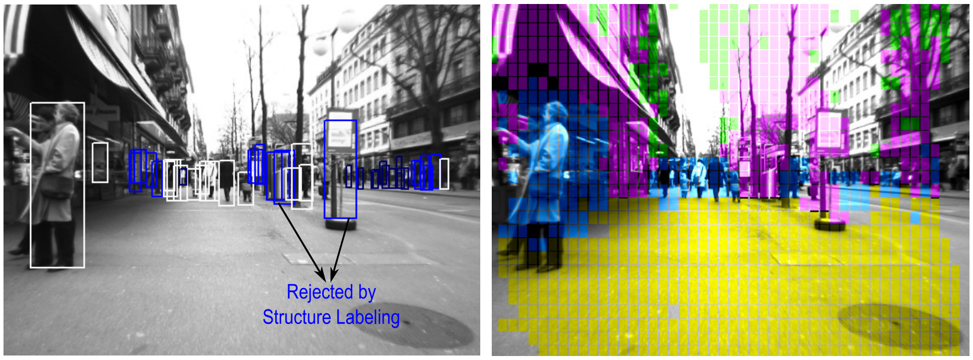
Pedestrian Detection with Depth-guided Structure Labeling

Mayank Bansal, Bogdan Matei, Harpreet Sawhney, Sang-Hack Jung, Jayan Eledath
Sarnoff Corporation
Princeton, NJ, USA*
{mbansal,bmatei,hsawhney,sjung,jeledath}@sarnoff.com



Abstract

We propose a principled statistical approach for using 3D information and scene context to reduce the number of false positives in stereo based pedestrian detection. Current pedestrian detection algorithms have focused on improving the discriminability of 2D features that capture the pedestrian appearance, and on using various classifier architectures. However, there has been less focus on exploiting the geometry and spatial context in the scene to improve pedestrian detection performance.

We make several contributions: (i) we define a new 3D feature, called a Vertical Support Histogram*, from dense stereo range maps to locally characterize 3D structure; (ii) we estimate the likelihoods of these 3D features using kernel density estimation, and use them within a Markov Random Field (MRF) to enforce spatial constraints between the features, and to obtain the Maximum A-Posteriori (MAP) scene labeling; (iii) we employ the MAP scene labelings to reduce the number of candidate windows that are tested by a standard, state-of-the-art pedestrian appearance classifier. We evaluate our algorithm on a very challenging, publicly available stereo dataset and compare the performance with state-of-the-art methods.*

1. Introduction

Detecting pedestrians from images is a fundamental problem in numerous applications ranging from mobile robotics to autonomous driving and intelligent vehicles. Large pose variations of the human body, appearance changes due to clothing, background and illumination variations, and occlusions in complex urban environments, all contribute to making pedestrian detection a very challenging computer vision task.

During the last few years, significant progress on pedestrian detection has been reported in the literature using both stereo [8, 10] and monocular cameras [16, 4, 19, 18, 17, 6]. However, substantial improvements are still required to reduce the number of false positives per frame to an acceptable level for practical applications, while maintaining high detection rates. For example, Dollar et al.[5] recently reported benchmark performance of current state of the art approaches on large dataset, where the best performance reaches on average 60% detection rate with one false positive per frame. This paper proposes an effective scheme of reducing false detections by adopting structural scene classification based on stereo depth map. Although stereo-based depth map is often noisy and computationally expensive to compute, the proposed approach exploits 3D scene geometry and discovers scene structure from noisy depth representation. This approach can in fact achieve overall a computationally more efficient solution and produce improved classification performance.

The pedestrian detection system can be largely divided into monocular camera and stereo-based systems. Monocular pedestrian detectors typically employ cascaded classifiers trained using features that capture the 2D appearance of a person. Examples include the HoG (Histogram of Gradients) descriptor of Dalal and Triggs [4], or the covariance descriptor of Tuzel et al. [18]. Since no scale information for potential targets is directly available in monocular images, classifiers are typically run at multiple scales, evaluating a large number of hypotheses per image frame. This tends to incur more false detections some of which may not lead to any valid interpretation in terms of scene structure.

*This work was supported in part by the Federal Highway Administration under an EARP contract DTFH61-07-H-00039.

2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops
978-1-4244-4441-0/09/$25.00 ©2009 IEEE

Recently, more discriminative features and new classification methods have been proposed with improved results on public dataset, however relatively little focus has been put on exploiting scene geometry and spatial context that can potentially allow efficient target object detection.

Stereo vision has a significant advantage over monocular vision, because it produces 3D depth maps that can be used to extract scene geometry and classify structures such as ground plane, buildings, and vegetation. The structural cues in the scene can be used to constrain further computation such as appearance-based classifiers, to focus on relevant scene parts. In addition, stereo eliminates the need to search over a large number of scales. Recent research underscores the significant reduction in false positives and increased detection rates achieved when stereopsis is used [10, 14, 8]. Despite these advantages, existing stereo based pedestrian detectors make limited use of 3D structure for finding candidate regions on which a pedestrian classifier can be applied. For example in [10] a sparse stereo for the initial pedestrian detection is used, resulting in poor foreground/background separation [10]. Similarly, structure from motion (SfM) has been used to constrain 2D image based pedestrian detection [8].

We present a novel dense stereo based pedestrian detection algorithm that uses depth-guided structure labeling to substantially reduce the number of false positives while maintaining high computational efficiency and detection rates. We represent and exploit 3D scene geometry and context within a principled statistical framework to reduce the number of pedestrian candidate windows that are subsequently tested by a state-of-the-art appearance based classifier. We make several contributions in this paper: (i) we define a new 3D feature, called a *Vertical Support Histogram*, from dense stereo range maps to locally characterize 3D structure; (ii) we learn the likelihoods of these 3D features using kernel density estimation, and use them within a Markov Random Field (MRF) to enforce spatial relationships between the features, and to obtain the MAP scene labelings. The MRF can be extended to use more discriminative features such as 3D shape context or spin images obtained from more accurate, global stereo algorithms such as [22]; (iii) we employ the MAP scene labelings returned by the MRF to prune candidate windows returned by a 3D template matching algorithm such as [3], and classify the remaining windows using a standard state-of-the-art classifier cascade trained using appearance features. We evaluate our algorithm on a very challenging stereo based dataset, publicly available from [8] obtaining promising results, and also compare the performance with state-of-the-art methods.

The paper is organized as follows. Section 2 presents related work beyond what is discussed above. Starting with an overview in Section 3, the depth-guided structure classification approach is described in Section 4. In Section 5 we briefly discuss the data and evaluation methodology followed by results and comparison with other approaches.

2. Related Work

Ess et al. [8] describe a stereo-based system for 3D dynamic scene analysis from a moving platform, which integrates sparse 3D structure estimation with multi-cue image based descriptors (*shape context* computed at Harris-Laplace and DoG features [15]) to detect pedestrians. The authors show that the use of sparse 3D structure significantly improves the performance of pedestrian detection. Still, the best performance cited is 40% probability of detection at 1.65 false positives per frame. While the structure estimation is done in real time, the pedestrian detection is significantly slower.

Gavrila and Munder [10] propose PROTECTOR, a real-time stereo system for pedestrian detection and tracking. PROTECTOR employs sparse stereo and temporal consistency to increase the reliability and to mitigate misses. The authors report 71% pedestrian detection performance at 0.1 false alarms per frame without using a temporal contraint with pedestrians located less than 25 meters from the cameras. However, the datasets used were from relatively sparse, uncluttered environments. Recently, Dollar et al. [5] introduced a new pedestrian dataset and benchmarked a number of existing approaches.

Hoeim et al. [12] present a method for learning 3D context from a single image, using appearance cues to infer simple geometric labelings. Hoiem et al. [13] present a probabilistic detection framework which exploits the overall 3D context extracted using [12]. The authors argue that object recognition cannot be solved locally, but requires statistical reasoning over the whole image [13].

Wojek & Schiele [21] propose a probabilistically sound combination of scene labeling and object detection using a Conditional Random Field but their method relies on appearance rather than 3D. Brostow et al. [2] investigate the use of 3D features from structure-from-motion to classify patches in the scene.

3. Overview

The proposed approach actively utilizes depth information obtained from stereo computation. Given a depth map of a scene, first a 3D template-based object detector is applied to find candidate target object hypotheses. Simultaneously, the depth map is processed with generic scene descriptor to identify image regions that match predefined structure classes. The scene labeling from these image regions is then combined with object detector hypothesis to produce a final set of object candidates. The resulting hypotheses are passed to appearance-based pedestrian classifiers for further processing. A brief description of template based object detector and pedestrian classifier is given below.

Stereo based pedestrian detector We employ a stereo based generic object detection algorithm similar to [3] to generate initial pedestrian candidate windows exclusively from range maps. The algorithm from [3] used template matching (through correlation) of pre-rendered 3D templates of objects (e.g., pedestrians and vehicles) with the depth map to detect objects. The 3D template matching was conducted in a coarse to fine manner over a 2D grid overlayed onto the local XZ horizontal plane; at each grid location, a 3D template was matched to the range image data by searching around the X and Z directions, and around the Y (vertical) direction according to the local pitch uncertainty due to calibrations and bumps in the road surface. Locations on the horizontal grid corresponding to local maximal correlation were returned as candidate object locations.

Appearance based pedestrian classifier For the final pedestrian detection we have chosen the HoG-SVM method of Dalal and Triggs [4]. Briefly, the HoG-SVM pedestrian detection method uses a local distribution of intensity gradients to capture the pedestrian appearance. A weak geometric context between the gradients is enforced by computing the histograms within local small regions (cells). The combination of the histogram entries represents the HoG descriptor. After appropriate normalization, the HoG features are used to train a support vector machine (SVM) classifier. The SVM classifier is employed for making the final detection of a candidate window into a pedestrian or non-pedestrian.

4. Structure Classification

A key step in our method for pedestrian detection is depth-based classification of the scene into a few major structural components. Given an image and a sparse and noisy range map, the goal is to probabilistically label each pixel as belonging to one of the following scene classes (see Table 1 for a legend): ground, tall vertical structure, overhang and (pedestrian) object candidates. The intuition behind our formulation is as follows: An occupied cell in the range map of a scene provides evidence for the presence of one or more of the structure classes. The structure classes outlined above typically span multiple adjacent cells in a scene with discontinuities at the boundaries of the classes. Therefore, local evidence for the presence / absence of a class can be combined with neighborhood constraints to probabilistically estimate the class labels.

The range map from stereo does not provide enough resolution to differentiate between a group of people and a vehicle and hence we label all vehicle-like objects as object candidates and let the appearance-based classifier resolve any detections in these regions. Note also that these classes have been chosen to competitively label pixels amongst a few commonly occurring structures as a precursor to pedestrian versus non-pedestrian classification. This is in contrast with traditional detectors that directly apply pedestrian / non-pedestrian classification in which the negative examples themselves form a large set of structured classes. We separate the structured classes further into ones that are distinct from the pedestrian class. In our method, if large numbers of pixels can be rejected as being part of generic structural classes, we can substantially reduce the number of false hypotheses that are presented to a pedestrian / non-pedestrian classifier - gaining us both in performance (FP rate) and computation.

Table 1. Structure classes used for scene labeling

V	Tall vertical structure (magenta)
O	Overhanging structure (green)
G	Ground (yellow)
C	Candidate objects (blue)

We perform structure classification using depth maps. An example depth map is shown in Fig. 1. The map is pseudo-colored with red denoting close-range objects, cyan denoting far-off objects and black denoting missing depth. The depth-map illustrates a number of issues: (i) objects appear bloated in the range-map due to the stereo integration window, (ii) the characteristic noise in the range values is observable as scattered fragments, and (iii) the occlusion boundaries between objects are very noisy.

To robustly handle depth-map errors, first, we define a structure called the Vertical Support Histogram to accumulate 3D information over voxels in the vertical direction. In a given frame, we will compute a feature vector using this structure and subsequently use the feature vector to learn the likelihood of each pixel belonging to a given structural class. Next, we make use of the scene-context constraints arising from the camera viewpoint by formulating the labeling problem as an MRF where the smoothness constraints allow us to reason about the relative positioning of the 3D structure labels in the image. This reduces error in labeling due to depth inaccuracies and gives us an overall smooth labeling of the scene.

4.1. Bayesian Labeling

The problem of Bayesian Labeling is concerned with deriving a labeling $L = \ell$ of image patches Π using a set of image observables $\boldsymbol{r}$. Suppose that we know both the a priori probabilities $P(\ell)$ of labelings ℓ and the likelihood densities $p(\boldsymbol{r}|\ell)$ of the observation $\boldsymbol{r}$. The best estimate one can get from these is one that maximizes the a posteriori probability (MAP) which can be computed using the Bayesian rule,

$$P(\ell|\boldsymbol{r}) = p(\boldsymbol{r}|\ell)\, P(\ell)/p(\boldsymbol{r}) \tag{1}$$

where $p(\boldsymbol{r})$, the density function of $\boldsymbol{r}$, does not affect the MAP solution. In the following, we describe our approach

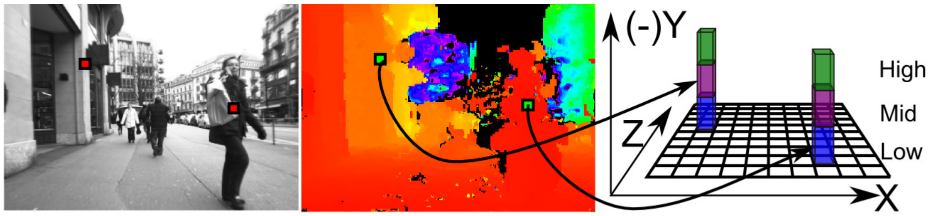


Figure 1. Vertical Support Histogram. Points from the range map are projected to the bins of a 2D histogram in the ground-plane coordinate system. Each histogram bin captures a different height band. The diagram illustrates a 3-bin histogram.

to estimate the likelihood densities $p(\boldsymbol{r}|\ell)$ and the prior probabilities $P(\ell)$ for this labeling problem.

4.2. Likelihood Densities of Structural Labels

4.2.1 Vertical Support Histogram

We first represent the 3D scene as distributions of reconstructed 3D points with respect to a ground plane coordinate system.[1] The ground plane (XZ in our convention) is divided into a regular grid at a resolution of $X_{res} \times Z_{res}$. At each grid cell, we create a histogram of distribution of 3D points according to their heights. All the image pixels that map into a given X, Z coordinate participate in that cell's histogram. The heights, Y coordinate, of all the points in a cell are mapped into a k-bin histogram where each bin represents a vertical height range. We call this structure by the name *Vertical Support Histogram (VSH)* and denote it by V. At any given grid cell, $V[g(X,Z)] = [\boldsymbol{s}_1^g, \boldsymbol{s}_2^g, \cdots, \boldsymbol{s}_k^g]^\top$, where the entry $\boldsymbol{s}_i^g$ measures the support for the i^{th}-bin of the histogram. Fig. 1 shows how image points and the corresponding depth estimates are mapped to 3D distributions for an example histogram with $k = 3$ bins.

In order to compute the supports, $\boldsymbol{s}^g$, robustly from noisy range estimates at each pixel, we use a mean-around-the-median robust estimate of range. We define a $w \times h$ patch at each pixel (x, y). A robust range estimate is computed for each patch (in the following, we use pixel and patch interchangeably, with the idea that the context makes the sense clear). Image points, (x, y), with the robust range estimate Z, are mapped to the corresponding (X, Z) grid cell with height estimate Y. This value Y is used to increment the appropriate bin of the VSH at (X, Z).

Each cell of the histogram is normalized by dividing with the maximum number of pixels that can project to this cell. For a cell at a distance Z from the camera (with horizontal and vertical focal-lengths f_x and f_y respectively), the maximum number of pixels in each image row is,

$$N_{row}^{max}(Z) = X_{res} \cdot f_x / Z$$

and the maximum number of image rows in the height-band $[H_{min}, H_{max}]$ is,

$$N_{col}^{max}(Z) = (H_{max} - H_{min}) \cdot f_y / Z$$

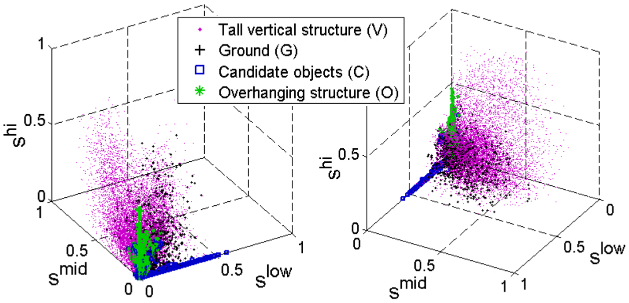


Figure 2. Two views of the feature space showing the distribution of vectors from which the class conditional likelihoods are estimated (note that the fourth component, H_p is not shown).

where H_{max} is determined taking into account the maximum height that is visible in the image at the distance Z. This gives the normalizing factor for the cell to be,

$$N(Z) = N_{row}^{max}(Z) \cdot N_{col}^{max}(Z)$$

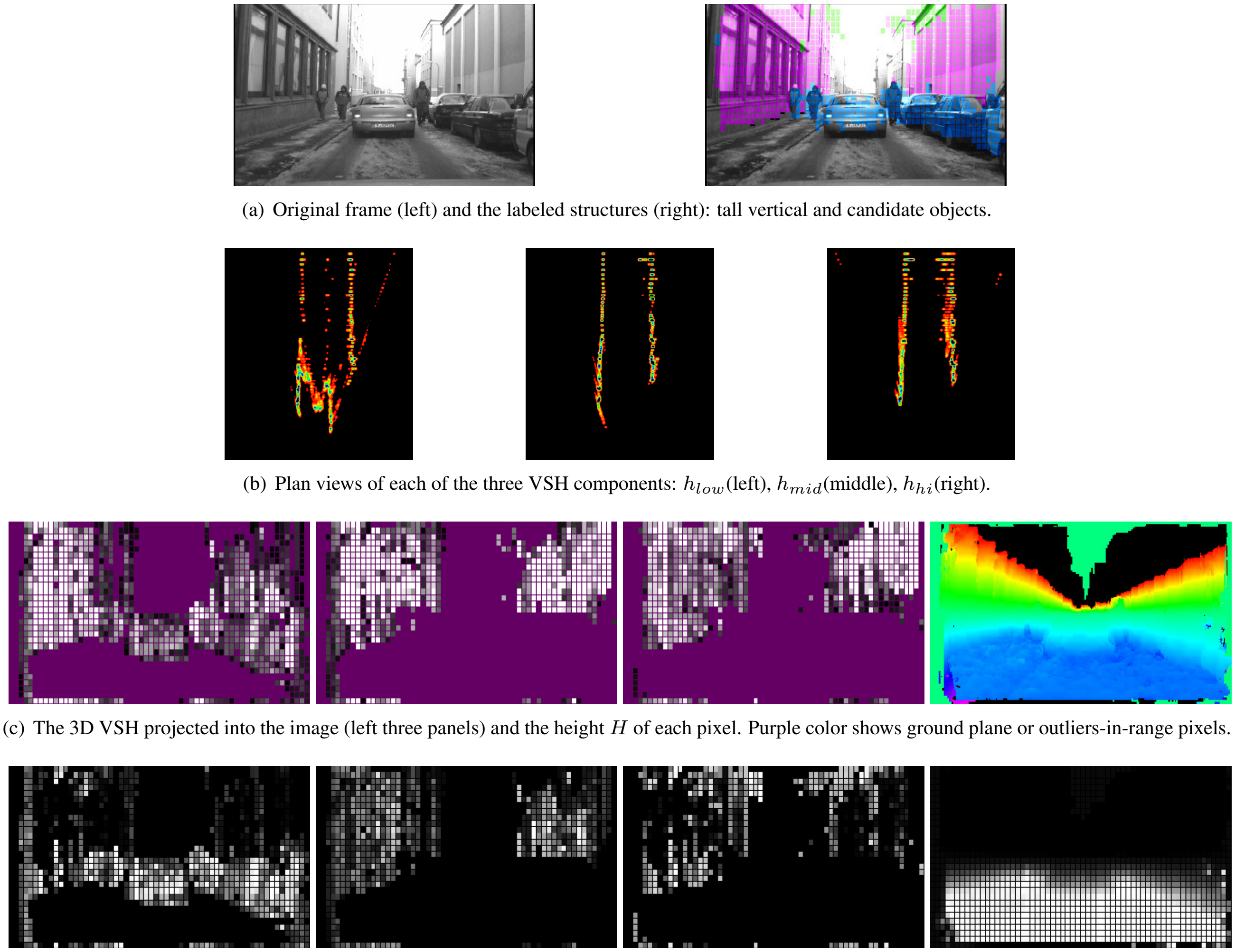
$V(X, Z)$ is defined in 3D space. We transfer this 3D representation to the 2D image and augment it with the $3D$ height. At a given image patch p, we use the robust range estimate Z to project this patch to a footprint (collection of cells) in the XZ-grid coordinate system. An aggregate of the VSH values for the cells within this footprint serves as the total support of p. We define H^p as the average height estimate of the image pixels within the patch. Subsequently, each such image patch, p, is associated with a $(k+1)$-D feature vector: $\boldsymbol{r}_p = [V(X,Z)^\top, H^p]^\top = [\boldsymbol{s}_1^p, \boldsymbol{s}_2^p, \cdots, \boldsymbol{s}_k^p, H^p]^\top$.

4.2.2 Learning the Likelihood Densities

The vertical support histogram captures the distribution of 3D points in any given scene in terms of quantized height bins. $V(X, Z)$ is a representation of the scene in front of a camera. In order to associate each image patch with structural labels, we compute likelihoods for the augmented feature vector, $\boldsymbol{r}_p$, conditioned on the specific structural labels defined earlier.

We randomly sample a small number (≈ 100) of frames from sequences in typical urban driving scenarios. In each frame, structures are coarsely hand-labeled as tall vertical structures (buildings), candidate objects (pedestrians, vehicles), ground and overhanging structures. We experimented

[1] The ground plane can be estimated using any of a number of well-known techniques applied to the reconstructed stereo points, see e.g. [14].



(a) Original frame (left) and the labeled structures (right): tall vertical and candidate objects.

(b) Plan views of each of the three VSH components: h_{low}(left), h_{mid}(middle), h_{hi}(right).

(c) The 3D VSH projected into the image (left three panels) and the height H of each pixel. Purple color shows ground plane or outliers-in-range pixels.

(d) Image likelihoods conditioned on the four labels, respectively, candidate objects, vertical structures, overhangs and the ground plane.

Figure 3. Likelihood density estimation for structure classification. In (b), tall vertical objects (like buildings) span all three histogram bins while objects with low profile (like vehicles and people) span just the first bin. In (c), the bins are projected back to image space and the structures that projected to the histogram bins can be identified. The bin values provide a feature representation at each pixel from which the label likelihoods in (d) are estimated. These likelihoods can be seen to map well to the actual structure in the scene.

with the number of histogram bins k and the placement of the bin boundaries and empirically derived the 3 most discriminative feature components (bins in this case). Feature vectors along these three most discriminative components for all the labeled patches $\{\boldsymbol{r}_p\}$ are shown in Fig. 2, with different colors denoting different ground truth labels. The bin boundary values for these bins are given in Table 2. This separation is not very surprising as it can be explained at an intuitive level: (i) all buildings should at least have support in h_{mid}; (ii) all candidate objects should have a low H_p and at least have support from h_{low} (and some support from h_{hi} when under overhanging structures) and all overhanging structures should have a high H_p and at least have support from h_{high} and lack of support from h_{mid}.

We perform kernel density estimation [11] on the feature-space obtained above to compute the likelihood densities, $p(\boldsymbol{r}|\ell)$, for each of the four class labels (ℓ): tall vertical structures (buildings), ground plane, candidate objects and overhanging structures.

$$p(\boldsymbol{r}|\ell) = \frac{1}{n^c} \sum_{i=1}^{n^c} K_H(\boldsymbol{r} - \boldsymbol{r}_i^c) \tag{2}$$

where $\boldsymbol{r}_i^c$ are the feature vectors of all the patches i in the training set belonging to the class c, $K_H(\boldsymbol{u}) = \alpha(H)K(H^{-1/2}\boldsymbol{u})$ is a kernel function and H is a bandwidth matrix which scales the kernel support to be radially symmetric. In our implementation, we define $K(\boldsymbol{u}) = k(\boldsymbol{u}^\top \boldsymbol{u})$ and use the following biweight kernel,

$$k(u) = \begin{cases} (1-u)^3 & 0 \le u \le 1 \\ 0 & u > 1 \end{cases}$$

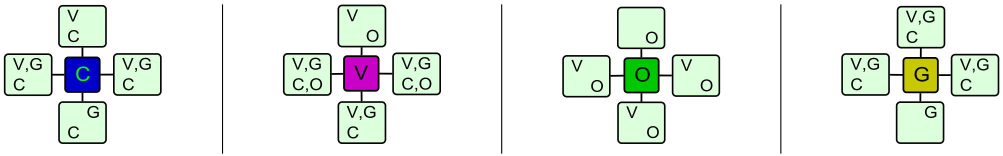


Figure 4. Smoothness constraints in the MRF. In each subimage, we show the possible labels of each of the neighboring patches given the label of the (colored) center patch. Note that the constraint is dependent on the position of the neighbor relative to the center patch. V = Tall vertical structure, O = Overhanging structure, G = Ground, C = Candidate objects.

The biweight kernel is efficient to compute and we found that it gave performance comparable to other more complex kernels.

Fig. 3 shows the various steps of the likelihood density estimation process for one frame. Note that, in particular, the vertical structure likelihoods in Fig. 3(d) capture the visible extent of the buildings all the way to the base - something difficult to achieve with a simple heuristic on H.

4.3. Prior Probabilities

4.3.1 MRF Priors for Structure Classification

In addition to the likelihoods of structural labels, we model the smoothness inherent in scene structures through Markov Random Field (MRF) priors on a pairwise basis. The a priori joint probability of labels, $P(L = \ell)$, is difficult to define in general but is tractable for MRFs. If L is represented as an MRF, then the prior probability $P(\ell)$ is a Gibbs distribution [1] given by,

$$P(\ell) \propto e^{-E_S(\ell)} \tag{3}$$

where $E_S(\ell)$ is the cost associated with ℓ and is modeled as a pairwise smoothness term between neighboring patches. L can be formulated as an MRF on the grid-graph represented by the patch-grid Π, with the 4-connectivity imposed by the grid structure defining the edges, if the following conditions are satisfied: (i) L is a random field, and (ii) the label for a particular patch given those of all other patches, depends only on the labels of its neighboring patches. These are reasonable assumptions in this scenario. For example, the identification of a patch as a building patch might depend on whether its neighboring patches are ground but has little to do with the identity of the patches spatially far removed from it.

The next step is to define the smoothness cost E_S from which the prior probability $P(L = \ell)$ can be computed.

4.3.2 Smoothness Cost

We use the smoothness term to model valid configurations of scene objects possible from the camera viewpoint. Thus, for each patch, we will consider its neighboring patch and define the cost of associating a pair of labels with the two patches. The neighboring patch is defined as the patch which is 4-connected to this patch and is also close in its world depth Z^w. Thus, we treat two patches which are neighbors in the image space but distant in the world space to have no conditional dependence on each other's labeling in the MRF network. This condition essentially cuts the grid-graph along depth discontinuities before the MRF framework starts any label propagation. The remaining neighbors are now depth neighbors as well and it is easier to reason about what objects can (or cannot) be near what objects.

Let p and q be two neighboring patches from the patch-grid and Z_p^w and Z_q^w represent the world depths of these patches. Define a binary variable $\rho_n = \delta(t)$ such that,

$$\delta(t) = \begin{cases} 1 & \frac{|Z_p^w - Z_q^w|}{Z_p^w} < Z_n \\ 0 & \text{otherwise} \end{cases}$$

for testing depth neighborhood using a ratio threshold Z_n. Then the smoothness cost assigned to the patch pair (p, q) is,

$$E_S = \rho_{bp}\rho_n D(p, q, L(p), L(q)),$$

where ρ_{bp} is the constant weight factor applied to the smoothness term and is set empirically, $L(p)$ and $L(q)$ are the labels of p and q and $D(.)$ is a function that measures the compatibility between those labels.

The function $D(.)$ is defined by considering not only the labels $L(p)$ and $L(q)$ (which is usual in typical BP formulations) but also considering if patch p is a left, right, top or bottom neighbor of patch q. The function can enforce different costs for the same pair of labels $(L(p), L(q))$ if p is below q than if p is above q. For example, if p is a building patch and q is below p, then q can be one of building, candidate object or ground. However, if q is above p, then q can only be a building patch since one cannot expect to see either ground or candidate objects along the top edge of a building. Note that in the first scenario, the candidate label is included to allow (say) a pedestrian patch very close in depth to the building patch to occlude the lower part of the building. The allowed choices for $L(q)$ would also be the same as the first scenario if p and q are horizontal neighbors. The function $D(.)$ is a binary function which imposes a penalty 1 if a pair of labels is inconsistent and a penalty 0 otherwise. The set of allowed labels for each patch pair is presented in Fig. 4.

In our implementation, the MAP estimation (1) is done with the max-product belief propagation algorithm [20].

Table 2. Parameter settings for structure classification

Patch Grid	w	h			
	12 px	16 px			
XZ Histogram	X_{res}	Z_{res}	h_{low}	h_{mid}	h_{hi}
	0.1 m	0.1 m	0-2 m	2-4 m	4-8 m
MRF	Z_n	ρ_{bp}			
	0.1	1.0			

5. Experiments

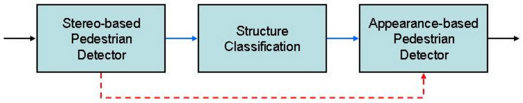

Stereo-based Pedestrian Detector
Structure Classification
Appearance-based Pedestrian Detector

Figure 5. Evaluation system flow diagram

5.1. Data and Evaluation Methodology

Data. We experimentally validate our approach on the public dataset available from [8]. This dataset consists of four challenging test sequences (640 × 480 at 15Hz) of busy shopping streets with multiple people moving in different directions, taken on different days and under different weather conditions.

Evaluation Methodology. For evaluation, we use the annotations available with the dataset. All the sequences are completely annotated up to a distance of ≈25 m. The exact subset of annotations used by Ess et al.[8] for evaluating their system is not available. Table 3 compares the sizes of the annotation subsets used by us with [8]. The subset of annotations we use includes pedestrians which: (i) are not severely occluded by other pedestrians, (ii) are not significantly clipped by the camera field-of-view, and (iii) are at least 50 pixels high. For a detection to be counted as correct, it has to overlap with an annotation by more than 50% using the intersection-over-union measure [9].

Parameter Settings. Table 2 lists all the parameter settings used in our system.

5.2. Experimental Results

We evaluated the efficacy of structure-classification in removing non-pedestrian image regions in this dataset. In table 4, we present some figures on the average percentage area of the image rejected as non-candidate-pedestrian region. On an average, we are able to reject more than 80% of the image area using structure classification itself. This leaves less than 20% of the image area where potential pedestrians may be present which, in fact, results in only 10-20 pedestrian ROIs which have to be classified by the appearance-based classifier. This is a significant gain both from system speed as well as performance standpoint.

The ROC curves for our system were obtained by varying the decision boundary threshold for the appearance-based classifier stage in our pipeline. For the solid-red curves the detections from the stereo-based detector were directly fed to the appearance-based classifier (dotted-red path in Fig. 5). For the solid-blue curves (solid-blue path in Fig. 5), the detection ROIs were first validated by the structure-classification (SC) module and any ROIs with more than 75% non-candidate label patches (from amongst all labeled patches within the ROI) were removed. The remaining boxes were then fed to the appearance-based classifier as before. In this evaluation methodology, any performance gain with the use of the SC module clearly indicates that the ROIs removed by SC were in fact difficult examples for the appearance classifier.

Table 3. Evaluation data used for our experiments

Sequence Name	#Frames	Annotations used	
		Ess et al.[8]	Us
Seq00	499	1578	1581
Seq01	1000	5193	5207
Seq02	451	2359	1731
Seq03	354	1828	1724

Table 4. Pixel area removed by structure classification

Seq00	Seq01	Seq02	Seq03
88.00%	82.25%	85.13%	76.00%

To validate our implementation of the appearance-based classifier on this dataset, we ran the publicly available HoG pedestrian detector code from Dalal and Triggs [4] on Seq01. The ROCs are shown in Fig. 8. The method is run on all frames of Seq01 considering only annotations bigger than 100 pixels high as valid (giving a total of 2072 annotations), since in their implementation the candidate boxes are required to be of sizes close to or bigger than 64 × 128 pixels. The same annotations were used to evaluate our system in this plot which shows that our classifier performs equivalent to their implementation.

Fig. 6 compares the performance of our system individually on each of the test sequences with a few other approaches from the literature. The results from Ess et al. [8] are included for Seq01, Seq02 and Seq03. They used Seq00 for training and no test results are available. For Seq01, they show an improvement in performance in [7] which is also shown. In the other tests shown in the rest of the ROC curves, pedestrian sizes vary tremendously. Output of our baseline system (without SC, the red curve) can be assumed to be similar to [4] if it is run on these sets. Notice the improvement in performance achieved by using the SC module. In particular, we are able to remove a substantial number of FPs as we tend towards high detection rates.

The proposed method is implemented on an in-vehicle pedestrian detection system. Our implementation takes about 25ms per frame for stereo computation, template pedestrian detection and structure classification on an Intel Dual-Core processor. The overall system including the appearance classifier runs at about 10Hz.

Acknowledgments. The authors thank Ted Camus, Ben Southall (Sarnoff), Wei Zhang, Ann Do (FHWA) and John Harding and Jennifer Percer (NHTSA) for helpful discussions and feedback.

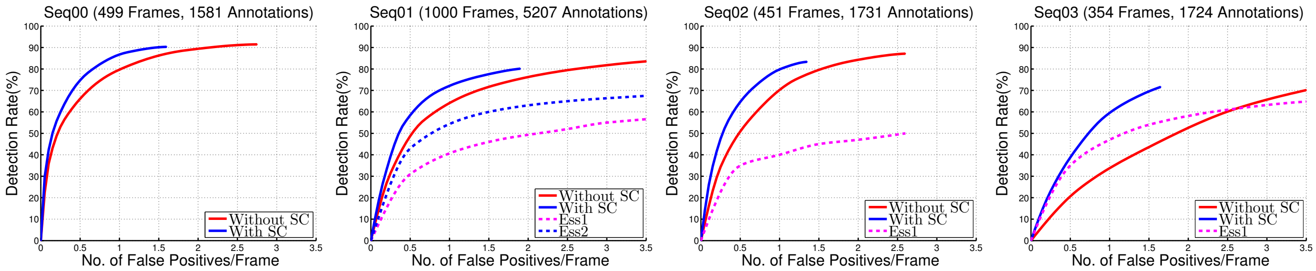


Figure 6. ROC curves showing our system performance on the four sequences Seq00-03. Also shown are comparisons with other representative approaches from the literature: *Ess1* [8], *Ess2* [7], *Dalal* [4].

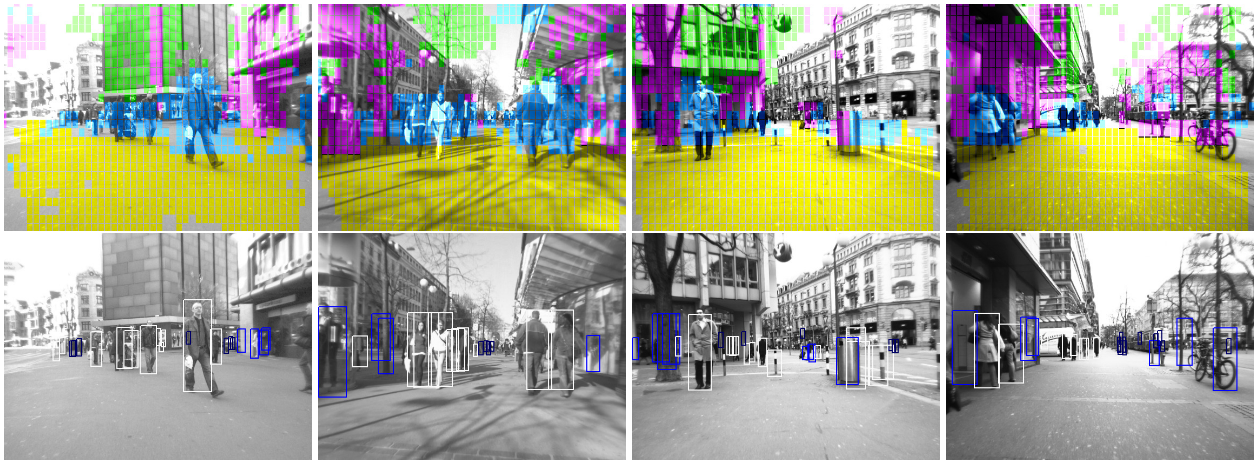


Figure 7. Examples of Structure Classification. First row shows scene labeling from sequences in [8]. In the second row the input boxes from the stereo-based pedestrian detector are denoted in blue if they are rejected by structure classification and in white otherwise.

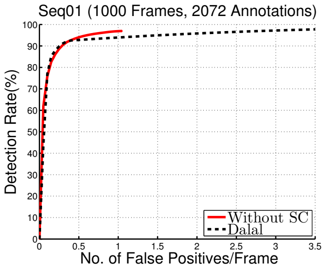


Figure 8. ROC curve comparing our system performance on Seq01 with Dalal and Triggs [4] for pedestrians > 100 pixels high.

References

[1] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 32(2), 1974.

[2] G. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *ECCV*, 2008.

[3] P. Chang, D. Hirvonen, T. Camus, and B. Southall. Stereo-based object detection, classification, and quantitative evaluation with automotive applications. In *IEEE Int. Workshop on Machine Vision for Intelligent Vehicles*, 2005.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, pages 886–893, 2005.

[5] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: A benchmark. In *CVPR*, 2009.

[6] M. Enzweiler and D. Gavrila. A mixed generative-discriminative framework for pedestrian classification. In *CVPR*, 2008.

[7] A. Ess, B. Leibe, K. Schindler, and L. V. Gool. A mobile vision system for robust multi-person tracking. In *CVPR*, pages 734–741, 2008.

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

[9] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2008 (VOC2008). http://www.pascal-network.org/challenges/VOC/voc2008/workshop/index.html.

[10] D. M. Gavrila and S. Munder. Multi-cue pedestrian detection and tracking from a moving vehicle. *IJCV*, 73:41–59, 2007.

[11] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*. Springer, 2001.

[12] D. Hoiem, A. A. Efros, and M. Hebert. Geometric context from a single image. In *ICCV*, October 2005.

[13] D. Hoiem, A. A. Efros, and M. Hebert. Putting objects in perspective. In *CVPR*, June 2006.

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

[15] K. Mikolajczyk and C. Schmid. A performance evaluationof local descriptors. *PAMI*, 27, 2005.

[16] A. Shashua, Y. Gdalyahu, and G. Hayun. Pedestrian detection for driver assistance systems: Single-frame classification and system level performance. In *In Proc. of the IEEE Intelligent Vehicle Symposium*, 2004.

[17] D. Tran and D. A. Forsyth. Configuration estimates improve pedestrian finding. In *In NIPS*, 2007.

[18] O. Tuzel, F. Porikli, and P. Meer. Human detection via classification on riemannian manifolds. In *CVPR*, 2007.

[19] P. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. *IJCV*, 63:153–161, 2005.

[20] Y. Weiss and W. T. Freeman. On the optimality of solutions of the max-product belief propagation algorithm in arbitrary graphs. *IEEE Trans. Information Theory, Special Issue on Codes on Graphs and Iterative Algorithms*, 47, 2001.

[21] C. Wojek and B. Schiele. A dynamic conditional random field model for joint labeling of object and scene classes. In *ECCV*, 2008.

[22] Q. Yang, L. Wang, R. Yang, H. Stewenius, and D. Nister. Stereo matching with color-weighted correlation, hierarchical belief propagation and occlusion handling. In *CVPR*, 2006.