



# Multimodal inverse perspective mapping



Miguel Oliveira<sup>a,\*</sup>, Vitor Santos<sup>b,a</sup>, Angel D. Sappa<sup>c</sup>

<sup>a</sup> Institute of Electronics and Telematics Engineering of Aveiro, Campus Universitario de Santiago, 3800 Aveiro, Portugal

<sup>b</sup> Department of Mechanical Engineering, University of Aveiro, Campus Universitario de Santiago, 3800 Aveiro, Portugal

<sup>c</sup> Computer Vision Center, Campus UAB, 08193 Bellaterra, Barcelona, Spain

## ARTICLE INFO

### Article history:

Received 27 August 2013

Received in revised form 17 February 2014

Accepted 9 September 2014

Available online 22 September 2014

### Keywords:

Inverse perspective mapping

Multimodal sensor fusion

Intelligent vehicles

## ABSTRACT

Over the past years, inverse perspective mapping has been successfully applied to several problems in the field of Intelligent Transportation Systems. In brief, the method consists of mapping images to a new coordinate system where perspective effects are removed. The removal of perspective associated effects facilitates road and obstacle detection and also assists in free space estimation. There is, however, a significant limitation in the inverse perspective mapping: the presence of obstacles on the road disrupts the effectiveness of the mapping. The current paper proposes a robust solution based on the use of multimodal sensor fusion. Data from a laser range finder is fused with images from the cameras, so that the mapping is not computed in the regions where obstacles are present. As shown in the results, this considerably improves the effectiveness of the algorithm and reduces computation time when compared with the classical inverse perspective mapping. Furthermore, the proposed approach is also able to cope with several cameras with different lenses or image resolutions, as well as dynamic viewpoints.

© 2014 Elsevier B.V. All rights reserved.

## 1. Introduction

Intelligent Transportation Systems offers one of the most relevant frameworks for data fusion [1]. Within this scope, Advanced Driver Assistance Systems (ADAS) are considered to have paramount importance, since they have a significant impact in the safety of both passengers and pedestrians alike [2].

One very important aspect of ADAS is road detection or lane marker detection techniques [3], i.e., the automatic estimation by the vehicle of its position with respect to the road. Robust road position awareness is one of the primary features that an intelligent vehicle should present [4].

This paper focuses on a particular technique called Inverse Perspective Mapping (IPM) which is very often used in vision-based road estimation algorithms as a pre-processing component. IPM uses information from the camera's position and orientation towards the road to produce a bird's eye view image where perspective effects are removed. The correction of perspective allows much more efficient and robust road detection, lane marker tracking, or pattern recognition algorithms to be implemented. In reality, IPM has been employed not only with the purpose of detecting the vehicle's position with respect to the road, but also

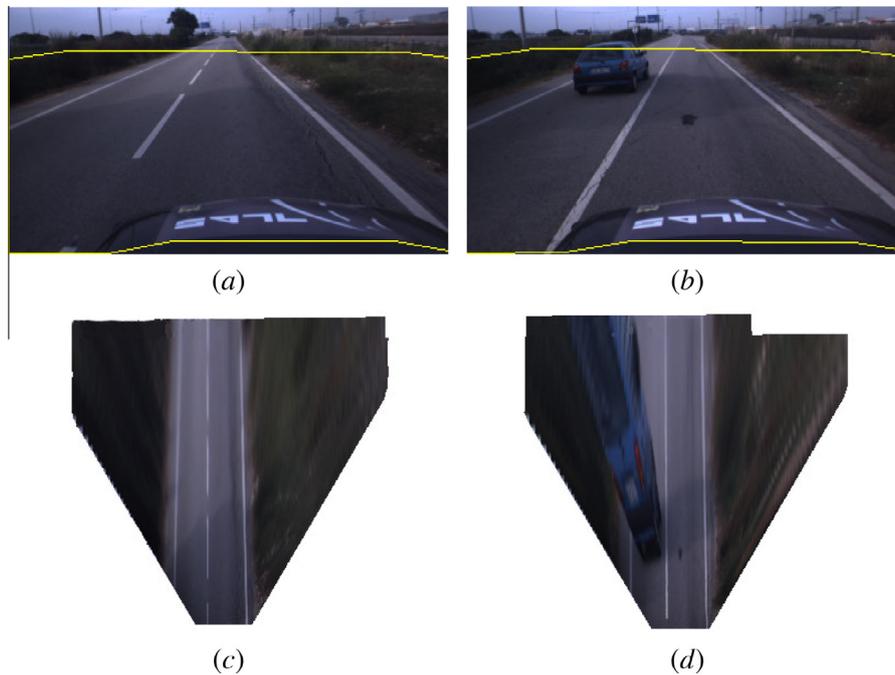
in many other ADAS related applications, e.g., obstacle detection [5,6], free space estimation [7], pedestrian detection [8] or ego motion estimation [9].

Therefore, IPM is of paramount importance to a large number of automated tasks that should be handled by an intelligent vehicle. If IPM provides input to many other algorithms, then special care should be given to it. In this paper, we focus on the robustness of the IPM algorithm, since it still presents some limitations when applied in the context of on-board road mapping. IPM works under three core assumptions: the road must be a flat surface, there should be a rigid body transformation from the camera to the road, and the road should be free of obstacles. We focus on the last assumption: obstacle free road. This is the least realistic assumption of all, since very often the roads are populated by other vehicles, protection walls, pedestrians, etc. In fact, the classical IPM algorithm fails to produce an accurate projected image when the input images from the cameras contain other things other than the road itself. An example of this is shown in Fig. 1: in (a) the road is free of obstacles, which leads to an IPM image (c) which is very accurate. In Fig. 1(b), there is another vehicle on the road. As a consequence, the resulting IPM image (d) contains artifacts which might be misleading for example for pattern recognition algorithms.

In this paper, we propose an extension to the classical IPM that can still compute accurate IPM images when obstacles are in front of the vehicle. To accomplish this, we make use of an additional

\* Corresponding author.

E-mail addresses: [mriem@ua.pt](mailto:mriem@ua.pt) (M. Oliveira), [vitor@ua.pt](mailto:vitor@ua.pt) (V. Santos), [asappa@cvc.uab.es](mailto:asappa@cvc.uab.es) (A.D. Sappa).



**Fig. 1.** Two input images (a) and (b) and their corresponding IPM projected images, respectively (c) and (d). In (b), the presence of a blue color vehicle on the road causes the IPM image to present blue artifacts (d). The yellow lines show the areas of the images that are used for IPM projection. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

laser range finder (LRF). These sensors are especially designed for obstacle detection purposes. In fact, nowadays, some-of-the-shelf vehicles already have these sensors installed to assist collision avoidance or collision mitigation systems.<sup>1</sup> In comparison with alternative on-board distance measuring devices, such as stereo vision or radar, LRFs produce more accurate data at higher output frequencies. They are also robust, since they work in challenging conditions such as fog, shadows or even at night. The core idea behind the proposed approach is to fuse the laser data with the pixels in the image in an attempt to identify which pixels should be used as input for IPM. As will be shown, this multimodal sensor fusion framework is capable of producing much more accurate IPM images when compared to the classical IPM approach.

In addition to this, we will show that the proposed approach is faster to compute than the classical IPM. The reason for this is that our approach employs a faster direct projection mechanism (from 3D world to image, i.e., 3D points to pixels) to discover mappable pixels, before operating the slower IPM based inverse projection (from pixels to 3D points). Classical IPM approaches make no considerations on this topic (e.g., [10–12]).

The paper is organized as follows. First, Section 2 presents the state of the art on IPM techniques, then Section 3 presents the mathematical formulation of the problem to be used in Section 4; the new reference system, proposed in the current work, is introduced in Section 5. Finally, Section 6 describes the proposed multimodal inverse perspective mapping through the computation of the mappable pixels. Experimental results are given in Section 7, and conclusions are presented in Section 8.

## 2. Related work

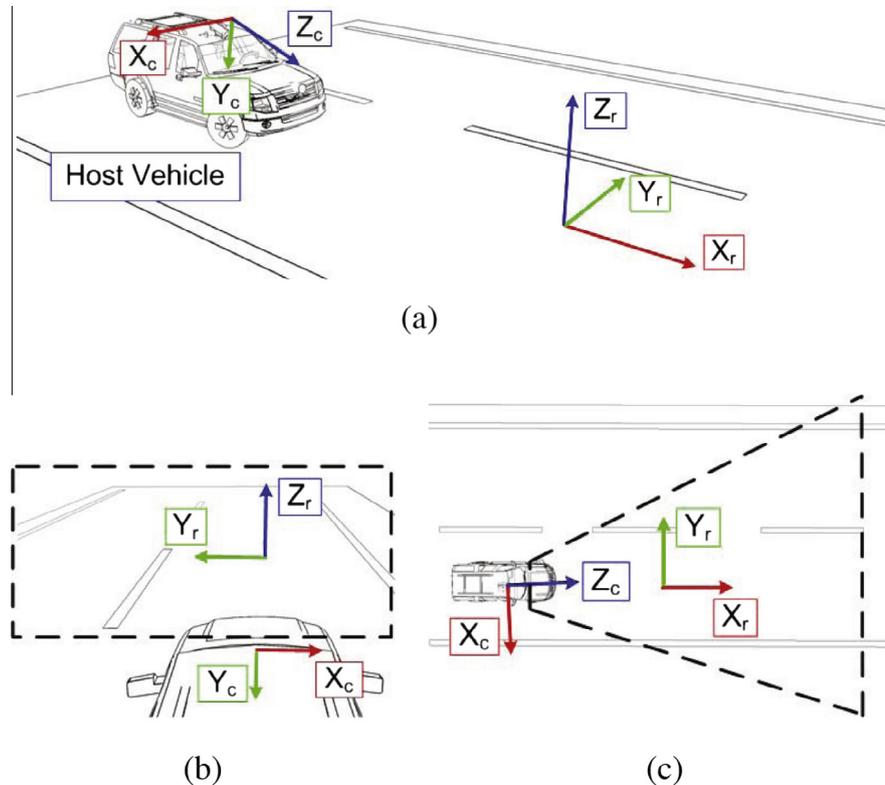
Over the last decades, IPM has been successfully applied to several problems, especially in the field of Intelligent Transportation Systems. Although it was some years ago that

authors began to mention the advantages of IPM (e.g., [13,14]), several recent publications (e.g., [15–17]) show that this is still a topic of interest to the robotics, computer vision and Intelligent Transportation Systems communities. The core application of IPM is the determination of the vehicle’s position with respect to the road, commonly referred to as “road detection” or “lane marker detection”. There are several examples of using IPM for assisting road detection in the literature (e.g., [18,12,19,20]). The usage of IPM on-board a vehicle may also aid other automatic detection systems such as generic obstacle detection [21,22], free space estimation [23,7], pedestrian detection [8,24], or optical flow computation [25].

The IPM method receives as input the image from the camera, the 6D position of the camera with respect to the road reference system (i.e., extrinsic parameters), and a description of the properties of the lens (i.e., intrinsic parameters). Under the assumption that the road ahead of the vehicle is flat, that there is a fixed rigid body transformation from the camera to the road’s reference frame, and that there are no obstacles present, the input image pixels are mapped to the road reference system, and a new image is produced where perspective effects are removed. The image that is produced by the IPM will, henceforward, be named simply IPM image. Considering on-board road detection setups, cameras are usually mounted somewhere close to the rear view mirror inside the vehicle, facing the road in front of it. The camera’s position and orientation induces perspective associated effects to the captured road images. The IPM technique consists of transforming the images by mapping the pixels to a new reference frame where the perspective effect is corrected. This reference frame is usually defined on the road plane, so that the resulting image becomes a top view of the road. Fig. 2(a) shows an example of a road scene; Fig. 2(b) depicts the input image captured by the camera; and Fig. 2(c) represents the image produced using IPM.

One of the advantages of IPM is that the subsequent perception algorithms can be computed in the IPM resulting image, which is defined in a new reference system where the geometric properties of road painted patterns are independent from the perspective of

<sup>1</sup> [http://en.wikipedia.org/wiki/Collision\\_avoidance\\_system](http://en.wikipedia.org/wiki/Collision_avoidance_system), last accessed February 2014.



**Fig. 2.** (a) A typical road scene with a camera mounted on the host vehicle facing the road. The camera reference system is labelled  $X_c, Y_c, Z_c$  and the road reference system is labelled  $X_r, Y_r, Z_r$ . (b) An example of an image captured by the camera. This image is used as input to IPM. (c) The output image of IPM. Since the road is viewed from above no perspective distortion is present.

the camera, i.e., from the position of the camera. In [14], the authors claim that the parallelization of road features is crucial for curvature determination. Another advantage is that, since the perspective effect associates different meanings to different image pixels, depending on their position in the image, after the removal of the perspective effect, each pixel represents the same portion of the road, allowing a homogeneous distribution of the information among the pixels of the resulting IPM image [22]. Other authors have also employed steerable filters for lane markings detection and sustain that filtering on the IPM image allowing a single kernel size to be used over the entire area of interest [18]. Furthermore, since images are mapped to a new reference system, several cameras may be used to produce a single IPM image mosaicking, which is a subject also present in the literature [22,20]. It should also be noted that IPM requires no explicit feature detection, which contributes to the overall robustness of the algorithm. In addition, there are also dedicated hardware systems being developed to compute the IPM images [26].

Given this, it is fair to say that IPM is a cornerstone in the development of on-board video processing systems. It assists, or is very frequently a primary step, in road modelling, obstacle and pedestrian detection, free space estimation and many other advanced drivers assistance systems.

Despite the advantages of IPM, the current state of the art on this method has some limitations. These derive mostly from the fact that the classical IPM algorithm makes three assumptions: static position of the camera with respect to the road, flat road plane, and obstacle free road. Each of these assumptions and proposed solutions are described in the following lines.

Since the position of the camera with respect to the road plane is considered static, pitch and roll variations from the host vehicle (and thus of the camera which is rigidly attached to it) are neglected. Pitch variations occur during demanding brake or

acceleration maneuvers, while roll changes are expected to appear during hard turns. When the vehicle rolls or pitches, the position of the camera with respect to the road changes. As a consequence, the accuracy of IPM decreases during these maneuvers. This problem has been identified in [27,28,19,29]. In fact, some authors claim that even a small error in the vehicle's roll/pitch estimation leads to a massive terrain classification error [30]. In [31] an algorithm is proposed that also fuses vision and laser data. However, in this case, the objective is to correct the laser range measurements, rather than to correct the projection of the image pixels, as is proposed in the current paper. In that paper, a stereo vision system is used to detect the road plane and thus estimate the position of the lasers with respect to the road. With this information, it is possible to compensate for roll or pitch variations continuously, which in turn is used for correcting raw laser scan data.

Another assumption that is generally made, is to consider the road as a flat surface. The approximation of the road surface to a plane is acceptable. Nonetheless, in some specific cases such as a road climbs, this could also be a factor for low IPM accuracy. In [32] a solution to this problem is proposed, where the "height" of the lane markings is estimated with respect to a reference plane. Using this technique, it is possible to compute IPM images in sloped roads.

The final assumption is that there are no obstacles on the road. This is often the case when other vehicles, buildings or pedestrians appear in the image. When these obstacles are present in the image, the mapping of IPM is disrupted because, in the classical IPM approach, all pixels from the input image are assumed to be on the road plane and are thus used in the projection. In real automotive applications it is unfeasible to assume an obstacle-free scenario. Nonetheless, no previous solution has been proposed. In this paper we propose a multimodal laser vision sensor fusion strategy that addresses the obstacle-free road assumption.

### 3. Problem formulation

Let  ${}^c\mathbf{R}_r$  be the classical  $3 \times 3$  rotation matrix in 3D and  ${}^c\mathbf{T}_r$  be the  $3 \times 1$  translation vector in 3D that relates two coordinate systems. Their combination maps a point in the 3D road coordinate system  $\mathbf{Q}_r = [X_r \ Y_r \ Z_r]^T$  to a point in the camera's coordinate system  $\mathbf{Q}_c = [X_c \ Y_c \ Z_c]^T$ :

$$\mathbf{Q}_c = {}^c\mathbf{R}_r \cdot \mathbf{Q}_r + {}^c\mathbf{T}_r. \quad (1)$$

Let  $\mathbf{K}$  be the intrinsic parameters matrix of a given camera, represented as:

$$\mathbf{K} = \begin{bmatrix} \alpha_x & \beta & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (2)$$

where  $\alpha_x$  and  $\alpha_y$  are the lens scaling factors in both directions,  $x_0$  and  $y_0$  the principal point coordinates in pixels and  $\beta$  the skewness factor. These parameters can be obtained by an offline calibration since they are constant for each camera-lens setup.

The projection of an arbitrary 3D point  $\mathbf{Q} = [X \ Y \ Z]^T$  to a point  $\mathbf{q}_h = [u \ v \ w]^T$  in the camera's homogeneous image coordinate system, is described as:

$$\mathbf{q}_h = \mathbf{K}({}^c\mathbf{R}_r \cdot \mathbf{Q} + {}^c\mathbf{T}_r). \quad (3)$$

Finally, the coordinates of a pixel  $\mathbf{q} = [x \ y]^T$  are obtained by adjusting the homogeneous coordinates with the scaling factor  $w$ :

$$\mathbf{q} = \frac{\mathbf{q}_h}{w}. \quad (4)$$

For simplification purposes, the current paper will use the following notation:

$$\mathbf{K} \cdot {}^c\mathbf{R}_r = \mathbf{P} = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix}, \quad (5)$$

and also:

$$\mathbf{K} \cdot {}^c\mathbf{T}_r = \mathbf{t} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}. \quad (6)$$

The above formulation may describe the projection of a point to a pixel in the image (*direct projection*), or it may be used to obtain the 3D point from the pixel coordinates (*inverse projection*). The *direct projection* ( $dp$ ) may be formulated as  $dp: \mathbb{R}^3 \rightarrow \mathbb{Z}^2$ ,  $\mathbf{Q} \rightarrow \mathbf{q}$ . In the case of inverse perspective mapping, what is sought is the 3D coordinates of a given pixel. This is the *inverse projection* ( $ip$ ), defined as  $ip: \mathbb{Z}^2 \rightarrow \mathbb{R}^3$ ,  $\mathbf{q} \rightarrow \mathbf{Q}$ .

### 4. Solutions for direct and inverse projections

The following subsections present the general form solutions for the direct and inverse projections.

#### 4.1. Direct projection

As discussed in Section 3, the direct projection aims at obtaining the pixel coordinates of a 3D world point projected to the image. Eq. (3) may then be rewritten as:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}. \quad (7)$$

Using (4), we get the definition of the direct projection  $dp$

$$\begin{cases} x = \frac{p_{11}X + p_{12}Y + p_{13}Z + t_1}{p_{31}X + p_{32}Y + p_{33}Z + t_3}, \\ y = \frac{p_{21}X + p_{22}Y + p_{23}Z + t_2}{p_{31}X + p_{32}Y + p_{33}Z + t_3}, \end{cases} \quad (8)$$

this system of equations defines the direct projection of a point in the world reference system  $\mathbf{Q} = [X \ Y \ Z]^T$  to a pixel in image coordinates  $\mathbf{q} = [x \ y]^T$ .

#### 4.2. Inverse projection

The inverse projection is the problem of obtaining the real world coordinates of a point from a pixel in the image. The problem is under-defined, since the three real world coordinates are sought from only two pixel coordinates. In IPM, the system is completed by defining the plane onto which the pixel is projected. Let an arbitrary plane, defined as:

$$\Pi: aX + bY + cZ + d = 0, \quad (9)$$

be the plane that contains the projection of the pixel. The system of equations in (3) may be extended to include the constraint of the projection plane, defined in (9):

$$w \begin{bmatrix} x \\ y \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & 0 \\ p_{21} & p_{22} & p_{23} & 0 \\ p_{31} & p_{32} & p_{33} & 0 \\ a & b & c & d \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ 0 \end{bmatrix}, \quad (10)$$

rearranging this formulation, the equations for inverse perspective mapping can be obtained. First, variable  $d$  may be moved inside the translation vector:

$$w \begin{bmatrix} x \\ y \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & 0 \\ p_{21} & p_{22} & p_{23} & 0 \\ p_{31} & p_{32} & p_{33} & 0 \\ a & b & c & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ d \end{bmatrix}, \quad (11)$$

then, (11) may be rearranged:

$$\begin{bmatrix} -t_1 \\ -t_2 \\ -t_3 \\ -d \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & 0 \\ p_{21} & p_{22} & p_{23} & 0 \\ p_{31} & p_{32} & p_{33} & 0 \\ a & b & c & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} - w \begin{bmatrix} x \\ y \\ 1 \\ 0 \end{bmatrix}, \quad (12)$$

and finally, the vector of pixel coordinates can be embedded inside the projection matrix:

$$\begin{bmatrix} -t_1 \\ -t_2 \\ -t_3 \\ -d \end{bmatrix} = \underbrace{\begin{bmatrix} p_{11} & p_{12} & p_{13} & -x \\ p_{21} & p_{22} & p_{23} & -y \\ p_{31} & p_{32} & p_{33} & -1 \\ a & b & c & 0 \end{bmatrix}}_A \begin{bmatrix} X \\ Y \\ Z \\ w \end{bmatrix}, \quad (13)$$

rearranging the system of equations results in the inverse projection ( $ip$ ) of a pixel to a known plane:

$$\begin{bmatrix} X \\ Y \\ Z \\ w \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & -x \\ p_{21} & p_{22} & p_{23} & -y \\ p_{31} & p_{32} & p_{33} & -1 \\ a & b & c & 0 \end{bmatrix}^{-1} \begin{bmatrix} -t_1 \\ -t_2 \\ -t_3 \\ -d \end{bmatrix}, \quad (14)$$

this is a valid solution whenever matrix  $A$  is invertible and not singular. In other words, the projection formulation is invalid when the projection plane and the image plane are parallel and the projection plane is behind the image plane. The term *behind* will be clarified in Section 6.1 with the introduction of the half space of projection.

## 5. Road and vehicle reference systems

In the classic IPM formulation the camera and road reference systems have a known static transformation between them. The IPM projection will transform the pixels from the camera to the road reference system. In the current paper we use an additional reference system, the vehicle reference system. The vehicle reference system is fixed to the host vehicle. It is the reference system to which all sensors on the vehicle are related. Therefore, a fixed, rigid body transform is used to represent the pose of the camera with respect to the vehicle reference system. Hence, three reference systems are used: the camera system  $\{X_c, Y_c, Z_c\}$ , the road reference system  $\{X_r, Y_r, Z_r\}$  and the vehicle reference system  $\{X_v, Y_v, Z_v\}$ . Fig. 3 shows the reference systems for the vehicle, road and camera.

The general camera to road reference systems transformation was introduced in (1). Let the rotation and translation matrices of (1) be assembled into a global transformation matrix  ${}^c\mathbf{H}_r$  in homogeneous format, so that:

$$\mathbf{Q}_c = {}^c\mathbf{H}_r \cdot \mathbf{Q}_r, \quad (15)$$

the global transformation from the camera to the road is obtained as the product of a fixed camera to vehicle transformation and a dynamic (pitch, roll, therefore time dependent) vehicle to road transformation.

$$\mathbf{Q}_c = {}^c\mathbf{H}_v \cdot {}^v\mathbf{H}_r(t) \cdot \mathbf{Q}_r. \quad (16)$$

In the general mathematical model proposed here, the classic IPM approach may still be used:  ${}^v\mathbf{H}_r(t)$  is constant for all values of  $t$ , i.e., the coefficients of (9) are defined to represent the  $X_v, Y_v$  plane  $\Pi_{road} : a_r = b_r = d_r = 0$  and  $c_r = 1$ ; or the road plane may be actually detected, if  ${}^v\mathbf{H}_r(t)$  is estimated over time using stereo or laser sensors pointed towards the road, i.e., some estimation function of the parameters in (9) is running continuously. An example of real time estimation of road to vehicle transformation is presented in [33].

## 6. Mappable versus unmappable pixels

IPM is the application of (14) to the pixels in the image. However, in a given image, not all pixels may be interesting or even possible to project. The current work addresses this problem by using a laser sensor to detect mappable regions, together with a set of criteria to select which pixels should be mapped.

In summary, the fusion mechanism we propose is the following: using several criteria, we compute a set of polygons in 3D (defined in the road's reference frame). Each polygon delimits the area of the road which, in accordance with the corresponding criteria, should be mapped using IPM. Then, we fuse all these criteria by

computing a polygon (the projection polygon) which results from the intersection of the several criteria driven polygons. The projection polygon now encodes the region of the road that should be mapped using IPM. However, the projection polygon is defined in the road's reference frame (in 3D). Hence, to use this information as input to an IPM projection, we first need to project the projection polygon onto the image plane (this is done using a direct projection mechanism), which we call the image projection polygon. The image projection polygon is the tool that allows the pixels to be labelled as mappable or not: pixels inside this polygon should be mapped and pixels outside the polygon are skipped. Note that in this approach we are fusing multimodal data, since that some of the criteria we use are related to the vision sensor and others to the LRF sensor data.

The following subsections present the different criteria used to find which pixels in an image are possible to be projected.

### 6.1. Half space of projection

Eq. (14) is the mathematical solution of the intersection of the optical ray of a given pixel with the road plane. Because of this, a pixel above the horizon line in the image will be projected to the back of the camera's plane. Fig. 4 shows the projection rays of two pixels, one is projectable and the other should be discarded.

Although the presented solution is a valid mathematical solution, for the proposed model, however, the unprojectable pixels must be handled in accordance. This is done by first computing the image plane. The image plane divides the three-dimensional Euclidean space into two parts. One of them is called half space of projection. It is defined as the region of the Euclidean space where all points contained by it may be virtually projected into the image plane. The image plane is defined as  $\Pi_{image} : a_i X + b_i Y + c_i Z + d_i = 0$ ; it is obtained as follows: Let  $\mathbf{M}_0$ ,  $\mathbf{M}_1$  and  $\mathbf{M}_2$  be three non collinear points in the  $X_r, Y_r$  plane of the road reference system. As an example  $\mathbf{M}_0 = [0 \ 0 \ 0]^T$ ,  $\mathbf{M}_1 = [1 \ 0 \ 0]^T$  and  $\mathbf{M}_2 = [0 \ 1 \ 0]^T$ . The points are projected from the camera's reference frame by means of the transformation matrix defined in (1). In the camera's reference system, those points are contained by the image plane and may be used to define two vectors whose cross product defines the vector normal to the image plane:

$$\begin{bmatrix} a_i \\ b_i \\ c_i \end{bmatrix} = [{}^c\mathbf{R}_v \cdot (\mathbf{M}_0 - \mathbf{M}_1)] \otimes [{}^c\mathbf{R}_v \cdot (\mathbf{M}_0 - \mathbf{M}_2)], \quad (17)$$

where  $\otimes$  denotes the cross product. The remaining image plane parameter  $d_i$  is obtained by substituting in the plane equation one of the projected points:

$$d_i = -(a_i X_0 + b_i Y_0 + c_i Z_0). \quad (18)$$

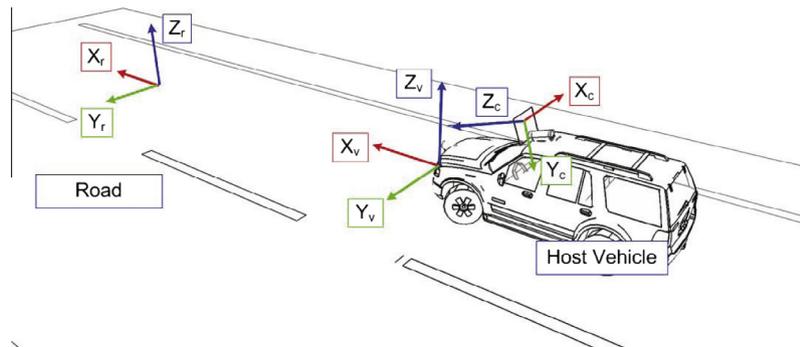
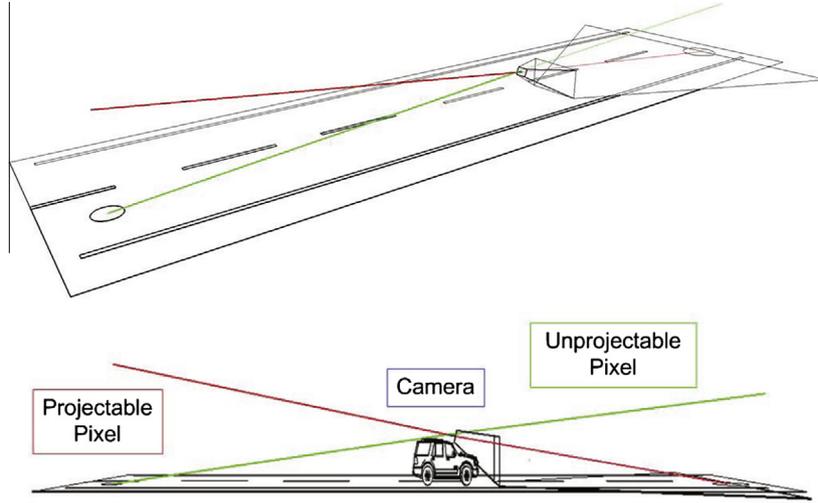


Fig. 3. A typical road scene. The host vehicle has a camera mounted on the roof. Note that the figure shows the reference systems of both the vehicle and the road, since they may not coincide.



**Fig. 4.** An example of a pixel that cannot be projected (*green*) since its optical ray intersects the road plane on the back of the image plane. Inversely, the pixel in *red* is projectable. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

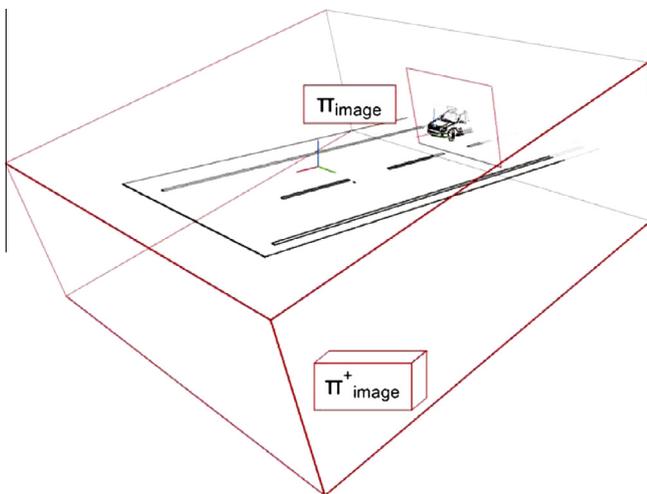
Having the parameters of the image plane, and a test point  $\mathbf{Q}_t = [X_t \ Y_t \ Z_t]^T$  that is sure to be inside the half space of projection (for example a point a couple of meters in front of the host vehicle), a test is devised to assess if a point  $\mathbf{Q} = [X \ Y \ Z]^T$  belongs to the half space of projection (denoted as  $\Pi_{image}^+$ ):

$$\begin{cases} \mathbf{Q} \in \Pi_{image}^+, & \text{if } (a_i X_t + b_i Y_t + c_i Z_t + d_i)(a_i X + b_i Y + c_i Z + d_i) > 0, \\ \mathbf{Q} \notin \Pi_{image}^+, & \text{otherwise.} \end{cases} \quad (19)$$

The half space of projection in (19) is shown in Fig. 5. It is used to define projectable polygons in 3D, as detailed in the following sections.

### 6.2. Desired area of perception

For an autonomous system, it is important to define the area of perception that it requires to effectively navigate. A very large perception area increases the computational cost, while a small perception area might make the system unfit to handle quick variations in the road scenario. This section addresses the desired perception limits, i.e., how the programmer can effectively set an



**Fig. 5.** The half space of projection computed after the image plane.

area of interest for the host vehicle to perform the IPM operation. In the case of a vehicle travelling in urban scenarios for example, perhaps 30 m of view range are sufficient. The desired area of perception is formally defined as a polygon  $\psi_{dap}$  in the road's projection plane. This polygon must be contained in the half space of projection ( $\psi_{dap} \subset \Pi_{image}^+$ ). Fig. 6 shows an example of an area of perception.

Currently,  $\psi_{dap}$  is set as a four vertices polygon, defining, in the road plane, a rectangle in front of the host vehicle. The rectangle's side in the direction of the vehicle's movement may dynamically increase size depending on the vehicle speed.

### 6.3. Image boundaries

Besides the desired area of perception, other regions of the road plane must be defined in order to perform an effective IPM operation. The camera lens properties and orientation towards the road plane define a possible area of projection. Let  $\gamma$  be the list of pixels in the image boundaries, obtained from all image pixels  $\mathbf{q}$  ( $\gamma \subseteq \mathbf{q}$ ) that are in accordance with:

$$\begin{cases} \mathbf{q}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix} \in \gamma, & \text{if } (x_i = 1 \vee x_i = W \vee y_i = 1 \vee y_i = H), \\ \mathbf{q}_i \notin \gamma, & \text{otherwise,} \end{cases} \quad (20)$$

where  $W$  and  $H$  are the image width and height respectively. The boundaries of the image are then projected onto the road plane using the inverse projection  $ip$  from (14), and the real world coordinates of the image boundary pixels  $\Gamma$  are obtained. The half space of projection is again used to assert the validity of 3D points:

$$\Gamma = ip(\gamma), \quad \forall ip(\gamma) \in \Pi_{image}^+ \quad (21)$$

The list of world points  $\Gamma$  are used to form the vertices of the polygon  $\psi_\Gamma$  (an illustration is shown in Fig. 7).

### 6.4. Laser generated polygon

The IPM technique requires that the road surface seen from the cameras is flat. This might not always be the case, particularly when other vehicles or obstacles lie on the road, as shown in the IPM resulting images published by some authors [18,22,10]. In these examples, artifacts are generated in the regions of the image where the flat road assumption fails. Vehicles are mapped as if they had been painted on the road (see Fig. 1(b) and (d)). Some authors

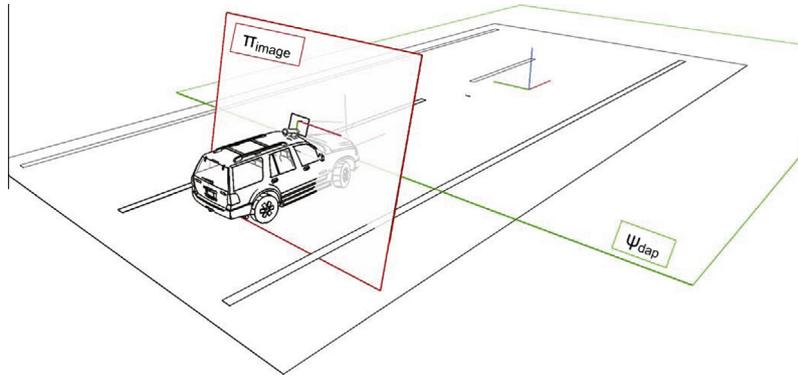


Fig. 6. The desired area of perception, polygon ( $\psi_{dap}$ ) in green. All vertices of this polygon should be contained by the half space of projection, according to (19). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

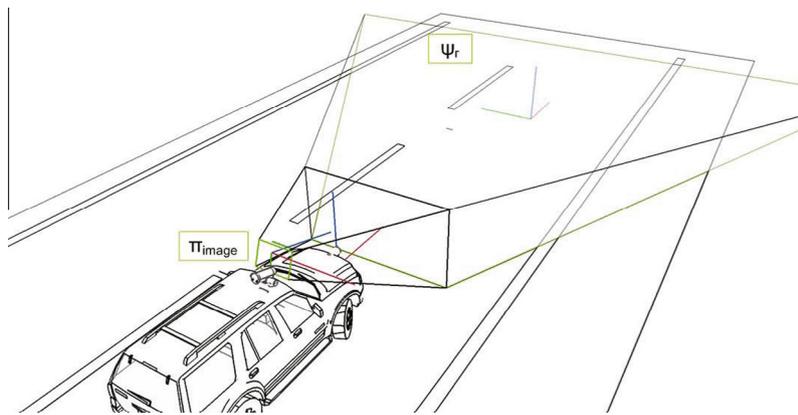


Fig. 7. The projection of image boundary pixels onto the road plane results in the image boundaries polygon ( $\psi_r$ ).

have taken advantage of this phenomenon to detect obstacles in the road, by using the differences in two IPM images, from a pair of stereo cameras [22]. This method is called stereo IPM. Although the latest is a valid approach, the fact is that calibration issues tend to disrupt the perfect mapping of stereo images. Because of this, it may sometimes be difficult to distinguish if disparities in the IPM stereo are due to a sub-optimum calibration or to an obstacle that lies on the road surface. There is also work related to sensor integration using both vision and laser in autonomous vehicles [31],

but in this case the objective was to enhance obstacle detection. Fig. 8 shows a typical urban road scenario with several obstacles near the host vehicle.

Let  $\mathbf{Q}_{laser} = [X_{laser} \ Y_{laser} \ Z_{laser}]^T$  be the 3D points obtained by the laser range finder, referenced in the world coordinate system. Assuming that objects picked up by the laser have a vertical expression, the coordinates where obstacles touch the floor, i.e., the object baseline  $\mathbf{Q}_{bln}$ , is obtained by the vertical projection of laser points onto the road plane:

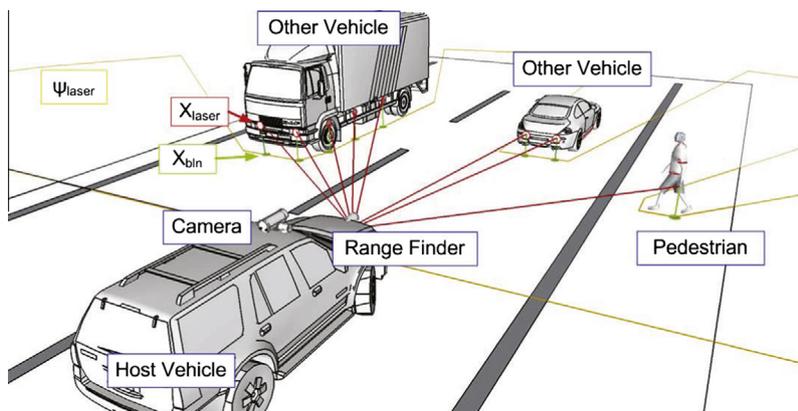


Fig. 8. A typical urban road scenario with several obstacles near the host vehicle.

$$\mathbf{Q}_{bln} = \begin{bmatrix} X_{laser} \\ Y_{laser} \\ -\frac{(a_r X_{laser} + b_r Y_{laser} + d_r)}{C_r} \end{bmatrix}. \quad (22)$$

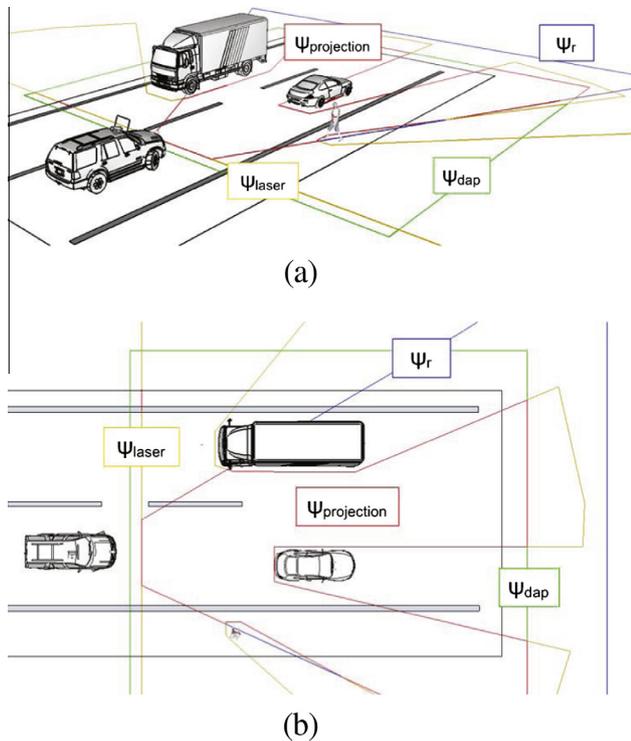
The laser generated polygon  $\psi_{laser}$  is defined by the list of vertices at generic coordinates given by  $\mathbf{Q}_{bln}$ .

### 6.5. Image projection polygon

As stated before, the core of IPM is applying (14) to the pixels in the image that are known to be on the projection plane. The objective is to be able to define for the input image which pixels are possible (and desirable) to map. The proposed approach defines three polygons in the road plane: a polygon defining the desired area of perception ( $\psi_{dap}$ ), a polygon corresponding to the boundaries of the image ( $\psi_r$ ) and a polygon defining the laser scanned objects ( $\psi_{laser}$ ). The resultant projection polygon ( $\psi_{projection}$ ) is obtained by the intersection of the three other polygons:

$$\psi_{projection} = \psi_{dap} \cap \psi_r \cap \psi_{laser}, \quad (23)$$

where  $\cap$  represents polygon intersection. The projection polygon is composed of a list of vertices, i.e., 3D points defined the road reference system (see fig. 9). The vertices defined in the road reference system are direct projected into the image plane using (8). The result is a list of 2D vertices that define a polygon in the image plane. This is called the image projection polygon. Inside the polygon are all pixels that should be mapped using IPM. Since perspective transformation is an affine transformation, the image projection polygon is calculated as the direct projection of the vertices of the projection polygon in the road plane.



**Fig. 9.** A road scenario with several obstacles: isometric view (a) and top view (b). The projection polygon ( $\psi_{projection}$ ) is shown in red. It is obtained by the intersection of the desired area of perception ( $\psi_{dap}$ ) in green, the image boundaries polygon ( $\psi_r$ ) in blue, and the laser generated polygon ( $\psi_{laser}$ ) in yellow. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

## 7. Results

Several experiments have been devised to obtain quantitative results of the proposed IPM methodology. First, the platforms used to obtain the results are presented: a dual camera pan and tilt unit (PTU), a small scale robot and finally, a full scale autonomous vehicle. The computational performance of the proposed approach is compared to the classic IPM using a measure of the accuracy of IPM. Results are presented for the accuracy of the proposed approach. Also, a comparative study of the classic IPM versus the laser assisted IPM shows that the accuracy of the latest is much better when obstacles appear in the area of projection. Finally, this section ends with some qualitative results, providing images of IPM from on-board cameras of a full scale vehicle.

### 7.1. Test platforms

In order to assess the performance of the proposed methodology, the test platforms depicted in Fig. 10 were used. Fig. 10(a) shows the dual camera PTU. The servo actuated PTU controlled through RS232 serial protocol was used so that the IPM is tested when the cameras move into different positions. The cameras have different lenses and also different image resolution. Camera 0 has a wide angle lenses and a resolution of  $800 \times 600$  pixels, while Camera 1 has a tele-lens and a resolution of  $320 \times 240$  pixels. This platform is used to assess the computational performance of the proposed approach. The time taken to perform IPM on both cameras is measured during a test where the PTU moves the cameras to different positions.

Fig. 10(c) shows the ATLASMV robotic platform. It is a small scale autonomous robot built for participating in an autonomous driving competition. It is equipped with four cameras and a LRF. The side cameras (Fig. 10(b)), used to map the road in front of the robot, have wide angle lenses and produce images with a resolution of  $320 \times 240$  pixels. The ATLASMV is used in two tests: one for measuring the accuracy of IPM, another to assess the effects of using the LRF to assist IPM. The quantitative results obtained from the accuracy of IPM are calculated using a color<sup>2</sup> calibrated grid (shown in Fig. 10(c), below the robot and, in Fig. 11, viewed from the cameras).

The grid is a  $3 \times 1$  m sheet of paper marked with a special colored pattern. The grid is placed in a known position in front of the cameras. Using the position and rotation of the cameras with respect to the calibration grid, a virtual image of the grid is produced to overlap the resultant IPM image. This virtual image of the grid serves as a test mask for measuring the IPM accuracy ( $\eta_{IPM}$ ): after projection using IPM, pixels are labelled with a color that should match the color of the virtual image. The accuracy of the projection is obtained as the ratio between correctly projected pixels and the total projected pixels:

$$\eta_{IPM} = \frac{\text{number\_of\_correct\_projections}}{\text{total\_number\_of\_projections}}. \quad (24)$$

Fig. 11(a) shows all the pixels of a given projection. Pixels classified as correctly and incorrectly projected are displayed in Fig. 11(b) and (c) respectively. Also, the virtual grid is overlaid onto the images.

The final test platform is the ATLASCAR (Fig. 12), a real scale robotic platform (<http://atlas.web.ua.pt>) [34] used for research on autonomous driving and advanced driver assistance systems. It is equipped with cameras and lasers. Results will show IPM images using the three on-board cameras.

<sup>2</sup> For interpretation of color in Figs. 10, 11 and 20, the reader is referred to the web version of this article.

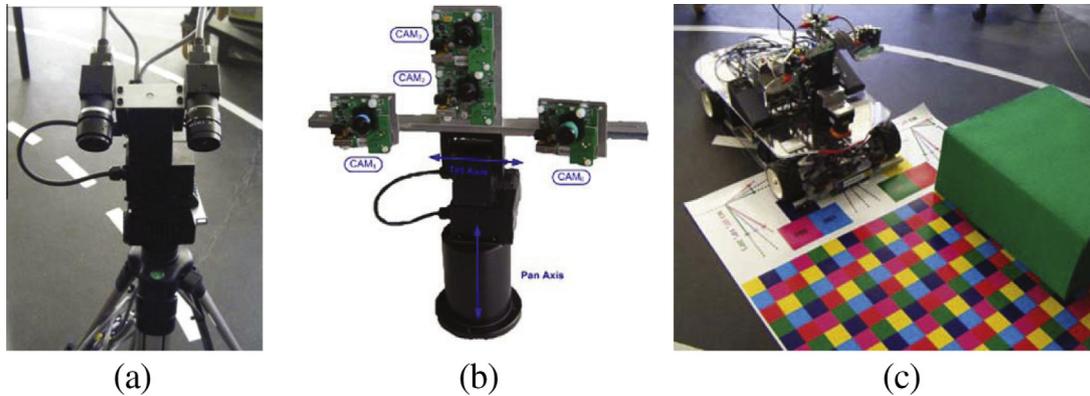


Fig. 10. Two of the test platforms used for testing the proposed approach: dual camera PTU unit (a); the ATLASMV small scale robot (c), equipped with a LRF and a multi-camera perception unit (b).

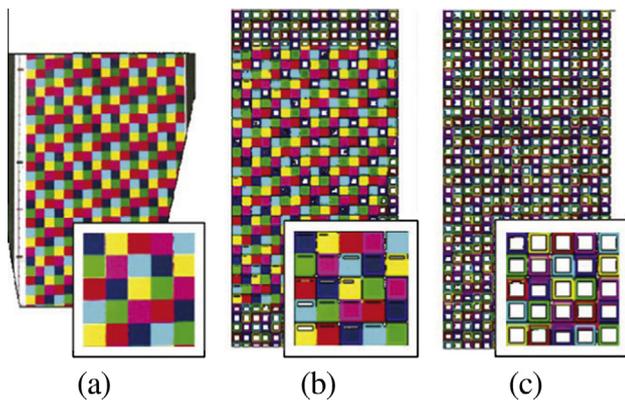


Fig. 11. The entire projection obtained using *Camera 0* of the ATLASMV test platform (a). The correctly projected pixels (b). Pixels that were incorrectly projected (c). Below each image, an enlarged region of the pixels is shown.



Fig. 12. The ATLASCAR full scale robotic platform. It is equipped with an active perception unit (A), a stereo rig (B), three LRF (C, D, H), a thermal vision camera (F), GPS (G) and an inertial measurement unit (E).

## 7.2. Computational performance

The computational performance of the IPM transformation has been a concern of some authors [21,35]. Its implementation on on-board systems requires real time performance from the systems. In order to test the performance of the proposed approach, the dual camera PTU setup was used (Fig. 10(a)).

In a classic IPM, all pixels in a given image are inverse projected, i.e., (14) is applied to all pixels. On the other hand, the proposed approach first computes the image projection polygon, and then applies (14) only to the pixels that should be projected. The computational demand of an IPM operation depends on the amount of projected pixels, which in turn depends on the camera's pose towards the projection plane. For example, a camera pointing to the sky will have only a small amount of pixels viewing the road plane.

To compare the performance of classic IPM with the proposed approach a 14 s test sequence was devised. Since the orientation of the camera's towards the road plane changes the amount of projectable pixels, during the 14 s of the test, the PTU is ordered to go to specific positions:

- State 1 (0–5.5 s) the PTU is moving upwards. This causes an increasingly smaller amount of mappable pixels for both cameras.
- Stage 2 (5.5–8.5 s) moves the PTU down and the inverse phenomena occurs.
- Stage 3 (8.5–14 s) maintains a fixed tilt and the PTU pans increasingly to the left, which will make *Camera 1* to have increasingly less mappable pixels.

Fig. 13 shows some IPM resulting images of key points in the test sequence. Fig. 14(a) compares the projection time of both cameras using the classic IPM and the proposed approach. Fig. 14(b) indicates the amount of projected pixels and the time saved using the proposed approach in relation to the classic IPM.

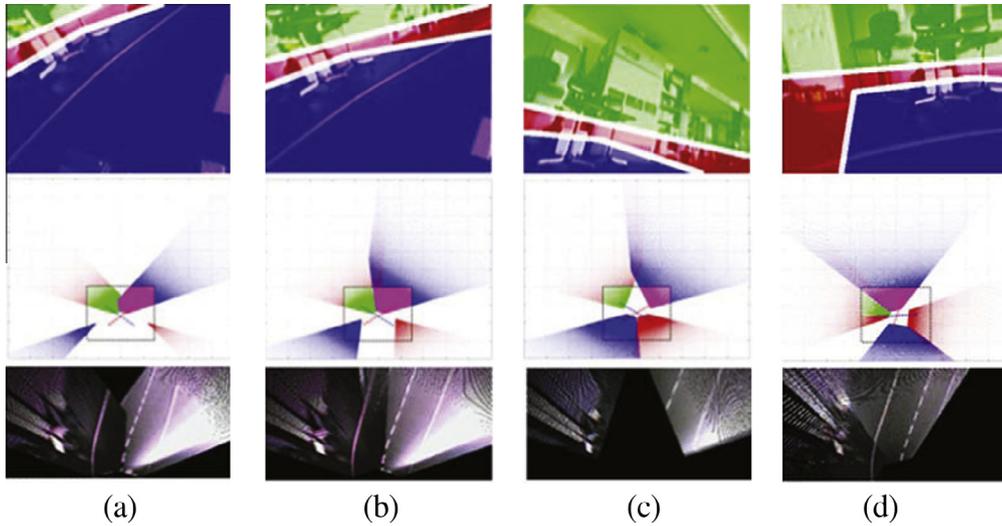
From 0 to 5.5 s, the PTU is moving upwards and so the pitch angle of the cameras is changing. This is observable in the difference of mapping in Fig. 13(a)–(c). Fig. 14(b) shows a reduction in the number of projected pixels for each camera. In Fig. 14(a), a reduction of IPM projection time using the proposed approach is clearly noted. *Camera 0* takes more time to project than *Camera 1* because the resolution of the images is different ( $800 \times 600$  pixels and  $320 \times 240$  pixels, respectively).

From 5.5 to 8.5 s the PTU is moving downwards and the effects are the inverse.

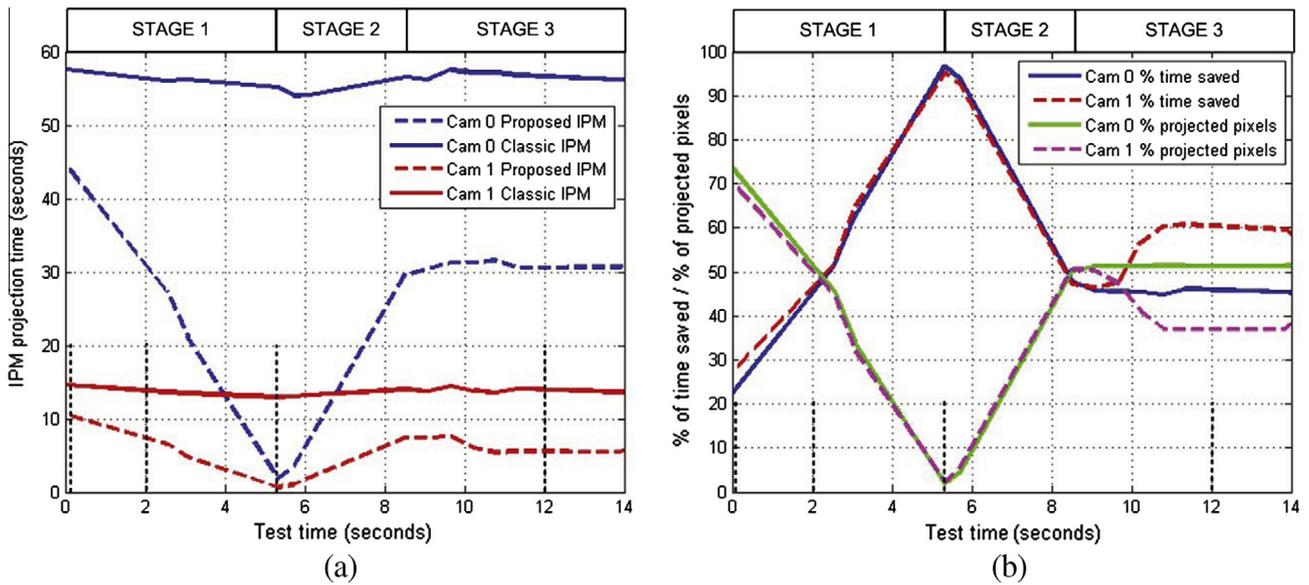
From 8.5 to 14 s the change in pan angle causes *Camera 1* to view increasingly less of the desired area of perception. Fig. 14(b) shows a decrease in the number of projected pixels during this period.

## 7.3. IPM accuracy

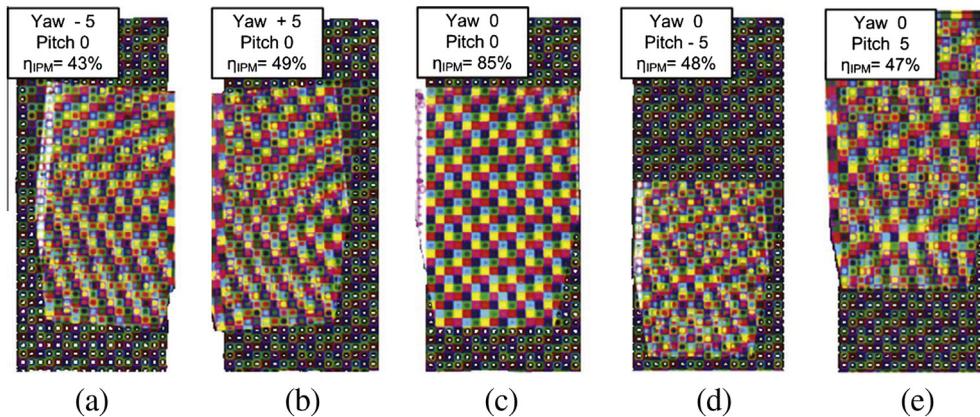
Although many researchers have employed the IPM operation in order to ease the road recognition process [2–5], the fact is that



**Fig. 13.** Some key frames of the test sequence. *First row:* images taken from *Camera 0*, the blue area is the area of projected pixels, the red is the area outside the desired area of perception and the green area is the area outside the half plane of projection; *Second row:* a map of the projection. Projected/unprojected pixels from *Camera 0* in green/red. Projected/unprojected pixels from *Camera 1* in magenta/blue; *Third row:* the IPM resulting image after mapping both cameras. In columns, different snapshots of the test sequence: 0 (a), 2 (b), 5.5 (c) and 12 (d) seconds. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



**Fig. 14.** (a) Time taken to perform the IPM projection for both cameras. The classic IPM time and the time of the proposed approach are shown. (b) Percentage of time saved and number of projected pixels. Proposed approach compared to the classic approach. Key frames of Fig. 13 signaled as the vertical black lines.



**Fig. 15.** The resulting IPM projection when errors in yaw (a), (b) and in pitch (d), (e) are introduced in the calculation. The reference projection, where no errors were introduced, is shown in (c).

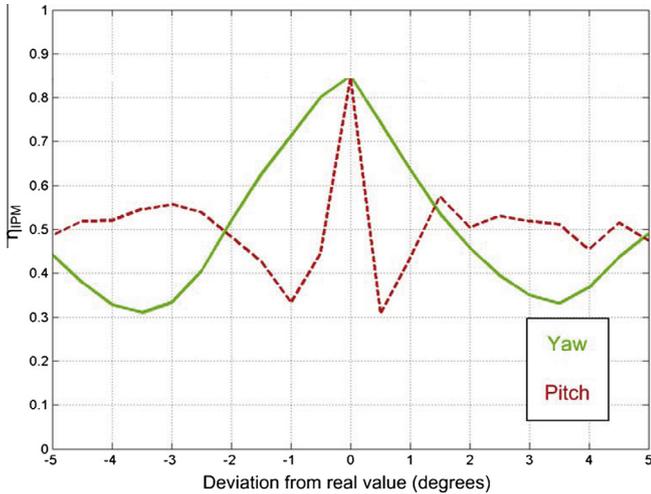


Fig. 16. IPM accuracy ( $\eta_{IPM}$ ) scores for errors in camera pose. Results are presented for errors in yaw and pitch angles.

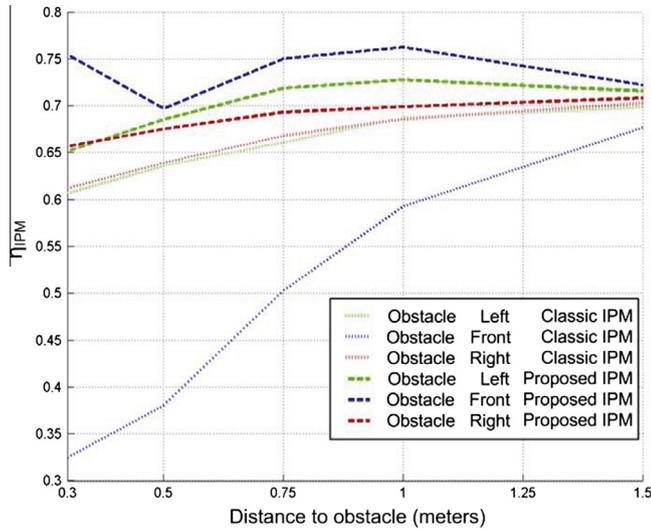


Fig. 17. IPM accuracy ( $\eta_{IPM}$ ) for the classic IPM (dotted lines) and the proposed approach (dashed lines).

no reporting of the accuracy of each implementation was found in the literature. Despite some insights on the topic of accuracy measurement for general projective geometry [10,11], a method had to be devised for this particular application to provide a quantitative analysis of the proposed method. For this experience the dual camera PTU setup was used (Fig. 10(a)). The calibration grid

presented in Section 7.1 was employed and an accuracy of  $\eta_{IPM} = 0.85$  was achieved for the system. Because the current paper is the first to present such quantitative results, a measure of the quality of this value is not possible.

The second experiment is intended to assess how important is to have accurate measures of the camera’s position and orientation with the road plane. In other words, how does the uncertainty of the camera pose estimation reflect on the final accuracy of the projection. For this purpose, errors in the yaw and pitch angles of the camera were introduced, and the IPM accuracy was calculated. Fig. 15 shows the resulting IPM of mappings with some errors (a), (b), (d) and (e) and the resulting image with no errors (c).

Fig. 16 shows the decrease in IPM accuracy with the increase of error in yaw and pitch. The pitch angle is the most relevant for the projection accuracy, since a half degree error changes the accuracy from 0.85 to 0.30. Variations in yaw also drop the accuracy value to 0.30, but only after a 3.5 degree deviation. This is consistent with the concerns of several researchers worldwide that mention on-board camera’s pitch estimation to be a cumbersome problem. In [30], for example, it is stated that “a small error in the vehicle’s roll/pitch estimation leads to a massive terrain classification error forcing the vehicle off the road. Such situations occur even for roll/pitch errors below 0.5 degrees”.

7.4. IPM accuracy using LRF

In order to test the usage of the LRF on the IPM projection the ATLASMV robot was used. An obstacle with 0.2 m height (green box in Fig. 10(c)) was placed over the calibration grid in front of the robot at several distances and in several positions (to the left, right or in front of the robot). For each obstacle position the accuracy was computed. Fig. 17 shows the  $\eta_{IPM}$  results both for the classic and the proposed IPM approach.

The laser polygon introduced in Section 6.4 is able to classify pixels that view the obstacle as unapplicable. Because of this, the proposed IPM approach (Fig. 17, dashed lines) consistently gets better accuracy results than the classic IPM (Fig. 17, dotted lines). When the obstacle is very close (0.3 m, Fig. 18(a)), using a classic IPM operation would be catastrophic (0.33 accuracy ratio) but the proposed approach remains accurate enough (0.75).

In theory, the performance of IPM should increase when the obstacle is moved to a higher distance. This is not always observable in Fig. 17: for example the curve *Obstacle Front, Proposed IPM* shows a decrease from 0.3 to 0.5 m. We believe that these variations in the performance measurement methodology might possibly be caused by noise in the input images. In fact, some other experiences also show a decrease in performance when the distance increases. For example, *Obstacle Left, Proposed IPM*, also decreases performance from 1 m to 1.25 m. Nonetheless, the main point is that, in the cases where an obstacle is present in the image, despite these variations in performance measurements, it is

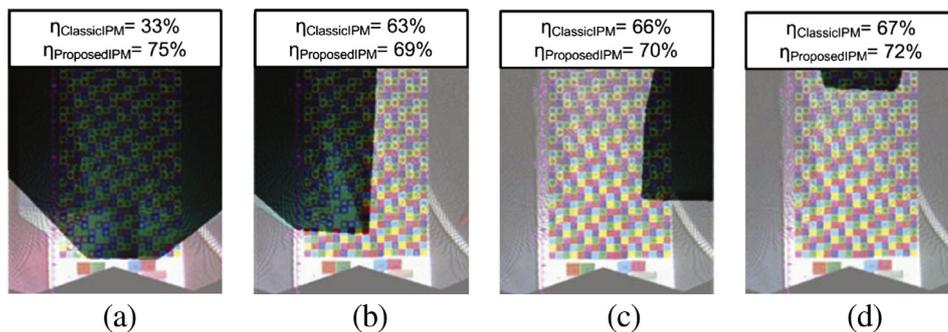


Fig. 18. Some tested scenarios: (a) obstacle at 0.3 m in front; (b) obstacle at 0.5 m to the left; (c) obstacle at 0.75 m to the right; and (d) obstacle at 1.5 m in front.

InputImage	Classical IPM	Multimodal IPM

**Fig. 19.** Comparison of the classical IPM (middle column) with the proposed Multimodal IPM (right column). The input image (left column) shows the image projection polygon highlighted in yellow. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

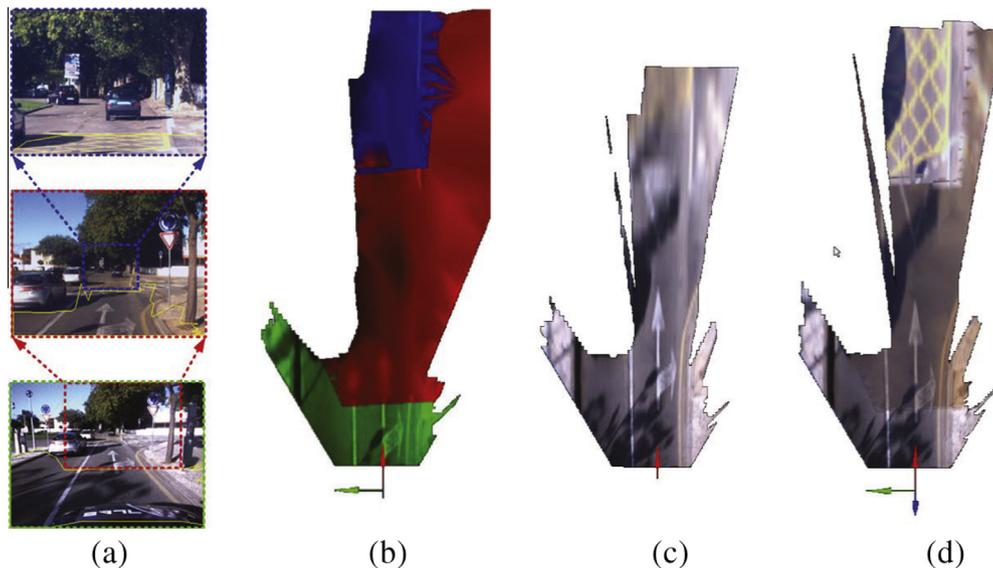
evident that the proposed IPM always shows a better performance when compared to the classic IPM.

Fig. 18 shows the IPM resultant images for some of the tested scenarios.

7.5. Tests in real environments

For the final validation of the proposed approach, several tests in real road scenarios with a full scale vehicle were done. The test

platform used was the ATLASCAR (Fig. 12). The platform is equipped with three cameras (each with a different focal distance lens) and several lasers. Several hours of data from urban and highway roads were used for validating the algorithm. The proposed approach is less time consuming, is able to deal with pitch/roll variations due to brake/turning maneuvers, and using the LRF copes with obstacles present in the projection area. Fig. 19 provides a qualitative comparison of the classical IPM with the proposed multimodal IPM. It is possible to observe that in the



**Fig. 20.** Using the proposed IPM approach in real scenarios. (a) Images of the three cameras on-board the ATLASCAR; (b) the distribution of mapping for each camera; (c) IPM using just green camera; and (d) IPM using all cameras

presence of other vehicles or obstacles, the classical IPM produces several artifacts on the resultant image. On the contrary, the multimodal approach to IPM is able to cope with obstacles and removes them from the resultant image. Even in free road scenarios, as is the case of Fig. 19 (fourth line), the artifacts produced by the parked cars could reduce the effectiveness of a road detection approach.

The flexibility of the proposed approach can also handle the usage of several input cameras. In Fig. 20, the three cameras on-board the ATLASCAR (Fig. 12), each with different focal distance lenses, are used to obtain a more detailed mapping of the road. Fig. 20(a) shows images from the three cameras. The IPM is mapped to the road plane and the distribution of pixels supplied by each camera is shown in Fig. 20(b). Using a single camera to map the road (the green camera), shows the classical problems of lack of accuracy at long distances (the yellow traffic pattern in Fig. 20(c)). However, if multiple cameras are employed, the tele-camera (blue camera) can provide a high resolution view at long distances, which leads to a high resolution view of the yellow pattern of the road (Fig. 20(d)).

Several video sequences showing results from classical and proposed IPM can be found at <https://dl.dropboxusercontent.com/u/18364290/IF/index.html>.

## 8. Conclusions

The current paper presents a flexible mathematical model for performing IPM. The methodology is to fuse laser data with vision data in order to improve the accuracy of the IPM projection. The fusion mechanism is based on the intersection of polygons defined according to several criteria: the algorithm computes the polygons generated from the image boundaries, the laser obstacles and the desired area of perception; then, the combination of these polygons (projection polygon) is projected back to the image plane, resulting in the image projection polygon. The image projection polygon is defined in the image coordinate system and can therefore be directly used to as a criterion to indicate which pixels are to be mapped through IPM, and which should not be mapped.

Different test platforms, from a small scale robot to a full scale autonomous vehicle, were used to obtain both quantitative and qualitative results. Results show that the proposed approach is computed in less time than the classic IPM, and that the IPM image produced by the proposed approach has higher accuracy when obstacles are present in the road. A study of the influence of errors in the camera's pose estimation to the IPM projection accuracy which corroborates previous findings is also presented. Finally, several hours of data both from urban roads as well as highways were qualitatively analyzed to evaluate the robustness and efficiency of the proposed approach.

In sum, this paper proposes a novel algorithm that solves a common problem of the classical IPM: the disrupting of the IPM image when obstacles are present in the road. The proposed solution is to fuse the information from the images with the data from a LRF in order to specify in the image which pixels are viewing the ground plane and should therefore be mapped using IPM. Results show that the proposed method is more efficient than classical IPM.

## Acknowledgements

Part of this work was developed under the grant SFRH/BD/43203/2008, from the Portuguese National Science Foundation – FCT, and supported by the Spanish Research Project reference TIN2011-25606.

## References

- [1] Federal Highway Administration, Data Fusion for Delivering Advanced Traveler Information Services, United States Department of Transportation Intelligent Transportation Systems Joint Program Office, 2003, pp. 247–254.
- [2] N.-E. El Faouzi, H. Leung, A. Kurian, Data fusion in intelligent transportation systems: progress and challenges – a survey, *Inform. Fusion* 12 (1) (2011) 4–10.
- [3] C. Stiller, F.P. León, M. Kruse, Information fusion for automotive applications – an overview, *Inform. Fusion* 12 (4) (2011) 244–252.
- [4] A. Sathyannarayana, P. Boyraz, J.H. Hansen, Information fusion for robust 'context and driver aware' active vehicle safety systems, *Inform. Fusion* 12 (4) (2011) 293–303.
- [5] V. Kastrinaki, M. Zervakis, K. Kalaitzakis, A survey of video processing techniques for traffic applications, *Image Vision Comput.* 21 (4) (2003) 359–381.
- [6] N. Simond, M. Parent, Obstacle detection from ipm and super-homography, in: *IROS, IEEE, 2007*, pp. 4283–4288.
- [7] P. Cerri, P. Grisleri, Free space detection on highways using time correlation between stabilized sub-pixel precision IPM images, in: *Proceedings of the IEEE International Conference on Robotics and Automation, Barcelona, Spain, 2005*, pp. 2223–2228.
- [8] M. Guanglin, P. Su-Birm, S. Miiller-Schneiders, A. Ioffe, A. Kummert, Vision-based pedestrian detection-reliable pedestrian candidate detection by combining IPM and a 1D profile, in: *Proceedings of the IEEE Intelligent Transportation Systems Conference, Seattle, Washington, USA, 2007*, pp. 137–142.
- [9] C. Lundquist, T.B. Schön, Joint ego-motion and road geometry estimation, *Inform. Fusion* 12 (4) (2011) 253–263.
- [10] M. Aly, Real time detection of lane markers in urban streets, in: *Proceedings of the IEEE Intelligent Vehicles Symposium, Eindhoven, Netherlands, 2008*, pp. 7–12.
- [11] J. McCall, O. Achler, M. Trivedi, Design of an instrumented vehicle test bed for developing a human centered driver support system, in: *Proceedings of the IEEE Intelligent Vehicles Symposium, Parma, Italy, 2004*, pp. 483–488.
- [12] A. Muad, A. Hussain, S. Samad, M. Mustafa, B. Majlis, Implementation of inverse perspective mapping algorithm for the development of an automatic lane tracking system, in: *Proceedings of the IEEE International Conference TENCON, vol. A, Chiang Mai, Thailand, 2004*, pp. 207–210.
- [13] H. Mallot, H. Bülthoff, J. Little, S. Bohrer, Inverse perspective mapping simplifies optical flow computation and obstacle detection, *Biol. Cybernet.* 64 (3) (1991) 177–185.
- [14] D. Pomerleau, RALPH: rapidly adapting lateral position handler, in: *Proceedings of the IEEE Intelligent Vehicles Symposium, Detroit, USA, 1995*, pp. 506–511.
- [15] T. Ehgen, T. Pajdla, D. Ammon, Eliminating blind spots for assisted driving, *IEEE Trans. Intell. Transp. Syst.* 9 (4) (2008) 657–665.
- [16] H. Fang, M. Yang, R. Yang, C. Wang, Ground-texture-based localization for intelligent vehicles, *IEEE Trans. Intell. Transp. Syst.* 10 (3) (2009) 463–468.
- [17] S. Li, Y. hai, Easy calibration of a blind-spot-free fisheye camera system using a scene of a parking space, *IEEE Trans. Intell. Transp. Syst.* 12 (1) (2011) 232–242.
- [18] J. McCall, M. Trivedi, Performance evaluation of a vision based lane tracker designed for driver assistance systems, in: *Proceedings of the IEEE Intelligent Vehicles Symposium, Las Vegas, NV, USA, 2005*, pp. 153–158.
- [19] F. Dornaika, J. Álvarez, A. Sappa, A. López, A new framework for stereo sensor pose through road segmentation and registration, *IEEE Trans. Intell. Transp. Syst.* 12 (4) (2011) 954–966.
- [20] C. Guo, S. Mita, D. McAllester, Stereovision-based road boundary detection for intelligent vehicles in challenging scenarios, in: *Proceedings of the IEEE International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 2009*, pp. 1723–1728.
- [21] M. Bertozzi, A. Broggi, A. Fascioli, Real-time obstacle detection on a massively parallel linear architecture, in: *3rd International Conference on Algorithms and Architectures for Parallel Processing, Melbourne, Australia, 1997*, pp. 535–542.
- [22] M. Bertozzi, A. Broggi, GOLD: a parallel real-time stereo vision system for generic obstacle and lane detection, *IEEE Trans. Image Proc.* 7 (1) (1998) 62–81.
- [23] S. Tuohy, D. O'Cuailain, E. Jones, M. Glavin, Distance determination for an automobile environment using inverse perspective mapping in OpenCV, in: *Proceedings of the IET Irish Signals and Systems Conference, Cork, Ireland, 2010*, pp. 100–105.
- [24] A. Broggi, P. Cerri, S. Ghidoni, P. Grisleri, H.G. Jung, Localization and analysis of critical areas in urban scenarios, in: *Proceedings of the IEEE Intelligent Vehicles Symposium, Eindhoven, Netherlands, 2008*, pp. 1074–1079.
- [25] S. Tan, J. Dale, A. Anderson, A. Johnston, Inverse perspective mapping and optic flow: a calibration method and a quantitative analysis, *Image Vision Comput.* 24 (2) (2006) 153–165.
- [26] L. Luo, I. Koh, S. Park, R. Ahn, J. Chong, A software-hardware cooperative implementation of bird's-eye view system for camera-on-vehicle, in: *Proceedings of the IEEE International Conference on Network Infrastructure and Digital Content, Beijing, China, 2009*, pp. 963–967.
- [27] P. Coulombeau, C. Laugeau, Vehicle yaw, pitch, roll and 3D lane shape recovery by vision, in: *Proceedings of the IEEE Intelligent Vehicles Symposium, Versailles, France, 2002*, pp. 619–625.

- [28] 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, 2003, pp. 31–36.
- [29] M. Nieto, L. Salgado, F. Jaureguizar, J. Cabrera, Stabilization of inverse perspective mapping images based on robust vanishing point estimation, in: *Proceedings of the IEEE Intelligent Vehicles Symposium*, Istanbul, Turkey, 2007, pp. 315–320.
- [30] S. Thrun, M. Montemerlo, A. Aron, *Probabilistic terrain analysis for high-speed desert driving*, in: *Proceedings of the Robotics Science and Systems Conference*, University of Pennsylvania, University of Pennsylvania, Philadelphia, PA, 2006, pp. 21–28.
- [31] A. Broggi, S. Cattani, P.P. Porta, P. Zani, A laserscanner-vision fusion system implemented on the terramax autonomous vehicle, in: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, 2006, pp. 111–116.
- [32] A.F. Massimo Bertozzi, Alberto Broggi, An extension to the inverse perspective mapping to handle non-flat roads, in: *IEEE International Conference on Intelligent Vehicles*. *Proceedings of the 1998 IEEE International Conference on Intelligent Vehicles*, 1998, pp. 305–310.
- [33] A. Sappa, F. Dornaika, D. Ponsa, D. Gerónimo, A. López, *An efficient approach to on-board stereo vision system pose estimation*, *IEEE Trans. Intell. Transp. Syst.* 9 (3) (2008) 476–490.
- [34] V. Santos, J. Almeida, E. Avila, D. Gameiro, M. Oliveira, R. Pascoal, R. Sabino, P. Stein, *Atlascar – technologies for a computer assisted driving system on board a common automobile*, in: *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, Madeira Island, Portugal, 2010, pp. 1421–1427.
- [35] O.D. Evans, Y. Kim, *Efficient implementation of image warping on a multimedia processor*, *Real-Time Imaging* 4 (6) (1998) 417–428.