Pedestrian Detection: A Benchmark

Piotr Dollár[1] Christian Wojek[2] Bernt Schiele[2] Pietro Perona[1]

[1]Dept. of Electrical Engineering
California Institute of Technology
{*pdollar,perona*}*@caltech.edu*

[2]Dept. of Computer Science
TU Darmstadt
{*wojek,schiele*}*@cs.tu-darmstadt.de*

Abstract

Pedestrian detection is a key problem in computer vision, with several applications including robotics, surveillance and automotive safety. Much of the progress of the past few years has been driven by the availability of challenging public datasets. To continue the rapid rate of innovation, we introduce the Caltech Pedestrian Dataset, which is two orders of magnitude larger than existing datasets. The dataset contains richly annotated video, recorded from a moving vehicle, with challenging images of low resolution and frequently occluded people. We propose improved evaluation metrics, demonstrating that commonly used per-window measures are flawed and can fail to predict performance on full images. We also benchmark several promising detection systems, providing an overview of state-of-the-art performance and a direct, unbiased comparison of existing methods. Finally, by analyzing common failure cases, we help identify future research directions for the field.

1. Introduction

Detecting people in images is a problem with a long history [37, 13, 35, 27, 16, 41, 23, 5]; in the past two years there has been a surge of interest in pedestrian detection [6, 9, 11, 18, 20, 21, 25, 30, 32, 33, 36, 38, 42]. Accurate pedestrian detection would have immediate and far reaching impact to applications such as surveillance, robotics, assistive technology for the visually impaired, content based indexing (*e.g.* Flickr, Google, movies), advanced human machine interfaces and automotive safety, among others. Automotive applications [12, 14, 34] are particularly compelling as they have the potential to save numerous lives [39].

Publicly available benchmarks, the most popular of which is the INRIA dataset [5], have contributed to spurring interest and progress in this area of machine vision. However, as algorithm performance improves, more challenging datasets are necessary to continue the rapid pace of progress and to inspire novel ideas. Existing pedestrian datasets often contain a limited range of scale, occlusion and pose variation, and are fairly small, making it difficult to assess real world performance (see Sec. 2.4). As we will demonstrate, the established methodology of evaluating pedestrian detectors, which uses *per-window* measures of performance, is flawed and can fail to predict actual *per-image* performance.

Figure 1. Example images (cropped) and annotations. The solid green boxes denote the full pedestrian extent while the dashed yellow boxes denote the visible regions. The Caltech Pedestrian Database, collected from a vehicle driving through regular traffic in an urban environment, consists of 350,000 labeled pedestrian bounding boxes in 250,000 frames.

Our contribution is fourfold. (1) We introduce the Caltech Pedestrian Dataset[1], which is two orders of magnitude larger than any existing dataset. The pedestrians vary widely in appearance, pose and scale; furthermore, occlusion information is annotated (see Fig. 1). These statistics are more representative of real world applications and allow for in depth analysis of existing algorithms. (2) We propose improved performance metrics. (3) We benchmark seven algorithms [40, 5, 7, 30, 11, 42, 21], either obtained directly from the original authors or reimplemented in-house. (4) We highlight situations of practical interest under which existing methods fail and identify future research directions.

We introduce the Caltech Pedestrian Dataset and describe its statistics in Sec. 2. In Sec. 3, we discuss the pitfalls of per-window metrics and describe our evaluation methodology, based on the PASCAL criteria [28]. In Sec. 4 we report a detailed performance evaluation for seven promising methods for pedestrian detection. We summarize our findings and discuss open problems in Sec. 5.

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

978-1-4244-3991-1/09/$25.00 ©2009 IEEE

2. Dataset

Challenging datasets are catalysts for progress in computer vision. The Berkeley Segmentation Dataset [22], the Barron *et al*. [3] and Middlebury [2] optical flow datasets, the Middlebury Stereo Dataset [31] and the Caltech 101 object categorization dataset [10] all improved performance evaluation and helped drive innovation in their respective fields. Much in the same way, our goal in introducing the Caltech Pedestrian Dataset is to provide a better benchmark and to help identify conditions under which current detection methods fail and thus focus research effort on these difficult cases.

2.1. Images and Ground Truth

We collected approximately 10 hours of 30Hz video ($\sim 10^6$ frames) taken from a vehicle driving through regular traffic in an urban environment (camera setup shown in Fig. 2). The driver was independent from the authors of this study and had instructions to drive normally through areas where pedestrians were frequently present. The video was captured in the greater Los Angeles metropolitan area from neighborhoods chosen for their relatively high concentration of pedestrians: LAX, Santa Monica, Hollywood, Pasadena, and Little Tokyo.

The CCD video resolution is 640×480, and, not unexpectedly, the overall image quality is lower than that of still images of comparable resolution. There are minor variations in the camera position due to repeated mountings of the camera. The video was stabilized to remove effects of the vehicle pitching, primarily to simplify annotation. To perform the stabilization, we implemented a differential camera tracker based on the system described in [45].

Figure 2. Camera setup.

total frames	∼1000K
labeled frames	∼250K
frames w peds.	∼132K
# bounding boxes	∼350K
# occluded BB	∼126K
# unique peds.	∼2300
ave ped. duration	∼5s
labeling effort	∼400h

Figure 3. Database Summary.

We annotated 250,000 frames (in 137 approximately minute long segments) for a total of 350,000 labeled bounding boxes and 2300 unique pedestrians. To make such a large scale labeling effort feasible we created a user-friendly labeling tool, briefly described in Fig. 4.

For every frame in which a given pedestrian is visible, labelers drew a tight bounding box (BB) that indicated the full extent of the *entire* pedestrian. For occluded pedestrians this involves estimating the location of hidden parts; in addition a second BB was used to delineate the visible region. During an occlusion event, the estimated full BB stays relatively constant while the visible BB may change rapidly. For comparison, in the PASCAL labeling scheme [28] only the visible BB is labeled and occluded pedestrians are marked as 'truncated'.

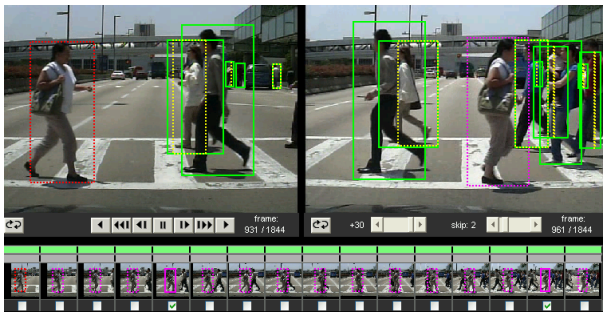


Figure 4. Screenshot of the video labeler. It is designed so that users can efficiently navigate and annotate the video with a minimum amount of labor. The most salient aspect of the labeler an interactive procedure where the user labels only a sparse set of frames and the system automatically labels intermediate frames by interpolation.

Each sequence of BBs belonging to a given object was assigned one of three labels. Individual pedestrians were labeled 'Person' (∼1900 instances). Large groups of pedestrians for which it would have been tedious or impossible to label individuals were delineated using a single BB and labeled as 'People' (∼300). In addition, the label 'Person?' was assigned when clear identification of a pedestrian was ambiguous or easily mistaken (∼110). Example images with overlaid annotations are shown in Fig. 1.

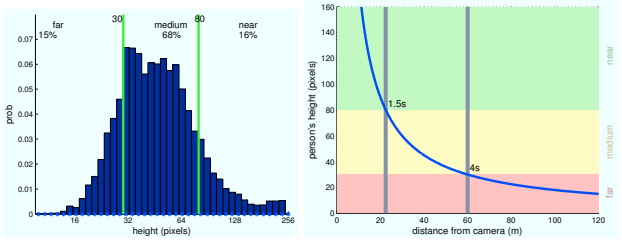
2.2. Dataset Statistics

A summary of the database is given in Fig. 3. About 50% of the frames have no pedestrians, while 30% have two or more. Pedestrians are visible for 5s on average. Below, we give detailed analysis of the distribution of pedestrian scale, occlusion and location. This will serve as a foundation for establishing the requirements for a real world system.

Scale: We group pedestrians by their image size (height in pixels) into three scales: *near* (80 or more pixels), *medium* (between 30-80 pixels) and *far* (30 pixels or less). This division into three scales is motivated by the distribution of sizes in the dataset, human performance and automotive system requirements.

In Fig. 5(a), we histogram the heights of the 350,000 BBs in our database using logarithmic sized bins. Cutoffs for the *near/far* scales are marked. Note that ∼68% of the pedestrians lie in the *medium* scale, and that the cutoffs for the *near/far* scales correspond to about ± 1 standard deviation from the mean height (in log space). One expects to see the number of pedestrians decrease with the square of their height, *i.e*. proportionally with their image area. The decrease at the other end, below 30 pixels, is due to annotators having difficulty identifying small pedestrians reliably.

Detection in the *medium* scale is essential for automotive



(a) Height distribution. (b) Distance vs height at 55 km/h.

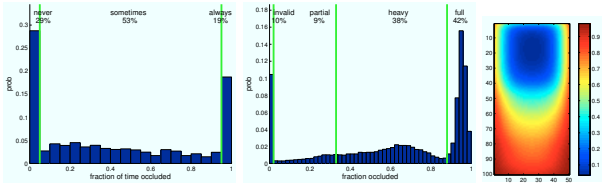
Figure 5. We define the *near* scale to include pedestrians 80 pixels or taller, the *medium* scale as 30-80 pixels, and the *far* scale as 30 pixels or less. Most pedestrians are observed at the *medium* scale, human performance is excellent at this scale, and for automotive applications detection must also occur at this scale. However, most current research targets the *near* scale and performance is poor even in the *medium* scale (see Sec. 4).

applications. We chose a camera setup that mirrors expected automotive applications: vertical field of view of 27°, resolution of 640×480, and focal length fixed at 7.5mm. Assuming 1.8m tall pedestrians, we can obtain an estimate of the distance to a pedestrian of observed pixel height h: $d \approx 1800/h$ m. With the vehicle traveling at an urban speed of 55 km/h (∼15 m/s), an 80 pixel person is just 1.5s away, while a 30 pixel person is 4s away (see 5(b)). Thus detecting *near* scale pedestrians may leave insufficient time to alert the driver, while *far* scale pedestrians are less relevant.

We shall use the *near*/*medium*/*far* distinction throughout this work. As described, most pedestrians are observed at the *medium* scale and for safety systems detection must also occur in this scale. Moreover, human performance is quite good in the *near* and *medium* scales but degrades noticeably at the *far* scale. However, most current algorithms are designed for the *near* scale and perform poorly even at the *medium* scale (see Sec. 4). Thus there is an important mismatch in current research efforts and the requirements of real world systems.

Occlusion: Little previous work has been done to quantify detection performance in the presence of occlusion (using real data). As described, occluded pedestrians are annotated with two BBs that denote the visible and full pedestrian extent. In Fig. 6(a), we plot frequency of occlusion, *i.e.*, for each pedestrian we measure the fraction of frames in which the pedestrian was occluded. The distribution has three distinct peaks: pedestrians that are never occluded (29%), occluded in some frames (53%) and occluded in all frames (19%). Note that over 70% of pedestrians are occluded in at least one frame.

For each occluded pedestrian, we can compute the fraction of occluded area as one minus the fraction of visible area over total area (calculated from the visible and full BBs). Aggregating, we obtain the histogram in Fig. 6(b). Over 80% occlusion typically indicates full occlusion, while 0% is used to indicate that a BB could not represent the extent of the visible region (*e.g.* due to a diagonal occluder). The interesting cases occur in between, which we further subdivide into *partial* occlusion (1-35% area occluded) and *heavy* occlusion (35-80% occluded).



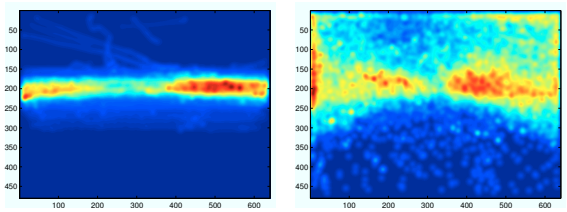
(a) Occlusion frequency. (b) Occlusion amount. (c) Occ. mask.

Figure 6. Occlusion statistics, see text for details.

Finally, in Fig. 6(c), we display a heat map that indicates which regions of a pedestrian were most likely to be occluded (obtained by averaging the occlusion masks). There is a strong bias for the lower portion of the pedestrian to be occluded and for the top portion to be visible, *i.e.* the typical occluder is lower to the ground. This bias contradicts the common assumption that probability of occlusion is uniform.

Position: Viewpoint and ground plane geometry (Fig. 2) constrain pedestrians to appear only in certain regions of the image. We compute the expected center position (over the 350,000 BBs) and plot the resulting heat map, log-normalized, in Fig. 7(a). As can be seen pedestrians are typically located in a narrow band running horizontally across the center of the image (y-coordinate varies somewhat with distance/height). Note that the same constraints are not valid when photographing a scene from arbitrary viewpoints, *e.g.* in the INRIA dataset.

In the collected data, many objects, not just pedestrians, tend to be concentrated in this same region. In Fig. 7(b) we show a heat map obtained by using BBs generated by the HOG [5] pedestrian detector with a low threshold. About half of the detections, including both true and false positives, occur in the same band as the true positives. Thus incorporating this constraint would considerably speed up detection but it would only moderately improve performance.



(a) Distribution of ped. position. (b) Distribution of HOG detections.

Figure 7. Expected center location of pedestrian BBs for (a) ground truth and (b) HOG detections. The heat maps are log-normalized, meaning pedestrian location is even more concentrated than immediately apparent.

2.3. Training and Testing Data

We split the database into training/testing data and specify our evaluation methodology. This will allow different research groups to compare their methods directly; as such, we urge authors to adhere to one of three training/testing scenarios described below.

Our data was captured over 11 sessions, each filmed in one of 5 city neighborhoods. We divide the data roughly in half, setting aside 6 sessions for training (0-5) and 5 for testing (sessions 6-10). For detailed statistics about the amount of training/testing data see bottom row of Table 1.

Here we focus on evaluating existing, pre-trained pedestrian detectors. Authors are encouraged to re-train their systems on our larger training set. We specify three training/testing scenarios:

- *Scenario-A:* Train on any *external* data, test on sessions 6-10. The results reported here use this setup as it allows for a broad survey of existing methods without any retraining.
- *Scenario-B*: Perform 6-fold cross validation using sessions 0-5. In each phase use 5 sessions for training and the 6^{th} for testing, then merge results on the validation sets and report performance on the entire training set (sessions 0-5).
- *Scenario-C*: Train using sessions 0-5, test on sessions 6-10.

We are *not* releasing the test data (sessions 6-10) at this time. Instead we ask authors to submit final, trained classifiers which we shall proceed to evaluate. Our aim is to help prevent overfitting and to extend the dataset's lifespan. Furthermore, it ensures that all algorithms are evaluated in precisely the same manner. Scenario-B allows authors to compare to other groups prior to having us evaluate using the full test set under Scenario-C.

2.4. Comparison to Existing Datasets

Existing datasets may be grouped into two types: (1) 'person' datasets containing people in unconstrained pose in a wide range of domains and (2) 'pedestrian' datasets containing upright people (standing or walking), typically viewed from more restricted viewpoints but often containing motion information and more complete labeling. The most widely used 'person' datasets include subsets of the MIT LabelMe data [29] and the PASCAL VOC datasets [28]. In this work we focus primarily on pedestrian detection, which is more relevant to certain applications including surveillance, robotics and automotive safety.

Table 1 provides a detailed overview of existing pedestrian datasets. Pedestrians can be labeled in photographs [5], surveillance video [26], and images taken from a mobile recording setup, such as a robot or vehicle [8]. Datasets gathered from photographs suffer from *selection bias*, as photographs must be manually chosen to contain only upright people and negative images are chosen according to arbitrary criteria, while surveillance videos have restricted backgrounds. Datasets collected with a mobile recording setup largely eliminate selection bias. In addition, unlike all previous pedestrian datasets, our dataset was not built to demonstrate the effectiveness of a particular method, and thus provides for an impartial, challenging test bed.

	Training			Testing			Height			Properties					
	# pedestrians	# neg. images	# pos. images	# pedestrians	# neg. images	# pos. images	10% quantile	median	90% quantile	color images	per-image ev.	no selec. bias	video seqs.	temporal corr.	occ. labels
MIT[27]	924	–	–	–	–	–	128	128	128	✓					
USC-A[43]	–	–	–	313	–	205	70	98	133		✓				
USC-B[43]	–	–	–	271	–	54	63	90	126		✓				
USC-C[44]	–	–	–	232	–	100	74	108	145		✓	✓			
CVC[14]	1000	6175†	–	–	–	–	46	83	164	✓		✓			
TUD-det[1]	400	–	400	311	–	250	133	218	278	✓	✓				
INRIA[5]	1208	1218	614	566	453	288	139	279	456	✓					
DC[24]	2.4k	15k†	–	1.6k	10k†	–	36	36	36			✓			
ETH[8]	2388	–	499	12k	–	1804	50	90	189	✓	✓	✓	✓		
Caltech	192k	61k	67k	155k	56k	65k	27	48	97	✓	✓	✓	✓	✓	✓

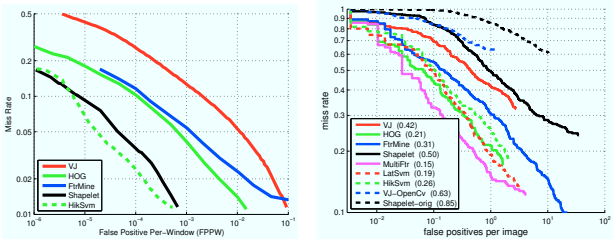
Table 1. Comparison of pedestrian datasets. The first six columns indicate the amount of training/testing data in each dataset, with 'k' used to denote thousands (1k=10^3). The columns are: number of unique pedestrian BBs (not counting reflections, shifts, *etc.*), number of images containing no pedestrians (a † indicates cropped negative BBs only), and number of images containing at least one pedestrian. Note that the proposed dataset is two orders of magnitude larger than existing datasets. The next three columns give the 10^{th} percentile, median and 90^{th} percentile pixel heights of the pedestrians, showing the range of scales found in each dataset. The final columns summarize additional properties of each dataset.

The INRIA dataset [5] has helped drive recent advances in pedestrian detection and remains the most widely used. However, it is biased toward large, mostly unoccluded pedestrians. The other most relevant datasets are the DaimlerChrysler (DC) [24] and ETH [8] datasets. The DC data, also captured in an urban setting, contains only very small, cropped pedestrians. The ETH data, captured using a pair of cameras attached to a stroller, has reasonable scale variation and a significant amount of labeled data; however, occlusions are not annotated and each frame is labeled independently.

We conclude by summarizing the most novel and important aspects of the Caltech Pedestrian Dataset. It includes O(10^5) pedestrian BBs labeled in O(10^5) frames, two orders of magnitude more than any other dataset. The dataset includes color video sequences and contains pedestrians with a large range of scales and more pose variability than typical pedestrian datasets. Finally, as far as we know, this is the first dataset with temporal correspondence between BBs and detailed occlusion labels.

3. Evaluation Methodology

The established methodology for evaluating pedestrian detectors is flawed. Most authors compare *per-window* performance, *e.g.* this is the accepted methodology for the INRIA dataset [5], as opposed to the *per-image* measures frequently used in object detection [28]. In real applications,



(a) INRIA per-window results. (b) INRIA per-image results.

Figure 8. Results on the INRIA datasets (each algorithm is described in more detail in Sec. 4). The per-window results, when available, are reproduced from the original publications (the VJ curve is extracted from [5]). Typically results are reported on cropped positives, but the INRIA dataset also contains full images with the same pedestrians but within the original context. We computed the per-image results using the 288 full images (each containing at least one pedestrian) and the methodology described in Sec. 3.1. Note the reordering of the classification performance between the per-window and per-image results.

a per-window detector is densely scanned across an image and nearby detections merged, *e.g.* using non maximal suppression (NMS). Instead, Dalal & Triggs suggest evaluating a detector by classifying cropped windows centered on pedestrians against windows sampled at a fixed density from images without pedestrians, thus avoiding NMS or other post processing. The typical assumption is that better per-window scores will lead to better performance on entire images; however, in practice per-window performance can fail to predict per-image performance (see Fig. 8).

There may be a number of explanations. Per-window evaluation does not measure errors caused by detections at incorrect scales or positions or arising from false detections on body parts, nor does it take into account the effect of NMS (which can reduce false positives at varying rates for different methods). Detectors may require different sampling schemes [36], particularly those that are somewhat invariant to changes in position and scale; furthermore, there can be complex interactions between sampling density and NMS. Together, these factors make evaluating a classifier independently of the overall detection scheme difficult.

Of course, not all detection systems are based on sliding windows [19, 17], and per-window evaluation of such systems is impossible. Perhaps the biggest pitfall of the per-window scheme pertains to use of cropped positives and uncropped negatives for training and testing: classifiers may exploit window boundary effects as discriminative features leading to good per-window performance but poor per-image performance. We observed this in two of the algorithms evaluated [30, 21] [2].

[2]Both groups have acknowledged this. *E.g.*, see the advisory posted at Mori's website: `www.cs.sfu.ca/~mori/research/papers/sabzmeydani_shapelet_cvpr07.html`. For both algorithms we evaluate updated, corrected versions.

3.1. Per-image evaluation

We perform single frame evaluation using a modified version of the scheme laid out in the PASCAL object detection challenges [28]. A detection system needs to take in an image and return a BB and a score or confidence for each detection. The system should perform multiscale detection and any necessary NMS or other post processing. Evaluation is performed on the final generated list of detected BBs.

A detected BB (BB_{dt}) and a ground truth BB (BB_{gt}) form a potential match if their areas overlap sufficiently. Specifically, we employ the PASCAL measure, which states that the area of overlap a_0 must exceed 50%:

$$a_o = \frac{area(BB_{dt} \cap BB_{gt})}{area(BB_{dt} \cup BB_{gt})} > 0.5 \quad (1)$$

The threshold of 50% is arbitrary but reasonable.

Each BB_{dt} and BB_{gt} may be matched at most once. We resolve any assignment ambiguity by matching detections with highest confidence first. In rare cases this assignment may be suboptimal, especially in crowded scenes [32], but in practice the effect should be negligible. Unmatched BB_{dt} count as false positives and unmatched BB_{gt} as false negatives. To compare methods we plot miss rate against false positives per-image by varying the threshold on detection confidence. This is preferred to precision recall curves for certain tasks, *e.g.* automotive applications, as typically there is an upper limit on the acceptable false positives per-image rate independent of pedestrian density.

To evaluate performance on different subsets of the ground truth, we introduce the notion of *ignore* regions. Ground truth BBs selected to be ignored, denoted using BB_{ig}, need not be matched, however, matches are not considered mistakes either. *E.g.*, to evaluate performance on unoccluded pedestrians, we set all BBs that contain occluded pedestrians to ignore. Matching proceeds as before, except BB_{dt} matched to BB_{ig} do *not* count as true positives, and unmatched BB_{ig} do *not* count as a false negatives (matches to BB_{gt} are therefore preferred). Note that setting a BB to ignore is not the same as removing that BB from the ground truth; in the latter case detections in the ignore regions would count as false positives.

Four types of ground truth are always set to ignore: any BB_{gt} under 20 pixels high or near image borders (to avoid boundary effects), containing a 'Person?' (difficult or ambiguous cases), or containing 'People'. In addition, each 'People' BB is broken down into multiple overlapping BB_{ig}, each having the same height as the 'People' BB. Detections in these regions do not affect performance.

We conclude by listing additional details. Some detectors output BBs with padding around the pedestrian (*e.g.* HOG outputs 128×64 BBs around 96 pixel tall people), such BBs are cropped appropriately. Methods usually detect pedestrians at some minimum size, to coax smaller de-

	low-level features	classifier	original implementation	trained on INRIA data	per-image evaluation	sec / frame	model height (in pixels)	scale stride	publication
VJ[40]	Haar	AdaBoost		✓		7.0	96	1.05	'04
HOG[5]	HOG	linear SVM	✓	✓		13.3	96	1.05	'05
FtrMine[7]	gen. Haar	AdaBoost	✓	✓		45	96	1.20	'07
Shapelet[30]	gradients	AdaBoost		✓		60.1	96	1.05	'07
MultiFtr[42]	HOG+Haar	AdaBoost	✓	✓	✓	18.9	96	1.05	'08
LatSvm[11]	HOG	latent SVM	✓		✓	6.3	80	1.05	'08
HikSvm[21]	HOG-like	HIK SVM	✓	✓		140	96	1.20	'08

Table 2. Overview of tested algorithms. All approaches use sliding windows and NMS (all except LatSvm use kernel density estimation for NMS, as proposed in [4]). All use variants of HOG or Haar features and are trained with variations of boosting or SVMs. LatSvm was trained on the PASCAL dataset, the others on the INRIA pedestrian dataset. Only LatSvm and MultiFtr reported results using per-image measures, the rest of the algorithms were originally evaluated using per-window measures. Runtime per 640×480 image, model height used for training and the scale stride used for testing are also listed. The tested implementations of Shapelet and HikSvm have been corrected so they no longer overfit to boundary effects (see Sec. 3). Due to time and memory constraints, we were unable to run HikSvm on upscaled images. This adversely affects HikSvm's overall performance as small pedestrians are not detected.

tections we upscale the input images. For ground truth, the full BB is always used for matching, not the visible BB, even for partially occluded pedestrians. Finally, all reported results are computed using every 30^{th} frame in the test data.

4. Evaluation Results

To measure performance we evaluated seven promising pedestrian detectors (Table 2). We obtained the detectors directly from their authors, the only exceptions were the VJ and Shapelet detectors which were reimplemented in [42] (these outperform the OpenCV VJ code and the original Shapelet code, respectively, see Fig. 8(b)). We focus on evaluating existing, pre-trained pedestrian detectors (Scenario-A described in Sec. 2.3). We use the evaluation methodology outlined in Sec. 3.1, plotting miss rate versus false positives per-image (FPPI) in log-log scale (lower curves indicate better performance). We use the miss rate at 1 FPPI as a common reference point to compare results (note that on average there are 1.4 pedestrians per image).

Overall: We begin by plotting results on the entire dataset in Fig. 9(a). MultiFtr outperforms the remaining methods, with HOG as a close second. However, absolute performance is quite poor, with a miss rate of over 80% at 1 FPPI. Performance should improve somewhat upon retraining. To understand where the methods fail we examine performance on varying subsets of the data.

Scale: As discussed in Sec. 2.2, we group pedestrians according to their pixel height into the *near* (80 or more pixels), *medium* (30-80 pixels) and *far* (30 pixels or less) scales. Results for each scale, on unoccluded pedestrians only, are shown in Fig. 9(d)-9(f). For unoccluded

Figure 10. Selected HOG false negatives (left) and high confidence false positives (right) for *near* scale unoccluded pedestrians.

near pedestrians, purely gradient based detectors such as HikSvm, LatSvm and especially HOG perform best, with a miss rate under 40% at 1 FPPI. At the *medium* scale, which contains over 68% of the annotated pedestrians, MultiFtr achieves the best relative performance but absolute performance is quite poor with 72% miss rate at 1 FPPI. HOG performs similarly at this scale. At the *far* scale performance is rather abysmal; none of the algorithms is able to achieve more than 8% recall at 1 FPPI. Results for HikSvm at the medium/far scales are not shown (see Table 2 for details).

Occlusion: The impact of occlusion on detecting pedestrians with a minimum height of 50 pixels is shown in Fig. 9(g)-9(i). As discussed in Sec. 2.2, we classify pedestrians as unoccluded, *partially* occluded (1-35% area occluded) and *heavily* occluded (35-80% occluded). Performance drops significantly even under partial occlusion, leading to a maximum recall of slightly under 30% at 1 FPPI achieved by MultiFtr. For heavy occlusion the situation becomes worse, with maximum recall dropping to 7% at 1 FPPI. Note that LatSvm, which is part-based, degrades least.

Aspect ratio: The mean aspect ratio of BBs in the proposed dataset is about 0.43 and has a standard deviation of 0.1. Atypical aspect ratios (outside of one standard deviation) frequently correspond to variations in viewpoint and pose. Results on unoccluded, 50 pixel or taller pedestrians, split according to aspect ratio, are shown in Fig. 9(b) and 9(c). Performs clearly degrades for atypical aspect ratios, from a maximum recall of about 56% at 1 FPPI on typical aspect ratios, achieved by HOG, to about 40% recall achieved by both HOG and MultiFtr. However, the impact is not as severe as for occlusion and scale.

Summary: HOG, MultiFtr and FtrMine tend to outperform the other methods surveyed. VJ and Shapelet perform poorly. LatSvm likely suffers from being trained on the Pascal dataset, while results for HikSvm are artificially depressed since small people are not detected. HOG performs best on *near*, unoccluded pedestrians (typical errors are shown in Fig. 10). MultiFtr ties or outperforms HOG on more difficult cases (smaller scales, occlusion, atypical aspect ratios), and as these comprise the bulk of the dataset MultiFtr achieves a slightly higher overall performance. However, absolute performance in these cases is still poor.

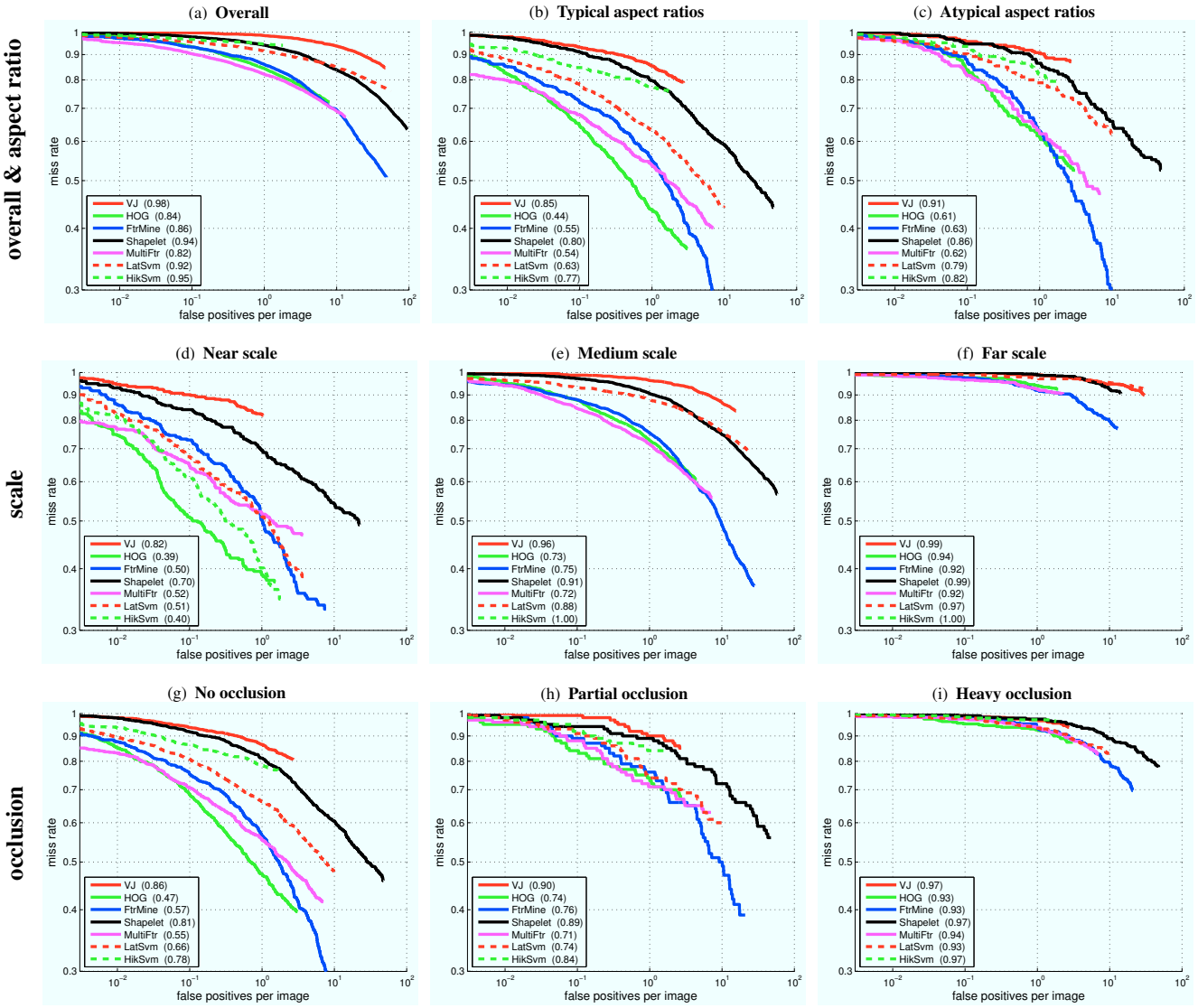


Figure 9. Miss rates versus false positive per-image curves shown for various subsets of the data. Lower curves indicate better performance; miss rate at 1 FPPI for each algorithm is shown in plot legends. (a) Overall performance on the entire dataset. (b-c) Performance w.r.t. aspect ratio (computed for unoccluded pedestrians 50 pixels or taller). (d-f): Performance w.r.t. scale (computed for unoccluded pedestrians). (g-i): Performance under varying levels of occlusion (computed for pedestrians 50 pixels or taller). Due to time and memory constraints, we were unable to run HikSvm on upscaled images; this adversely affects HikSvm's performance on many of the plots shown.

5. Discussion and Future Work

We introduced the large, richly annotated Caltech Pedestrian Dataset for training and evaluating pedestrian detectors and benchmarked a number of promising methods. Although recent literature would suggest otherwise, our analysis shows that HOG remains competitive, especially when properly benchmarked (using per-image metrics).

For unoccluded pedestrians over 80 pixels high, HOG achieves 60% recall at 1 FPPI on the proposed dataset (see Fig. 9(d)). This is worse but comparable to the 80% recall at 1 FPPI on the INRIA data on which HOG was trained. Under these conditions performance is reasonable but still below levels necessary for real world applications.

Under more realistic and challenging conditions, performance degrades rapidly. Two under explored cases stand out as being particularly frequent and relevant in the data gathered: pedestrians at lower resolution and under partial occlusion. Note that pedestrians in the *medium/far* scales represent more than 80% of the data; furthermore, in automotive tasks it is crucial to identify pedestrians early to give ample warning to the driver. Occlusion is likewise common, only 30% of pedestrians remain unoccluded for the entire time they are present. Yet, as our analysis has shown, these are precisely the tasks for which current methods fail. Fur-

ther research addressing detection at smaller scales and of partially occluded pedestrians is crucial.

A number of cues should help improve performance at low resolutions and under occlusion. The first of these is context, both spatial [9, 15] and temporal [4]. Discriminative part-based approaches [6, 11] may also provide more robustness to occlusion, although those may be ill-suited for low resolution pedestrians.

We are planning to extend our benchmark to explore two more issues. Of primary importance is to repeat the evaluation of each algorithm after re-training on our dataset (Scenario-C). We are also interested in evaluating detectors that utilize features computed over 2-4 frames [41, 4] and also algorithms that integrate information over longer time scales. The database, annotation tool and evaluation code are available on the project website.

Acknowledgments: We would like to thank Eugene Bart, Ryan Gomes and Mohamed Aly for valuable help and feedback, and Irina Bart for her many long hours labeling small pedestrians. This work was partially supported by the Office of Naval Research grant N00014-06-1-0734 and a gift from an automobile manufacturer who wishes to remain anonymous.

References

[1] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *CVPR*, 2008.

[2] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. Black, and R. Szeliski. A database and eval. methodology for optical flow. In *ICCV*, 2007.

[3] J. L. Barron, D. J. Fleet, S. S. Beauchemin, and T. A. Burkitt. Performance of optical flow techniques. *IJCV*, 12(1):43–77, 1994.

[4] N. Dalal. *Finding People in Images and Videos*. PhD thesis, Institut National Polytechnique de Grenoble, 2006.

[5] N. Dalal and B. Triggs. Histogram of oriented gradient for human detection. In *CVPR*, 2005.

[6] P. Dollár, B. Babenko, S. Belongie, P. Perona, and Z. Tu. Multiple component learning for object detection. In *ECCV*, 2008.

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

[8] A. Ess, B. Leibe, and L. V. Gool. Depth and appearance for mobile scene analysis. In *ICCV*, 2007.

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

[10] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *PAMI*, 28(4):594–611, 2006.

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

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

[13] D. M. Gavrila and V. Philomin. Real-time object detection for "smart" vehicles. In *ICCV*, 1999.

[14] D. Gerónimo, A. Sappa, A. López, and D. Ponsa. Adaptive image sampling and windows classification for on-board pedestrian detection. In *Inter. Conf. on Computer Vision Systems*, 2005.

[15] D. Hoiem, A. A. Efros, and M. Hebert. Putting objects in perspective. In *CVPR*, volume 2, pages 2137 – 2144, 2006.

[16] S. Ioffe and D. A. Forsyth. Probabilistic methods for finding people. *IJCV*, pages 45–68, 2001.

[17] C. H. Lampert, M. B. Blaschko, and T. Hofmann. Beyond sliding windows: Object localization by efficient subwindow search. In *CVPR*, 2008.

[18] B. Leibe, N. Cornelis, K. Cornelis, and L. V. Gool. Dynamic 3D scene analysis from a moving vehicle. In *CVPR*, 2007.

[19] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segm. *IJCV*, pages 259–289, 2008.

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

[21] S. Maji, A. Berg, and J. Malik. Classification using intersection kernel SVMs is efficient. In *CVPR*, 2008.

[22] D. Martin, C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *PAMI*, 26(5):530–549, 2004.

[23] K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a prob. assembly of robust part det. In *ECCV*, 2004.

[24] S. Munder and D. M. Gavrila. An experimental study on pedestrian classification. *PAMI*, pages 1863–1868, 2006.

[25] S. Munder, C. Schnörr, and D. Gavrila. Pedestrian detection and tracking using a mixture of view-based shape-texture models. In *IEEE Transactions on Intelligent Transportation Systems*, 2008.

[26] A. T. Nghiem, F. Bremond, M. Thonnat, and V. Valentin. ETISEO, performance eval. for video surveillance systems. In *AVSS*, 2007.

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

[28] J. Ponce, T. Berg, M. Everingham, D. Forsyth, M. Hebert, S. Lazebnik, M. Marszałek, C. Schmid, C. Russell, A. Torralba, C. Williams, J. Zhang, and A. Zisserman. Dataset issues in object rec. In *Towards Category-Level Object Rec.*, pages 29–48. Springer, 2006.

[29] B. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. LabelMe: A database and web-based tool for image annotation. *IJCV*, 77(1-3):157–173, 2008.

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

[31] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47:7–42, 2002.

[32] E. Seemann, M. Fritz, and B. Schiele. Towards robust pedestrian detection in crowded image sequences. In *CVPR*, 2007.

[33] V. Sharma and J. Davis. Integrating appearance and motion cues for simultaneous detection and segmentation of ped. In *ICCV*, 2007.

[34] A. Shashua, Y. Gdalyahu, and G. Hayun. Pedestrian detection for driving assistance systems: single-frame classification and system level performance. In *Intelligent Vehicles Symposium*, 2004.

[35] Y. Song, X. Feng, and P. Perona. Towards detection of human motion. In *CVPR*, 2000.

[36] D. Tran and D. Forsyth. Configuration estimates improve pedestrian finding. In *NIPS*, volume 20, 2008.

[37] T. Tsukiyama and Y. Shirai. Detection of the movements of persons from a sparse sequence of tv images. *PR*, 18(3-4):207–213, 1985.

[38] O. Tuzel, F. M. Porikli, and P. Meer. Pedestrian det. via classification on riemannian manifolds. *PAMI*, 30(10):1713–1727, 2008.

[39] United Nations Economic Commission for Europe. *Statistics of road traffic accidents in Europe and North America*. Switzerland, 2005.

[40] P. Viola and M. Jones. Robust real-time object detection. *IJCV*, 57(2):137–154, 2004.

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

[42] C. Wojek and B. Schiele. A performance evaluation of single and multi-feature people detection. In *DAGM*, 2008.

[43] B. Wu and R. Nevatia. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In *ICCV*, 2005.

[44] B. Wu and R. Nevatia. Cluster boosted tree classifier for multi-view, multi-pose object detection. In *ICCV*, 2007.

[45] H. Yang, M. Pollefeys, G. Welch, J. Frahm, and A. Ilie. Differential camera tracking through linearizing the local appearance manifold. In *CVPR*, 2007.