Probabilistic Inverse Sensor Model based Digital Elevation Map Creation for an Omnidirectional Stereovision System

Szilárd Mandici, Sergiu Nedevschi[1]
Technical University of Cluj-Napoca, Romania

Abstract—**The objective of the paper is to present an original solution for building a high accuracy Digital Elevation Map (DEM), from down-looking omnidirectional stereo system data, used for surrounding perception. For this reason, an accurate probabilistic inverse sensor model of the omnidirectional stereo sensor is estimated based on training data. The obtained model considers not only the Gaussian spread of 3D points but also the systematic translations and errors from the calibration and rectification processes. The inverse sensor model is obtained by calculating the prior probabilities of 3D points corresponding to each DEM cell and the direct sensor model, describing the way measurements are acquired. The direct sensor model is calculated using an umbrella based modified Shepard trilinear interpolation of individual measurements in space. The results of the interpolation (σ_x, σ_y, σ_z, μ_x, μ_y, μ_z) are stored in a 3D lookup table which performs a discretization of 3D space into cuboids. For each 3D point the probability of correspondence to the neighboring cells is calculated and the obtained values are added to the height histogram of each cell. Instead of adding to a single bucket in the histogram, the contribution is spread based on the standard deviation of the height. In order to increase the contribution of individual points in sparse areas and to decrease it in dense areas, the relative density of 3D points in local patches is precomputed and is used as a decreasing exponential term. Based on the obtained models, an improved DEM creation algorithm is applied. The obtained elevation map provides better results both in terms of accuracy and detection rate.**

I. Introduction

There are different ways of representing the environment based on the 3D points obtained from dense stereo reconstruction. For surrounding perception applications the most important final aspect is the accuracy and detection rate of objects. Because of the real-time requirement of such systems, an intermediate representation is usually used which allows for faster processing. Digital Elevation Maps are a common solution for processing dense stereo data in the literature [1, 2, 3, 4]. Digital Elevation Maps (DEM) are 2D Cartesian grid maps, where each cell contains height information as well.

The primary focus of the paper is to improve the quality of the single frame DEM in the case of an omnidirectional stereovision system, considering the tradeoff between the accuracy and detection rate of objects.

In [2] a Multivariate Gaussian probability based Digital Elevation Map creation algorithm is presented based on a canonical uncertainty model.

In this work a more accurate inverse sensor model is developed based on experiments performed with template objects. The algorithm is improved significantly and adapted to the case of the omnidirectional stereovision system used for Automated Guided Vehicles (AGV) [5]. The omnidirectional stereo system uses fisheye lenses mounted at the top of the AGV at a height of approximately 4m, downwards-looking and allowing a 360^o perception of the surrounding environment.

The omnidirectional stereovision system performs multi-channel decomposition of the original fisheye images [6] in 3 pairs of rectified images, allowing the use of the standard SORT-SGM [7] dense stereo algorithm for stereo reconstruction. In order to obtain a single Digital Elevation Map, the obtained 3D reconstructed points are collected and aggregated considering the coordinate systems and the reconstruction uncertainty corresponding to each virtual stereo camera.

Using the canonical reconstruction uncertainty model [2] developed for pinhole cameras is not satisfactory for such a complex system where the original fisheye images are highly distorted. The proposed solution with the inverse sensor model estimated from training data is meant to correct the errors of the other components of the system (calibration, rectification and stereo reconstruction). The inverse sensor model is developed for each channel. Each 3D point with the corresponding inverse sensor model parameters is added to the intermediate data structure, meant to calculate the final Digital Elevation Map.

In order to calculate the inverse sensor model, the proposed algorithm makes use of the prior occupancy probability of each DEM cell estimated from training data and the direct sensor model. The direct sensor model describes the way measurements are acquired by the sensor. There are 2 components of the direct sensor model in the proposed solution: systematic translation errors and the Gaussian multivariate noise around the true location. Both components are obtained by placing objects of known dimensions in known positions and measuring the 3D points reconstructed from the surface of the objects. To generalize the model to the whole ROI (Region of Interest), trilinear interpolation is performed in space using an Inverse Distance Weighting method: modified Shepard interpolation [8]. The interpolation algorithm is modified by applying a virtual

[1]The authors are with the Faculty of Automation and Computer Science, Technical University of Cluj-Napoca, Romania (mandici.szilard@gmail.com, Sergiu.Nedevschi@cs.utcluj.ro). The paper was written within the PAN-Robots project. The research to this project was funded by the European Union Seventh Programme Framework (FP7/2007-2013) under grant agreement no. 314193

978-1-4673-8200-7/15/$31.00 ©2015 IEEE

umbrella in polar coordinates around the points which are taken into consideration.

For each DEM cell a histogram of heights is calculated where instead of incrementing the corresponding accumulator, the probability of occupancy is added [2]. In the proposed solution the standard deviation of height is also taken into consideration and instead of contributing to a single accumulator, the neighboring accumulators defined by the height variance are considered. The contribution to each accumulator is weighted by a 1D Gaussian distribution. The relative density of 3D points in different local patches is also computed based on the prior probability map obtained from experiments. This value is precomputed for each DEM cell and it is fed into a decreasing exponential function. This has the effect of giving a larger weight to single 3D measurements in areas where there are usually less points and decreasing the weight of single 3D points in dense areas, equalizing the distribution of cells with high confidence in the ROI.

II. Related Work

Various solutions have been proposed in the literature for constructing Digital Elevation Maps [2, 6, 9, 10, 11]. The simpler methods which do not take into account the stereo reconstruction uncertainty can be directly applied for an omnidirectional stereo system as well. The algorithms which take into account the uncertainties are harder to adapt to such systems. The aim of the DEM construction algorithms is to calculate the height in each cell as close to reality as possible. There are different types of algorithms which differ mostly in the quality of results and the processing time [2].

The simplest algorithms for constructing Digital Elevation Maps are the ones which consider only the 3D points directly falling in a certain DEM cell and take the lower/upper extremes in height [9]. In [6] the 2 possibilities are mixed: the lower extreme is used as road surface candidate points which is then fitted with RANSAC and the higher extreme is used as the height value in DEM cells. This approach would potentially work well in the case of bimodal cells (cells containing both road surface and object points). It can be basically represented as 2 elevation maps used for different purposes: one for road surface detection and the other for object detection. This solution is directly applied to the context of the studied omnidirectional stereovision system. Another simple and computationally cheap approach is computing the average of height values. These solutions do not take into consideration the fact that a 3D reconstructed point may correspond to other adjacent DEM cells.

In [12] a probabilistic Gaussian model is used in the context of occupancy grid maps. The occupancy level of a map cell is composed of the contribution of different individual 3D measurements based on the relative position with respect to the elevation map cell [12]:

$$P_t^{occ}(c_i \mid P_C) = K \cdot \oint_{Area} \frac{\exp\left(-\frac{1}{2}\left(\frac{P_C - P_{cell}}{\sigma}\right)^2\right)}{\sigma\sqrt{2\pi}} \quad (1)$$

In (1) the occupancy probability is expressed for the cell c_i corresponding to the measurement P_C as the integral under the area of the elevation map cell of a 2D Gaussian probability function around the cell with standard deviation σ. K is a constant which represents the level to which a single 3D point can contribute to the occupancy of a cell and it depends on the camera system [12].

In [9], DEMs are computed by considering a histogram for each cell and recording the points which fall in it. The final height in the cell is considered to be the largest consistent height. This decision of choosing the largest consistent value was taken in order to handle the case of bimodal cells by taking the value of the larger elevation.

A stereo reconstruction uncertainty based elevation map estimation algorithm is proposed in [2]. The used uncertainty model is a canonical one which is widely used in the literature. Such a model is not necessarily accurate in the case of the omnidirectional fisheye stereo system for which the elevation map needs to be created.

Omnidirectional stereo system calibration and 3D points reconstruction are an intensive area of research [13, 14, 15] but it is mostly focused on the initial components of the system and not the 3D data processing in the form of Digital Elevation Maps or other intermediate data structures.

III. Proposed Solution

The algorithm proposed in this paper uses an enhanced and high accuracy inverse sensor model of the omnidirectional stereovision sensor. The solution is composed of 2 main parts: one part for finding the inverse sensor model for each virtual stereo camera and the other one for actually using these models to create the Digital Elevation Map. The inputs to the algorithm are the 3D reconstructed points and the estimated inverse sensor models and the output is the enhanced unified Digital Elevation Map.

In this chapter a very brief system overview of the PAN-Robots Omnidirectional Stereovision System is presented which is relevant for the DEM creation. The algorithm for estimating the inverse sensor models in offline mode and the solution for creating the high-accuracy elevation map is also described in detail, focusing on the original contributions of the paper.

A. System overview and multi-channel 3D point generation

The setup of the omnidirectional stereovision system for which the inverse sensor model is developed is depicted in Figure 1. The fisheye lenses with the cameras are mounted at approx. 4 meters on top of the AGV. The stereo system is looking downwards, ensuring a 360^{o} Field of View around the AGV [6]. The original highly distorted fisheye image is divided into 3 pinhole images obtaining 3 pairs of left-right rectified images corresponding in fact to 3 virtual pinhole stereo cameras looking in 3 different directions. This way the traditional dense stereo reconstruction algorithms can be applied to each pair and the 3D points for each channel are obtained.

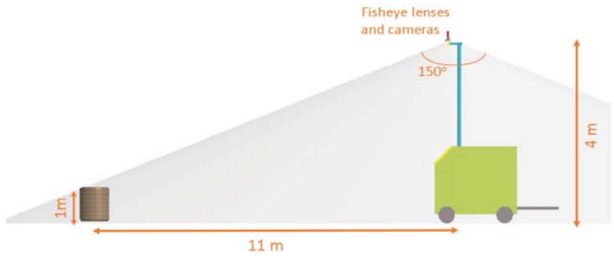

Figure 1. The position of fisheye lenses and cameras on top of the AGV [6].

The virtual imagers which are used for generating the 3 image pairs are presented in Figure 2. These virtual imagers are placed in such a way that they are covering the entire Region of Interest with partial overlapping. This makes the processing of 3D points as an uncertainty based DEM harder because a fusion between the 3 different data sources needs to be performed. A solution to this issue is also proposed in this paper.

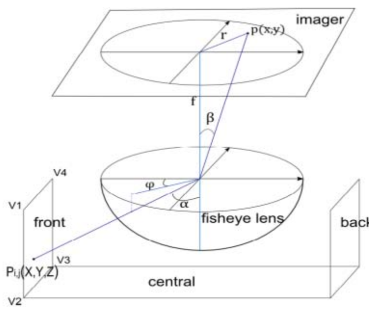

Figure 2. The 3 virtual imagers (front, central, back) used for rectification [6]

B. Direct and Inverse Sensor Model

The sensor model represents the way measurements are generated in the real world [16]. Hence, knowing the sensor model of the omnidirectional stereo sensor is a crucial aspect. The more accurate a sensor model is, the better the results of the algorithms processing the raw sensor data becomes. There are a variety of characteristics which can be taken into consideration for finding the sensor model of the omnidirectional stereovision sensor: relative position, color (wavelength), texture, image brightness, specular reflection etc [16]. A good model takes into account the most relevant characteristics which are computationally feasible to consider.

In sensorial robotics there are 2 complementary models which are used: the inverse and the direct sensor model. The direct sensor model captures the way the measurements are generated as a probability distribution:

$$P(z_t \mid x_t, m) \tag{2}$$

In (2) z_t is a random variable and represents the value of the measurement at time t, x_t represents the position of the robot (sensor) and m denotes the known position of objects in the environment.

In the context of the omnidirectional stereovision sensor we propose to use 2 components of the direct sensor model: systematic errors and Gaussian noise. The systematic errors are meant to model the errors which are always present in the system and comes mainly from stereo calibration and rectification imperfections. The Gaussian noise represents the way 3D reconstructed points are scattered around the true location (mean). It is basically a trivariate normal distribution in space which when projected to the ground plane becomes a bivariate model. In order to have an accurate representation of the direct model both components need to be determined. Taking these into consideration the direct model becomes:

$$P(z_t \mid x_t, m) = \begin{cases} N(z_t^*, z_t, \sigma); z_t \in ROI \\ 0 \end{cases} \tag{3}$$

In (3) z_t^* represents the true position, while σ stands for the parameters of the normal distribution and ROI is the Region of Interest.

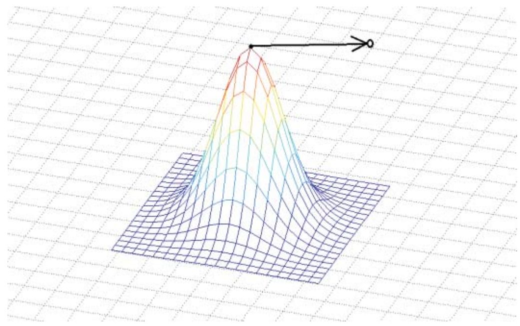

Figure 3. The direct sensor model with both components: Gaussian noise and systematic translation

The inverse sensor model is based on inverting the direct one in terms of the known and estimated parameters [16]:

$$P(m \mid z_t, x_t) \tag{4}$$

As observers, the inverse sensor model is the natural way of describing a sensor because one wants to obtain an estimation of the position of different objects in the environment (or the 3D points reconstructed from these objects) based on the position of the robot and the acquired measurements. One can define the inverse sensor model relative to the camera view and this way the inverse model in each position depends only on the acquired measurements:

$$P(m \mid z_t, x_t) = P(m \mid z_t) \tag{5}$$

Using this assumption and expressing the inverse sensor model as a function of the direct sensor model using Bayes' theorem the following is obtained:

$$P(m \mid z_t) = \frac{P(z_t \mid m) \cdot P(m)}{P(z_t)} \tag{6}$$

Equation (6) describes the probability of object positions m knowing the measurements z_t obtained at time t. If this inverse sensor model is applied to finding the Digital Elevation Map using raw 3D points obtained from dense stereo, the equations need to be reformulated accordingly. In the case of occupancy grid maps the usual assumption of conditional independence between elevation map cells is used:

$$P(m \mid z) = \prod_{i,j} P(c_{i,j} \mid z) \tag{7}$$

$$P(c_{i,j} \mid z) = \alpha \cdot P(z \mid c_{i,j}) \cdot P(c_{i,j}) \tag{8}$$

In (7) and (8) $P(c_{i,j}|z)$ represents the probability that a measurement z corresponds to the cell $c_{i,j}$. $P(c_{i,j})$ is the prior occupancy probability of cell $c_{i,j}$ while α is a constant representing the prior probability of any single measurement.

C. Components of the inverse sensor model from experiments

In order to define the inverse sensor model, both the prior probability for each cell and the direct sensor model need to be found. Because of the more complicated system setup these components are obtained through experiments.

The prior probability of correspondence of a reconstructed 3D point to a certain DEM cell can be defined as:

$$P_{sample}(c_{i,j}) = \frac{nr_{i,j}}{nr_{total}} \tag{9}$$

In (9) $nr_{i,j}$ represents the number of 3D reconstructed points which falls in a certain cell in a given frame, while nr_{total} stands for the total number of reconstructed 3D points in the frame. A single frame does not provide enough information and in order to obtain a better estimate of the prior probability several frames are taken into consideration in the following way:

$$P(c_{i,j}) = \frac{\sum_{k=1}^{n} P_{sample\_k}(c_{i,j})}{n} \tag{10}$$

In (10) n represents the number of frames being considered while $P_{sample\_k}(c_{i,j})$ is the sample probability of frame k. The obtained results are stored in a LUT in order to be easily accessible. In order to avoid numerical errors and instabilities of double precision numbers the average in a certain cell is calculated using the following recurrence:

$$avg_n = avg_{n-1} \cdot \left(\frac{n-1}{n}\right) + curr \cdot \frac{1}{n} \tag{11}$$

In (11) avg_n represents the average of the prior probability up to frame n, while $curr$ represents the prior probability obtained in the n-th frame.

The other component which needs to be estimated is the direct model. As previously mentioned the direct model is composed of 2 parts: the systematic translations and the Gaussian noise around the true position. These components are estimated by placing template objects of size (10x10x10 cm^3) in known positions and performing the following algorithm:

- Mark the objects with a polygon in the rectified images
- Collect all the 3D points reconstructed inside the polygon for several frames
- Calculate the mean, standard deviation, covariance and the vector from the calculated mean to the ground truth position corresponding to each template object
- Filter the results
- Perform a trilinear interpolation in order to generalize the model to the whole ROI

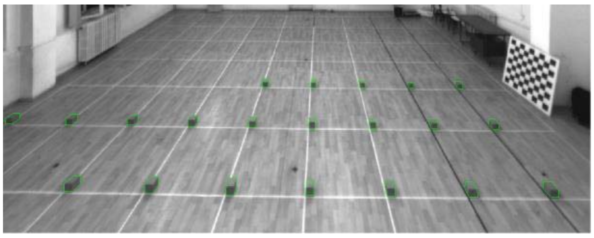

Figure 4. The marked template objects in the rectified channel 0

It is important that the size of the object is the same as an elevation map cell because the direct sensor model is specifically created for the Digital Elevation Map and the variance and covariance are calculated for the points which are situated in a single cell. The objects are placed at different heights as well in order to capture the change in parameters at different heights. This aspect is not too important in automotive applications but in the context of the omnidirectional stereovision system presented previously, significant changes might be present in parameters. After obtaining all the measurements, some of them are filtered based on certain variance thresholds (1 meter) and translation component thresholds (30 cm). The results of the remaining measurements need to be generalized for the whole ROI. There are basically 2 possibilities to approach this: treating it as a regression problem and fitting a function to the measurements or defining the model in each cell at each possible discretized height based on the measurements around the current position and storing it in a LUT. The second option was preferred because the first one presents many problems regarding the model to be chosen and the complexity of the function which needs to be fitted.

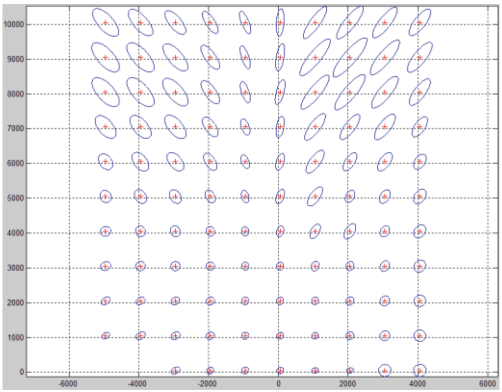

Figure 5. The obtained bivariate Gaussian models after interpolation in different areas of the environment

In the proposed solution an Inverse Distance Weighting (IDW) based trilinear interpolation method was used. The various interpolation methods mostly differ in the way weights are considered around the position of interest. The used method is called modified Shepard interpolation method [8, 17]. Only the measurements which are at a distance of at most R are considered and the weight in a certain position x with respect to a measurement which is situated in position x_k can be described in the following way:

$$w(x) = \left(\frac{\max(0, R - d(x, x_k))}{Rd(x, x_k)} \right)^2 \qquad (12)$$

This interpolation method was also adjusted in order to consider only a single measurement in a certain direction using the following algorithm:

- Consider the measurements in decreasing order of their Euclidean distance with respect to the target position *x*.
- For each new measurement at position x_k do
    - Check if the direction in polar coordinates overlaps with any of the previously added measurements
    - If it does not overlap, calculate the weight as in (10) and add it to the set of added measurements
- Based on the added measurements and their weights calculate the weighted average

The previous algorithm is performed for each DEM cell at each discretized height for each reconstruction channel. The height discretization was performed at a resolution of 10 cm which in a ROI between 3m and -1m corresponds to 40 values for each DEM cell. Basically each added point creates an umbrella around it and no measurement on the other side of the umbrella is considered. The angles used in the different directions is of ±5º for either angle in polar coordinates. The variance, covariance and the 3 components of the translation vector were calculated using this method. The results of the interpolation for each cuboid are placed in a LUT.

The presented algorithm can be further optimized using a data structure which directly gives only the values up to a distance *R* using k-d trees, quadtrees, octrees, segment trees or any combination of these data structures. Because of the fact that the algorithm is run only once offline for the stereo setup, it is not of crucial importance to have the best theoretically possible algorithm in terms of complexity.

D. Relative density of 3D points in local patch

One problem that appears when using a DEM based only on the inverse sensor model is that the cells which are closer to the cameras usually have higher confidence values as opposed to the cells which are farther away from the camera. This stems from the fact that there are more 3D reconstructed points closer to the observer. Having a single threshold for the confidence for all elevation map cells means that the ones which are farther away should have roughly the same number of 3D reconstructed points in their neighborhood as the ones which are closer to the cameras in order to be considered valid. To address this problem there are basically 2 possibilities: either use an adaptive threshold for the confidence in case of different environment areas or adjust the weight of the single 3D measurements in order to equalize the necessary confidence threshold. A solution is proposed which uses the second option based on the relative density of 3D points in local patches.

A 2D map of relative density is calculated based on the prior probability map previously obtained through experiments. For each local patch of 1m x 1m the sum of probabilities is evaluated. The maximum value is stored, and at the end, for each patch the ratio between the sum of probabilities in the patch and the maximum value is calculated. This will result in values between 0 and 1 where 1 will correspond to the densest area.

In order to apply smaller weight for the individual 3D measurements in dense areas and to apply larger weight in the other areas, a decreasing function in [0, 1] is needed. Several functions were tried but the best results were obtained by using the exponential function $f(x)=e^{-x}$. Using these observations, the contribution of a specific 3D point to the height value in a cell becomes:

$$Contr(c_{i,j}, z) = \alpha \cdot P(z \mid c_{i,j}) \cdot P(c_{i,j}) \cdot e^{-y(i,j)} \qquad (13)$$

In (13) $c_{i,j}$ represents the cell, z is the measurement, α is a constant, $P(z|c_{i,j})$ is the direct sensor model component, $P(c_{i,j})$ is the prior probability in the cell and $y(i,j)$ gives the relative density of 3D points in the patch centered at i,j.

E. Spread of height values using the measured standard deviation

In order to make the solution more robust and more accurate, instead of using directly the height of each 3D measurement, the scattered value of each height value is taken into consideration as a 1D Gaussian distribution. The standard deviation in each cell is the one which was obtained previously through experiments and trilinear interpolation. The contribution of a single 3D reconstructed point is spread probabilistically around the actual height of the point.

Usually the height values need to be discretized. The level of discretization used is 1 cm. The weight corresponding to a certain range of heights can be obtained by calculating their standard score or z-score. The z-score is the parameter of the normal distribution with mean 0 and standard deviation 1.

$$z = \frac{x - \mu}{\sigma} \quad (14)$$

$$f(z) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx \quad (15)$$

$$P_{from,to} = f(z_{to}) - f(z_{from}) \quad (16)$$

In (14), (15) and (16) z represents the z-score, x is the height value being considered, μ is the actual height of the 3D point, σ is the standard deviation in height in the current cell, f is the integral of the normal distribution with mean 0 and standard deviation 1 and $P_{from,to}$ is the weight for the discretized height interval [*from,to*]. The used standard deviation coefficient is 3, obtaining the sum of 0.997 in weights for the different height intervals. In order to avoid calculating the probability corresponding to a certain z-score, a LUT is generated in MATLAB with the calculated values.

F. Digital Elevation Map Creation

The used DEM creation algorithm is similar to the one described in [2] where a canonical uncertainty model is used but with signficant changes in order to adapt it to the context of the studied omnidirectional stereovision system and to use the previously presented 2-component inverse sensor models and enhancement techniques. The algorithm can be summarized as follows:

- For each reconstruction/rectification channel collect all the valid 3D reconstructed points in the ROI.
  - For each 3D point consider the cells around it using a standard deviation coefficient
    - Calculate the correspondence contribution for each cell as shown in (13) and add this value to the histogram of the cell (taking into account the channel from which the 3D point comes from)
- For each cell in the DEM
  - Generate a Gaussian kernel based on the standard deviation in height of the cell
  - Apply this kernel to the histogram obtaining the spread of weights described in *E.*
- Decide for each cell whether it has enough confidence in each histogram based on an empirically specified threshold of sum of weights
- For the valid cells choose as the height value the one which corresponds to the highest consistent weight in the obtained histogram
- Each elevation map cell is classified as obstacle/road surface based on the height value and a threshold of correspondence to the road surface

The 1D Gaussian spread specified in *E* is not calculated for each point separately. It is only calculated after adding all the 3D measurements for each bucket in the histogram of the cell in order to reduce the running time. One problem arises in the environment areas where 3D points coming from multiple channels are present. This issue is solved by calculating the weighted average of the standard deviation in height. The considered weights are the prior probabilities from experiments of the different channels in that cell.

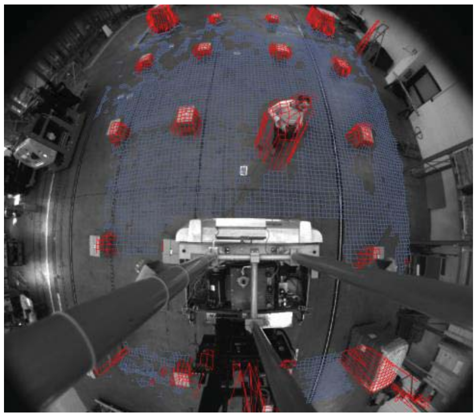

Figure 6. The obtained digital elevation map projected on the original fisheye image

IV. EXPERIMENTAL RESULTS

The created DEM is used by other components of the omnidirectional stereovision system mainly for object detection. The object detection quality is usually the primary concern when using DEMs in real applications. In order to be able to compare the results of the proposed solution with the DEM estimation algorithm presented in [6], both were evaluated on the same test data. The algorithm was evaluated in terms of detection rate, localization error, width error, height error and length error of detected objects. The position of the ground truth objects was measured using the AGV's localization system providing an accuracy of ±1cm [6].

The improvement for localization error, width error, length error and height error was calculated using the formula *(old_val – new_val)/old_val.* The improvement in detection rate was calculated as the difference between the 2 detection rates. For the evaluation, the confidence threshold of $9*10^{-4}$ was used with a standard deviation coefficient of 2. The confidence threshold can be decreased resulting in a higher detection rate but the other errors become larger. The objects which are not detected are situated usually at the edge of the elevation map at around 10m distance from the AGV. The running time with the mentioned parameters is around 170-180 ms/frame but setting the standard deviation coefficient to 1 halves the running time.

TABLE I. EVALUATION OF THE PROPOSED SOLUTION

Evaluation Criteria	*Result*	*Improvement w.r.t. [6]*
Detection Rate	90,7%	14.7%
Mean Localization Error	16.5 cm	25%
Mean Width Error	16.1 cm	38%
Mean Length Error	19.1 cm	40.3%
Mean Height Error	5.6 cm	49%
Frames Number	900	
Total Objects	9947	
Missing Objects	922	

The algorithm was additionally evaluated on the number of false positive objects detected based on the elevation map by grouping the elevation map cells classified previously as obstacle cells. Using the settings already mentioned, the number of false positive objects is really low. Evaluating as a human observer, only 4 false positive objects were detected in 260 observed frames.

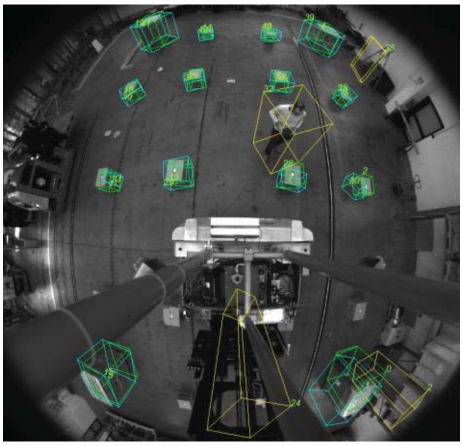

Figure 7. Sample from the evaluation set. (ground truth objects - green; detected and matched objects - blue; other detected objects - yellow)

V. CONCLUSION

The most relevant contribution of this paper is the high accuracy Digital Elevation Map estimation algorithm for omnidirectional stereo sensors. Other significant original contributions are the accurate inverse and direct omnidirectional stereovision sensor models estimation algorithms, the correction of the cell level height histograms using adaptive probabilistic Gaussian kernels and the introduction of the relative density of 3D points in local patches for equalizing the confidence threshold in different environment areas.

The proposed solution was extensively tested and it was demonstrated that it improves the quality of single frame Digital Elevation Maps in terms of detection rate and accuracy.

Even though the proposed algorithm is focused only on a single frame, as future work a more complex solution could be elaborated which takes into account subsequent frames performing temporal fusion between frames. Applying such a technique would improve the single frame DEM significantly. The different thresholds used in the algorithm and the size of the window for generating the relative density of 3D points are also subject to future research.

The running time can also be improved by applying further parallelization on CPU/GPU of different components of the algorithm.

REFERENCES

[1] F. Oniga and S. Nedevschi, "Processing dense stereo data using elevation maps: Road surface, traffic isle, and obstacle detection", IEEE Trans. Veh. Technol., vol. 59, no. 3, Mar. 2010, pp. 1172-1182.

[2] S. Mandici and S. Nedevschi, "Aggregate Road Surface based Environment Representation using Digital Elevation Maps", Intelligent Computer Communication and Processing (ICCP), Sept. 2014.

[3] J. Siegemund, U. Franke, and W. Forstner, "A temporal filter approach for detection and reconstruction of curbs and road surfaces based on conditional random fields", in Proc. IEEE Intelligent Vehicles Symp., 2011, pp. 637-642.

[4] A. Rankin, A. Huertas and L. Matthies, "Stereo-vision-based terrain mapping for off-road autonomous navigation", Proc. SPIE 7332, Unmanned Systems Technology XI, 733210, April 2009.

[5] http://www.pan-robots.eu/

[6] M. Drulea, I. Szakats, A. Vatavu and S. Nedevschi, "Omnidirectional stereo vision using fisheye lenses", Intelligent Computer Communication and Processing (ICCP), Sept. 2014.

[7] I. Haller, S. Nedevschi, "Design of Interpolation Functions for Subpixel-Accuracy Stereo-Vision Systems". IEEE Transactions on Image Processing 21(2): 889-898, 2012.

[8] D. Shepard, "A two-dimensional interpolation function for irregularly-spaced data", Proceedings of the 1968 ACM National Conference, 1968, pp. 517–524.

[9] T. Andersson and M. Lorentzon, "Road surface modeling using stereo vision", Performed at Autoliv Electronics, Linköping, Sweden. Master thesis, 2012.

[10] F. Oniga, S. Nedevschi, M. Meinecke and T. B. To, "Road surface and obstacle detection based on elevation maps from dense stereo", Proc. IEEE Intell. Transp. Syst. Conf., 2007, pp.859-865.

[11] F. Oniga, R. Danescu, and S. Nedevschi, "Mixed road surface model for driving assistance systems", IEEE International Conference on Intelligent Computer Communication and Processing (ICCP), Aug. 2010, pp.185- 190.

[12] N. Thien-Nghia, B. Michaelis, A. Al-Hamadi, M. Tornow, and M. Meinecke, "Stereo-camera-based urban environment perception using occupancy grid and object tracking", IEEE Trans. Intelligent Transportation System, vol. 13, no. 1, Mar. 2012, pp. 154-165.

[13] Z. Zhu, "Omnidirectional Stereo Vision", Workshop on Omnidirectional Vision in the 10th IEEE ICAR, Aug. 22-25, 2001.

[14] A. Nagy, I. Szakats, T. Marita and S. Nedevschi, "Development of an omnidirectional stereo vision system", in Intelligent Computer Communication and Processing (ICCP), 2013 IEEE International Conference on, 2013, pp. 235-242.

[15] Y. Negishi, J. Miura and Y. Shirai, "Calibration of Omnidirectional Stereo for Mobile Robots" in Proceedings of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Sendai, Japan, Sept./Oct. 2004.

[16] S. Thrun, W. Burgard and D. Fox, Probabilistic Robotics (Intelligent Robotics and Autonomous Agents), The MIT Press, 2005, ch 9.

[17] W.I. Thacker, J. Zhang, L.T. Watson, J.B. Birch, M.A. Iyer, M.W. Berry, "Algorithm XXX: SHEPPACK: Modified Shepard Algorithm for Interpolation of Scattered Multivariate Data" in ACM Trans. Math. Software (2009) 37, Article 34, 1–20, 2009.