Superpixel-based Obstacle Segmentation from Dense Stereo Urban Traffic Scenarios Using Intensity, Depth and Optical Flow Information

Ion Giosan, Sergiu Nedevschi, *Member, IEEE*

Abstract— Obstacle detection is a necessary task in every driving assistance system. An accurate obstacle segmentation is very important for further processing tasks that are using the obstacle ROI as input, like obstacle classification. This paper presents a real time approach for obstacle segmentation from traffic scenarios, based on superpixels clustering. A pair of gray levels stereo-cameras is used for scene image acquisition. The stereo-reconstruction uses a sub-pixel level optimized semi-global matching (SORT-SGM) resulting in a very accurate 3D points map. Optical flow is computed using a Lukas-Kanade pyramidal approach. A novel paradigm integrating intensity, depth and optical flow information on superpixels is used for obstacle segmentation. SLIC superpixels are computed first based on intensity information. Multiple features are computed for each superpixel and used for clustering superpixels in obstacles. Depth cues are used for clustering the superpixels in obstacles and then optical flow information refines the obstacles clusters based on their motion. A qualitative and quantitative evaluation of the proposed approach and a comparison with other obstacle detection technique are finally presented.

I. INTRODUCTION

The number of intelligent vehicles is growing rapidly due to the technological possibilities that are into a continuous development process. Each intelligent vehicle is equipped with a driving assistance system (DAS). Usually it includes many safety functions like obstacle collision warning, lane departure warning, lane keeping assistance, speed keeping assistance, etc. There exists also a special category of intelligent vehicles in which the manufacturers built some protection parts that are automatically triggered in case of an imminent collision with pedestrians in order to reduce the risk of fatal injuries. Here comes the researchers' motivation for building high accuracy obstacle detection modules. Basically, an obstacle detection module provides a region of interest (ROI) from traffic scene as input for a follow up module which usually classifies it into a specific obstacle class. It is very important that the ROI is computed as accurate as possible, because the features that characterize the obstacle are usually determined from its analysis and may further influence the classification. In computer vision and mainly in urban traffic scenarios, the depth computation of

I. Giosan, S. Nedevschi are with Technical University of Cluj-Napoca, Computer Science Department, 28 G. Baritiu St., 400027, Cluj-Napoca, Romania (phone: 0040-264-401-484; e-mail: Ion.Giosan@cs.utcluj.ro, Sergiu.Nedevschi@cs.utcluj.ro).

each scene element is very important. This could be achieved by using stereo-cameras for images acquisition. We use two gray level cameras in a stereo setup due to their low cost and their specifications that are sufficient for achieving our goal. The stereo sensor offers us the possibility to accurately determine the depth distance value for the scene points and further to assign the optical flow vectors to them.

We present a novel real time approach for obstacle detection from urban traffic scenarios based on superpixel clustering. The superpixels are extracted with a modified SLIC approach using just the gray level information. We characterize each superpixel by multiple features based on intensity, depth and optical flow information. We cluster the superpixel into the corresponding obstacles using depth and road surface information. A refinement of the clusters, based on the optical flow information, is finally made. We also present an evaluation of our obstacle detection approach and we highlight its benefits in comparison with other obstacle detection technique.

II. RELATED WORK

Computer vision researchers carry out a lot of work for developing better and better solutions for obstacle detection from both monocular vision and stereo vision setup cameras.

In the case of monocular vision, common features like color or gray intensities [1], symmetry [2], edges [3], shadows [4] and textures [5] are widely used for obstacle detection. Motion from optical flow [6] may also be computed for detecting moving obstacles by subtracting the ego motion of the vehicle.

A stereo vision system acquires much more traffic scene information using at least two cameras. This allows the obstacle detection [7] to be done by analyzing both color/intensity and depth information [8] which considerably reduces the amount of noisy information. The approach is continued with a module that gathers the reconstructed points into obstacles by using a paradigm of points grouping [9] and density maps [10], followed by optical flow and motion computation for obstacle tracking [11]. A framework based on 3D points clouds representation is used for segmentation and classification of range image, which annotates class labels to the data clusters obtained through a graph-based segmentation, is presented in [12]. In [13], moving objects detection is achieved by using only spatial information in conjunction with an ego-motion estimation by sparse matched feature points. Another method that works well on sparse, noisy point clouds for semantic segmentation, based on 3D point clouds derived from ego-motion, is developed in

[14]. A technique for localization of scene elements through sparse stereovision, targeted at obstacle detection is described in [15]. Precise extraction of depth with robust and fast detection of moving objects in DAS is achieved in [16] with a powerful fusion of depth and motion information for image sequences taken from a moving observer. In contrast with other works that try to explicitly identify the obstacles, in [17], a computationally efficient approach to obstacle detection by a graph traversal on 2D grid of cells and 3D points elevation is presented. In [18], an obstacle detection system using stereo vision sensors is developed. The system uses feature matching, epipolar constraints and feature aggregation in order to robustly detect the initial corresponding pairs. After the initial detection, the system executes the tracking algorithm for the obstacles. In [19], a probabilistic representation of the uncertainty for stereo-vision, which takes advantage of distance and disparity is proposed. This model is then applied to obstacle detection, using the occupancy grid framework. A similar algorithm for the detection of free space and obstacles in the traffic scene by analyzing the 3D reconstructed structures of an environment modeled using probabilistic volume polar grid map is presented in [20]. Stereo-vision based obstacle detection systems usually require a dense disparity map and, then, locate the obstacles according to the depth information. Computing the correspondence for each pixel is very time consuming. An algorithm which significantly reduces the complexity disparity calculations and locate the obstacles is presented in [21].

In order to obtain very good performance, the obstacle detection system must have as an input high quality stereo-images with high quality stereo-reconstruction. We use the SORT-SGM stereo reconstruction [22] implemented on GPU, which has the advantage of providing a dense stereo depth map with high accuracy in a short processing time. The depth map is denser and more accurate than the one obtained with a local matching technique implemented on a classic hardware stereo-machine [23]. An approach for locally estimating the road parameters is presented in [24]. It assumes that the road is locally planar. This offers the possibility of dealing also with non-flat roads by splitting the geometry in a sequence of quasi-planar geometry parts. The method does not need lane markings extraction and takes into account the other relevant road information like texture, shadows, edges etc. Using this estimation, the system achieves a robust and accurate detection of the on road obstacles. In order to separate the road reconstructed points from the obstacle points, the road surface should be determined. We use the elevation maps approach presented in [25]. A quadratic road surface model is fitted to the region in front of the ego vehicle. This is followed by a region growing process driven by the 3D uncertainty model of the stereo sensor which refines the initial surface.

Superpixels consist in local coherent clusters of pixels based on local image features. They are usually used for reducing the complexity of subsequent image processing tasks with applications in depth estimation, image segmentation, body model estimation and object localization. In our system we use a modified SLIC superpixels [26] approach that works great for gray levels images.

III. System Overview

The obstacle detection system architecture with all its component modules and their interfaces (input and output) is depicted in Figure 1.
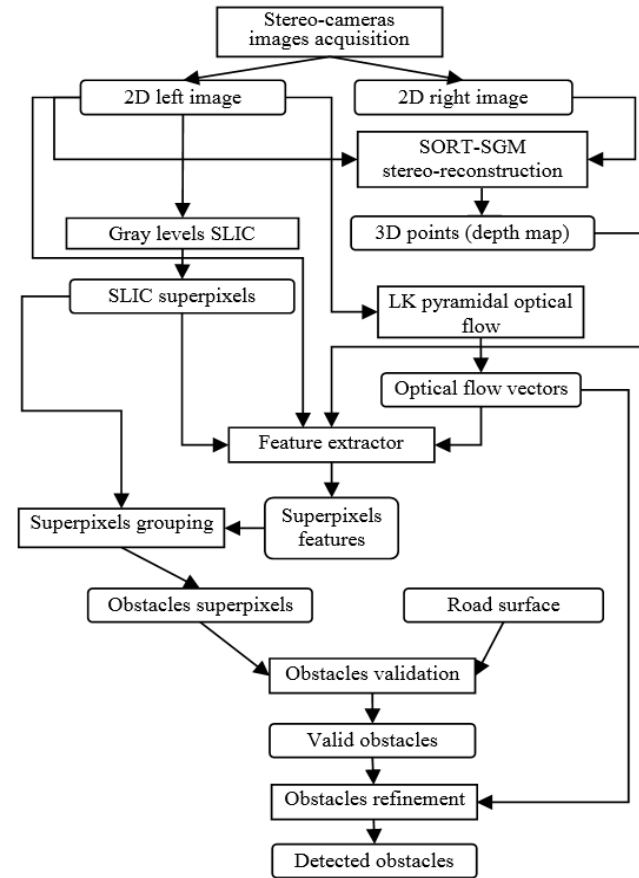

Figure 1. Superpixel based obstacle detection system architecture

Gray levels stereo traffic images with resolution of 512x383 pixels are acquired. Stereo-reconstruction is performed using a semi-global optimized algorithm (SORT-SGM) on a NVIDIA GeForce GTX 580 GPU having as input the two intensity images (undistorted and rectified). This results in an accurate and dense depth map that is essential in further processing. The depth map stores, for each reconstructed point, its distance from the stereo cameras setup. Optical flow vectors are computed considering "good-features-to-track" [27] into the Lukas-Kanade pyramidal approach. The implementation is also made on the same GPU. SLIC superpixels are extracted based on the intensity information. They are used further in a feature extractor module that computes specific intensity, depth and motion features for each superpixel. A labeling procedure based on the superpixels adjacency and similarity is used for their grouping into separate obstacles. The road surface is computed using elevation maps built upon the 3D points map. The obstacle validation block filters the spurious clusters based on the size and considering their scene positioning (they must be on the road surface). A final refinement using a special closing operation on superpixels and optical flow for separating the erroneous groups of multiple objects is applied. All these blocks are described in details in the next chapters.

IV. Gray Levels SLIC Superpixels

The first step in our superpixel based obstacle detection consists in the intensity image segmentation in a set of superpixels. We set a region of interest (ROI) over the input image (see Figure 2) where we compute the SLIC superpixels. Its position and size in pixels is defined by: (*left*=0, *top*=100, *right*=512, *bottom*=320). A number of *N*=2000 seeds are evenly distributed in the intensity image. The initial area (*ns*) of each superpixel cell (rectangular shape) is defined in equation (1).

$$ns = \frac{\text{width}(ROI) \cdot \text{height}(ROI)}{N} \qquad (1)$$

Each seed is considered as being the center of a separate cluster C_i, with $i=1\ldots N$ containing a total of *ns* pixels around it. In order to obtain a good segmentation, all the seeds are then shifted into the lowest gradient magnitude location. We consider a scanning neighborhood of 3x3 pixels around the initial position. The Prewitt convolution kernel is used for computing the derivative on both *x* and *y* directions. The gradient magnitude is the sum of the two absolute derivatives. A k-means clustering algorithm is then applied in order to assign each ROI pixel to a cluster. This is an iterative approach and we considered that 10 iterations is enough for a good convergence. The centers of the clusters are defined by the average on each of the three components dimensions: *x*, *y* and gray level *g*. We define a distance metric $d(P,C_k)$ from a point *P* inside the ROI to a cluster C_k assuming that the location (*x, y*) and the gray level *g* is known as in equation (2).
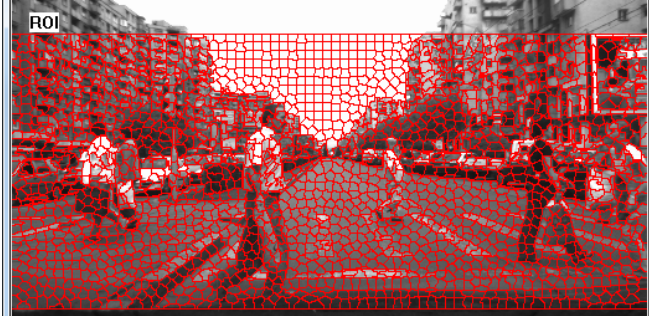

Figure 2. Scene image with ROI and final refined superpixel cells

$$d(P,C_k) = \frac{\left|g(P) - g(C_k)\right|}{ng} + \frac{\left|x(P) - x(C_k)\right| + \left|y(P) - y(C_k)\right|}{2 \cdot ns} \quad (2)$$

where *ng* is a weighting factor and considered *ng*=50.

After this process it might appear few superpixels with very small size. They are merged to a larger superpixel neighbor. We consider a superpixel to be very small sized if its area is less than a quarter from *ns*. The final result is depicted in Figure 2.

V. Superpixel Features Extraction

We compute a set of features *F* for each superpixel previously determined. The area *A* counts the total number of image points in each superpixel. Inside each superpixel, the *gray* intensities are known in every point, the (*X, Y, Z*) world position of the reconstructed points and the *OF* vectors where the optical flow displacements (*p, q*) are computed.

$$\boldsymbol{F} = (MI, MD, MedD, MedH, MedX, CD, \\ MMagOF, MAngOF, COF) \qquad (3)$$

MI is the mean intensity, *MD* is the mean depth, *MedD* is the median depth (*Z*), *MedH* is the median height (*Y*), *MedX* is the median of horizontal points coordinates (*X*), *CD* is the 3D points coverage, *MMagOF*, *MAngOF* define the magnitude and angle of the optical flow mean vector, *COF* is the optical flow vectors coverage (see equation (4)).

$$MI = \frac{\sum_{P \in A} \text{gray}(P)}{\|A\|}, MD = \frac{\sum_{P \in RecPoints} Z(P)}{\|RecPoints\|}, CD = \frac{\|RecPoints\|}{\|A\|}$$

$$MedD = \underset{P \in RecPoints}{\text{median}}\left(Z(P)\right)$$

$$MedH = \underset{P \in RecPoints}{\text{median}}\left(Y(P)\right)$$

$$MedX = \underset{P \in RecPoints}{\text{median}}\left(X(P)\right) \qquad (4)$$

$$MMagOF = \sqrt{MMagOF_p^2 + MMagOF_q^2}$$

$$MAngOF = \text{arctg}\frac{MMagOF_q}{MMagOF_p}$$

$$MMagOF_p = \frac{\sum_{P \in OF} p(P)}{\|OF\|}, MMagOF_q = \frac{\sum_{P \in OF} q(P)}{\|OF\|}, COF = \frac{\|OF\|}{\|A\|}$$

VI. Superpixels Clustering

Superpixels grouping is based on its feature vector *F* similarity. Our goal is to classify each superpixel in one of the three classes: road surface superpixel, obstacle superpixel and beyond driving area superpixel. Using the elevation map points flags (road surface points or above road surface points), we compute the road surface coverage of each superpixel (see equation (5)).

$$Road = \frac{\sum_{P \in RecPoints} \text{IsRoadPoint}(P)}{\|RecPoints\|} \qquad (5)$$

A superpixel is assigned to the road surface if *Road*>0.25 and *CD*>0.30 (see Figure 3).

We define the 3D region of interest volume as a rectangular parallelepiped delimited by the negation of *MedD*, *MedX*, *MedH* inequalities from the conditions set (6).
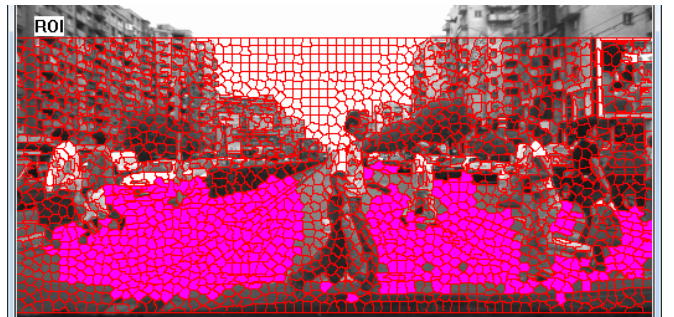

Figure 3. Road surface superpixels (with magenta color)

A superpixel is considered to be beyond the driving area if at least one of the following conditions is satisfied:

$$\begin{cases} CD = 0, \\ MedD > 40000\,\text{mm}, \\ MedH > 3500\,\text{mm}, \\ MedX < -10000\,\text{mm}, MedX > 10000\,\text{mm} \end{cases} \quad (6)$$

If the superpixels are neither road surface nor beyond driving area then they are considered to be obstacles superpixels. We implement a labeling algorithm, based on the vicinity of the superpixels and the features similarity between them. Each time when an unlabeled superpixel is found, a new label is generated. With a breadth first search approach this label is propagated to its neighbors and so on until no similar superpixels are found. The labeling results consist in a set of superpixel clusters. Each cluster is an obstacle candidate. In the next chapter, we describe the methodology for validating each obstacle candidate and for refining the valid obstacles.

An unlabeled superpixel T is considered to be similar with a superpixel S that is already part of an obstacle, if both of the following conditions are satisfied:

$$\begin{cases} CD_T > 0.55 \\ \left| MedD_S - MedD_T \right| < \text{Thr}\left(\min(MedD_S, MedD_T) \right) \end{cases} \quad (7)$$

The function Thr(d) defines a variable with depth threshold. Due to the sparseness and erroneous reconstructed points with depth, the function Thr(d) should have lower value for closer points and higher values for farther points. We empirically choose such a function (see equation (8)). Its plot is depicted in Figure 4.

$$\text{Thr}(d) = \alpha \left(1 + \log^{\gamma}\left(1 + \frac{d}{\beta} \right) \right) \quad (8)$$
$$\text{where}: \alpha = 300\,\text{mm}, \beta = 2000\,\text{mm}, \gamma = 8$$
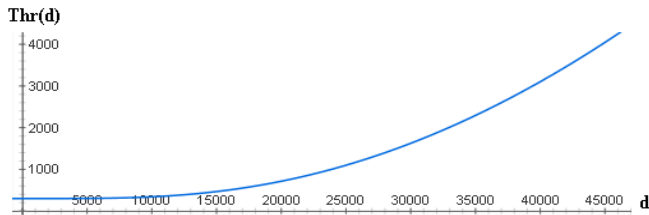

Figure 4. Graph of Thr(d) function with the specified parameters

In Figure 5 each superpixel is labeled as: road surface or beyond driving area or obstacle. Every different superpixels cluster is painted with a different random color.
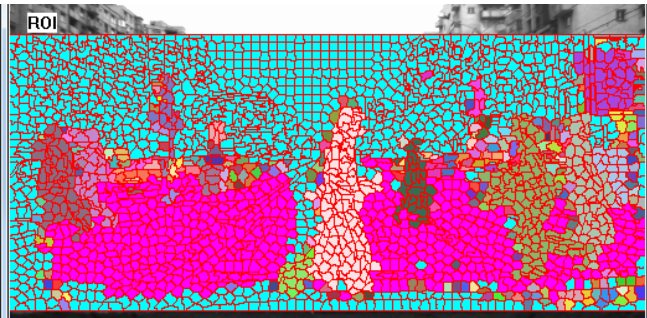

Figure 5. Clusters of superpixels (road surface with magenta color; beyond driving area with cyan color; candidate obstacles with other random colors)

VII. Obstacles Validation and Refinement

Each superpixels cluster represents an obstacle candidate. We have to validate first these candidates in order to determine the real traffic obstacles and then to refine them for setting the final shape.

In Figure 5, besides the road surface and beyond driving area superpixels, there are some spurious obstacle superpixels that appears due to the noise. Usually they appear individual or in clusters with very few members. There are also some clusters that are not lying on the road surface. All these false superpixels clusters should be removed. Every cluster is considered to be a valid obstacle only if all the three following conditions are satisfied:

$$\begin{cases} \left\| ObstacleSuperpixels \right\| > 5 \\ \left\| RoadSuperpixels \right\|_{\substack{neighbors\ of\ obstacle}} \geq 1 \\ MY > 300\,\text{mm}, \end{cases} \quad (9)$$

where MY is the average of $MedY$ for the containing obstacle superpixels:

$$MY = \frac{\displaystyle\sum_{S \in ObstacleSuperpixels} MedY(S)}{\left\| ObstacleSuperpixels \right\|} \quad (10)$$

In Figure 6 the validation of the obstacles superpixels clusters shown in Figure 5 is presented. A number of 8 valid obstacles can be seen.
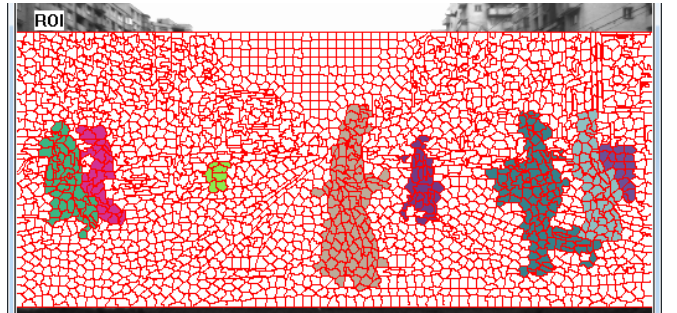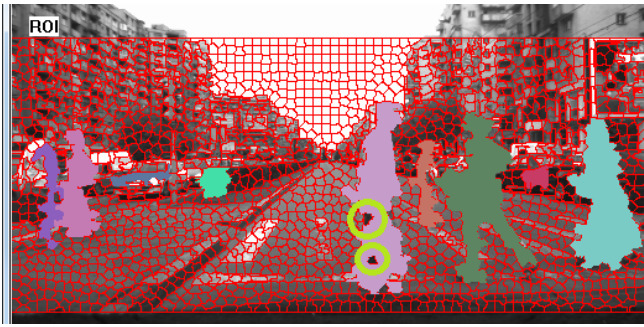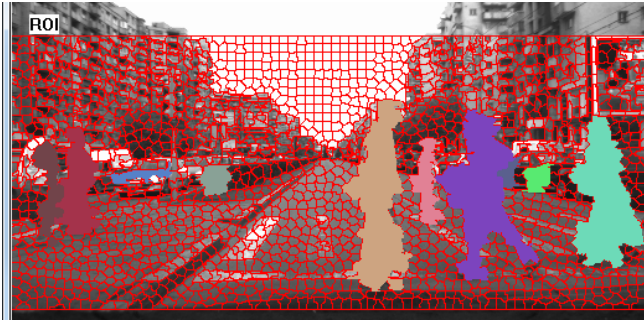

Figure 6. Valid obstacles superpixels (each obstacle with a different random color; background with color white)

There are situations when some singular interior superpixel discontinuities appear on the obstacles surfaces. This is the result of erroneous features computation due to depth points' measurement errors (see Figure 7a)). These small gaps are filled in by a special closing operation. We count the neighbors superpixel cells of each gap that already belongs to a valid obstacle. Then the gap superpixel is assigned to that obstacle that has the maximum number of superpixels in the neighborhood. The assignment is made only in the case when this maximum value is at least 4 superpixels, otherwise it stays still.

The final refinement consists in separating the erroneous grouped multiple obstacles in individual obstacles. This fact might appear when the distance between the obstacles is very small. The refinement is done by using the optical flow information. This is possible only in the case when there is a clear separation based on optical flow vectors.
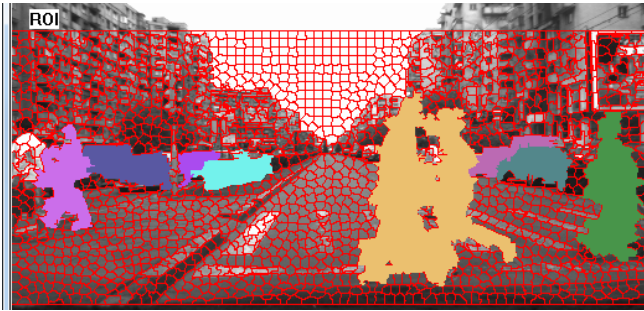
a)

b)

Figure 7. Valid obstacles superpixels: a) containing interior discontinuities – encircled with green; b) after applying the special closing operation

a)

b)

Figure 8. a) Multiple obstacles erroneous grouped in one single obstacle; b) Optical flow vectors – opposite directions for the middle two obstacles

Each optical flow vector *OF* is decomposed on the two perpendicular directions. The magnitude on the horizontal direction is *p* and on the vertical direction is *q*. According to equation (4) we have computed the mean magnitude and orientation of the optical flow vectors in each superpixel cell. A histogram of superpixels *OF* orientation is computed at each obstacle level. We split the orientation range of 0-360 degrees in 30 bins evenly distributed. Each superpixel vote with 1 in the corresponding bin of the mean *OF* angle. If the histogram has more than one local maximum value, we are in

the case when a group of multiple obstacles should be split into separate obstacles. In Figure 9, we have computed the histogram for the middle two obstacles (see Figure 8a) wrongly grouped in one large obstacle. We take a scanning interval of 7 bins centered on every bin location (from 0 to 29). A bin location is considered to be a local maximum if it has the maximum number of votes in the scanning interval bins and it is above the average of the values with 10 votes. The local maxima are marked with red filled circles. A separation (threshold) is set in the middle distance from every two consecutive maximum values. These values are marked with vertical green lines. A multilevel thresholding procedure based on the separation angle values is applied in order to re-label the superpixels. Each obstacle superpixel is re-assigned to a new obstacle determined by the corresponding closest local maximum. The result of optical flow based separation is depicted in Figure 10.
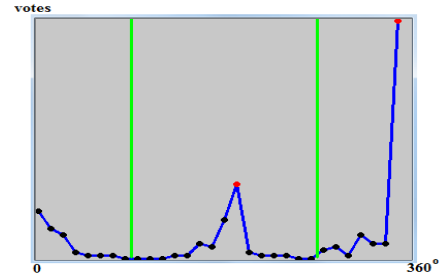

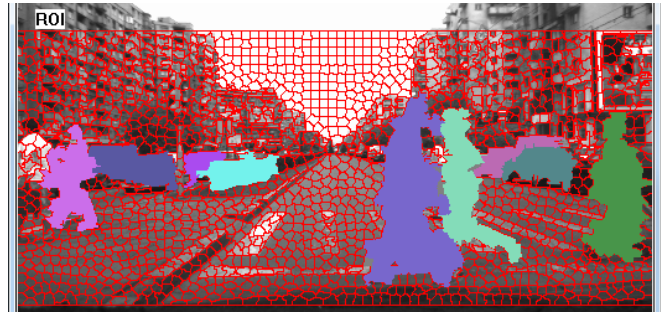Figure 9. Example of the angular *OF* histogram for two separable obstacles

Figure 10. Optical flow based separation of the two middle obstacles

VIII. EXPERIMENTAL RESULTS

In this chapter we present and analyze the superpixels obstacle detection results achieved by using the multi-paradigm of intensity, depth and optical flow information previously described. The method was tested on grayscale video sequences containing thousands of frames from different traffic scenarios.

A qualitative evaluation on the obstacle detection results (see Figure 11) shows that the obstacles in front of the ego-vehicle are very well detected. Miss-detections occur just in cases when there are no or poor reconstructed points on the obstacles surface. This cases are very rare due to the quality of the SORT-SGM stereo-reconstruction. In very crowded traffic scenarios, the obstacles that are very close and have similar intensity, depth and optical flow features might appear as a single obstacle (see Figure 11 bottom).

A quantitative evaluation regarding the accuracy of the superpixel-based obstacle detection approach is presented in TABLE I. We manually labeled a series of ground truth images with precise obstacles surfaces. We checked the surfaces from our detection against the ground truth surfaces

(see Figure 12). The results represents the intersection (common surface) of detected obstacles over the ground truth and vice-versa.

TABLE I. ACCURACY OF SUPERPIXELS BASED OBSTACLE DETECTION

	Ground Truth ∩ Obstacles	Ground Truth	Obstacles
Surface (pixels)	13,546,170	15,724,170	16,562,700
Coverage percentage		86.2%	81.8%

In comparison with another obstacle detection technique based on 3D points grouping and density maps which encapsulate them in bounding cuboids, this approach using superpixels and integrating intensity, depth and optical flow is clearly superior in finding the obstacles and in their shape segmentation (see Figure 13). The system runs real time, at about 20 fps, on an Intel i5 processor @ 3.33 GHz.

IX. CONCLUSION

A novel multi-paradigm that combines intensity, depth and optical flow information was used with SLIC superpixels for achieving accurate obstacle detection. We implemented the SLIC superpixels method on grayscale images, using a custom grouping metric which relays on both pixels position and intensity, obtaining very good results. We proposed also a novel distance metric and algorithm for clustering the superpixels in obstacles. The novel integration of optical flow in the superpixel obstacle detection system clearly refines the superpixels grouping results. Our superpixel-based obstacle detection system shows a superior segmentation than other stereo approaches that use only raw 3D points grouping which is important for other DAS' modules that takes this as an input.
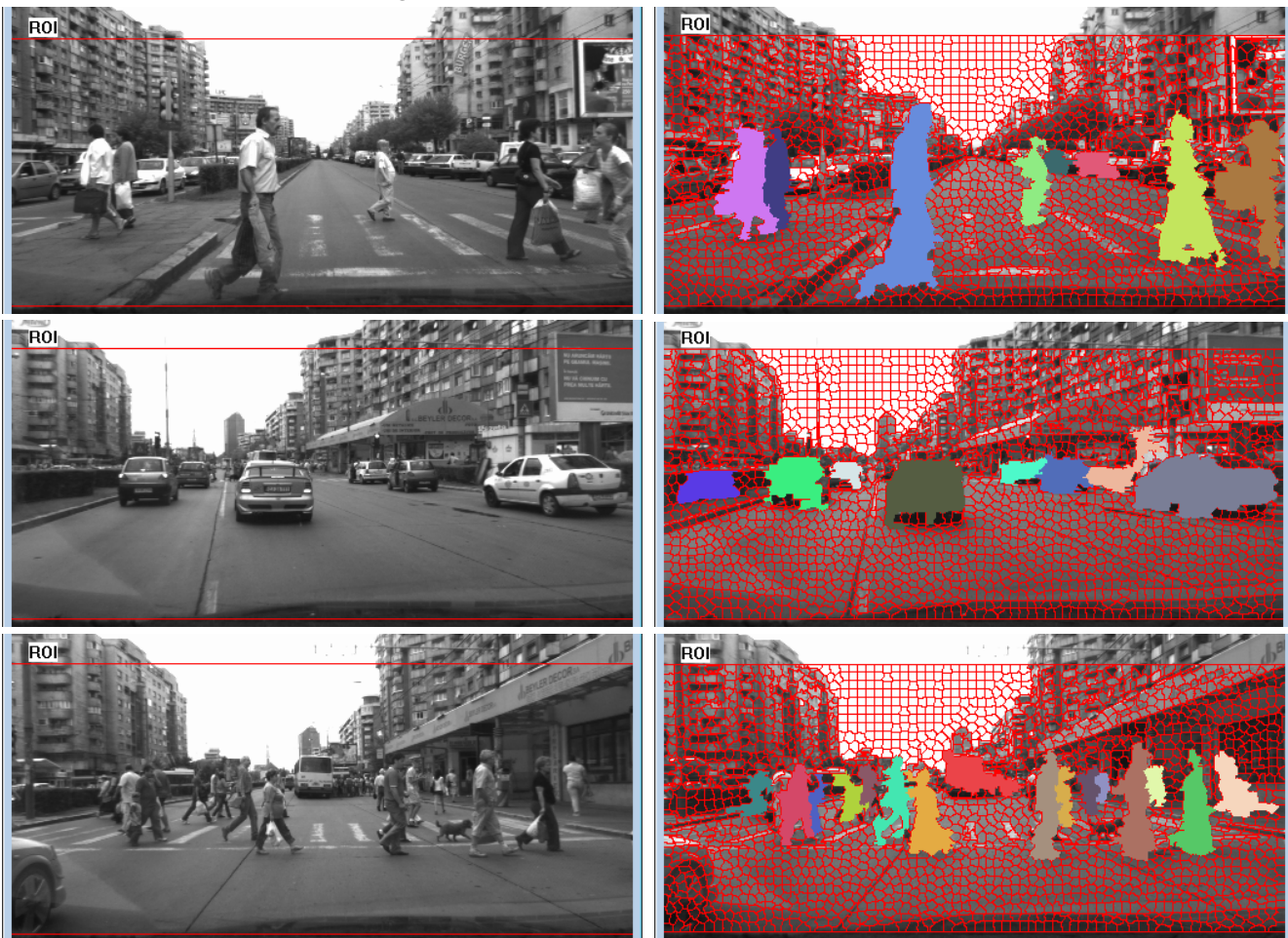

Figure 11. Results of superpixels obstacle detection: left – grayscale scene image; right – superpixels obstacles, each obstacle with a different random color

Figure 12. Evaluation method for one frame: left – obstacle detection result, right – manually labeled ground truth
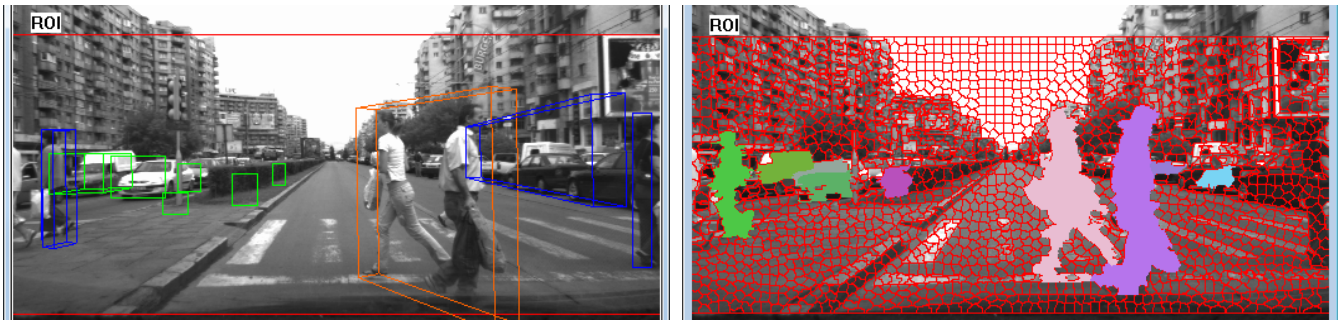
Figure 13. Comparison with other approach: left – obstacle detection based on 3D points grouping in cuboids; right – more accurate superpixels approach

REFERENCES

[1] G. Dong, T. Fraichard, X. Ming, and C. Laugier, "Color modeling by spherical influence field in sensing driving environment," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, 2000, pp. 249-254.

[2] M. Bertozzi, A. Broggi, A. Fascioli, and S. Nichele, "Stereo vision-based vehicle detection," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, 2000, pp. 39-44.

[3] M. Bertozzi and A. Broggi, "GOLD: a parallel real-time stereo vision system for generic obstacle and lane detection," *IEEE Transactions on Image Processing,* vol. 7, pp. 62-81, 1998.

[4] H. Mori and N. M. Charkari, "Shadow and rhythm as sign patterns of obstacle detection," in *IEEE International Symposium on Industrial Electronics*, 1993, pp. 271-277.

[5] M. Heikkila and M. Pietikainen, "A Texture-Based Method for Modeling the Background and Detecting Moving Objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 28, pp. 657-662, 2006.

[6] N. S. Boroujeni, S. A. Etemad, and A. Whitehead, "Fast obstacle detection using targeted optical flow," in *IEEE International Conference on Image Processing (ICIP)*, 2012, pp. 65-68.

[7] D. F. Llorca, M. A. Sotelo, A. M. Hellín, A. Orellana, M. Gavilan, I. G. Daza*, et al.*, "Stereo regions-of-interest selection for pedestrian protection: A survey," *Transportation research part C: emerging technologies,* vol. 25, pp. 226-237, 2012.

[8] S. Nedevschi, R. Danescu, T. Marita, F. Oniga, C. Pocol, S. Sobol*, et al.*, "A Sensor for Urban Driving Assistance Systems Based on Dense Stereovision," in *IEEE Intelligent Vehicles Symposium*, 2007, pp. 276-283.

[9] C. Pocol, S. Nedevschi, and M. A. Obojski, "Obstacle Detection for Mobile Robots, Using Dense Stereo Reconstruction," in *IEEE International Conference on Intelligent Computer Communication and Processing*, 2007, pp. 127-132.

[10] S. Nedevschi, S. Bota, and C. Tomiuc, "Stereo-Based Pedestrian Detection for Collision-Avoidance Applications," *IEEE Transactions on Intelligent Transportation Systems,* vol. 10, pp. 380-391, 2009.

[11] R. Danescu, S. Nedevschi, M. M. Meinecke, and T. Graf, "Stereovision Based Vehicle Tracking in Urban Traffic Environments," in *Intelligent Transportation Systems Conference*, 2007, pp. 400-404.

[12] Z. Xiaolong, Z. Huijing, L. Yiming, Z. Yipu, and Z. Hongbin, "Segmentation and classification of range image from an intelligent vehicle in urban environment," in *International Conference on Intelligent Robots and Systems*, 2010, pp. 1457-1462.

[13] L. You and Y. Ruichek, "Moving objects detection and recognition using sparse spatial information in urban environments," in *IEEE Intelligent Vehicles Symposium*, 2012, pp. 1060-1065.

[14] G. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and Recognition Using Structure from Motion Point Clouds," in *Computer Vision – ECCV 2008*. vol. 5302, D. Forsyth, P. Torr, and A. Zisserman, Eds., ed: Springer Berlin Heidelberg, 2008, pp. 44-57.

[15] S. Kramm and A. Bensrhair, "Obstacle detection using sparse stereovision and clustering techniques," in *IEEE Intelligent Vehicles Symposium*, 2012, pp. 760-765.

[16] U. Franke, C. Rabe, H. Badino, and S. Gehrig, "6D-vision: fusion of stereo and motion for robust environment perception," presented at the Proceedings of the 27th DAGM conference on Pattern Recognition, Vienna, Austria, 2005.

[17] S. Kuthirummal, A. Das, and S. Samarasekera, "A graph traversal based algorithm for obstacle detection using lidar or stereo," in *International Conference on Intelligent Robots and Systems*, 2011, pp. 3874-3880.

[18] K. Huh, J. Park, J. Hwang, and D. Hong, "A stereo vision-based obstacle detection system in vehicles," *Optics and Lasers in Engineering,* vol. 46, pp. 168-178, 2// 2008.

[19] M. Perrollaz, A. Spalanzani, and D. Aubert, "Probabilistic representation of the uncertainty of stereo-vision and application to obstacle detection," in *IEEE Intelligent Vehicles Symposium*, 2010, pp. 313-318.

[20] K. Jungwon and C. Myung-Jin, "Stereo-vision based free space and obstacle detection with structural and traversability analysis using probabilistic volume polar grid map," in *IEEE Conference on Robotics, Automation and Mechatronics*, 2011, pp. 245-251.

[21] Z. Zhen, W. Yifei, J. Brand, and N. Dahnoun, "Real-time obstacle detection based on stereo vision for automotive applications," in *European DSP Education and Research Conference*, 2012, pp. 281-285.

[22] C. D. Pantilie and S. Nedevschi, "SORT-SGM: Subpixel Optimized Real-Time Semiglobal Matching for Intelligent Vehicles," *IEEE Transactions on Vehicular Technology,* vol. 61, pp. 1032-1042, 2012.

[23] J. I. Woodlill, G. Gordon, and R. Buck, "Tyzx DeepSea High Speed Stereo Vision System," in *IEEE Conference on Computer Vision and Pattern Recognition Workshop*, 2004, pp. 41-41.

[24] R. Labayrade, D. Aubert, and J. P. Tarel, "Real time obstacle detection in stereovision on non flat road geometry through "v-disparity" representation," in *IEEE Intelligent Vehicle Symposium*, 2002, pp. 646-651 vol.2.

[25] F. Oniga and S. Nedevschi, "Processing Dense Stereo Data Using Elevation Maps: Road Surface, Traffic Isle, and Obstacle Detection," *IEEE Transactions on Vehicular Technology,* vol. 59, pp. 1172-1182, 2010.

[26] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "SLIC Superpixels Compared to State-of-the-Art Superpixel Methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 34, pp. 2274-2282, 2012.

[27] J. Shi and C. Tomasi, "Good features to track," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1994, pp. 593-600.