Efficient Pedestrian Detection via Rectangular Features Based on a Statistical Shape Model

Shanshan Zhang, *Student Member, IEEE*, Christian Bauckhage, *Member, IEEE*, and Armin B. Cremers

***Abstract*—Automatic pedestrian detection for advanced driver assistance systems (ADASs) is still a challenging task. Major reasons are dynamic and complex backgrounds in street scenes and variations in clothing or postures of pedestrians. We propose a simple yet effective detector for robust pedestrian detection. Observing that pedestrians usually appear upright in video data, we employ a statistical model of the upright human body in which the head, upper body, and lower body are treated as three distinct components. Our main contribution is to systematically design a pool of rectangular features that are tailored to this shape model. As we incorporate different kinds of low-level measurements, the resulting multimodal and multichannel Haar-like features represent characteristic differences between parts of the human body but are robust against variations in clothing or environmental settings. Our approach avoids exhaustive searches over all possible configurations of rectangular features nor does it rely 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 well-established INRIA, Caltech, and KITTI pedestrian data sets show that our detector reaches state-of-the-art performance at low computational costs and that our features are robust against occlusions.**

***Index Terms*—Advanced driver assistance systems (ADASs), channels, Haar-like features, pedestrian detection.**

I. Introduction

VISION-BASED pedestrian detection attracts increasing attention in the academic community since it is a topic of considerable practical interest, for instance in video surveillance and on-board driving assistance [1]. For surveillance settings in which the camera is fixed and the background is static, significant progress has been made [2]. However, pedestrian detection for advanced driver assistance systems (ADASs) is still a challenging problem, primarily because of camera motion, dynamic backgrounds, and changing illumination conditions in complex outdoor environments and, particularly, variances in clothing, appearance, viewpoint, and posture of pedestrians.

Manuscript received March 11, 2014; revised May 19, 2014; accepted July 14, 2014. The Associate Editor for this paper was B. Morris.

S. Zhang is with the Department of Computer Science III, University of Bonn, 53117 Bonn, Germany (e-mail: zhangs@iai.uni-bonn.de).

C. Bauckhage is with Fraunhofer Institute for Intelligent Analysis and Information Systems (IAIS), 53757 Sankt Augustin, Germany, and also with Bonn–Aachen International Center for Information Technology (B-IT), 53113 Bonn, Germany (e-mail: christian.bauckhage@iais.fraunhofer.de).

A. B. Cremers is with the Department of Computer Science III, University of Bonn, 53117 Bonn, Germany, and also with Bonn–Aachen International Center for Information Technology (B-IT), 53113 Bonn, Germany (e-mail: abc@iai.uni-bonn.de).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2014.2341042

Over the last decade, vision-based pedestrian detection has been extensively investigated [3], [4]. Efforts were made toward the design of new features [5]–[8]. Other work focused on improving classification methods [9]–[11] or emphasized occlusion handling with part-based models [12], [13]. Yet, from looking at the recent literature works, it appears that there is a significant general trend in work on pedestrian detection. Huge feature pools and high-dimensional feature vectors are becoming increasingly popular, mainly because they yield reasonable performance through simple integration with classical classifiers, such as support vector machines (SVMs), and do not employ complicated models for the handling of variances in viewpoints, body parts, occlusions, or context.

Unfortunately, "There's no such thing as a free lunch." Approaches employing very high-dimensional features come at a price and pose a computational bottleneck in practice. In particular, they rely on the availability of powerful computers and GPU computation, particularly at training time. Addressing this problem, we aim at more *compact features* that require less memory and fewer computational costs yet guarantee reasonable and robust performance.

In this paper, we propose *compact features* that incorporate prior knowledge as to the appearance of the upright human body. Our approach is inspired by prior work on detecting objects of relatively low intraclass variability. In particular, histograms of oriented gradients (HOGs) [5] and cascaded Haar-like features [14] have become the *de facto* methods of choice in this area. However, we note that previous features are selected either by means of exhaustive searches over all possible variations [14] or by means of less exhaustive random sampling [15]. As an alternative, we propose a method that marks a middle ground; we design compact discriminative Haar-like features selected from a particular *template pool*, which reflects the statistics of the appearance of the pedestrian upright body shape.

Our previous findings about compact Haar-like features were published in [16]; in this paper, we provide more implementation details and additional experimental results, as well as deeper insights into these features. The procedure of our new pedestrian detector is shown in Fig. 1, in which we provide three major contributions.

Statistical Pedestrian Shape Model: From statistical gradient data, we find that upright walking pedestrians share a common visual appearance, particularly with respect to (w.r.t.) the geometry of the head and shoulder region of the body. We model pedestrian shapes in terms of three rectangles geared toward different body parts, i.e., head, upper body, and lower body.

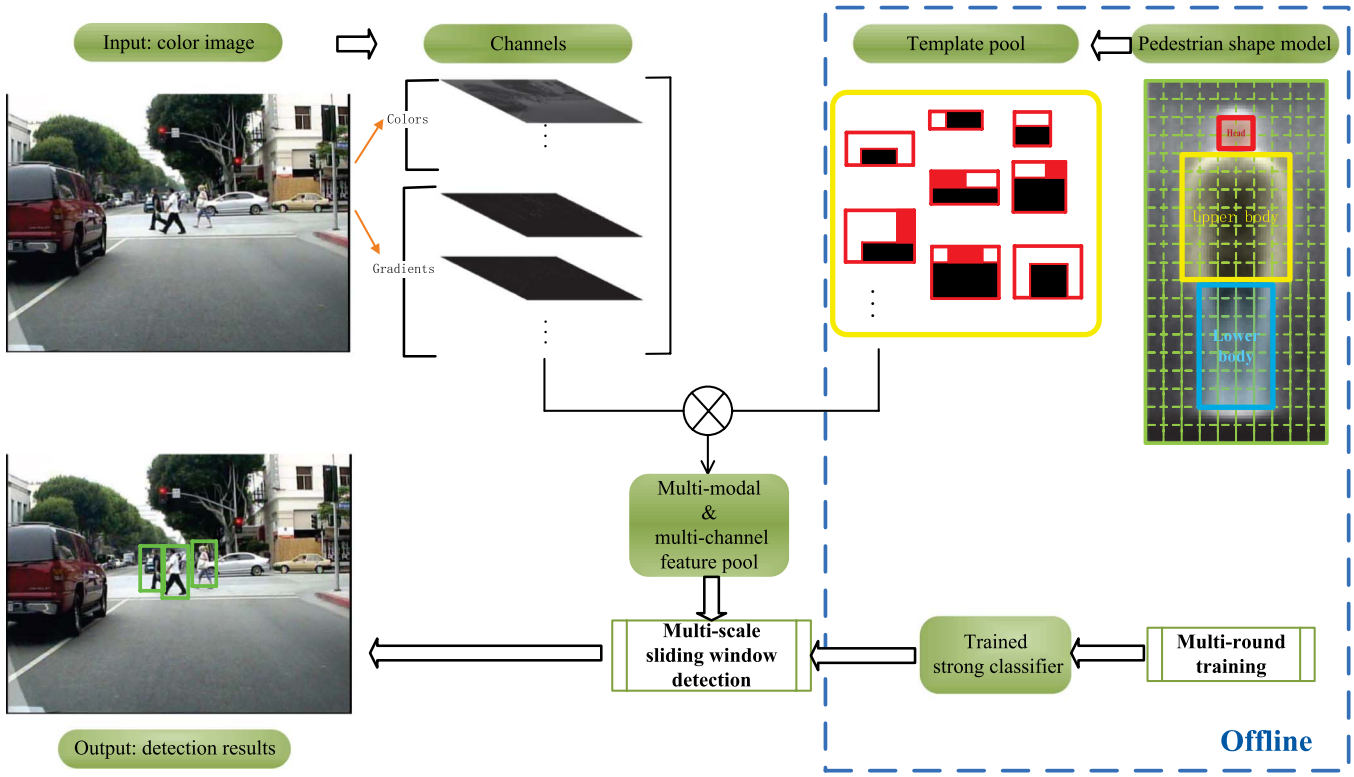


Fig. 1. Overview of our pedestrian detector. The dotted blue bounding box indicates the offline procedure.

Multimodal Haar-like Template Pool: Based on the pedestrian shape model, we design a pool of templates that are better tailored to pedestrian shapes 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; thus, they significantly reduce training times. We use two template modalities, i.e., binary and ternary, for Haar-like templates. The ternary modality is specifically proposed to represent corner regions found along the pedestrian silhouette to enable rectangle features to represent more complex geometric configurations.

Multichannel Haar-like Features: In order to incorporate rich information from image data, we consider rectangle descriptors not only w.r.t. colors but also w.r.t. gradients, yielding a multichannel Haar-like feature pool. This addresses challenges due to variations in the choice of clothes.

This paper contains an overview of related work in Section II, a description of our multimodal and multichannel features in Section III, and our feature selection scheme in Section IV. A thorough set of experiments is presented in Section V, in which the impact of different parameters is investigated and comparisons with state-of-the-art detectors from the literature are made. Afterward, we discuss several important issues regarding the feature design in Section VI. Finally, we summarize our contributions and findings and discuss several directions for future work in Section VII.

II. Related Work

Due to 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.

To start, the arguably most popular features for visual pedestrian detection are *HOGs*, as introduced in [5]. HOG features brought about significant improvements and therefore establish an important baseline. In order to improve performance, several researchers extended the feature pool by combining HOGs with other features. Wang and Han [HogLbp] [17] combined HOG features with a particular *local binary pattern* (LBP) feature in order to cope with partial occlusions; Liu *et al.* [18] introduced the idea of a granularity space, i.e., a family of descriptors ranging from edgelets to HOGs; Walk *et al.* [8] combined HOG features with self-similarity features related to color channels [MultiFtr+CSS] and motion features [MultiFtr+Motion] in order to better integrate spatial and temporal information. Other researchers aim at building stronger HOG detectors through integration with part-based models. Prioletti *et al.* [19] successfully applied the HOG detector for different body parts in verification stage, resulting in significant improvements.

Deviating from the popular framework of "HOG+SVM" computations, Dollár *et al.* [20] proposed another strong baseline [ChnFtrs], which applied integral channel features. At that time, [ChnFtrs] outperformed previous detectors significantly in terms of both detection accuracy and efficiency. An immediate extension of this approach has been called the "Fastest Pedestrian Detection in the West" [FPDW] [21] and was shown to enable real-time multiscale detection. Later, many new variants [22], [23] emerged, and several authors obtained even better performance by extending the feature pool in various

ways. Benenson *et al.* [Roerei] [24] used irregular rectangles resulting in a 718 080-dimensional feature pool; Lim *et al.* [SketchTokens] [25] added self-similarity features, yielding a 21 350-dimensional feature vector for image patches of a size of 35 pixels × 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.

Haar-like features became well known after Papageorgiou and Poggio [26] proposed a Haar wavelets-based system for object detection. The epitome of such approaches is found in the work by Viola and Jones [14] who used Haar-like features in combination with boosting algorithms to build a successful face detector. In fact, an early attempt of Haar wavelets for pedestrian detection can be found in [27] where it was demonstrated that wavelet templates can be used to define the shape of an object. Alonso *et al.* [28] evaluated Haar wavelets and other features, e.g., gradients and co-occurrence matrix to look for the most appropriate features for each body part. Unfortunately, Haar-like features, considered as second-order channel features, are not as successful as HOGs and 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 but may fail to capture visual characteristics of human body. As a remedy, we propose designing particularly tailored templates for upright body shapes.

III. Multimodal Multichannel Haar-Like Features

In this section, we describe our feature extraction procedure for visual pedestrian detection. First, *channels information* in terms of colors and gradients is computed from the input color images. In addition, a statistical pedestrian shape model is defined according to an average edge map. From this information, a template pool is generated based on the predefined pedestrian shape model. Finally, multimodal and multichannel Haar-like features are extracted by convolution between templates and each channel map.

A. Statistical Pedestrian Shape Model

Based on common sense and everyday knowledge, we assume that pedestrian bodies share common geometry structures and seek to corroborate this expectation based on empirical data. We choose the INRIA data set, which contains annotated image patches showing pedestrians scaled to a height of 96 pixels, and with 12 pixels padded in four directions to include contextual information. Consequently, we perform a statistical analysis on pedestrian images of size 60 pixels × 120 pixels. We compute an average edge map based on gradient magnitude extracted from each sample image, regardless of viewpoints or postures. The resulting average edge map is shown in Fig. 2 and clearly resembles a human body.

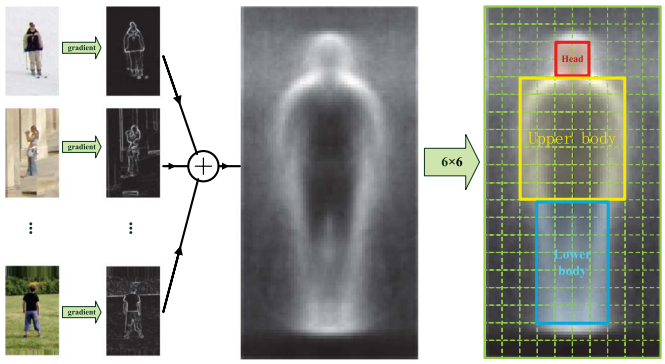


Fig. 2. Procedure of our statistical pedestrian shape model (rightmost) generation. We collect all the pedestrian sample images from the INRIA data set and compute an average edge map, as shown in the middle, which is divided by rectangular cells. In this example, cell size is chosen to be 6 pixels × 6 pixels. Three bounding boxes approximately indicate the head, the upper body, and the lower body parts.

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 an example of cells of size 6 pixels × 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.

This model is a statistical model because it is built on an average gradient magnitude map, which we computed from statistical data. The boundaries of each part are defined manually according to prior knowledge on human body parts, as well as the silhouette from the average gradient magnitude map. We vary those boundaries by choosing different cell sizes. In order to obtain the optimal model, we implement experiments with different cell sizes in Section V.

B. Multimodal Haar-Like Template Pool

In this section, we describe how a multimodal Haar-like template pool is generated based on the statistical pedestrian shape model discussed above.

We begin by explaining why we consider multimodal Haar-like templates rather than more involved features based on local histograms. In our following discussion, traditional Haar-like features are referred to as a binary modality because they only carry two possible weights (i.e., $+1$ and -1) for different rectangles. This binary modality is ill suited to represent cusps or cornerlike structures of the human silhouette. This is to say that it hardly adapts to the description of the content of bounding boxes that contain three different logical components such as the head, the upper body, and parts of the scene background. Yet, for efficient subsequent classification, we are interested in computing the difference between such parts w.r.t. two of them at a time. We therefore propose to consider ternary templates. An example is given in Fig. 3, in which ternary 2 × 2 templates capture the local geometry of the image region where head, shoulders, and background meet in joint corners.

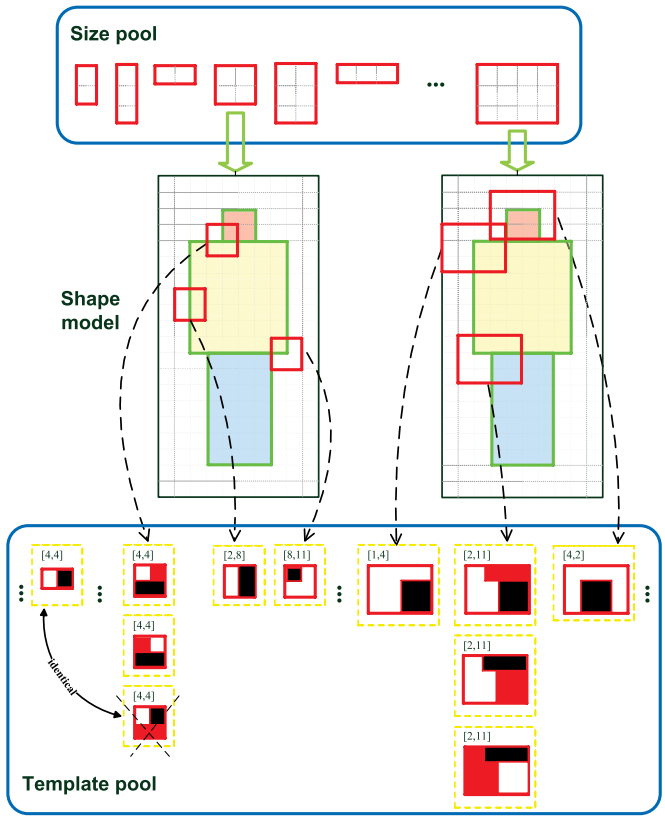


Fig. 3. Overview of our template pool generation procedure. Note that the number array above each template indicates the x, y coordinates of its left-top cell w.r.t. the shape model; those templates at [4, 4] and [2, 11] are ternary (shown as white, black, and red areas), which are given the weights of $+1$, -1, and 0, respectively. An example of redundancy removal is given. Two identical templates are found in the pool, then one of them is discarded.

An overview of our template pool generation procedure is shown in Fig. 3.

First, a size pool S is defined as

$$S = \left\{(w,h)|w \leq w_m,\ h \leq h_m, w, h \in \mathbb{N}^+\right\} \tag{1}$$

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. Note that 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.

Second, we assign a label to each cell based on the pedestrian shape model. 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 $c(i,j)$ exactly one label $L(i,j)$ that indicates which component is 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. At each position, we first decide the modality. If it is binary, then only one $\binom{2}{1}$ template is generated; otherwise, three $\binom{3}{2}$ templates are generated.

In the following, a template is denoted as $t(x,y,(w,h),W)$ or in a slightly simplified way as $t(x,y,s,W),\ s \in S$, where x and y indicate the location of a template w.r.t. the human shape model, w and h indicate the width and height of template w.r.t. cells, and W is a weight matrix that is determined according to the matrix L of labels for all cells.

Sometimes, templates like these may be redundant, as shown in the example in Fig. 3. At position [4, 4], the 2×1 template is identical to the third 2×1 template. The lower two cells of the 2×2 template are both assigned weight of 0. That is, only the upper two cells actually contribute to the feature response; thus, we can easily simplify it to a 2×1 template. Once another identical template is found in the template pool, the current template is discarded.

We develop a simple method to efficiently check for redundancy. Given two templates $t_1(x,y,(w_1,h_1),W_1)$ and $t_2(x,y,(w_2,h_2),W_2)$ at the same location (x,y), we define a maximal size $s_{\max}(w,h)$ as

$$\begin{cases} w = \max(w_1, w_2) \\ h = \max(h_1, h_2). \end{cases} \tag{2}$$

Then, we expand two weight matrices to $s_{\max}(w,h)$ by filling blanks with weights of 0. Next, we compute the difference between two new weight matrices W_1' and W_2'

$$W_d = W_1' - W_2'. \tag{3}$$

Templates t_1 and t_2 are considered to be identical if and only if all the elements of W_d are zero.

To cope with individual differences, each template is shifted along four directions with a step of one cell, resulting in a larger template pool. Therefore, for each template $t(x,y,s,W)$, the original template and a group of shifted templates are added to the template pool. We denote this template group as

$$\begin{cases} t(x,y,s,W) \\ t_L(x-1,y,s,W) \\ t_R(x+1,y,s,W) \\ t_U(x,y-1,s,W) \\ t_D(x,y+1,s,W). \end{cases} \tag{4}$$

Notably, some templates at the border of a training image patch cannot be extended by means of shifting.

Finally, the full template pool after redundancy removal and shifting is given as a set of templates of various sizes, with two modalities and at different positions

$$T = \left\{(x,y,s,W)|x,y \in \mathbb{N},\ s \in S,\ W \in \mathbb{R}^2\right\} \tag{5}$$

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.

Algorithm 1 Generating templates for pedestrian shape model through sliding rectangles

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**
      $W(label == l_1) \leftarrow -1$;
      $W(label == l_2) \leftarrow 1$;
      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

C. Multichannel Cell Descriptor

To integrate color and gradient information, we build a multichannel descriptor for each cell. We refer to the settings in detector [ChnFtrs], which are also commonly used in various approaches known from the literature.

A total of 10 different channels are used following the suggestions in [20]: three channels for LUV colors, one channel for gradient magnitude information, and six channels for HOGs. The authors of [20] also report that pre-smoothing with a binomial filter [29] of radius 1, i.e., $\sigma \approx 0.87$, improved the performance, whereas post-smoothing on channel values had little effect on performance.

Details as to how we choose the channels and on the impact of pre-smoothing on image data and post-smooth on channel values are discussed in Section V-C, in which performances under different parameter settings are compared. Although some parameters on channel features have been previously discussed in publications on detector [ChnFtrs], we further contribute new insights since our features consider local differences rather than absolute values.

D. Feature Matrix

Assume we are given a template $t = (x, y, (w, h), W)$. We normalize the weight matrix W inside each template by first counting 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

$$W_{\text{avg}} = \frac{\text{sgn}(W)}{n_{\text{add}}} + \frac{\text{sgn}(-W)}{n_{\text{sub}}}. \tag{6}$$

Each template goes through multiple channels to yield a multichannel feature pool. Assume we have N_t templates in total and consider N_c channels, a $N_t \times N_c$ feature matrix is generated as our final feature pool $\vec{f}$. The feature value of any template $t(t < N_t)$ for any channel $k(k < N_c)$, e.g., color or gradient information, can be then computed as a weighted sum

$$\vec{f}(t, k) = \sum_{i=1}^{h} \sum_{j=1}^{w} \sigma(x + i,\ y + j, k) W_{\text{avg}}(i, j) \tag{7}$$

where $\sigma(i, j, k)$ denotes the sum of values in $cell(i, j)$ along channel k, which can be computed very efficiently using integral images.

IV. Selecting Features for Pedestrian Detection

Our detector employs the multimodal and multichannel Haar-like features proposed in Section III. Note that these features are built on channel features ([ChnFtrs [20]) but interpret local differences between rectangular regions over multiple channels rather than over channel values themselves.

We present our feature size in the following. Given 6×6 cells and templates size ranging from 1×2 to 4×3 cells, we obtain 266 templates at different positions. Shifting templates along four 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. Considering this amount of features, we choose a fast version of AdaBoost [30] for learning since it offers a convenient and fast approach to select from a large number of candidate features.

As in any boosting algorithm, the final strong classifier is built from a collection of weak classifiers. We use decision trees of depth 2 as our weak classifiers and choose the number of weak classifiers to be 2000. Similar to classic detectors [5], [20], we also employ a multiround training strategy that has been shown to lead to a better performance than a simple one-round training procedure with the same number of negative samples. For the first round, initial negative training samples are randomly cropped from the negative example images; in the following rounds, hard negative samples are searched using the classifier built in the previous round, over all negative example images. This procedure is iterated until no significant performance gains are observed with further retraining. From our experiments, three rounds of retraining were observed to yield optimal performance; additional rounds did not show significant improvements. An illustration of how performance gains at each training round on the INRIA data set can be found in Fig. 4. We collect 5000 negative samples at each round, resulting in a large negative sample pool of 20 000.

In order to look into which features are more informative, we plot a weight image of the top 100 features, with highest weights from the final strong classifier, as shown in Fig. 5. To generate this figure, we added the weight of each selected feature to the cells it covers and used 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.

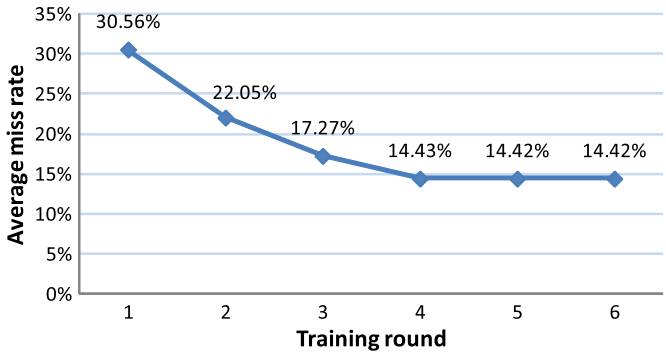


Fig. 4. Illustration of how performance gains at each training round on the INRIA data set. After four-round training, additional rounds do not show significant improvements.

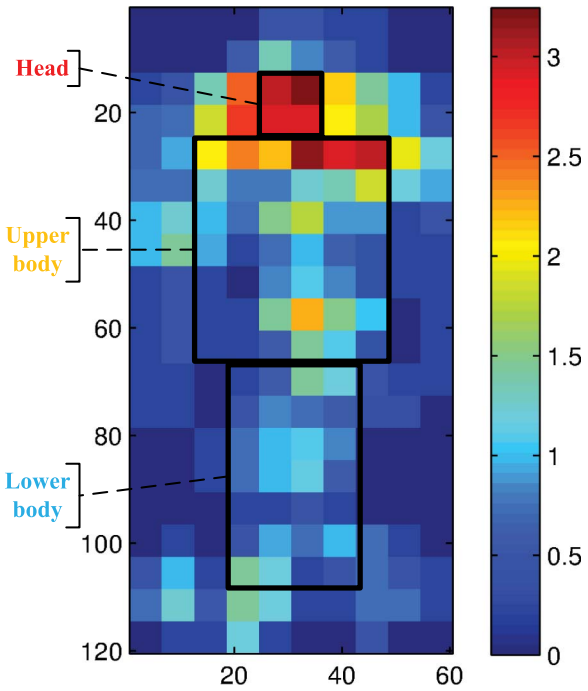


Fig. 5. Illustration of representative features. Different colors are used to indicate the accumulative weight of each cell after boosting. Three black bounding boxes indicate three body parts respectively. The head–shoulder area shows to be more discriminative for pedestrian detection than other body parts.

The most discriminative features 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 nonmaximal suppression procedure [20] to suppress nearby detections.

V. Experiments

In this section, we describe the benchmark data sets and evaluation protocol used in our experiments, discuss the impact of parameter settings on performance, compare our optimal detector to other state-of-the-art detectors, and provide an analysis on runtimes.

TABLE I
Statistics of Three Pedestrian Data Sets Used for Experiments

		INRIA [5]	Caltech [32]	KITTI-Train [31]
Properties	imaging setup	photo	mobile	mobile
	color images	√	√	√
	video seqs.	×	√	×
	occlusion labels	×	√	×
Training	# pedestrians	1208	192k	1800
	# pos. images	614	67k	3471
	# neg. images	1218	61k	3471
Testing	# pedestrians	566	155k	1962
	# pos. images	288	65k	3470
	# neg. images	453	56k	3470

A. Benchmark Data Sets

Experiments are conducted on three well-established public benchmark data sets (see Table I): the INRIA pedestrian data set [5], the Caltech pedestrian data set [3], and the KITTI-Train pedestrian data set [31]. Note that it is infeasible to run experiments on data sets that only consist of grayscale images, e.g., Daimler data set [4], because our approach uses three color channels.

INRIA Pedestrian Data Set[1]: This is arguably the most popular data set for people detection and comes along with predefined subsets for training and testing. For training, there are 2416 positive samples, by mirroring from 1208 different pedestrian images; there are 12 180 natural images, where no pedestrian appears, and negative samples can be selected by randomly cropping. For test, there are 288 positive samples and 453 negative samples. In consistency with conventions in this area, the test is only implemented on the positive samples.

Caltech Pedestrian Data Set[2]: This is currently the largest and most challenging data set for pedestrian detection, consisting of approximately 10 h of 640 × 480 30-Hz video taken from a vehicle driving through regular traffic in an urban environment. About 250 000 frames with a total of 350 000 bounding boxes and 2300 unique pedestrians were annotated. The training data (set00–set05) consist of six training sets, along with all annotation information (see [32] for details). The testing data (set06–set10) consist of five sets.

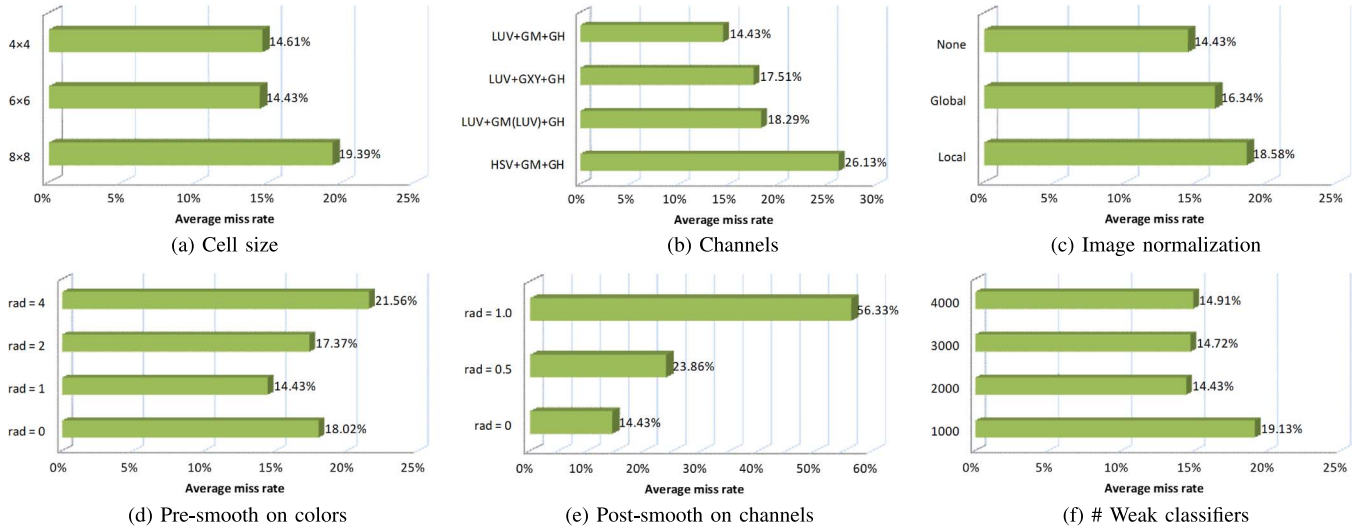
KITTI-Train Pedestrian Data Set[3]: This data set is captured by driving around the mid-size city of Karlsruhe, in rural areas and on highways. It consists of 7481 images and 3762 pedestrian annotations. We split the data set evenly into two parts, i.e., one for training and another for testing. The KITTI-Test data set is not considered because its ground truth annotations are not publicly available, resulting in that evaluations under our experimental settings are not allowed.

B. Evaluation Protocol

In the following, we explain details of our evaluation protocol in four aspects, which are consistent with the conventions in this field.

Ground Truth Filtering: For each experiment, a subset of all ground truth data is considered according to its specific

[1]http://pascal.inrialpes.fr/data/human
[2]http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians
[3]http://www.cvlibs.net/datasets/kitti/eval_object.php



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

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

Fig. 6. Evaluation of different parameters on the INRIA pedestrian data set. (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.

purpose. Outliers are marked with an *ignore* label, which means they need not be matched; however, matches are not considered as mistakes either. We specify four settings used in this paper.

1) *Reasonable*: Overall results in Fig. 7 are obtained under this setting. Pedestrians at a resolution of over 50 pixels in height and visibility of more than 65% are considered.
2) *No occlusion*: Pedestrians with 100% visibility are considered.
3) *Partial occlusion*: Pedestrians with more than 65% visibility are considered.
4) *Heavy occlusion*: Pedestrians with 20%–65% visibility are considered.

Detection Results Filtering: We filter out detection results using an expanded filtering method [3]; hence, detection results far outside the evaluation scale range should not be considered. When evaluating a scale range of $[S_1, S_2]$, only detections in $[S_1/\xi, S_2\xi]$ are considered for evaluation. In our evaluation, we set $\xi = 1.25$.

Bounding Box Matching Rules: Filtered ground truth bounding boxes and detection results bounding boxes are annotated by B_{gt} and B_{dt}, respectively. A detected bounding box and a ground truth bounding box match if and only if the ratio of overlap to the union of their areas exceeds a given threshold [3]

$$\mathrm{match}(B_{\mathrm{dt}}, B_{\mathrm{gt}}) = \frac{\mathrm{area}(B_{\mathrm{dt}}) \cap \mathrm{area}(B_{\mathrm{gt}})}{\mathrm{area}(B_{\mathrm{dt}}) \cup \mathrm{area}(B_{\mathrm{gt}})} \overset{!}{>} 0.5. \quad (8)$$

Performance Measurements: We perform full image evaluation instead of per-window evaluation as the former one provides a natural measure of error of an overall detection system. In order to compare different detectors, we plot miss rate against false positives per image (FPPI) curves in logarithmic scales by varying the threshold on the detection confidence of the classifiers. We only plot the curves in FPPI between $(-\infty, 10^0]$ as more than 10^0 FPPI is unacceptable for ADASs applications. In addition to this miss rate versus FPPI curves, we calculate a single numerical measurement to summarize detector performance. We use the *average miss rate* [3], which is computed by averaging the miss rate at nine FPPI rates evenly sampled in log-space in the range of $[10^{-2}, 10^0]$. This *average miss rate* generally gives a more stable and informative assessment of the overall performance for different detectors than the miss rate at only 10^{-1} FPPI according to [3].

C. Parameter Settings

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

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 pixels, 6×6 pixels, and 8×8 pixels. In Fig. 6(a), we find that a cell size of 6 pixels $\times$ 6 pixels produces the best results; hence, 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 [20], we only try alternatives for color and gradient magnitude channels. In Fig. 6(b), 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) rather than one maximal magnitude channel results in approximately 4% miss rate increase; using two gradient components (along the x- and y-directions, respectively) also leads to slight performance decrease. To specify, the optimal channel choice is to use LUV three color channels, plus with one maximal gradient magnitude channel and six gradient histogram channels.

Image Normalization: We analyze the influence of intensity normalization on our features as previous works on rectangular features typically employ various ways of normalization.

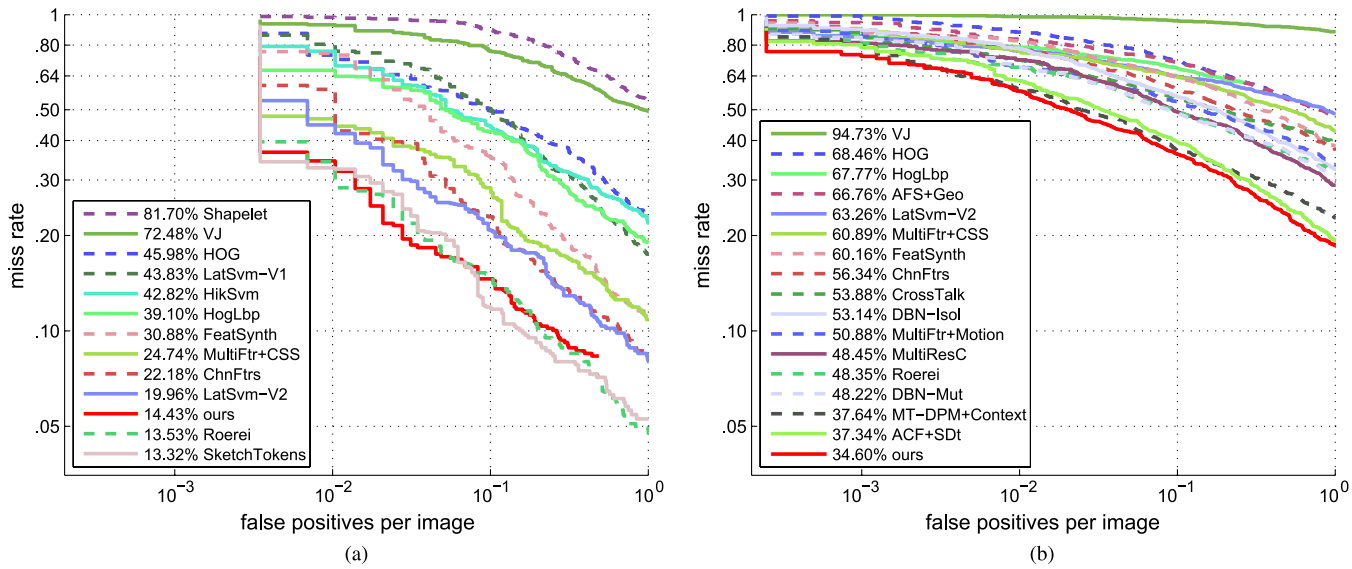


Fig. 7. Overall results of different detectors on the INRIA and Caltech data sets under standard evaluation settings. (a) INRIA. (b) Caltech test.

Viola and Jones [VJ] [14] used local normalization inside each detection window; [Roerei] [24] reported performance improvements by applying global normalization on the input images. However, according to the results in Fig. 6(c), our features obtain best results without image 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 is improved by approximately 5% when using 2000 rather than 1000 weak classifiers but 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; and 2000 weak classifiers.

D. Comparisons With State-of-the-Art Detectors

In this section, we compare the performance our detector to other state-of-the-art detectors whose results are publicly available,[4] using the experimental protocol explained in Section V-B.

The results in Fig. 7(a) 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 50 times larger and are about 100 times slower in training.

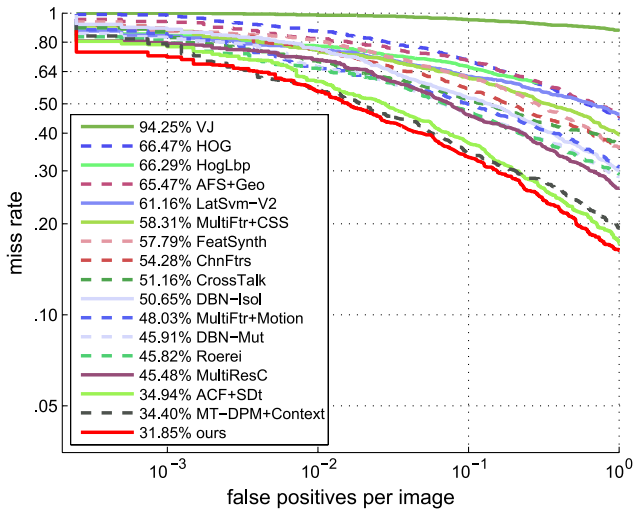
[4]http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians

On the Caltech pedestrian data set, our detector outperforms not only the baseline detector [ChnFtrs] by about 20% but also yields the overall best performance, as shown in Fig. 7(b). In particular, we note that it even outperforms detectors that consider additional motion information [8], [40]. We also show several detection examples of our detector under different scenarios from the Caltech pedestrian data set in Fig. 11.

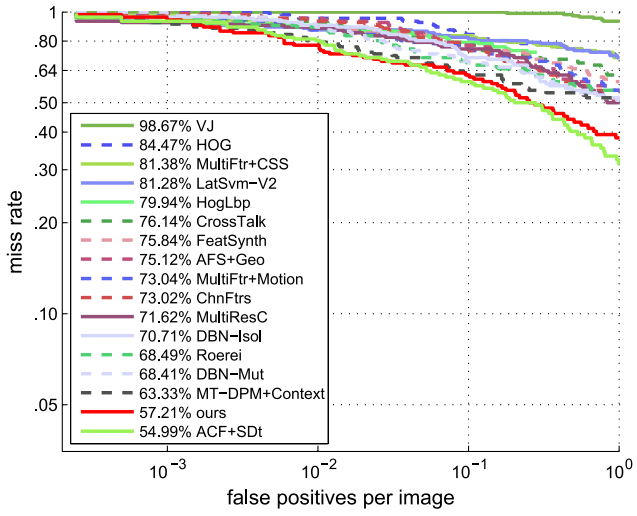
Fig. 8 shows evaluation results under different occlusion conditions for the Caltech pedestrian test data. As in [3], we use three occlusion levels: no occlusion (0% occluded), partial occlusion (1%–35% occluded), and heavy occlusion (35%–80% occluded). The performance of all the detectors significantly drops 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.

The KITTI-Train data set is considered as a more difficult data set, and experimental results are shown in Fig. 9. Unfortunately, we are not able to make extensive comparisons as on the INRIA and Caltech data set due to the lack of results from other state-of-the-art detectors. Only compared with our baseline detector [ChnFtrs], we notice that our approach obtains a significant improvement of around 18% in terms of average miss rate.

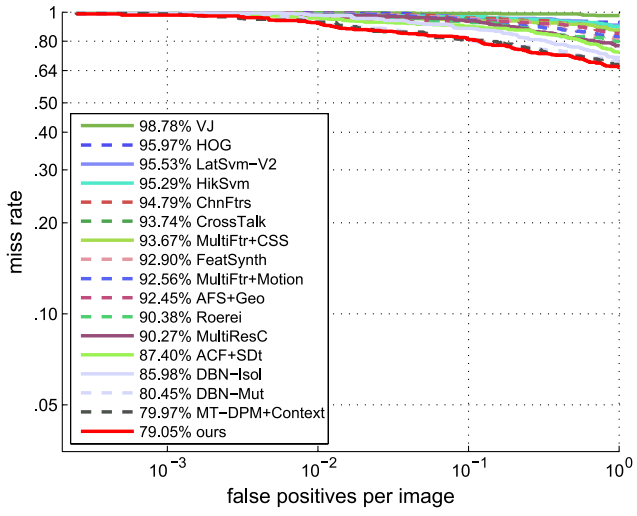
To provide a more comprehensive comparison among all the state-of-the-art detectors w.r.t. detector components and performance, we list detail information of each detector in Table II. First, as most detectors use HOG features or channel features in various forms, we group detectors into three categories: HOG-based, channel-based, and others, according to which kind of features they employ in major. Then we indicate which classifiers they use and whether they apply part-based models,



(a)



(b)



(c)

Fig. 8. Evaluation results under different occlusion conditions on the Caltech pedestrian test data set. (a) No occlusion. (b) Partial occlusion (1%–35% occluded). (c) Heavy occlusion (35%–80% occluded).

occlusion handling strategy, or motion information in the third to sixth column for all the detectors considered in this paper. In the last two columns, corresponding performance on the INRIA

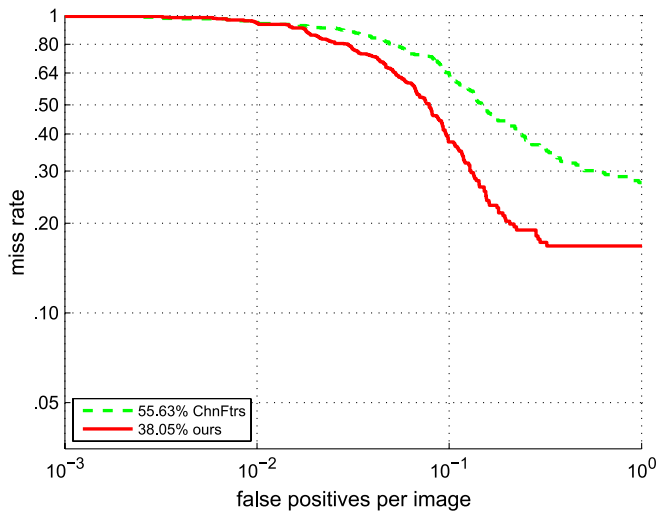


Fig. 9. Comparison to the baseline detector [ChnFtrs] on the KITTI-Train data set under standard evaluation settings.

and Caltech pedestrian data sets are demonstrated, respectively. We summarize our insights as follows.

1) HOG-based detectors are still in majority, whereas more recent detectors tend to employ channel features, which obtain better performance.
2) Most HOG-based detectors utilize SVMs as classifiers, whereas channel-based ones all use AdaBoost. This is because channel features are usually of higher dimensions, and AdaBoost is more efficient to select the most discriminative ones.
3) It is interesting that all the channel-based detectors do not employ part-based models or occlusion handling strategy; in contrast, more efforts have been explored for HOG-based detectors. Reasonable performance achieved by simple utilization of channel features implies that more promising results can be expected through integration with more complex models.
4) Over all the detectors, motion information is rarely used. One reason may be that it is computationally expensive to obtain accurate and dense optical flow maps, which directly describe the motion between successive frames. On the other hand, it is still an open problem about how to design motion-based features, which provide rather different information from colors or gradients.

E. Runtimes

We do not provide an exhaustive comparison of runtimes among state-of-the-art detectors in this paper because different detectors are implemented on different machines, some even heavily rely on GPU computations [22], [24]. It therefore does not make much sense to list runtimes from different computing architectures.

Our detector is implemented in MATLAB on an Intel Core-i7 CPU (3.5 GHz). On the Caltech data set, it takes 1 h for training with four rounds and 1.6 s ([ChnFtrs] 2 s) for testing a 640 $\times$ 480 image using the optimal parameters, as illustrated in Section V-C. In addition to channel computation, our feature computation includes local sums and differences, both of which

TABLE II
COMPREHENSIVE COMPARISONS FOR STATE-OF-THE-ART PEDESTRIAN DETECTORS. EACH ROW IN THIS TABLE SUMMARIZES INFORMATION AS TO CLASSIFIERS, PART-BASED MODELS, OCCLUSION HANDLING, AND MOTION INFORMATION USED IN A PARTICULAR APPROACH, AND DISPLAYS THE CORRESPONDING AVERAGE PERFORMANCE IN TERMS OF AVERAGE MISS RATES ON BOTH DATA SETS. THE APPROACH PROPOSED IN THIS PAPER YIELDS STATE-OF-THE-ART PERFORMANCE ON THE INRIA DATA SET AND CONSISTENTLY BETTER RESULTS THAN PREVIOUSLY REPORTED ON THE CALTECH DATA SET. WE ANNOTATE THE TOP THREE DETECTORS FOR EACH DATA SET WITH A ⋆ FOLLOWING EACH AVERAGE MISS RATE

Category	Detector	Classifier	Part-based	Occlusion handling	Motion	Average miss rate INRIA	Caltech
HOG-based	HOG [5]	linear SVM	×	×	×	45.98%	68.46%
	MultiFtr [33]	AdaBoost	×	×	×	36.50%	68.62%
	MultiFtr+CSS [8]	AdaBoost	×	×	×	24.74%	60.89%
	MultiFtr+Motion [8]	linear SVM	×	×	√	/	50.88%
	HikSvm [9]	HIK SVM	×	×	×	42.82%	73.39%
	HogLbp [17]	linear SVM	×	√	×	39.10%	67.77%
	LatSvm-V1 [12]	latent SVM	√	×	×	43.83%	79.78%
	LatSvm-V2 [13]	latent SVM	√	×	×	19.96%	63.26%
	FeatSynth [34]	linear SVM	√	×	×	30.88%	60.16%
	MultiResC [35]	latent SVM	×	×	×	/	48.45%
	AFS+Geo [36]	linear SVM	√	×	×	/	66.76%
	MT-DPM+Context [37]	latent SVM	√	×	×	/	37.64%★
	DBN-Isol [38]	DeepNet	√	√	×	/	53.14%
	DBN-Mut [39]	DeepNet	√	√	×	/	48.22%
Channel-based	ChnFtrs [20]	AdaBoost	×	×	×	22.18%	56.34%
	CrossTalk [23]	AdaBoost	×	×	×	18.98%	53.88%
	VeryFast [22]	AdaBoost	×	×	×	15.96%	/
	SketchTokens [25]	AdaBoost	×	×	×	13.32%★	/
	Roerei [24]	AdaBoost	×	×	×	13.53%★	48.35%
	ACF+SDt [40]	AdaBoost	×	×	√	/	37.34%★
	ours	AdaBoost	×	×	×	14.43%★	34.60%★
Others	VJ [14]	AdaBoost	×	×	×	72.48%	94.73%
	Shapelet [41]	AdaBoost	×	×	×	81.70%	91.37%

can be parallelized for further speedup. Our detector is expected to reach real-time efficiency running on a powerful machine and with GPU computation enabled.

VI. DISCUSSIONS

In this section, we discuss several important issues regarding our feature pool and performance.

Compactness of Features: We aim to design compact features; hence, we do not gather all random features but carefully design a relatively small feature pool based on a statistical shape model. Compared with recent state-of-the-art detectors, e.g., [Roerei] [24] and [SketchTokens] [25], our feature pool (12 760 dimensional) is more than 50 times smaller. We obtain competitive results on the INRIA data set and consistently better results on the Caltech data set with a much smaller feature pool, thus the compactness of our features.

Generality Versus Specificity: As the pedestrian body shape shown in Fig. 2 looks like from the front or back views, readers may argue that how our features adapt to crossing pedestrians, which are an important concern for safety and show quite different shapes from the average pedestrian body shape in Fig. 2. In fact, our features are largely invariant against viewpoints or postures. We provide two evidences from our experiments. The first one is that, regardless of viewpoints, the most informative features selected by our classifier are always found in the head–shoulder area (see Fig. 5), where minimal variance exists w.r.t. viewpoints; moreover, our detector achieves stable performance under different occlusion levels (see Fig. 8). From our observation, occlusions happen more at lower body (see Fig. 10), yet relatively rare in the head–shoulder area. Our stable performance approves that features from the lower body

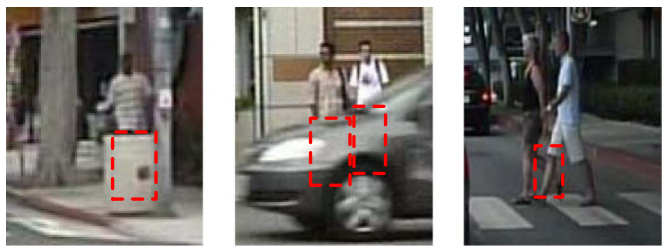


Fig. 10. Three examples of occlusions happen at lower body. Example images are from the Caltech pedestrian data set, and red dashed bounding boxes indicate the occluded part for pedestrians. In the given examples, lower body parts are occluded by a dustbin, a moving car, and another pedestrian, respectively.

are automatically ranked as less informative so that different limb postures do not have a negative effect.

First- Versus Second-Order Channel Features: Readers may expect better performance while including the first-order features to the final feature pool, as they describe the uniform texture inside each body part. However, these features were excluded because we found them to slightly decrease the performance and we assume them to be redundant. As an ensemble, our templates cover the whole body after shifting and characteristics such as uniform texture on clothing can be represented as minimal differences on those templates only cover one body part.

VII. CONCLUSION

Pedestrians are very vulnerable in urban traffic environments; hence, pedestrian protection based on visual sensing forms a prominent component of ADASs. The key problem of a pedestrian protection system is apparently to detect pedestrians when they are still in a safe distance to ensure enough time for

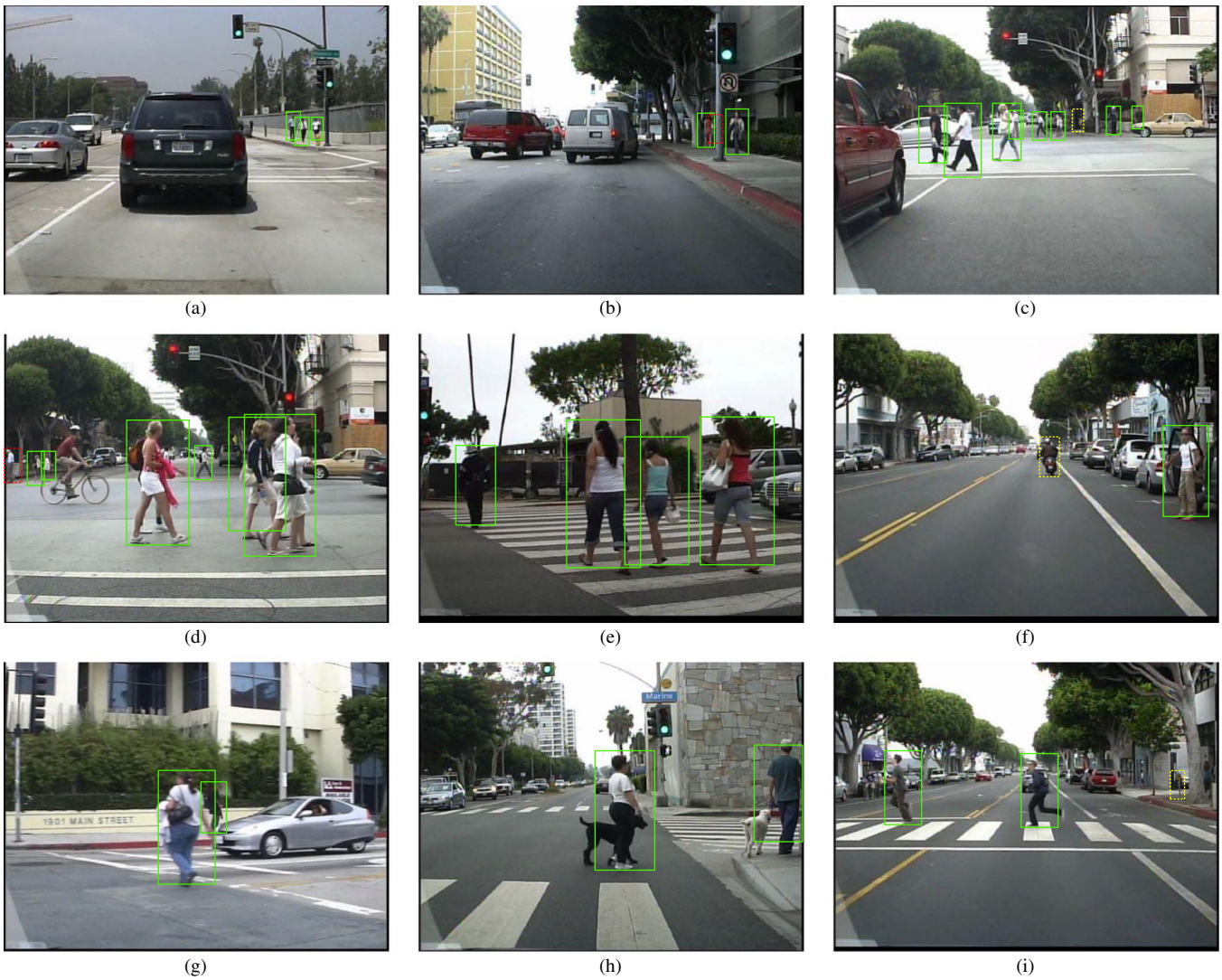


Fig. 11. Detection examples of our detector under different scenarios from the Caltech pedestrian data set. Green solid bounding boxes, yellow dotted bounding boxes, and red dotted bounding boxes indicate true positive, false positive, and false negative (missed) results, respectively. (a) Small-scale pedestrians walking along the street. (b) One missed pedestrian due to heavy occlusion ($> 70\%$). (c) Complex scenario at one intersection with one false positive occurring at one tree. (d) Pedestrians with occlusions. (e) Multiple pedestrians walking across the street. (f) One motorcyclist falsely detected as a pedestrian. (g) One pedestrian of low contrast. (h) Pedestrians with pets. (i) One traffic sign falsely detected as a pedestrian.

the driver or the vehicle to act for collision avoidance. Vision-based pedestrian detection is an effective and efficient way to detect pedestrians from on-board video data.

In this paper, the particular approach we have presented 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.

In addition, 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 accuracy. We therefore explored more compact features that could yield state-of-the-art performance in pedestrian detection if they were designed based on prior information as to the appearance of the upright human body.

Given a large data set 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 multimodal and multichannel 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 data sets, we found our detector to achieve state-of-the-art performance on the INRIA pedestrian data set; for the Caltech pedestrian data set, we found it to outperform all other recent approaches considered in our tests (see Fig. 11 for several challenging

detection examples); for the more challenging KITTI data set, our detector also obtains a significant improvement compared with the baseline detector. 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. In addition, we see more challenging extensions, e.g., to define multiple shape models w.r.t. parts or viewpoints for particular object classes, thus enabling more shape-variant object detection.

REFERENCES

[1] T. Gandhi and M. Trivedi, "Pedestrian protection systems: Issues, survey, challenges," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 413–430, Sep. 2007.

[2] J. Zhang, B. Tan, F. Sha, and L. He, "Predicting pedestrian counts in crowded scenes with rich and high-dimensional features," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1037–1046, Dec. 2011.

[3] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 743–761, Apr. 2011.

[4] M. Enzweiler and D. M. Gavrila, "Monocular pedestrian detection: Survey and experiments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2179–2195, Dec. 2009.

[5] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE CVPR*, 2005, pp. 886–893.

[6] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *Proc. ECCV*, 2006, pp. 428–441.

[7] Y.-F. Kao *et al.*, "Comparison of granules features for pedestrian detection," in *Proc. IEEE ITSC*, 2012, pp. 1777–1782.

[8] S. Walk, N. Majer, K. Schindler, and B. Schiele, "New features and insights for pedestrian detection," in *Proc. IEEE CVPR*, 2010, pp. 1030–1037.

[9] S. Maji, A. C. Berg, and J. Malik, "Classification using intersection kernel support vector machines is efficient," in *Proc. IEEE CVPR*, 2008, pp. 1–8.

[10] Q. Ye, J. Liang, and J. Jiao, "Pedestrian detection in video images via error correcting output code classification of manifold subclasses," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 193–202, Mar. 2012.

[11] L. Oliveira, U. Nunes, and P. Peixoto, "On exploration of classifier ensemble synergism in pedestrian detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 1, pp. 16–27, Mar. 2010.

[12] P. F. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Proc. IEEE CVPR*, 2008, pp. 1–8.

[13] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.

[14] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137–154, May 2004.

[15] J. Gall, A. Yao, N. Razavi, L. V. Gool, and V. Lempitsky, "Hough forests for object detection, tracking, action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 11, pp. 2188–2202, Nov. 2011.

[16] S. Zhang, C. Bauckhage, and A. B. Cremers, "Informed Haar-like features improve pedestrian detection," in *Proc. IEEE CVPR*, Columbus, OH, USA, 2014, pp. 947–954.

[17] X. Wang and T. X. Han, "An HOG-LBP human detector with partial occlusion handling," in *Proc. IEEE ICCV*, 2009, pp. 32–39.

[18] Y. Liu, S. Shan, W. Zhang, X. Chen, and W. Gao, "Granularity-tunable gradients partition descriptors for human detection," in *Proc. IEEE CVPR*, 2009, pp. 1255–1262.

[19] A. Prioletti *et al.*, "Part-based pedestrian detection and feature-based tracking for driver assistance: Real-time, robust algorithms, evaluation," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1346–1359, Sep. 2013.

[20] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel features," in *Proc. BMVC*, 2009, pp. 1–11.

[21] P. Dollár and P. Perona, "The fastest pedestrian detector in the west," in *Proc. BMVC*, 2010, pp. 1–11.

[22] R. Benenson, M. Mathias, R. Timofte, and L. V. Gool, "Pedestrian detection at 100 frames per second," in *Proc. IEEE CVPR*, 2012, pp. 2903–2910.

[23] P. Dollár, R. Appel, and W. Kienzle, "Crosstalk cascades for frame-rate pedestrian detection," in *Proc. ECCV*, 2012, pp. 645–659.

[24] R. Benenson, M. Mathias, T. Tuytelaars, and L. V. Gool, "Seeking the strongest rigid detector," in *Proc. IEEE CVPR*, 2013, pp. 3666–3673.

[25] J. J. Lim, C. L. Zitnick, and P. Dollár, "Sketch tokens: A learned midlevel representation for contour and object detection," in *Proc. IEEE CVPR*, 2013, pp. 3158–3165.

[26] C. Papageorgiou and T. Poggio, "A trainable system for object detection," *Int. J. Comput. Vis.*, vol. 38, no. 1, pp. 15–33, Jun. 2000.

[27] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio, "Pedestrian detection using wavelet templates," in *Proc. IEEE CVPR*, 1997, pp. 193–199.

[28] I. Alonso *et al.*, "Combination of feature extraction methods for SVM pedestrian detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 2, pp. 292–307, Jun. 2007.

[29] R. Haddad, "A class of orthogonal nonrecursive binomial filter," *IEEE Trans. Audio Electroacoust.*, vol. AE-19, no. 3, pp. 296–304, Dec. 1971.

[30] R. Appel, T. Fuchs, P. Dollár, and P. Perona, "Quickly boosting decision trees-pruning underachieving features early," in *Proc. ICML*, 2013, pp. 594–602.

[31] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE CVPR*, 2012, pp. 3354–3361.

[32] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," in *Proc. IEEE CVPR*, 2009, pp. 304–311.

[33] C. Wojek and B. Schiele, "A performance evaluation of single and multi-feature people detection," in *Proc. DAGM*, 2008, pp. 82–91.

[34] A. Bar-Hillel, D. Levi, E. Krupka, and C. Goldberg, "Part-based feature synthesis for human detection," in *Proc. ECCV*, 2010, pp. 127–142.

[35] D. Park, D. Ramanan, and C. Fowlkes, "Multiresolution models for object detection," in *Proc. ECCV*, 2010, pp. 241–254.

[36] D. Levi, S. Silberstein, and A. Bar-Hillel, "Fast multiple-part based object detection using KD-Ferns," in *Proc. IEEE CVPR*, 2013, pp. 947–954.

[37] J. Yan, X. Zhang, Z. Lei, S. Liao, and S. Z. Li, "Robust multi-resolution pedestrian detection in traffic scenes," in *Proc. IEEE CVPR*, 2013, pp. 3033–3040.

[38] W. Ouyang and X. Wang, "A discriminative deep model for pedestrian detection with occlusion handling," in *Proc. IEEE CVPR*, 2012, pp. 3258–3265.

[39] W. Ouyang, X. Zeng, and X. Wang, "Modeling mutual visibility relationship with a deep model in pedestrian detection," in *Proc. IEEE CVPR*, 2013, pp. 3222–3229.

[40] D. Park, C. L. Zitnick, D. Ramanan, and P. Dollár, "Exploring weak stabilization for motion feature extraction," in *Proc. IEEE CVPR*, 2013, pp. 2882–2889.

[41] P. Sabzmeydani and G. Mori, "Detecting pedestrians by learning shapelet features," in *Proc. IEEE CVPR*, 2007, pp. 1–8.

Shanshan Zhang (S'14) received the M.Sc. degree in signal and information processing from Tongji University, Shanghai, China, in 2011. She is currently working toward the Ph.D. degree with the Intelligent Vision Systems Group, University of Bonn, Bonn, Germany.

She was a Visiting Researcher with the Multimedia Information Research Division, National Institute of Informatics, Tokyo, Japan. Her main research interests include computer vision and pattern recognition and their applications in intelligent transportation systems.

Christian Bauckhage (M'02) received the M.Sc. and Ph.D. degrees in computer science from Bielefeld University, Bielefeld, Germany, in 1998 and 2002, respectively.

He is currently a Professor of media informatics and pattern recognition with University of Bonn, Bonn, Germany, and a Lead Scientist for media engineering with Fraunhofer Institute for Intelligent Analysis and Information Systems, Sankt Augustin, Germany. Prior to his current position, he was a Postdoctoral Researcher with the Centre for Vision Research, Toronto, ON, Canada, and a Senior Research Scientist with Deutsche Telekom Laboratories, Berlin, Germany, where he conducted and coordinated industrial information and communication technology research. His research focuses on large-scale descriptive data mining and efficient statistical machine learning for multimedia applications. In these areas, he regularly publishes conference papers and journal articles and frequently serves on program committees and editorial boards.

Armin B. Cremers received the Doctoral degree in mathematics from University of Karlsruhe, Karlsruhe, Germany, in 1972.

He has been with the computer science faculties of University of Southern California, Los Angeles, CA, USA, and University of Dortmund, Dortmund, Germany. Since 1990 he has been with the Department of Computer Science, University of Bonn, Bonn, Germany, where he is the Head of the Artificial Intelligence/Intelligent Vision Systems Research Groups. In 2002 he became the Founding Director of Bonn–Aachen International Center for Information Technology (B-IT), Bonn, jointly with RWTH Aachen University, Aachen, Germany, and Fraunhofer Society.