Monocular Pedestrian Detection: Survey and Experiments

Markus Enzweiler, *Student Member*, *IEEE*, and Dariu M. Gavrila

Abstract—Pedestrian detection is a rapidly evolving area in computer vision with key applications in intelligent vehicles, surveillance, and advanced robotics. The objective of this paper is to provide an overview of the current state of the art from both methodological and experimental perspectives. The first part of the paper consists of a survey. We cover the main components of a pedestrian detection system and the underlying models. The second (and larger) part of the paper contains a corresponding experimental study. We consider a diverse set of state-of-the-art systems: wavelet-based AdaBoost cascade [74], HOG/linSVM [11], NN/LRF [75], and combined shape-texture detection [23]. Experiments are performed on an extensive data set captured onboard a vehicle driving through urban environment. The data set includes many thousands of training samples as well as a 27-minute test sequence involving more than 20,000 images with annotated pedestrian locations. We consider a generic evaluation setting and one specific to pedestrian detection onboard a vehicle. Results indicate a clear advantage of HOG/linSVM at higher image resolutions and lower processing speeds, and a superiority of the wavelet-based AdaBoost cascade approach at lower image resolutions and (near) real-time processing speeds. The data set (8.5 GB) is made public for benchmarking purposes.

Index Terms—Pedestrian detection, survey, performance analysis, benchmarking.

1 Introduction

Finding people in images is a key ability for a variety of important applications. In this paper, we are concerned with those applications where the human body to be detected covers a smaller portion of the image, i.e., is visible at lower resolution. This covers outdoor settings such as surveillance, where a camera is watching down onto a street, or intelligent vehicles, where an onboard camera watches the road ahead of possible collisions with pedestrians. It also applies to indoor settings such as a robot detecting a human walking down the hall. Hence our use of the term "pedestrian" in the remainder of the paper, rather than the more general "people" or "person." We do not consider more detailed perception tasks such as human pose recovery or activity recognition.

Pedestrian detection is a difficult task from a machine vision perspective. The lack of explicit models leads to the use of machine learning techniques, where an implicit representation is learned from examples. As such, it is an instantiation of the multiclass object categorization problem (e.g., [79]). Yet the pedestrian detection task has some of its own characteristics, which can influence the methods of choice. Foremost, there is the wide range of possible pedestrian appearance, due to changing articulated pose, clothing, lighting, and background. The detection component is typically part of a system situated in a physical environment, which means that prior scene knowledge (camera calibration, ground plane constraint) is often available to improve performance. Comparatively large efforts have been spent to collect extensive databases; this study, for example, benefits from the availability of many thousands of samples. On the other hand, the bar regarding performance and processing speed lies much higher, as we will see later.

Pedestrian detection has attracted an extensive amount of interest from the computer vision community over the past few years. Many techniques have been proposed in terms of features, models, and general architectures. The picture is increasingly blurred on the experimental side. Reported performances differ by up to several orders of magnitude (e.g., within the same study [74] or [39] versus [74]). This stems from the different types of image data used (degree of background change), the limited size of the test data sets, and the different (often, not fully specified) evaluation criteria such as localization tolerance, coverage area, etc.

This paper aims to increase visibility by providing a common point of reference from both methodological and experimental perspectives. To that effect, the first part of the paper consists of a survey, covering the main components of a pedestrian detection system: hypothesis generation (ROI selection), classification (model matching), and tracking.

The second part of the paper contains a corresponding experimental study. We evaluate a diverse set of state-of-the-art systems with identical test criteria and data sets as follows:

- Haar wavelet-based AdaBoost cascade [74];
- histogram of oriented gradient (HOG) features combined with a linear SVM [11];

- *M. Enzweiler is with the Department of Mathematics and Computer Science, Image and Pattern Analysis Group, University of Heidelberg, Speyerer St. 4, 69115 Heidelberg, Germany. E-mail: uni-heidelberg.enzweiler@daimler.com.*
- *D.M. Gavrila is with the Environment Perception Department, Assistance Systems & Chassis, Daimler AG Group Research, Wilhelm Runge St. 11, 89081 Ulm, Germany, and the Intelligent Systems Lab, Faculty of Science, University of Amsterdam, Kruislaan 403, 1098 SJ Amsterdam, The Netherlands. E-mail: dariu.gavrila@daimler.com.*

Manuscript received 18 Jan. 2008; revised 14 July 2008; accepted 8 Oct. 2008; published online 17 Oct. 2008.
Recommended for acceptance by T. Darrell.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2008-01-0039.
Digital Object Identifier no. 10.1109/TPAMI.2008.260.

0162-8828/09/$25.00 © 2009 IEEE Published by the IEEE Computer Society

TABLE 1
Overview of Publicly Available Pedestrian Data Sets with Ground-Truth

Dataset	Training Set Pedestrian / Non-Pedestrian	Test Set Pedestrian / Non-Pedestrian	Comments
MIT CBCL Pedestrian Database [46]	924 / 0 (cut-outs), no separation into training and test images		single images, frontal and back views only
INRIA Person Dataset [28]	2416 (cut-outs) / 1218 (full images)	1132 (cut-outs) / 453 (full images)	single images (color)
Mobile Scene Analysis Dataset [16]	490 (full images), 1578 ped. labels	1803 (full images), 9380 ped. labels	camera at walking speed (stroller on urban sidewalks)
PETS Datasets (2001, 2003, 2004) [54]	-	2688, 2500, 13112 (full images)	16 image sequences from static cameras
DaimlerChrysler Pedestrian Classification Benchmark [49]	14400 / 15000 (cut-outs) + 1200 (full images)	9600 / 10000 (cut-outs)	single images
Daimler Pedestrian Detection Benchmark (current paper)	**15660 (cut-outs) / 6744 (full images)**	**21790 (full images), 56492 labels: 14132 fully visible ped. labels in 259 trajectories, 37236 partial ped. labels, 5124 other labels (bicyclists, motorcyclists, etc.)**	**test set corresponds to a 27 min drive through urban traffic**

- neural network using local receptive fields (NN/LRF) [75]; and
- combined hierarchical shape matching and texture-based NN/LRF classification [23].

In terms of evaluation, we consider both a generic and an application-specific test scenario. The generic test scenario is meant to evaluate the inherent potential of a pedestrian detection method. It incorporates no prior scene knowledge as it uses a simple 2D bounding box overlap criterion for matching. Furthermore, it places no constraints on allowable processing times (apart from practical feasibility). The application-specific test scenario focuses on the case of pedestrian detection from a moving vehicle, where knowledge about camera calibration, location of the ground plane, and sensible sensor coverage areas provide regions of interest. Evaluation takes place in 3D in a coordinate system relative to the vehicle. Furthermore, we place upper bounds on allowable processing times (250 ms versus 2.5 s per frame). In both scenarios, we list detection performance both at the frame and trajectory levels.

The data set is truly large-scale; it includes many tens of thousands of training samples as well as a test sequence consisting of 21,790 monocular images at 640×480 resolution, captured from a vehicle in a 27-minute drive through urban traffic. See Table 1. Compared to previous pedestrian data sets, the availability of sequential images means that also hypothesis generation and tracking components of pedestrian systems can be evaluated, unlike with [28], [46], [49]. Furthermore, the data set excels in complexity (dynamically changing background) and realism for the pedestrian protection application onboard vehicles.

The scope of this paper is significantly broader than our previous experimental study [49], which focused on pedestrian classification using low-resolution pedestrian and nonpedestrian cutouts (18×36 pixels). Here, we evaluate how robust and efficient pedestrians can be localized in image sequences in both generic and application-specific (vehicle) settings. Among the approaches considered, we include those that rely on coarse-to-fine image search strategies, e.g., see Section 4.4.

The remainder of this paper is organized as follows: Section 2 surveys the field of monocular pedestrian detection. After introducing our benchmark data set in Section 3, Section 4 describes the approaches selected for experimental evaluation. The result of the generic evaluation and the application-specific pedestrian detection from a moving vehicle are listed in Section 5. After discussing our results in Section 6, we conclude the paper in Section 7.

2 Survey

A number of related surveys exist, albeit with a different focus than ours. The authors of [21], [47], [57] cover methods for people detection, body pose estimation, and activity recognition. Gandhi and Trivedi [20] focus on the pedestrian protection application in the intelligent vehicle domain. They cover both passive and active safety techniques, the latter using (possibly) multiple vision and nonvision sensors, together with methods for collision risk assessment. We decompose pedestrian detection into the generation of initial object hypotheses (ROI selection), verification (classification), and temporal integration (tracking). While the latter two require models of the pedestrian class, e.g., in terms of geometry, appearance, or dynamics, the initial generation of regions of interest is usually based on more general low-level features or prior scene knowledge.

2.1 ROI Selection

The simplest technique to obtain initial object location hypotheses is the sliding window technique, where detector

windows at various scales and locations are shifted over the image. The computational costs are often too high to allow for real-time processing [11], [12], [48], [53], [60], [68]. Significant speedups can be obtained by either coupling the sliding window approach with a classifier cascade of increasing complexity [45], [52], [63], [71], [74], [76], [80], [83] or by restricting the search space based on known camera geometry and prior information about the target object class. These include application-specific constraints such as the flat-world assumption, ground-plane-based objects and common geometry of pedestrians, e.g., object height or aspect ratio [15], [23], [39], [50], [62], [82]. In case of a moving camera in a real-world environment, varying pitch can be handled by relaxing the scene constraints [23] or by estimating the 3D camera geometry online [39].

Other techniques to obtain initial object hypotheses employ features derived from the image data. Besides approaches using stereo vision [2], [7], [16], [23], [50], [81], which are out of the scope in this survey, object motion has been used as an early cueing mechanism. Surveillance approaches using static cameras often employ background subtraction [51], [66], [82]. Generalizations to moving cameras mostly assume translatory camera motion and compute the deviation of the observed optical flow from the expected ego-motion flow field [15], [56]. Another attention focusing strategy employs interest-point detectors to recover regions with high information content based on local discontinuities of the image brightness function that often occur at object boundaries [1], [39], [40], [42], [61].

2.2 Classification

After a set of initial object hypotheses has been acquired, further verification (classification) involves pedestrian appearance models, using various spatial and temporal cues. Following a rough categorization of such models into generative and discriminative models [72], we further introduce a delineation in terms of visual features and classification techniques. In both the generative and discriminative approaches to pedestrian classification, a given image (or a subregion thereof) is to be assigned to either the pedestrian or nonpedestrian class, depending on the corresponding class posterior probabilities. The main difference between generative and discriminative models is how posterior probabilities are estimated for each class.

2.2.1 Generative Models

Generative approaches to pedestrian classification model the appearance of the pedestrian class in terms of its class-conditional density function. In combination with the class priors, the posterior probability for the pedestrian class can be inferred using a Bayesian approach.

Shape models. Shape cues are particularly attractive because of their property of reducing variations in pedestrian appearance due to lighting or clothing. At this point, we omit discussion of complex 3D human shape models [21] and focus on 2D pedestrian shape models that are commonly learned from shape contour examples. In this regard, both discrete and continuous representations have been introduced to model the shape space.

Discrete approaches represent the shape manifold by a set of exemplar shapes [22], [23], [67], [70]. On one hand, exemplar-based models imply a high specificity since only plausible shape examples are included and changes of topology need not be explicitly modeled. On the other hand, such models require a large amount of example shapes (up to many thousands) to sufficiently cover the shape space due to transformations and intraclass variance. From a practical point of view, exemplar-based models have to strike a balance between specificity and compactness to be used in real-world applications, particularly with regard to storage constraints and feasible online matching. Efficient matching techniques based on distance-transforms have been combined with precomputed hierarchical structures, to allow for real-time online matching of many thousands of exemplars [22], [23], [67].

Continuous shape models involve a compact parametric representation of the class-conditional density, learned from a set of training shapes, given the existence of an appropriate manual [9], [25], [26] or automatic [4], [5], [14], [34], [50] shape registration method. Linear shape space representations which model the class-conditional density as a single Gaussian have been employed by Baumberg [4] and Bergtholdt et al. [9]. Forcing topologically diverse shapes (e.g., pedestrian with feet apart and with feet closed) into a single linear model may result in many intermediate model instantiations that are physically implausible. To recover physically plausible regions in the linear model space, conditional density models have been proposed [9], [14]. Further, nonlinear extensions have been introduced at the cost of requiring a larger number of training shapes to cope with the higher model complexity [9], [14], [25], [26], [50]. Rather than modeling the nonlinearity explicitly, most approaches break up the nonlinear shape space into piecewise linear patches. Techniques to determine these local subregions include fitting a mixture of Gaussians via the EM-algorithm [9] and K-means clustering in shape space [14], [25], [26], [50].

Compared to discrete shape models, continuous generative models can fill gaps in the shape representation using interpolation. However, online matching proves to be more complex since recovering an estimate of the maximum-a-posteriori model parameters involves iterative parameter estimation techniques, i.e., Active Contours [9], [50].

Recently, a two-layer statistical field model has been proposed to increase the robustness of shape representations to partial occlusions and background clutter by representing shapes as a distributed connected model [77]. Here, a hidden Markov field layer to capture the shape prior is combined with an observation layer, which associates shape with the likelihood of image observations.

Combined shape and texture models. One way to enrich the representation is to combine shape and texture information within a compound parametric appearance model [8], [9], [14], [17], [34]. These approaches involve separate statistical models for shape and intensity variations. A linear intensity model is built from shape-normalized examples guided by sparse [9], [14], [17] or dense correspondences [8], [34]. Model fitting requires joint estimation of shape and texture parameters using iterative error minimization schemes [17], [34]. To reduce the complexity of parameter estimation, the relation of the

fitting error and associated model parameters can be learned from examples [9].

2.2.2 Discriminative Models

In contrast to the generative models, discriminative models approximate the Bayesian maximum-a-posteriori decision by learning the parameters of a discriminant function (decision boundary) between the pedestrian and nonpedestrian classes from training examples. We will discuss the merits and drawbacks of several feature representations and continue with a review of classifier architectures and techniques to break down the complexity of the pedestrian class.

Features. Local filters operating on pixel intensities are a frequently used feature set [59]. Nonadaptive Haar wavelet features have been popularized by Papageorgiou and Poggio [53] and adapted by many others [48], [64], [74]. This overcomplete feature dictionary represents local intensity differences at various locations, scales, and orientations. Their simplicity and fast evaluation using integral images [41], [74] contributed to the popularity of Haar wavelet features. However, the many-times redundant representation, due to overlapping spatial shifts, requires mechanisms to select the most appropriate subset of features out of the vast amount of possible features. Initially, this selection was manually designed for the pedestrian class, by incorporating prior knowledge about the geometric configuration of the human body [48], [53], [64]. Later, automatic feature selection procedures, i.e., variants of AdaBoost [18], were employed to select the most discriminative feature subset [74].

The automatic extraction of a subset of nonadaptive features can be regarded as optimizing the features for the classification task. Likewise, the particular configuration of spatial features has been included in the actual optimization itself, yielding feature sets that adapt to the underlying data set during training. Such features are referred to as local receptive fields [19], [23], [49], [68], [75], in reference to neural structures in the human visual cortex [24]. Recent studies have empirically demonstrated the superiority of adaptive local receptive field features over nonadaptive Haar wavelet features with regard to pedestrian classification [49], [68].

Another class of local intensity-based features is codebook feature patches, extracted around interesting points in the image [1], [39], [40], [61]. A codebook of distinctive object feature patches along with geometrical relations is learned from training data followed by clustering in the space of feature patches to obtain a compact representation of the underlying pedestrian class. Based on this representation, feature vectors have been extracted including information about the presence and geometric relation of codebook patches [1], [39], [40], [61].

Others have focused on discontinuities in the image brightness function in terms of models of local edge structure. Well-normalized image gradient orientation histograms, computed over local image blocks, have become popular in both dense [11], [62], [63], [80], [83] (HOG, histograms of oriented gradients) and sparse representations [42] (SIFT, scale-invariant feature transform), where sparseness arises from preprocessing with an interest-point detector. Initially, dense gradient orientation histograms were computed using local image blocks at a single fixed scale [11], [62] to limit the dimensionality of the feature vector and computational costs. Extensions to variable-sized blocks have been presented in [63], [80], [83]. Results indicate a performance improvement over the original HOG approach. Recently, local spatial variation and correlation of gradient-based features have been encoded using covariance matrix descriptors which increase robustness toward illumination changes [71].

Yet others have designed local shape filters that explicitly incorporate the spatial configuration of salient edge-like structures. Multiscale features based on horizontal and vertical co-occurrence groups of dominant gradient orientation have been introduced by Mikolajczyk et al. [45]. Manually designed sets of edgelets, representing local line or curve segments, have been proposed to capture edge structure [76]. An extension to these predefined edgelet features has recently been introduced with regard to adapting the local edgelet features to the underlying image data [60]. So-called shapelet features are assembled from low-level oriented gradient responses using AdaBoost, to yield more discriminative local features. Again, variants of AdaBoost are frequently used to select the most discriminative subset of features.

As an extension to spatial features, spatiotemporal features have been proposed to capture human motion [12], [15], [65], [74], especially gait [27], [38], [56], [75]. For example, Haar wavelets and local shape filters have been extended to the temporal domain by incorporating intensity differences over time [65], [74]. Local receptive field features have been generalized to spatiotemporal receptive fields [27], [75]. HOGs have been extended to histograms of differential optical flow [12]. Several papers compared the performance of otherwise identical spatial and spatiotemporal features [12], [74] and reported superior performance of the latter at the drawback of requiring temporally aligned training samples.

Classifier architectures. Discriminative classification techniques aim at determining an optimal decision boundary between pattern classes in a feature space. Feed-forward multilayer neural networks [33] implement linear discriminant functions in the feature space in which input patterns have been mapped nonlinearly, e.g., by using the previously described feature sets. Optimality of the decision boundary is assessed by minimizing an error criterion with respect to the network parameters, i.e., mean squared error [33]. In the context of pedestrian detection, multilayer neural networks have been applied particularly in conjunction with adaptive local receptive field features as nonlinearities in the hidden network layer [19], [23], [49], [68], [75]. This architecture unifies feature extraction and classification within a single model.

Support Vector Machines (SVMs) [73] have evolved as a powerful tool to solve pattern classification problems. In contrast to neural networks, SVMs do not minimize some artificial error metric but maximize the margin of a linear decision boundary (hyperplane) to achieve maximum separation between the object classes. Regarding pedestrian classification, linear SVM classifiers have been used in

combination with various (nonlinear) feature sets [11], [12], [51], [63], [64], [80], [83].

Nonlinear SVM classification, e.g., using polynomial or radial basis function kernels as implicit mapping of the samples into a higher dimensional (and probably infinite) space, yielded further performance boosts. These are, however, paid for with a significant increase in computational costs and memory requirements [2], [48], [49], [51], [53], [68].

AdaBoost [18], which has been applied as automatic feature selection procedure (see above), has also been used to construct strong classifiers as weighted linear combinations of the selected weak classifiers, each involving a threshold on a single feature [60], [62]. To incorporate nonlinearities and speed up the classification process, boosted detector cascades have been introduced by Viola et al. [74] and adopted by many others [45], [52], [63], [71], [76], [80], [83]. Motivated by the fact that the majority of detection windows in an image are nonpedestrians, the cascade structure is tuned to detect almost all pedestrians while rejecting nonpedestrians as early as possible. AdaBoost is used in each layer to iteratively construct a strong classifier guided by user-specified performance criteria. During training, each layer is focused on the errors the previous layers make. As a result, the whole cascade consists of increasingly more complex detectors. This contributes to the high processing speed of the cascade approach, since usually only a few feature evaluations in the early cascade layers are necessary to quickly reject nonpedestrian examples.

Multipart representations. Besides introducing new feature sets and classification techniques, many recent pedestrian detection approaches attempt to break down the complex appearance of the pedestrian class into manageable subparts. First, a mixture-of-experts strategy establishes local pose-specific pedestrian clusters, followed by the training of a specialized expert classifier for each subspace [23], [51], [62], [64], [76], [80]. Appropriate pose-based clustering involves both manually [51], [62], [64], [76] and automatically established [80] mutually exclusive clusters, as well as soft clustering approaches using probabilistic assignment of pedestrian examples to pose clusters, obtained by a preprocessing step, e.g., shape matching [23].

An additional issue in mixture-of-experts architectures is how to integrate the individual expert responses to a final decision. Usually, all experts are run in parallel, where the final decision is obtained as a combination of local expert responses using techniques such as maximum selection [51], [76], majority voting [64], AdaBoost [62], trajectory-based data association [80], and probabilistic shape-based weighting [23].

Second, component-based approaches decompose pedestrian appearance into parts. These parts are either semantically motivated (body parts such as head, torso, and legs) [2], [45], [48], [62], [65], [76] or concern codebook representations [1], [39], [40], [61]. A general trade-off is involved at the choice of the number and selection of the individual parts. On one hand, components should have as small spatial extent as possible, to succinctly capture articulated motion. On the other hand, components should have sufficiently large spatial extent to contain discriminative visual structure to allow reliable detection. Part-based approaches require assembly techniques to integrate the local part responses to a final detection, constrained by spatial relations among the parts.

Approaches using partitions into semantic subregions train a discriminative feature-based classifier (see above), specific to a single part, along with a model for geometric relations between parts. Techniques to assemble part-based detection responses to a final classification result include the training of a combination classifier [2], [48], [62] and probabilistic inference to determine the most likely object configuration given the observed image features [45], [65], [76]. Codebook approaches represent pedestrians in a bottom-up fashion as assemblies of local codebook features, extracted around salient points in the image, combined with top-down verification [39], [40], [61].

Component-based approaches have certain advantages compared to full-body classification. They do not suffer from the unfavorable complexity related to the number of training examples necessary to adequately cover the set of possible appearances. Furthermore, the expectation of missing parts due to scene occlusions or interobject occlusions is easier addressed, particularly if explicit interobject occlusion reasoning is incorporated into the model [39], [40], [61], [76]. However, these advantages are paid for with higher complexity in both model generation (training) and application (testing). Their applicability to lower resolution images is limited since each component detector requires a certain spatial support for robustness.

2.3 Tracking

There has been extensive work on the tracking of pedestrians to infer trajectory-level information. One line of research has formulated tracking as frame-by-frame association of detections based on geometry and dynamics without particular pedestrian appearance models [2], [23]. Other approaches utilize pedestrian appearance models (Section 2.2) coupled with geometry and dynamics [4], [26], [32], [39], [43], [50], [55], [58], [65], [70], [76], [77], [80], [82]. Some approaches furthermore integrate detection and tracking in a Bayesian framework, combining appearance models with an observation density, dynamics, and probabilistic inference of the posterior state density. For this, either single [4], [26], [55], [70], [76] or multiple cues [32], [43], [50], [58], [65] are used.

The integration of multiple cues [66] involves combining separate models for each cue into a joint observation density. The inference of the posterior state density is usually formulated as a recursive filtering process [3]. Particle filters [30] are very popular due to their ability to closely approximate complex real-world multimodal posterior densities using sets of weighted random samples. Extensions that are especially relevant for pedestrian tracking involve hybrid discrete/continuous state-spaces [26], [50] and efficient sampling strategies [13], [32], [36], [44].

An important issue in real-world pedestrian tracking problems is how to deal with multiple targets in the image. Two basic strategies with regard to the tracking of multiple objects have been proposed. First, the theoretically most sound approach is to construct a joint state-space involving the number of targets and their configurations which are

Fig. 1. Overview of the Daimler pedestrian detection benchmark data set: (a) Pedestrian training samples, (b) nonpedestrian training images, (c) test images with annotations.

inferred in parallel. Problems arise regarding the significantly increased and variable dimensionality of the state-space. Solutions to reduce the computational complexity have involved grid-based or precalculated likelihoods [32], [69] and sophisticated resampling techniques such as Metropolis-Hastings sampling [36], partitioned sampling [44], or annealed particle filters [13]. Second, some approaches have been proposed to limit the number of objects to one per tracker and employ multiple tracker instances instead [31], [35], [50], [52]. While this technique simplifies the state-space representation, a method for initializing a track along with rules to separate neighboring tracks is required. Typically, an independent detector process is employed to initialize a new track.

Incorporating the independent detector into the proposal density tends to increase robustness by guiding the particle resampling toward candidate image regions. Competition rules between multiple tracker instances have been formulated in terms of heuristics [35], [50]. In contrast to joint state-space approaches, the quality of tracking is directly dependent on the capability of the associated object detector used for initialization.

3 Benchmark Data Set

Fig. 1 shows an excerpt from the Daimler pedestrian detection benchmark data set used in this work. Data set statistics are shown in Table 1. Training images were recorded at various daytimes and locations with no constraints on illumination, pedestrian pose, or clothing, except that pedestrians are fully visible in an upright position. The number of pedestrian (positive) samples provided as training examples is 15,660. These samples were obtained by manually extracting 3,915 rectangular position labels from video images. Four pedestrian samples were created from each label by means of mirroring and randomly shifting the bounding boxes by a few pixels in horizontal and vertical directions to account for localization errors in the application system. The addition of jittered samples was shown earlier to substantially improve the performance [14]. Pedestrian labels have a minimum height of 72 pixels so that there is no upscaling involved in view of different training sample resolutions for the systems under consideration. Further, we provide 6,744 full images not containing any pedestrians, from which all approaches under consideration extract negative samples for training.

Our test data set consists of an independent image sequence comprising 21,790 images (640×480 pixels) with 56,492 manual labels, including 259 trajectories of fully visible pedestrians, captured from a moving vehicle in a 27-minute drive through urban traffic. In contrast to other established benchmark data sets (see Table 1), the size and complexity of the current data allows to draw meaningful conclusions without appreciable overfitting effects. The data set has a total size of approximately 8.5 GB.[1]

4 Selected Pedestrian Detection Approaches

We select a diverse set of pedestrian detection approaches in terms of features (adaptive, nonadaptive) and classifier architecture for evaluation (see Section 5): Haar wavelet-based cascade [74], neural network using LRF features [75], and histograms of oriented gradients combined with a linear SVM [11]. In addition to these approaches, used in sliding window fashion, we consider a system utilizing coarse-to-fine shape matching and texture-based classification, i.e., a monocular variant of [23]. Temporal integration is incorporated by coupling all approaches with a 2D bounding box tracker.

We acknowledge that, besides the selected approaches, there exist many other interesting lines of research in the field of monocular pedestrian detection (see Section 2). We encourage other authors to report performances using the proposed data set and evaluation criteria for benchmarking. Here, we focus on the most widely used approaches.[2]

Our experimental setup assigns the underlying system parameters (e.g., feature layout, and training procedure) to the values reported to perform best in the original publications [11], [23], [49], [74], [75]. Two different resolutions of training samples are compared. We consider

1. The data set is made freely available to academic and nonacademic entities for research purposes. See http://www.science.uva.nl/research/isla/downloads/pedestrians/index.html or contact the second author.

2. Total processing time for training, testing, and evaluation was several months of CPU time on a 2.66 GHz Intel processor, using implementations in C/C++.

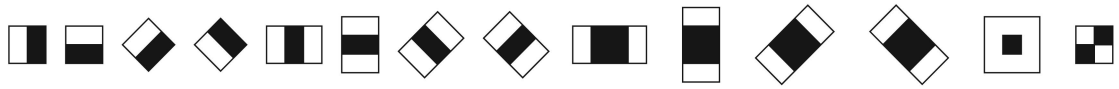

Fig. 2. Overview of the employed set of Haar wavelets. Black and white areas denote negative and positive weights, respectively.

training samples with an actual pedestrian height of 32 pixels (small scale) and 72 pixels (medium scale). To this a fixed fraction of border pixels (background) is added. Details are given below.

4.1 Haar Wavelet-Based Cascade

The Haar wavelet-based cascade framework [74] provides an efficient extension to the sliding window approach by introducing a degenerate decision tree of increasingly complex detector layers. Each layer employs a set of nonadaptive Haar wavelet features [48], [53]. We make use of Haar wavelet features at different scales and locations, comprising horizontal and vertical features, corresponding tilted features, as well as point detectors, see Fig. 2. Sample resolution for the small scale training set is 18×36 pixels with a border of two pixels around the pedestrian. No constraints on scales or locations of wavelets are imposed other than requiring the features to lie completely within our training samples. The total number of possible features is 154,190. The medium scale training set consists of samples at 40×80 pixels with a border of four pixels around the pedestrian which leads to over 3.5 million possible features. Here, we have to constrain the features to allow for feasible training: We require a minimum area of 24 pixels with a two-pixel scale step for each feature at a spatial overlap of 75 percent, which results in 134,621 possible features. In each cascade layer, AdaBoost [18] is used to construct a classifier based on a weighted linear combination of selected features, which yield the lowest error on the training set consisting of pedestrian and nonpedestrian samples.

We investigated the performance after N_l layers and found that performance saturated after incorporating $N_l = 15$ layers for both training resolutions. Each cascade layer is trained on a new data set consisting of the initial 15,660 pedestrian training samples and a new set of 15,660 nonpedestrian samples that is generated by collecting false positives of the cascade up to the previous layer on the given set of nonpedestrian images. Negative samples for the first layer are randomly sampled. Performance criteria for each layer are set to 50 percent false positive rate at 99.5 percent detection rate. Adding further cascade layers reduced the training error, but performance on the test set was observed to run in saturation. The total number of features selected by AdaBoost for the whole 15-layer cascade using small (medium) resolution samples is 4,070 (3,751), ranging from 15 (14) features in the first layer to 727 (674) features in the final layer. Experiments are conducted using the implementation found in the Intel OpenCV library [29].

4.2 Neural Network Using Local Receptive Fields (NN/LRF)

Adaptive local receptive fields (LRF) [19] have been shown to be powerful features in the domain of pedestrian detection, in combination with a multilayer feed-forward neural network architecture (NN/LRF) [75]. Although the combination of LRF features and nonlinear support vector machine classification (SVM/LRF) has been shown to yield slightly better performance [49], we opted for an NN/LRF in this work since training a nonlinear SVM/LRF classifier on our large data set was infeasible due to the excessive memory requirements.

In contrast to multilayer perceptrons, where the hidden layer is fully connected to the input layer, NN/LRF introduces the concept of N_B branches B_i $(i = 1, \ldots, N_B)$, where every neuron in each branch only receives input from a limited local region of the input layer, its receptive field. See Fig. 3. Since synaptical weights are shared among neurons in the same branch, every branch can be regarded

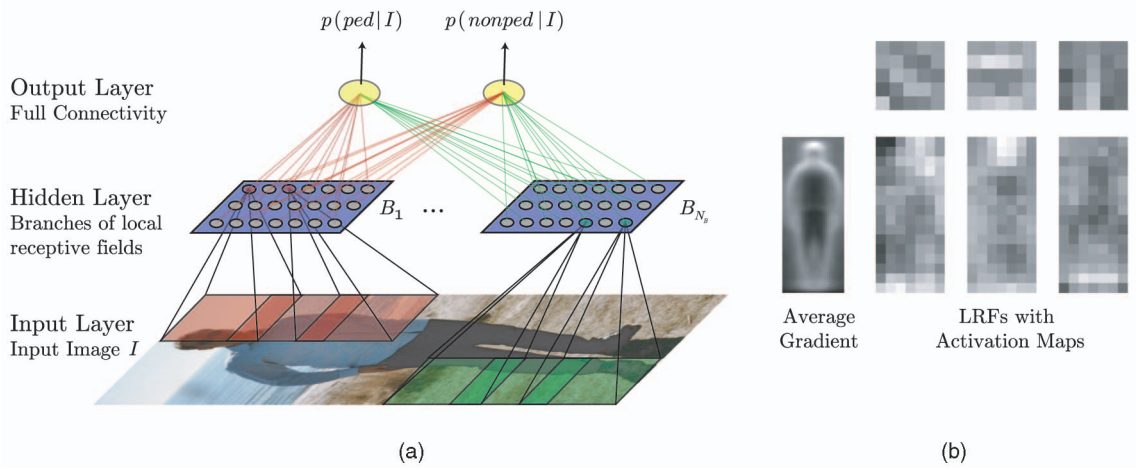

Fig. 3. (a) Overview of NN/LRF architecture. (b) Average gradient image along with three exemplary 5×5-pixel local receptive field features (hidden layer weights) and their activation maps (output layer weights) for the "pedestrian" output neuron, highlighting regions, where corresponding LRFs are most discriminative for the pedestrian class.

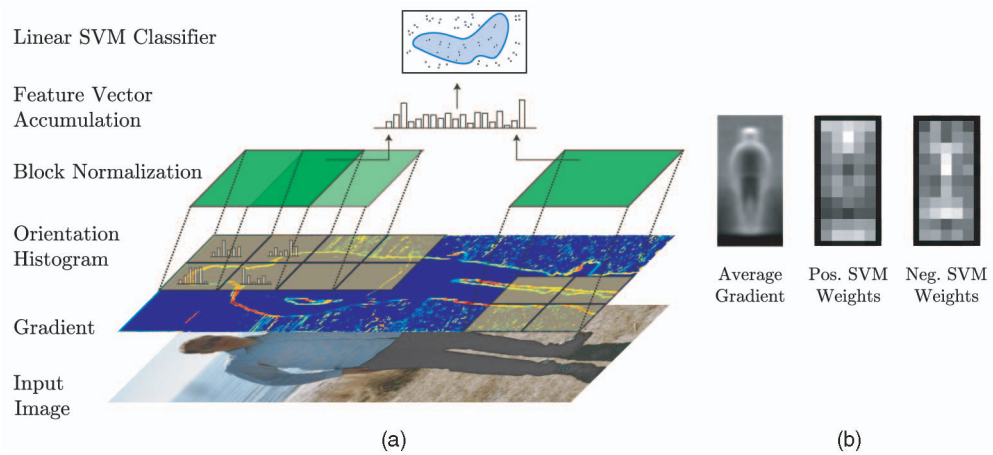

Fig. 4. (a) Overview of HOG/linSVM architecture. Cells on a spatial grid are shown in yellow, whereas overlapping normalization blocks are shown in green. (b) Average gradient image along with visualization of positive and negative SVM weights, which highlight the most discriminative regions for both the pedestrian and nonpedestrian classes.

as a spatial feature detector on the whole input pattern and the amount of parameters to be determined during training is reduced, alleviating susceptibility to overfitting.

We use an NN/LRF consisting of $N_B = 16$ branches B_i. For the small scale training samples at a resolution of 18×36 pixels with a two pixel border, 5×5-pixel receptive fields are utilized, shifted at a step size of two pixels over the training images. Receptive fields of 10×10-pixel are shifted at a step size of five pixels over the medium scale training samples, which are scaled to 40×80 pixels including a border of four pixels.

The output layer consists of two neurons, where the output of each neuron represents a (scaled) estimate of posterior probability for the pedestrian and nonpedestrian classes, respectively. Initial training data consist of the given 15,660 pedestrian samples, along with 15,560 randomly selected samples from the set of negative images. We further apply a bootstrapping strategy by shifting the trained NN/LRF classifier over the images containing no pedestrians and augmenting the negative training set by collecting 15,660 false positives in each iteration. Finally, the classifier is retrained using the extended negative training data. Bootstrapping is applied iteratively until test performance saturates. The higher complexity of the bootstrapped data set is accounted for by incorporating additional eight branches in each iteration to increase classifier complexity.

4.3 Histograms of Oriented Gradients with Linear SVM (HOG/linSVM)

We follow the approach of Dalal and Triggs [11] to model local shape and appearance using well-normalized dense histograms of gradient orientation (HOG), see Fig. 4. Local gradients are binned according to their orientation, weighted by their magnitude, within a spatial grid of cells with overlapping blockwise contrast normalization. Within each overlapping block, a feature vector is extracted by sampling the histograms from the contributing spatial cells. The feature vectors for all blocks are concatenated to yield a final feature vector, which is subject to classification using a linear support vector machine (linSVM).

Our choice of system parameters is based on the suggestions by Dalal and Triggs [11]. Compared to the Haar wavelet-based cascade and the NN/LRF, we employ a larger border to ensure ample spatial support for robust gradient computation and binning at the pedestrian boundary. Hence, small-scale training samples are utilized at a resolution of 22×44 pixels with a border of six pixels, whereas a resolution of 48×96 pixels with a border of 12 pixels is employed for medium-scale training.

We utilize fine scale gradients ($(-1, 0, 1)$ masks without smoothing), fine orientation binning (9 bins), coarse spatial binning (2×2 blocks of either 4×4 pixel cells for small-scale and 8×8 pixel cells for medium-scale training) as well as overlapping block contrast normalization (L_2-norm). The descriptor stride is set to half the block width, in order to have 50 percent overlap. This amounts to four pixels for small-scale and eight pixels for medium-scale training.

Similar to the training of the NN/LRF (see Section 4.2), the initial 15,560 negative samples are randomly sampled from the set of negative images. We apply bootstrapping by extending the training set by 15,660 additional false positives in each iteration until test performance saturated. As opposed to the NN/LRF classifier (see Section 4.2), the complexity of the linear SVM is automatically adjusted during training by increasing the number of support vectors as the training set becomes more complex. Experiments are conducted using the implementation by Dalal and Triggs [11].

4.4 Combined Shape-Texture-Based Pedestrian Detection

We consider a monocular version of the real-time *PROTECTOR* system [23] by cascading shape-based pedestrian detection with texture-based pedestrian classification. Shape-based detection is achieved by coarse-to-fine matching of an exemplar-based shape hierarchy to the image data at hand. The shape hierarchy is constructed offline in an automatic fashion from manually annotated shape labels, extracted from the 3,915 pedestrian examples in the training set (see Section 2). Online matching involves traversing the shape hierarchy with the Chamfer distance [6] between a shape template and an image subwindow as smooth and robust similarity measure. Image locations where the similarity between shape and image is above a

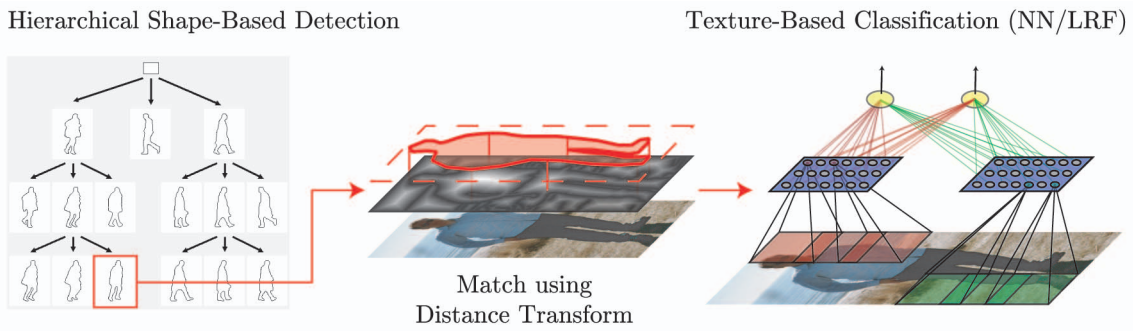

Fig. 5. Overview of combined shape-based detection and texture-based classification.

user-specified threshold are considered detections. A single distance threshold applies for each level of the hierarchy. Additional parameters govern the edge density on which the underlying distance map is based. All parameters have been optimized using a sequential ROC optimization technique [23].

Detections of the shape matching step are subject to verification by a texture-based pattern classifier. Here, we employ the multilayer feed-forward neural network operating on local adaptive receptive field features, NN/LRF, with parameters given in Section 4.2, on the small-scale training set. See Fig. 5. The initial negative training samples for the NN/LRF classifier were extracted by collecting false positives of the shape-based detection module (with a relaxed threshold) on the given set of negative images. Finally, bootstrapping is applied to the NN/LRF, as described in Section 4.2.

4.5 Temporal Integration—Tracking

Temporal integration of detection results allows us to overcome gaps in detection, suppress spurious false positives, and provide higher level temporally fused trajectory information for detected objects. Detections on the trajectory level are fundamental to many real-world attention focusing or risk assessment strategies, for instance, in vehicle-based collision-mitigation systems or visual surveillance scenarios. In this study, we employ a rudimentary 2D bounding box tracker with an object-state model involving bounding box position (x, y) and extent (w, h). Object-state parameters are estimated using an $\alpha - \beta$ tracker, involving the classical Hungarian method for data assignment [37]. A new track is started whenever a new object appears in m successive frames and no active track fits to it. It ends if the object corresponding to an active track has not been detected in n successive frames. We acknowledge the existence of more sophisticated trackers, see Section 2.3, whose performance evaluation remains for future work. The generality and simplicity of our tracker has the advantage of allowing a straightforward integration into other detector approaches to be considered.

5 Experiments

5.1 Methodology

Performance evaluation of the pedestrian detection systems is based on comparing system output (alarms) with manually labeled ground-truth (events) given by bounding box locations of pedestrians using the proposed benchmark test sequence consisting of 21,790 monocular images (see Section 3). We differentiate between the scenarios of generic pedestrian detection and (near) real-time pedestrian detection from a moving vehicle. There exists a wide range of possible applications of the first scenario, e.g., ranging from surveillance to advanced robotics. The second scenario is geared toward collision mitigation/avoidance in the context of intelligent vehicles [20], [23]. The two scenarios differ in the definition of the area of interest and match criteria. Additionally, the vehicle scenario involves restrictions on average processing time.

In both scenarios, we consider many-to-many data correspondences, that is, an event is matched if there is at least one alarm within localization tolerances, e.g., the systems are not required to detect each individual pedestrian in case of a pedestrian group. Multiple detector responses at near-identical locations and scales are addressed in all approaches by applying confidence-based nonmaximum suppression to the detected bounding boxes using pairwise box coverage: Two system alarms a_i and a_j are subject to nonmaximum suppression if their coverage

$$\Gamma(a_i, a_j) = \frac{A(a_i \cap a_j)}{A(a_i \cup a_j)},$$

the ratio of intersection area and union area, is above θ_n, with $\theta_n = 0.5$ in our evaluation. The detection with the lowest confidence is discarded, where confidence is assessed by the detectors, i.e., cascade (final layer), NN/LRF and SVM decision values. An alternative is to use kernel-based voting for position and scale of detected bounding boxes [10].

Performance is evaluated at both the frame and trajectory levels. Frame-level performance is measured in terms of sensitivity, precision, and false positives per frame. Sensitivity relates to the percentage of true solutions that were detected, whereas precision corresponds to the percentage of system solutions that were correct. We visualize frame-level performance in terms of ROC curves, depicting the trade-off between sensitivity and false positives per frame based on the corresponding match criteria. ROC curves for the NN/LRF and HOG/linSVM technique are generated by varying the corresponding detector output thresholds along the curve. In case of the wavelet-based cascade and the cascaded shape-texture pedestrian detection system, there are multiple thresholds (one for each cascade module) that can be varied simultaneously to determine ROC performance. Each

TABLE 2
Overview of Sliding Window Parameter Sets S_i for Generic Evaluation

	S_1	S_2	S_3	S_4	S_5	S_6
Spatial Stride (Δ_x, Δ_y)	(0.1,0.025)	(0.15,0.05)	(0.3,0.075)	(0.1,0.025)	(0.15,0.05)	(0.3,0.075)
Scale Step Δ_s	1.1	1.1	1.1	1.25	1.25	1.25
# of detection windows	184392	61790	20890	90982	30608	10256

multidimensional set of thresholds corresponds to a single point in ROC space, where the final ROC curve is computed as the Pareto-optimal frontier of this point cloud [23].

After incorporating temporal integration (tracking), trajectory-level performance is evaluated in terms of the percentage of matched ground-truth trajectories (sensitivity), the percentage of correct system trajectories (precision), and the number of false trajectories per minute. We distinguish between two types of trajectories (see [23]): "class-B" and "class-A" trajectories that have at least one or at least 50 percent of their events matched. "class-B" trajectories include "class-A" trajectories, but the latter demand stronger application performance. Further, we quantify the reduction in frame-level false positives resulting from the incorporation of the tracking component.

5.2 Generic Pedestrian Detection

In the evaluation of generic pedestrian detection, no additional (3D) scene knowledge and constraints are employed. Instead, we consider pedestrian detection solely as a 2D problem, where fully visible ground-truth pedestrians (see Table 1) of at least 72 pixels height are marked as required, which corresponds to real-world pedestrians of 1.5 meters height at a distance of 25 meters in our camera setup. Smaller or partially occluded pedestrians and bicyclists or motorcyclists are considered optional in that the systems are not rewarded/penalized for correct/false/missing detections. In our experiments, we consider in isolation the resolution of the training data (see Section 4), the size of the detector grid, as well as the effect of adding additional negative training samples by bootstrapping or cascading.

Combined shape-texture-based detection (Section 4.4) is disregarded here since the shape-based detection component, providing fast identification of possible pedestrian locations, is mainly employed because of processing speed, which is not considered in this evaluation scenario. We instead evaluate the NN/LRF classifier in isolation, which is the second (and more important) module of the combined shape-texture-based detection system.

This leaves us with a total of three approaches: the Haar wavelet-based cascade (Section 4.1), NN/LRF (Section 4.2), and HOG/linSVM (Section 4.3), which are used in a multiscale sliding window fashion. With s denoting the current scale, detector windows are both shifted through scale with a step factor of Δ_s and through location at fractions $s\Delta_x$ and $s\Delta_y$ of the base detector window size W_x and W_y (see Section 4) in both the x and y dimensions. The smallest scale s_{min} corresponds to a detector window height of 72 pixels, whereas the largest scale s_{max} has been chosen so that the detector windows still fit in the image. As a result, detector grids for all systems are identical. Several detector parameter settings $S_i = (\Delta_x^i, \Delta_y^i, \Delta_s^i)$, defining spatial stride (detector grid resolution) and scale, have been considered for all approaches, see Table 2. The 2D match criterion is based on bounding box coverage between a system alarm a_i and a ground-truth event e_j, where a correct detection is given by $\Gamma(a_i, e_j) > \theta_m$, with $\theta_m = 0.25$. Results are given in Figs. 6, 7, and 8.

Fig. 6a shows the effect of different training sample resolutions using detector parameters S_1. While the performance difference between small and medium resolutions for the wavelet-based cascade and the NN/LRF detectors is minor, the HOG/linSVM approach performs significantly worse at a small resolution. The reason for that may lie in the reduced spatial support for histogramming. Further experiments involve only the best performing resolution for each system: small resolution for the wavelet-based cascade and the NN/LRF detector and medium resolution for the HOG/linSVM approach.

Figs. 6b, 6c, and 6d show the localization tolerance of each detector, that is, the sensitivity to the granularity of the detection grid. Two observations can be made: First, all detectors perform best using the detection grid at the finest granularity (parameters S_1). Second, the localization tolerances of the approaches vary considerably. The NN/LRF detector performs almost identical for all parameter sets under consideration, with false positives per frame at constant detection rates being reduced by approximately a factor of 1.5, comparing the best (S_1) and the worst (S_6) settings. The wavelet-based cascade and HOG/linSVM approaches show a stronger sensitivity to the detection grid resolution, with a difference in false positives by approximately a factor of 3 and 5.5, respectively. We attribute this to the fact that the NN/LRF uses comparatively the largest features (5×5 pixel receptive fields at a sample size of 18×36 pixels, see Section 4.2), whereas 8×8 pixel cells are used in the HOG/linSVM approach with a sample size of 48×96 pixels (see Section 4.3). The wavelet-based cascade employs features at different scales, as shown in Section 4.1.

In the following experiments, we restrict ourselves to the detector parameter set S_1, which was identified as the best setting for all the techniques. We now evaluate the effect of adding negative samples to the training set, in terms of additional bootstrapping iterations for NN/LRF and HOG/linSVM and show the performance of individual layers of the wavelet-based cascade, each of which is trained on a different and increasingly more difficult set of negative samples. See Figs. 7a and 7b. All detectors show an initial performance improvement, but then saturate after 15 layers

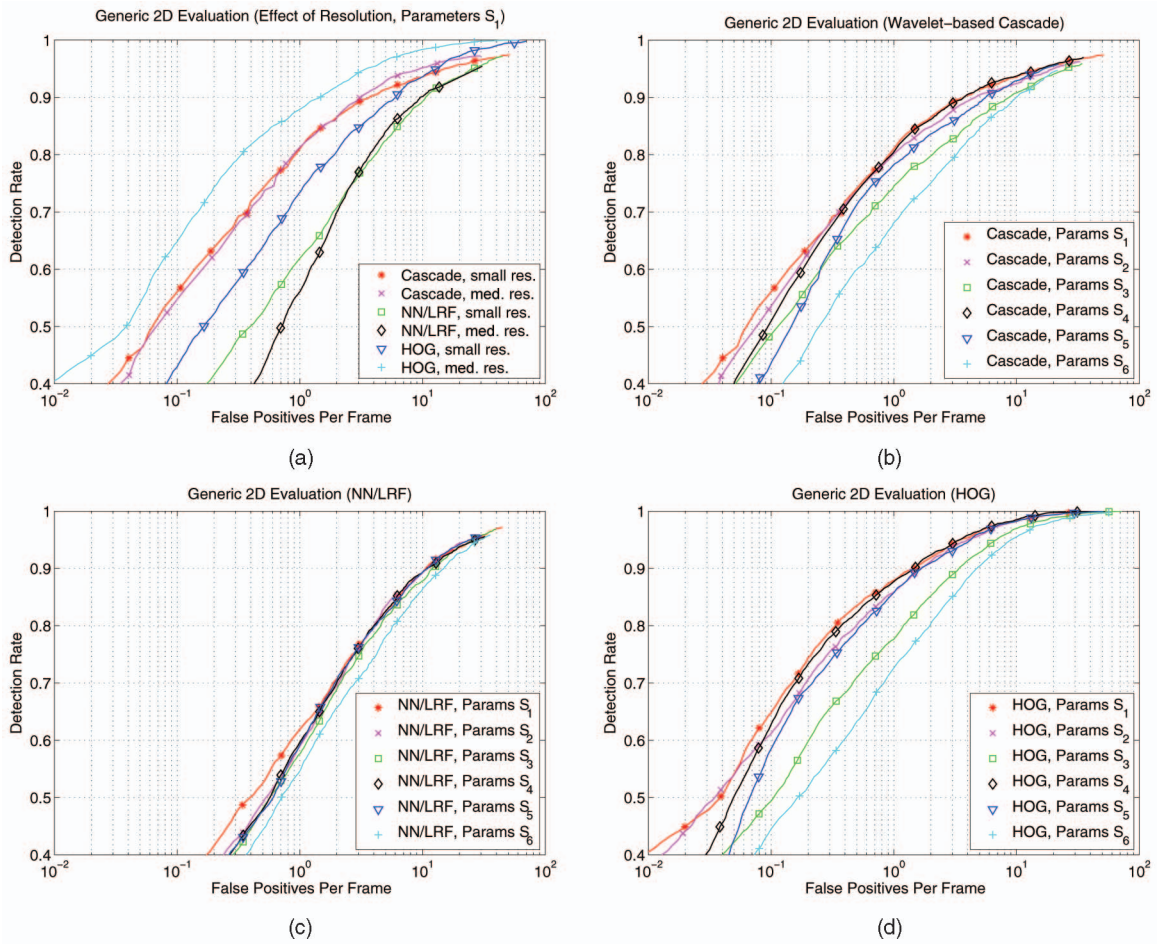

Fig. 6. Evaluation of generic pedestrian detection. (a) Effect of different training resolutions. (b)-(d) Effect of varying detector grid for (b) wavelet-based cascade, (c) NN/LRF (1 bootstrapping iteration), and (d) HOG/linSVM (1 bootstrapping iteration).

(wavelet-based cascade) or three (HOG/linSVM) and four (NN/LRF) bootstrapping iterations, respectively. The obtained performance improvements of the wavelet-based cascade and the NN/LRF detectors are paid for with an increase of computational costs, since the classifiers become more complex in case of more difficult training sets (recall that NN/LRF complexity was increased by design during bootstrapping, see Section 4.2). However, in the case of the HOG/linSVM detector, the processing time for the evaluation of a single detection window is constant. For a linear SVM, the processing time is independent from the actual number of support vectors [78], which becomes larger as

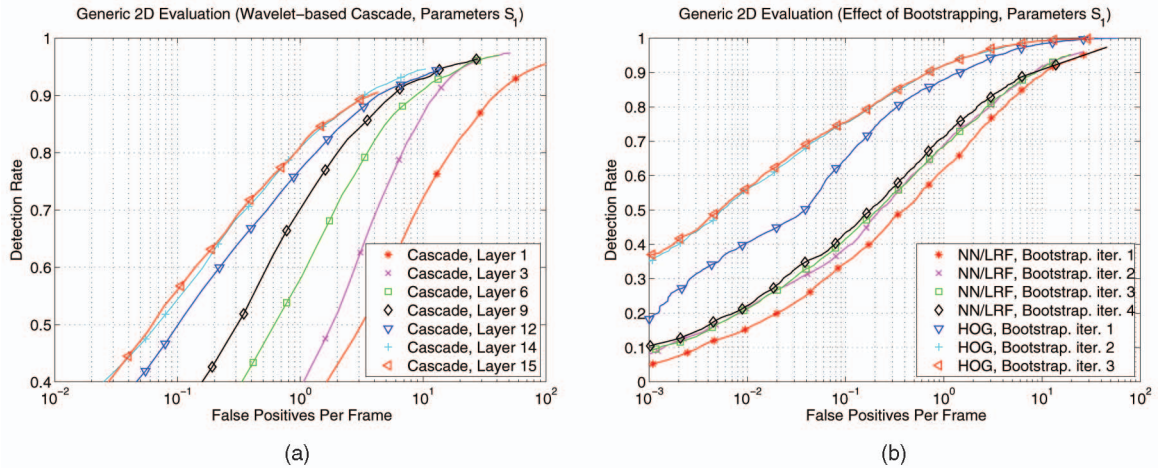

Fig. 7. Evaluation of generic pedestrian detection. (a) Performance of individual cascade layers. (b) Effect of bootstrapping on NN/LRF and HOG/linSVM.

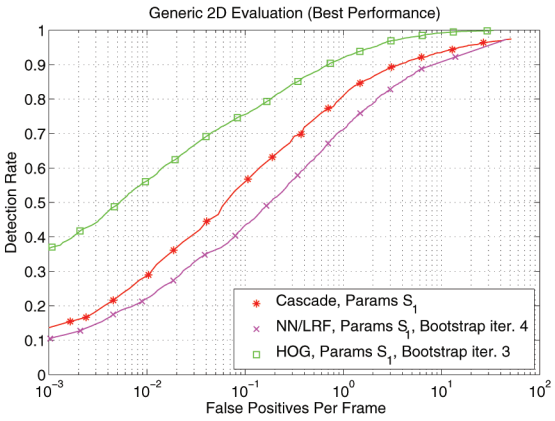

Fig. 8. Evaluation of generic pedestrian detection: best performance of each approach.

more bootstrapping iterations are conducted. Fig. 8 shows the best performance of each system on our test data set. The HOG/linSVM approach clearly outperforms both the wavelet-based cascade and NN/LRF. At a detection rate of 70 percent, false positives per frame for the HOG/linSVM detector amount to 0.045, compared to 0.38 and 0.86 for the wavelet-based cascade and NN/LRF. This is a reduction by a factor of 8 and 19, respectively.

Next, temporal integration is incorporated into all approaches using the 2D bounding box tracker (see Section 4.5) with parameters $m = 2$ and $n = 2$. Inputs to the tracker are system detections, with system parameterization selected from the corresponding ROC curves, as depicted in Fig. 8, at a common reference point of 60 percent sensitivity. Results are given in Table 3. One observes that the relative performance differences as shown in Fig. 8 still apply after tracking. The HOG/linSVM approach achieves a significantly higher precision at the same sensitivity level compared to the wavelet-based cascade and the NN/LRF detector.

5.3 Onboard Vehicle Application

In case of (near) real-time pedestrian detection from a moving vehicle, application-specific requirements are specified in 3D. In particular, the sensor coverage area is defined in relation to the vehicle as 10-25 m in longitudinal and ± 4 m in lateral direction. Given a system alarm a_i and ground-truth event e_j, we enforce a maximum positional deviation in 3D to count the alarm as match, where both 2D ground-truth and 2D detections are backprojected into 3D using known camera geometry and the assumption that pedestrians are standing on the ground plane (ground-plane constraint). Since this ground-plane assumption is only valid for fully visible pedestrians, partially visible pedestrians are not backprojected into 3D, but matched in 2D with a box coverage of $\theta_m = 0.25$, as shown in Section 5.2. Only fully visible ground-truth pedestrians (see Table 1) within the sensor coverage area are considered required. Partially visible pedestrians and pedestrians outside the sensor coverage area are regarded as optional (i.e., detections are neither credited nor penalized).

Localization tolerances are defined as percentage of distance for lateral (X) and longitudinal (Z) directions with respect to the vehicle. Here, we consider tolerances of $X = 10\%$ and $Z = 30\%$, with a larger tolerance in longitudinal direction to account for nonflat road surface and vehicle pitch in case of backprojection of (monocular) ground-truth and detections into 3D, i.e., at 20 m distance, we tolerate a localization error of ± 2 m and ± 6 m in lateral and longitudinal directions.

All systems are evaluated by incorporating 3D scene knowledge into the detection process: We assume pedestrians of heights 1.5-2.0 m to be standing on the ground. Initial object hypotheses violating these assumptions are discarded. Nonflat road surface and vehicle pitch are modeled by relaxing the ground-plane constraint using a pitch angle tolerance of $\psi = \pm 2$ degree.

We consider constraints on average processing times of 2.5 s and 250 ms (± 10 percent tolerance) per image. To enforce these constraints, we chose to maintain the fundamental system parameters, e.g., sample resolution or feature layout, as reported by the original authors, see Section 4. Instead, we use the size of the detection grid as a proxy for processing speed. Sliding window parameters T_i subject to processing time constraints are given in Table 4. The detector grids are finer grained in the y-direction than in the x-direction. This results in higher localization accuracy in the y-direction, which adds robustness to depth estimation by backprojecting detections into 3D. Instead of a sliding window approach, the combined shape-texture detector uses a coarse-to-fine hierarchical shape matching scheme yielding a variable number of ROIs per image, which are processed by the subsequent NN/LRF classifier. Hence, the hierarchy level thresholds of the shape matching module have the largest influence on

TABLE 3
System Performance After Tracking F/A/B Denote Frame and Trajectory-Level Performance

	Cascade			NN/LRF			HOG/linSVM		
	F	A	B	F	A	B	F	A	B
Sensitivity	65.4%	61.9%	73.0%	65.3%	69.8%	81.7%	64.1%	61.6%	76.2%
Precision	56.1%	47.3%	53.8%	33.5%	27.5%	33.3%	90.2%	84.9%	87.2%
FP 10^3 fr., min	156	19.0	16.7	307	35.7	35.1	16	2.0	1.7
Reduction False Positives	34.3 %	-	-	50.9 %	-	-	22.3 %	-	-
Avg. Proc. Time / 10^3 windows	20 ms			660 ms			430 ms		

False positives "FP" are given per 10^3 frames and per minute for frame level and trajectory performance.

TABLE 4
Overview of Sliding Window Parameter Sets T_i for Onboard Vehicle Evaluation

	Cascade		NN/LRF		HOG/linSVM	
	T_1 (2.5s)	T_4 (250ms)	T_2 (2.5s)	T_5 (250ms)	T_3 (2.5s)	T_6 (250ms)
Spatial Stride (Δ_x, Δ_y)	(0.05,0.025)	(0.05,0.025)	(0.1,0.025)	(0.3,0.08)	(0.1,0.025)	(0.3,0.08)
Scale Step Δ_s	1.05	1.05	1.1	1.25	1.1	1.25
# of detection windows	11312	11312	5920	617	5920	617

processing time. We have incorporated time constraints into the parameter optimization [23], to optimize these thresholds for the given processing time requirements.

Performance is evaluated for the full 15-layer cascade, the shape-texture detector, as well as the HOG/linSVM and NN/LRF approaches after every bootstrapping iteration to find the best compromise between performance and processing speed under the given time constraints. In contrast to the results of the generic evaluation, the best performance of the NN/LRF classifier is reached after the second bootstrapping iteration since the higher computational costs of more complex NN/LRF detectors require a too large reduction in detection grid resolution to meet the time constraints. In case of the wavelet-based cascade, identical parameter settings T_1 and T_4 are used for both time constraints settings. This is due to a very dense detection grid resolution even at time constraints of 250 ms per frame since each detection window can be evaluated very rapidly. A further increase of grid resolution does not yield any performance improvements. We attribute this effect to the preprocessing of the training data, where robustness to localization errors is explicitly modeled in terms of shifting the training labels by a few pixels, as described in Section 3. Results are given in Figs. 9a and 9b.

With processing time constraints of 2.5 s per frame, the relative performance of all detector variants is similar to the case of generic evaluation, see Figs. 8 and 9a. Compared to the application of the NN/LRF in isolation, the combined shape-texture detector further improves the performance, particularly at low false positive rates. Further restricting processing time constraints to 250 ms per frame effects a massive drop in the performance of the HOG/linSVM detector, whereas the performance of the NN/LRF decreases only slightly. Again, this is an effect of the different localization tolerances, as evaluated in Section 5.2. The performance of the combined shape-texture detector remains approximately constant. This indicates the powerful pruning capability of the shape detection module that allows to quickly focus the subsequent costly texture classification on promising image regions, which reduces computational costs. At tight processing time constraints, the wavelet-based cascade significantly outperforms every other detector considered, benefiting from its high processing speed. The combined shape-texture detector delivers the second best performance, admittedly at a proper gap.

As in the case of generic pedestrian detection (see Section 5.2), the bounding box tracker is incorporated. As a common reference point, we again use 60 percent sensitivity, obtained from the ROC curves depicted in Figs. 9a and 9b. Results are given in Table 5. For both time constraint settings, the relative performance order of various systems does not change in comparison to Figs. 9a and 9b. However, differences in the beneficial effect of the tracker can be observed. For all systems except HOG/linSVM, the benefit of the tracker is similar for the two time constraint settings, approximately 25-35 percent, see Table 5. For the HOG/linSVM detector at time constraints of 2.5 s per image, most false detections turn out to exhibit strong temporal coherence and cannot be eliminated by the tracker. The reduction in false positives only amounts to 12.5 percent.

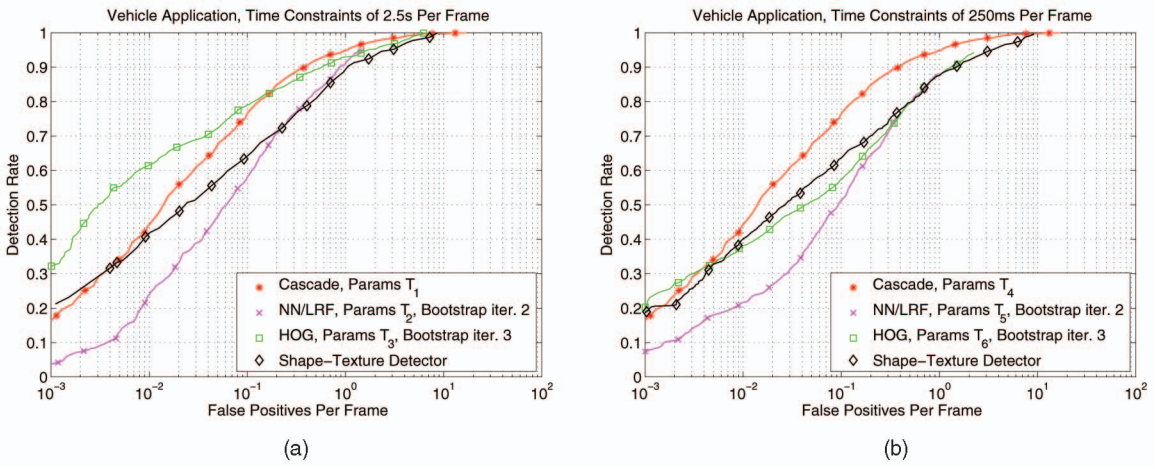

Fig. 9. Results of onboard vehicle application using time constraints of (a) 2.5 s/frame and (b) 250 ms/frame.

TABLE 5
System Performance After Tracking

		Cascade			NN/LRF			HOG/linSVM			Shape-Texture Rec.		
		F	A	B	F	A	B	F	A	B	F	A	B
Sensitivity	(TC 2.5s)	64.9%	58.2%	79.1%	65.5%	67.1%	82.1%	64.3%	58.2%	68.7%	64.6%	65.6%	85.0%
Precision	(TC 2.5s)	77.2%	71.5%	75.5%	53.4%	58.3%	63.1%	88.7%	81.2%	84.8%	59.3%	52.7%	62.1%
FP 10^3 fr., min	(TC 2.5s)	32	5.5	5.1	102	8.8	7.8	11.7	1.7	1.4	78	9.5	9.1
Reduction FP	(TC 2.5s)	23.6 %	-	-	30.6 %	-	-	12.5 %	-	-	28.9 %	-	-
Sensitivity	(TC 250ms)	64.9%	58.2%	79.1%	67.0%	71.6%	80.6%	67.4%	65.7%	79.1%	63.1%	65.2%	80.1%
Precision	(TC 250ms)	77.2%	71.5%	75.5%	43.4%	45.6%	52.2%	47.6%	50.8%	55.8%	59.2%	51.3%	61.9%
FP 10^3 fr., min	(TC 250ms)	32	5.5	5.1	171	17.2	15.0	143	14.5	13.0	81	9.1	8.7
Reduction FP	(TC 250ms)	23.6 %	-	-	31.3 %	-	-	37.3 %	-	-	26.1 %	-	-
Avg. Proc. Time / 10^3 windows		20 ms			440 ms			430 ms			approx. 620 ms		

F/A/B denote frame- and trajectory-level performance under processing time constraints "TC" of 2.5 s and 250 ms per image. False positives "FP" are given per 10^3 frames and per minute for frame-level and trajectory performance.

The stronger benefit of the tracker for the HOG/linSVM detector at 250 ms per image can be explained by the fact that fewer detection windows can be evaluated per image. To reach a sensitivity of 60 percent, a more relaxed threshold setting is required. As a result, additional false positives are introduced, which are observed to be less temporally coherent; these can be successfully suppressed by the tracker.

The average processing time per 10^3 detection windows is given in Table 5 using implementations in C/C++ on a 2.66 GHz Intel processor. In comparison to the other approaches, the wavelet-based cascade architecture has a massive advantage in processing time, i.e., it is approximately 20 times faster. Note that the combined shape-texture detector has the highest processing time per detection window. However, due to the efficient pruning of the search space by the coarse-to-fine shape matching module, the number of detection windows per image is greatly reduced in comparison to the sliding window approaches, while maintaining similar performance levels.

6 Discussion

We obtained a nuanced picture regarding the relative performance of methods tested, where the latter depends on the pedestrian image resolution and the spatial grid size used for probing (used as proxy for processing speed). At low-resolution pedestrian images (e.g., 18×36 pixels), dense Haar wavelet features represent the most viable option. HOG features, on the other hand, perform best at intermediate resolutions (e.g., 48×96 pixels). Their need for a larger spatial support limits their use in some application scenarios, for example, in our camera setup of Section 5.3, pedestrians further away than 25 m from the vehicle appear in the image with a height of less than 72 pixels. We would expect component-based or codebook [1], [39], [40], [61] approaches to be the natural choice for those applications involving yet higher resolution pedestrian images.

In terms of overall systems, results indicate a clear advantage of the HOG-based linear SVM approach at intermediate pedestrian image resolutions and lower processing speeds, and a superiority of the wavelet-based AdaBoost cascade approach at lower pedestrian image resolutions and (near) real-time processing speeds. Not surprisingly, tracking improves the performances of all considered systems, it also decreases the absolute performance differences among the systems. We observe that the tested systems in this study tend to make rather similar mistakes, although they are based on different features. For all systems, typical false detections occur in local regions, which are dominated by strong vertical structure, as shown in Fig. 10.

It is instructive to place the best performance obtained in context by comparing what would be necessary in a realistic application. Let us consider for this the intelligent vehicle application, which is described in Section 5.3. If we assume an assistance system using monocular vision that acoustically warns the driver of possible collisions with pedestrians, a correct detection rate upward of 80 percent on trajectory-level would be sensible, say, at a rate of less than one false alarm per

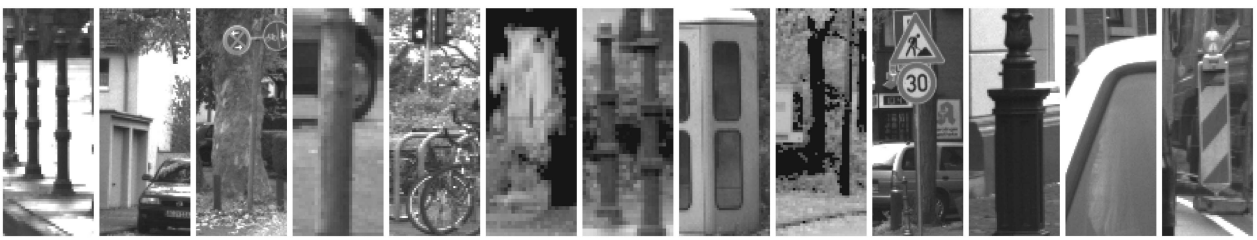

Fig. 10. Typical false positives of all systems. Most errors occur in local regions with strong vertical structure.

10 hours driving in urban traffic. Looking at the results currently obtained within 250 ms per frame (assuming that optimization would result in a real-time implementation), see Table 5, we see the best performance of approximately six false trajectories per minute at a detection rate of 60 percent for the wavelet-based cascade. One might be tempted to conclude that a performance gap of three orders of magnitude exists. This would be overly pessimistic, though, since Table 5 reflects the average performance over all pedestrian trajectories within the defined coverage area (10-25 m in distance, up to ± 4 m laterally). In practice, trajectories that are collision-relevant tend to be longer and individual detections are easier as they come closer to the vehicle. Our preliminary investigations show that detection performance on such trajectory subsets can be up to one order of magnitude higher, leaving a performance gap of two orders of magnitude.

How could one close the remaining performance gap? The most effective solution is to incorporate a preprocessing stage to constrain the image search space, based on alternate cues such as motion [15], [56] and depth [7], [23], [81]. For example, [23] reports performance gain of an order of magnitude by the inclusion of stereo-based obstacle detection (a similar boost can be expected in a surveillance setting by the incorporation of background subtraction).

Any remaining performance gain (i.e., one order of magnitude for the intelligent vehicle application listed above) would likely need to be derived from improving the actual classification methods. For example, in the shape-texture approach described in Section 4.4, hierarchical shape matching can be performed probabilistically, with improved performance [22]. The particular shape template matched could furthermore index into a set of classifiers (experts), each attuned to a particular body pose. Gavrila and Munder [23] report a performance improvement of about 30 percent from such a mixture-of-experts architecture. The cascade approach could be paired up with more powerful features, e.g., local receptive fields (Section 4.2) or gradient histograms (Section 4.3). Zhu et al. [83] presented initial work on cascade detectors using HOG features and reported real-time processing speeds at performance levels similar to the original HOG/linSVM approach [11].

Or perhaps it is the data that matters most, after all. A recent study on pedestrian classification [49] showed that the benefit of selecting the best combination of features and pattern classifiers was less pronounced than the gain obtained by increasing the training set, even though the base training set already involved many thousands of samples [49].

7 Conclusion

This paper presented a survey on recent work on monocular pedestrian detection from both a theoretical and an experimental perspective. In order to strike a suitable balance between generality and specificity, we considered two evaluation settings: a generic setting, where evaluation is done without scene and processing constraints, and one specific to an application onboard a moving vehicle in traffic.

Results show a nuanced picture regarding the relative performance of methods tested, where the latter depends on the pedestrian image resolution and the spatial grid size used for probing (used as proxy for processing speed). The HOG-based linear SVM approach significantly outperformed all other approaches considered at little or no processing constraints (factors of 10-18 and 3-6 less false class-A trajectories at no time constraints and at 2.5 s per frame, respectively). This suggests that feature representations based on local edge orientation are well suited to capture the complex appearance of the pedestrian object class. As tighter processing constraints are imposed, the Haar wavelet-based cascade approach outperforms all other detectors considered (factor of 2-3 less false class-A trajectories at 250 ms per frame).

For all systems, performance is enhanced by incorporating temporal integration and/or restrictions of the search space based on scene knowledge. The tracking component tends to decrease the absolute performance differences of the systems. From a real-world application perspective, the amount of false trajectories is too high by at least one order of magnitude, which shows that significant effort is further necessary on this complex but important problem.

Acknowledgments

The authors acknowledge the support of the *Studienstiftung des deutschen Volkes* and *Bundesministerium für Wirtschaft und Technologie, BMWi* in the context of the *AKTIV-SFR* initiative. They furthermore thank Professor Dr. Christoph Schnörr (Image and Pattern Analysis Group, University of Heidelberg, Germany), who provided helpful comments and discussions.

References

[1] S. Agarwal, A. Awan, and D. Roth, "Learning to Detect Objects in Images via a Sparse, Part-Based Representation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 26, no. 11, pp. 1475-1490, Nov. 2004.

[2] I.P. Alonso et al. "Combination of Feature Extraction Methods for SVM Pedestrian Detection," *IEEE Trans. Intelligent Transportation Systems,* vol. 8, no. 2, pp. 292-307, June 2007.

[3] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A Tutorial on Particle Filters for On-Line Non-Linear/Non-Gaussian Bayesian Tracking," *IEEE Trans. Signal Processing,* vol. 50, no. 2, pp. 174-188, Feb. 2002.

[4] A. Baumberg, "Hierarchical Shape Fitting Using an Iterated Linear Filter," *Proc. British Machine Vision Conf.,* pp. 313-323, 1996.

[5] M. Bergtholdt, D. Cremers, and C. Schnörr, "Variational Segmentation with Shape Priors," *Handbook of Math. Models in Computer Vision,* N. Paragios, Y. Chen, and O. Faugeras, eds., Springer, 2005.

[6] G. Borgefors, "Distance Transformations in Digital Images," *Computer Vision, Graphics, and Image Processing,* vol. 34, no. 3, pp. 344-371, 1986.

[7] A. Broggi, A. Fascioli, I. Fedriga, A. Tibaldi, and M.D. Rose, "Stereo-Based Preprocessing for Human Shape Localization in Unstructured Environments," *Proc. IEEE Intelligent Vehicles Symp.,* pp. 410-415, 2003.

[8] T.F. Cootes, S. Marsland, C.J. Twining, K. Smith, and C.J. Taylor, "Groupwise Diffeomorphic Non-Rigid Registration for Automatic Model Building," *Proc. European Conf. Computer Vision,* pp. 316-327, 2004.

[9] T.F. Cootes and C.J. Taylor, "Statistical Models of Appearance for Computer Vision," technical report, Univ. of Manchester, 2004.

[10] N. Dalal, "Finding People in Images and Videos," PhD thesis, Institut Nat'l Polytechnique de Grenoble, 2006.

[11] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* pp. 886-893, 2005.

[12] N. Dalal, B. Triggs, and C. Schmid, "Human Detection Using Oriented Histograms of Flow and Appearance," *Proc. European Conf. Computer Vision,* pp. 428-441, 2006.

[13] J. Deutscher, A. Blake, and I.D. Reid, "Articulated Body Motion Capture by Annealed Particle Filtering," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* pp. 126-133, 2000.

[14] M. Enzweiler and D.M. Gavrila, "A Mixed Generative-Discriminative Framework for Pedestrian Classification," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 2008.

[15] M. Enzweiler, P. Kanter, and D.M. Gavrila, "Monocular Pedestrian Recognition Using Motion Parallax," *Proc. IEEE Intelligent Vehicles Symp.,* pp. 792-797, 2008.

[16] A. Ess, B. Leibe, and L. van Gool, "Depth and Appearance for Mobile Scene Analysis," *Proc. Int'l Conf. Computer Vision,* 2007.

[17] L. Fan, K.-K. Sung, and T.-K. Ng, "Pedestrian Registration in Static Images with Unconstrained Background," *Pattern Recognition,* vol. 36, pp. 1019-1029, 2003.

[18] Y. Freund and R.E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," *Proc. European Conf. Computational Learning Theory,* pp. 23-37, 1995.

[19] K. Fukushima, S. Miyake, and T. Ito, "Neocognitron: A Neural Network Model for a Mechanism of Visual Pattern Recognition," *IEEE Trans. Systems, Man, and Cybernetics,* vol. 13, pp. 826-834, 1983.

[20] T. Gandhi and M.M. Trivedi, "Pedestrian Protection Systems: Issues, Survey, and Challenges," *IEEE Trans. Intelligent Transportation Systems,* vol. 8, no. 3, pp. 413-430, Sept. 2007.

[21] D.M. Gavrila, "The Visual Analysis of Human Movement: A Survey," *Computer Vision and Image Understanding,* vol. 73, no. 1, pp. 82-98, 1999.

[22] D.M. Gavrila, "A Bayesian Exemplar-Based Approach to Hierarchical Shape Matching," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 29, no. 8, pp. 1408-1421, Aug. 2007.

[23] D.M. Gavrila and S. Munder, "Multi-Cue Pedestrian Detection and Tracking from a Moving Vehicle," *Int'l J. Computer Vision,* vol. 73, no. 1, pp. 41-59, 2007.

[24] B.E. Goldstein, *Sensation and Perception,* sixth ed. Wadsworth, 2002.

[25] T. Heap and D. Hogg, "Improving Specificity in PDMs Using a Hierarchical Approach," *Proc. British Machine Vision Conf.,* pp. 80-89, 1997.

[26] T. Heap and D. Hogg, "Wormholes in Shape Space: Tracking through Discontinuous Changes in Shape," *Proc. Int'l Conf. Computer Vision,* pp. 344-349, 1998.

[27] B. Heisele and C. Wöhler, "Motion-Based Recognition of Pedestrians," *Proc. Int'l Conf. Pattern Recognition,* pp. 1325-1330, 1998.

[28] INRIA Person Dataset, http://pascal.inrialpes.fr/data/human/, 2007.

[29] Intel OpenCV Library, http://www.intel.com/technology/computing/opencv/, 2007.

[30] M. Isard and A. Blake, "CONDENSATION—Conditional Density Propagation for Visual Tracking," *Int'l J. Computer Vision,* vol. 29, no. 1, pp. 5-28, 1998.

[31] M. Isard and A. Blake, "ICONDENSATION: Unifying Low-Level and High-Level Tracking in a Stochastic Framework," *Proc. Int'l Conf. Computer Vision,* pp. 893-908, 1998.

[32] M. Isard and J. MacCormick, "BraMBLE: A Bayesian Multiple-Blob Tracker," *Proc. Int'l Conf. Computer Vision,* pp. 34-41, 2001.

[33] A.K. Jain, R.P.W. Duin, and J. Mao, "Statistical Pattern Recognition: A Review," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 1, pp. 4-37, Jan. 2000.

[34] M.J. Jones and T. Poggio, "Multidimensional Morphable Models," *Proc. Int'l Conf. Computer Vision,* pp. 683-688, 1998.

[35] H. Kang and D. Kim, "Real-Time Multiple People Tracking Using Competitive Condensation," *Pattern Recognition,* vol. 38, no. 7, pp. 1045-1058, 2005.

[36] Z. Khan, T. Balch, and F. Dellaert, "MCMC-Based Particle Filtering for Tracking a Variable Number of Interacting Targets," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 27, no. 11, pp. 1805-1819, Nov. 2005.

[37] H.W. Kuhn, "The Hungarian Method for the Assignment Problem," *Naval Research Logistics Quarterly,* vol. 2, pp. 83-97, 1955.

[38] S. Lee, Y. Liu, and R. Collins, "Shape Variation-Based Frieze Pattern for Robust Gait Recognition," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 2007.

[39] B. Leibe, N. Cornelis, K. Cornelis, and L.V. Gool, "Dynamic 3D Scene Analysis from a Moving Vehicle," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 2007.

[40] B. Leibe, E. Seemann, and B. Schiele, "Pedestrian Detection in Crowded Scenes," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* pp. 878-885, 2005.

[41] R. Lienhart and J. Maydt, "An Extended Set of Haar-Like Features for Rapid Object Detection," *Proc. Int'l Conf. Image Processing,* pp. 900-903, 2002.

[42] D.G. Lowe, "Distinctive Image Features from Scale Invariant Keypoints," *Int'l J. Computer Vision,* vol. 60, no. 2, pp. 91-110, 2004.

[43] J. MacCormick and A. Blake, "Partitioned Sampling, Articulated Objects and Interface-Quality Hand Tracking," *Proc. European Conf. Computer Vision,* pp. 3-19, 2000.

[44] J. MacCormick and A. Blake, "A Probabilistic Exclusion Principle for Tracking Multiple Objects," *Int'l J. Computer Vision,* vol. 39, no. 1, pp. 57-71, 2000.

[45] K. Mikolajczyk, C. Schmid, and A. Zisserman, "Human Detection Based on a Probabilistic Assembly of Robust Part Detectors," *Proc. European Conf. Computer Vision,* pp. 69-81, 2004.

[46] MIT CBCL Pedestrian Database, http://cbcl.mit.edu/cbcl/software-datasets/PedestrianData.html, 2008.

[47] T.B. Moeslund and E. Granum, "A Survey of Advances in Vision-Based Human Motion Capture and Analysis," *Computer Vision and Image Understanding,* vol. 103, nos. 2/3, pp. 90-126, 2006.

[48] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-Based Object Detection in Images by Components," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, no. 4, pp. 349-361, Apr. 2001.

[49] S. Munder and D.M. Gavrila, "An Experimental Study on Pedestrian Classification," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 28, no. 11, pp. 1863-1868, Nov. 2006.

[50] S. Munder, C. Schnörr, and D.M. Gavrila, "Pedestrian Detection and Tracking Using a Mixture of View-Based Shape-Texture Models," *IEEE Trans. Intelligent Transportation Systems,* vol. 9, no. 2, pp. 333-343, June 2008.

[51] C. Nakajima, M. Pontil, B. Heisele, and T. Poggio, "Full-Body Recognition System," *Pattern Recognition,* vol. 36, pp. 1997-2006, 2003.

[52] K. Okuma, A. Taleghani, N. de Freitas, J. Little, and D. Lowe, "A Boosted Particle Filter: Multitarget Detection and Tracking," *Proc. European Conf. Computer Vision,* pp. 28-39, 2004.

[53] C. Papageorgiou and T. Poggio, "A Trainable System for Object Detection," *Int'l J. Computer Vision,* vol. 38, pp. 15-33, 2000.

[54] PETS Data sets, http://www.cvg.rdg.ac.uk/slides/pets.html, 2007.

[55] V. Philomin, R. Duraiswami, and L.S. Davis, "Quasi-Random Sampling for Condensation," *Proc. European Conf. Computer Vision,* pp. 134-149, 2000.

[56] R. Polana and R. Nelson, "Low-Level Recognition of Human Motion," *Proc. IEEE Workshop Motion of Non-Rigid and Articulated Objects,* pp. 77-92, 1994.

[57] R. Poppe, "Vision-Based Human Motion Analysis: An Overview," *Computer Vision and Image Understanding,* vol. 108, pp. 4-18, 2007.

[58] D. Ramanan, A.D. Forsyth, and A. Zisserman, "Strike a Pose: Tracking People by Finding Stylized Poses," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* pp. 271-278, 2005.

[59] T. Randen and J.H. Husøy, "Filtering for Texture Classification: A Comparative Study," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 21, no. 4, pp. 291-310, Apr. 1999.

[60] P. Sabzmeydani and G. Mori, "Detecting Pedestrians by Learning Shapelet Features," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 2007.

[61] E. Seemann, M. Fritz, and B. Schiele, "Towards Robust Pedestrian Detection in Crowded Image Sequences," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 2007.

[62] A. Shashua, Y. Gdalyahu, and G. Hayon, "Pedestrian Detection for Driving Assistance Systems: Single-Frame Classification and System Level Performance," *Proc. IEEE Intelligent Vehicles Symp.,* pp. 1-6, 2004.

[63] V.D. Shet, J. Neumann, V. Ramesh, and L.S. Davis, "Bilattice-Based Logical Reasoning for Human Detection," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 2007.

[64] H. Shimizu and T. Poggio, "Direction Estimation of Pedestrian from Multiple Still Images," *Proc. IEEE Intelligent Vehicles Symp.,* pp. 596-600, 2004.

[65] H. Sidenbladh and M.J. Black, "Learning the Statistics of People in Images and Video," *Int'l J. Computer Vision,* vol. 54, nos. 1-3, pp. 183-209, 2003.
[66] M. Spengler and B. Schiele, "Towards Robust Multi-Cue Integration for Visual Tracking," *Machine Vision and Applications,* vol. 14, no. 1, pp. 50-58, 2003.
[67] B. Stenger, A. Thayananthan, P.H.S. Torr, and R. Cipolla, "Model-Based Hand Tracking Using a Hierarchical Bayesian Filter," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 28, no. 9, pp. 1372-1385, Sept. 2006.
[68] M. Szarvas, A. Yoshizawa, M. Yamamoto, and J. Ogata, "Pedestrian Detection with Convolutional Neural Networks," *Proc. IEEE Intelligent Vehicles Symp.,* pp. 223-228, 2005.
[69] L. Taycher, G. Shakhnarovich, D. Demirdjian, and T. Darrell, "Conditional Random People: Tracking Humans with CRFs and Grid Filters," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* pp. 222-229, 2006.
[70] K. Toyama and A. Blake, "Probabilistic Tracking with Exemplars in a Metric Space," *Int'l J. Computer Vision,* vol. 48, no. 1, pp. 9-19, 2002.
[71] O. Tuzel, F. Porikli, and P. Meer, "Human Detection via Classification on Riemannian Manifolds," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 2007.
[72] I. Ulusoy and C.M. Bishop, "Generative versus Discriminative Methods for Object Recognition," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* pp. 258-265, 2005.
[73] V.N. Vapnik, *The Nature of Statistical Learning Theory.* Springer, 1995.
[74] P. Viola, M. Jones, and D. Snow, "Detecting Pedestrians Using Patterns of Motion and Appearance," *Int'l J. Computer Vision,* vol. 63, no. 2, pp. 153-161, 2005.
[75] C. Wöhler and J. Anlauf, "An Adaptable Time-Delay Neural-Network Algorithm for Image Sequence Analysis," *IEEE Trans. Neural Networks,* vol. 10, no. 6, pp. 1531-1536, Nov. 1999.
[76] B. Wu and R. Nevatia, "Detection and Tracking of Multiple, Partially Occluded Humans by Bayesian Combination of Edgelet Based Part Detectors," *Int'l J. Computer Vision,* vol. 75, no. 2, pp. 247-266, 2007.
[77] Y. Wu and T. Yu, "A Field Model for Human Detection and Tracking," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 28, no. 5, pp. 753-765, May 2006.
[78] K. Zapien, J. Fehr, and H. Burkhardt, "Fast Support Vector Machine Classification Using Linear SVMs," *Proc. Int'l Conf. Pattern Recognition,* pp. 366-369, 2006.
[79] H. Zhang, A. Berg, M. Maire, and J. Malik, "SVM-KNN: Discriminative Nearest Neighbor Classification for Visual Category Recognition," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 2006.
[80] L. Zhang, B. Wu, and R. Nevatia, "Detection and Tracking of Multiple Humans with Extensive Pose Articulation," *Proc. Int'l Conf. Computer Vision,* 2007.
[81] L. Zhao and C. Thorpe, "Stereo and Neural Network-Based Pedestrian Detection," *IEEE Trans. Intelligent Transportation Systems,* vol. 1, no. 3, pp. 148-154, Sept. 2000.
[82] T. Zhao and R. Nevatia, "Tracking Multiple Humans in Complex Situations," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 26, no. 9, pp. 1208-1221, Sept. 2004.
[83] Q. Zhu, S. Avidan, M. Yeh, and K. Cheng, "Fast Human Detection Using a Cascade of Histograms of Oriented Gradients," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* pp. 1491-1498, 2006.

Markus Enzweiler received the MSc degree in computer science from the University of Ulm, Germany, in 2005. Since 2006, he has been working toward the PhD degree with the Image and Pattern Analysis Group at the University of Heidelberg, Germany, while on site at Daimler Research in Ulm, Germany. In 2002 and 2003, he was a visiting student researcher at the Centre for Vision Research at York University, Toronto, Canada. His current research focuses on statistical models of human appearance with application to pedestrian recognition in the domain of intelligent vehicles. He holds a PhD scholarship from the Studienstiftung des deutschen Volkes (German National Academic Foundation) and is an IEEE student member. More details about his research and background can be found at http://www.markus-enzweiler.de.

Dariu M. Gavrila received the MSc degree in computer science from the Free University of Amsterdam in 1990 and the PhD degree in computer science from the University of Maryland at College Park in 1996. Since 1997, he has been a senior research scientist at Daimler Research in Ulm, Germany. He was a visiting researcher at the MIT Media Laboratory in 1996. In 2003, he became a professor in the Faculty of Science at the University of Amsterdam, chairing the area of Intelligent Perception Systems (part time). Over the last decade, he has focused on visual systems for detecting human presence and recognizing activity, with application to intelligent vehicles and surveillance. He has published more than 20 papers in this area and received the I/O Award 2007 from the Netherlands Organization for Scientific Research (NWO). More details about his research and background can be found at http://www.gavrila.net.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.