

On-board Stereo Sensor for Intersection Driving Assistance. Architecture and Specification

Sergiu Nedevschi, Marita Tiberiu, Radu Danescu, Florin Oniga, Silviu Bota
Computer Science Department, Technical University of Cluj-Napoca, Romania
Sergiu.Nedevschi@cs.utcluj.ro

Abstract

This paper presents the specifications and architecture for a stereovision sensor to be used in intersection assistance. The intersection problem imposes a wide field of view, reasonable accuracy for the typical intersection length, and fast response time. The image and 3D data provided by the low level routines are used to generate two kinds of environment descriptions – an unstructured description, composed of elevation maps, occupancy grids, and polylines delimiting obstacle areas and curbs, and a structured description, composed of lanes, cuboidal objects, and classified pedestrians. The descriptions can be further combined, and additional data sources can be used, in order to provide a complete and accurate description of the intersection.

1. Introduction

Intersections are the most complex traffic situations, as they can be both confusing and dangerous. Due to the demanding nature of the scenario, the research community is increasingly focused on solving the perception and acting problems related to the intersection. A new joint research project, INTERSAFE-2 (www.intersafe-2.eu), aims at developing a system for cooperative sensor data fusion, based on state of the art passive and active on-board sensors, navigation maps, information from other traffic participants and from intelligent infrastructure, in order to generate a complete and accurate description of the complex intersection environment [1].

The dense stereovision sensor is maybe the sensor that provides the highest amount of usable information, as it combines the visual data with the dense 3D information that can be deduced through precise inference from the binocular view. A reliable stereovision sensor for urban driving assistance has been developed in [2]. However, the intersection scenario has some specific demands from a stereo sensor, demands that impose changes in the physical

setup and in the software algorithms. In [3] the features provided by a stereovision sensor usable in intersection assistance applications are highlighted.

The main objective of this paper is to provide the detailed specifications of the stereovision sensor (architecture and functions) in accordance with the INTERSAFE-2 project requirements [4]. A second goal is to highlight the new methods that are proposed as solutions, together with their experimental evaluation.

2. Functions of an intersection driving assistance system

The analysis of the user needs for Intersection Safety Assistance Systems identified the following driving assistance functions [4]:

- 1. Left turn assistance (LTA)** - intended to support drivers making a safe left turn after having come to a stop in the intersection. The sensor system is required to detect oncoming vehicles, and the gaps between them, as well as localize the host vehicle within the intersection. Oncoming vehicles to be detected include cars, trucks and motorcycles.
- 2. Intersection crossing assistance (ICA)** - is intended to support the driver when crossing an intersection in straight path and to resolve potential collisions with crossing traffic (both vehicles and vulnerable road users). The sensor system is required to detect crossing vehicles and vulnerable road users crossing the host vehicle's intended path.
- 3. Right turn assistance for cars (RTA)** - is intended to support the driver when turning right in an intersection. It is supposed to resolve potential collisions with crossing traffic, particularly with vulnerable road users. The sensor system is required to detect crossing vehicles and vulnerable road users crossing the host vehicle's intended path.
- 4. Right of way and stop line assistance (SLA)** – is intended to support the driver when approaching an intersection at which they may be required to give

way to other traffic. It is intended to avoid collision risk with crossing traffic. The sensor system is required to detect crossing vehicles and localize the host vehicle on its approach to the intersection.

3. Requirements for the stereo sensor

The main roles of the stereovision sensor in an intersection driving assistance system are related to the sensing and perception. The identified roles of the stereo system are described in Table 1. In the table below *vehicles* will refer to: cars, trucks and motorcycles while *vulnerable road users* will refer to pedestrians and cyclists.

Table 1. Roles of the stereovision sensor

	Role	L T A	I C A	R T A	S L S
1	Detect the presence and measure the relative position and velocity of the <i>oncoming vehicles</i>	◀			
2	Detect the presence of the <i>crossing vehicles and crossing vulnerable road users</i> and measure their position and velocity relative to the host vehicle.	◀	▲	▶	■
3	Localize the host vehicle within the intersection.	◀	▲	▶	
4	Detect road markings and lane boundaries in front of the host vehicle and measure their relative position to the host vehicle	◀	▲	▶	■
5	Localize the host vehicle when it is close to the stop line				■
6	Detect painted road signs (inverted triangular give-way and octagonal stop, turn left, turn right, go ahead signs) conforming to [5] in front of the host vehicle and measure their relative position to the host	◀	▲	▶	■
7	Measure the acceleration of <i>oncoming</i> and <i>crossing vehicles</i>	◀	▲	▶	■
8	Deliver stable tracks of detected <i>oncoming</i> and <i>crossing vehicles</i>	◀	▲	▶	■

4. Architecture of the stereo sensor

The intersection scenario imposes radical changes in the physical setup and in the processing algorithms of a stereo sensor. Due to the need for a wider field of view, which comes with distortions and reduced depth accuracy, increased accuracy in calibration and dense stereo reconstruction is required. The stereo matching

process has to be performed on rectified images, by a dedicated stereo board, to free processor time for the high-level algorithms. In order to cope with the complex nature of the intersection, the proposed solution perceives the environment in two modes: a structured approach, for the scenarios where the road geometry is estimated from lane delimiters, and an unstructured approach, where the road geometry is estimated from elevation maps. The structured mode provides the parameters of the lane, and the position, size, speed and class of the static and dynamic objects, while the unstructured mode provides an occupancy grid having the cells labeled as free space, obstacle areas, curbs and isles.

4.1 The stereo image acquisition architecture

The image sensors can be integrated in a dedicated stereo-head or built from stand-alone cameras mounted rigidly on a stereo-rig. The cameras can be interfaced through a frame-grabber or a common PC interface (Firewire, USB). The image acquisition is controlled by a software interface customized for the used cameras (Figure 1). The quality of the acquired images is tuned through the Gain and Exposure parameters of the cameras for automatic adaptation to the lighting conditions [6].

4.2. Low level stereo processing overview – sensorial data processing

The stereo images are acquired at the full resolution of the image sensors. They are further rectified (to meet the canonical configuration), corrected (to eliminate the lens distortions) and down-sampled (to meet the processing capacity of the hardware dense-stereo reconstruction engine – e.g. 512 pixels in width). This process can be performed on a dedicated hardware or implemented in software. The software implementation of this process is reduced to an image warping approach performed in a single step using reverse mapping and bilinear interpolation in order to minimize the noise introduced by the digital rectification and image correction [7]. The software implementation is optimized for processing speed-up using MMX instructions and lookup tables. The rectified (canonical) down-sampled images are fed to the stereo reconstruction module which is implemented in hardware. From the depth map provided by the stereo-engine the 3D coordinates of the reconstructed points are computed knowing the camera parameters and are further provided to the processing modules (Figure 1).

For a stereo setup using 6.5 mm lenses, 2/3" imagers and a baseline of 320 mm a stereo detection area with the following parameters is obtained (fig. 2):

- minimum distance: 0.5 m in front of the ego car;
- horizontal field of view: 72 deg;
- reliable detection range: 0.5 ... 35 m;
- delimiters of the current lane are visible at 1.0 m;

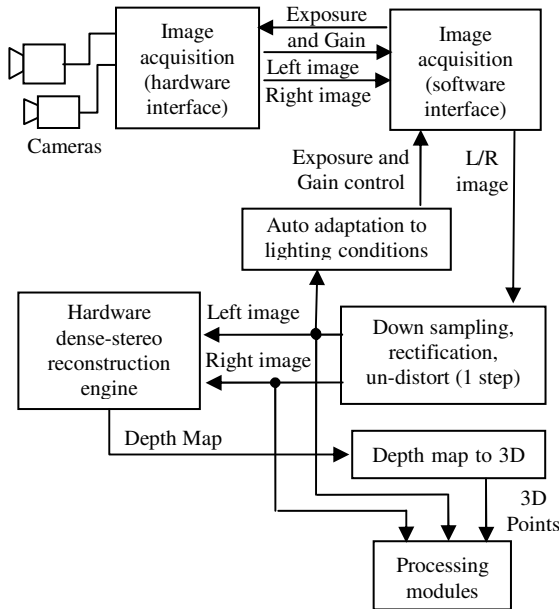


Figure 1. The stereovision system architecture.

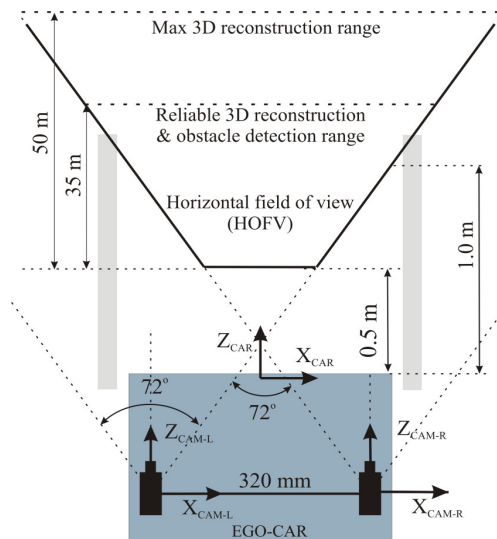


Figure 2. Detection area of the current stereo system setup.

High accuracy calibration of the camera intrinsic and extrinsic parameters is required [8]. Intrinsic parameters are defining the optical and geometrical characteristics of each camera and are: the principal

point coordinates, the focal length and the lens distortions coefficients. The absolute extrinsic parameters are describing the position (T_L and T_R translation vectors) and orientation (R_L and R_R rotation matrices) of each camera relative to the reference coordinate system (related to the ego car). From the absolute extrinsic parameters the stereo system parameters are derived: relative position and orientation of the two cameras, epipolar geometry parameters etc.

4.3. Overview of the high level stereo perception functions

The perception functions will exploit the 3D and 2D data together with the ego vehicle motion data. The processing is focused on the stereo detection area shown in (Figure 2). The functions will be focused on both static and dynamic objects.

Static road and intersection environment perception functions are:

- Current and side lanes detection based on lane delimiters (lane markings, curbs, fences, traffic poles and trees);
- Target lane and side lanes detection based on same lane delimiters;
- Lane turning based on lane delimiters turning (or lane parameters changing);
- Stop line detection,
- Pedestrian crossing detection
- Painted signs (stop, give way, turn right, turn left, and go ahead) – figure 3;
- The lane turnings, stop lines, pedestrian crossings and painted signs will generate intersection hypotheses and will allow intersection corners detection.
- Static obstacle detection including parked vehicles, road construction sites delimiters.

Dynamic road and intersection environment perception functions are:

- preceding, oncoming and crossing vehicles detection and tracking;
- preceding, oncoming and crossing vulnerable road users detection and tracking

Tracking of the dynamic objects will provide relative position, speed and acceleration information. Using them, the movement history of the ego and tracked vehicle is inferred.

4.4. High level perception paradigms

Due to variability of the driving scenarios complexity, three perception paradigms are used:

- the structured environment paradigm in which the road geometry is estimated from the lane/road delimiters and the obstacles are modeled as cuboids, having position, size, orientation and speed. This paradigm corresponds to un-crowded environments with visible lane delimiters.
- the unstructured environment paradigm in which the road geometry and obstacles are detected from the digital elevation map. This paradigm corresponds to overcrowded environments like intersections in which the lane delimiters are not visible. Starting from the representation of the dense 3D data as an elevation map, an occupancy grid which defines the drivable area, 3D curbs, traffic isles and obstacles is generated.
- the fusion of the results provided by the above mentioned paradigms will refine the results both in qualitative and quantitative parameters.

5. High level perception methods

5.1. Structured environment perception methods

5.1.1. Lane perception

The lane the vehicle is currently on will be described using the geometry parameters: *width*, *width variation* (the variation of the instant width along the lane, linear), *curvature*, *lateral offset*, and the angles of *pitch*, *yaw* and *roll*. The description of the current lane will include information about the lane delimiters, which consists of *delimiter class* (interrupted marking, continuous marking, curb, or undefined), and *delimiter width*, for each side, when the width can be computed (the delimiter width applies to lane markings, as it is undefined for curbs or other types of lane delimiters). The lane parameters will include a *lane length*, which will be useful when the vehicle approaches the intersection. The length of the lane can be at most the viewing distance for reliable road data, and therefore depends on the camera and lens setup. For the current lane, the reference point is the origin of the car coordinate system, and therefore known.

The lane perception algorithms combine stereo 3D information and grayscale image information into model-based probabilistic trackers that continuously estimate the parameters of the lane geometry, appearance and position [9]. The 3D information allows the direct estimation of the pitch angle, and the separation between the road features and the rest of the features in the image. The grayscale information allows the extraction of the lane markings. The particle filter-based tracking framework allows the combination and experimentation with multiple cues. Besides the current lane, which will be estimated to the maximum accuracy

possible, the system will detect additional lanes, which could be target lanes after the car clears the intersection.

The *target lanes* (or any types of lanes that we are not currently on, but we can detect) will be modeled in the same manner as the current lane, possibly using a more restricted set of parameters, as some of the parameters are shared with the current lane (pitch angle, roll angle) and some are too demanding to be accurately detected from a distance (such as width variation). However, these lanes need to have additional parameters, specifying their reference point on the road surface (the lateral and longitudinal coordinate).

5.1.2. Painted road object perception

Sometimes, the painted road markings are not lane delimiters, or cannot be used to directly estimate a lane shape. These objects, marked with white (or yellow) color on the darker surface of the asphalt, are usually trapezoid-shaped, can be parts of an interrupted lane delimiter, parts of a pedestrian crossing pattern, part of a road arrow, and so on. By detecting these objects as trapezoids, we can further analyze their position and find the complex shape. The painted road objects will be detected using the features that are contained in the road surface (identified using the stereo information) and the grayscale information that indicates the dark-light-dark transitions, and tracked using their parametrical model (3D oriented trapezoid) and probabilistic methods.

The generic painted road object can be defined by the following parameters: *position* (lateral and longitudinal; belonging to the road, it has no height or vertical position), *base width*, *width variation*, *length* and *orientation*. Basically, this object is an oriented trapezoid. This object can successfully represent a wide range of specific road objects, such as: segments of an interrupted lane marking, segments of a pedestrian crossing marking, parts of the arrows, and so on. The generic road object will be detected and tracked, and then the pieces will be combined to determine the compound structures.

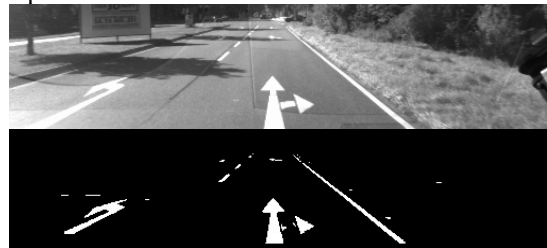


Figure 3. Preliminary results for painted road objects detection.

5.1.3. Obstacle detection

The obstacle detection algorithms analyze, on the compressed top view of the scene, the local density and vicinity of the 3D points, and determine the occupied areas which are then fragmented into obstacles with cuboidal shape: without concavities and only with 90° convexities [10].

The fragmentation of the occupied areas into the individual cuboidal obstacles is the next step.

The orientation of the obstacles is determined in order to get a very good fitting of the cuboidal model to the obstacles in the scene ahead and, consequently, to minimize the free space which is encompassed by the cuboids.

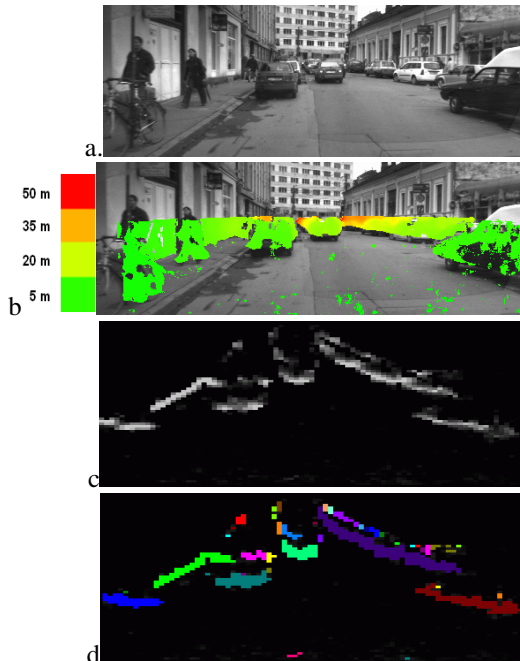


Figure 4. 3D points grouping in occupied areas: a) Gray scale image, b) 3D points – perspective view, c) The compressed space – a bi-dimensional histogram counting 3D points. d). Occupied areas are identified by cell labeling.

The obstacles are described using the oriented cuboidal model, described by its corners. The obstacle’s speed and class are also supplied. The supplied classes are pedestrian, bicyclist, car, truck, pole, and tree.

5.1.4. Obstacle tracking

An alternative to the Kalman filter based obstacle tracking [11] is based on multiple particle filters, and does not require points grouping [3]. It is working directly on the top view of the raw obstacle data, or on the occupancy grid computed from the elevation map (see paragraph 5.2)

This solution is a multiple object tracking scheme that uses a two-level approach. First, a particle filter-based tracker that starts with an initial probability density function that covers the whole possible object parameter state space runs freely on the 3D data, and its particles will tend to cluster around one object.



Figure 5. Results on obstacle detection in intersection scenarios

When the clustering is achieved, the particle state is passed to an individual object tracker, that will track the object until it disappears, and the initialization tracker is reset and will start searching for another object.

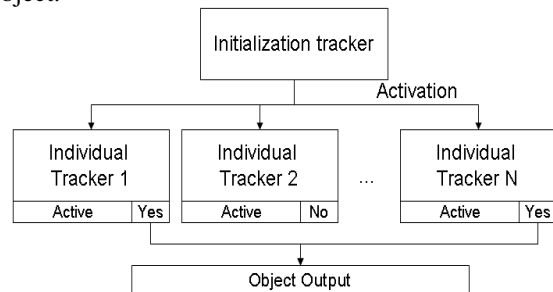


Figure 6. Top-level architecture of the particle filter-based tracking system

Although the individual trackers and the initialization tracker are built around similar particle filters, their behavior is different. Fig. 7.a shows the flowchart for the initialization tracker. We’ll provide a short description for each block. A more detailed description is presented in [15].

Resample – the process that transforms the weighted particle set describing the past probability density into an equivalent set of weightless particles, by random weighted sampling.

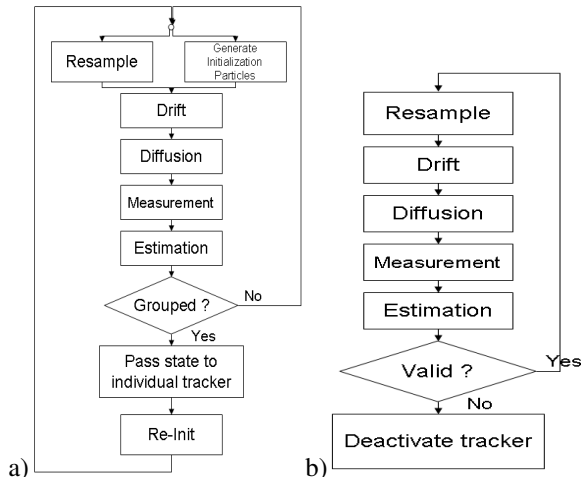


Figure 7. a) Flowchart of operation for the initialization tracker, b) Flowchart of operation for the individual object tracker

Generation of initialization particles – a fraction of the particle set will be sampled from the initial probability distribution, that is, from the whole range of possible object states. These particles help the tracker evade a false hypothesis faster.

Drift – the uniform motion dynamic model is applied to the particles (details in the next section)

Diffusion – a random value is applied to each particle, accounting for the uncertainties of state transition.

Measurement – each particle is compared to the measurement data, and weighted according to a fitness score.

Estimation – an estimated object state is computed by weighted average of the particle values. This step also computes a standard deviation value for each estimated parameter.

“Grouped” decision - The initialization tracker uses only the standard deviations of the object position to decide whether the particles are grouped. If they are not grouped, the tracker starts another cycle. If they are grouped, the whole particle distribution is passed to an individual tracker, which is made active, and the initialization tracker is reset.

The operation of the active individual object tracker is depicted in Fig. 7.b. The basic blocks are similar to the initialization tracker, but there are some notable differences. First, there are no initialization particles, as these trackers operate at local level, their target already selected. Second, after estimation the next block is a “valid” condition, which tests the grouping of the particles, the position and size of the estimated object (if grows too big, or too small, or goes out of the field of view, it becomes invalid). If the tracker becomes

invalid, it goes into the inactive state, and the tracked object is declared lost.

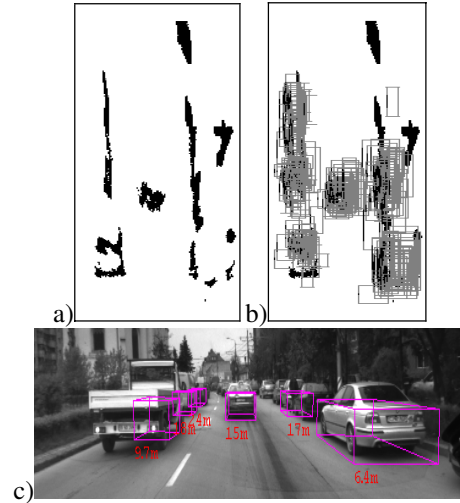


Figure 8. a) Obstacle areas in 3D top view, b) Particles of the object trackers, c) Results (estimations) in perspective projection

5.1.5. Obstacle class detection

In order to assess the dynamic intersection environment it is useful to know the class of the detected obstacles. Interesting classes are pedestrians (being the most vulnerable road users), cyclists, cars, trucks, poles, trees etc. Classification will be based on multiple features, some of which apply to all classes and some of which are class specific [12]. Global features are the obstacle’s dimensions, and its speed. Class specific features are contour shape, motion signature, texture etc. Classifiers will be implemented using decision trees and adaboost.

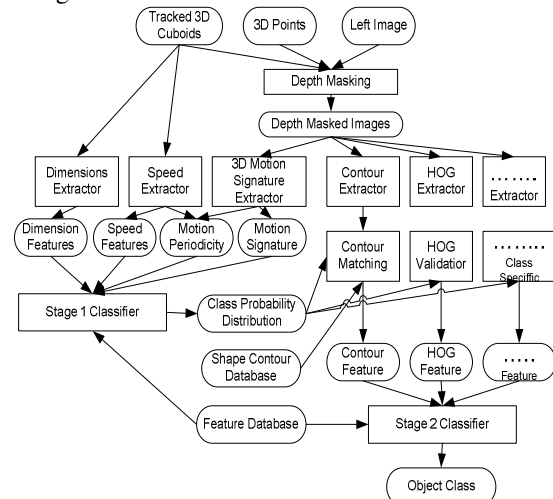


Figure 9. Architecture of the object classifier



Figure 10. Pedestrian contour matching. Green contours are extracted from the images and yellow contours are templates matched with these contours.

5.2. Unstructured environment perception methods

The digital elevation map (DEM) is a grid of heights computed from the set of 3D points. Each grid cell covers a 10x10 cm horizontal square in the 3D world and contains a height value representative for the 3D points contained by the cell (points that project vertically into the cell). The default size of the grid is 128x400 cells. The top view size of the 3D space of interest is 12.8 x 40 meters. These values can be modified to be more appropriate for intersection scenarios.

The intersections in urban areas often have no lane markings or these are occluded by crowded traffic. The lanes cannot be detected in such particular scenes, thus the obstacle/road separation becomes difficult. An alternative/complementary method must be used to detect elevated areas (obstacles), regions where the ego vehicle cannot be driven into. Complementary, the obstacle-free road areas can be considered as drivable. Starting from the representation of the dense 3D data as an digital elevation map, an occupancy grid is computed. Each grid cell has an associated label: drivable (road), traffic isle, obstacle (static/dynamic, or others), etc. This way the following entities are defined: drivable area, traffic isles, and obstacles [13].

The 3D curbs can be accurately detected from the DEM. The algorithm presented in [14] detects curbs as chains of segments applying edge detection methods.

Primary 3D road delimiters are available in two representations: an advanced occupancy grid with labeled cells or a set of polylines associated to the occupancy grid entities [16]. Polylines, on the other hand, offer a low memory cost representation, more suitable for CAN transmission.

Recently we have developed a method of cell-based tracking, which is able to extract dynamic parameters of grid regions, without the need of clustering (fig. 14). This tracking method uses as input the occupancy grid, and models the state of the cell and the errors of the measurement grid using probabilistic methods.

5.3. Fusion of the results provided by the structured and unstructured environment perception methods

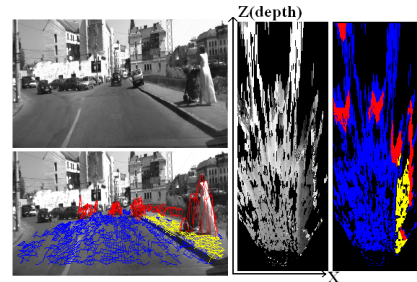


Figure 11. top-left: A common urban scene, bottom-left: the occupancy grid projected onto the left image. Middle - the elevation map. Right - the occupancy grid resulting after classification: road cells with blue, traffic isle cells with yellow, and obstacle cells with red.



Figure 12. A curb is detected in front of the ego while turning left in an intersection.

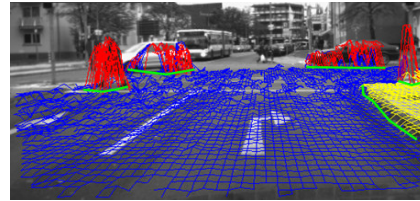


Figure 13. Poly-line representation derived from the occupancy grid.

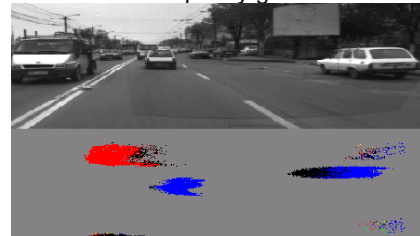


Figure 14. Unstructured grid tracking example: blue – regions moving away (vehicle in front), red – incoming regions (vehicle in the left lane), black – stationary regions (parked white car on the right).

The two types of representations can be used independently for different driving assistance functions, or they can be fused to produce a more accurate description of the environment.

One type of fusion, that is already implemented, is the use of curbs, provided by elevation map analysis, into the model-based lane estimation process, as the curbs have the same lane delimiting value as the road markings. Another type of fusion is the use of obstacle areas in the reconstruction and tracking of structured objects.

The fusion of structured and unstructured environment descriptions must not always be the use of unstructured results in the refining of structured parameters. As future work, we can use the model-based tracking of lanes and obstacles to refine the grid description. The possibilities of structured/unstructured fusion are multiple, and they are bounded only by the computation cost limits.

6. Conclusions/ Discussions

A sensor based on stereo image processing can provide a wide range of valuable information to the intersection driving assistance system. Even though it lacks the accuracy of the radar and of the laser sensors, it compensates by the richness of data, and by the endless possibilities of high level processing. The fusion of stereo results with active sensors, such as laser, with map information, and with odometry data, will ensure a complete and accurate description of the intersection environment.

7. Acknowledgement

This work was conducted within the research project INTERSAFE-2. It is part of the 7th Framework Programme, funded by the European Commission. The partners of INTERSAFE-2 thank the European Commission for supporting the work of this project.

8. References

- [1] INTERSAFE-2 Consortium, "User needs and requirements for INTERSAFE-2 assistant system" deliverable D3.1, available via <http://www.intersafe-2.eu/public/> cited 28th April 2009.
- [2] S. Nedevschi, R. Danescu, T. Marita, F. Oniga, C. Pocol, S. Sobol, C. Tomiuc, C. Vancea, M. M. Meinecke, T. Graf, T. B. To, M. A. Obojski, "A Sensor for Urban Driving Assistance Systems Based on Dense Stereovision", in *Proceedings of 2007 IEEE IV Symposium*, June 13-15, 2007, Istanbul, pp.278-286
- [3] S. Nedevschi, R. Danescu, T. Marita, F. Oniga, C. Pocol, S. Bota, „Stereovision-Based Sensor for Intersection Assistance”, in *Advanced Microsystems for Automotive Applications 2009*, Springer, ISBN 978-3-642-00744-6.
- [4] INTERSAFE-2 Consortium, "Specification and Architecture documentation" deliverable D4.1, available via <http://www.intersafe-2.eu/public/>, cited 28th April 2009.
- [5] Economic Commission for Europe - Inland Transport Committee, Convention on Road Signs and Signals,

(<http://www.unece.org/trans/conventn/signalse.pdf>), Vienna, Nov. 1968.

- [6] M. Negru, S. Nedevschi, "Camera Response Estimation. Radiometric Calibration", in *Proceedings of 2009 IEEE ICCP*, August 27-29, 2009, Cluj-Napoca.
- [7] C. Vancea, S. Nedevschi, "Analysis of different image rectification approaches for binocular stereovision systems", in *Proceedings of 2006 IEEE ICCP*, September 1-2, 2006, Cluj-Napoca, Romania, vol. 1, pp. 135-142.
- [8] T. Marita, F. Oniga, S. Nedevschi, T. Graf, R. Schmidt, "Camera Calibration Method for Far Range Stereovision Sensors Used in Vehicles", *Proceedings of 2006 IEEE IV Symposium*, Tokyo, Japan, 13-15 June 2006, pp. 356-363
- [9] R. Danescu, S. Nedevschi, "Probabilistic Lane Tracking in Difficult Road Scenarios Using Stereovision", accepted for publication in *IEEE Transactions on Intelligent Transportation Systems*, 2009, Digital Object Identifier 10.1109/TITS.2009.2018328, [Pre-published in IEEE Xplore](#)
- [10] C. Pocol, S. Nedevschi, M.M. Meinecke, "Obstacle Detection Based on Dense Stereovision for Urban ACC Systems", in *Proceedings of WIT 2008*, Hamburg, Germany, March, 2008, pp 13-18.
- [11] R. Danescu, S. Nedevschi, M.M. Meinecke, T. Graf, "Stereovision Based Vehicle Tracking in Urban Traffic Environments", in *Proceedings of 2007 IEEE ITSC*, Seattle, USA, 2007, pp 400-404.
- [12] S. Nedevschi, S. Bota, C. Tomiuc, Stereo-Based Pedestrian Detection for Collision-Avoidance Applications, accepted for publication in *IEEE Transactions on Intelligent Transportation Systems*, 2009, Digital Object Identifier 10.1109/TITS.2008.2012373, [Pre-published in IEEE Xplore](#)
- [13] F. Oniga, S. Nedevschi, M. M. Meinecke, T. T. Binh, "Road Surface and Obstacle Detection Based on Elevation Maps from Dense Stereo", *Proceedings of 2007 IEEE ITSC*, Seattle, USA, Sept. 30 – Oct. 3, 2007, pp. 859-865.
- [14] F. Oniga, S. Nedevschi, M-M. Meinecke, "Curb Detection Based on a Multi-Frame Persistence Map for Urban Driving Scenarios", in *Proceedings of 2008 IEEE ITSC*, Oct. 2008, Beijing, China, pp. 67-72.
- [15] R. Danescu, F. Oniga, S. Nedevschi, M-M. Meinecke, "Tracking Multiple Objects Using Particle Filters and Digital Elevation Maps", in *Proceedings of 2009 IEEE IV Symposium*, June, 2009, Xi' An, China, pp. 88-93.
- [16] S. Nedevschi, A. Vatavu, F. Oniga, "Forward Collision Detection based on Elevation Map from Dense Stereo", in *Proceedings of the IROS 2008 2nd Workshop on Planning, Perception and Navigation for Intelligent Vehicles*, 2009, Nice, pp 76-81.