Detection of Sudden Pedestrian Crossings for Driving Assistance Systems

Yanwu Xu, *Member, IEEE*, Dong Xu, *Member, IEEE*, Stephen Lin, *Member, IEEE*, Tony X. Han, *Member, IEEE*, Xianbin Cao, *Senior Member, IEEE*, and Xuelong Li, *Senior Member, IEEE*

***Abstract*—In this paper, we study the problem of detecting sudden pedestrian crossings to assist drivers in avoiding accidents. This application has two major requirements: to detect crossing pedestrians as early as possible just as they enter the view of the car-mounted camera and to maintain a false alarm rate as low as possible for practical purposes. Although many current sliding-window-based approaches using various features and classification algorithms have been proposed for image-/video-based pedestrian detection, their performance in terms of accuracy and processing speed falls far short of practical application requirements. To address this problem, we propose a three-level coarse-to-fine video-based framework that detects partially visible pedestrians just as they enter the camera view, with low false alarm rate and high speed. The framework is tested on a new collection of high-resolution videos captured from a moving vehicle and yields a performance better than that of state-of-the-art pedestrian detection while running at a frame rate of 55 fps.**

***Index Terms*—Coarse to fine, pedestrian detection, performance evaluation, spatiotemporal refinement, sudden pedestrian crossing.**

I. Introduction

HUMAN ACTION and activity detection/analysis [13], [23] has attracted much attention in computer vision because its of wide-range applications, including surveillance [12], [15], [24], [29], [33], robotics [3], content-based image/video retrieval, video annotation, assisted living, intelligent vehicles [32], and advanced user interfaces [11], [20]. In this paper, we address a particular problem in this area that can have a significant impact on people's lives, namely, the detection of sudden pedestrian crossings to assist drivers in accident avoidance. Our work is motivated by two factors. One is that the proposed problem has great social meaning and application value. According to the traffic safety data from the National Highway Traffic Safety Administration [26] and the EU [11], many people are killed/injured each year in pedestrian–motor collisions; most of which occur when pedestrians attempt a road crossing at nonintersections. Second, the proposed problem has special requirements that make it different from existing related research. Drivers must be alerted to crossing pedestrians as early as possible for evasive maneuvers to be most effective. For this, crossing pedestrians should be identified even before they come into full view. The need for a combination of high processing speed, detection of partially visible pedestrians as they enter the scene, rejection of static or exiting pedestrians, and handling of unconstrained camera motion distinguishes this application from related work on event/action detection, generic (image/video based) pedestrian detection [7], and even current methods in intelligent vehicle systems [10], [11].

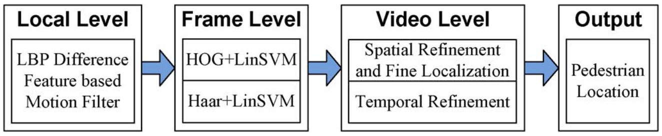


Fig. 1. Flowchart of the proposed three-level framework.

In this paper, we address the aforementioned issues with a proposed three-level coarse-to-fine framework, shown in Fig. 1. A natural approach to this problem is to locate pedestrians and then determine from motion analysis whether he/she is entering the road scene. In the first stage, we perform analysis at a local level by employing sparse sliding window sampling with a novel local binary pattern (LBP) difference-based motion filter to rapidly identify image regions containing possible pedestrian motion. Windows pass the first stage and then undergo further verification at the frame level in the second stage using a pair of generic pedestrian detectors trained with half-sized pedestrian samples. Lastly, in the third stage, appearance and motion-based spatiotemporal refinement is computed at the video level to pinpoint pedestrian locations and to reduce false alarms, including static and exiting pedestrians. This coarse-to-fine approach, together with cascaded classifiers and a sparse sampling strategy, is shown to yield high accuracy while running at real-time frame rates.

Manuscript received July 4, 2010; revised December 8, 2010, May 30, 2011, and August 13, 2011; accepted August 24, 2011. Date of publication November 30, 2011; date of current version May 16, 2012. This work was supported by the National Basic Research Program of China (973 Program) (Grant 2012CB316400), by the Singapore A$*$STAR SERC Grant (082 101 0018), by the National Natural Science Foundation of China (Grants 61125106, 60972103, and 61072093), and by the Open Project Foundation of State Key Laboratory of Industrial Control Technology, Zhejiang University (Grant ICT1105). This paper was recommended by Associate Editor S. X. Yang.

Y. Xu and D. Xu are with the School of Computer Engineering, Nanyang Technological University, Singapore 639798 (e-mail: ywxu@ntu.edu.sg; dongxu@ntu.edu.sg).

S. Lin is with Microsoft Research Asia, Beijing 100080, China (e-mail: stevelin@microsoft.com).

T. X. Han is with the Department of Electrical and Computer Engineering, University of Missouri, Columbia, MO 65211 USA (e-mail: hantx@missouri.edu).

X. Cao is with the School of Electrical Information Engineering, Beihang University, Beijing 100191, China (e-mail: xbcao@buaa.edu.cn).

X. Li is with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, China (e-mail: xuelong_li@opt.ac.cn).

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

Digital Object Identifier 10.1109/TSMCB.2011.2175726

The proposed framework is tested on a newly collected data set consisting of high-resolution videos of sudden pedestrian crossings captured from a moving vehicle. Unlike data sets used for general onboard pedestrian detection [6], [7], [31], the examined application requires high-resolution data since it generally deals with pedestrians with an otherwise small image footprint, e.g., side-views of partially visible pedestrians and young children who may rush out onto a street unaware of oncoming traffic. With the rapidly decreasing costs of high-resolution cameras, we believe that they will be the standard in intelligent vehicle systems. In addition to the new data set, we propose novel evaluation criteria that are also targeted to our application.

In summary, the major contributions of this paper are as follows: the first systematic study on sudden pedestrian crossings; the development of a real-time algorithm; the collection of a new data set suitable for this problem; and the new evaluation criteria. With this approach, we obtained a performance of 73% true positive detection with 0.01 false positives per frame, which is an improvement over state-of-the-art image-based pedestrian detection techniques [6], [7], even though our method processes only partially visible pedestrians and provides real-time performance. The processing speed of our system is 55 fps, which is both suitable for practical applications in urban areas without high vehicle speeds and considerably faster than computing only dense histogram of oriented gradient (HOG) features alone (0.75 fps).

II. Previous Work on Image-/Video-Based Pedestrian Detection

Research on pedestrian detection can be roughly categorized into two types according to the data source, i.e., image based and video based. Detecting pedestrians in images is a challenging task that has attracted much attention in computer vision since only limited static appearance information can be used. In contrast, video-based detection benefits from additional information in the forms of motion data and depth data, which can be used to efficiently infer regions of interest (ROIs) and can be used together with appearance features to increase classification accuracy. However, both of these approaches share the same general framework, including candidate selection, feature representation, classification, and final decision. Of course, some preprocessing (e.g., image smoothing or enhancement and video stabilization) and postprocessing (e.g., tracking) are essential in achieving high performance. Similar to other object detection problems, features (representation, selection, and dimensionality reduction) and classification algorithm highly influence detection precision and play important roles in the whole system. Candidate selection is usually used to balance speed and accuracy, i.e., by quickly filtering out most of the nonpedestrian areas. The final decision is made to further improve precision by removing redundant detections. We review related work from these four aspects in the following sections.

A. Candidates

In the literature, sliding-window-based approaches have been shown to outperform others and have become the predominant method at present [28]. Sliding windows at various scales and locations are first examined to detect ROIs based on certain features, which may be global [1], [11], [17], [19], [20], [22], [25] or local [2], [21] and single [4], [27] or multiple [30]. Then, single or multiple classifiers (e.g., support vector machines (SVMs) and variants of boosting algorithms) are used to judge whether the sliding windows in ROIs bound a person or not.

Since the basic sliding window method performs an exhaustive search, it may obtain a higher accuracy; however, the speed is comparatively low. To elevate speed, some candidate selection methods are used to roughly locate the ROIs that have a higher possibility of containing pedestrians. For example, Gavrila *et al.* proposed a chamfer system based on edge template matching to locate candidate regions before neural network classification [10]. Huang *et al.* used stereo segmentation to obtain the candidate region [38]. In the widely used OpenCV toolbox, Canny pruning is used as the candidate selection module for Haar-like feature-based pedestrian detection [39]. Furthermore, tracking previous detection results can also be treated as a form of candidate selection.

B. Features

Various features have been used for pedestrian detection, which can be roughly categorized into appearance/static features and motion features. For image-based methods, only appearance features can be used, while motion features usually provide more information for video-based pedestrian detection. For example, Viola *et al.* [27] used Haar-like features to represent both appearance and motion and trained a cascaded Adaboost classifier to detect walking pedestrians. Dalal *et al.* [5] extracted oriented histograms of flow from two sequential frames to serve as motion features, which are used together with HOG-based appearance features to improve detection accuracy. However, motion information is difficult to use since it also brings in more noise, especially in unconstrained videos; therefore, some existing systems use only static information to detect pedestrians in videos [28], [43].

We can also categorize existing pedestrian detection systems according to the number of feature types adopted. Single features such as edges [11], shapelets [25], histograms of image patches (e.g., LBP and HOG) [1], [4], [17], [19], local representative fields [7], [20], wavelet coefficients [22], and Haar-like features [27] are widely used for pedestrian detection and general object detection in the literature. Many methods have also been proposed to combine several types of features. For example, different kinds of histogram features can be directly joined to form new features [28]. Different features can be used to train several classifiers individually, and a final decision is made by majority voting or in a cascaded manner [5]. Different features can be selected and integrated by boosting [30], [43], and a single/cascaded classifier can be obtained.

Moreover, features can be categorized into global ones and local ones. Global features are extracted directly from each sliding window or sample image, while local features are extracted by dividing a sliding window into different subregions [40], where each region can be taken as a unit from which to extract one or more kinds of features [2], [21].

C. Classification

Currently, for classification algorithms, supervised learning methods are the dominant approaches. A classifier is trained offline using completely labeled training data, where sliding windows which bound a person form positive samples, while the others are negative samples. Typically, SVMs, especially linear SVMs (LinSVM) and variants of boosting algorithms, are used because of their high classification accuracy and efficiency. Bootstrapping is also frequently employed since it has been empirically proven to improve the generalization capacity of classifiers [20], [30]. In recent years, active learning [41] and multiple instance learning [42] have been introduced as suitable techniques for pedestrian detection.

Features and classifiers may have some frequently used combinations according to their characteristics. Usually, high-dimensional features are combined with boosting classifiers, which can identify a small number of optimal features during classifier training, while low-dimensional features are combined with an effective linear SVM classifier. For further details on features and classification algorithms for pedestrian detection, readers are referred to a recent survey on this topic [7].

D. Decision

Final results are typically determined after employing non-maximal suppression (NMS) to merge overlapping windows, which can greatly reduce false alarms at the image level. To further improve system performance at the video level, e.g., with greater robustness to camera motion, tracking methods such as Kalman filtering and mean shift tracking [34]–[37] have been used in many vision systems. Moreover, tracking by detection has achieved higher accuracy recently. For example, overlapping spatiotemporal windows detected in each frame have been merged in order to reduce false alarms at the trajectory level [7]. The final decision is made by evaluating each trajectory; therefore, tracking accuracy is heavily reliant on image-based detection.

E. Differences of Our Work

Our problem of detecting sudden pedestrian crossings differs from generic image-/video-based pedestrian detection in two significant respects. One is the necessity of fast processing to alert the driver as early as possible. Although complex appearance and motion features from densely sampled sliding windows have often been used to guarantee a high detection rate, this leads to very low detection speeds that are unsuitable for our real-time application. We observe that, for the purpose of alerting drivers, a pedestrian need not be detected in every frame in which he/she appears; therefore, we propose a sparse sliding window scanning strategy to speed up the detection process. In addition, our system focuses on partially visible pedestrians before they have even entered into full view of the camera.

Another difference from previous works is that our system aims to alert drivers only to sudden pedestrian crossings *that may take the driver by surprise*. It is in these cases that a warning alarm is most critical. We disregard static pedestrians and pedestrians that cross at a far distance from the camera, which drivers should be able to see with plenty of time to react. Our system considers only pedestrians that exhibit greater motion than the background over the course of several frames. Toward this end, we propose a new LBP difference-based motion filter to discard regions that lack significant motion in order to greatly speed up detection without decreasing the detection rate. By considering only pedestrians crossing near to the camera, we can assume that the sudden pedestrian crossing events originate at the sides of the camera view, which limits the area of each video frame that needs to be examined. By substantially reducing the search space in this manner, it is possible to integrate various computationally expensive features without a major sacrifice in processing speed.

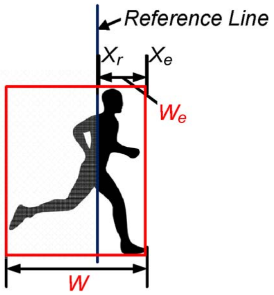


Fig. 2. Notation for a pedestrian crossing event.

III. Definition of Pedestrian Crossing Event

In this paper, a pedestrian crossing event is defined as a spatiotemporal volume that encompasses a given range of pedestrian visibility as he/she enters the camera view. To quantify pedestrian visibility, we first define the pedestrian entering ratio. As shown in Fig. 2, when a pedestrian enters the camera view from the left side, the entering ratio α is defined as

$$\alpha = \frac{X_e - X_r}{W} \tag{1}$$

where X_e is the X-axis value of the *right edge* of the pedestrian's bounding box, X_r is the X-axis value of a vertical *reference line*, and W is the horizontal width of the bounding box. In real applications of our system, the reference line is taken as the left/right edge of each video frame. However, for testing our framework, the reference line may be placed within the frame so that the bounding box is completely visible and the actual entering ratios are known for evaluation purposes. We also define $W_e = X_e - X_r$ as the *entering width*. We utilize this arbitrary definition of entering ratio to more easily account for variations in entering style, shown in Fig. 3.

Based on the definition of entering ratio α, the spatiotemporal cubic of a pedestrian crossing event starts from a predefined threshold α_e and ends when the entering ratio reaches a certain threshold α_l.

IV. Efficient Spatiotemporal Coarse-to-Fine Framework

To detect pedestrian crossing events, we propose a three-level coarse-to-fine approach based on sliding windows. As shown in Fig. 1, the levels are defined as the following: 1) sparse

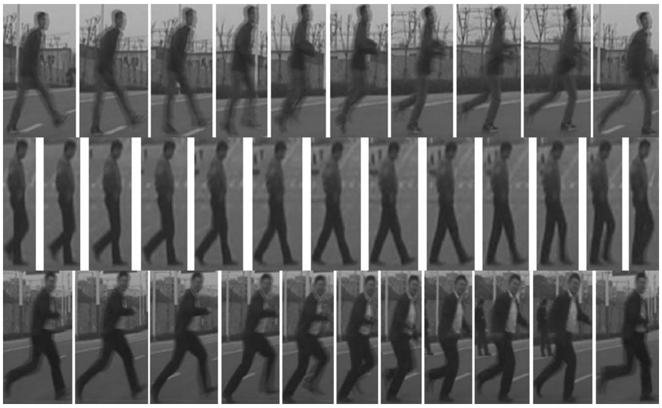


Fig. 3. Examples of various pedestrian entering styles. The first row illustrates a pedestrian jumping into the camera view, the second row illustrates a strolling entrance, and the third row exhibits running.

sliding window sampling with a motion filter based on a new LBP difference feature at the local level; 2) coarse detection and rough localization at the frame level; and 3) spatiotemporal refinement and fine localization at the video level. Each of these levels is described in the following sections.

A. Local Level

At the local level, we search for pedestrians using a bundle of sliding windows at different scales. For efficient yet accurate operation of sliding windows, we take advantage of three properties of sudden pedestrian crossings. Since a crossing pedestrian need not be detected in every frame of the event in order to issue an alert to the driver, a sparse sliding window scanning strategy is used to improve processing speed. In addition to this, only sliding windows that contain significant motion according to an LBP difference feature need to be considered since crossing pedestrians must be moving. Third, our system examines windows only at the left or right edges of each frame, where sudden crossings of nearby pedestrians are presumed to begin.

For sliding windows, typical image-based methods use fine sampling of scales and shifts, and then merge the dense detection results using NMS to obtain precise pedestrian locations. We refer to this as a *fine-to-fine* approach, which is computationally slow and unsuitable for real-time onboard detection with high-resolution images. For our application of video-based detection of sudden pedestrian crossing events, we instead employ a *coarse-to-fine* approach. At the local level, an approximate location of a pedestrian is found using a coarse sampling of window scales and shifts. Then, later at the video level, the pedestrian position is refined. In our implementation, we use sliding windows with sizes that range from 128 × 32 to 512 × 128, with a scaling factor of $\rho = 1.25$ and with shifts of 1/8 of the window height. The large window shifts are feasible in our application since it is unnecessary to detect each frame of a pedestrian as he/she is crossing, and vehicle motion often introduces vertical displacements of pedestrians in different frames.

In this paper, a new LBP [1], [28] difference feature is used to detect regions with significant motion. For each frame, the $LBP_{8,1}$ transformation is performed on each pixel in the ROI that covers all of the sliding windows, and then, the LBP histogram of each of these windows is efficiently calculated using integral images [27], which are computed concurrently with the LBP transformation and is normalized by its L1-norm. The histogram is subtracted from a cached LBP histogram of a previous frame to obtain the LBP difference. If the magnitude of the LBP difference in a sliding window is larger than a threshold f, which is empirically set in our experiments, this sliding window will be tagged for further processing, and the cached LBP histogram of this sliding window is replaced by the present one. The cached histogram is also replaced by the current one if its corresponding window was not tagged in the previous θ_f frames (six in our implementation).

For our method, the LBP difference-based motion filter has several advantages over alternative motion filters. In contrast to basic pixel-based frame subtraction, the LBP difference filter is unaffected by illumination variation and some amount of rotation. Moreover, compared with tracking or motion estimation methods such as Kalman filtering and mean shift tracking, which may fail or have very low accuracy when there is a large sudden motion, this LBP filter can also handle fast motion. At the same time, this method of sparse sampling does not degrade the positive detection rate, as we will demonstrate in Section VII-B.

B. Frame Level

The sliding windows that pass the motion filter are then passed to a cascade of classifiers for coarse detection and rough localization. First, a HOG+LinSVM [30] classifier is used because HOG [4] and HOG-like [14], [17] features have demonstrated great success in various object detection problems, especially in pedestrian detection. For HOG feature extraction, the gradient of each pixel in the windows is computed, and then, the gradient magnitude is inserted into one of the nine histogram bins that span a 180° range. In these histograms, an 8 × 8 cell size is used, and 2 × 2 cells form a block. Each block half overlaps each of its neighbors and is normalized using the L2-norm. The final HOG vector is composed of all normalized block histograms, with a total dimension of 1620 for the 128 × 32 sliding windows employed in our work. Since the Y-axis shift of our sliding window sampling strategy is exactly one block size, the computation of HOG features can be expedited by saving and reusing the feature computation and normalization among overlapping blocks.

Since HOG features represent only edge information, we additionally utilize a texture feature to obtain lower false positive rates. As reported in [30], the combination of HOG and Haar wavelets achieves its best performance with the help of Adaboost and bootstrapping. However, since Haar wavelets are densely sampled, they require much computation time. For efficiency, we only use global Haar wavelet features with a dimension of 128 [18]. The Haar wavelet classifier is cascaded after the HOG-based classifier for the following reasons: 1) concatenating the 1620-D HOG feature with the 128-D Haar feature to form a new longer feature will make the Haar feature insignificant due to its relatively much lower dimensionality, and 2) the 128-D Haar feature requires double the computation of a 1620-D HOG, so it should come second for speed considerations. For both the HOG and Haar wavelet classifiers, the linear SVMs are implemented with the libLinear toolbox [9].

C. Video Level

For each sliding window that passes the coarse detection at the frame level, its surrounding spatiotemporal volume is analyzed for further refinement and fine localization. First, spatial refinement is performed to merge sliding windows in order to locate pedestrians more accurately. Then, temporal refinement is done with respect to both appearance and motion for greater elimination of false positives.

1) Spatial Refinement and Fine Localization: In this procedure, overlapping positive windows from the frame level are merged according to their overlap ratios, and then, a finely sampled sliding window with the HOG-based linear SVM classifier is used to recompute the pedestrian location. The procedural details are described in the following.

1) *Window merging with overlap ratio matrix*: various NMS methods have been proposed to reduce redundancy in sliding-window-based detection. Our method performs grouping of the overlapping positive windows and then selects a single window from each group by using NMS. The following steps are used.
   a) From n windows [denoted as $W_i (i = 1, 2, \ldots, n)$] classified as positive at the frame level, an overlap ratio matrix $\mathbf{O}_{n\times n}$ is obtained, where $\mathbf{O}(i, j)$ denotes the overlap ratio between the ith and jth windows.
   b) Scan each row of $\mathbf{O}_{n\times n}$ from top to bottom. For the ith row, windows that overlap the ith window with a ratio larger than a threshold $\omega = 0.5$ are grouped together, where the overlap ratio is computed by using the "intersection-over-union" method [6], [7]. Once a window is grouped, its row is ignored in the remainder of the grouping procedure.
2) *Location refinement with NMS*: for a given window group, location refinement is performed in a *refinement region* with a new window size, as described in the following.
   a) *New sliding window size*: the new sliding window size is set to the average window size in this group.
   b) *Refinement region*: the minimum rectangle $\mathbf{R}$ that encompasses all of the windows in the group is determined and is then expanded vertically toward the top and toward the bottom of the frame by 10% of its height.
   c) *Location refinement*: to compute the refined location, a HOG-based linear SVM is applied to sliding windows along the reference line, with shift steps of 1/128 of the new window height. Each sliding window and its decision value are added into the original group. The window with the maximum decision value and minimum area is selected as the representative window of this group.

2) Temporal Refinement: In this procedure, to each window output by the spatial refinement, appearance and motion refinement are performed in the temporal space according to the following two steps.

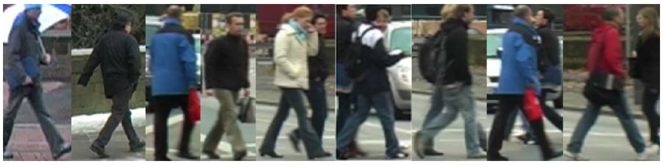


Fig. 4. Various cases of cropped training samples, with different entering angles/directions, with/without occlusions, and with/without crosswalks.

1) *Appearance refinement with HOG-based linear SVM*: for a given window, the previous $p = 3$ frames are cached, and the corresponding windows in these frames are classified by the HOG+LinSVM classifier. A positive decision is made when $(p - \mu)$ of these corresponding windows is classified as positive, with μ set to two in our experiments.
2) *Movement direction refinement with optical flow*: for windows that pass the preceding appearance refinement, the displacements of its pixels are computed in the p preceding frames by optical flow [16]. If the mean X-axis displacement among the p frames is negative[1] and has a magnitude that exceeds a threshold θ_m, then the pedestrian is considered to be entering the road scene, and a positive detection is made.

V. Data Sets

Large data sets of pedestrian images have been collected by many research groups, and several video data sets have recently become available [6], [7], [31]. However, these data sets do not contain many sudden pedestrian crossings and are thus unsuitable for our application. In this section, we describe the process for constructing our image and video data sets and present the video data set statistics. We intend to make our image and video data sets publicly available in the near future.

A. Training Image Data Set

For training the classifiers used at the frame level, we compiled a set of 2500 grayscale side-view pedestrian images, which differs from standard pedestrian image sets that mainly consist of pedestrians at a frontal view. To obtain the training image set, we first tightly crop side-view pedestrian images from other pedestrian training/testing sets [8], movies, and Internet images. As shown in Fig. 4, the training samples consist of people with different entering angles and poses, occlusions, and cases with/without crosswalks. The cropped images are then resized to a height of 128 pixels, and all left-facing pedestrian images are reflected to face rightward. These resized images are turned into final training samples with a uniform size of 128 × 32 as follows: if the width of the resized image exceeds 32 pixels, we cropped the rightmost 128 × 32 portion of the image to form a training sample; otherwise, we resized this image to 128 × 32.

[1] In our implementation, the present frame is treated as the reference frame. Therefore, the previous p frames of the video sequence covering an entering process may have *negative* X-axis displacements.

B. New Testing Video Data Set

The video data set used in testing our method was acquired using a vehicle-mounted high-definition (HD) video camera (SONY HDR-SR5E) of 1440 × 1080 resolution and 25 fps frame rate. Video clips were captured of people who were instructed to walk/run/jump from left to right in front of the moving vehicle. To our knowledge, this is the first HD pedestrian data set captured from a moving vehicle.[2] Unlike 640 × 480 videos in which adults are less than 80 pixels in height at a distance of 30 m from the camera [6], [7], adults in our data set are about 120 pixels tall at the same measured distance. The statistics of this data set are given in the following.

1) Video Set: The data set consists of 55 video clips that are divided into five sets according to vehicle speed V_v[3] and pedestrian overlap: set 1 contains seven videos with no camera motion (**no speed**; $V_v = 0$), while sets 2–4 were captured with a moving camera. Set 2 contains six videos at a **very low** vehicle speed ($V_v \leq 10$ km/h), set 3 contains 23 videos at a **low** vehicle speed (10 km/h $< V_v \leq 20$ km/h), set 4 contains 14 videos at a **medium** vehicle speed (20 km/h $< V_v \leq 35$ km/h), and set 5 contains five videos with two entering and overlapping pedestrians, with a vehicle speed of about 20–30 km/h. These videos were converted to *seq* and *mat* file formats and labeled using the toolbox of [6]. Please note that pedestrians standing still or moving to the left are not labeled; thus, a total of 5250 pedestrian images were collected from these videos.

2) Sudden Crossing Event Statistics: A sudden crossing event is defined to start at an entering ratio of $\alpha_e = 0.25$ and to end at an entering ratio of $\alpha_l = 1.5$. For an entering ratio of less than 25%, it is often difficult even for humans to tell whether it is a pedestrian. To enlarge the test data set, we shift the reference line to 11 sampled locations at intervals of 64 pixels along the X-axis, thus simulating up to 11 sudden crossing events with each crossing pedestrian. Although the pedestrian remains the same with these shifts, the motion and background will, in general, not be identical. The leftmost reference line is set to $X = 64$ in order to compute the ground truth bounding box for analysis purposes. Note that, in a real application of our system, the reference line is set to $X = 0$. With this expansion of the testing data set, there are totally 314 sudden crossing events from 605 video clips.

VI. Evaluation Criteria

In pedestrian detection, various false positive detection measures have been used for evaluation and comparison. For this paper, we examine false positives per image (FPPI), which is generally considered to be more appropriate for sliding-window-based methods than false positives per window [6].

For true positive evaluation, we consider an event as a unit, i.e., we evaluate the detection rate in terms of true positives per event in the experiments. In a sudden pedestrian crossing event, we define a true positive detection as an entering pedestrian that is detected correctly in at least one frame. For evaluation at the frame level, we use the PASCAL measure [6], in which a true positive detection is recorded when at least one detected bounding box BB_{dt} overlaps with the ground truth pedestrian bounding box BB_{gt} by a minimum ratio of $\theta_o = 0.5$. The overlap ratio β is defined as

$$\beta = \frac{area(BB_{dt} \bigcap BB_{gt})}{area(BB_{dt} \bigcup BB_{gt})} \tag{2}$$

where the left edge of bounding box BB_{gt} lies on the reference line when the entering ratio is less than 1 (i.e., when the pedestrian has not yet fully entered the camera view).

Performance may be evaluated in terms of quantity (i.e., $+1$ for each positive detection and -1 for each false alarm) or quality (i.e., positive and negative scores that account for overlap ratio and entering ratio). Here, we reward high overlap ratios and low entering ratios for positive detections and penalize accordingly for negative detections. Based on this idea, different from traditional piecewise/binary scores, we propose to score true/false positives on a continuous scale as follows:

$$S_{tp}(\alpha, \beta) = \frac{\alpha_e}{1-\theta_o} \cdot \frac{\beta - \theta_o}{\min(\alpha, 1)} \quad \text{if } \beta \geq \theta_o,\ \alpha_e \leq \alpha \leq \alpha_l \tag{3}$$

$$S_{fp}(\alpha, \beta) = \begin{cases} \frac{\alpha_e}{\theta_o} \cdot \frac{\beta-\theta_o}{\min(\alpha,1)} & \text{if } \beta < \theta_o, \alpha_e \leq \alpha \leq \alpha_l \\ -1 & \text{if } \beta < \theta_o, (\alpha < \alpha_e \text{ or } \alpha > \alpha_l) \end{cases} \tag{4}$$

$$S(\alpha, \beta) = \begin{cases} S_{tp}(\alpha, \beta) & \text{if it is a true positive detection} \\ S_{fp}(\alpha, \beta) & \text{if it is a false positive detection} \end{cases} \tag{5}$$

where α and β are defined by (1) and (2), $\alpha_e/(1-\theta_o)$ is a coefficient for normalizing S_{tp} to [0, 1], and α_e/θ_o is a normalization coefficient for S_{fp}. We note that the formulation of (3), which is the score of a positive bounding box, generally leads to very small true positive scores since a full score of 1 requires 100% overlap between the detection bounding box and the pedestrian and must occur at the minimum entering ratio of $\alpha_e = 25\%$. After a pedestrian has entered at a 50% ratio, the maximum score of a detection is only 0.5 even if the detection bounding box perfectly overlaps the ground truth. This evaluation criterion is particularly stringent for sliding-window-based approaches that have a fixed aspect ratio for windows. We smooth the score S using a sigmoid function

$$H_g(S) = \frac{1+e^{-b}}{1-e^{-b}} \cdot \frac{1-e^{-bS}}{1+e^{-bS}} \qquad S \in [-1, 1] \tag{6}$$

where b is a positive scalar parameter and $(1+e^{-b})/(1-e^{-b})$ is a scale factor for normalizing H_g to $[-1, 1]$ (i.e., $H_g(-1) = -1$, $H_g(1) = 1$).

Although these criteria are specifically proposed for performance evaluation of partially visible pedestrian detection, it can readily be used in any general pedestrian detection application. For example, existing evaluation methods output $+1$ if the

[2] Other benchmark videos were captured at resolutions of 320 × 240, 640 × 480, or 720 × 576.

[3] For the purpose of this application, the crossing pedestrians are not too far from the vehicle but, at the same time, not too close due to safety reasons. Since the application targets urban environments, the vehicle is not moving too fast.

overlap ratio between the detection bounding box and the ground truth box is greater than a predefined threshold θ_o and -1 if the overlap ratio is less than θ_o. However, the manually labeled ground truth may not be accurate, so we believe that a criterion with a continuous score is more reasonable and accurate. Since entering ratios are not used for fully visible pedestrians, we can simply remove the α's in (3) and (4) to obtain the following definitions for scoring fully visible pedestrian detection results:

$$S = \begin{cases} \frac{\beta-\theta_o}{1-\theta_o} & \text{if } \beta \geq \theta_o \text{ (true positive detection)} \\ \frac{\beta}{\theta_o} - 1 & \text{if } \beta < \theta_o \text{ (false positive detection).} \end{cases} \tag{7}$$

This score may also be postprocessed using (6).

One can see that the linear method [i.e., (5) and (7)] and the sigmoid method output continuous scores with respect to β (i.e., the overlap ratio between the detection bounding box and the annotated ground truth); by contrast, the binary method sharply separates the possible scores even though the manually labeled ground truth may not be completely accurate. Unlike the linear method, the sigmoid method can more clearly separate scores that lie away from the decision boundary. Moreover, when the input $S \in [-1, 1]$, the output H_g approaches the binary evaluation method for large values of b, while it approaches the linear method for small positive values of b. The sigmoid method offers greater generality, with the existing binary evaluation method as a special case.

VII. Experimental Results

Using the acquired data sets and presented evaluation criteria, we report the performance of our method on this problem. The experiments were performed on a 2.67-GHz PC with 1.5-GB DDRII RAM using a single thread, with implementations of HOG classification and optical flow that were revised from OpenCV2.0. It is shown that the three-level framework obtains solid detection performance at a high detection speed. Our experiments examine cases with only pedestrians that enter from the left side. To also handle pedestrians entering from the right, each video frame can be flipped about the Y-axis to double the number of input frames and to reduce the processing speed by half. The processing speeds reported in the Abstract and in Section I reflect this two-sided configuration, while the speeds given in this section assume left-entering pedestrians only.

A. Parameters

A total of nine parameters, listed in Table I, are used in our algorithm. Four of them are related to event definition and evaluation and do not affect pedestrian detection performance. Four of the remaining five parameters have fixed values throughout all experimentation, while one parameter requires tuning.

The following parameters affect system performance.

1) The threshold ω is used in NMS for merging overlapped windows with positive detections. Since it is employed with the commonly used intersection-over-union evaluation metric, we set it to the typical value of 0.5.

TABLE I
PARAMETER SETTINGS

Category		Notation	Value/Range
Performance Independent	Event Definition	α_e	0.25
		α_l	{0.75,1.5}
	Evaluation Metric	θ_o	{0.25,0.5}
		b	5
Performance Related	Fixed value	ω	0.5
		p	3
		μ	2
	To be tuned	f	[0,1]
		θ_f	6

TABLE II
PERFORMANCE OF EACH LEVEL

Level #	AIW#	AOW#	TPPE	FFPI	Time/frame
1. Local	184	8.056	1.000	7.716	5.797ms
2. Frame	8.056	0.137	0.971	0.077	2.280ms
3. Video	0.137	0.061	0.911	0.035	0.980ms

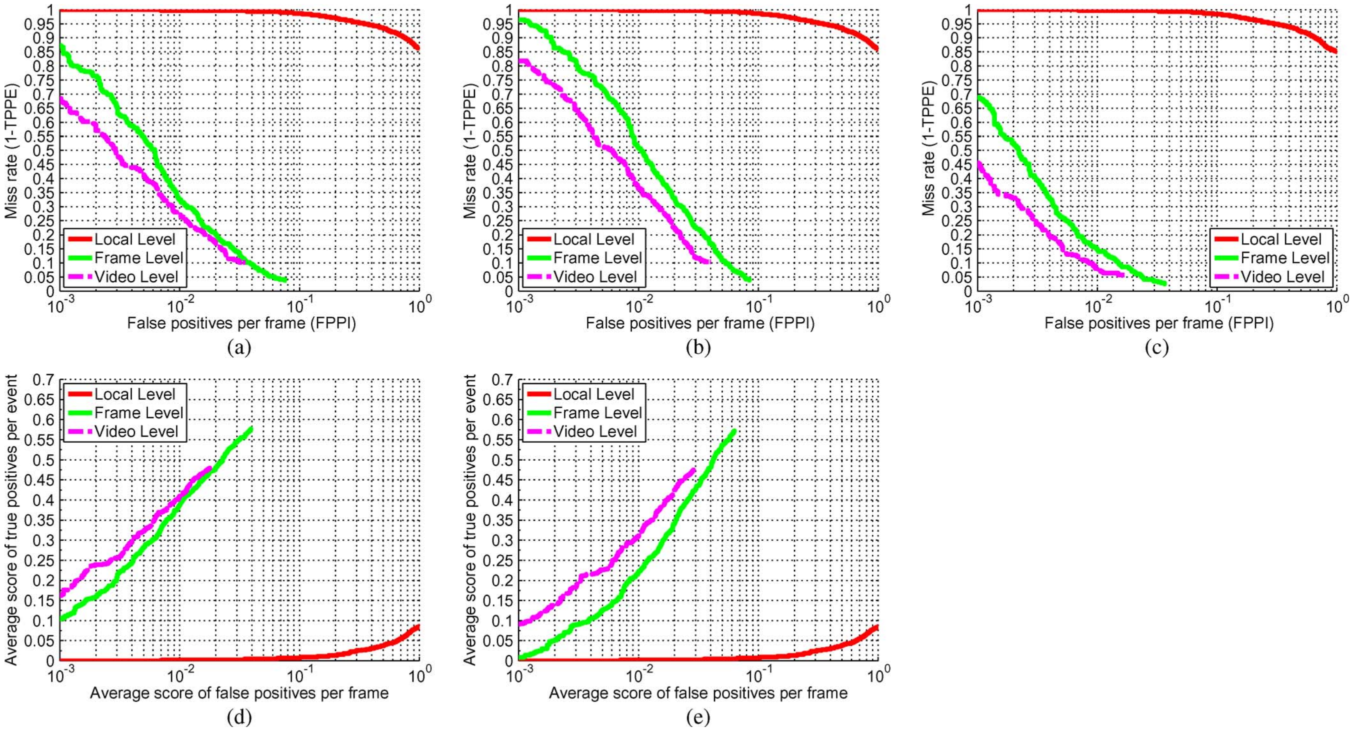
2) The p and μ in the temporal refinement step are parameters used in examining the support of a positive detection among previous frames. Here, we set the parameters to require at least one of the preceding three frames to also have a positive response.

3) In the LBP difference-based local level detection, f and θ_f need to be set to balance detection accuracy and speed. We empirically fix θ_f to 6 and tune f according to vehicle speed. In our experiments, f can take one of two values, depending on whether the camera is fixed or moving. In this way, only one parameter needs to be tuned in this paper.

B. Performance of Each Level

In the coarse-to-fine framework, each level contributes to detection performance and accelerating the detection process. The effectiveness of each level in terms of candidate window reduction, false positive reduction, and execution time is shown in Table II for the event definition parameters $\alpha_e = 0.25$ and $\alpha_l = 1.5$ and true positive evaluation threshold $\theta_o = 0.5$. In the table, AIW# denotes the average number of input windows to that level, and AOW# refers to the output windows. The corresponding ROC curves in Fig. 5(a) and scored curves in Fig. 5(d) show that each level improves detection accuracy. In Table II, the benefit of the sparse sliding window strategy is evident, with only 184 windows in each frame that need to be processed. Moreover, the LBP difference-based motion filter rejects most of the negative windows while retaining 100% of the true positives for later processing. This reduces the 184 windows down to 8 on average, which allows for more complex processing at later levels with fast computation times. At the frame level, the coarse detection with the cascaded SVM classifiers removes some false positives, and then, the spatiotemporal refinement and fine localization reduce the false positive rate further.

For a more challenging case where detection is constrained only to partially visible pedestrians, for example, when $\alpha_e =$



(a)

(b)

(c)

(d)

(e)

Fig. 5. (a) ROC curves and (d) scored ROC curves of different levels with parameters $\alpha_l = 1.5$ and $\theta_0 = 0.5$. (b) ROC curves and (e) scored ROC curves of different levels with parameters $\alpha_l = 0.75$ and $\theta_0 = 0.5$. (c) ROC curves of different levels with parameters $\alpha_l = 1.5$ and $\theta_0 = 0.25$.

0.25 and $\alpha_l = 0.75$, we obtain the ROC curves and scored curves shown in Fig. 5(b) and (e). While the trend from level to level remains the same with these settings, the effect of the spatiotemporal refinement is seen to be more pronounced, with a detection rate improvement from 49% to 64% at 0.01 FPPI, equivalent to a 30.6% relative improvement. By contrast, when $\alpha_l = 1.5$, the improvement is from 67% to 73%, equivalent to a 9.0% relative improvement. This comparison illustrates the particular importance of video-level refinement in early detection of sudden pedestrian crossings.

The reported results for true positives follow that of PASCAL with $\theta_o = 0.5$ [6]. If this threshold is relaxed to 0.25 [7], the performance increases appreciably. As shown in Fig. 5(c), for $\alpha_e = 0.25$ and $\alpha_l = 1.5$, our method achieves a detection rate of 92% at 0.01 FPPI and 54% at 0.001 FPPI, which is equivalent to 54% true positive detection and 1.5 false positives per minute. This performance is promising for practical applications, especially when considering the real-time processing on simple hardware.

C. Performance With Different Vehicle Speeds and Pedestrian Occlusions

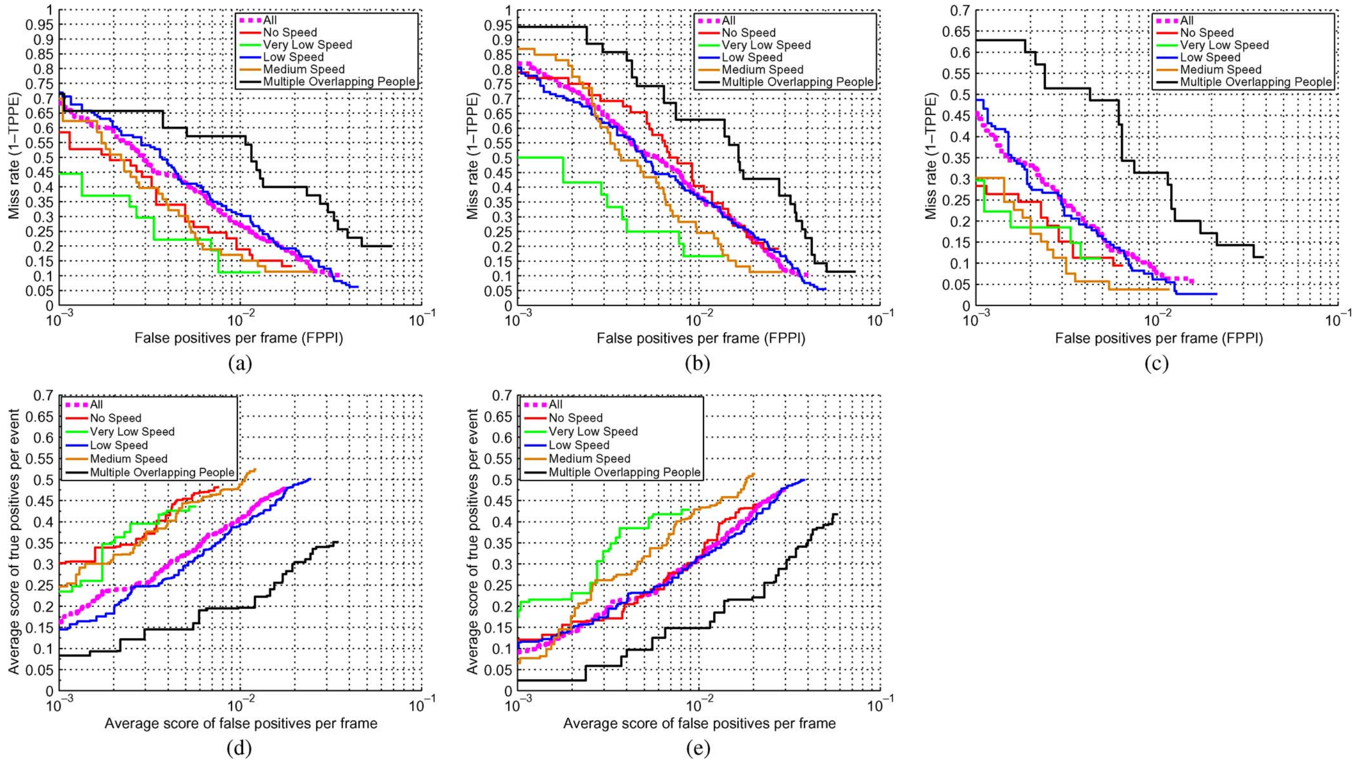
In this section, we investigate the performance of the proposed approach under different vehicle speeds and also different occlusions due to pedestrian overlap. We asked the pedestrians to enter with different movement styles and speeds. Vehicle speed was purposely kept at one of four levels. We did not control the entering of other people, vehicles, and bicycles in the scene.

From the ROC curves shown in Fig. 6, one can make a conclusion that the proposed system exhibits reduced performance in the case of multiple overlapping pedestrians in comparison to single individuals. This is as expected since the multiple overlapping pedestrian case is particularly difficult. The occluded person usually cannot be correctly detected since only a very small part is visible and may be included in the bounding box of another visible person. We note, however, that acceptable results can be obtained in practice if the person in front (the occluder) is correctly detected.

Except for this difficult case, we can observe that the system performance is no worse than the reported average performance; however, the effect of vehicle speed remains unclear. In Fig. 6(a) and (b), the performance is comparable for different vehicle speeds, except for the very low speed case. With our proposed scored criteria, in Fig. 6(d), the performance for different vehicle speeds is comparable, except for the low speed case; in Fig. 6(e), only the **very low speed** case has higher performance, especially at low false positive rates (e.g., 0.001FPPI). In Fig. 6(c), when the threshold θ_o is relaxed to 0.25, the performance is almost the same at a low false positive rate of 0.001FPPI with different vehicle speeds, except for the very low speed case. We believe that the slight differences in performance may be the result of different background and pedestrian appearances among the videos at different speeds.

D. Performance Comparison to Dense Sliding Windows

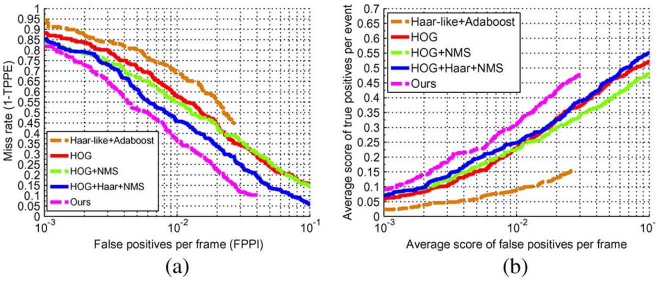
Our method, which utilizes sparse sampling of sliding windows, is compared here to methods that employ dense sliding window sampling. In the dense sampling, the scaling factor is set to 1.05, and the shift step is 1/32 of the window height, only along the Y-axis as in our method, to give 2956 sliding windows in total. For a fair comparison, all of the classifiers



(a) (b) (c)

(d) (e)

Fig. 6. (a) ROC curves and (d) scored ROC curves of different vehicle speeds with parameters $\alpha_l = 1.5$ and $\theta_0 = 0.5$. (b) ROC curves and (e) scored ROC curves of different vehicle speeds with parameters $\alpha_l = 0.75$ and $\theta_0 = 0.5$. (c) ROC curves for different vehicle speeds with parameters $\alpha_l = 1.5$ and $\theta_0 = 0.25$.



(a) (b)

Fig. 7. (a) ROC curves and (b) scored ROC curves of dense sliding-window-based methods and our method ($\alpha_l = 0.75$ and $\theta_0 = 0.5$).

are trained using the same data set and the same implementation of HOG, Haar, and NMS. We also compare these HOG+LinSVM-based methods with the well-known Haar-like+Adaboost method. The OpenCV toolbox [39] is adopted to train the classifier by using the default parameter setting with the same number of positive samples and 15 000 negative samples, i.e., much more negative samples are used. Finally, 515 out of 2 938 688 features are selected for the Adaboost classifier with 15 layers.

From the ROC curves shown in Fig. 7(a), one can draw the following conclusion: 1) the Haar-like+Adaboost method has the lowest performance; this observation is consistent with previous studies [6], [7], which indicates that the HOG feature has more discriminative power for larger sized pedestrian detection; 2) NMS increases the accuracy of HOG classification, which is consistent with existing work [31]; 3) a cascaded HOG+Haar classifier further increases accuracy because the global texture feature can overcome the shortcoming of the edge sensitive HOG feature; and 4) the spatiotemporal-based coarse-to-fine framework has the highest detection rate, and the scored ROC curves shown in Fig. 7(b) indicate that the three-level framework detects crossing pedestrians earlier and/or with a better bounding box fit. In addition, as shown in Fig. 5(a), when $\alpha_l = 0.75$, 1.0, and 1.5, our method, respectively, gives a 9%, 10%, and 9% absolute improvement in detection rate using the binary evaluation method at 0.01 FPPI over the dense HOG+Haar+NMS method.

E. Detection Speed

In the proposed framework, some acceleration measures are used in order to perform real-time detection. As shown in Table II, the average time cost of our method is 9.06 ms per frame excluding I/O costs, and there is no significant difference between videos captured at different vehicle speeds. Compared with the traditional dense scanning strategy with only HOG features and NMS, the processing speed increases by about two orders of magnitude from 765.60 to 9.06 ms per frame. Moreover, as shown in Fig. 7(a), for entering ratios of 25%–75%, the detection rate increases significantly from 45% to 63% at 0.01 FPPI, equivalent to about a 40.0% relative improvement.

F. Evaluation Criteria

Comparing Fig. 5(d) with Fig. 5(a) and Fig. 5(e) with Fig. 5(b), the proposed evaluation criteria defined in (5) are seen to be in accordance with traditional quantitative measures. However, as shown in Fig. 7(a) and (b), the proposed criteria additionally account for the quality of positive detections. For example, the performance differences between

Haar-like+Adaboost and HOG-based methods are more distinct by evaluating with the proposed criteria [Fig. 7(b)]. Similarly, in Fig. 7(a), the HOG curve and HOG+NMS curve overlap in the region of [0.01, 0.1] FPPI, while the scored curves in Fig. 7(b) indicate the higher accuracy of the HOG without NMS method within this interval. This can be explained by the NMS reducing false positives by merging overlapped boxes and removing boxes with smaller decision values; however, it also removes some correct positive detections, and this will lead to a lower positive score in the region of comparatively high false positive scores. In regions where the nonscored ROC curves are indistinct, the scored curves can provide a more detailed and physically meaningful comparison.

VIII. Conclusion

In this paper, the problem of detecting sudden pedestrian crossings has been defined and studied as an application for driving assistance systems. A three-level framework has been proposed to locate crossing pedestrians as early as possible (i.e., before fully entering the camera view) with a low false alarm rate as needed in practical systems. With a newly collected data set and the proposed evaluation criteria, the effectiveness of this approach has been demonstrated.

In future work, we plan to elevate the performance in four ways. 1) We plan to include sensor (camera) characteristics into the detection algorithm and establish the distance relationships between virtual reality and physical reality. 2) We want to improve the performance in the frame-level processing, such as by investigating new static features. 3) We wish to introduce kinematics knowledge into the detection algorithm for better utilization of motion information. 4) We plan to improve classification accuracy by adopting new algorithms based on likelihoods. We also would like to extend this system to handle a broader set of obstacles, such as other vehicles.

References

[1] T. Ahonen, A. Hadid, and M. Pietikainen, “Face description with local binary patterns: Application to face recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006.

[2] I. P. Alonso, D. F. Llorca, M. A. Sotelo, L. M. Bergasa, P. R. de Toro, J. Nuevo, M. Ocana, and M. A. G. Garrido, “Combination of feature extraction methods for SVM pedestrian detection,” *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 2, pp. 292–307, Jun. 2007.

[3] N. Bellotto and H. S. Hu, “Multisensor-based human detection and tracking for mobile service robots,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 1, pp. 167–181, Feb. 2009.

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

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

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

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

[8] A. Ess, B. Leibe, K. Schindler, and L. van Gool, “A mobile vision system for robust multi-person tracking,” in *Proc. IEEE Conf. CVPR*, 2008, pp. 1–8.

[9] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, “LIBLINEAR: A library for large linear classification,” *J. Mach. Learn. Res.*, vol. 9, pp. 1871–1874, Jun. 2008.

[10] D. M. Gavrila, J. Giebel, and S. Munder, “Vision-based pedestrian detection: The protector system,” in *Proc. Intel. Veh. Symp.*, 2004, pp. 13–18.

[11] D. M. Gavrila and S. Munder, “Multi-cue pedestrian detection and tracking from a moving vehicle,” *Int. J. Comput. Vis. (IJCV)*, vol. 73, no. 1, pp. 41–59, Jun. 2007.

[12] K. Huang, D. Tao, Y. Yuan, X. Li, and T. Tan, “Biologically inspired features for scene classification in video surveillance,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 41, no. 1, pp. 307–313, Feb. 2011, DOI: 10.1109/TSMCB.2009.2037923.

[13] K. Huang, D. Tao, Y. Yuan, X. Li, and T. Tan, “View-independent human behavior analysis,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 4, pp. 1028–1035, Aug. 2009.

[14] I. Laptev, “Improving object detection with boosted histograms,” *Image Vis. Comput.*, vol. 27, no. 5, pp. 535–544, Apr. 2009.

[15] X. Li, S. Maybank, Y. Yan, D. Tao, and D. Xu, “Gait components and their application to gender recognition,” *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 38, no. 2, pp. 145–155, Mar. 2008.

[16] B. D. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” in *Proc. Int. Joint Conf. Artif. Intell.*, 1981, pp. 674–679.

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

[18] B. S. Manjunath and W. Y. Ma, “Texture features for browsing and retrieval of image data,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, no. 8, pp. 837–842, Aug. 1996.

[19] Y. Mu, S. Yan, Y. Liu, T. Huang, and B. Zhou, “Discriminative local binary patterns for human detection in personal album,” in *Proc. IEEE Conf. CVPR*, 2008, pp. 1–8.

[20] S. Munder and D. Gavrila, “An experimental study on pedestrian classification,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1863–1868, Nov. 2006.

[21] S. Paisitkriangkrai, C. Shen, and J. Zhang, “An experimental study on pedestrian classification using local features,” in *Proc. Int. Symp. Circuits Syst.*, 2008, pp. 2741–2744.

[22] C. Papageorgiou and T. Poggio, “Trainable pedestrian detection,” in *Proc. Int. Conf. Image Process.*, 1999, pp. 35–39.

[23] R. Poppe, “Vision-based human motion analysis: An overview,” *Comput. Vis. Image Understand.*, vol. 108, no. 1/2, pp. 4–18, Oct. 2007.

[24] Y. Ran, Q. Zheng, R. Chellappa, and T. M. Strat, “Applications of a simple characterization of human gait in surveillance,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 4, pp. 1009–1020, Aug. 2010.

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

[26] S. Turner, L. Sandt, J. Toole, R. Benz, and R. Patten, *Federal Highway Administration University Course on Bicycle and Pedestrian Transportation*, 2006. [Online]. Available: http://www.tfhrc.gov/safety/pedbike/pubs/05085/pdf/combinedlo.pdf

[27] P. Viola, M. Jones, and D. Snow, “Detecting pedestrians using patterns of motion and appearance,” *Int. J. Comput. Vis. (IJCV)*, vol. 63, no. 2, pp. 153–161, 2005.

[28] X. Y. Wang, T. X. Han, and S. Yan, “A HOG-LBP human detector with partial occlusion handling,” in *Proc. IEEE ICCV*, 2009, pp. 32–39.

[29] X. Wang, S. Wang, and D. W. Bi, “Distributed visual-target-surveillance system in wireless sensor networks,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 5, pp. 1134–1146, Oct. 2009.

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

[31] C. Wojek, S. Walk, and B. Schiele, “Multi-cue onboard pedestrian detection,” in *Proc. IEEE Conf. CVPR*, 2009, pp. 794–801.

[32] Y. W. Xu, X. B. Cao, and H. Qiao, “An efficient tree classifier ensemble-based approach for pedestrian detection,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 41, no. 1, pp. 107–117, Feb. 2010.

[33] Y. Yuan, Y. Pang, J. Pan, and X. Li, “Scene segmentation based on IPCA for visual surveillance,” *Neurocomputing*, vol. 72, no. 10–12, pp. 2450–2454, Jun. 2009.

[34] H. Zhou, Y. Yuan, and C. Shi, “Non-rigid object tracking in complex scenes,” *Pattern Recognit. Lett.*, vol. 30, no. 2, pp. 98–102, Jan. 2009.

[35] J. Steffens, E. Elagin, and H. Neven, “Person spotter—Fast and robust system for human detection, tracking and recognition,” in *Proc. 3rd IEEE Int. Conf. Autom. Face Gesture Recog.*, 1998, pp. 516–521.

[36] H. Zhou, A. Wallace, and P. Green, “Efficient tracking and ego-motion recovery using gait analysis,” *Signal Process.*, vol. 89, no. 12, pp. 2367–2384, Dec. 2009.

[37] H. Zhou, Y. Yuan, and C. Shi, “Object tracking using SIFT features and mean shift,” *Comput. Vis. Image Understand.*, vol. 113, no. 3, pp. 345–352, Mar. 2009.

[38] Y. Huang, S. Fu, and C. Thompson, "Stereovision-based object segmentation for automotive applications," *EURASIP J. Appl. Signal Process.*, vol. 2005, pp. 2322–2329, Jan. 2005.

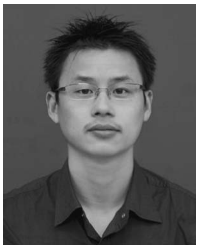
[39] OpenCV. [Online]. Available: http://opencv.willowgarage.com/wiki/

[40] A. Shashua, Y. Gdalyahu, and G. Hayun, "Pedestrian detection for driving assistance systems: Single-frame classification," in *Proc. Intell. Veh. Symp.*, 2004, pp. 1–6.

[41] T. Yang, J. Li, Q. Pan, C. Zhao, and Y. Zhu, "Active learning based pedestrian detection in real scenes," in *Proc. ICPR*, 2006, pp. 904–907.

[42] M. Stikic and B. Schiele, "Activity recognition from sparsely labeled data using multi-instance learning," in *Proc. o4th Int. Symp. LoCA*, 2009, pp. 156–173.

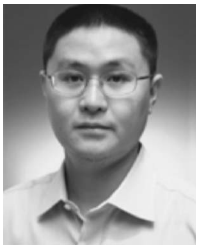
[43] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel features," in *Proc. BMVC*, 2009, pp. 1–11.



Yanwu Xu (M'08) received the B.Eng. degree in computer science and the Ph.D. degree in applied computer science from the University of Science and Technology of China, Hefei, China, in 2004 and 2009, respectively.

He is currently a Postdoctoral Research Fellow at Nanyang Technological University, Singapore. His current research interests include onboard pedestrian detection system, intelligent transportation systems, video-based event analysis, machine learning, and visual computing.

Dr. Xu received the President Award of the Chinese Academy of Sciences in 2009.



Dong Xu (M'07) received the B.E. and Ph.D. degrees from the University of Science and Technology of China, Hefei, China, in 2001 and 2005, respectively.

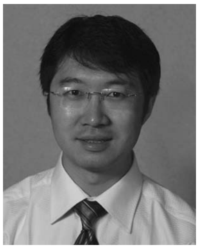
While working toward his Ph.D. degree, he was with Microsoft Research Asia, Beijing, China, and the Chinese University of Hong Kong, Shatin, Hong Kong, for more than two years. He was a Postdoctoral Research Scientist with Columbia University, New York, NY, for one year. He is currently an Assistant Professor with Nanyang Technological University, Singapore. His current research interests include computer vision, statistical learning, and multimedia content analysis.

Dr. Xu was the coauthor of a paper that won the Best Student Paper Award in the prestigious IEEE International Conference on Computer Vision and Pattern Recognition in 2010.

Stephen Lin (M'01) received the B.S.E. degree from Princeton University, Princeton, NJ, and the Ph.D. degree from the University of Michigan, Ann Arbor.

He is currently a Senior Researcher with the Internet Graphics Group, Microsoft Research Asia, Beijing, China. His research interests include computer vision and computer graphics.

Dr. Lin has served as a Program Cochair for the IEEE International Conference on Computer Vision 2011 and the Pacific Rim Symposium on Image and Video Technology 2009, as a General Chair for the IEEE Workshop on Color and Photometric Methods in Computer Vision 2003, and as an Area Chair for the IEEE International Conference on Computer Vision 2007 and 2009.



Tony X. Han (M'01) received the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign, Urbana, in 2007.

He is currently an Assistant Professor of electrical and computer engineering with the University of Missouri, Columbia. His specialties lie in computer vision and machine learning, with emphasis on human/object detection, large-scale image retrieval, object tracking, action recognition, video analysis, and biometrics.

Dr. Han was a recipient of a CSE fellowship. His research team was a joint winner of the action recognition task in the worldwide grand challenge PASCAL 2010. The human detector developed by his group was ranked second in the worldwide grand challenge PASCAL 2009. His research team together with UIUC joint team also won the first place in Facial Expression Recognition and Analysis Challenge (FERA) in 2011.

Xianbin Cao (M'08–SM'10) received the B.S. degree in computer science and the M.S. degree in information and system from Anhui University, Hefei, China, in 1990 and 1993, respectively, and the Ph.D. degree in intelligent information processing from the University of Science and Technology of China (USTC), Hefei, in 1996.

He joined the Department of Computer Science and Technology, USTC, in 1996 and became an Associate Professor and a Professor in 1999 and 2005, respectively. He was the Vice Director of the department and the Artificial Intelligence Research Center from 2006 to 2009. Since 2005, he has been the Administrative Director of the Anhui Province Key Laboratory in Computing and Communication. He is currently a Professor with the School of Electronic and Information Engineering, Beihang University, Beijing, China. He has published more than 100 books, book chapters, and papers in these areas since 1993. His current research interests include natural computation, intelligent transportation systems, and information security.

Xuelong Li (M'02–SM'07) is a Researcher (Full Professor) with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, China.