In "Binocular Vision: Development, Depth Perception and Disorders", Chapter 9, NOVA publishers, April 2010.
http://www.novapublishers.com

Stereo-based candidate generation for pedestrian protection systems

David Geronimo*, Angel D. Sappa* and Antonio M. López*+
*Computer Vision Center and +Computer Science Department
Universitat Autònoma de Barcelona, 08193, Bellaterra, Barcelona, Spain
dgeronimo@cvc.uab.es

July 29, 2009

Abstract

This chapter describes a stereo-based algorithm that provides candidate image windows to a latter 2D classification stage in an on-board pedestrian detection system. The proposed algorithm, which consists of three stages, is based on the use of both stereo imaging and scene prior knowledge (i.e., pedestrians are on the ground) to reduce the candidate searching space. First, a successful road surface fitting algorithm provides estimates on the relative ground-camera pose. This stage directs the search toward the road area thus avoiding irrelevant regions like the sky. Then, three different schemes are used to scan the estimated road surface with pedestrian-sized windows: (a) uniformly distributed through the road surface (3D); (b) uniformly distributed through the image (2D); (c) not uniformly distributed but according to a quadratic function (combined 2D-3D). Finally, the set of candidate windows is reduced by analyzing their 3D content. Experimental results of the proposed algorithm, together with statistics of searching space reduction are provided.

1 Introduction

According to the World Health Organization, every year almost 1.2 million people are killed and 50 million are injured in traffic accidents worldwide [11]. These dramatic statistics highlight the importance of the research in traffic safety, which involves not only motor companies but also governments and universities.

Since the early days of the automobile, in the beginning of 20th century, and along with its popularization, different mechanisms were successfully incorporated to the vehicle with the aim of improving its safety. Some examples are turn signals, seat-belts and airbags. These mechanisms, which rely on physical devices, were focused on improving safety specifically when accidents where happening. In the 1980s a sophisticated new line of research began to pursue

safety in a preventive way: the so-called *advanced driver assistance systems* (ADAS). These systems provide information to the driver and perform active actions (e.g., automatic braking) by the use of different sensors and intelligent computation. Some ADAS examples are *adaptive cruise control* (ACC), which automatically maintains constant distance to a front-vehicle in the same lane, and *lane departure warning* (LDW), which warns when the car is driven out the lane unadvertently.

One of the more complex ADAS are *pedestrian protection systems* (PPSs), which aim at improving the safety of these vulnerable road users. Attending to the number of people involved in vehicle-to-pedestrian accidents, e.g., 150 000 injured and 7 000 killed people each year in the European Union [6], it is clear that any improvement in these systems can potentially save many human lifes. PPSs detect the presence of people in a specific area of interest around the host vehicle in order to warn the driver, perform braking actions and deploy external airbags in the case of an unavoidable collision. The most used sensor to detect pedestrians are cameras, contrary to other ADAS such as ACC, in which active sensors like radar or lidar are employed. Hence, Computer Vision (CV) techniques play a key role in this research area, which is not strange given that vision is the most used human sense when driving. People detection has been an important topic of research since the beginning of CV, and it has been mainly focused on applications like surveillance, image retrieval and human-machine interfaces. However, the problem faced by PPSs differs from these applications and is far from being solved. The main challenges of PPSs are summarized in the following points:

- Pedestrians have a high variability in pose (human body can be viewed as a highly deformable target), clothes (which change with the weather, culture, and people), distance (typically from 5 to at least 25m), sizes (not only adults and children are different, but also there are many different human constitutions), viewpoints (e.g., front, back or side viewed).

- The variability of the scenarios is also considerable, i.e., the detection takes place in outdoor dynamic urban roads with cluttered background and illumination changes.

- The requirements in terms of misdetections and computational cost are hard: these systems must perform real-time actions at very low miss rates.

The first research works in PPSs were presented in the late 1990s. Papageorgiou et al. [10] proposed to extract candidate windows by exhaustively scanning the input image and classify them with support vector machines based on Haar Wavelet features. This two-step candidate generation and classification scheme has been used in a countless number of detection systems: from faces [14], vehicles or generic object detection to human surveillance and image retrieval [3]. The simplest candidate generation approach is the exhaustive scan, also called sliding window: it consists in scanning the input image with pedestrian-sized windows (i.e., with a typical aspect ratio around 1/2) at all the possible scales

and positions. Although this candidate generation method is generic and easy to implement, it can be improved by making use of some prior knowledge from the application. Accordingly, during the last decade researchers have tried to exploit the specific aspects of PPSs to avoid this generation technique. Some cues used for generating candidates are vertical symmetry [1], infrared hot spots [4] and 3D points [7]. However, the proposed techniques that exploit them pose several problems that make the systems not reliable in real-world scenarios. For example, in the case of 2D analysis, the number of false negatives (i.e., discarded pedestrians) can not be guaranteed to be low enough: symmetry relies on vertical edges, but in many cases the illumination conditions or background clutter make them disappear. Hot spot analysis in infrared images holds a similar problem because of the environmental conditions [2]. On the other hand, although stereo stands as a more reliable cue, the aforementioned techniques also hold problems. In the case of [7], the algorithm assumes a constant road slope, so the problems appear when the road orientation is not constant which is common in urban scenarios.

This chapter presents a candidate generation algorithm that reduces the number of windows to be classified while minimizes the number of wrongly discarded targets. This is achieved by combining a prior-knowledge criterion, *pedestrians-on-the-ground*, and using 3D data to filter the candidates. This procedure can be seen as a conservative but reliable approach, which in our opinion is the most convenient option for this early step of the system.

The remainder of the manuscript is organized as follows. First, we introduce the proposed candidate generation algorithm with a brief description of its components and their objective. Then, the three stages in which the algorithm is divided are presented: Sect. 3 describes the road surface estimation algorithm, Sect. 4 presents the road scanning and Sect. 5 addresses the candidate filtering. Finally, Sect. 6 provides experimental results of the algorithm output. In Sect. 7, conclusions and future work is presented.

2 Algorithm Overview

A recent survey on PPSs by Gerónimo et al. [8] proposes a general architecture that consists of six modules, in which most of the existing systems can be fit. The modules (enumerated in the order of the pipeline process) are: 1) preprocessing, 2) foreground segmentation, 3) object classification, 4) verification and refinement, 5) tracking and 6) application. As can be seen, modules 2) and 3) correspond to the steps presented in the introduction. The algorithm presented in this chapter consists in a candidate generation algorithm to be used in the foreground segmentation module, which gets an input image and generates a list of candidates where a pedestrian is likely to appear, to be sent to the next module, the classifier. There are two main objectives to be carried out in this module. The first is to reduce the number of candidates, which directly affects the performance of the system both in terms of speed (the fewer the candidates sent to the classifier the less the computation time is) and detection rates (neg-

atives can be pre-filtered by this module). The second is not to discard any pedestrian, otherwise the later modules will not be able to correct the wrong filtering.

The proposed algorithm is divided into three stages, as illustrated in Fig. 1.

1. **Road surface estimation** computes the relative position and orientation between the camera and the scene (Sect. 3).
2. **Road scanning** places 3D windows over the estimated road surface using a given scanning method (Sect. 4).
3. **Candidate filtering** filters out windows that do not contain enough stereo evidence of containing vertical objects (Sect. 5).

Next sections describe each stage in detail.

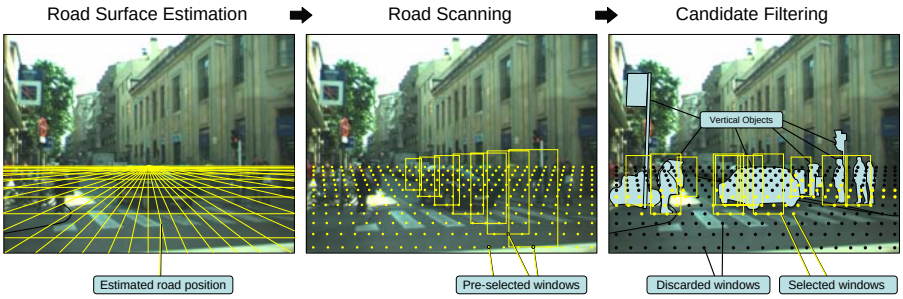


Figure 1: Stages of the proposed algorithm.

3 Road Surface Estimation

The first stage is focused on adjusting the candidate searching space to the region where the probability of finding a pedestrian is higher. In the context of PPSs, the searching space is the road, hence irrelevant regions like the sky can be directly ommited from the processing. The main targets of road surface estimation are two-fold: first, to fit a surface (a plane in the current implementation) to the road; second, to compute the relative position and orientation (pose) of the camera[1] with respect to such a plane.

A world coordinate system (X_W, Y_W, Z_W) is defined for every acquired stereo image, in such a way that: the $X_W Z_W$ plane is contained in the current road fitted plane, just under the camera coordinate system (X_C, Y_C, Z_C); the Y_W axis contains the origin of the camera coordinate system; the $X_W Y_W$ plane contains the X_C axis and the $Z_W Y_W$ plane contains the Z_C axis. Due to that, the six extrinsic parameters (three for the position and three orientation angles) that

[1] Also referred to as camera extrinsic parameters.

refer the camera coordinate system to the world coordinate system reduce to just three, denoted in the following as (Π, Φ, Θ) (i.e., camera height, roll and pitch). Figure 2 illustrates the world and camera coordinate systems.

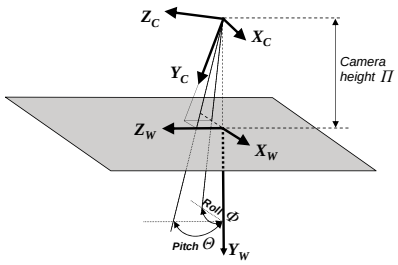


Figure 2: Camera coordinate system (X_C, Y_C, Z_C) and world coordinate system (X_W, Y_W, Z_W).

From the (Π, Φ, Θ) parameters, in most situations the value of Φ (roll) is very close to zero. This condition is fulfilled as a result of a specific camera mounting procedure that fixes Φ at rest, and because in normal urban driving situations this value scarcely varies [9].

The proposed approach consists of two substages detailed below (more information in [13]): *i*) 3D data point projection and cell selection and *ii*) road plane fitting and ROIs setting.

3.1 3D data point projection and cell selection

Let $D(r, c)$ be a depth map provided by the stereo pair with R rows and C columns, in which each array element (r, c) is a scalar that represents a scene point of coordinates (x_C, y_C, z_C), referred to the camera coordinate system (Fig. 2). The aim at this first stage is to find a compact subset of points, ζ, containing most of the road points. To speed up the whole algorithm, most of the processing at this stage is performed over a 2D space. Initially, 3D data points are mapped onto cells in the $(Y_C Z_C)$ plane, resulting in a 2D discrete representation $\psi(o, q)$; where $o = \lfloor D_Y(r, c) \cdot \varsigma \rfloor$ and $q = \lfloor D_Z(r, c) \cdot \varsigma \rfloor$, ς representing a scale factor that controls the size of the bins according to the current depth map (Fig. 3). The scaling factor is aimed at reducing the projection dimensions with respect to the whole 3D data in order to both speed up the plane fitting algorithm and be robust to noise. It is defined as: $\varsigma = ((R + C)/2)/(\Delta X + \Delta Y + \Delta Z)/3)$; $(\Delta X, \Delta Y, \Delta Z)$ is the working range in 3D space. Every cell of $\psi(o, q)$ keeps a reference to the original 3D data points projected onto that position, as well as a counter with the number of mapped points.

From that 2D representation, one cell per column (i.e., in the Y-axis) is selected, relying on the assumption that the road surface is the predominant geometry in the given scene. Hence, it picks the cell with the largest number of points in each column of the 2D projection. Finally, every selected cell is rep-

resented by the 2D barycenter $(0, (\sum_{i=0}^{n} y_{C_i})/n, (\sum_{i=0}^{n} z_{C_i})/n)$ of its n mapped points. The set of these barycenters defines a compact representation of the selected subset of points, ζ. Using both one single point per selected cell and a 2D representation, a considerable reduction in the CPU time is reached during the road plane fitting stage.

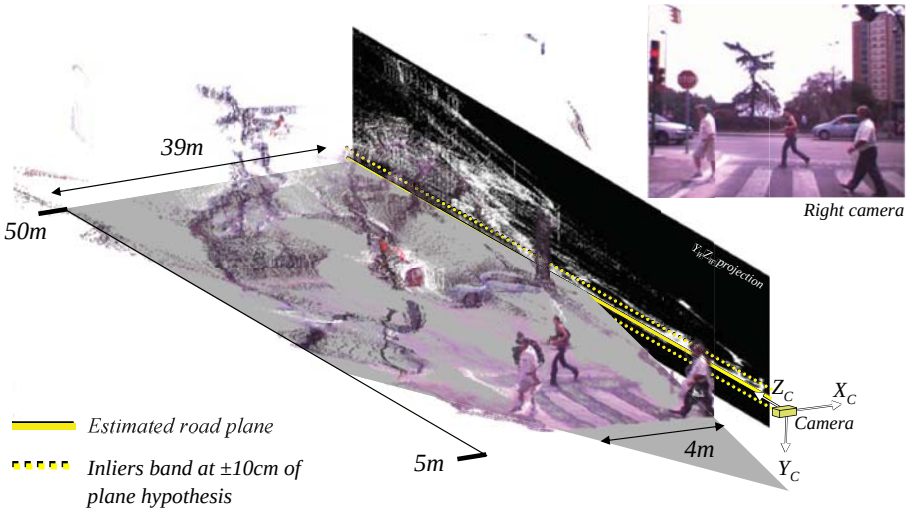


Figure 3: YZ Projection and road plane estimation.

3.2 Road plane fitting

The outcome of the previous substage is a compact subset of points, ζ, where most of them belong to the road. As stated in the previous subsection, Φ (roll) is assumed to be zero, hence the projection is expected to contain a dominant 2D line corresponding to the road together with noise coming from the objects in the scene.

The plane fitting stage consits of two steps. The first one is a 2D straight line parametrisation, which selects the dominant line corresponding to the road. It uses a RANSAC based [5] fitting applied over 2D barycenters intended for removing outlier cells. The second step computes plane parameters by means of a least squares fitting over all 3D data points contained into inlier cells.

Initially, every selected cell is associated with a value that takes into account the amount of points mapped onto that position. This value will be considered as a probability density function. The normalized probability density function is defined as follows: $pdf_i = n_i/N$; where n_i represents the number of points mapped onto the cell i and N represents the total amount of points contained in the selected cells.

Next, a cumulative distribution function, F_j, is defined as: $F_j = \sum_{i=0}^{j} pdf_i$; If the values of F are randomly sampled at n points, the application of the

inverse function F^{-1} to those points leads to a set of n points that are adaptively distributed according to pdf_i.

3.2.1 Dominant 2D Straight Line Parametrisation

At the first step a RANSAC based approach is applied to find the largest set of cells that fit a straight line, within a user defined band. In order to speed up the process, a predefined threshold value for inliers/outliers detection has been defined (a band of ± 10 cm was enough for taking into account both data point accuracy and road planarity); an automatic threshold could be computed for inliers/outliers detection, following robust estimation of standard deviation of residual errors [12]. However, it would increase CPU time since robust estimation of standard deviation involves computationally expensive algorithms (e.g., sorting functions).

Repeat L times

(a) Draw a random subsample of 2 different barycenter points (P_1, P_2) according to the probability density function pdf_i using the above process;

(b) For this subsample, indexed by l ($l = 1, ..., L$), compute the straight line parameters $(\alpha, \beta)_l$;

(c) For this solution, compute the number of inliers among the entire set of barycenter points contained in ζ, as mentioned above using a ± 10 cm margin.

3.2.2 Road Plane Parametrisation

(a) From the previous 2D stright line parametrisation choose the solution that has the highest number of inlier;

(b) Compute (a, b, c) plane parameters by using the whole set of 3D points contained in the cells considered as inliers, instead of the corresponding barycenters. To this end, the least squares fitting approach [15], which minimizes the square residual error $(1 - ax_C - by_C - cz_C)^2$ is used;

(c) In case the number of inliers is smaller than 40% of the total amount of points contained in ζ (e.g., severe occlusion of the road by other vehicles), those plane parameters are discarded and the ones corresponding to the previous frame are used as the correct ones.

4 Road Scanning

Once the road is estimated, candidates are placed on the 3D surface and then projected to the image plane to perform the 2D classification. The most intuitive scanning scheme is to distribute windows all over the estimated plane in a

uniform way, i.e., in a $n_x \times n_z$ grid, with n_x sampling points in the road's X axis and n_z in the Z axis. Each sampling point on the road is used to define a set of scanning windows, to cover the different sizes of pedestrian, as will be described later.

Let us define $Z_{Cmin} = 5$m as the minimum ground point seen from the camera[2], $Z_{Cmax} = 50$m as the furthest point, and $\tau = 100$ the number of available sampling positions along the Z_C axis of the road plane (a, b, c). Given that the points are evenly placed over the 3D plane, the corresponding image rows can be computed by using the plane and projection equations. Hence, the sampled rows in the image are:

$$y = y_0 + \frac{f}{bz} - f\frac{c}{b} \ , \tag{1}$$

where $z = Z_{Cmin} + i\delta_Z \ \forall i \in \{0, .., n_z - 1\}$; $\delta_Z = (Z_{Cmax} - Z_{Cmin})/n_z$ is the 3D sampling stride; (a, b, c) are the plane parameters; f is the camera focal; and y_0 is the y coordinate of the center point of the camera in the image. The same procedure is applied to the X xis, e.g., from $X_{C_{min}}$ to $X_{C_{max}}$ with the n_x sampling points. We refer to this scheme as *Uniform World Scanning*.

As can be appreciated in Fig. 4(a), this scheme has two main drawbacks: it oversamples far positions (i.e., Z close to Z_{Cmax}) and undersamples near positions (i.e., the sampling is too sparse when Z is close to the camera). In order to ammend these problems, it is clear that the sampling cannot rely only on the world but must be focused on the image. In fact, the sampling is aimed at extracting candidates in the 2D image. According to this, we compute the minimum and maximum image rows corresponding to the Z range:

$$y_{Z_{Cmax}} = y_0 + \frac{f}{bZ_{Cmax}} - f\frac{c}{b} \ , \tag{2}$$

$$y_{Z_{Cmin}} = y_0 + \frac{f}{bZ_{Cmin}} - f\frac{c}{b} \ , \tag{3}$$

and evenly place the sampling points between these two image rows using:

$$y = y_{Z_{Cmin}} + i\delta_{im} \quad \forall i \in \{0..n_z - 1\} \ , \tag{4}$$

where $\delta_{im} = (y_{Z_{Cmin}} - y_{Z_{Cmax}})/n_z$. In this case, the corresponding z in the plane (later needed to compute the window size) is

$$z = \frac{f}{c + b(y - y_0)} \ . \tag{5}$$

In the case of X axis, the same procedure as in the first scheme can be used. This scheme is called *Uniform Image Scanning*. In this case, it is seen in Fig. 4(b) that although the density of sampling points for the closer Z_C is appropiate,

[2]With a camera of $6mm$ focal, oriented to the road avoiding to capture the hood, the first road point seen is around 4 to 5 meters from the camera.

the far Z_C are undersampled, i.e., the space between sampling points is too big (see histogram of the same figure).

Figure 5 displays the sampling functions with respect to the Z_C scanning positions and the image Y axis. The *Uniform to Image*, in dotted-dashed-blue, draws a linear function since the windows are evenly distributed over the available rows. On the contrary, the *Uniform to Road*, in dashed-red, takes the form of an hyperbola as a result of the perspective projection. The aforementioned over- and under-sampling in the top and bottom regions of this curve can be also seen in this figure. Attending to the problems of these two approaches, we finally propose the use of a non-uniform scheme that provides a more sensible sampling, i.e., neither over- nor under-sampling the image or the world. The idea is to sample the image with a curve in between the two previous schemes, and adjust the row-sampling according to our needs, i.e., mostly linear in the bottom region of the image (close Z) and logarithmic-like for further regions (far Z), but avoiding over-sampling. In our case, we use a quadratic function of the form $y = ax^2 + bx + c$, constrained to pass through the intersection points between the linear and hyperbolic curves and by a user defined point (i_{user}, y_{user}) between the two original functions. The curve parameters can be found by solving the following system of equations:

$$\begin{bmatrix} {i_{max}}^2 & i_{max} & 1 \\ {i_{min}}^2 & i_{min} & 1 \\ {i_{user}}^2 & i_{user} & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} y_{Z_{C\,max}} \\ y_{Z_{C\,min}} \\ y_{user} \end{bmatrix} , \tag{6}$$

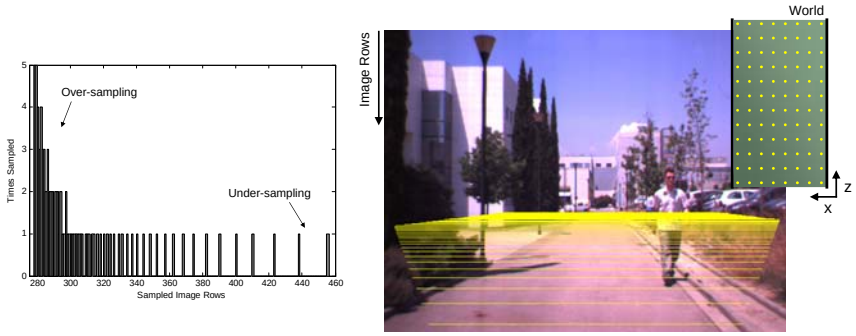
where $i_{min} = 0$ and $i_{max} = n_z - 1$. For example, in the non-uniform curve in Fig. 5 (solid-black line), $y_{user} = i_{min} + (i_{max} - i_{min}) \times \kappa$ and $i_{user} = i_{min} + (i_{max} - i_{min}) \times \lambda$, where $\kappa = 0.6$ and $\lambda = 0.25$. For the X_C axis we follow the same procedure as with the other schemes. The resulting scanning, called *non-uniform scanning*, can be seen in Fig. 4(c).

Once we have the set of 3D windows on the road, they are used to compute the corresponding 2D windows to be classified. We assume a pedestrian to be $h = 1.70$m high, with an standard deviation $\sigma = 0.2$m. In the case of body width, the variability is much bigger than height, so a width margin is used to adjust most of human proportions and also leave some space for the extremities. Hence, the width is defined as a ratio of the height, specifically 1/2. For example, the mean pedestrian window sizes 1.70×0.85m, independently of the extra-margin taken by the classifier[3].

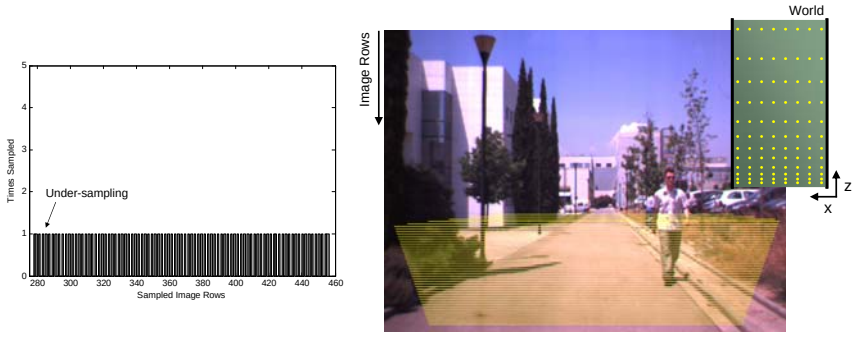
5 Candidate Filtering

The final stage of the algorithm is aimed at discarding candidate windows by making use of the stereo data (Fig. 6). The method starts by aligning the camera coordinate system with the world coordinate system (see Fig. 2) with the aim of compensating pitch angle Θ, computed in Sect. 3. Assuming that

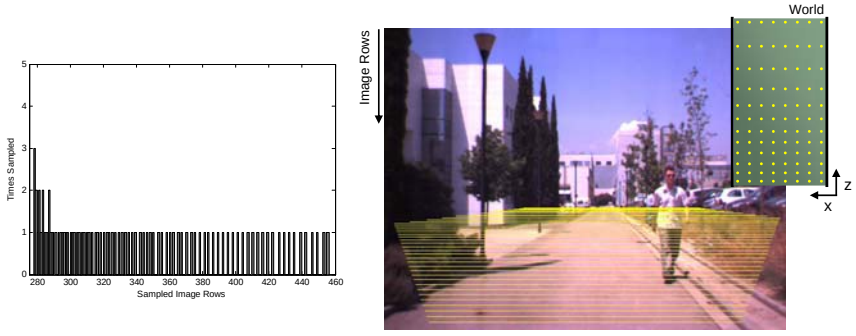
[3]Dalal et al. [3] demonstrate that adding some margin to the window (33% in their case) results in a performance improvement in their classifier.



(a) Uniform Road Scanning



(b) Uniform Image Scanning



(c) Non-Uniform Scanning

Figure 4: The three different scanning schemes. Right column shows the scanning rows using the different schemes and also a representation of the scan over the plane. In order to enhance the figure visualization just 50% of the lines are shown. The histograms of sampled image rows are shown on the left column; under- and over-sampling problems can be seen.

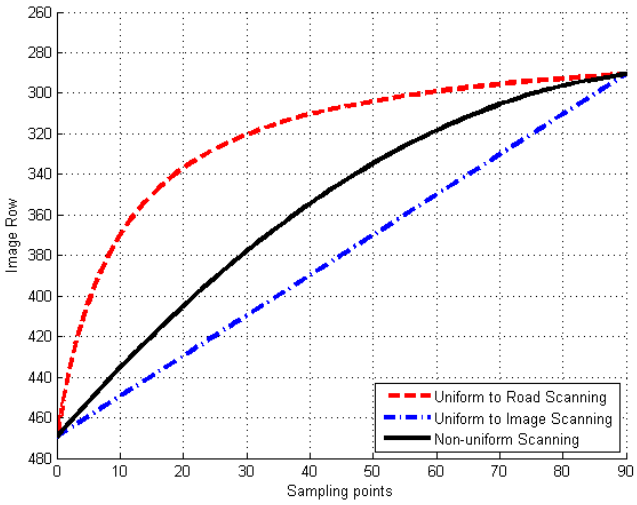


Figure 5: Scanning functions. A non-uniform road scanning with parameters $\kappa = 0.6$ and $\lambda = 0.25$ is between the uniform to road and to image curves, hence achieving a more sensible scan.

roll is set to zero, as described in the aforementioned section, the coordinates of a given point $p_{(x,y,z)}$, referred to the new coordinate system, are computed as follows:

$$\begin{aligned} p_{x_R} &= p_x \\ p_{y_R} &= cos(\Theta)p_y - sin(\Theta)p_z \\ p_{z_R} &= sin(\Theta)p_y + cos(\Theta)p_z \ . \end{aligned} \tag{7}$$

Then, rotated points located over the road[4] are projected onto a uniform grid G_P in the fitted plane (Sect. 3), where each cell has a size of $\sigma \times \sigma$. A given point $p(x_R, y_R, z_R)$ votes into the cell (i, j), where $i = \lfloor x_R/\sigma \rfloor$ and $j = \lfloor z_R/\sigma \rfloor$. The resulting map G_P is shown in Fig. 7(b). As can be seen, cells far away from the sensor tend to have few projected points. This is caused by two factors. First, the number of projected points decreases directly with the distance, as a result of perspective projection. Second, the uncertainty of stereo reconstruction also increases with distance, thus the points of an ideal vertical and planar object would spread wider into G_P as the distance of these points increases. In order to amend this problem, the number of points projected onto each cell in G_P are

[4]Set of points placed in a band from 0 to $2m$ over the road plane, assuming that this is the maximum height of a pedestrian.

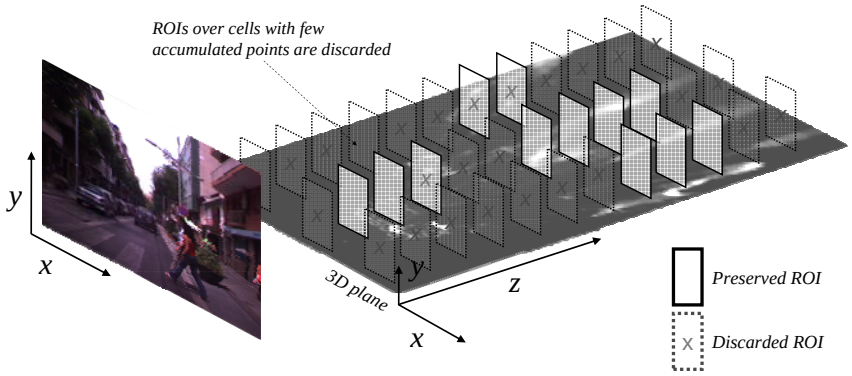


Figure 6: Schematic illustration of the candidate filtering stage.

reweighted and redistributed. The reweighting function is

$$G_{RW}(i,j) = j\sigma G_P(i,j) \ , \tag{8}$$

where $j\sigma$ corresponds to the real depth of the cell. The redistribution function consists in propagating the value of G_{RW} to its neighbours as follows:

$$G(i,j) = \sum_{s=i-\eta/2}^{i+\eta/2} \sum_{t=j-\eta/2}^{j+\eta/2} G_{RW}(s,t) \ , \tag{9}$$

where η is the stereo uncertainty at a given depth (in cells): $\eta = \text{uncertainty}/\sigma$. Uncertainty is computed as a function of disparity values:

$$\text{uncertainty} = f{\cdot}\text{baseline}\frac{\mu}{\text{disparity}^2} \ , \tag{10}$$

where baseline is the baseline of the stereo pair in meters, f is the focal length in pixels and μ is the correlation accuracy of the stereo. The resulting map G, after reweighting and redistribution processes, is illustrated in Fig. 7(c). The filtering consists in discarding the candidate windows that are over cells with less than χ points, which is set experimentally. In our implementation, this parameter is low in order to fulfill the conservative criterion mentioned in the introduction, i.e., in this early system module false positives are preferred than false negatives.

6 Experimental Results

The evaluation of the algorithm has been made using data taken from an on-board stereo rig (Bumblebee from Point Grey, `http://www.ptgrey.com`, Fig. 8). The stereo pair has a baseline of 0.12m and each camera has a focal of 6mm and provides a resolution of 640×480 pixel (the figures in the paper show the

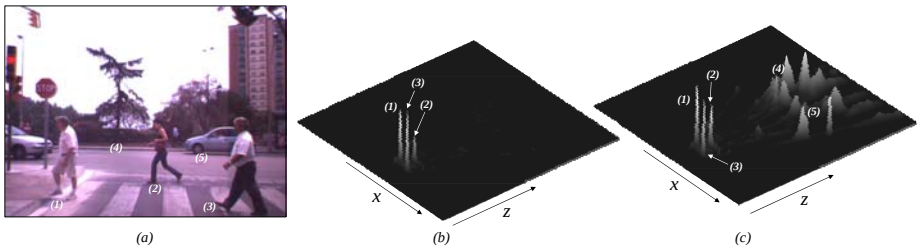


Figure 7: Probability map of vertical objects on the road plane. (a) Original frame. (b) Raw projection G_P. (c) Reweighted and redistributed vertical projection map of the frame 3D points.

right sensor image). The HFOV is 43° and the VFOV is 33°, which allows to detect pedestrians at a minimum of 5m, and the camera reconstruction software provides 3D information until 50m, which coincides with the parameters described in Sect. 3.

As introduced in Sect. 1, one of the most used candidate generation methods is sliding window. Although this method does not perform an explicit foreground segmentation, which is our motivation, it is useful as a reference to evaluate the benefits of our proposal. Let us say that we must detect pedestrians up to 50m, which measure around 12×24 pixels (of course the size will slightly differ depending on the focal and the size of the sensor pixels). On the other hand, the nearest pedestrian fully seen, at 5m, is about 140×280 pixels. Hence, a regular exhaustive scan algorithm must place windows of the scales between these two distances at all the possible positions. If a scale variation is assumed to be 1.2 and the position stride is 4 pixels, the number of windows is over 100 000. However, smaller windows need a smaller stride between them, so the number can range between from 200 000 to 400 000.

We have selected 50 frames taken from urban scenarios with the aforementioned stereo camera and applied the proposed algorithm. The parameters for the road surface estimation are $L = 100$ and $\varsigma = 0.68$. In the case of the scanning, we have used the non-uniform scheme with $\tau = 90$ sampling points, $\kappa = 0.5$ and $\lambda = 0.25$. The scanning in the X_C axis is made in $X_C = \{-10, .., 10\}$m with a stride of 0.075m. For each selected window, 10 different sizes are tested (the smallest 0.75×1.5m and the biggest 0.95×1.8m). The algorithm selects about 50 000 windows, which is a reduction of about $75 - 90\%$ with respect to the sliding window, depending on the stride of this latter. Then, we apply the filtering stage with a cell size of $\sigma = 0.2$ and $\chi = 2000$, reducing again a 90% the number of candidates. This represents a reduction of $97 - 99\%$ compared to the sliding window. Figure 9 illustrates the results in six of the frames used to test the algorithm. As can be seen, the pedestrians in the scenario are correctly selected as candidates, while other free-space areas are discarded to be classified. In addition, attending to the results, the number of false negatives is marginal, which is a key factor for the whole sytem performance.

Figure 8: Stereo pair used in our acquisition system.

7 Conclusions

We have presented a three-stage candidate generation algorithm to be used in the foreground segmentation module of a PPS. The stages consist of road surface estimation, road scanning and candidate filtering. Experimental results demonstrate that the number of candidates to be sent to the classifier can be reduced by a $97-99\%$ compared to the typical sliding window approach, while minimizing the number of false negatives to around 0%. Future work will be focused on the research of algorithms to fuse the cues used to select the candidates, which can potentially improve the proposed pipeline process.

8 Acknowledgements

The authors would like to thank Mohammad Rouhani for his ideas with the road scanning section. This work was supported by the Spanish Ministry of Education and Science under project TRA2007-62526/AUT and research programme Consolider Ingenio 2010: MIPRCV (CSD200700018); and Catalan Government under project CTP 2008 ITT 00001. David Gerónimo was supported by Spanish Ministry of Education and Science and European Social Fund grant BES-2005-8864.

References

[1] M. Bertozzi, A. Broggi, R. Chapuis, F. Chausse, A. Fascioli, and A. Tibaldi. Shape–based pedestrian detection and localization. In *Proc. of the IEEE International Conference on Intelligent Transportation Systems* , pages 328–333, Shangai, China, 2003.

[2] C.-Y. Chan and F. Bu. Literature review of pedestrian detection technologies and sensor survey. Technical report, Institute of Transportation Studies, Uni. of California at Berkeley, 2005.

[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 886–893, San Diego, CA, USA, 2005.

[4] Y. Fang, K. Yamada, Y. Ninomiya, B. Horn, and I. Masaki. A shape-independent method for pedestrian detection with far-infrared images. *IEEE Trans. on Vehicular Technology* , 53(6):1679–1697, 2004.

[5] M. Fischler and R. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Graphics and Image Processing*, 24(6):381–395, June 1981.

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

[7] D.M. Gavrila, J. Giebel, and S. Munder. Vision–based pedestrian detection: The PROTECTOR system. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 13–18, Parma, Italy, 2004.

[8] D. Gerónimo, A. López, A.D. Sappa, and T. Graf. Survey of pedestrian detection for advanced driver assistance systems. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (in press)*, 2009.

[9] R. Labayrade and D. Aubert. A single framework for vehicle roll, pitch, yaw estimation and obstacles detection by stereovision. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 31–36, Columbus, OH, USA, June 2003.

[10] C. Papageorgiou and T. Poggio. A trainable system for object detection. *International Journal on Computer Vision* , 38(1):15–33, 2000.

[11] M. Peden, R. Scurfield, D. Sleet, D. Mohan, A.A. Hyder, E. Jarawan, and C. Mathers. *World Report on road traffic injury prevention*. World Health Organization, Geneva, Switzerland, 2004.

[12] P. Rousseeuw and A. Leroy. *Robust Regression and Outlier Detection*. John Wiley & Sons, New York, 1987.

[13] A.D. Sappa, F. Dornaika, D. Ponsa, D. Gerónimo, and A. López. An efficient approach to onboard stereo vision system pose estimation. *IEEE Trans. on Intelligent Transportation Systems*, 9(3):476–490, 2008.

[14] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 511–518, Kauai, HI, USA, 2001.

[15] C. Wang, H. Tanahashi, H. Hirayu, Y. Niwa, and K. Yamamoto. Comparison of local plane fitting methods for range data. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 663–669, Kauai Marriot, HW, USA, December 2001.

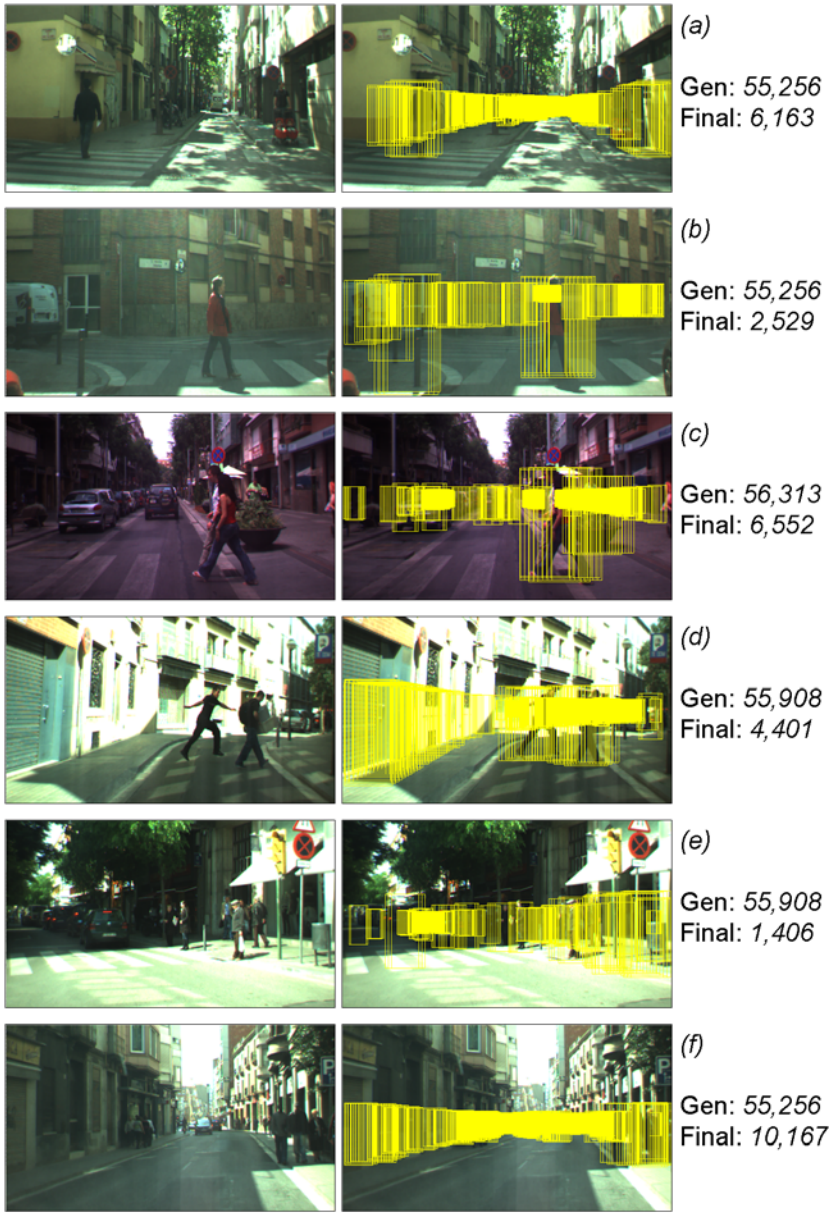


Figure 9: Experimental results. The left column shows the original real urban frames in which the proposed algorithm is applied. The middle column corresponds to the final windows after the filtering step. The right column shows the number of windows generated after the scanning (*Gen*) and after the filtering (*Final*). In order to enhance the visualization the different scales tested for each sampling point are not shown, so just one candidate per point was drawn.