Obstacle Localization and Recognition for Autonomous Forklifts using Omnidirectional Stereovision

Arthur D. Costea, Andrei Vatavu and Sergiu Nedevschi

Abstract— **In this paper we propose an approach for obstacle localization and recognition using omnidirectional stereovision applied to autonomous fork-lifts in industrial environments. We use omnidirectional stereovision with two fisheye cameras for the 3D perception of the surrounding environment. Using the reconstructed 3D points, a Digital Elevation Map (DEM) is constructed consisting of a 2.5D grid of elevation cells. Each cell is then classified as ground or obstacle. Further, we use the classified DEM to generate obstacle hypotheses. To ensure a higher detection rate we also propose a fast sliding window based approach relying on the monocular fisheye intensity image. The detections from both approaches are merged and are subjected to a tracking mechanism. Finally each obstacle is classified using boosting over Visual Codebook type features. The classification is refined using the classification history available from tracking. The presented approaches are integrated into a 3D visual perception system for AGVs and are of real time performance.**

I. Introduction

Automated guided vehicles (AGVs) are more and more encountered in industrial environments. The use of AGV fleets has been analyzed in [1-5] and appears to be an efficient solution for modern industrial environments. Their use can play an important role in the efficiency of factory logistics which is still a bottleneck in production and packaging of mass products. They are a flexible, cost effective and safe solution for increasing the automation of factory logistics [5]. Forklift AGVs are able to pick up and deliver autonomously pallets with different type of goods.

In an industrial environment shared by humans and multiple AGVs the surrounding perception by AGVs is crucial. Each AGV has to be able to recognize the surrounding obstacles for path planning and collision avoidance. Most common sensors for AGVs are laser scanners that provide 2D perception of the environment. Stereovision based perception can provide a more complex 3D understanding of the environment and recognition of obstacles.

There are several solutions for stereovision based perception. Some approaches focus on direct processing of each 3D point. For example, in [6] the authors propose a 6D vision approach based on tracking each individual 3D point using a GPU optical flow solution. In order to handle the real-time requirements, various compromise solutions were proposed to reduce the size of the stereo data and to ensure high perception accuracy. These solutions mostly use an intermediate representation for stereo data such as Digital Elevation Maps (DEM) [7], [8] or Occupancy Grids [13], [14]. DEM can be regarded as an improved grid-based representation where, beside the occupancy value, each cell is also described by its height information. Compared to other environment modeling solutions, this type of intermediate representation is more suitable for crowded environments. The resulted compact 2.5D model can be easily used by the subsequent processing steps that need both high accuracy and high performance. For example, in [8] the authors use DEM based representation for determining the traversable terrain and detecting obstacles in off-road scenarios for autonomous ground vehicles.

Fig. 1. Multiple AGVs in a warehouse.

For AGVs in crowded industrial environments (see Fig. 1) there is a high need to perceive the entire surrounding world. Based on our previous work [11], in this paper we propose an omnidirectional stereovision system for the localization and recognition of obstacles. We use fisheye cameras [10] for 360 degree depth perception. The 3D point cloud obtained from stereo reconstruction is transformed into an intermediate digital elevation map based representation [11], [12]. We use the classified digital elevation map to generate obstacle hypotheses. Considering the higher risk of pedestrians, we employ an additional sliding window based approach, trained specifically for pedestrian detection. This way, pedestrians are detected both from stereo and also from monocular vision. Using a list of obstacle hypotheses we track each of the obstacles. For each obstacle we obtain speed and direction. Finally the obstacles are classified as: *pedestrian*, *AGV*, *large obstacle*, *small obstacle*. Tracking allows obtaining a class history for each obstacle that can be further used for classification refinement. An overview of the proposed obstacle recognition process is given in Fig. 2. We have evaluated the proposed obstacle recognition approach

Arthur D. Costea, Andrei Vatavu and Sergiu Nedevschi are with the Image Processing and Pattern Recognition Research Center, Computer Science Department, Technical University of Cluj-Napoca, Romania (e-mail: arthur.costea@cs.utcluj.ro; andrei.vatavu@cs.utcluj.ro; sergiu.nedevschi@cs.utcluj.ro).

Preprint submitted to 2015 IEEE Intelligent Vehicles Symposium.
Received January 29, 2015.

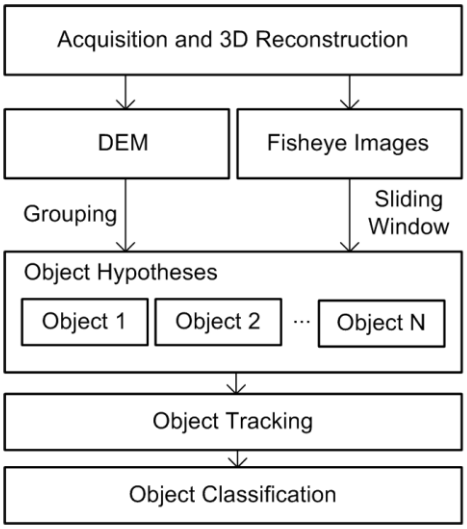

Fig. 2: Object recognition process

in an industrial warehouse environment with multiple AGVs and pedestrians. The approach has been integrated into a perception system and is running in real-time on fork-lift AGVs.

II. OMNIDIRECTIONAL STEREOVISION

In our work we use the omnidirectional stereovision system proposed in [11]. The system uses a pair of fisheye lenses [10] mounted over the fork-lift AGV at a height of 4.5 meters as seen in Fig. 3.a. The cameras are faced downward and provide a horizontal Field Of Vision (FOV) of 360 degrees. The longitudinal FOV (in driving direction) is of 150 degrees and the transversal FOV is of 100 degrees. The area of interest is an ellipsoid cone with the radius of 11 meters and 4 meters.

Multi-channel rectification [11] is used to obtain three rectified image pairs from the fisheye image pair. A GPU based implementation of the stereo reconstruction proposed in [16] is used to obtain a 3D point cloud of the surrounding environment. The 3D points are used to construct a DEM [11], [15] a 2D grid of heights. The classified DEM is obtained by labeling each cell as “ground” or “obstacle” and is used to generate a set of obstacle hypotheses.

III. OBSTACLE DETECTION

We use two approaches to generate obstacle hypotheses. The first technique consists in the grouping of DEM obstacle cells into connected blobs (clusters). The second one is a fast sliding window based approach using monocular vision. Both methods generate a list of object candidates, each hypothesis being described by two models: a 3D cuboid and a free-form polygonal model. The 3D boxes are used for defining the region of interest for the object classification step, while the free-form polygons are used in the tracking stage.

A. *Obstacle Detection from Classified DEM*

At this stage, the DEM cells classified as “object” are grouped into individual clusters. The grouping process is performed in 2D by using the projection of the Elevation Map cells on the ground plane. The spatial proximity criterion between DEM cells is used to determine the connected entities. For each separate group of cells an oriented bounding box is determined.

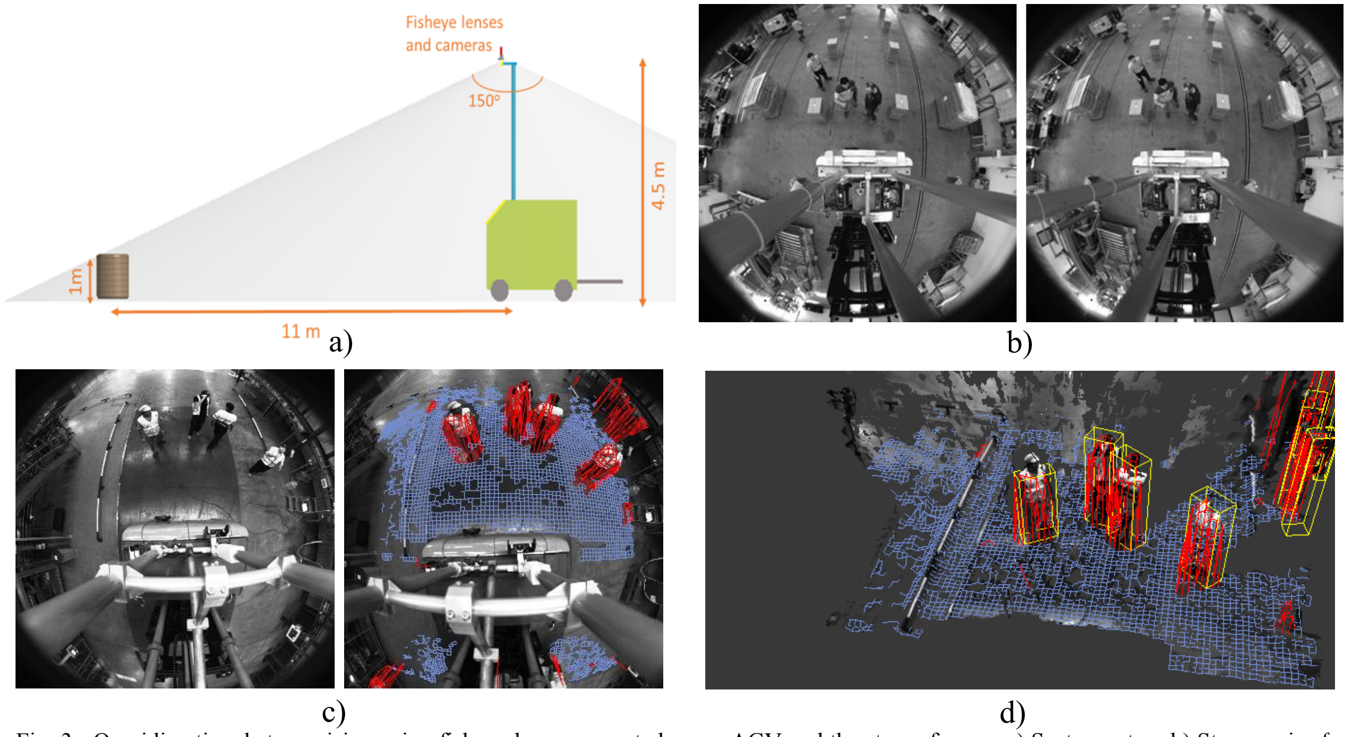

Fig. 3: Omnidirectional stereovision using fisheye lenses mounted on an AGV and the stereo frames. a) System setup. b) Stereo pair of fisheye images. c) Left grayscale image and the elevation map projected on the left image. d) The 3D view representation including the elevation map. The points are classified as Ground (blue) or Object (red).

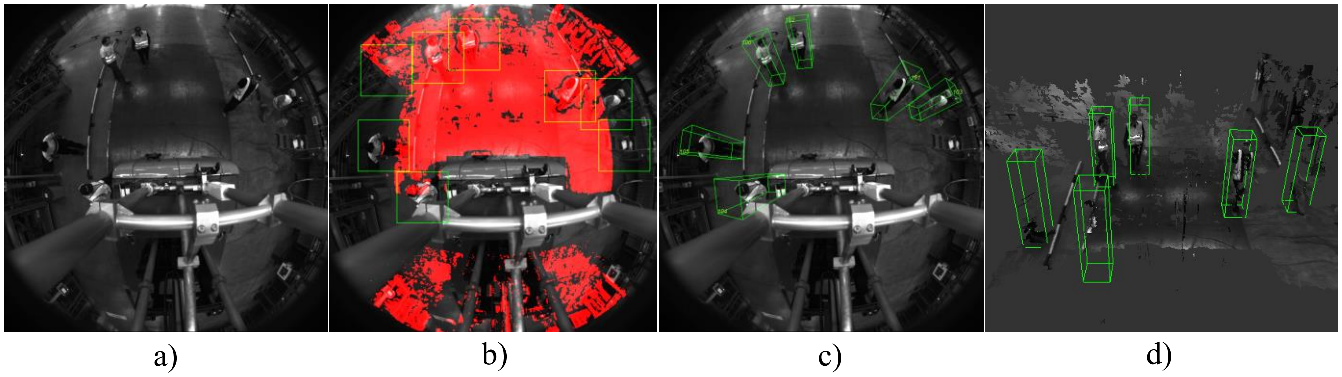

Fig. 4: 2D Sliding window based 3D cuboid estimation: a) Left image; b) 2D sliding window detection results (green) and the reconstructed 3D points projected on the left image (red); c) Estimated 3D cuboids in the left image; d) 3D cuboids in 3D space

B. Sliding window over fisheye image

Considering the higher importance of pedestrians due to their vulnerability we use another approach for their detection. This approach is based on monocular vision and is independent from the stereo based elevation map. We use a sliding window directly over the fisheye intensity image in order to detect pedestrians. We observed that the size of pedestrians changes only slightly over the distance from camera. However, the orientation of the pedestrian changes with its position relative to the camera. Considering that the pedestrian orientations are symmetrical with respect to the image center, a solution would be to rotate each individual detection window before classification, but at a high computation cost. A more efficient solution is to train a classifier with a larger training dataset, consisting of pedestrians at multiple orientations. For the training dataset we used scenarios with multiple pedestrians walking around the AGV at different distances and orientations. We applied additional rotations over the extracted pedestrians to extend the dataset.

In our experiments we used a 80 x 80 pixel size detection window. The image is scanned densely with a step rate of 4 pixels over the fisheye frame. The fisheye frame is resized to 512 x 512 pixels. Due to the small scale change of pedestrians in the fisheye image, we are able to use only a single detection window scale. In order to classify a detection window as pedestrian or non-pedestrian, we follow the Aggregated Channel Features (ACF) based detection approach proposed by Dollar et al. in [18], 8 image channels are computed for the fisheye intensity frame: one channel for grayscale intensity, one for gradient magnitude and 6 for oriented gradient magnitudes (using six orientations). 8 aggregate channels are obtained by computing an average for 4 x 4 pixel cells. The resulting channels have a size of 128 x 128 pixels. The classification of the sliding windows is achieved using the aggregated channel features with a boosting classifier. The boosting classifier uses 2048 two level decision trees.

For each 2D detection from the fisheye frame we estimate a 3D cuboid. We map each reconstructed 3D point to the fisheye frame and compute a median for the points that are mapped in the 10 x 10 pixel region in the center of the detection window. The median 3D point is projected on the ground plane and a 0.5 x 0.5 x 2.0 meter cuboid is generated in 3D over that point, as seen in Fig.4. These obstacles, together with the previous obstacles, are merged into a single obstacle list.

C. Obstacle Representation with Attributed Polygonal Models

For each object hypothesis described by a 3D cuboid , we also extract an attributed polygonal model (see Fig. 5) which provides a better object approximate with a small subset of points. The free-form polylines are computed by adapting the BorderScanner algorithm previously developed in [9]. The main idea is to extract an object model by selecting the most visible (not occluded) parts from the camera position. This is achieved by using a scanning axis which extends from the observation point and moves in a radial direction with fixed

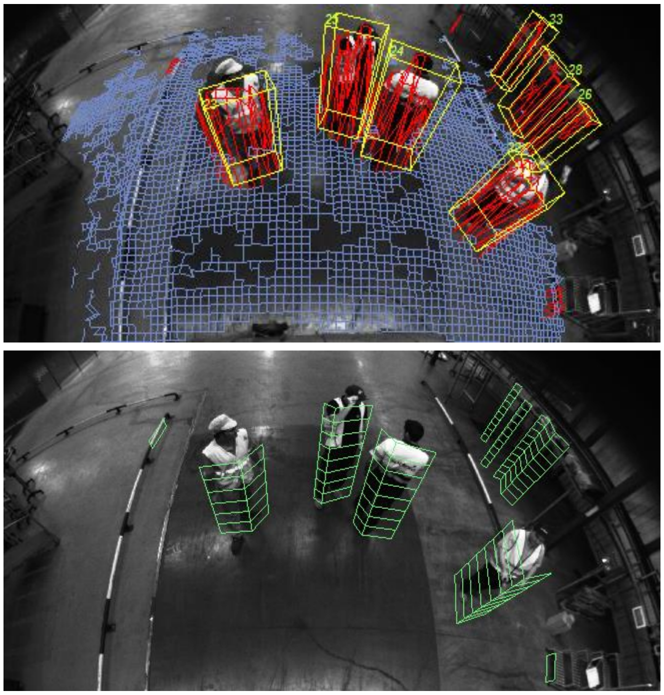

Fig. 5: Top: detected objects represented as 3D oriented boxes. Bottom: the free-form polygonal representation.

steps. At each step, the most visible cell that is classified as object is marked as a delimiter cell. The extracted contours are used to compute polygonal structures so that each individual DEM cluster is described by a separate free-form polygonal representation (see Fig. 5, bottom).

IV. Obstacle tracking

The object tracking (see Fig. 6) consists in estimating, recursively in time, the object state given all observations up to the present time *t*. The tracking technique can be divided into two separate stages: *motion estimation* and *filtering*. The motion estimation approach takes into consideration the previously extracted delimiters. For each associated contour pair that identifies the same object in the consecutive frames we compute an optimal translation *T* and rotation *R* that minimize the alignment error. For this we use the Iterative Closest Point (ICP) approach, previously presented in [17]. According to the ICP technique, each obstacle can be described by two set of points: a model set $\{p_1,p_2, ..., p_M\}$ that defines the obstacle contour in the previous frame, and a data set $\{q_1,q_2, ..., q_K\}$ that defines the obstacle contour in the current frame. The optimal transformation is estimated by minimizing the following objective function:

$$\varepsilon(R,T)=\sum_{i=1}^{N}\left\|Rp_i+T-q_i\right\|^2 \tag{1}$$

where N represents the number of point-to-point correspondences (p_i ,q_i). In order to stabilize the results, the object positions and the extracted speed vectors are subjected to a standard Kalman filtering technique.

V. Obstacle classification

Having a list of tracked obstacles, our goal is to classify each of them based on visual features. We use three main classes: Pedestrian, AGV, Other obstacle. The obstacles are represented as 3D cuboids. We have to obtain a 2D image in order to compute visual features. This is obtained by projecting the 3D cuboid into the fisheye frame and cropping

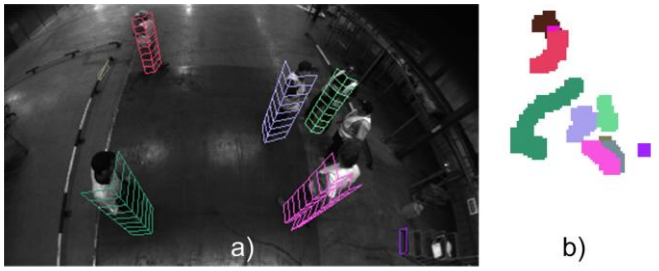

Fig. 6: Object tracking. a) Each object is labeled with a unique ID (different color). b) The trajectories of the dynamic obstacles (top view).

it out as a rectangular image. Considering the nature of the fisheye image, the image is radially symmetrical with respect to the image center, as it can be seen in Fig. 7. The obstacles that are exactly in the front of the AGV, are oriented upwards. We compute the angle with longitudinal axis as illustrated in Fig. 7.a and rotate the obstacle with this angle. This way the obstacles will be always oriented upwards (see Fig. 7.b). In order to obtain scale invariance, the 2D images of the obstacles are resized to have a fixed height of 100 pixels if the height is greater than the width or to a fixed width of 100 pixels otherwise. The aspect ratio is not changed during the resizing. We use this image for computing visual codebook based features.

A. Classification feature computation

Visual codebook or bag of features based classification is a popular approach used for general image classification [19], [20] or image segmentation (pixel classification) [21]. In our previous works [22] and [23] we managed to adapt their use for real-time applications.

We employ a HOG type local descriptor that we used also in [22] and [23]. The local descriptor can be computed in any position of the image and describes a 16 x 16 pixel neighborhood with a descriptor vector. The pixel neighborhood is partitioned into 4 cells of 4 x 4 pixels. For

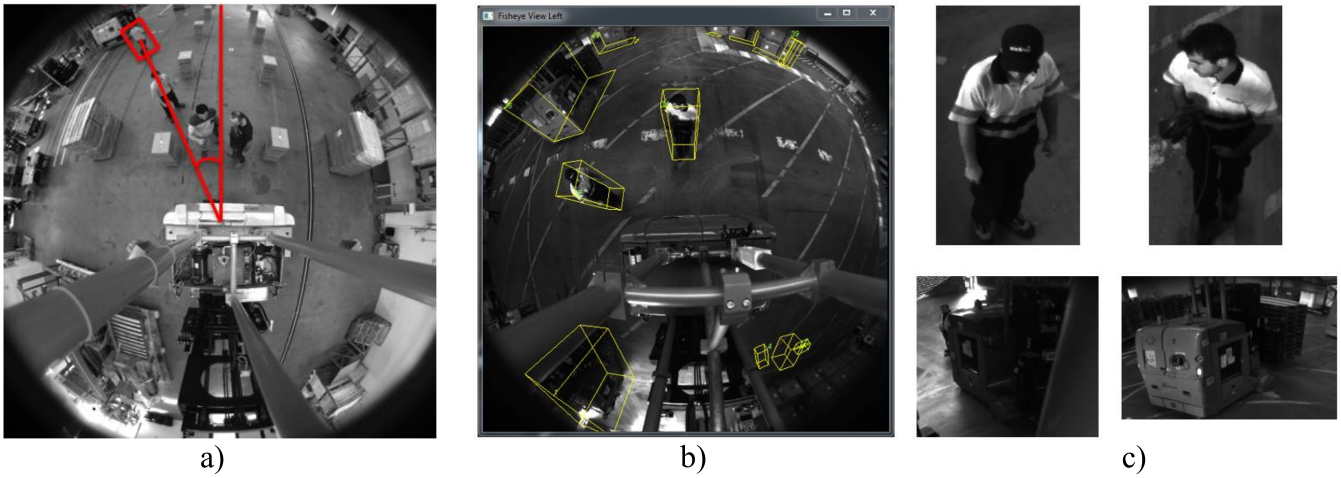

Fig. 7: 2D image generation for a 3D cuboid: a) obstacle angle with respect to the longitudinal axis; b) 3D cuboids; c) resulting 2D images after cropping and rotation.

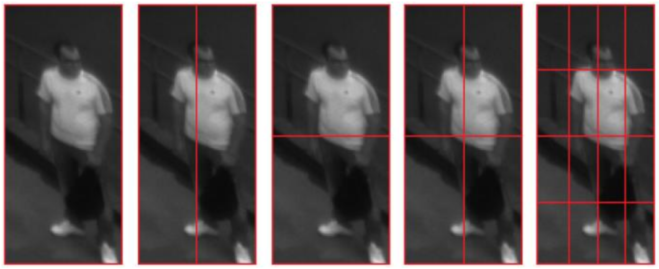

Fig. 8: The 25 image regions used for histograms of visual words.

each of these cells we compute a histogram of oriented gradients using 6 orientations (an increment of 30 degrees). The 4 histograms are concatenated and result in a 24 dimensional descriptor vector.

We select a training set with obstacle images. Around 1000000 descriptor vectors are sampled randomly. We apply K-means clustering over the samples with K=100. The 100 resulting centroids build up the visual codebook.

After training a codebook, any local descriptor vector can be matched to the closest visual word (centroid) from the codebook in Euclidean distance. In order to extract the classification features for an obstacle image, we compute the local descriptors densely, at each pixel position. Each of the computed local descriptors is matched to the closest codebook word. We propose 25 image regions obtained from 5 different partitionings: 1x1, 1x2, 2x1, 2x2 and 4x4 (see Fig. 8). We use as classification features the histogram of visual words in those regions. 25 regions and codebook of 100 words result in 2500 features for an obstacle.

B. *Obstacle classification*

We use two binary classifiers for AGVs and pedestrians. The obstacles that are not classified as AGV or pedestrian are classified as "large other obstacle" if the obstacle is higher than 50 cm or "small other obstacle" otherwise. We need to train each binary classifier with a training dataset consisting of positive and negative samples using the previously defined 2500 classification features. We use Ada-boost [24] with 2048 rounds and two level decision trees as weak learners. Using the learned classifier models any obstacle can be classified based on the visual codebook based features. If an obstacle is classified positively by both classifiers, then we choose the class label with the highest probability estimate.

Due to obstacle tracking we have access to a classification history. If an object is tracked for more than 5 frames, we take into consideration the last 5 classifications and apply majority voting. This way we are able to filter out temporary false classifications.

C. *Detection fusion*

We have proposed two different detection approaches for proposing obstacles hypotheses. Pedestrians can be detected by both approaches. We consider that two 3D cuboids are overlapping if the ratio between the intersection volume and the unified volume is more than 50%. In case of overlapping detections we retain the obstacle with the highest classification probability.

VI. Experimental Results

The proposed solution was implemented and integrated into a visual perception system in the framework of FP7 EU PAN-Robots project [25]. The system runs at 10 frames/second on a GPU equipped industrial PC that was mounted on a forklift AGV. Fig. 9 illustrates some obstacle classification results in industrial warehouse environments.

In order to train and evaluate the obstacle classifiers we created an obstacle database consisting of 2287 pedestrians, 454 AGVs and over 20000 other obstacles. We used 75% for training and 25% for evaluation. In Table I we provide the classification accuracy and precision for each of the classes.

The DEM based detection approach is a generic obstacle detection technique based on grouping cells classified as objects, while the sliding window based approach focuses only on pedestrians. For being able to evaluate the recognition rates of each individual obstacle detection approach, as well as to compare them with the proposed combined solution we setup the following scenario. A ground truth sequence was created with a high number of pedestrians including different challenging situations. The scenario includes dynamic pedestrians, grouped pedestrians, different occlusions cases and all seen from different distances and angles, from standing or moving AGV. Over 2000 frames were manually annotated resulting in 7793 ground truth pedestrians. Table II show the recall and precision rate for each approach. It can be noticed that the proposed combined solution in this work provides a significant increase in detection rate.

TABLE I
Obstacle Classification Evaluation

Class	Recall	Precision
Pedestrian	95 %	94 %
AGV	85 %	80 %
Other	92 %	93 %

TABLE II
Detection Rate Evaluation

Approach	Recall	Precision
Sliding Window	78 %	96 %
DEM	85 %	97 %
DEM + Sliding Window	93 %	94 %

VII. Conclusions

In this paper we proposed an omnidirectional stereovision system for the localization and recognition of obstacles. In order to perceive the surrounding world, the presented solution employs fisheye cameras for 360 degree depth perception. First, the reconstructed dense stereo data is mapped into an intermediate classified elevation map. Then, a set of obstacle hypotheses are generated by grouping the object cells from the elevation map. Taking into account the higher risk of pedestrians, an additional sliding window based solution is used to detect in particular pedestrians. The generated list of candidates by both approaches is subjected to a tracking mechanism. Finally, the resulted set of trackers is classified as: *pedestrian*, *AGV*, *large obstacle*. The proposed combined approach in this work proves to be efficient in terms of detection accuracy and precision.

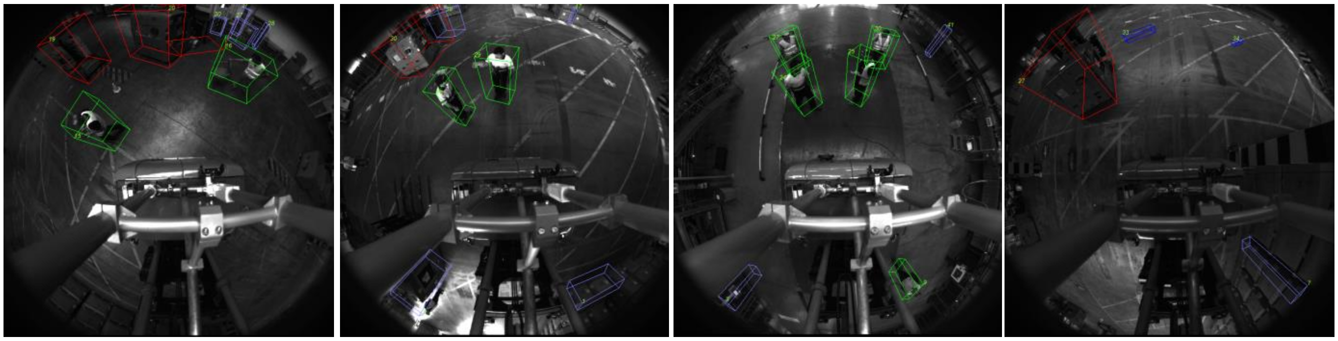

Fig. 9: Obstacle classification in different scenarios: Red – AGV, Green – Pedestrian, Blue – Other obstacle

ACKNOWLEDGMENT

This work was supported by the research project PAN-Robots. PAN-Robots is funded by the European Commission, under the 7th Framework Programme Grant Agreement n. 314193. The partners of the consortium thank the European Commission for supporting the work of this project.

REFERENCES

[1] F. Oleari, M. Magnani, D. Ronzoni, and L. Sabattini, "Industrial AGVs: Toward a pervasive diffusion in modern factory warehouses," in *IEEE International Conference on Intelligent Computer Communication and Processing*, Cluj-Napoca, Romania, 2014, pp.233-238.

[2] V. Digani, F. Caramaschi, L. Sabattini, C. Secchi, and C. Fantuzzi, "Obstacle avoidance for industrial AGVs," in *IEEE International Conference on Intelligent Computer Communication and Processing*, Cluj-Napoca, Romania, 2014, pp. 227-232.

[3] L. Schulze and L. Zhao, "Worldwide development and application of automated guided vehicle systems," in *International Journal of Services Operations and Informatics*, vol. 2, no. 2, p. 164, 2007.

[4] I. F. Vis, "Survey of research in the design and control of automated guided vehicle systems," in *European Journal of Operational Research*, vol. 170, no. 3, pp. 677–709, May 2006

[5] L. Sabattini, V. Digani, C. Secchi, G. Cotena, D. Ronzoni, M. Foppoli, and F. Oleari, "Technological roadmap to boost the introduction of agvs in industrial applications," in *IEEE International Conference on Intelligent Computer Communication and Processing*, Cluj-Napoca, Romania, 2013, pp. 203–208.

[6] U. Franke, C. Rabe, H. Badino, and S. Gehrig, "6d-vision: Fusion of stereo and motion for robust environment perception," in *DAGM Symposium*, 2005, pp. 216-223.

[7] F. Oniga and S. Nedevschi, "Processing Dense Stereo Data Using Elevation Maps: Road Surface, Traffic Isle and Obstacle Detection," in *IEEE Transactions on Vehicular Technologies*, vol. 59, no. 3, pp. 1172-1182, 2010.

[8] A. Broggi, E. Cardarelli, S. Cattani, and M. Sabbatelli, "Terrain mapping for off-road Autonomous Ground Vehicles using rational B-Spline surfaces and stereo vision," in *IEEE Intelligent Vehicles Symposium*, Gold Coast, Australia, 2013, pp. 648-653.

[9] A. Vatavu, Sergiu Nedevschi, and Florin Oniga, "Real Time Object Delimiters Extraction for Environment Representation in Driving Scenarios," in *International Conference on Informatics in Control, Automation and Robotics*, Milano, Italy, 2009, pp 86-93.

[10] A. Mäyrä, M. Aikio, and M. Kumpulainen, "Fisheye optics for omnidirectional perception," in *IEEE International Conference on Intelligent Computer Communication and Processing*, Cluj-Napoca, Romania, 2014, pp. 259-263.

[11] M. Drulea, I. Szakats, A. Vatavu, and S. Nedevschi, "Omnidirectional stereo vision using fisheye lenses," in *IEEE International Conference on Intelligent Computer Communication and Processing*, Cluj-Napoca, Romania, 2014, pp. 251-258.

[12] S. Mandici and S. Nedevschi, "Aggregate Road Surface based Environment Representation using Digital Elevation Maps," in *IEEE International Conference on Intelligent Computer Communication and Processing*, Cluj-Napoca, Romania, 2014, pp. 149-156.

[13] T. N. Nguyen, B. Michaelis, A. Al-Hamadi, M. Tornow, and M. M. Meinecke, "Stereo-Camera-Based Urban Environment Perception Using Occupancy Grid and Object Tracking," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 154-165, March 2012.

[14] H. Lategahn. W. Derendarz, T. Graf, B. Kitt, and J. Effertz, "Occupancy grid computation from dense stereo and sparse structure and motion points for automotive applications," in *IEEE Intelligent Vehicles Symposium*, San Diego, CA, USA, 2010, pp. 819-824.

[15] S. Mandici and S. Nedevschi, "Aggregate Road Surface based Environment Representation using Digital Elevation Maps," in *IEEE International Conference on Intelligent Computer Communication and Processing*, Cluj-Napoca, Romania, 2014, pp. 149-156.

[16] I. Haller, C. Pantilie, F. Oniga, and S. Nedevschi, "Real-time semi-global dense stereo solution with improved sub-pixel accuracy", in *IEEE Intelligent Vehicles Symposium*, San Diego, California, USA, 2010, pp. 369–376.

[17] A. Vatavu and S. Nedevschi, "Real-time modeling of dynamic environments in traffic scenarios using a stereo-vision system," in *IEEE International Conference on Intelligent Transportation Systems*, Anchorage, Alaska, USA, 2012, pp.722-727.

[18] P. Dollar, R. Appel, S. Belongie, and P. Perona, "Fast Feature Pyramids for Object Detection," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1532-1545, Aug., 2014.

[19] G. Csurka, C.R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual Categorization with Bags of Keypoints," in *IEEE European Conference on Computer Vision*, 2004, pp. 1-22.

[20] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2006, pp. 2169 - 2178.

[21] L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr, "Associative Hierarchical Random Fields," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 6, pp. 1056-1077, June 2013.

[22] A. Costea and S. Nedevschi, "Word Channel Based Multiscale Pedestrian Detection Without Image Resizing and Using Only One Classifier," in *IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, Ohio, USA, 2014, pp. 2393-2400.

[23] A. Costea and S. Nedevschi, "Multi-class segmentation for traffic scenarios at over 50 FPS," in *IEEE Intelligent Vehicles Symposium*, Dearborn, Michigan, USA, 2014, pp. 1390-1395.

[24] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting," in *The Annals of Statistics*, vol. 38, no. 2, pp. 337–374, 2000.

[25] PAN-Robots - Plug And Navigate ROBOTS for smart factories, FP7 EU Project. [online] http://www.pan-robots.eu/