

Modeling and Tracking of Dynamic Obstacles for Logistic Plants using Omnidirectional Stereo Vision

Andrei Vatavu, Arthur D. Costea, and Sergiu Nedevschi, *Members, IEEE*

Abstract— In this work we present an obstacle detection and tracking solution applied to Automated Guided Vehicles (AGVs) in industrial environments. The proposed method relies on information provided by an omnidirectional stereo vision system enabling 360 degree perception around the AGV. The stereo data is transformed into a classified digital elevation map (DEM). Based on this intermediate representation we are able to generate a set of obstacle hypotheses, each represented by a 3D cuboid and a free-form polygonal model. The cuboidal model is used for the classification of each hypothesis as “Pedestrian”, “AGV”, “Large Obstacle” or “Small Obstacle”, while the free-form polylines are used for object motion estimation relying on an Iterative Closest Point (ICP) method. The obtained measurements are subjected to a Kalman filter based tracking approach, in which the data association takes into account also the classification results.

I. INTRODUCTION

Today’s modern factories deal with two main type of activities: product processing and logistic operations. Logistic operations include the transportation of products or raw materials to production lines, storage areas or shipment points. Despite the fact that the automation of product processing reached a high level of efficiency, logistic management is still marginal. Automated logistic operations can be carried out using a fleet of Automated Guided Vehicles (AGVs) and such solutions have been already described and analyzed in [2-4]. An evaluation of AGVs with different degrees of autonomy is provided in [5]. Considering the high number of AGVs working in a dynamical industrial environment, traffic management becomes a key aspect. Centralized and decentralized control strategies, such as [6-8], can be used for optimal coordination of AGVs.

The main purposes of AGVs are to offer a time efficient, cost effective, safe, green and less error-prone solution for factory logistic management. Using autonomous load handling systems forklift AGVs are able to work with various types of goods. For safe autonomous navigation and interaction with a dynamic environment, AGVs need to perceive the surroundings as well as to detect and track relevant obstacles such as other AGVs or pedestrians. Laser scanners are the most common perception sensors for AGVs. They offer a 2D perception of AGVs surroundings and can be used for navigation and obstacle avoidance [9-11]. If an



Figure 1. Automated warehouse environment

AGV detects an obstacle in the moving direction, it can apply automated braking or an avoidance maneuver.

Visual perception is an alternative to laser scanner based perception for mobile robots. It can be used in a similar manner for autonomous navigation and obstacle avoidance [12]. Vision based perception is a common solution in the automotive industry for advanced driving assistance systems or autonomous driving in traffic environments. The monocular or stereo cameras are mounted behind the windshield and the field of view is only in the driving direction. A review on vision based detection, tracking and behavior analysis approaches is provided in [13]. In the case of mobile robots there is higher interest in the perception of the surrounding environment in all directions. The use of omnidirectional cameras can allow a 360 degree visual perception and have been used for navigation on mobile robots on ground [14], [15] and also on micro-aerial vehicles [16]. Using a pair of omnidirectional cameras it is possible to achieve omnidirectional stereo vision that allows depth computation and a more complex 3D perception of the surrounding environment [17], [18].

Besides the surrounding world perception, an AGV system should be able to detect and also track the state of all relevant obstacles in real time and with high confidence. The motion information of obstacles allows better understanding of a dynamic environment and a more efficient risk assessment. The tracking results can be used as additional information for collision avoidance and path planning.

Usually, object tracking can be decomposed into three main steps: measurement extraction, data association and object state estimation. Various solutions have been proposed in literature. Some of them rely on directly tracking 3D depth data [19], [20] while other approaches try to transform the high volume of information into intermediate

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

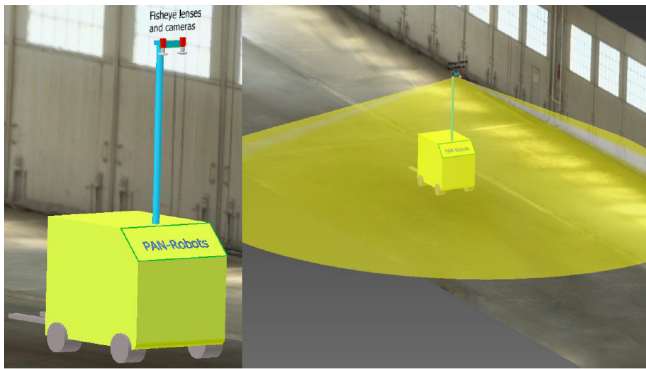
representations such as occupancy grids [21], [22], octree-based data structures [23], [26] or digital elevation maps [24], [25]. For example, in [26] the data provided by a lidar sensor is mapped using octree-based representations. The moving objects are detected from inconsistencies between different scans. In [25] a particle filter-based solution is used to track position and speed of each cell in a dynamic elevation map.

In order to reduce the processing time, various approaches rely on extracting and tracking high-level geometric models such as 2D or 3D bounding boxes [29], contours [27] or free-form polygons [28].

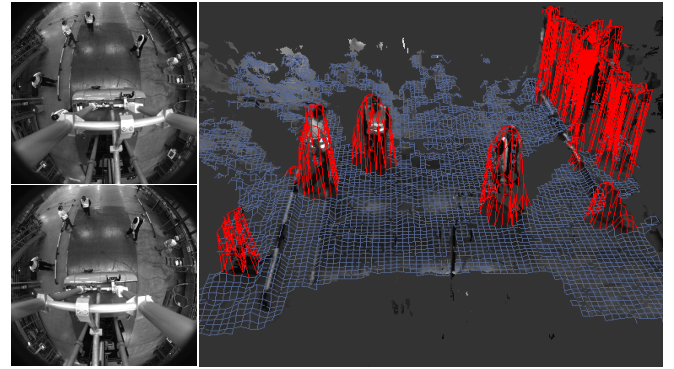
Although the use of more compact representations can provide better processing costs, sometimes it is not enough to achieve a robust tracking mechanism. As a solution, some approaches try to combine the geometric properties with color information [30], while other methods apply additional vision-based recognition steps in order to increase the robustness of the data association and object tracking [31], [32].

In this work we propose a solution for detecting and tracking obstacles in industrial environments for AGVs. In order to cover the entire surrounding of the AGV, we use an omnidirectional stereo vision based perception system. The employed fisheye cameras enable a 360 degree perception of the AGV's environment. The stereo data is transformed into a more compact and more practical representation mode in the form of a classified digital elevation map (DEM). The DEM is used to generate obstacle hypotheses, each represented as a cuboidal and a free form polygonal model. The cuboidal model is used for the classification of each hypothesis as "Pedestrian", "AGV", "Large Obstacle" or "Small Obstacle", while the free form polygonal model is used for estimating the object motion based on an Iterative Closest Point (ICP) approach. The obtained measurements are subjected to a Kalman filter based tracking solution, in which the data association takes into account also the classification results.

The proposed solution was developed in the framework of the PAN-Robots FP7 EU project [1] for obstacle detection, classification and tracking by AGVs in a warehouse environment.



(a)



(b)

(c)

Figure 3. (a) The omnidirectional stereo vision system. (b) Left and right fisheye images. (c) The classified digital elevation map (3D View) obtained from stereo fisheye images.

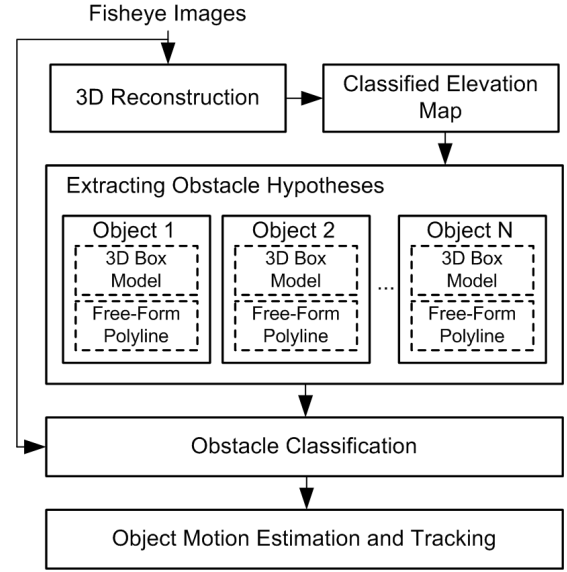


Figure 2. System Overview

II. ENVIRONMENT PERCEPTION AND INTERMEDIATE REPRESENTATION

In order to detect and track the obstacles around the AGV, first the surrounding environment has to be perceived, then it has to be represented. We employ the omnidirectional stereo vision system proposed in our previous work [18]. Two fisheye cameras [38] are used to obtain a 360 degree stereo perception around the AGV. The cameras are mounted on a rigid rig at a height of 4.5 meters over the AGV and are oriented downwards as illustrated in Fig. 3a. The image stability to vibrations and oscillations is ensured by the fact that the ground in a warehouse is flat, the AGV speed is low (about 3m/s), while the frame rate is 20 fps.

The fisheye image pairs (Fig. 3b) are decomposed into 3 rectified image pairs using the proposed multi-channel rectification approach [18]. The GPU accelerated stereo matching algorithm proposed in [40] is used to achieve 360 degree depth perception. The reconstructed 3D points are used to detect the ground plane and to build a digital elevation map (DEM), consisting of a 2D grid of cells with estimated heights. The size of a single cell is of 10x10cm and each cell is classified as "ground" or "obstacle" [18],

[39]. The classified DEM (see Fig. 3c) is used as a representation form for the surrounding 3D environment for further processing.

III. OBSTACLE DETECTION

The obstacle detection module consists in extracting a set of object hypotheses. First, the DEM cells are clustered into connected entities, called blobs. Then, for each individual blob, we extract a 3D bounding box and a free-form polygonal representation. Thus, each object hypotheses is defined by two separate models. The 3D cuboids are used to select the regions of interests for obstacle classification, while the polygonal models are used to extract the object motion by applying a fast Iterative Closest Point (ICP) alignment solution. The motivation of using polylines instead of more simplistic 3D boxes for the obstacle tracking comes from the need to improve the motion estimation accuracy. Thus, the ICP-based matching process relies on a set of free-form models that are able to approximate the real obstacle shape with a number of points as small as possible.

A. Extracting 3D Bounding Boxes

In order to compute the object blobs, the DEM cells are clustered based on a proximity criterion. The connected sets which contain a number of cells smaller than a given threshold are considered noise and are filtered. For each individual set of grouped DEM cells, a 3D oriented box is computed (see Fig. 4, top). The resulted cuboid model is described by its center of mass $P_c(X_c, Y_c, Z_c)$, width W , length L , height H and an orientation θ in the horizontal plane.

B. Extracting Free-From Polygonal Models

In addition to the cuboidal model, for each object candidate we compute a free-form polygonal representation (see Fig. 4, bottom). As the resulted polylines are able to better approximate the real shape of the obstacle, we use these models to determine the object motion by applying an ICP-based matching solution. For extracting the object delimiters, we use the Border Scanner algorithm, previously introduced in [33]. The basic idea of this approach is to collect the most visible obstacle points along virtual rays which extend from the origin position in the radial directions. Subsequently, the resulted contours are transformed into polylines. In order to avoid overlapping sub-problems (the cases when the same grid cell is traversed more than once) and to minimize the processing cost, an improved Border Scanner solution was proposed in [34]. Instead of re-computing the scanning axes at each frame, a predefined path structure is used to direct the searching process through the DEM space. This predefined map is called *Policy Tree* and is generated only once, in the initialization step. Thus, for accumulating the closest obstacle cells, a depth first search strategy is used. Compared to the previous Border Scanner variants, used in the context of driving assistance applications, the proposed solution includes two main differences. First, the obstacle delimiters are extracted by exploring the DEM grid corresponding to the entire area around the AGV. Second, unlike the previous

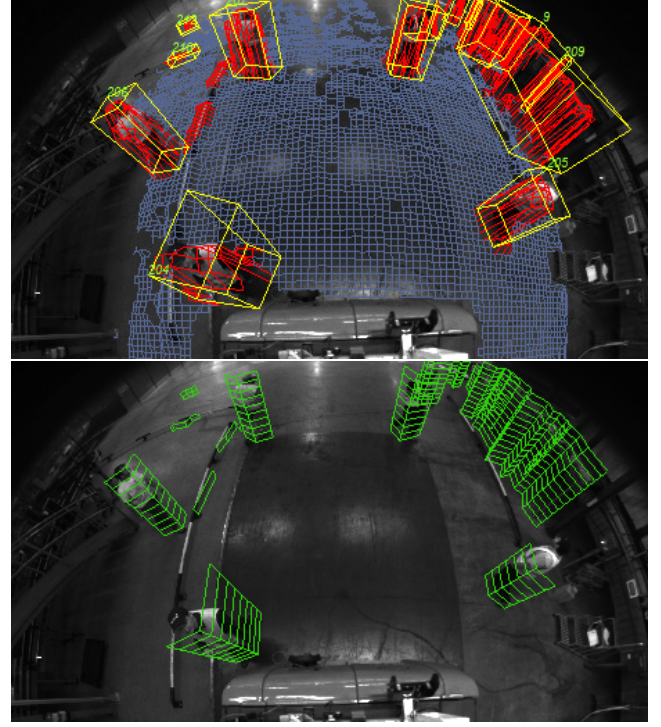


Figure 4. Obstacle representation (a region of interest). Top: oriented 3D cuboids. Bottom: free-form polylines.

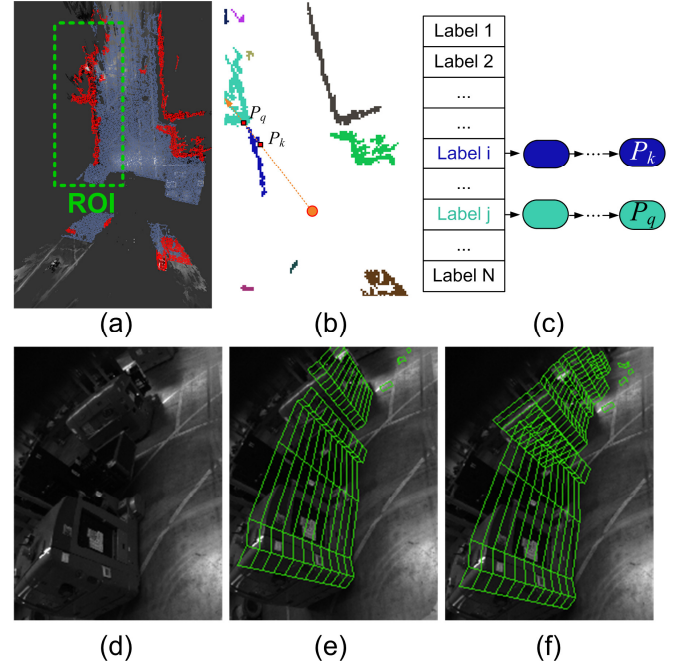


Figure 5. Extracting obstacle delimiters - an illustrative example. (a) The classified DEM (top view). The object cells are shown with red. The ground plane cells are highlighted with blue. (b) The DEM cells are grouped into individual clusters (blobs). The example illustrates how the delimiter points P_k and P_q are selected along a virtual ray. (c) The two contour points are accumulated into separate lists. (d) A part from the fisheye image corresponding to the selected Region of Interest (ROI) in (a). (e) The extracted obstacle delimiters by using the classical Border Scanner approach. (f) The Multiple Depth Border Scanner (current solution). It can be observed that the proposed solution is able to extract more complete object shapes, including the occluded parts.

implementations which consisted in extracting only the first visible obstacle points (one point per ray), current solution extracts the intersection points with other obstacles situated at different depths along the same ray (one point per obstacle). In other words, for each scanning axis and for each object intersecting that axis we collect its closest (not occluded) point (see Fig. 5).

IV. OBSTACLE CLASSIFICATION

For a better understanding of obstacles we classify them using visual codebook based image descriptors as: *pedestrian*, *AGV* or *other* obstacle. The 3D information is taken into consideration for limiting the minimum and maximum size of the obstacle candidates for each class (50% below minimum and 50% above maximum size).

To obtain the classification features for an obstacle, first, the 3D bounding box is projected into the left fisheye intensity frame and is cropped out as a rectangular image. Due to the nature of the fisheye lens, the image is radially symmetrical. Therefore, each obstacle image is rotated according to the polar angle of the obstacle's position in the fisheye image, by considering the fisheye image center as the origin for the polar reference system. Rotation is done relatively to the 90 degrees polar angle. The relationship between obstacle orientation and polar angle can be seen in Fig. 6. To achieve scale invariance, the image is resized to have a fixed width of 100 pixels, if the width is greater than the height. Otherwise it is resized to have a fixed height of 100 pixels.

Dense HOG descriptors are computed over the 2D obstacle image. A 24 dimensional descriptor vector is obtained at each pixel position. The descriptor vectors are discretized using a visual codebook (or dictionary) consisting of 100 visual words. After discretization each pixel position is represented by one of the hundred visual words and the image can be described by the distribution of words. Implementation details regarding descriptor computation and codebook training can be found in our work on pedestrian detection [41]. 21 image regions are considered by applying the following partitionings: 1×1, 2×2 and 4×4 (3 level spatial pyramid [42]). By computing the histogram of visual words for the 21 regions, we obtain 2100 individual classification features.

We train two binary Ada-boost [43] classifiers for

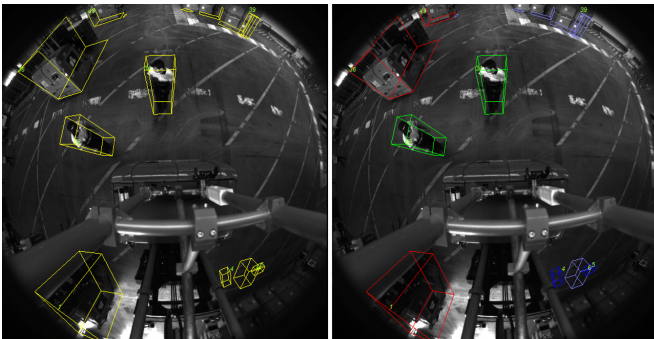


Figure 6. Obstacle classification: AGV – red; Pedestrian – green; Other Large – light blue; Other Small – dark blue.

pedestrians and AGVs. 2048 boosting rounds are used to train an ensemble of two-level decision trees learned over the 2100 classification features. The inverse of the boosting decision function value is used as probability estimate. If an obstacle is classified as pedestrian and also as AGV (for example when both are visible in the obstacle image) then we consider the class with the higher probability. If an obstacle is neither classified as pedestrian nor as AGV, then it is labeled as *other small obstacle* if it has a height of less than 50 cm, or *other large obstacle* otherwise.

V. OBSTACLE TRACKING

The object tracking module is aiming to estimate, recursively in time, the state of the detected obstacles around the AGV given all measurements up to the current time t . The object tracking solution is based on a Kalman filtering mechanism used for each individual detected target. In our case, the obstacle state X_t , at a time t , is defined by the following variables:

$$X_t = [x_t, z_t, vx_t, vz_t, W_t, L_t, H_t] \quad (1)$$

where x_t and z_t represent the object position, vx_t and vz_t are the object speed components and W_t , L_t and H_t describe the object width, length and height properties.

We consider that the coordinate system of the vision based perception module is situated in front of the AGV vehicle, with the X axis pointing to the right and Z axis pointing towards the AGV direction. The overall obstacle tracking solution can be described by the steps that follow below.

A. AGV Motion Compensation

Before estimating the dynamic properties of other obstacles, we should also take into consideration the AGV's motion. The PAN-Robots AGV localization parameters are provided by a dedicated self-localization system [35] that is able to estimate the AGV's position in the warehouse with an accuracy of 1cm. At each frame, the localization module provides the AGV's orientation α_{AGV} , its coordinates in the warehouse reference system, and a timestamp. Following a circular motion model, a point from the previous coordinate system is transformed into the current coordinate system according to:

$$\begin{aligned} \begin{bmatrix} x_{t-1}^* \\ z_{t-1}^* \end{bmatrix} &= R(-(\alpha_{AGV}^t - \alpha_{AGV}^{t-1})) \begin{bmatrix} x_{t-1} + tx_{CA} \\ z_{t-1} + tz_{CA} \end{bmatrix} \\ &\quad - R(-\alpha_{AGV}^{t-1}) \begin{bmatrix} tx_{AGV} \\ tz_{AGV} \end{bmatrix} - \begin{bmatrix} tx_{CA} \\ tz_{CA} \end{bmatrix} \end{aligned} \quad (2)$$

where, $[tx_{CA}, tz_{CA}]^T$ represents the distance between the camera and the origin of the AGV reference system, and $[tx_{AGV}, tz_{AGV}]^T$ is the AGV translation vector between the previous and current frames in the warehouse coordinate system.

B. State Prediction

Before incorporating the new measurements, the state of each track is predicted from the previous information X_{t-1} according to its state transition probability $p(X_t | X_{t-1})$. By considering that the tracked obstacles are described by a linear motion model, the state parameters are predicted from its previous state compensated with the AGV's motion $X_{t-1} = [x_{t-1}^*, z_{t-1}^*, v_{x,t-1}, v_{z,t-1}, W_{t-1}, L_{t-1}, H_{t-1}]^T$ according to:

$$X_t = AX_{t-1} + w \quad (3)$$

Equation (3) defines the motion model described by a state transition matrix A and a random noise, which is drawn from a zero mean Gaussian distribution $w \sim N(0, Q)$ with covariance Q . The covariance Q is adjusted by considering a maximum allowed obstacle acceleration.

C. Data Association

The data association consists in assigning the new extracted objects to the predicted targets. First we compute a distance based association metric by counting all point-to-point correspondences between the newly extracted object delimiters and the nearest tracks. First, each contour of a tracked object from the previous frame is projected into the current frame coordinate system according to the object's dynamic parameters and the AGV motion. Then, for each predicted contour point the closest measurement point is selected as the association candidate. In order to optimize this point-to-point selection phase, the closest associated points are determined in $O(1)$ by using pre-computed distance transforms as a look-up table, in which each cell stores the position of the closest contour point. The data association is performed in two directions: from measurements to targets and from targets to measurements. In order to avoid the ambiguous association cases, when the same observation may belong to multiple tracks or vice versa, we also take into account the object types provided by the classification module.

D. Computing the Object Motion

For extracting the obstacle motion we use an Iterative Closest Point-based solution [36] previously applied by us in a driving assistance application [37]. The ICP technique is used to compute the optimal transformation between the tracks and the associated observations by minimizing the alignment error. Having the set of points $S_{target} \{p_i^{target} | i=1..N_{target}\}$ that describe the tracked object contour and the set of points $S_{meas} \{p_j^{meas} | j=1..N_{meas}\}$ describing the extracted contour, the optimal rotation R and translation T are computed by iteratively minimizing the following objective function:

$$err(R, T) = \frac{1}{N} \sum_{i=1}^N \|Rp_k^{target} + T - p_k^{meas}\|^2 \quad (4)$$

where N is the number of point-to-point correspondences $(p_k^{target}, p_k^{meas})$ that are determined by selecting for each

point p_k^{target} in the target contour S_{target} the closest corresponding point p_k^{meas} in the newly observed contour S_{meas} . Since the estimated track-to-measurement transformations represent the difference between the predicted state variables and the current observation, the measured object speed can be calculated as the sum of the newly extracted motion components and the predicted speed parameters (the initial guess).

E. Object State Update

Having a measurement vector defined by the computed speed components and the extracted cuboid position and size, the new object state and its covariance are updated by using the standard Kalman filter equations.

VI. EXPERIMENTAL RESULTS

The proposed system was tested in various industrial warehouse environments including static and dynamic obstacles of different types. For the experiments we used a GPU equipped industrial PC that was installed on a PAN-Robots AGV [1]. The whole perception system runs in real-time at 10 frames/second.

Fig. 7 presents a scenario with moving pedestrians around the AGV. Each object is represented by a cuboid model with a velocity vector (red color). Fig. 8 illustrates how the obstacles are tracked in time. Each individual obstacle is



Figure 7. Dynamic obstacles and their speed vectors (red).

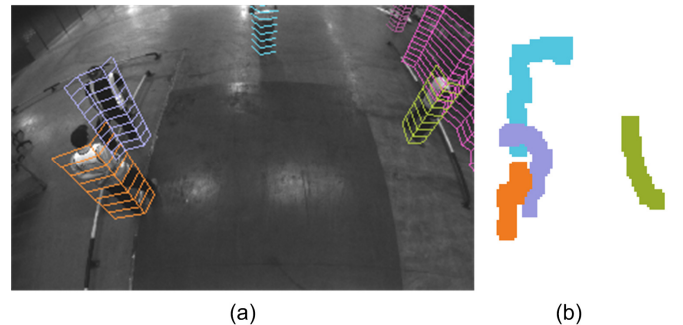


Figure 8. Obstacle Tracking. (a) The detected objects are illustrated with different colors. (b) The trajectories generated by each tracked object (top view).

TABLE I
TRACKING EVALUATION

Detection rate	94.3 %
Tracking rate	92.8 %
Number of miss-associations	31 (during 2165 frames)
Mean localization error	0.157 m
Mean velocity error	0.48 m/s

represented by a unique ID (a different color). Fig. 8b shows the generated trajectories by the tracked objects. A short video presenting the results can be accessed at [44].

For quantitative results we used a test scenario including static and dynamic obstacles. In order to generate ground truth data we tracked and annotated manually 16 obstacles for 2165 frames. The ground truth targets were in the perception range for a number of frames varying between 50 and 900. Table I. provides an overview of the computed performance metrics for tracking evaluation. During the 2165 frames there were 31 miss-associations, i.e. the tracking ID of an obstacle has been changed, mostly due to difficult occlusion cases. The ground truth obstacles were detected for 94.3% of the time and were correctly tracked for 92.8% of the time. The average localization error was of 0.157 m, while the average velocity error was of 0.48 m/s.

VII. CONCLUSION

The main purpose of this work was to provide a robust perception solution for AGVs in order to detect and track obstacles in dynamic industrial environments. We proposed a system that uses omnidirectional stereo cameras for 360 degree surround perception. The classified digital elevation map, resulting from the stereo data, is used to generate obstacle hypotheses. The obstacle tracking solution relies on two representations: a 3D cuboid used for obstacle classification and a free-form polygonal model used for its motion estimation.

The proposed solution was implemented and tested on AGVs in an industrial warehouse environment. The obtained experimental results are promising, however the maintaining of tracking for temporarily occluded obstacles is still a challenge and its further research will be of interest in the future. Also, the accuracy and stability can be improved by removing static obstacles like walls or by incorporating the intensity information in the data association stage.

ACKNOWLEDGMENT

This work was supported by the research project PAN-Robots, 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] PAN-Robots - Plug And Navigate ROBOTS for smart factories, FP7 EU Project. [online] <http://www.pan-robots.eu/>
- [2] 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
- [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] 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 (ICCP 2014)*, Cluj-Napoca, Romania, 2014, pp.233-238.
- [5] H. Yuste, L. Armesto, and J. Tornero, "Benchmark tools for evaluating AGVs at industrial environments", *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2010)*, pp.2657-2662, 18-22 Oct. 2010.
- [6] R. Olmi, C. Secchi, and C. Fantuzzi, "An efficient control strategy for the traffic coordination of AGVs", *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2011)*, pp. 4615-4620, 25-30 Sept. 2011.
- [7] V. Digani, L. Sabattini, C. Secchi, and C. Fantuzzi, "An automatic approach for the generation of the roadmap for multi-AGV systems in an industrial environment", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014)*, pp.1736-1741, 14-18 Sept. 2014.
- [8] V. Digani, L. Sabattini, C. Secchi, and C. Fantuzzi, "Hierarchical traffic control for partially decentralized coordination of multi AGV systems in industrial environments", in *IEEE International Conference on Robotics and Automation (ICRA 2014)*, pp.6144-6149, May 31 2014-June 7 2014.
- [9] 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 (ICCP 2014)*, Cluj-Napoca, Romania, 2014, pp. 227-232.
- [10] B. Massimo, B. Luca, B. Alberto, and C. Alessandro, "A Smart vision system for advanced LGV navigation and obstacle detection", in *International IEEE Conference on Intelligent Transportation Systems (ITSC 2012)*, pp. 508-513, 16-19 Sept. 2012
- [11] P. S. Pratama, S. K. Jeong, S. S. Park, and S. B. Kim, "Moving Object Tracking and Avoidance Algorithm for Differential Driving AGV Based on Laser Measurement Technology", in *International Journal of Science and Engineering*, vol. 4, no. 1, pp. 11-15, January, 2013.
- [12] A. Cherubini and F. Chaumette, "Visual navigation with obstacle avoidance", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2011)*, pp.1593-1598, 25-30 Sept. 2011.
- [13] S. Sivaraman and M. M. Trivedi, "Looking at Vehicles on the Road: A Survey of Vision-Based Vehicle Detection, Tracking, and Behavior Analysis", in *IEEE Transactions on Intelligent Transportation Systems (T-ITS 2013)*, vol.14, no.4, pp.1773-1795, Dec. 2013.
- [14] C. Lopez-Franco, N. Arana-Daniel, and E. Bayro-Corrochano, "Vision-based robot control with omnidirectional cameras and conformal geometric algebra", in *IEEE International Conference on Robotics and Automation (ICRA 2010)*, pp.2543-2548, 3-7 May 2010
- [15] P. Merveilleux, O. Labbani-Igbida, and E.-M. Mouaddib, "Real-time free space detection and navigation using omnidirectional vision and parametric and geometric active contours", in *IEEE International Conference on Robotics and Automation (ICRA 2011)*, pp.6312-6317, 9-13 May 2011.
- [16] R.J.D. Moore, K. Dantu, G.L. Barrows, and R. Nagpal, "Autonomous MAV guidance with a lightweight omnidirectional vision sensor", in *IEEE International Conference on Robotics and Automation (ICRA 2014)*, pp.3856-3861, May 31 2014-June 7 2014
- [17] S. Goto, A. Yamashita, R. Kawanishi, T. Kaneko, and H. Asama, "3D environment measurement using binocular stereo and motion stereo by mobile robot with omnidirectional stereo camera", in *IEEE International Conference on Computer Vision Workshops (ICCV Workshops 2011)*, pp.296-303, 6-13 Nov. 2011
- [18] 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 (ICCP 2014)*, Cluj-Napoca, Romania, 2014, pp. 251-258.
- [19] S. Vedula, S. Baker, P. Rander, R. T. Collins, and T. Kanade, "Three-Dimensional Scene Flow", in *IEEE Trans. Pattern Analysis and Machine Intelligence (T-PAMI 2005)*, 27(3): 475-480, 2005.
 - [20] U. Franke, C. Rabe, H. Badino, and S. Gehrig, "6d-vision: Fusion of stereo and motion for robust environment perception," in *DAGM '05*, 2005, pp. 216-223.
 - [21] T.-N. Nguyen, B. Michaelis, A. Al-Hamadi, M. Tornow, M. Meinecke, "Stereo-Camera-Based Urban Environment Perception Using Occupancy Grid and Object Tracking", in *IEEE Transactions on Intelligent Transportation Systems (T-ITS 2012)*, vol.13, no.1, pp.154-165, 2012.
 - [22] M. Perrollaz, J.-D. Yoder, A. Nègre, A. Spalanzani, and C. Laugier, "A visibility-based approach for occupancy grid computation in disparity space", in *IEEE Transactions on Intelligent Transportation Systems (T-ITS 2012)*, vol. 13, no. 3, Sep. 2012, pp. 1383-1393.
 - [23] M. A. Garcia and A. Solanas, "3D Simultaneous Localization and Modeling from Stereo Vision", in *Proc. of IEEE International Conference on Robotics and Automation (ICRA 2004)*, New Orleans, USA, April-May 2004, pp.847-853.
 - [24] A. Broggi, E. Cardarelli, S. Cattani, M. Sabbatelli, "Terrain mapping for off-road Autonomous Ground Vehicles using rational B-Spline surfaces and stereo vision", in *Proc. of 2013 IEEE Intelligent Vehicles Symposium (IV 2013)*, 23-26 June 2013, pp. 648-653.
 - [25] R. Danescu, S. Nedevschi, "A Particle-Based Solution for Modeling and Tracking Dynamic Digital Elevation Maps", *IEEE Transactions on Intelligent Transportation Systems (T-ITS 2014)*, vol. 15, No. 3, June 2014, pp. 1002-1015.
 - [26] A. Azim, O. Aycard, "Detection, classification and tracking of moving objects in a 3D environment," in *Proc. of IEEE Intelligent Vehicles Symposium (IV 2012)*, 3-7 June 2012, pp.802-807.
 - [27] M. Yokoyama and T. Poggio, "A Contour-Based Moving Object Detection and Tracking", in *Proc. of the IEEE 14th International Conference on Computer Communications and Networks (ICCCN '05)*, Washington, DC, USA, 2005, pp. 271-276.
 - [28] C.C. Wang, C. Thorpe, M. Hebert, S. Thrun, and H. Durrant-Whyte, "Simultaneous localization, mapping and moving object tracking", in *The International Journal of Robotics Research*, 26(6), June 2007.
 - [29] A. Petrovskaya and S. Thrun, "Model based vehicle tracking for autonomous driving in urban environments", In *Proceedings of Robotics: Science and Systems IV (RSS 2008)*, Zurich, Switzerland, 34, 2008.
 - [30] A. Broggi, S. Cattani, M. Patander, M. Sabbatelli and P. Zani, "A full-3D voxel-based dynamic obstacle detection for urban scenario using stereo vision", in *Proc. of 16th International IEEE Conference on Intelligent Transportation Systems - (ITSC 2013)*, 6-9 Oct. 2013, pp.71-76.
 - [31] H.Cho, Y. W. Seo, B. V. K. Kumar, and R.R. Rajkumar, "A multi-sensor fusion system for moving object detection and tracking in urban driving environments." In *2014 IEEE International Conference on Robotics and Automation (ICRA 2014)*, 2014, pp. 1836-1843.
 - [32] H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised on-line boosting for robust tracking". In *ECCV*, 2008, pp. 234-247.
 - [33] A. Vatavu, S. Nedevschi, and F. Oniga, "Real Time Object Delimiters Extraction for Environment Representation in Driving Scenarios", In *Proc. of ICINCO-RA 2009*, Milano, Italy, 2009, pp 86-93.
 - [34] A. Vatavu, R. Danescu, and S. Nedevschi, "Modeling and tracking of crowded traffic scenes by using policy trees, occupancy grid blocks and Bayesian filters", in *Proc. of 2014 IEEE 17th International Conference on Intelligent Transportation Systems (ITSC 2014)*, 8-11 Oct., 2014, pp. 1948-1955.
 - [35] C. Reinke and P. Beinschob, "Strategies for contour-based self-localization in large-scale modern warehouses", in *2013 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP 2013)*, 5-7 Sept. 2013, pp.223-227.
 - [36] P. Besl and N. McKay, "A method for registration of 3d shape", in *Trans. Pattern Analysis and Machine Intelligence (T-PAMI 1992)*, 12(2), 1992.
 - [37] A. Vatavu and S. Nedevschi, "Vision-based tracking of multiple objects in dynamic unstructured environments using free-form obstacle delimiters", in *2013 European Conference on Mobile Robots (ECMR'13)*, Barcelona, Catalonia, Spain, 2013, 25-27 Sept. 2013, pp. 367-372.
 - [38] A. Mäyrä, M. Aikio, and M. Kumpulainen, "Fisheye optics for omnidirectional perception," in *IEEE International Conference on Intelligent Computer Communication and Processing (ICCP 2014)*, Cluj-Napoca, Romania, 2014, pp. 259-263.
 - [39] 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 (ICCP 2014)*, Cluj-Napoca, Romania, 2014, pp. 149-156.
 - [40] 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 (IV 2010)*, San Diego, California, USA, 2010, pp. 369-376.
 - [41] 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 (CVPR 2014)*, Columbus, Ohio, USA, 2014, pp. 2393-2400.
 - [42] 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 (CVPR 2006)*, 2006, pp. 2169 - 2178.
 - [43] 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.
 - [44] Modeling and Tracking of Dynamic Obstacles – Demo Video: Available at: <http://cv.utcluj.ro/omnidirectional-stereo-vision---object-detection-and-tracking.html>