Informed Haar-like Features Improve Pedestrian Detection

Shanshan Zhang*, Christian Bauckhage†‡, Armin B. Cremers*‡
*University of Bonn, Germany †Fraunhofer IAIS, Germany
‡Bonn-Aachen International Center for Information Technology (B-IT)
{zhangs,abc}@iai.uni-bonn.de, christian.bauckhage@iais.fraunhofer.de

Abstract

We propose a simple yet effective detector for pedestrian detection. The basic idea is to incorporate common sense and everyday knowledge into the design of simple and computationally efficient features. As pedestrians usually appear up-right in image or video data, the problem of pedestrian detection is considerably simpler than general purpose people detection. We therefore employ a statistical model of the up-right human body where the head, the upper body, and the lower body are treated as three distinct components. Our main contribution is to systematically design a pool of rectangular templates that are tailored to this shape model. As we incorporate different kinds of low-level measurements, the resulting multi-modal & multi-channel Haar-like features represent characteristic differences between parts of the human body yet are robust against variations in clothing or environmental settings. Our approach avoids exhaustive searches over all possible configurations of rectangle features and neither relies on random sampling. It thus marks a middle ground among recently published techniques and yields efficient low-dimensional yet highly discriminative features. Experimental results on the INRIA and Caltech pedestrian datasets show that our detector reaches state-of-the-art performance at low computational costs and that our features are robust against occlusions.

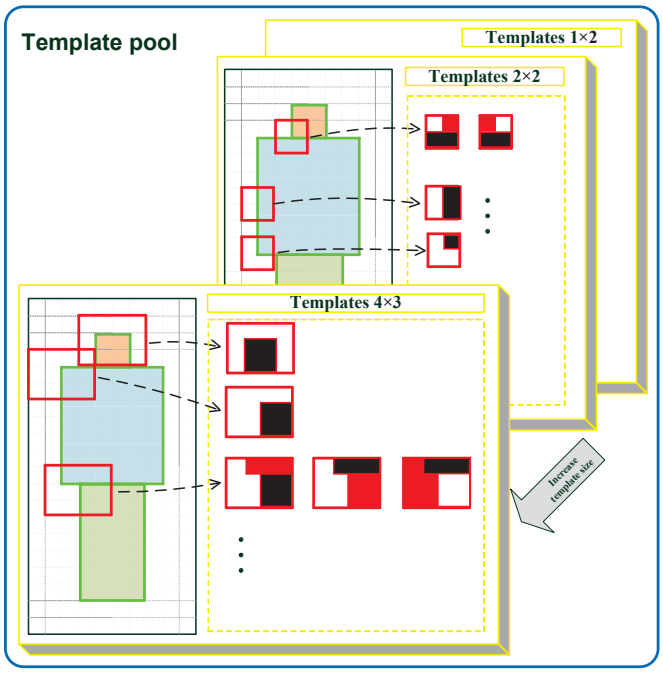


Figure 1: Illustration of our template pool. Templates are generated by sliding rectangular windows of pre-defined sizes over a pre-defined pedestrian shape model. Note that some templates are ternary (shown as white, black, and red areas) which are given the weights of $+1$, -1, and 0, respectively.

1. Introduction

Over the last decade, the question of how to detect pedestrians in images has been thoroughly investigated [10]. Yet, primarily because of random influences such as scene structure, lighting or people's choice of clothing, the problem remains challenging and continues to attract research.

A noticeable trend in this domain is that researchers increasingly rely on huge feature pools and high dimensional feature vectors [27] since it is commonly believed that more features integrate more information and thus lead to better performances. As a consequence, many recent approaches rely on the availability of powerful computers and GPU computation in order to be capable of real-time detection. Also, aspects due to the peculiar geometry of high dimensional spaces, *e.g.* concentration of measure and neighborliness, appear to be disregarded. This raises the question, if there are alternative approaches which require less memory and less computational resources but still perform robust and reliable?

In this paper, we propose more *compact features* which simultaneously ensure effectiveness and efficiency. In particular, we argue that by incorporating prior information as

to the appearance of the up-right human body, one can design reasonable features for pedestrian detection. In fact, from the point of view of visual perception, pedestrians form a class of high intra-class similarity. This is because strong regularities of up-right body shapes limit how pedestrians may appear in image data. In particular the head-shoulder area of the human body shows a geometry seldom found among other natural objects. Based on a careful exploration of these characteristics, we design new features that enable efficient, state-of-the-art pedestrian detection.

Our approach is motivated by prior work on detecting objects of rather low intra-class variability. In particular, HOGs [5] and cascaded Haar-like features [24] have become the de-facto methods of choice in this area. Yet, we note that corresponding features are either determined by means of exhaustive searches over all possible variations [24] or by means of less exhaustive random sampling [13]. In this paper, we propose a method that marks a middle ground; we design compact, discriminative Haar-like features selected from a particular *template pool* that reflects prior information about the pedestrian up-right body shape. Extensive experiments indicate that these features are highly characteristic and therefore enable very robust detection.

1.1. Related work

Because of its practical impact, research on pedestrian detection has noticeably intensified over the past decade and the literature on possible solutions is vast. Since an exhaustive survey is beyond the scope of this paper, our following review therefore focuses on *features* that have been proposed in this context.

As of this writing, the arguably most popular features for visual pedestrian detection are based on *Histograms of Oriented Gradients* (HOGs) as introduced in [5]. HOG features brought about significant improvements and therefore establish an important baseline. Felzenszwalb *et al*. [LatSvm] [12, 11] successfully employed HOG features in a part-based model for object detection; Wang *et al*. [HogLbp] [26] combined HOG features with a particular *Local Binary Pattern* (LBP) feature in order to cope with partial occlusions. Walk *et al*. [25] combined HOG features with self-similarity features related to color channels [MultiFtr+CSS] as well as motion features [MultiFtr+Motion] in order to better integrate spatial and temporal information.

Deviating from the popular framework of "HOG+SVM" computations, Dollár *et al*. [8] applied integral channel features which efficiently integrate multiple cues due to colors and gradients by means of employing integral images. For classification, they used boosting methods and thus obtained a real-time detector [ChnFtrs]. An extension of this approach has been called the "Fastest Pedestrian Detection in the West" [FPDW] [7] and was shown to enable particularly fast multiscale detection. Due to its efficiency and reasonable performance, many new detectors [3, 6] therefore consider [ChnFtrs] as a baseline and several authors obtained even better performance by extending the feature pool in various ways. Benenson *et al*. [Roerei] [4] used irregular rectangles resulting in a $718,080$ dimensional feature pool; Lim *et al*. [SketchTokens] [15] added self-similarity features, yielding a $21,350$ dimensional feature vector for image patches of a size of 35×35 pixels. Due to the extreme sizes of these feature pools, both corresponding detectors require powerful computing hardware and large amounts of memory at training time. Addressing issues like these, our work aims at building new detectors based on small but intelligently designed feature pools that enable state-of-the-art detection accuracy.

Pioneering attempts of using Haar wavelets for pedestrian detection are found in [17] where it was demonstrated that wavelet templates can be used to define the shape of an object. Later, Papageorgiou *et al*. [20] proposed a similar yet more general system for object detection and, subsequently, Haar-like features became popular in the object detection community. The epitome of such approaches is found in the work by Viola and Jones [24] who used Haar-like features in combination with boosting algorithms to build a successful face detector. Dollár *et al*. [9] proposed to use feature mining strategies to select informative features from a large amount of Haar wavelets. In this context, we note that, in the recent literature, Haar-like features are also referred to as second-order channel features [ChnFtrs]. However, Haar-like features are often discarded in pedestrian detection as they seem not to improve performance when combined with first-order channel features. In a closer analysis as to possible reasons for this behavior, we found that Haar-like templates that perform well for face detection are not necessarily suited for pedestrian detection as they may fail to capture visual characteristics of human body. As a remedy, we propose to design particularly tailored templates for up-right body shapes.

1.2. Contributions

Our main contribution in this paper is to model pedestrian shapes in terms of three rectangles that are geared towards different body parts. Based on this shape model, we design compact Haar-like features to describe local differences. Accordingly, we design a compact feature pool that is better tailored to pedestrian shapes than the ones covered in the above survey.

Template pool for pedestrian shape model: we find that up-right walking pedestrians share a common visual appearance especially w.r.t. the geometry of the head and shoulder region of the body. Based on this shape model, we design a pool of rectangle features (rectangular templates) that is adapted to these local structures. Our templates are

specific for pedestrians and therefore lead to a very good performance; on the other hand, they constitute only a small subset of the set of all possible rectangular templates so they significantly reduce training times.

Multi-modal & multi-channel Haar-like features: we use two template modalities –binary and ternary– for Haar-like features. The ternary modal is specifically proposed to represent corner regions found along the pedestrian silhouette so as to enable rectangle features to represent more complex geometric configurations. In terms of channel features, we consider rectangle descriptors not only w.r.t. colors but also w.r.t. gradients. This addresses challenges due to variations in the choice of clothes.

We evaluate our approach in extensive experiments on several benchmark datasets and demonstrate that by employing compact features, our new pedestrian detector achieves state-of-the-art performance while enjoying three advantages: it is **easy to implement**, **easy to train**, and **fast to apply** on real world data.

2. Template pool

In this section, we describe how to generate a template pool that is tailored towards visual pedestrian detection. For this purpose, we first define a pedestrian body shape model and then generate templates by sliding bounding boxes of different sizes over this shape model. Fig. 1 illustrates the whole template pool and shows examples of templates of different sizes.

2.1. A pedestrian body shape model

We define a pedestrian body shape based on statistical information. The INRIA dataset is arguably the most commonly used benchmark for image-based pedestrian detection. It contains annotated image patches showing pedestrians scaled to a height of 96 pixels; all patches are padded by 12 pixels in four directions in order to provide contextual information. We therefore perform a statistical analysis with pedestrian images of size 60×120 pixels. On these data, we compute an average edge map based on gradient magnitudes extracted from each sample. The resulting average edge map is shown in Fig. 2 and clearly resembles a human body.

Features derived from rectangular image regions typically allow for computational efficiency. We therefore decide to base our pedestrian detector on rectangular features and hence divide the edge map into square *cells* whose sizes may vary. Fig. 2 shows examples of cells of sizes 4×4 and 6×6 pixels. Given these grids of cells, the whole body is approximately divided into three parts: the head, the upper body, and the lower body. This is intended to increase robustness as these three parts generally appear in different colors or textures in real world images.

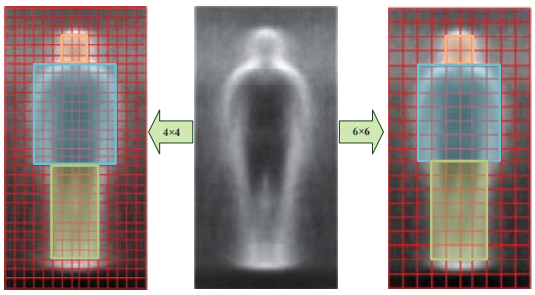


Figure 2: Illustration of a statistical pedestrian shape model in terms of an average edge map as shown in the middle. In this example, cell sizes are chosen to be 4×4 and 6×6 pixels, respectively. Three bounding boxes approximately indicate the head, the upper body, and the lower body parts.

2.2. Generating templates

We constrain our templates to be of rectangular form as these allow for convenient implementation and efficient computation. Statistical variations are coped with by considering different modalities as described in Sec. 3. First, however, we define a set of sizes

$$S = \{(w, h) \mid w \leq w_m, h \leq h_m, w, h \in \mathbb{N}^+\}, \quad (1)$$

where w and h indicate the width and height (in terms of covered cells) of a rectangular template; w_m and h_m are used to constrain the overall size of templates since we focus on local image information.

As shown in Fig. 2, images of pedestrians available in the INRIA data consist of four logical components: background, head, upper body, and lower body. We assign each cell $c(i, j)$ to a set of labels $L(i, j)$ that indicate which components are found in the cell.

Next, for each pair of sizes in S, we slide a corresponding rectangular window over the whole shape model to generate different templates at different positions and of different weights. At a certain position (x, y), the template to be created depends on how many different parts are contained in the rectangle. A binary template is generated if there are only two parts; ternary templates of different kinds are generated if there are three parts. Algorithm 1 provides details as to this procedure. The resulting full template pool is given as a set:

$$T = \{(x, y, s, W) \mid x, y \in \mathbb{N}, s \in S, W \in \mathbb{R}^2\}, \quad (2)$$

where x and y indicate the location of a template w.r.t. the human shape model and W is a weight matrix that is determined according to the matrix L of labels for all cells.

3. Multi-modal, multi-channel Haar features

In the following, traditional Haar-like features will be referred to as binary modalities as they only carry two possi-

Algorithm 1 Generating templates for pedestrian shapes

initialize template pool: $T \leftarrow \emptyset$;
for $i = 1$ to $nSize$ **do**
  for $x_1 \in [1, width - w_i]$ **do**
    for $y_1 \in [1, height - h_i]$ **do**
      $label = L(x_1 : x_1 + w_i, y_1 : y_1 + h_i)$;
      if unique(label)==2 **then**
        $W(label == l_1) \leftarrow 0$;
        $W(label == l_2) \leftarrow 1$;
        append $(x_1, y_1, (w_i, h_i), W)$ to T;
      else if unique(label)==3 **then**
        for $iCase \in [1, 3]$ **do**
          $W(label == l_{iCase}) \leftarrow 0$;
          $W(label == l_{(iCase+1)\%3}) \leftarrow -1$;
          $W(label == l_{(iCase+2)\%3}) \leftarrow 1$;
          append $(x_1, y_1, (w_i, h_i), W)$ to T;
        end for
      end if
    end for
  end for
end for
return T

ble weights ($+1$ and -1) for different rectangles. However, this binary modality is ill suited to represent cusps or corner-like structures of the human silhouette. That is to say, that it hardly adapts to the description of the content of bounding boxes that contain three different logical components such as, say, head, upper body, and background. Yet, for efficient subsequent classification we are interested in computing the difference between parts w.r.t. two of them at a time. We therefore propose to consider ternary templates. An example is given in Fig. 1 where ternary 2×2 templates capture the local geometry of the image region where head, shoulders, and background meet in joint corners.

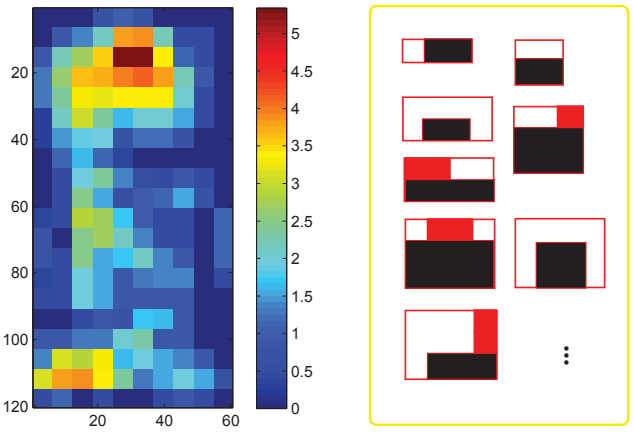
To integrate color and gradient information, we build a multi-channel descriptor for each cell. We consider a total of 10 different channels as it is done in the detector [ChnFtrs]: 3 channels for LUV colors, 1 channel for gradient magnitude information, and 6 channels for histograms of oriented gradients.

Assume we are given a template $t = (x, y, (w, h), W)$. We first count how often the weights $+1$ and -1 appear and denote these counts as n_{add} and n_{sub}. There are thus n_{add} additive cells and n_{sub} subtractive cells and we normalize each cell's weight by the total number of corresponding cells covered by a rectangle. This results in an average weight matrix:

$$W_{avg} = \frac{sgn(W)}{n_{add}} + \frac{sgn(-W)}{n_{sub}}. \quad (3)$$

The feature value of any template t for any channel k, *e.g.* color or gradient information, can then be computed as a weighted sum:

$$f(t, k) = \sum_{i=1}^{h} \sum_{j=1}^{w} \sigma(x + i, y + j, k) W_{avg}(i, j), \quad (4)$$



(a) (b)

Figure 3: Illustration of representative features. (a) Cell weight map: different colors are used to indicate the accumulative weight of each cell after boosting. (b) Most informative templates: these binary and ternary rectangle features obtained high accumulative weights after boosting.

where, $\sigma(i, j, k)$ denotes the sum of values in $cell(i, j)$ along channel k.

4. Selecting features for pedestrian detection

Our detector employs the multi-modal and multi-channel Haar-like features proposed in Sec 3. Note that these features are built on channel features as in [ChnFtrs], but interpret local differences between rectangular regions over multiple channels rather than over channel values themselves.

We apply a fast version of AdaBoost [1] for learning since it offers a convenient and fast approach to select from a large number of candidate features. We apply 2048 decision trees of depth 2 to build our final strong classifier. Initial negative training samples are randomly generated and, afterwards, hard negative samples are searched for three rounds over all negative example images so as to collect 20,000 negative samples in total. This multi-round training strategy is pivotal as it leads to a better performance than a simple one round training procedure with the same number of negative samples. From our experiments, three rounds of retraining were observed to yield optimal performance; additional rounds did not show significant improvements.

In order to look into which features are more informative, we plot a weight image of the top 100 features as shown in Fig. 3a. To generate this figure, we add the weight of each feature to the cells it covers and use different colors to indicate the accumulative weight of each cell after boosting. As expected, the head-shoulder area of the human body shows to be more discriminative for pedestrian detection than other body parts.

The most discriminative binary and ternary templates determined by the boosting algorithm are then used for pedestrian detection in still images. To this end, we slide a window over the whole image and consider multiple scales. The spatial step size is set identical to the cell size for speed and the scale step is set to be 1.09 so that there are 8 scales in each octave. We use a simplified non-maximal suppression (NMS) procedure [8] to suppress nearby detections.

5. Experiments

Experiments are conducted on two public benchmark datasets: the INRIA pedestrian dataset [5] and the Caltech pedestrian dataset [10]. The INRIA data is arguably the most popular dataset for people detection and comes along with pre-defined subsets for training and testing. The Caltech data is the largest and most challenging dataset for pedestrian detection and we consider subsets set00 - set05 for training and subsets set06 - set10 for testing.

5.1. Implementation details

To optimize our detector, we analyze the influences of different parameter settings. Next, we present various experimental results on the INRIA dataset.

Cell size: the pedestrian body shape can be covered by arrays of different cell sizes as shown in Fig. 2. We present experimental results for cell sizes of 4×4, 6×6 and 8×8. From Fig. 4a, we find that a cell size of 6×6 pixels produces the best results so we choose it as our default setting.

Channels: we plot the performance of various channel combinations. As gradient histograms have been shown as the most informative channels in [8], we only try alternatives for color and gradient magnitude channels. From Fig. 4b it appears that LUV color channels are more discriminative than HSV channels, both are commonly used in this area; using three gradient magnitude channels (one for each color channel) or two gradient components (along the x and y directions respectively) lead to slight decrease in performance rather than improvements.

Image normalization: we analyze the influence of intensity normalization on our features as previous works on rectangular features typically employ various ways of normalization. [VJ] [24] used local normalization inside each detection window; [Roerei] [4] reported performance improvements by applying global normalization on the input images. However, according to the results in Fig. 4c, our features obtain best results without normalization.

Smoothing: while pre-smoothing input images with binomial filters of radius 1 improves the performance by more than 3%, larger radii produce worse results; post-smoothing of channel features significantly decreases the performance and seems to inhibit characteristic local variations.

Number of weak classifiers: intuitively one would expect more weak classifiers to lead to better performance since decision boundaries become more accurate; on the other hand, too large number of weak classifiers may lead to overfitting of the training data. Accordingly, we find that detection performance starts to decrease slightly when the number of weak classifiers exceeds 2000.

For the results reported next, we therefore consider the following settings of our detector: cell size of 6×6;channels of LUV+GM+GH; image smoothing with binomial filters of radius 1; no channel smoothing; no image normalization; 2000 weak classifiers.

5.2. Comparisons with state-of-the-art detectors

In this section, we compare our detector to other state-of-the-art detectors whose results are publicly available[1]. We use the same experimental protocol as in [10] and evaluate performances in terms of ROC curves. Measurements of *average miss rates* are used to summarize the overall performances of different detectors. The overall results are produced on the *reasonable* [10] subset of each test set which show pedestrians at a resolution of over 50 pixels in height and a visibility of at least 65%.

The results in Fig. 5a show that our detector outperforms the baseline detector [ChnFtrs] by about 8% and reaches the state-of-the-art performance. The two detectors with better results than ours consider feature pools that are more than 20 times larger and are about 100 times slower in training.

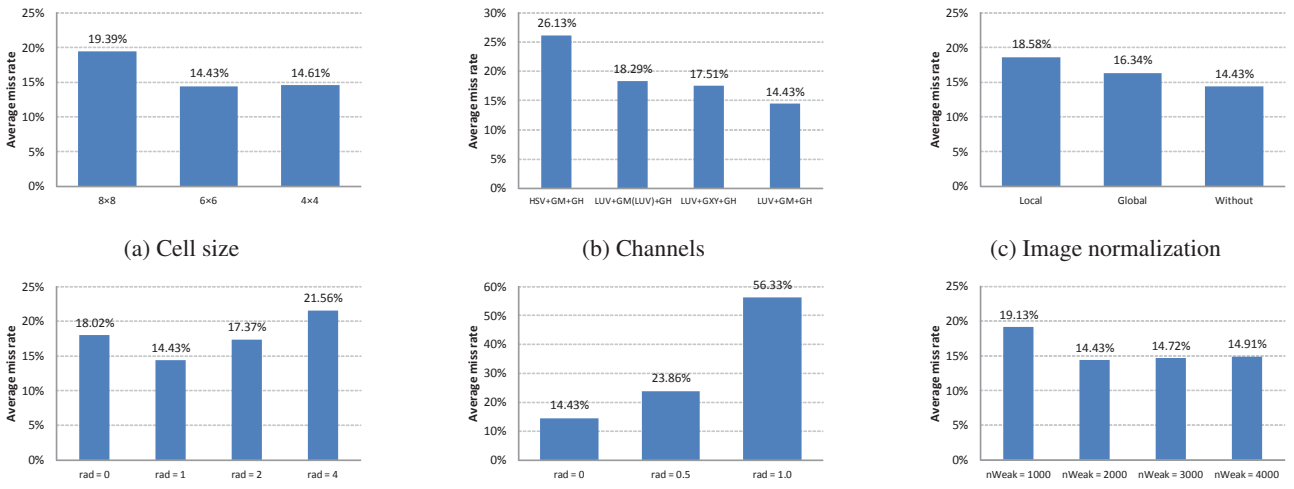
On the Caltech pedestrian dataset, our detector outperforms not only the baseline detector [ChnFtrs] by about 20% but also yields the overall best performance as shown in Fig. 5b. In particular, we note that it even outperforms detectors which consider additional motion information.

Fig. 6 shows evaluation results under different occlusion conditions for the Caltech pedestrian test data. As in [10], we use three occlusion levels: no occlusion (0% occluded), partial occlusion (1-35% occluded), and heavy occlusion (35%-80% occluded). The performance of all the detectors drops significantly as occlusion increases. Yet, our detector seems least affected by occlusion in the sense that it consistently ranks high for all occlusion levels. In fact, it achieves the best performance among all tested detectors for the cases of no and heavy occlusion and we conclude that the informed design of our features yields robustness against occlusions. Notably, our detector even outperforms those detectors that employ explicit occlusion handling strategies, *e.g.* [DBN-Isol] and [DBN-Mut], for all levels of occlusion.

5.3. Feature size and runtimes

We present our feature size with the optimal settings concluded from Sec. 5.1. Given 6×6 cells and templates size ranging from 1×2 to 4×3 cells, we obtain 266 templates

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



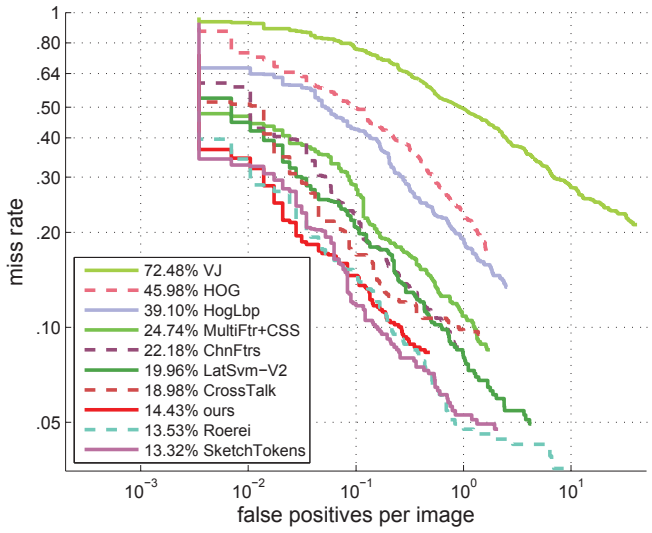
(a) Cell size (b) Channels (c) Image normalization

(d) Pre-smooth on colors (e) Post-smooth on channels (f) # Weak classifiers

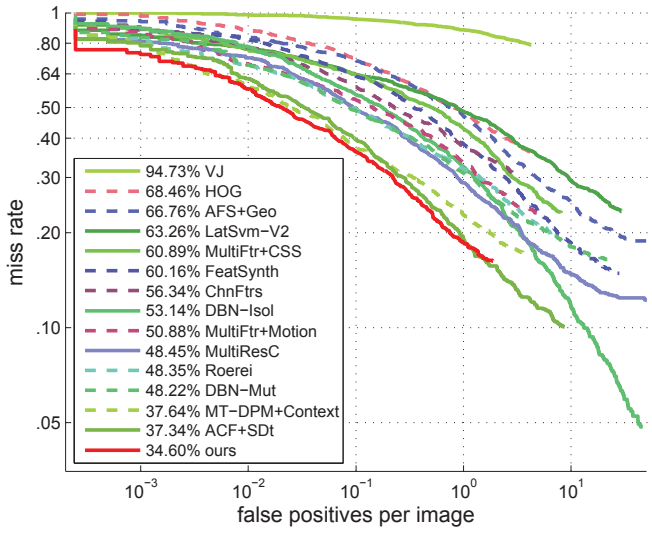
Figure 4: Evaluation of different parameters on the INRIA pedestrian dataset. (a) Cell sizes of the pedestrian shape model. (b) Channel combinations with color channels + gradient magnitude channels (GM) + gradient histogram channels (GH). (c) Image normalization methods. Local intensity normalization is done inside each detection window; global normalization is done for the whole input image. (d) Pre-smoothing of colors with binomial filters of different radii. (e) Post-smoothing of channels with binomial filters of different radii. (f) Number of weak classifiers.

Detector	Features	Classifier	Training data	Average miss rate	
				INRIA	Caltech
VJ[24]	Haar	AdaBoost	INRIA	72.48%	94.73%
HOG[5]	HOG	linear SVM	INRIA	45.98%	68.46%
Shapelet[23]	gradients	AdaBoost	INRIA	81.70%	91.37%
MultiFtr+CSS [25]	HOG + CSS	AdaBoost	INRIA	24.74%	60.89%
MultiFtr+Motion [25]	HOG + CSS + motion	linear SVM	TUD-Motion	/	50.88%
HikSvm [16]	HOG	HIK SVM	INRIA	42.82%	73.39%
HogLbp [26]	HOG + LBP	linear SVM	INRIA	39.10%	67.77%
LatSvm-V1 [12]	HOG	latent SVM	PASCAL	43.83%	79.78%
LatSvm-V2 [11]	HOG	latent SVM	INRIA	19.96%	63.26%
ChnFtrs [8]	channels	AdaBoost	INRIA	22.18%	56.34%
FeatSynth [2]	HOG + texture	linear SVM	INRIA	30.88%	60.16%
MultiResC [21]	HOG	latent SVM	Caltech	/	48.45%
CrossTalk [6]	channels	AdaBoost	INRIA	18.98%	53.88%
VeryFast [3]	channels	AdaBoost	INRIA	15.96%	/
SketchTokens [15]	channels	AdaBoost	INRIA	13.32%	/
Roerei [4]	channels	AdaBoost	INRIA	13.53%	48.35%
AFS+Geo [14]	HOG + texture	linear SVM	INRIA	/	66.76%
MT-DPM+Context [28]	HOG	latent SVM	Caltech	/	37.64%
DBN-Isol [18]	HOG	DeepNet	INRIA	/	53.14%
DBN-Mut [19]	HOG	DeepNet	INRIA	/	48.22%
ACF+SDt [22]	channels + motion	AdaBoost	Caltech	/	37.34%
ours-INRIA	Informed Haar	AdaBoost	INRIA	14.43%	/
ours-Caltech	Informed Haar	AdaBoost	Caltech	/	34.60%

Table 1: Performance comparisons for state-of-the-art pedestrian detectors. Each row in this table summarizes information as to features and classifiers used in a particular approach, and displays the corresponding average performance in terms of miss rates. The approach proposed in this paper yields state-of-the-art performance on the INRIA dataset and consistently better results than previously reported on the Caltech dataset.



(a) INRIA



(b) Caltech test

Figure 5: Results of different detectors on different datasets under standard evaluation settings.

at different positions. Shifting templates along 4 directions with a step of one cell yields a template pool of 1276 (some shifts are not possible at image borders); considering 10 channels, the final feature size is 12,760.

Our detector is implemented in Matlab, on an Intel Core-i7 CPU (3.5GHz). On the Caltech dataset, it takes 1 hour for training with 4 rounds and 1.6 seconds ([ChnFtrs] 2s) for testing a 640×480 image using the optimal parameters as illustrated in Sec. 5.1. In addition to channel computation, our feature computation includes local sums and differencing, both of which can be parallelized for further speed-up. Our detector is expected to reach real-time efficiency running on a powerful machine and with GPU computation enabled.

6. Conclusion

We considered the problem of efficient yet robust pedestrian detection from image data. The particular approach we presented in this paper was motivated by the observation that a current trend in work on pedestrian detection consists in analyzing feature vectors of ever increasing dimensions which necessitate the use of powerful hardware in order to guarantee real time capability.

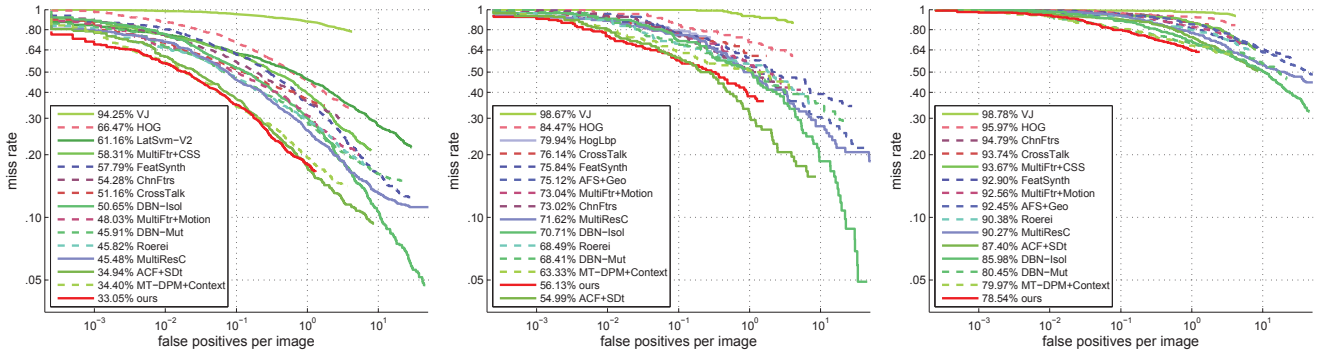
Also, because of the peculiar geometry of high dimensional spaces (concentration of measure and neighborliness) it is not necessarily guaranteed that additional efforts spent on computing high dimensions pay off in terms of recognition accuracies. We therefore explored more compact features could yield state-of-the-art performance in pedestrian detection if they were designed based on prior information as to the appearance of the up-right human body.

Given a large dataset of pedestrian images, we computed a statistical shape model which proved to consist of four clearly recognizable logical components. We covered this shape model with grids of cells and slid rectangular windows over these cell arrays to produce a set of location specific weighted binary or ternary Haar-like templates that incorporate information as to which of the four components of the shape are covered by a rectangle.

The weighting scheme provided us with a simple mechanism of generating multi-modal & multi-channel Haar-like features and we applied boosting to determine the most informative ones. As our approach does not require computing any possible configuration of rectangles within a sliding window nor is based on random sampling of rectangle features, it marks a middle ground among recently published similar approaches. Moreover, our detector is inherently simple to implement, easy to train, and fast during runtime.

In extensive experiments with standard benchmark datasets, we found our detector to achieve state-of-the-art performance on the INRIA pedestrian dataset and, for the Caltech pedestrian dataset, we found it to outperform all other recent approaches considered in our tests. In addition, our model-based rectangular features proved to be highly robust under occlusion and even outperformed methods that contain explicit mechanisms for occlusion handling.

Given these results, it appears promising to further explore model driven design of efficient rectangular features. Immediate extensions of the approach presented in this paper could be to incorporate additional channels such as motion information. More challenging extensions consist in adapting our scheme to scenarios where the objects to be detected show higher intra-class variations.



(a) No occlusion (b) Partial occlusion(1-35% occluded) (c) Heavy occlusion(35%-80% occluded)

Figure 6: Evaluation results under different occlusion conditions on the Caltech pedestrian test dataset.

References

[1] R. Appel, T. Fuchs, P. Dollár, and P. Perona. Quickly boosting decision trees-pruning underachieving features early. In *ICML*, 2013. 4

[2] A. Bar-Hillel, D. Levi, E. Krupka, and C. Goldberg. Part-based feature synthesis for human detection. In *ECCV*, 2010. 6

[3] R. Benenson, M. Mathias, R. Timofte, and L. V. Gool. Pedestrian detection at 100 frames per second. In *CVPR*, 2012. 2, 6

[4] R. Benenson, M. Mathias, T. Tuytelaars, and L. V. Gool. Seeking the strongest rigid detector. In *CVPR*, 2013. 2, 5, 6

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 2, 5, 6

[6] P. Dollár, R. Appel, and W. Kienzle. Crosstalk cascades for frame-rate pedestrian detection. In *CVPR*, 2012. 2, 6

[7] P. Dollár and P. Perona. The fastest pedestrian detector in the west. In *BMVC*, 2010. 2

[8] P. Dollár, Z. Tu, P. Perona, and S. Belongie. Integral channel features. In *BMVC*, 2009. 2, 5, 6

[9] P. Dollar, Z. Tu, H. Tao, and S. Belongie. Feature mining for image classification. In *CVPR*, 2007. 2

[10] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: an evaluation of the state of the art. *IEEE Trans. PAMI*, 34(4):743–761, 2011. 1, 5

[11] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. PAMI*, 32(9):1627–1645, 2010. 2, 6

[12] P. F. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, 2008. 2, 6

[13] J. Gall, A. Yao, N. Razavi, L. V. Gool, and V. Lempitsky. Hough forests for object detection, tracking, and action recognition. *IEEE Trans. PAMI*, 33(11):2188–2202, 2011. 2

[14] D. Levi, S. Silberstein, and A. Bar-Hillel. Fast multiple-part based object detection using kd-ferns. In *CVPR*, 2013. 6

[15] J. J. Lim, C. L. Zitnick, and P. Dollár. Sketch tokens: a learned mid-level representation for contour and object detection. In *CVPR*, 2013. 2, 6

[16] S. Maji, A. C. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. In *CVPR*, 2008. 6

[17] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio. Pedestrian detection using wavelet templates. In *CVPR*, 1997. 2

[18] W. Ouyang and X. Wang. A discriminative deep model for pedestrian detection with occlusion handling. In *CVPR*, 2012. 6

[19] W. Ouyang, X. Zeng, and X. Wang. Modeling mutual visibility relationship with a deep model in pedestrian detection. In *CVPR*, 2013. 6

[20] C. Papageorgiou and T. Poggio. A trainable system for object detection. *IJCV*, 38(1):15–33, 2000. 2

[21] D. Park, D. Ramanan, and C. Fowlkes. Multiresolution models for object detection. In *ECCV*, 2010. 6

[22] D. Park, C. L. Zitnick, D. Ramanan, and P. Dollár. Exploring weak stabilization for motion feature extraction. In *CVPR*, 2013. 6

[23] P. Sabzmeydani and G. Mori. Detecting pedestrians by learning shapelet features. In *CVPR*, 2007. 6

[24] P. Viola and M. J. Jones. Robust real-time face detection. *IJCV*, 57(2):137–154, 2004. 2, 5, 6

[25] S. Walk, N. Majer, K. Schindler, and B. Schiele. New features and insights for pedestrian detection. In *CVPR*, 2010. 2, 6

[26] X. Wang and T. X. Han. An HOG-LBP human detector with partial occlusion handling. In *ICCV*, 2009. 2, 6

[27] C. Wojek, S. Walk, and B. Schiele. Multi-cue onboard pedestrian detection. In *CVPR*, 2009. 1

[28] J. Yan, X. Zhang, Z. Lei, S. Liao, and S. Z. Li. Robust multi-resolution pedestrian detection in traffic scenes. In *CVPR*, 2013. 6