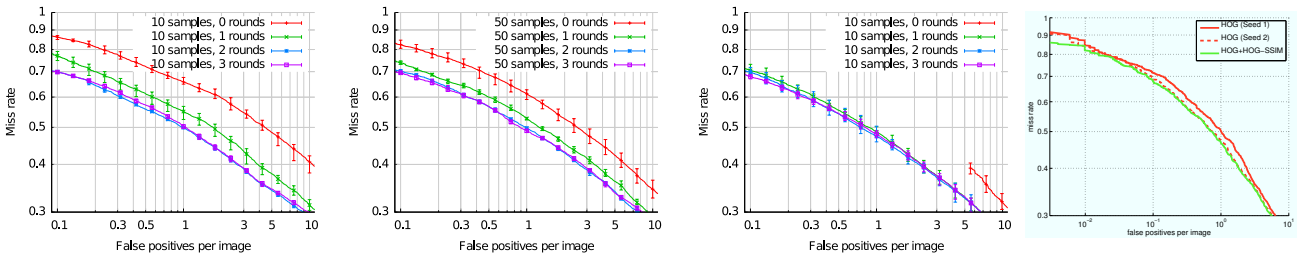
SVM Linear SVMs remain a popular choice for people detection because of their good performance and speed. Nonlinear kernels typically bring some improvement, but commonly the time required to classify an example is linear in the number of support vectors, which is intractable in practice. An exception is the (histogram) intersection kernel (HIK) [14], which can be computed exactly in logarithmic time, or approximately in constant time, while consistently outperforming the linear kernel.

MPLBoost Viola et al. [23] used AdaBoost in their work on pedestrian detection. However, it has since been shown that AdaBoost does not perform well on challenging datasets with multiple viewpoints [27]. MPLBoost remedies some of the problems by learning multiple (strong) classifiers in parallel. The final score is then the maximum score over all classifiers, allowing individual classifiers to focus on specific regions of the feature space without degrading the overall classification performance.

3.3. Training procedure

A crucial point in training, which is often underestimated in literature, is the search for hard examples in the negative dataset, more specificially the number of retraining ("bootstrapping") iterations that are used. Dalal and Triggs [3] state that after one round "additional rounds of retraining make little difference so we do not use them". Felzenszwalb et al. [10] prove that repeated retraining leads to convergence for SVMs and repeat their training procedure—including the search for hard samples—10 times. Dollár et al. [5] use two bootstrapping rounds.

In Fig. 3(a) one can clearly see the influence of repeated retraining. Shown are the mean recall and maximum deviation for a fixed false positive rate, computed over five runs with different randomly selected sets of initial negative samples. The results are shown on *TUD-Brussels* for the HOG classifier paired with a linear SVM (chosen here because of its popularity). 10 negative samples are selected per training image at random, for a total of 1920 initial negative samples. Two results are immediately visible: with less than two bootstrapping rounds, performance depends heavily on the initial training set. In fact the variance is in the same order of magnitude as typical performance gaps between algorithms, leading comparisons *ad absurdum*. Furthermore, the figure shows that *at least two* retraining rounds are re-



(a) 1920 initial negatives, linear SVM (b) 9600 initial negatives, linear SVM (c) 1920 initial negatives, MPLBoost (d) different random seeds, MPLBoost

Figure 3: Impact of additional retraining rounds.

quired to reach the full performance of the standard combination *HOG + linear SVM*.

One may argue that instead of additional bootstrapping rounds one could select more negative samples from the beginning. Fig. 3(b) shows that this is not the case: selecting 50 initial negatives per image (9600 total) somewhat alleviates the problem, but does not solve it. What's more, after 2—3 retraining rounds the advantages of using more initial samples vanishes, which confirms the strategy to concentrate on *hard* negatives.

For boosting classifiers (Fig. 3(c))[3], the situation is worse: although mean performance seems stable over bootstrapping rounds, the overall variance only decreases slowly—the initial selection of negative samples has a high influence on the final performance even after 3 bootstrapping rounds. Because of this, we show the superior performance of our new feature with HIKSVMs which have very good performance and where convergence during the iterative retraining phase is guaranteed [10]. We verified the required number of bootstrapping rounds experimentally.

Fig. 3(d) shows an example of a setting in which naive comparisons can even lead to unjustified conclusions. The used classifier is MPLBoost after 1 bootstrapping round, with either HOG features (red) or HOG plus self-similarity on HOG blocks (green). The only difference between the dashed and the solid green curve is the initial negative set. Had we only done one experiment with one bootstrapping round for this comparison, we might have come to the conclusion that self-similarity on HOG blocks helps significantly. It is important to make sure the result does not depend on the initial selection of negative samples, e.g. by retraining enough rounds with SVMs, as done in this paper.

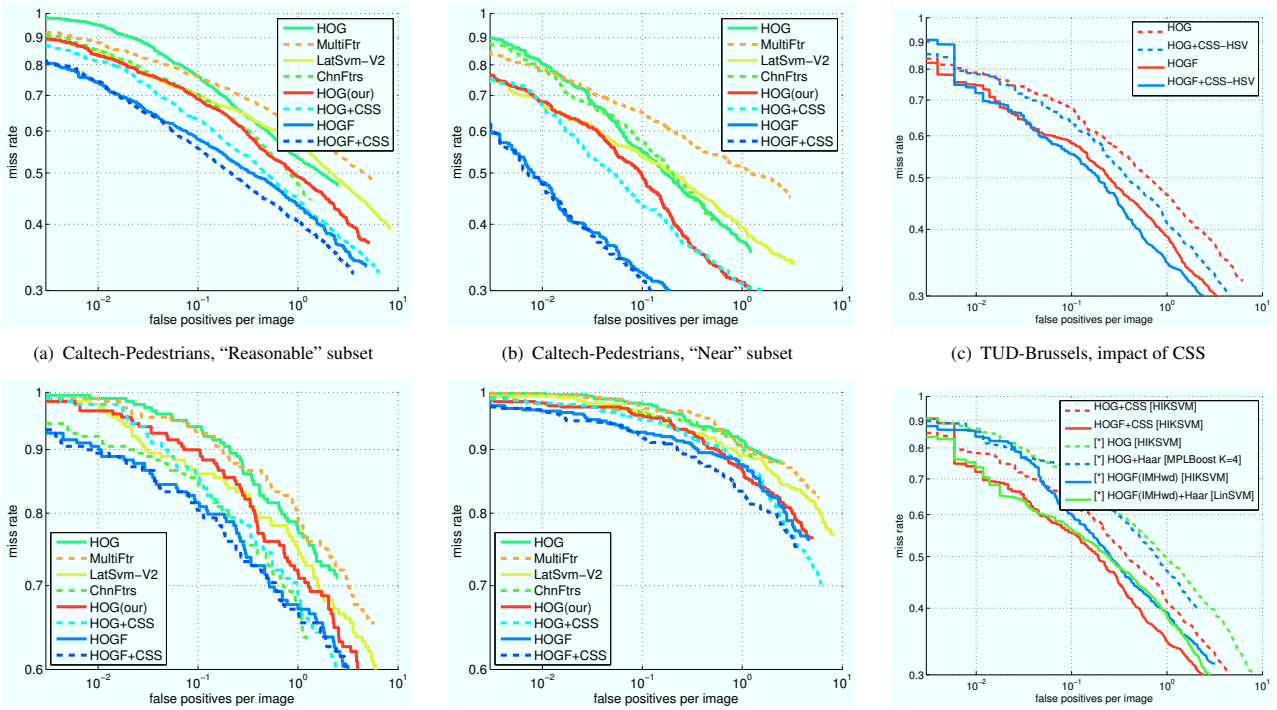
4. Results

We continue with a detailed description of the results obtained with different variants of our detector. On *Caltech Pedestrians*, we used the evaluation script provided with the dataset. The plots (in Fig. 4) are stretched horizontally to improve readability (they end at 10fppi, instead of 100fppi as in the original publication). For *TUD-Brussels* we evaluate on the full image, including pedestrians at the image borders (in contrast to [27]), who are particularly important for practical applications—e.g. for automotive safety, near people in the visual periphery are the most critical ones. Unless noted otherwise, the classifier used with our detector is HIKSVM.

Fig. 4(a) shows the performance on the "reasonable" subset of *Caltech Pedestrians*, which is the most popular portion of the data. It consists of pedestrians of $\geq$50 pixels in height, who are fully visible or less than 35% occluded. Our detector in its strongest incarnation, using HOG, HOF and CSS in a HIKSVM (HOGF+CSS), outperforms the previous top performers—the *channel features* (ChnFtrs) of [5] and the *latent SVM* (LatSvm-V2) of [10]—by a large margin: 10.9% at 0.01 fppi, 14.7% at 0.1 fppi and 7.0% at 1 fppi. Note that the interesting part of the plots is the left region, since more than 1fppi is usually not acceptable in any practical application. We also point out that our baseline, HOG with HIKSVM, is on par with the state of the art [5, 10], which illustrates the effect of correct bootstrapping, and the importance of careful implementation. We did not tune our detector to the dataset. Still, to make sure the performance gain is not dataset-specific, we have verified that our detector outperforms the original HOG implementation [3] also on *INRIAPerson* (cf. Fig. 2(a), also note that adding CSS provides an improvement for HOG+LBP). HOG+CSS is consistently better than HOG alone, providing an improvement of 5.9% at 0.1fppi, which indicates that color self-similarity is indeed complementary to gradient information. HOG+HOF improves even more over HOG, especially for low false positive rates: at 0.1fppi the improvement is 10.9%. This confirms previous results on the power of motion as a detection cue. Finally, HOG+HOF+CSS is better than only HOG+HOF, showing that CSS also contains information complementary to the flow, and achieves our best result of 44.35% recall at 0.1fppi.

In Fig. 4(b), the performance on the "near" subset (80

[3]The red curve is shortened because of precision issues – a lot of samples have scores very close to 1, which get mapped to 1 during serialization. This is not an issue for our detector since this vanishes after one round of bootstrapping and during training, no serialization happens (and all those examples are hard examples anyway).



(a) Caltech-Pedestrians, "Reasonable" subset (b) Caltech-Pedestrians, "Near" subset (c) TUD-Brussels, impact of CSS

(d) Caltech-Pedestrians, "Partial Occlusion" (e) Caltech-Pedestrians, "Heavy Occlusion" (f) Comparison with [27]

Figure 4: Results obtained with different combinations of features and classifiers on different datasets.

pixels or taller) is shown. Again, our baseline (HOG(our)) is at least on par with the state of the art [5, 10]. HOG+CSS provides better performance between 0.01 and 0.5 fppi, 6% at 0.1 fppi. Adding HOF to HOG (HOGF) adds 19.9% recall at 0.01 fppi. At 0.1fppi it beats the closest competitor HOG+CSS by 11% and the best published result (LatSvm-V2) by 21.2%. Adding CSS brings another small improvement for large pedestrians. The reason that HOF works so well on the "near" scale is probably that during multi-scale flow estimation compression artifacts are less visible at higher pyramid levels, so that the flow field is more accurate for larger people.

Fig. 4(d) and 4(e) show the evaluation for increasing occlusion levels. Not shown are the plots for the "no occlusion" subset, which are almost identical to Fig. 4(a), because only ≈5% of the "reasonable" pedestrians are partially occluded. Plots are also stretched vertically to provide for better readability. Evaluated on the partially occluded pedestrians alone (which is not a significant statistic, because there are only about 100 such examples), *latent SVM* and *channel features* slightly outperform our HOG, but again are dominated by HOG+HOF, with CSS again bringing a further small improvement. On the heavily occluded pedestrians (Fig. 4(e)), the performance of all evaluated algorithms is abysmal. A lack of robustness to heavy occlusion is a well-known issue for global detectors. Still, the relative improvement with our detector is very noticeable: at 0.1 fppi, the recall of HOG+HOF+CSS is at 7.8% compared to 3.9% for *ChnFtrs*, doubling the recall. At 1fppi, our full detector still performs best, with 5.9% higher recall than LatSvm-V2. That color self-similarity helps in the presence of occlusion may seem counter-intuitive at first, because occlusion of a local sub-region is likely to affect its similarity to all other sub-regions. However, in the case of *Caltech*, "heavy occlusion" mostly means that the lower part of the body is occluded, so that similarities between different parts of the upper body can still be used.

Fig. 4(c) shows the improvement gained by adding CSS on the *TUD-Brussels* dataset. CSS adds little in the high precision regime, but starting at 0.05 fppi there is a notable boost in performance, as recall is improved by 2.7% at 0.1fppi and 4.2% at 1 fppi. For static images with no flow information, the improvement starts earlier, reaching 3.6% at 0.1 fppi and 5.4% at 1 fppi.

Finally, Fig. 4(f) compares to the results of [27] on *TUD-Brussels*. In this paper Haar features did provide an improvement only on that dataset, on others they often cost performance. This is in contrast to CSS, which so far have produced consistent improvements, even on datasets with very different image quality and color statistics. Judging from the available research our feeling is that Haar features can potentially harm more than they help. We have nevertheless included the best results with *and* without using Haar features as reference.

For the static image setting, HOG+CSS (dashed red) consistently outperforms the results of [27] by 5%–8%

against HOG+Haar with MPLBoost (dashed blue), and by 7%–8% against HOG with HIKSVM (dashed green). Utilizing motion, the detector of [27] using HOG+HOF (in the `IMHwd` scheme), Haar features and a linear SVM (solid blue) is on par with HOG+HOF+CSS (solid red) for low false positive rates, but it starts to fall back at 0.2 fppi. The result of [27] using HOG+HOF with HIKSVM (solid green) is consistently worse by 3%–5% than HOG+HOF+CSS, especially at low false positive rates.

5. Some insights on evaluation

Another message of our investigation is that it is imperative to follow not only the same evaluation protocol, but to use *identical* scripts for the evaluation, in order to make results comparable, and even to make them meaningful at all. There are many design choices for evaluation scripts, some of which are often taken implicitely. Often, only the condition for two bounding boxes to match (e.g. the "Pascal condition" [9], $\frac{\text{intersection}}{\text{union}} > 50\%$) is specified, which is not enough, as we will show.

We therefore suggest that the release of a dataset should always be accompanied by a suitable evaluation script, and that the raw detections should be published together with the corresponding curves. We have in all cases used the tools and detections used in the original publications [6, 27] for the respective datasets.

Reliably finding every pedestrian in an image, regardless of size, is impossible even for a human. Therefore, and also to evaluate for a specific scale range or get rid of boundary effects, most of the time a subset of all annotated pedestrians is used in evaluations. This is often done in an underspecified way, and we will show how it distorts the results and introduces artifacts one has to be aware of.

As an example, let us examine the evaluation on the "far" subset of the *Caltech* dataset. In this setting, only pedestrians with an annotated height $20 \leq h < 30$ pixels are to be considered. Detections fulfilling the Pascal condition can be as small as 10 pixels or as large as 59 pixels. Any annotation inside the 20–30 pixel range can be matched by a detection outside the range. This introduces an asymmetry which is difficult to handle. The *Caltech* evaluation script discards all detections outside the considered range, resulting in situations where a pedestrian with an annotated height of 29 pixels and a detected height of 30 pixels counts as a missed detection, although $\frac{I}{U} > 90\%$.

This is clearly undesirable, especially if many annotations are close to the size limit (which is always the case for small size ranges). However, trying to fix this bias introduces other ones. One possibility is to establish correspondence with the full sets of annotation and detection, and prune for size afterwards. For the discussion, we split the set of annotations and detections into *considered* and *ignored* sets based on the evaluation criteria. Annotations can fall into the *ignored* set because of size, position, occlusion level, aspect ratio or non-pedestrian label in the Caltech setting. Detections can fall into the *ignored* set because of size. E.g. if we wish to evaluate on 50-pixel-or-taller, unoccluded pedestrians, any annotation labeled as occluded and any annotation or detection <50 pixels falls in the *ignored* set.

The situation is relatively clear-cut for *considered* detections: if they match a *considered* annotation they count as true positive, if they match no annotation, or only one that has already been matched to another detection[4], they count as false positive, and if they match an *ignored* annotation they are discarded. However, things are less clear for *ignored* detections: if an *ignored* detection matches an *ignored* annotation, it should be discarded. If an *ignored* detection matches no annotation, it seems reasonable to discard it, but this may introduce a bias, as will be seen shortly. If an *ignored* detection matches a *considered* annotation, applying the Pascal condition suggests counting it as a true positive, and this is also the most consistent way to handle it over different settings (otherwise the same pedestrian could count as a false negative in the "far" setting, but as a true positive in the "overall" setting). However, allowing these matches introduces another problem: if one at the same time discards *ignored* detections matching no annotation, then the evaluation becomes vulnerable to (intended or unintended) exploitation: when, for example, one targets the "far" experiment, one could densely flood the image with detections just above/below the size limit. These will contain a valid match for every annotation inside the size range, but will be ignored if they do not match an annotation, leading to 100% recall without a single false positive. This effect is not limited to malicious flooding: parameter values that generate false detections on ignored scales will appear favourable, so iterative tuning could unintentionally introduce this bias[5].

To summarize, there is no single correct way how to evaluate on a subset of annotations, and all choices have undesirable side effects. It is therefore imperative that published results are accompanied by detections, and that evaluation scripts are made public. As there are boundary effects in almost any setting (all realistic datasets have a minimum annotation size), it must be possible for others to verify that differences are not artifacts of the evaluation.

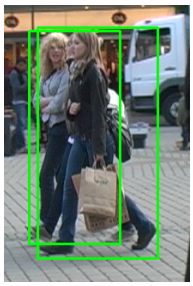


Figure 5: Pedestrians with overlapping bounding boxes so that $\frac{I}{U} \approx 0.59$.

There is another issue worth noting: [6] try to match *considered* annotations preferably, even if

[4]The *Caltech* dataset has the concept of multi-*people* regions, which are allowed to match multiple detections.

[5]The evaluation script used in [27] is susceptible to this. However, since the detector did not output detections below the threshold, this has no effect on the results in [27].

an *ignored* annotation is a better match (higher overlap). This leads to artifacts when a pedestrian occludes another one so that their bounding boxes overlap sufficiently, as is the case in Fig. 5: If the occluding pedestrian is detected and the occluded pedestrian is not, the detection will match the unoccluded pedestrian in the "unoccluded" setting, but it will count as having detected the occluded pedestrian in the "occluded" setting. A more reasonable method would be to perform the matching without looking at the *ignored* attribute, aiming to optimize overlap, and doing the evaluation afterwards. Upon publication, we will publicly release an evaluation program implementing (among others) all of the mentioned options.

6. Conclusion

Our paper pushes the state of the art in pedestrian detection in multiple ways: We have introduced powerful self-similarity features to pedestrian detection, which—when applied to color channels—provide a substantial improvement both in the single-frame setting and with additional motion information. A combination of carefully implemented HOG features, a variant of HOF to encode image motion, and the new CSS feature, together with HIKSVM as classifier, outperforms the state of the art published state of the art by 5%–20% over a wide range of precision.

Concerning classifier training, we have shown that care has to be taken when comparing competing feature combinations: the improvement gained by introducing a feature can vary, and even vanish, as a function of a commonly underestimated parameter, the number of bootstrapping rounds.

On a meta-level we have pointed out that carefully specified evaluation procedures are needed in order to yield sensible performance metrics. Even seemingly harmless measures can introduce unwanted biases in the evaluation, and comparisons are essentially meaningless unless conducted with the same evaluation script, on raw detector outputs.

In future work, we plan to explore further ways of utilizing self-similarity, and new robust ways of encoding color as additional feature. On a more conceptual level, we will look into ways of handling significant partial occlusion, which is a weakness of our current detector (as indeed of all global detectors we are aware of). Going further, we believe that depth is an important cue and has great potential especially for explicit occlusion reasoning. We therefore plan to investigate the role of stereo depth for detection.

Acknowledgements This work has been funded, in part, by Toyota Motor Europe. Further we thank the authors of FlowLib for publicly releasing their software, Piotr Dollár for providing datasets and results and Christian Wojek for providing code and helpful discussion.

References

[1] M. Andriluka, S. Roth, and B. Schiele. Pictorial structures revisited: People detection and articulated pose estimation. In *CVPR*, 2009.

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

[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

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

[5] P. Dollár, Z. Tu, P. Perona, and S. Belongie. Integral channel features. In *BMVC*, 2009.

[6] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: A benchmark. In *CVPR*, 2009.

[7] M. Enzweiler and D. M. Gavrila. Monocular pedestrian detection: Survey and experiments. *PAMI*, 2009.

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

[9] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results.

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

[11] D. M. Gavrila and S. Munder. Multi-cue pedestrian detection and tracking from a moving vehicle. *IJCV*, 73:41–59, 2007.

[12] I. N. Junejo, E. Dexter, I. Laptev, and P. Pérez. Cross-view action recognition from temporal self-similarities. In *ECCV*, 2008.

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

[14] S. Maji, A. C. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. In *CVPR*, 2008.

[15] P. Ott and M. Everingham. Implicit color segmentation features for pedestrian and object detection. In *ICCV*, 2009.

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

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

[18] W. Schwartz, A. Kembhavi, D. Harwood, and L. Davis. Human detection using partial least squares analysis. In *ICCV*, 2009.

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

[20] E. Shechtman and M. Irani. Matching local self-similarities across images and videos. In *CVPR*, 2007.

[21] C. Stauffer and W. E. L. Grimson. Similarity templates for detection and recognition. In *CVPR*, 2001.

[22] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluation of color descriptors for object and scene recognition. In *PAMI*, 2009. (in press).

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

[24] X. Wang, T. X. Han, and S. Yan. A HOG-LBP human detector with partial occlusion handling. In *ICCV*, 2009.

[25] T. Watanabe, S. Ito, and K. Yokoi. Co-occurrence histograms of oriented gradients for pedestrian detection. In *PSIVT*, 2009.

[26] M. Werlberger, W. Trobin, T. Pock, A. Wedel, D. Cremers, and H. Bischof. Anisotropic Huber-L1 optical flow. In *BMVC*, 2009.

[27] C. Wojek, S. Walk, and B. Schiele. Multi-cue onboard pedestrian detection. In *CVPR*, 2009.

[28] B. Wu and R. Nevatia. Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet part detectors. *IJCV*, 75(2):247–266, 2007.

[29] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime TV-L1 optical flow. In *DAGM*, 2007.

[30] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan. Fast human detection using a cascade of histograms of oriented gradients. In *CVPR*, 2006.