Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

A featureless and stochastic approach to on-board stereo vision system pose

Fadi Dornaika [a,*], Angel D. Sappa [b]

[a] University of the Basque Country, 20018 San Sebastian, Spain
[b] Computer Vision Center, 08193 Bellaterra, Barcelona, Spain

ARTICLE INFO

Article history:
Received 2 November 2006
Received in revised form 22 December 2008
Accepted 22 December 2008

Keywords:
On-board stereo vision system
Pose estimation
Featureless approach
Particle filtering
Image warping

ABSTRACT

This paper presents a direct and stochastic technique for real-time estimation of on-board stereo head's position and orientation. Unlike existing works which rely on feature extraction either in the image domain or in 3D space, our proposed approach directly estimates the unknown parameters from the stream of stereo pairs' brightness. The pose parameters are tracked using the particle filtering framework which implicitly enforces the smoothness constraints on the estimated parameters. The proposed technique can be used with a driver assistance applications as well as with augmented reality applications. Extended experiments on urban environments with different road geometries are presented. Comparisons with a 3D data-based approach are presented. Moreover, we provide a performance study aiming at evaluating the accuracy of the proposed approach.

1. Introduction

In recent years, several techniques to on-board vision pose estimation have been proposed [7,10,23,26,30,32]. The main application domain was advanced driver assistance. The proposed approaches can be broadly classified into two different categories: highways and urban. For each category, the vision sensor can be either a monocular camera or a stereo head. Most of the techniques proposed for highways environments are focused on lane and car detection, looking for an efficient driver assistance system. On the other hand, in general, techniques for urban environments are focused on collision avoidance or pedestrian detection. Although in both domains a similar objective is pursued, it is very challenging to develop a generic algorithm able to cope with both problems. The real-time estimation of on-board vision system pose—position and orientation—is a challenging task since (i) the sensor undergoes motions due to the vehicle dynamics and the road imperfections, and (ii) the viewed scene is unknown and continuously changing.

Of particular interest is the estimation of on-board camera's position and orientation related to the 3D road plane. Note that since the 3D plane parameters are expressed in the camera coordinate system, the camera's position and orientation are equivalent to the 3D plane parameters. Algorithms for fast road plane estimation are very useful for driver assistance applications as well as for augmented reality applications. For the former ones, the ability to use continuously updated plane parameters (vehicle pose) will considerably make the tasks of obstacles and objects detection more efficient [17,33]. For the latter ones, one can for instance insert real or synthetic objects into the video captured by the on-board vision system based on the estimated road plane parameters. These continuously updated parameters provided by the vision sensor will make the inserted objects seem as a physical part of the scene. If the used road plane parameters are constant then the inserted object may suffer from misalignment whenever the actual plane parameters change due to the car's dynamics and road's imperfections. However, dealing with an urban scenario is more difficult than dealing with highways scenario since the prior knowledge as well as visual features are not always available in these scenes.

In general, monocular vision systems avoid problems related to 3D Euclidean geometry by using the prior knowledge of the environment as an extra source of information. For instance, (a) a road with a constant width is assumed [13,12]; (b) the car is driven along two parallel lane markings, which are projected to the left and to the right of the image [25]; (c) after an initial calibration process the camera's position and pitch angle remain constant through the time [28]; to mention a few.

Although prior knowledge has been extensively used to tackle the driver assistance problem, it may lead to wrong results. Hence, considering a constant camera's position and orientation is not a valid assumption to be used in urban scenarios, since both of them are easily affected by road imperfections or artifacts (e.g., rough road, speed bumpers), car's accelerations, uphill/downhill driving, among others. Facing up to this problem [13] introduces a technique for estimating vehicle's yaw, pitch, and roll. However, since a single camera is used, this work is based on the assumption that

* Corresponding author.
E-mail addresses: dornaika@cvc.uab.es (F. Dornaika), sappa@cvc.uab.es (A.D. Sappa).

0262-8856/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2008.12.006

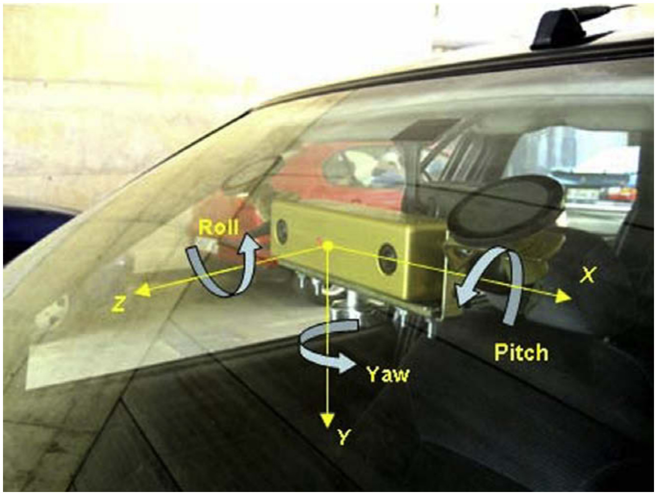

Fig. 1. On-board stereo vision sensor with its corresponding coordinate system.

some parts of the road have a constant width (e.g., lane markings). Similarly, Liang et al. [25] proposes to estimate camera's orientation by assuming that the vehicle is driven along two parallel lane markings. Unfortunately, none of these two approaches can be generalized to be used in urban scenarios, since in general lanes are not as well defined as those of highways. In [32], authors use a single mounted camera. An extended Kalman filter has been used in order to infer a state vector including the vehicle rigid motion (six degrees of freedom) and the camera pose where the measurements are given by the eight-parameter planar motion field and the readings of the velocity and yaw rate sensors.

The use of prior knowledge has also been considered by some stereo vision based techniques to simplify the problem and to speed up the whole processing by reducing the amount of information to be handled. For instance, some of the aforementioned assumptions are also considered when stereo systems are used: flat road [6,20] or constant camera pose [5], among others.

In the literature, many application-oriented stereo systems have been proposed. For instance, the edge based v-disparity approach proposed in [22], for an automatic estimation of horizon lines and later on used for applications such as obstacle or pedestrian detection (e.g., [4,11,21]), only computes 3D information over local maxima of the image gradient. A sparse disparity map is computed in order to obtain a real time performance. Additionally to the obstacle or pedestrian detection the authors present an atmospheric visibility measurement system using v-disparity information provided by stereo vision [18]. Recently, this v-disparity approach has been extended to a u–v-disparity concept in [19]. In this work, dense disparity maps are used instead of only relying on edge based disparity maps. Working in the disparity space is an interesting idea that is gaining popularity in on-board stereo vision applications, since planes in the original Euclidean space become straight lines in the disparity space.

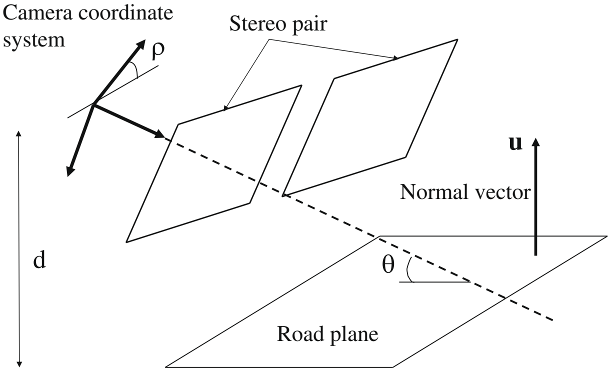

Fig. 2. The time-varying road plane parameters d and $\mathbf{u}$. θ denotes the pitch angle and ρ the roll angle.

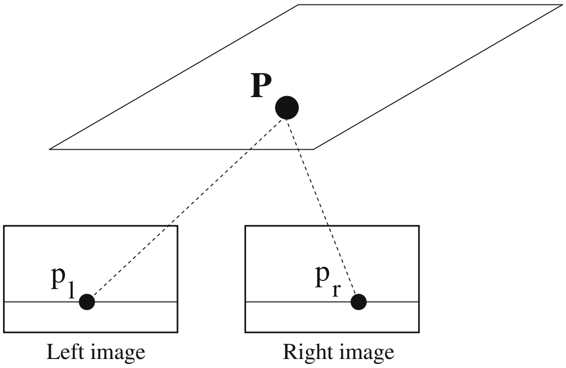

Fig. 3. The mapping between corresponding left and right road pixels is given by a linear transform.

In computer vision community, many works have addressed the detection and estimation of planes using images [1,9,27]. However, these works rely on feature extraction. This holds true for 3D camera motion methods (e.g., [24,34]).

In [29], we proposed an approach for on-line vehicle pose estimation using a commercial stereo head. Although the proposed technique does not require the extraction of visual features in the images, it is based on dense depth maps and on the extraction of a dominant 3D plane that is assumed to be the road plane.

As can be seen, existing works adopt the following main stream. First, features are extracted either in the image space (optical flow, edges, ridges, interest points) or in the 3D Euclidean space (assuming the 3D data are built online). Second, a deterministic estimation is then invoked in order to recover the unknown parameters. In this paper, we propose a novel paradigm that is based on raw stereo images provided by a stereo head. Moreover, the new paradigm includes a stochastic technique since the aim is to track the vehicle pose parameters—the road plane parameters—given stereo pairs arriving in a sequential fashion.

The stochastic tracking relies on the particle filtering framework. The proposed technique could be indistinctly used for urban or highway environments, since it is not based on a specific visual traffic feature extraction neither in 2D nor in 3D. The use of particle filtering schemes is useful for obtaining a lock on the estimated parameters even when perturbing factors such as occlusions and video streaming discontinuities appear.

Our proposed method has a significant advantage over existing methods since it does not require road segmentation neither dense matching—two difficult and time-consuming tasks. Moreover, to the best of our knowledge, the work presented in this paper is the first work estimating road parameters directly from the rawbrightness images using a particle filter.

The rest of the paper is organized as follows. Section 2 describes the problem we are focusing on as well as some backgrounds. Section 3 briefly describes a 3D data-based method. Section 4 presents the proposed stochastic technique. Section 5 gives some experimental results and method comparisons. Section 6 provides a performance study using synthesized stereo sequences. In the sequel, the "road plane parameters" and the "pose parameters" will refer to the same entity.

2. Problem formulation and background

2.1. Experimental setup

A commercial stereo vision system (Bumblebee from Point Grey[1]) was used. It consists of two Sony ICX084 color CCDs with 6 mm focal length lenses. Bumblebee is a pre-calibrated system that does not require in-field calibration. The baseline of the stereo head is 12 cm and it is connected to the computer by a IEEE-1394 connector. Right and left color images can be captured at a resolution of 640 × 480 pixels and a frame rate near to 30 fps. This vision system includes a software able to provide the 3D data. Fig. 1 shows an illustration of the on-board stereo vision system as well as its mounting device.

The problem we are focusing on can be stated as follows. Given a stream of stereo pairs provided by the on-board stereo head we like to recover the parameters of the road plane for every captured stereo pair. Since we do not use any feature that is associated with road structure, the computed plane parameters will completely define the pose of the on-board vision sensor. This pose is represented by the height d and the plane normal $\mathbf{u} = (u_x, u_y, u_z)^T$ from which two independent angles can be inferred (see Fig. 2). In the sequel, the pitch angle will refer to the angle between the camera's optical axis and the road plane; and the roll angle will refer to the angle between the camera horizontal axis and the road plane (see Fig. 2). Due to the reasons mentioned above, these parameters are not constant and should be estimated online for every time instant. Note that the three angles (pitch, yaw, and roll) associated with the stereo head orientation can vary. However, only the pitch and roll angles can be estimated from the 3D plane parameters.

2.2. Image transfer function

Before going into the details of the proposed approach, this section will describe the geometric relation between road pixels belonging to the same stereo pair—the left and right images. It is well-known [15,16] that the projections of 3D points belonging to the same plane onto two different images are related by a 2D projective transform having eight independent parameters—*homography*. In our setup, the right and left images are horizontally rectified.[2] Let $p_r(x_r, y_r)$ and $p_l(x_l, y_l)$ be the right and left projection of an arbitrary 3D point P belonging to the plane (d, u_x, u_y, u_z) (see Fig. 3). In the case of a rectified stereo pair where the left and right images have the same intrinsic parameters, the right and left coordinates of corresponding pixels belonging to the road plane are related by the following linear transform (the homography reduces to a linear mapping)

$$x_l = h_1 x_r + h_2 y_r + h_3 \tag{1}$$

$$y_l = y_r \tag{2}$$

where h_1, h_2, and h_3 are function of the intrinsic and extrinsic parameters of the stereo head and of the plane parameters. For our setup (rectified images with the same intrinsic parameters), those parameters are given by:

$$h_1 = 1 + b\frac{u_x}{d} \tag{3}$$

$$h_2 = b\frac{u_y}{d} \tag{4}$$

$$h_3 = -b\, u_0\frac{u_x}{d} - b\, v_0\frac{u_y}{d} + \alpha b\frac{u_z}{d} \tag{5}$$

where b is the baseline of the stereo head, α is the focal length in pixels, and (u_0, v_0) is the image center (principal point).

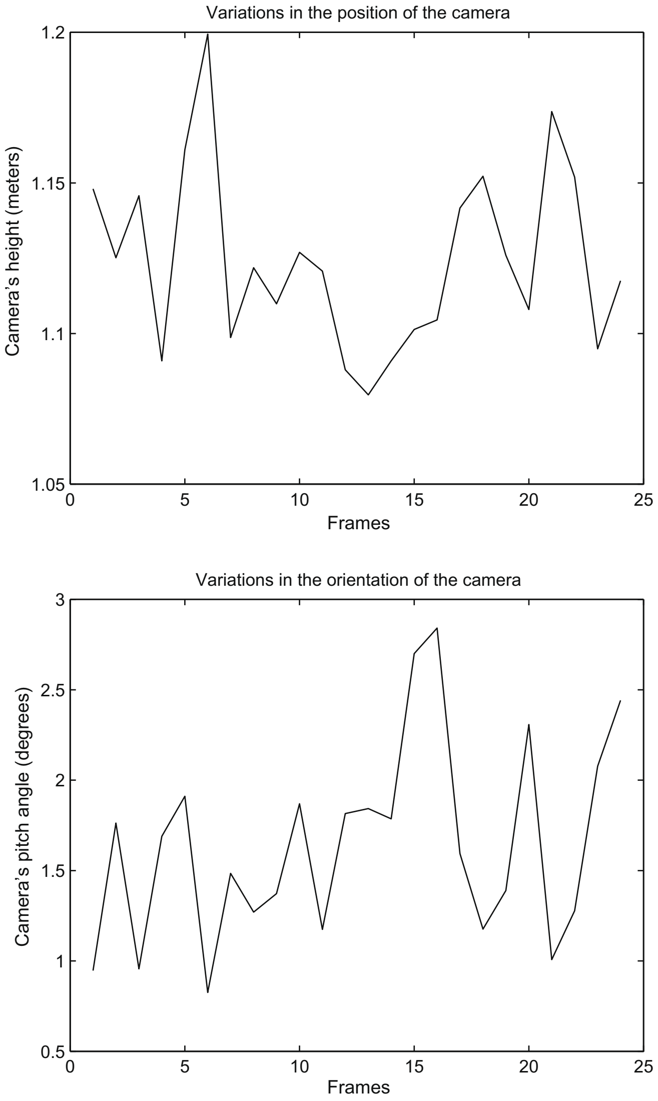

Fig. 4. The camera's position and orientation, related to the current plane fitting the road. Note that only 2 fps are plotted.

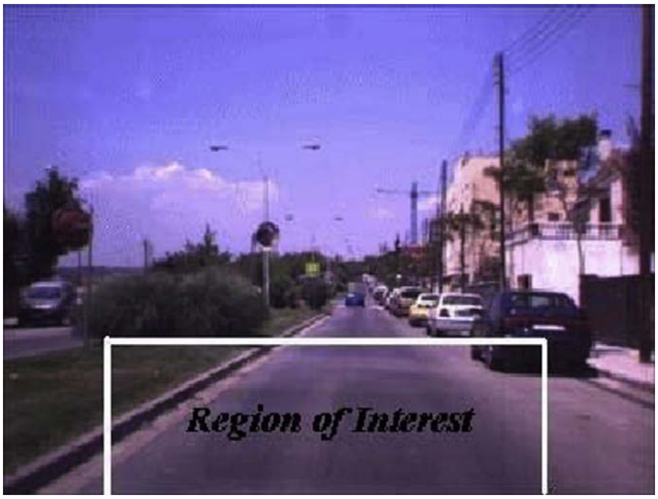

Fig. 5. A typical region of interest associated with the right image. In this example, the height of region of interest is set to one third of the total image height.

[1] www.ptgrey.com

[2] The use of non-rectified images will not have any theoretical impact on our developed method. However, the image transfer function will be given by a general homography.

1. Initialization $t = 0$: Generate N state samples $\mathbf{a}_0^{(1)}, \ldots, \mathbf{a}_0^{(N)}$ according to some prior density $p(\mathbf{b}_0)$ and assign them identical weights, $w_0^{(1)} = \ldots = w_0^{(N)} = 1/N$
2. At time step t, we have N weighted particles $(\mathbf{a}_{t-1}^{(j)}, w_{t-1}^{(j)})$ that approximate the posterior distribution of the state $p(\mathbf{b}_{t-1}|\mathbf{z}_{1:(t-1)})$ at previous time step
   (a) Resample the particles proportionally to their weights, *i.e.* keep only particles with high weights and remove particles with small ones. Resampled particles have the same weights
   (b) Draw N particles according to the dynamic model $p(\mathbf{b}_t|\mathbf{b}_{t-1} = \mathbf{a}_{t-1}^{(j)})$ (11), (12), and (13). These particles approximate the predicted distribution $p(\mathbf{b}_t|\mathbf{z}_{1:(t-1)})$
   (c) Weight each new particle proportionally to its likelihood:
   $$w_t^{(j)} = \frac{p(\mathbf{z}_t|\mathbf{b}_t = \mathbf{a}_t^{(j)})}{\sum_{m=1}^{N} p(\mathbf{z}_t|\mathbf{b}_t = \mathbf{a}_t^{(m)})}$$
   where $p(\mathbf{z}_t|\mathbf{b}_t)$ is given by (18). The set of weighted particles approximates the posterior $p(\mathbf{b}_t|\mathbf{z}_{1:t})$
   (d) Give an estimate of the state $\hat{\mathbf{b}}_t$ as the MAP:
   $$\hat{\mathbf{b}}_t = \arg\max_{\mathbf{b}_t} p(\mathbf{b}_t|\mathbf{z}_{1:t}) \approx \arg\max_{\mathbf{a}_t^{(j)}} w_t^{(j)}$$

Fig. 6. Estimating the road plane parameters using a particle filter.

3. 3D data-based approach

In [29], we have proposed an approach for on-line vehicle pose estimation using the above commercial stereo head. It aims to compute camera's position and orientation—the road plane parameters. The proposed technique consists of two stages. First, a dense depth map of the scene is computed by the provided reconstruction software that utilizes a dense matching technique. Second, the parameters of a 3D plane fitting to the road are estimated using a RANSAC based least squares fitting. Moreover, the second stage includes a filtering step that aims at reducing the number of 3D points that are processed by the RANSAC technique.

Independently of the road geometry, the provided results could be understood as a piecewise planar approximation, due to the fact that the road parameters are continuously computed and updated. The proposed technique could be indistinctly used for urban or highway environments, since it is not based on a specific visual traffic feature extraction but on raw 3D data points.

This technique has been tested on different urban environments. The proposed algorithm took, on average, 350 ms per frame. This CPU time does not include the dense stereo matching and 3D reconstruction tasks.

Fig. 4 presents the estimated camera's position and orientation during a short sequence. Notice that this short video sequence was recorded on a quite flat road; hence a maximum height variation of about 10 cm is produced after starting the motion, due to vehicle's acceleration.

The main drawback of the proposed 3D data technique is its high CPU time. Moreover, it requires a dense 3D reconstruction of the captured images. In the present study, this method is used for comparing the proposed stochastic technique. Moreover, it will be used to initialize the proposed approach in some cases.

4. A featureless and stochastic approach

Our aim is to estimate the pose parameters from the stream of stereo pairs. In other words, we track the stereo head pose over time.[3] In this section, we propose a novel approach that directly infers the plane parameters from the stereo pair using a particle filtering framework.

4.1. Background

The idea of a particle filter (also known as Sequential Monte Carlo (SMC) algorithm) was independently proposed and used by several research groups. These algorithms provide flexible tracking frameworks as they are neither limited to linear systems nor require the noise to be Gaussian and proved to be more robust to distracting clutter as the randomly sampled particles allow to maintain several competing hypotheses of the hidden state. Therefore, the main advantage of particle filtering methods is the fact that any loose of track will not lead to a permanent loss of the object. Note that when the noise can be modelled as Gaussian and the observation model is linear then the solution will be given by the Kalman filter.

Particle filtering is an inference process which can be used in the context of probabilistic tracking. It aims at estimating the unknown time-*t* state $\mathbf{b}_t$ from a set of noisy observations (images), $\mathbf{z}_{1:t} = \{\mathbf{z}_1, \cdots, \mathbf{z}_t\}$ arriving in a sequential fashion [3,8,14]. Two important components of this approach are the state transition and observation models whose most general forms can be given by:

State transition model $\mathbf{b}_t = D_t(\mathbf{b}_{t-1}, \mathbf{n}_t)$ (6)

Observation model $\mathbf{z}_t = O_t(\mathbf{b}_t, v_t)$ (7)

where $\mathbf{n}_t$ is the system noise, D_t is the dynamic model, v_t is the observation noise, and O_t models the observer. The particle filter approximates the posterior distribution $p(\mathbf{b}_t|\mathbf{z}_{1:t})$ by a set of weighted particles or samples $\{\mathbf{b}_t^{(j)}, \pi_t^{(j)}\}_{j=1}^N$. Each element $\mathbf{b}_t^{(j)}$ represents the hypothetical state of the object and $\pi_t^{(j)}$ is the corresponding discrete probability. Then, the state estimate can be set, for example, to the minimum mean square error or to the maximum *a posteriori* (MAP): $\arg\max_{\mathbf{b}_t} p(\mathbf{b}_t|\mathbf{z}_{1:t})$.

Based on such generative models, the particle filtering method is a Bayesian filtering method that recursively evaluates the posterior density of the target state at each time step conditionally to the history of observations until the current time.

4.2. Approach

4.2.1. Dynamic model

At any given time, the road plane parameters are given by the plane normal $\mathbf{u}_t$, a unit vector, and the distance d_t between the

[3] This is equivalent to tracking the road plane parameters.

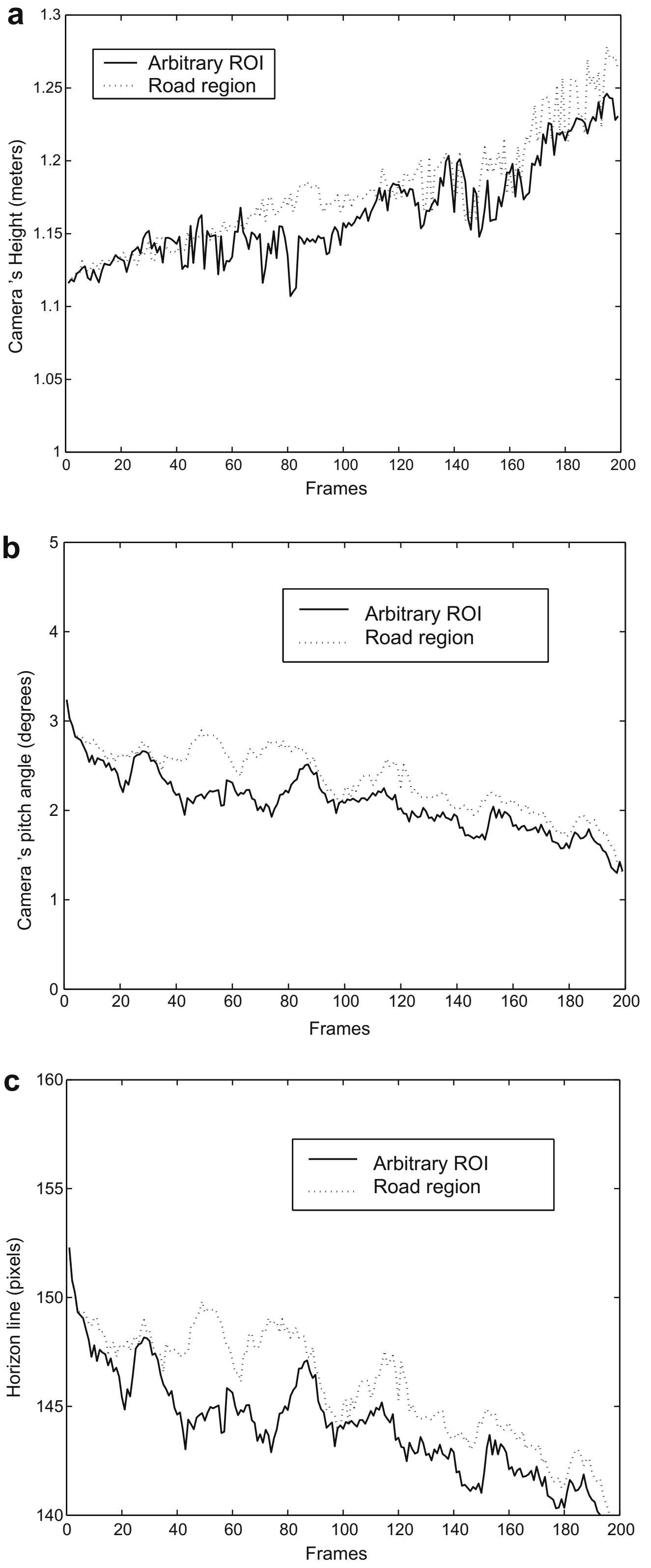

Fig. 7. Camera's position and orientation, estimated by the particle filter. The plotted solutions correspond to the maximum a posteriori solution. The solid curves correspond to a ROI having an arbitrary width. The dotted curves correspond to a ROI containing the road image only.

camera center and the plane. These parameters can be encapsulated into a 3-vector $\frac{\mathbf{u}_t}{d_t}$. Therefore, the state vector $\mathbf{b}_t$ representing the plane parameters will be given by

$$\mathbf{b}_t = (b_{x(t)}, b_{y(t)}, b_{z(t)})^T = \left(\frac{u_{x(t)}}{d_t}, \frac{u_{y(t)}}{d_t}, \frac{u_{z(t)}}{d_t}\right)^T \tag{8}$$

Note that the vector $\mathbf{b}_t$ fully describes the current road plane parameters since we have:

$$\mathbf{u}_t = \frac{\mathbf{b}_t}{\|\mathbf{b}_t\|} \tag{9}$$

$$d_t = \frac{1}{\|\mathbf{b}_t\|} \tag{10}$$

Therefore, the state vector has three independent degrees of freedom. One can notice that knowing the normal plane $\mathbf{u}_t$ the current two degrees of freedom associated with the camera orientation—the pitch and roll angles—can be easily recovered.

Since the camera's height and orientation, the plane parameters, are ideally constant, the dynamics of these parameters can be well modelled by a Gaussian noise:

$$b_{x(t)} = b_{x(t-1)} + \epsilon_t \tag{11}$$

$$b_{y(t)} = b_{y(t-1)} + \epsilon_t \tag{12}$$

$$b_{z(t)} = b_{z(t-1)} + \epsilon_t \tag{13}$$

where ϵ is a noise (scalar) drawn from a centered Gaussian distribution $\mathcal{N}(0, \sigma)$. The standard deviation σ can be computed from previously recorded camera pose variations. However, we believe that fixed standard deviations or context-based standard deviations are more appropriate since they are directly related to the kind of perturbations and to the video rate.

At first glance, one can think that the above dynamic model, given by Eqs. (11)–(13), cannot model all kinds of car motions such as car maneuvering and running over speed bumpers. However, according to the particle filtering literature, this dynamic model can handle all dynamics as long as the support of the noise distribution is large enough (large standard variation for a Gaussian noise), and hence a large number of particles is needed. Thus, the price to pay when using this simple model is an increased computational time.

In case where efficiency and accurate dynamic models are needed, more sophisticated dynamic models can be used such as the following adaptive dynamic model:

$$b_{x(t)} = b_{x(t-1)} + \delta b_x + \epsilon_t \tag{14}$$

$$b_{y(t)} = b_{y(t-1)} + \delta b_y + \epsilon_t \tag{15}$$

$$b_{z(t)} = b_{z(t-1)} + \delta b_z + \epsilon_t \tag{16}$$

where $(\delta b_x, \delta b_y, \delta b_z)^T$ is a deterministic shift in the state vector. In the literature, usually this shift is dynamically computed as the difference between the estimated states at two consecutive frames, i.e., $(\delta b_x, \delta b_y, \delta b_z)^T = \hat{\mathbf{b}}_{(t-1)} - \hat{\mathbf{b}}_{(t-2)}$. Notice that the simple dynamic model is a particular case of the model given by Eqs. (14)–(16).

We now analyze how the standard deviation σ is governing the diffusion of the plane parameters through Eqs. (11)–(13). For clarity purpose, we use a realistic setting for the current road plane parameters, i.e., $\mathbf{u} = (0, 1, 0)$ and $d = 1.2$ m. We assume that these values have been diffused three times using the above equations and each time the following three random vectors have been chosen $(\sigma, \sigma, \sigma)^T, (2\sigma, 2\sigma, 2\sigma)^T$, and $(3\sigma, 3\sigma, 3\sigma)^T$. If σ is set to 0.004 then the drawn heights d will be 1.194 m,1.188 m, and 1.183 m, respectively. The drawn normal vectors will have 0.39°, 0.77°, and 1.15° deviation from the nominal direction $\mathbf{u} = (0, 1, 0)$.

Since the interval of any Gaussian distribution is roughly given by $[-3\sigma, 3\sigma]$, the above example shows that the maximum height changes will be in the interval $[-0.017 \text{ m}, +0.017 \text{ m}]$ and the maximum cone aperture containing the drawn normals will be 2.3°.

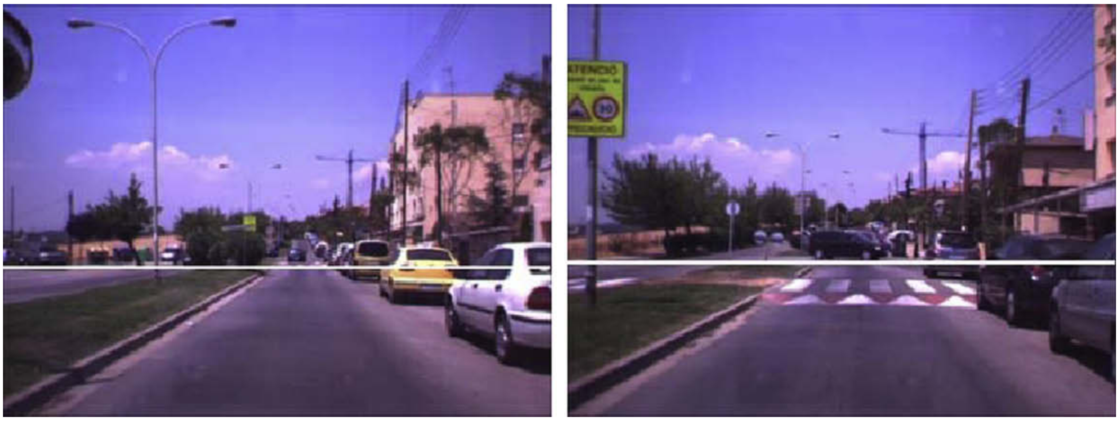

Fig. 8. The estimated horizon line associated with frames 55 and 182 using the road image only.

4.2.2. Observation model

The observation model should relate the state $\mathbf{b}_t$ (plane parameters) to the measurement $\mathbf{z}_t$ (stereo pair). We use the following fact: if the state vector $\mathbf{b}_t$ encodes the actual plane parameters—the distance d and the normal $\mathbf{u}$—then the registration error between corresponding road pixels in the right and left images should correspond to a local minimum. In our case, the measurement $\mathbf{z}_t$ is given by the current stereo pair. The registration error is simply the sum of squared differences between the right image and the corresponding left image computed over a given region of interest within the right image. The registration error is given by:

$$e(\mathbf{b}) = \frac{1}{N_p} \sum_{(x_r,y_r)\in ROI} \left(I_{r(x_r,y_r)} - I_{l(h_1x_r+h_2y_r+h_3,y_r)}\right)^2 \qquad (17)$$

where N_p is the number of pixels contained in the region of interest. The above summation is carried out over the right region of interest. The corresponding left pixels are computed according to the linear transform given by (1) and (2). The computed $x_l = h_1x_r + h_2y_r + h_3$ is a non-integer value. Therefore, the grey-level, $I_l(x_l)$, is set to a linear interpolation of the grey-level of two neighboring pixels—the ones whose horizontal coordinates bracket the value x_l.

Note that the region of interest is a user-defined region. Roughly speaking, it should correspond to the road image. Ideally, this region should not include non-road objects but as we will see in the experiments this is not a hard constraint. In our experiments, the ROI is set to a rectangular window that roughly covers the third bottom of the image. Fig. 5 illustrates a typical region of interest.

The observation likelihood is given by

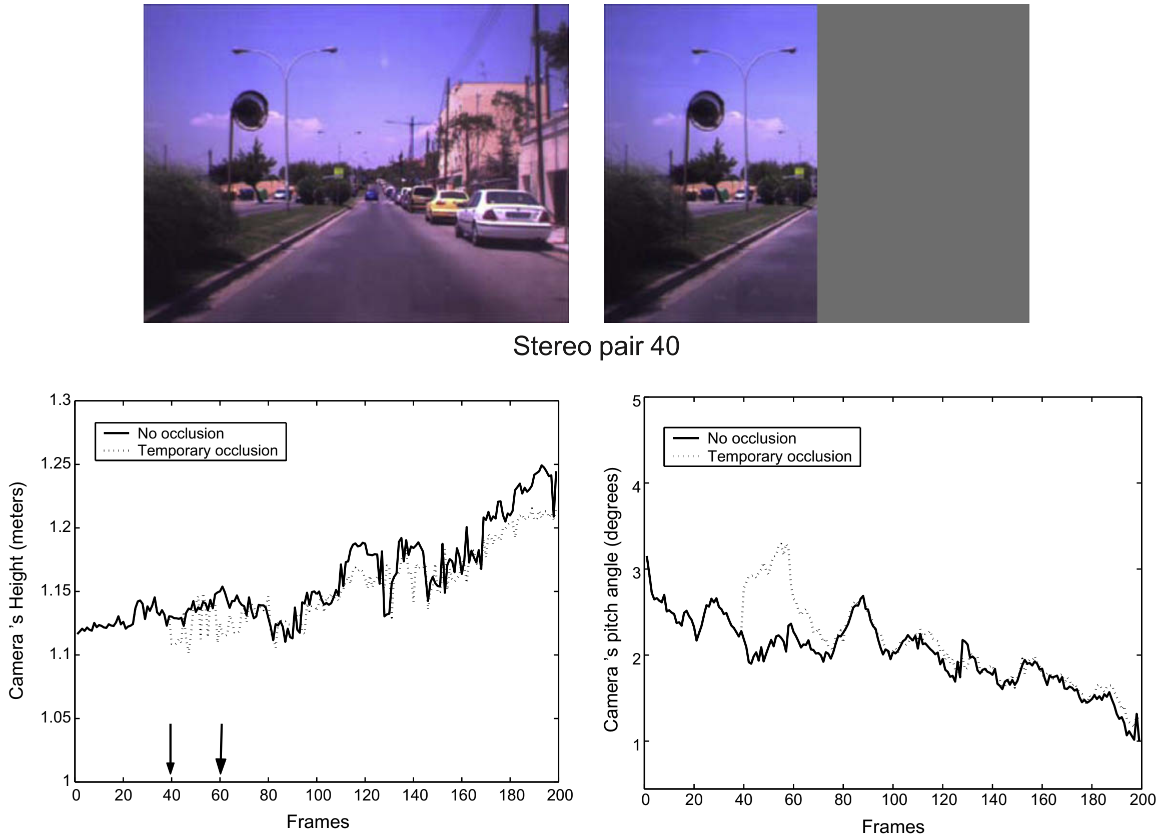

Fig. 9. Comparing the pose parameters when a significant occlusion occurs. The solid curves are obtained with the non-occluded images associated with the above sequence. The dotted curves are obtained when 20 frames of right images of the same sequence are artificially occluded. The occlusion is simulated by setting the vertical half of the right images to a fixed color. This occlusion starts at frame 40 and ends at frame 60.

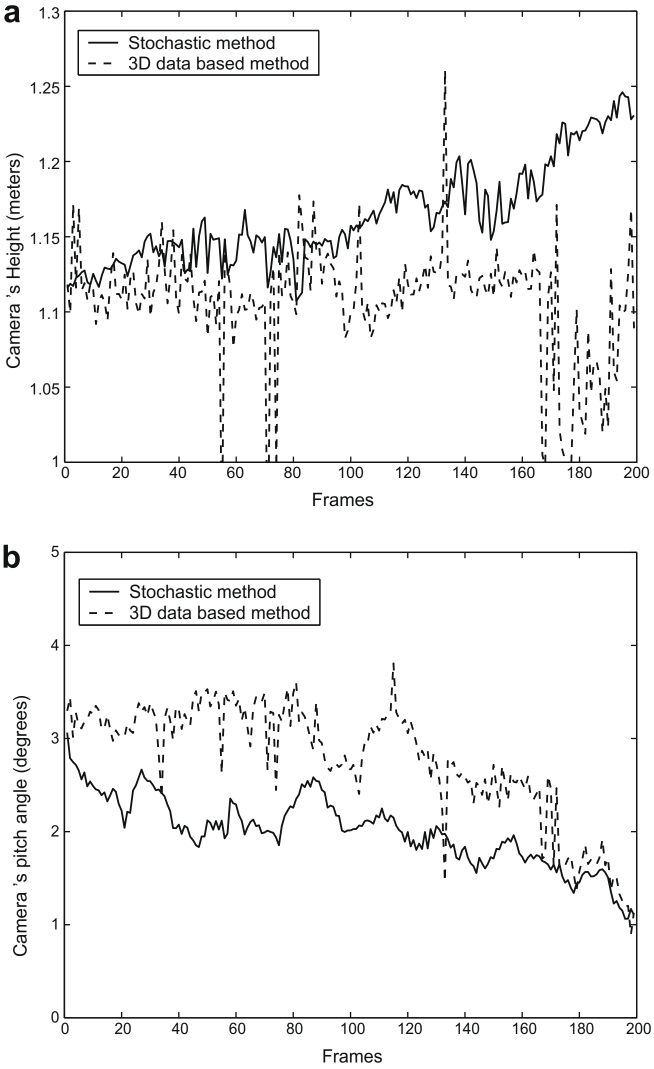

Fig. 10. Comparing two methods for estimating the camera's pose parameters. The solid curves correspond to the developed stochastic approach. The dashed curves correspond to the 3D data-based approach obtained with full resolution images, i.e., 640 × 480. One can see that the stochastic method is providing a consistent solution.

$$p(\mathbf{z}_t|\mathbf{b}_t) = \frac{1}{\sqrt{2\pi}\sigma_e} \exp\left(-\frac{e(\mathbf{b}_t)}{2\sigma_e^2}\right) \tag{18}$$

where σ_e is a parameter controlling the aperture of the Gaussian function.

Computing the state vector $\mathbf{b}_t$ from the previous posterior distribution $p(\mathbf{b}_{t-1}|\mathbf{z}_{1:t-1})$ is carried out using the particle filtering framework. This is described in Fig. 6. The state vector is given by $\mathbf{b} = (b_x, b_y, b_z)^T = (\frac{u_x}{d}, \frac{u_y}{d}, \frac{u_z}{d})^T$.

4.2.3. Initialization

Note that the initial distribution $p(\mathbf{b}_0)$ can be either a Dirac or Gaussian distribution centered on a solution provided by the 3D data-based algorithm or manually specified. Alternatively, this solution can be obtained using a differential evolution algorithm [31].

5. Experiments

The proposed technique has been tested on different urban environments. In this section, we will provide results obtained with three different videos associated with different road structures. Moreover, we provide a performance study using synthetic videos with ground-truth data.

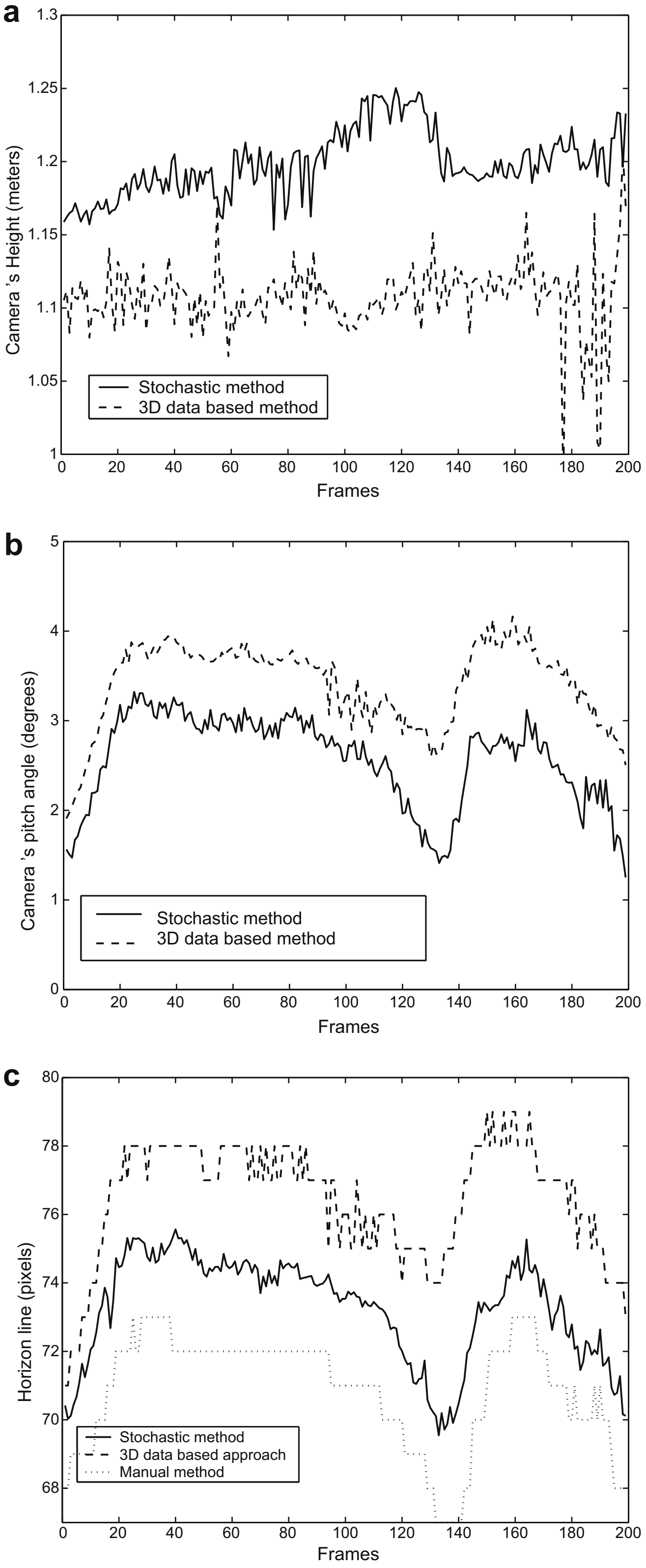

Fig. 11. Comparing two methods for estimating the camera's pose parameters. The solid curves correspond to the developed stochastic approach. The dashed curves correspond to the 3D data-based approach obtained with full resolution images, i.e., 640 × 480. The bottom plot displays the manually computed horizon line—the dotted curve.

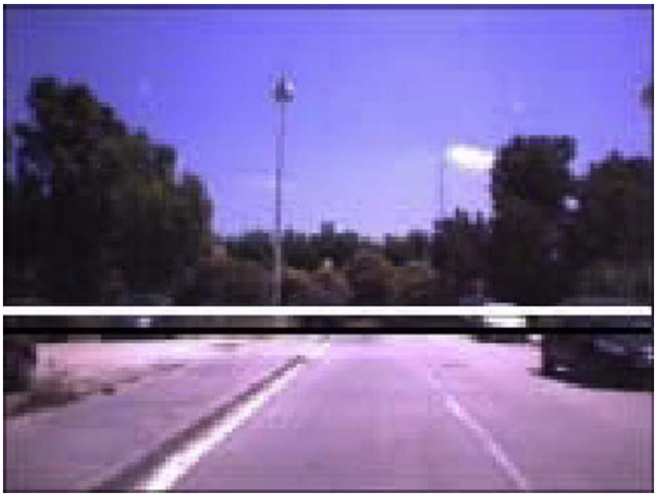

Fig. 12. The estimated horizon line associated with frame 160. The white line is obtained with the proposed method and the black one is obtained with the 3D data-based approach.

5.1. First experiment

The first experiment has been conducted on a short sequence of stereo pairs corresponding to a typical urban environment (see Fig. 5). The stereo pairs are of resolution 320 × 240. Here, the road is almost flat and the changes in the pose parameters are mainly due to the car's accelerations and decelerations. Fig. 7(a) and (b) depict the estimated camera's height and orientation as a function of the sequence frames, respectively. The plotted solutions correspond to the maximum a posteriori solution. The solid curves correspond to an arbitrary ROI of size 270 × 80 pixels centered at the bottom of the image. The dotted curves correspond to a ROI covering the road region only (here the ROI is manually set to 200 × 80 pixels centered at the bottom of the image). The arbitrary ROI includes some objects that do not belong to the road plane—the vehicles on the right bound. As can be seen, the estimated solutions associated with the two cases are almost the same, which suggests that the obstacles will not have a significant impact on the solution. In this experiment the number of particles N was set to 200 and the parameters were as follows $\sigma = 0.002$ and $\sigma_e = 1$.

In the literature, the pose parameters—plane parameters—can be used to compute the horizon line. In our case, since the roll angle is very small, the horizon line can be represented by an horizontal line in the image. Once the 3D plane parameters d and $\mathbf{u} = (u_x, u_y, u_z)^T$ are computed, the vertical position of the horizon line will be given by

$$v_h = v_0 + \frac{\alpha d}{u_y Z_\infty} - \frac{\alpha u_z}{u_y} \qquad (19)$$

The above formula is derived by projecting a 3D point $(0, Y_p, Z_\infty)$ belonging to the road plane and then taking the vertical coordinate $v = \alpha \frac{Y_p}{Z_\infty} + v_0$. Z_∞ is a large depth value. In our experiments, Z_∞ is set to 6000 m. Fig. 7(c) depicts the vertical position of the horizon line as a function of the sequence frames. Fig. 8 illustrates the computed horizon line for frames 55 and 182.

In order to study the algorithm behavior in presence of significant occlusions, we conducted the following experiment. We used the same sequence of Fig. 5. We run the proposed technique described in Section 4 twice. In the first run the stereo images were used as they are. In the second run, the right images were modified to simulate a significant occlusion. To this end we set the vertical half of a set of 20 right images to a fixed color. The left images were not modified. Fig. 9 compares the pose parameters obtained in the two runs. The solid curves were obtained with the non-occluded images. The dotted curves were obtained when the right images of the same sequence are artificially occluded. This occlusion starts at frame 40 and ends at frame 60. The top of the figure illustrates the stereo pair 40. As can be seen, the discrepancy that occurs at the occlusion is considerably reduced when the occlusion disappears.

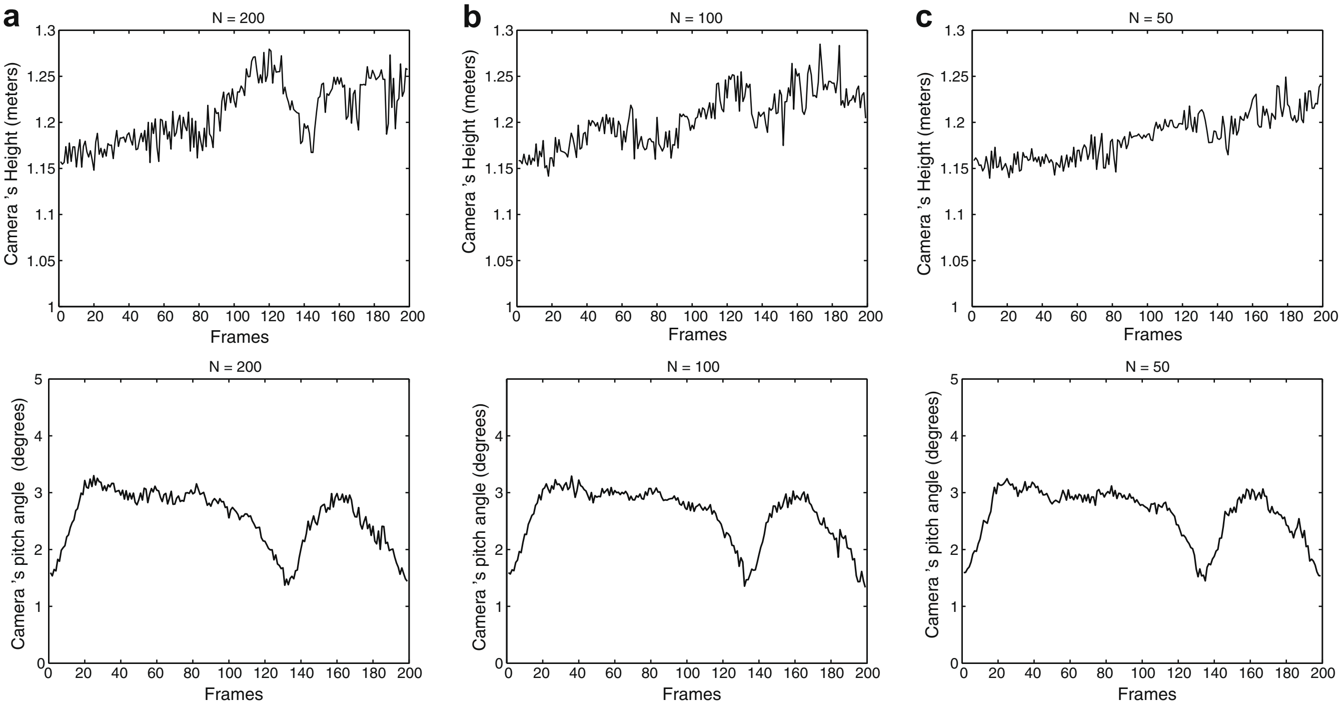

Fig. 13. The estimated camera's position (first row) and pitch angle (second row) obtained with the proposed stochastic approach for different numbers of particles. (a–c) Correspond to $N = 200, 100$, and 50, respectively. As can be seen the estimated parameters are very consistent and are not highly affected by the choice of the particle set size.

Fig. 14. A frame from the third video sequence.

Fig. 10(a) and (b) depict the estimated camera's height and orientation as a function of the sequence frames using two different methods. The solid curves correspond to the developed stochastic approach and the dashed curves to the 3D data-based approach outlined in Section 3. This method has used full resolution stereo images, i.e., 640 × 480. One can see that despite some discrepancies the stochastic method is providing the same behavior of changes. Moreover, the variations obtained with the proposed technique are smoother than the ones obtained with the 3D data-based method.

5.2. Second experiment

The second experiment has been conducted on another short sequence corresponding to an uphill driving. The stereo pairs are of resolution 160 × 120. Fig. 11(a) and (b) depict the estimated camera's height and orientation as a function of the sequence frames, respectively. The solid curves correspond to the developed stochastic approach. The dashed curves correspond to the 3D data-based approach obtained with full resolution images, i.e., 640 × 480. Fig. 11(c) depicts the estimated position of the horizon line as estimated by three methods: the above two methods (solid and dashed curves) and a manual method (dotted curve) based on the intersection of two parallel lines. As can be seen, the horizon line estimated by the proposed featureless approach is closer to the manually estimated horizon line—assumed to be very close to the ground-truth data. This suggests that the estimated horizon line is more accurate than the one estimated by the 3D based approach. Since the horizon line mainly depends on the plane orientation, it follows that the orientation estimated by the proposed approach is more accurate than the one estimated by the 3D based approach. This can be explained by the fact that the latter approach is based on an explicit 3D reconstruction followed by a 3D plane extraction that can exclude many measurements related to the road. Note that the 4 pixel discrepancy between the solid curve (proposed method) and the dashed curve (3D data-based method) corresponds to 16 pixel discrepancy in the original image (the processed images are sub-sampled).

Fig. 12 displays the estimated horizon line for frame 160. The white line is obtained with the proposed method and the black one is obtained with the 3D data-based approach. A short video showing the computed horizon line by the two automatic methods can be found at www.cvc.uab.es/~sappa/UphillDriving.avi. Once again, the white line was computed by the proposed technique and the black one was computed by the 3D based technique.

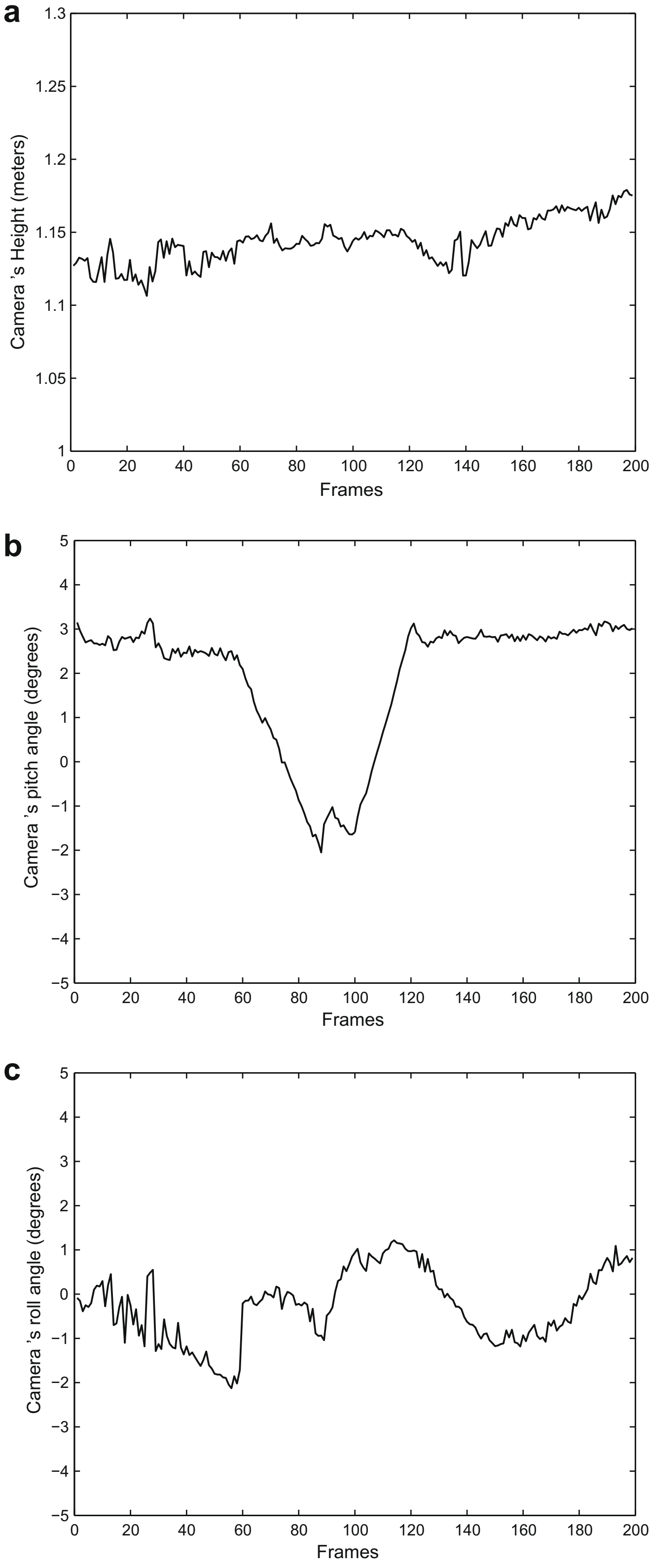

Fig. 15. The camera height (a), pitch angle (b), and roll angle (c) associated with the third experiment. In this experiment, the vehicle performs a turning manoeuvre from frame 50 to 140. As can be seen, the pitch angle variation is somewhat significant.

Fig. 13 displays the estimated camera's height and pitch angle associated with the sequence of Fig. 12 when the number of particles is set to 200, 100, and 50. One can notice that when the num-

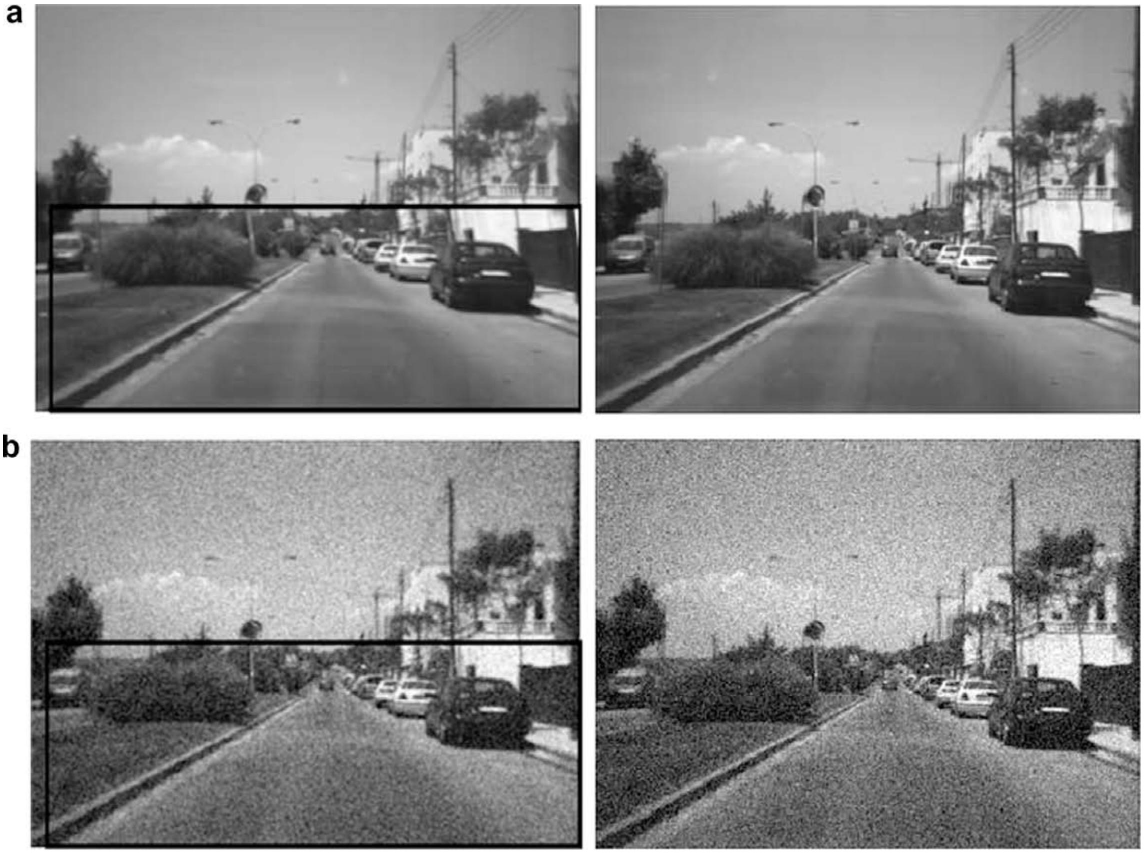

Fig. 16. (a) Depicts a noise-free synthesized stereo pair where the left image (bounded box) is a warped version of the right image using the road plane ground-truth parameters. (b) Depicts the same stereo pair after a Gaussian noise is added to both images. The standard deviation of the noise was 20.

ber of particles is very small, i.e. 50, the estimated camera height is slightly affected but there is no impact on the estimated pitch angle.

5.3. Third experiment

Fig. 15 presents the computed camera height (a), pitch angle (b), and roll angle (c) associated with another video sequence (see Fig. 14). In this experiment, the vehicle performs a turning manoeuvre from frame 50–140. As can be seen, the pitch angle variation is somewhat significant. The car speed was not high. It was about 30 km/h.

A non-optimized C code processes one stereo pair in 30 ms assuming the size of the ROI is 6000 pixels and the number of particles is 100. Therefore, the proposed approach runs almost 12 times faster than the 3D data-based approach (Section 3).

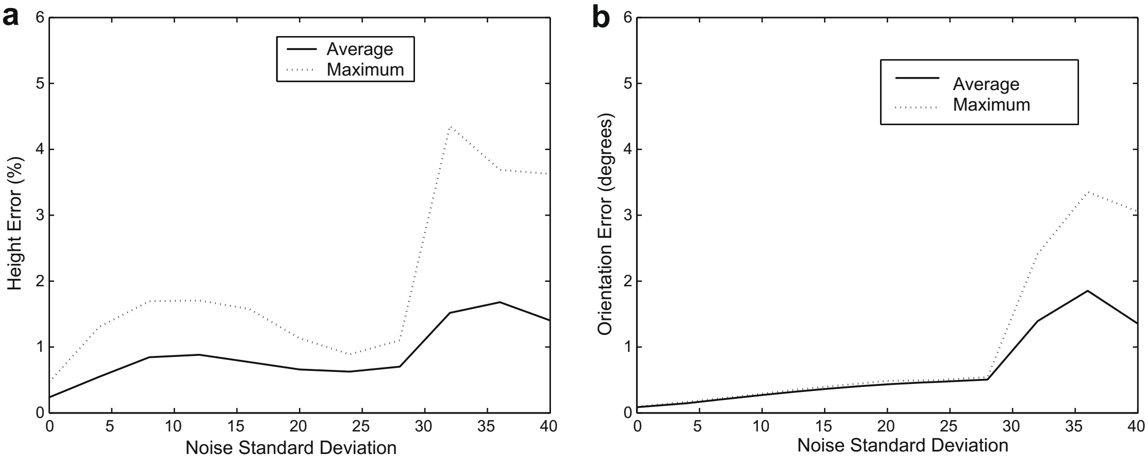

Fig. 17. The errors associated with the plane parameters as a function of the noise standard deviation using synthesized video sequences. (a) Depicts the height errors. (b) Depicts the plane orientation errors. Each point of the curves—each noise level—corresponds to 5000 stereo pairs corresponding to 25 realizations each of which is a sequence of 200 perturbed stereo pairs. The solid curves correspond to the average of errors over the 5000 stereo pairs and the dashed curves correspond to the maximum average over the 25 realizations.

6. Performance study

So far in this paper the evaluation of the proposed method has been carried out on real video sequences, including a comparison with a 3D data-based approach. However, it is very challenging to get ground-truth data for the on-board camera pose. In this section, we propose a simple scheme that can provide the ground-truth data for the road parameters. To this end, we use a 200-frame-long real video captured by the on-board stereo camera. For each stereo pair, we can fix the distance and the plane normal. Those ones can be fixed for the whole sequence or can vary according to a predefined model. Then each left image in the original sequence is synthesized by warping the corresponding right image using the image transfer function encapsulating the road parameters. Then the obtained stereo pairs will be perturbed by adding Gaussian noise to the grey levels of all pixels.

Fig. 16(a) shows a typical synthetic stereo pair where the right image is used as it is in the original sequence. However, the left image—the sub-image bounded by the black box—is synthesized by warping the right image using the 3D road parameters. In this image, the synthesized box is superimposed with the original left image and one can see easily that the warped non-road objects are not aligned with their original image—the planar parallax. Fig. 16(b) depicts the same stereo pair when the grey levels have been perturbed by an additive Gaussian noise whose standard deviation is set to 20. Here the grey-level of the images has 256 values. The proposed approach is then invoked to estimate the road parameters from the noisy stereo pairs. The performance can be directly evaluated by comparing the estimated parameters with the ground-truth parameters. Since there are two kinds of parameters: (i) the camera height (plane distance), and (ii) the plane normal, there will be two errors: the distance error and the orientation error. The former one is simply the absolute value of the relative error. The latter one is defined by the angle between the direction of the ground-truth normal and the direction of the estimated one.

Fig. 17 summarizes the obtained errors associated with the synthetic stereo pairs. Fig. 17(a) depicts the distance error and Fig. 17(b) the orientation error. Here one percent error corresponds to 1.13 cm. Each point of the curves—each noise level—corresponds to 5000 stereo pairs corresponding to 25 realizations each of which is a sequence of 200 perturbed stereo pairs. The solid curves correspond to the global average of errors over the 5000 stereo pairs and the dashed curves correspond to the maximum average over the 25 realizations.

Fig. 18 shows the obtained error corresponding to one synthesized 200-frame stereo sequence. The standard deviation of the added noise is 32. Fig. 18(a) depicts the height error. Fig. 18(b) depicts the plane orientation error.

7. Discussion and future work

A featureless and stochastic technique for real-time estimation of on-board stereo head pose has been presented. The method adopts a particle filtering scheme that uses images' brightness in its observation likelihood. The advantages of the proposed technique are as follows. First, the technique does not need any feature extraction neither in the image domain nor in 3D space. Second, the technique inherits the strengths of stochastic tracking approaches. A good performance has been shown in several scenarios—uphill, downhill and flat roads. Furthermore, the technique can handle significant occlusions. Although it has been tested on urban environments, it could be also useful on highways scenarios. Experiments on real and synthetic stereo sequences have shown that the accuracy of the orientation is better than the height accuracy, which is consistent with 3D pose algorithms.

We point out that even in cases where the particle filtering has not provided the correct parameters for a few frames (a fraction of a second), the particle filtering method is able to estimate the correct parameters for the subsequent frames. The aim of the particle filter is to keep a lock on the tracked parameters despite possible inaccuracies affecting some frames.

In the current study, we assumed that the ROI contained a significant part of the road. Moreover, we have shown that non-road objects such as cars and occluding objects do not significantly bias the estimated parameters. The proposed approach can be nicely linked to a road segmentation algorithm. Our laboratory is developing an illuminant invariant road segmentation based on an online non-parametric road model (see [2]).

We believe that the size of the road in the stereo pair has not a major impact on the accuracy of the plane parameter estimation. Indeed, there are three degrees of freedom that are estimated through image registration of a ROI having thousands of pixels in both images.

Future work will investigate the use of a particle filtering scheme adopting mixed states. Within this scheme, we wish to incorporate more accurate dynamic models for common driving schemes such as car's acceleration, deceleration, and constant velocity. Moreover, the use of multiple nested Regions of Interest will be one research direction.

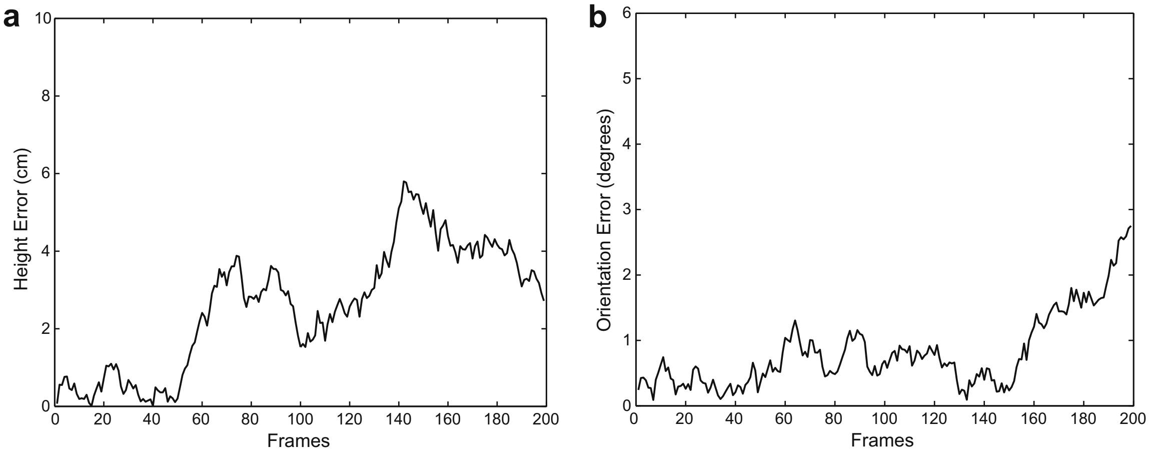

Fig. 18. The error corresponding to one synthesized 200-frame stereo sequence. The standard deviation of the added noise is 32. (a) Depicts the height error. (b) Depicts the plane orientation error.

Acknowledgments

This work was partially supported by the Government of Spain under MEC project TRA2007-62526/AUT and research programme Consolider Ingenio 2010: MIPRCV (CSD2007-00018).

References

[1] K.R.T. Aires, H. Araujo, A.A.D. Medeiros, Plane detection from monocular image sequences, in: Alvey Vision Conference, 2008.

[2] J.M. Alvarez, A. López, R. Baldrich, Illuminant-invariant model-based road segmentation, in: IEEE Intelligent Vehicles Symposium, 2008.

[3] S. Arulampalam, S.R. Maskell, N.J. Gordon, T. Clapp, A tutorial on particle filters for on-line nonlinear/non-Gaussian Bayesian tracking, IEEE Transactions on Signal Processing 50 (2) (2002).

[4] M. Bertozzi, E. Binelli, A. Broggi, M. Del Rose, Stereo vision-based approaches for pedestrian detection, In: Proceedings of the Computer Vision and Pattern Recognition, San Diego, USA, June 2005.

[5] M. Bertozzi, A. Broggi, GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection, IEEE Transactions on Image Processing 7 (1) (1998) 62–81.

[6] M. Bertozzi, A. Broggi, R. Chapuis, F. Chausse, A. Fascioli, A. Tibaldi, Shape-based pedestrian detection and localization, in: Proceedings of the IEEE International Conference on Intelligent Transportation Systems, Shangai, China, October 2003, pp. 328–333.

[7] M. Bertozzi, A. Broggi, A. Fascioli, T. Graf, M. Meinecke, Pedestrian detection for driver assistance using multiresolution infrared vision, IEEE Transactions on Vehicular Technology 53 (6) (2004) 1666–1678.

[8] A. Blake, M. Isard, Active Contours, Springer-Verlag, 2000.

[9] B. Boufama, D.J. Oconnell, Identification and matching of planes in a pair of uncalibrated images, International Journal of Pattern Recognition Artificial Intelligence 17 (7) (2003) 1127–1143.

[10] A. Broggi, M. Bertozzi, A. Fascioli, M. Sechi, Shape-based pedestrian detection, in: Proceedings of the IEEE Intelligent Vehicles Symposium, Dearborn, USA, October 2000, pp. 215–220.

[11] A. Broggi, A. Fascioli, I. Fedriga, A. Tibaldi, M. Del Rose, Stereo-based preprocessing for human shape localization in unstructured environments, in: Proceedings of the IEEE Intelligent Vehicles Symposium, Columbus, OH, USA, June 2003, pp. 410–415.

[12] J.M. Collado, C. Hilario, A. Escalera, J.M. Armingol, Adaptive road lanes detection and classification, in: Advanced Concepts for Intelligent Vision Systems, 2006, pp. 1151–1162.

[13] P. Coulombeau, C. Laurgeau, Vehicle yaw, pitch, roll and 3D lane shape recovery by vision, in: Proceedings of the IEEE Intelligent Vehicles Symposium, Versailles, France, June 2002, pp. 619–625.

[14] A. Doucet, N. Freitas, N. Gordon, Sequential Monte Carlo Methods in Practice, Springer-Verlag, New York, 2001.

[15] O. Faugeras, Three-Dimensional Computer Vision: a Geometric Viewpoint, The MIT Press, 1993.

[16] O. Faugeras, Q.T. Luong, The Geometry of Multiple Images, The MIT Press, 2001.

[17] D.M. Gavrilla, S. Munder, Multi-cue pedestrian detection and tracking from a moving vehicle, International Journal on Computer Vision 73 (1) (2007) 41–59.

[18] N. Hautière, R. Labayrade, D. Aubert, Real-time disparity contrast combination for onboard estimation of the visibility distance, IEEE Transactions on Intelligent Transportation Systems 7 (2) (2006) 201–212.

[19] Z. Hu, K. Uchimura, U-V-Disparity: an efficient algorithm for stereovision based scene analysis, in: Proceedings of the IEEE Intelligent Vehicles Symposium, Las Vegas, USA, June 2005, pp. 48–54.

[20] R. Labayrade, D. Aubert, In-vehicle obstacles detection and characterization by stereovision, in: Proceedings of the 1st International Workshop In-Vehicle Cognitive Computer Vision System, Graz, Austria, April 2003, pp. 13–19.

[21] R. Labayrade, D. Aubert, A single framework for vehicle roll, pitch, yaw estimation and obstacles detection by stereovision, in: Proceedings of the IEEE Intelligent Vehicles Symposium, Columbus, OH, USA, June 2003, pp. 31–36.

[22] R. Labayrade, D. Aubert, J. Tarel, Real time obstacle detection in stereovision on non flat road geometry through "V-disparity" representation, in: Proceedings of the IEEE Intelligent Vehicles Symposium, Versailles, France, June 2002, pp. 646–651.

[23] D. Lefée, S. Mousset, A. Bensrhair, M. Bertozzi, Cooperation of passive vision systems in detection and tracking of pedestrians, in: Proceedings of the IEEE Intelligent Vehicles Symposium, Parma, Italy, June 2004, pp. 768–773.

[24] T. Lertrusdachakul, T. Aoki, H. Yasuda, Camera motion estimation by image feature analysis, in: International Conference on Advances in Pattern Recognition, 2005.

[25] Y. Liang, H. Tyan, H. Liao, S. Chen, Stabilizing image sequences taken by the camcorder mounted on a moving vehicle, in: Proceedings of the IEEE International Conference on Intelligent Transportation Systems, Shangai, China, October 2003, pp. 90–95.

[26] X. Liu, K. Fujimura, Pedestrian detection using stereo night vision, IEEE Transactions on Vehicular Technology 53 (6) (2004) 1657–1665.

[27] B. Micusik, H. Wildenauer, M. Vincze, Towards detection of orthogonal planes in monocular images of indoor environments, in: IEEE International Conference on Robotics and Automation, 2008.

[28] D. Ponsa, A. López, F. Lumbreras, J. Serrat, T. Graf, 3D vehicle sensor based on monocular vision, in: Proceedings of the IEEE International Conference on Intelligent Transportation Systems, Vienna, Austria, Sepember 2005, pp. 1096–1101.

[29] A. Sappa, F. Dornaika, D. Ponsa, D. Gerónimo, A. López, An efficient approach to on-board stereo vision system pose estimation, IEEE Transactions on Intelligent Transportation Systems 9 (3) (2008) 475–490.

[30] G. Stein, O. Mano, A. Shashua, A robust method for computing vehicle ego-motion, in: IEEE Intelligent Vehicles Symposium, 2000.

[31] R. Storn, K. Price, Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces, Journal of Global Optimization 11 (1997) 341–359.

[32] T. Suzuki, T. Kanade, Measurement of vehicle motion and orientation using optical flow, in: IEEE Intelligent Vehicles Symposium, 1999.

[33] P. Viola, M.J. Jones, D. Snow, Detecting pedestrians using patterns of motion and appearance, International Journal of Computer Vision 63 (2) (2005) 153–161.

[34] T. Zhang, C. Tomasi, Fast, robust, and consistent camera motion estimation, in: IEEE Conference on Computer Vision and Pattern Recognition, 1999.