Development of an Omnidirectional Stereo Vision System

Aliz-Eva Nagy, Istvan Szakats, Tiberiu Marita, Sergiu Nedevschi
Computer Science Department
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
{alizeva.nagy, szaki.ms}@gmail.com, {Tiberiu.Marita, Sergiu.Nedevschi}@cs.utcluj.ro

Abstract—**Although stereo systems built up of omnidirectional cameras offer the possibility of providing information for a 360 grade field of view, the research in the field is still in an early stage compared to perspective systems. We provide in this study an extensive overview on existing techniques for camera calibration, stereo rectification, stereo matching methods and 3D depth reconstruction on omnidirectional images. A comparison of methods, together with a few ideas of improvement is presented, establishing an equivalency, and transformation between an omnidirectional and a perspective vision system.**

Keywords—omnidirectional vision; stereo vision; camera calibration; image rectification; stereo reconstruction.

I. INTRODUCTION

Omnidirectional cameras are built either as a combination of a camera-mirror system, or as a combination of multiple cameras, to capture a scene with a field of view of 180 degrees or greater in the horizontal plane. For our system we chose a hyperbolic camera-mirror setup in a stereo configuration that provides an omnidirectional field of view of 360 degrees horizontally and 75 degrees in the vertical plane, with a single effective viewpoint. The motivation behind choosing omnidirectional cameras versus perspective ones lies mainly in the existence of a wider field of view, a property that perspective cameras can only approximate using complex stitching algorithms, with obvious errors in case of repetitive patterns, or a scene lacking in prominent features.

The main areas where omnidirectional systems can offer a significant advantage include: robot vision – by providing the possibility of seeing all-around and thus helping significantly in collision avoidance and fast object detection; large scale video surveillance – increasing the coverage of the surveillance area, reducing the number of images, therefore making movement detection and identification much easier with automated tools as well as manually; assisted driving and navigation – by eliminating black spots not covered by perspective imaging; map building and mosaicking – by reducing the number of necessary stitches and increasing coverage.

The increasing interest in omnidirectional cameras, their spreading employment in security systems and the growing number of publications on the subject all indicate that it is a promising branch of the field of computer vision with a potential need for extended research.

In our research we wish to cover all the necessary stages to obtain a high-performance stereo vision system with two catadioptric cameras, with hyperbolic mirrors. The main steps of developing such a system include finding the best projection model for a catadioptric camera, calibration of a single-camera system, stereo calibration of two omnidirectional cameras in a fixed configuration, omnidirectional image unwrapping and rectification, choosing an appropriate stereo-matching algorithm and finally reconstruction of 3D points from the two calibrated and rectified views.

In chapters II-V we present the theoretical background of our work, including the state-of-the-art of the domain completed with a few of our own suggestions and divided by development steps: calibration, image unwrapping, rectification respectively stereo matching and reconstruction.

In chapter VI we present our experimental results, in comparison to existing ones, and we draw our conclusions in chapter VII.

II. PROJECTION MODELS AND CALIBRATION

A. Projection Models

Several approaches exist for establishing the relation between 3D and image coordinates for omnidirectional cameras with hyperbolic mirrors. Each of them basically consists in a coordinate transform based on the mirror parameters, followed by the perspective projection of the corresponding camera.

The direct approach presented in [17] and [18] first computes the intersection of the incoming ray with the mirror surface, subsequently using a perspective projection matrix to determine the corresponding image point.

The Unified Projection Model for central catadioptric cameras, described in [7] is based on the same idea of computing the point of intersection, but uses a different parameterization, obtaining the final projection equation:

978-1-4799-1494-4/13/$31.00 ©2013 IEEE

$$(u,v)=\left(\frac{\pm\frac{2dp}{\sqrt{d^2+4p^2}}X}{\frac{d}{\sqrt{d^2+4p^2}}r\pm Z},\frac{\pm\frac{2dp}{\sqrt{d^2+4p^2}}Y}{\frac{d}{\sqrt{d^2+4p^2}}r\pm Z}\right) \quad (1)$$

Where **P** = (X,Y,Z) is a 3D point, **p** = (u,v) is an image point, *d* and *p* are the mirror parameters, *d* being the distance between the two foci of the hyperbola, 4*p* the latus rectum, and $r=\sqrt{X^2+Y^2+Z^2}$. A slightly modified version of this model, taking into account skew and distortion of the lenses, is used in [1], here the projection process is split into three steps:

1. Projection onto the normalized image plane:

$$(x',y')=\left(\frac{X}{Z+r\varepsilon},\frac{Y}{Z+r\varepsilon}\right), \quad (2)$$

2. Application of radial and tangential distortion:

$$(x'',y'')=\left(k_d x'+2k_3x'y'+k_4(y'^2+3x'^2),\, k_d y'+k_3(x'^2+3y'^2)+2k_4x'y'\right), \quad (3)$$

where: $k_d=1+k_1\rho+k_2\rho^2+k_5\rho^3, \rho=x'^2+y'^2$,

3. A pinhole projection:

$$(u,v)=(\gamma_x x''+\alpha y''+c_x,\gamma_y y''+c_y) \quad (4)$$

The inverse transformation is then obtained applying the inverse of each step in reversed order.

A general model, applicable for multiple types of catadioptric sensors is presented in [5], and extended in [6]. Instead of using the concrete equation of the mirror transform, these papers present an approach, where the relation between a 3D space point and a point in the sensor plane is approximated by a Taylor polynomial expressed in the form:

$$(X,Y,Z)=\lambda\left(x'',y'',f\left(\sqrt{x''^2+y''^2}\right)\right)$$
$$f(\rho)=a_0+a_1\rho+a_2\rho^2+\ldots+a_n\rho^n \quad (5)$$

This polynomial includes the mirror transformations, and any distortions induced. A pixel in the omnidirectional image and a point in the sensor plane are related by an affine transformation:

$$(x'',y'')^T=\mathbf{A}(u,v)^T+\mathbf{t}\ . \quad (6)$$

The affine transformation accounts for the small misalignments between the axes, and the information loss due to the digitizing process.

B. Single Camera Calibration

Since previous calibration procedures yielded similar results for the models presented in [7] and [6] we decided to employ both of them to choose the optimal one for our case.

The calibration can be done with the help of planar chessboard grids of known geometry placed at various distances and angles from the camera. The corners of each cell of the chessboard are extracted with sub-pixel accuracy; the parameters of the corresponding model are then estimated from the corner points in the training set.

For the first model, corresponding to equations (2),(3) and (4), the parameters estimated are the extrinsic parameters for each chessboard view (4 parameters for the rotation represented by quaternions, and 3 parameters for the translation: $\{q_i^1,q_i^2,q_i^3,q_i^4,t_i^1,t_i^2,t_i^3\}$, the mirror parameter ε, the distortion coefficients $\{k_1,k_2,k_3,k_4,k_5\}$, and the pinhole parameters $\{\alpha,\gamma_1,\gamma_2,c_1,c_2\}$, giving 7*k*+11 parameters, where *k* is the number of planar grids. After choosing the initial values of the parameters carefully, Mei [3] uses the Levenberg-Marquardt approach for nonlinear optimization of the parameter values.

The second calibration method developed by Scaramuzza, described in [16] uses the projection model given by equations (5) and (6). The parameters to estimate are the extrinsic parameters (7 for each chessboard view), the coefficients of the polynomial *f*, and the elements of the affine transformation {*c,d,e*}, where $\mathbf{A}=\begin{pmatrix}c & d\\ e & 1\end{pmatrix}^{-1}$, and the image center , with the translation $\mathbf{t}=-\mathbf{A}\,(c_x,c_y)$.

The algorithm proposed by the authors of [16] for finding these values includes a rough initial estimate for the intrinsic parameters, an iterative linear estimation of the extrinsic and intrinsic parameters successively, and a final nonlinear refinement over all parameters.

C. Stereo configurations and calibration

Two basic configurations of the cameras were considered: a vertical and a horizontal placement (Fig.1). The horizontal placement is more suitable for installment on a moving platform (for example on a car), but makes the rectification process more difficult, in the sense that the epipolar curve of a point in the first image is not an epipolar line but an ellipse on the surface of the cylinder.

In the case of a vertical configuration a point corresponds to a degenerated ellipse, which in case of proper alignment appears as a straight line on the second image. The advantages of this method, which are clearly reflected by the experimental data are a higher precision in rectification and a smaller stereo reprojection error, the main disadvantage being the difficulty of developing a practical system due to space limitations.

We employed two different stereo calibration methods: computation of the relative rotation and translation between the two cameras and the absolute parameters in the world coordinate frame.

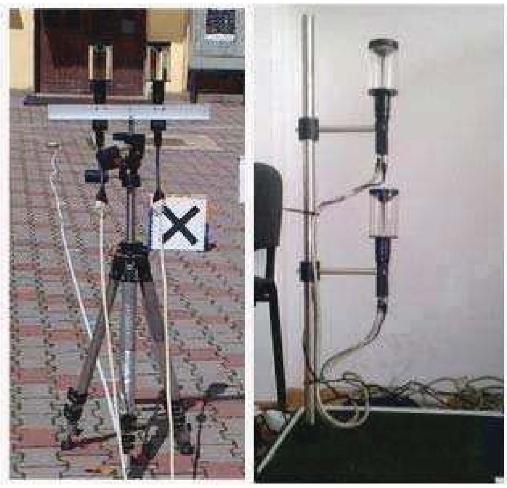

Fig. 1. Horizontal (left) and vertical (right) camera configurations

1) Computation of relative parameters

This method computes the relative position of the two cameras, and was mainly used to evaluate the quality of the calibration, measuring the symmetric reprojection error in both images.

Here we use as input the same set of corner points extracted from both images, and the extrinsic parameters estimated from the calibration of each individual camera. The scaling parameter for a given point is estimated by knowing the exact distance between neighboring corners of the chessboard. The method is mainly based on a least squares minimization.

Assuming that the two cameras are related by a rotation and a translation, one can write:

$$\mathbf{R}\mathbf{X}_1 + \mathbf{t} = \mathbf{X}_2 \tag{7}$$

where $\mathbf{X}_1$ represents the coordinates of a fixed point in the frame of the first camera, and $\mathbf{X}_2$ in the frame of the second.

The set of corner coordinates estimated at the individual calibration phase gives us a set of points which will be the coefficients of the over determined system of equations.

The solution in least squares sense is[2]:

$$[\mathbf{R}\ \mathbf{t}] = (\mathbf{A}^{\mathrm{T}} \times \mathbf{A})^{-1} \times \mathbf{x}_2,\ \mathbf{A} = [\mathbf{x}_1\ \mathbf{I}_{1,\mathbf{n}}], \mathbf{x}_{1,2} \in \mathbf{M}_{3,\nu} \tag{8}$$

where n is the number of corners extracted.

Since in this phase we treat each parameter of the rotation individually, the result does not always possess the properties of a true rotation matrix, since the training data is not completely free of errors. We surpassed this problem by using the polar decomposition of the matrix, and re-computing the translation by taking the mean of the difference between the coordinates in the second camera frame and the rotated coordinates in the first frame, providing thus the true rotation closest to our solution in the least squares sense.

Since some of the points are extracted with smaller precision, and some of the extrinsic parameters are estimated erroneously we use a RANSAC-type method to robustly estimate the rotation and the translation, computing the result for several subsets of points and choosing the set of points for training which gives the maximum number of inliers for a given error threshold. The threshold and the size of the training set are established experimentally.

2) Computing the absolute rotation and translation

This method is described in detail for perspective cameras in [19]. In this case we take a world coordinate system centered in a fixed point at the base of our stereo camera system, and measure the absolute distance on the three main axes to targets in the shape of an "X". We use the two equations:

$$\begin{aligned} \mathbf{R}_1 \mu_1 \mathbf{X}_1 + \mathbf{t}_1 &= \mathbf{X}_\mathrm{W} \\ \mathbf{R}_2 \mu_2 \mathbf{X}_2 + \mathbf{t}_2 &= \mathbf{X}_\mathrm{W} \end{aligned}, \tag{9}$$

where $\mathbf{X}_1$ and $\mathbf{X}_2$ are the coordinates of the target centers (extracted with sub-pixel accuracy), reprojected into space, and μ_1 and μ_2 are unknown scale factors. Since there are three elementary equations for each vector equation, we can eliminate the scale factor leaving two equations for each target center, and each camera. The rotation and translation parameters will then be estimated using these two equations for each target, together with the constraints of the rotation matrix individually for each camera. The optimal parameters are found using a Gauss-Newton minimization over all equations.

After finding the absolute rotation and translation, the relative position of the two cameras can be described as:

$$\mathbf{R} = \mathbf{R}^{-1}_2 \mathbf{R}_1 \ \textit{and}\ \mathbf{t} = \mathbf{R}_2^{-1}(\mathbf{t}_1 - \mathbf{t}_2) \tag{10}$$

3) The role of parameters in 3D reconstruction

After computing the absolute rotation and translation for both cameras, we can estimate the coordinates of any point with known pixel coordinates in the two images in the world coordinate frame, a procedure also known as triangulation. We employed two triangulation methods and compare their results. The first is the basic linear triangulation algorithm presented in [8], and the second the optimal triangulation method described in [12]. They are presented in detail in Chapter V.

III. Image unwrapping

Since we desire to use algorithms developed for perspective cameras in the future, without significant modifications, a necessary step is transforming the omnidirectional image into an equivalent panoramic image (unwrapping).

Two approaches were considered for unwrapping: the first method uses a simple circle to rectangle mapping, where a column on the rectangle will correspond to a radial line in the omnidirectional image, and each row to a concentric circle. Although this method is relatively fast, it doesn't take into account the distortions introduced by the mirror. The second method uses the equation of the mirror to reproject the images into the 3D space centered in the camera, on the surface of a cylinder. Since reprojecting the whole image is computationally slow, saving the whole mapping into memory is advised. This method is presented and optimized for memory efficiency using the eight-way symmetry of images in [11].

Regarding rectification, several methods were considered. Since we use the unwrapped panoramic images, rectification methods for perspective images are suitable.

We propose the idea of transforming not only the image, but also the projection equations onto their perspective equivalent, to avoid working with the nonlinear projection model of the omnidirectional camera. When using the cuboid reprojection of an omnidirectional image, we create four imaginary projection planes, or perspective sensor planes, onto which the image is projected from every direction. The only disadvantage of the method lies in eventual continuity problems, coming from parameter estimation errors on the borders of the four cameras. Since we know the exact correspondence among the border points on each perspective image an interpolation method can be used to correct for these continuity errors later (which should be only minor if the parameters were estimated correctly). Contrary to the case of using four physical perspective cameras we obtain in the end a continuous surrounding 3D image, without the need of stitching and searching for correspondences in two neighboring images. Each virtual camera accounts for 90 degrees of the viewing plane, and the new projection becomes a simple pinhole model.

For computation of the extrinsic parameters we can improve our approximation, if we compute these individually for each of the cameras, using only the set of relevant points. The clear advantages of this method can be seen now: one of: the perspective algorithm for the estimation of the extrinsic parameters can be used directly without further approximations.

IV. Stereo Rectification

Since in real applications searching for the correspondence of a point from the first image through the second image as a whole is too expensive, rectification is necessary to align the corresponding points on the same row in the case of the horizontal alignment, and on the same column in the case of the vertical alignment.

To rectify the images we need to know first their epipolar geometry. While in case of perspective cameras, the image of a ray emanating from an image point is a line in the corresponding image, in case of catadioptric cameras, the image becomes a conic. Reprojecting into 3D space, in case of a horizontal configuration an epipolar curve will be represented by an ellipse on the surface of the cylinder, corresponding to a curve on the panoramic image, starting and ending at the same height, while for vertically placed cameras we have to deal with simple lines.

Among the most popular rectification methods for perspective cameras we can mention the ones presented in [8] respectively [11]. They both require knowledge of the fundamental matrix **F** relating x_2 and x_1, the corresponding points from the two views:

$$\mathbf{x}_2^{\mathrm{T}}\mathbf{F}\mathbf{x}_1 = 0\,. \tag{11}$$

Similar approaches based on the fundamental matrix, applied to panoramic images are tackled in [12] resp [14]. These are applied directly to the panoramic images, computing a homography for each image individually, taking into account only the image-point correspondences, independently of the computed rotation and translation in world coordinates. We measured the rectification error for homographies computed for the whole images as well as for homographies computed for corresponding parts of the images, reducing the rectification error.

For computing the fundamental matrix we compared different methods described in literature [10], including the eight-point algorithm, least-squares estimation and RANSAC. In each case the coordinates were first normalized. For the rectification we used the result obtained with the RANSAC algorithm, which provided the best tradeoff between robustness and accuracy on the training set. This random sampling algorithm works with a threshold error limit imposed on the inliers. We establish this limit adaptively by iterating through a given range of limits, giving a correct fundamental matrix, and a reasonably high number of inliers, and plotting the learning curve of rectification error as a function of this threshold. The chosen threshold will be the one lying at the point, where the second derivative of the learning curve reaches its maximum, practically choosing the highest threshold that still provides a reasonable error rate.

Besides the two rectification methods mentioned above we conducted experiments with another method described in [4].

The second method is specially developed for omnidirectional images, and instead of using the classical epipolar geometry of perspective images, generates intersections of the imaginary cylinder around the omnidirectional image, with a set of planes. The planes are generated in such a way that they intersect the two cylinders maximally, the intersection curves becoming the new rows in the resulting image. Although this process, also called epiline sampling, respects more rigorously the epipolar geometry of catadioptric images, computing the rectification is significantly slower, and requires saving the whole mapping between the rectified and original images, instead of a 3x3 homography matrix.

V. Stereo Matching and Reconstruction

Stereo matching algorithms aim to find corresponding points between two images of the same scene, usually by using an objective function representing the dissimilarity between two points which it aims to minimize in combination with enforcing some kind of continuity of the correspondences throughout the image. The matching function can range from the simple Euclidean distance, through cross-correlation, normalized cross-correlation, to other complex distance metrics for a fixed or dynamically sized window. Often the images are first transformed to obtain tolerance to luminosity, contrast, rotation and other changes in the image.

Regarding the enforcement of continuity we can distinguish local, semi-global and global algorithms.

Local algorithms use a small to medium sized window around the featured image point, and evaluate the matching function in this neighborhood. Global matching methods use ordering of pixels, unique correspondence constraints, and smoothness constraints to find an optimal pixel-wise matching of the two images. While local algorithms provide excellent speed and global algorithms excellent quality, none of them provide overall satisfying results. Semi-global matching algorithms provide significantly better results than local algorithms, but at the same time are feasible for a real-time application. They usually impose constraints only in one direction, reducing the algorithm complexity and execution time, but ensuring smoothness in the direction of epipolar lines. The semi algorithm introduced in [9] performs optimizations in several directions to simulate the performance of global matching methods, allowing also real time execution at the same time.

The method employed by us is an improved version of the SGM algorithm by Hirschmuller, and is described in [15]. The authors used as matching function, the hamming distance of the census transform of the image, applied for a 9x9 window. The census transform can be computed as:

$$\mathbf{R}_{T(x,y)} = \otimes_{i=-n,n} \otimes_{j=-m,m} \xi\big(I(x,y), I(x+i, y+j)\big),$$
$$\xi(x,y) = \begin{cases} 0, \mathit{if}\ \ x < y \\ 1, \mathit{if}\ \ x = y \\ 2, \mathit{if}\ \ x > y \end{cases} \tag{12}$$

The SGM algorithm performs an energy minimization in several directions, to provide a smoothness to the disparity function. The original authors of SGM recommended minimization in 8 or 16 directions, however it was proven in [15] that 4 directions provide satisfying results, with no significant loss of quality versus 8 or 16 directions, but clear gains in execution speed. The equation leading the minimization in four directions is:

$$E(D) = \sum_{p} C(p, D_p) + \sum_{q \in N_p} P_1 T\big[|D_p - D_q| = 1\big] + \sum_{q \in N_p} P_2 T\big[|D_p - D_q| > 1\big] \tag{13}$$

The algorithm searches the point along an epipolar line, which minimizes expression (13), where *C* is the matching cost obtained by the hamming distance, P_1 is a penalty term for small changes in disparity, and P_2 is a higher penalty for discontinuities in disparity. By not only trying to minimize the cost of matching, but introducing penalties for discontinuities, a smoothness and continuity of reconstructed surfaces is obtained, together with a smaller error rate.

For the simple pinhole perspective model, brought to canonical form by calibrated rectification, 3D reconstruction is possible by computing the direction of the ray projected onto a given pixel, and computing the distance by measuring the disparity between the corresponding points in the two images.

This method however does not work well for uncalibrated stereo configurations. Since in 3D space the geometry of the virtual cameras does not equal a pure translation, and also because the rectifying homographies induce a skew factor in the projection matrix, the distance of the points is not estimated correctly using the pinhole model. We considered two alternative solutions, a homogeneous method based on the direct linear transformation and an inhomogeneous linear triangulation algorithm (see [12]).

For two corresponding image points and camera matrices $\mathbf{P}_1$ and $\mathbf{P}_2$, and a 3D point **P**, we have $x_1 = \mathbf{P}_1\mathbf{P}$ and $x_2 = \mathbf{P}_2\mathbf{P}$, taking the fact that the vector product of two vectors with the same direction is 0, we obtain the linear system:

$$\mathbf{AX} = 0, \mathit{where}\ \ \mathbf{A} = \begin{bmatrix} x_1 p_1^{3^T} - p_1^{1^T} \\ x_2 p_1^{3^T} - p_1^{2^T} \\ y_1 p_2^{3^T} - p_2^{1^T} \\ y_2 p_2^{3^T} - p_2^{2^T} \end{bmatrix} \tag{14}$$

The inhomogeneous method represents the vector **P** as $[X,Y,Z,1]^{\mathrm{T}}$, and the set of homogeneous equations from equation (14) reduces to a set of inhomogeneous equations, which can be solved with linear least squares optimization, using the normal equation.

The homogeneous method uses singular value decomposition to solve the linear system from equation (14). The advantage of this method is that it also works for points close to the origin (with *0 Z* coordinate, or in our case with *0 X* or *Y* coordinate), contrary to the inhomogeneous method, which for points close to the origin on the *Z* axis gives erroneous results. Implementation of the singular value decomposition is however slower and more complicated than implementing the normal equation.

VI. Experimental results

A. Single Camera calibration

For both calibration methods we performed several experiments, retaining the results which gave the smallest reprojection error, both individually as well as after performing stereo calibration. The images with unsuccessful corner extraction were removed for both methods individually.

We performed the first calibration according to the unified projection model, using Mei's toolbox [13], with 88 images of a planar grid of 8x8 squares, placed at different angles and distances from the two cameras. As one can see in Table I, the average reprojection error for the training images was around 0.38 pixels, with a standard deviation of the same order.

The second calibration method, using Scaramuzza's omnidirectional camera calibration toolbox [16] was performed on the same number of images, with the same grid pattern, for polynomials of the 3rd and 4th degree. Although this method is more general and can be applied for different types of catadioptric sensors, in our case a slightly higher average reprojection error was obtained of around 0.5 pixels (Table II).

TABLE I. CAMERA PARAMETERS (MEI)

Significance	**Variable**	**Value(Left)**	**Value(Right)**	**MU**
Focal length	μ_1 μ_2	437.8133 437.4077	422.46367 422.89815	mm
Principal point	c_1 c_2	699.1116 506.6615	696.72415 526.37908	pixel
Mirror coef.	ε	1.4377	1.37719	-
Skew	α	0.0000	0.0000	-
Distortion coefs.	k_1 k_2 k_3 k_4 k_5	0.2233 0.5281 0.0009 0.0039 0.0000	0.2323 0.3333 0.0016 0.0011 0.0000	-
Reprojection error	$\sqrt{e_x^2 + e_y^2}$	0.38369	0.38155	pixel

TABLE II. CAMERA PARAMETERS (SCARAMUZZA)

Significance	**Variable**	**Value(Left)**	**Value(Right)**	**MU**
Image center	c_x c_y	691.97975 507.08323	707.79377 530.98057	pixel
Affine transformatio n	c d e	1.000013 -8.32780 -9.82916	1.0000003 -7.4254888 -4.29391347	-
Taylor polynomial	a_0 a_1 a_2 a_3	1.681829282 0 0.0000152558 -0.0000000044	-1.713962922 0 0.0000144366 -0.000000003	-
Reprojection error	$\sqrt{e_x^2 + e_y^2}$	0.56804	0.623365	pixel

B. *Relative camera parameters and calibration verification*

Since we discovered that the calibration results are strongly dependent on the size and quality of the training set, we used the computation of relative reprojection error as a measure of quality for the single camera calibrations. As described in Chapter II we obtain this measure, by taking the extrinsic parameters computed at the calibration phase, computing the relative position of the two cameras which minimizes the Euclidean distance between the two of them, and computing the average distance in pixels between the points in the frame of one camera transformed in the frame of the other, and projected on the image, and the extracted pixel point for its correspondent.

Fig. 2 illustrates the reprojection errors of 4 different calibrations (in vertical and horizontal configurations) with the first model, after computing the relative coordinates with a least squares minimization, and after accounting for the properties of the rotation. One can see clearly that when increasing the number of training images the calibrations become more accurate and the relative reprojection error decreases drastically.

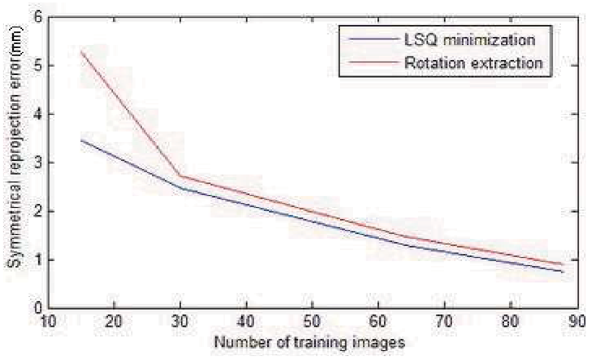

Fig. 2. Stereo reprojection error

For accurate calibration the compensation for the symmetry of the rotation increases the error only slightly.

The best result obtained (for 88 training images and a vertical configuration) was a reprojection error of 0.74 pixels before the polar decomposition, and of 0.88 pixels after. The average distance in 3D in the frame of a camera between a point and its correspondent rotated and translated from the frame of the other is 19 mm.

The results were also similar for the second calibration method, but since this is more general than the first, we decided to conduct our experiments further with Scaramuzza's model.

We also tested the extrinsic calibration method for both the horizontal and the vertical configurations. 13 x-shaped targets were used, of different height and size, placed at different distances, the target centers were extracted with subpixel precision, and their position was measured in the world reference frame using a high-precision GPS sensor with a precision of 2 cm.

The mean reconstruction error was between 5.2% and 8% for the test set for the four virtual cameras. The absolute reconstruction error grew quadratically with the distance as expected in stereo reconstruction algorithms (Fig. 3).

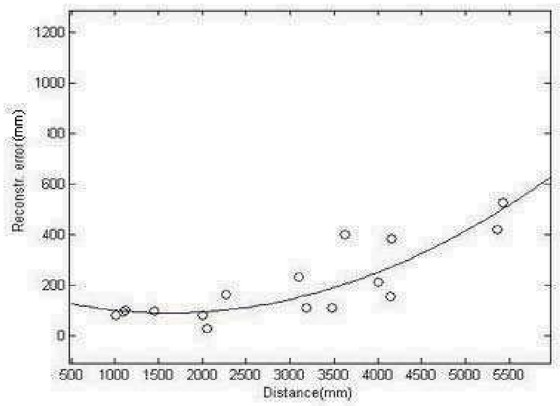

Fig. 3. Stereo reconstruction error for the test set

C. Fundamental matrix computation and rectification results

We compared the different algorithms for the computation of the fundamental matrix using the set of points extracted at the calibration phase, both for the cylindrical and the perspective unwrapping techniques. This includes the 8-point algorithm, with RANSAC, with Least-squares and with the L-meds method. RANSAC was proven to perform significantly better than the 8-point algorithm, and similarly to the L-meds method. The least-squares algorithm performed even worse than the 8-point method, explainable by a significant number of outliers which skewed the result. One can also see that for the perspective reprojection the fundamental error on the parts of the image is significantly smaller than for the whole image.(Table III).

TABLE III. FUNDAMENTAL MATRIX COMPUTATION RESULTS

Method	**Error on whole image [pixel]**	**Avg. error on quarters [pixel]**
8-point alg	2.0949	0.7199
RANSAC	1.69402	0.4323
Least squares	2.3888	1.1091
L-meds	1.6432	0.4202

We decided to use for rectification the value of the fundamental matrix obtained by the RANSAC estimation, since there was no significant difference versus the L-meds method, and the number of inliers was slightly higher. The test set for measuring the error was separated from the training set, and the number of inliers was grater than 99% of the test data. The results obtained for different methods on quarters on the image are comparable to the ones obtained for simple perspective images in urban scenes presented in [20].

We tested first of all the methods for rectification presented by Hartley in [8] and Ma in [12]. The errors from the fundamental matrix naturally were transmitted to the rectification step, not increasing significantly. The linear methods had a superior performance relative to the epiline sampling method (Table IV), which can be explained by the fact that while the first two are using purely image properties, the second also uses the model parameters which might also induce some errors. The rectification transforms were saved as lookup tables, and a real-time mapping is performed in the moment of image acquisition.

TABLE IV. RECTIFICATION ERROR

Method	**Error on whole image [pixel]**	**Avg. error on quarters [pixel]**
Hartley [8]	1.72386	0.55119
Ma [12]	1.658924	0.59834
Epiline sampling [4]	2.3888	-

We have also performed experiments with rectification methods based on the extrinsic parameters of the cameras in the 3D space, but the higher reprojection error, and the reduced number of matchings in the stereo matching phase determined us the adapt Hartley's method.

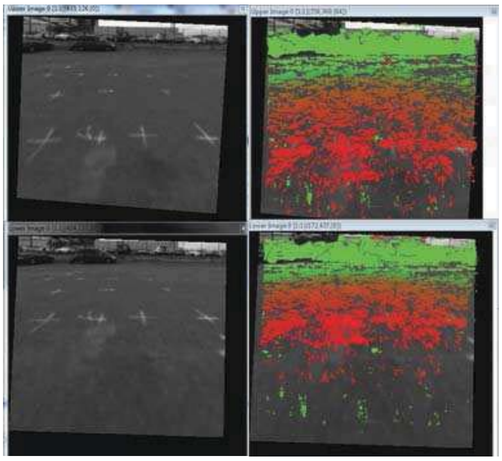

Fig. 4. Stereo disparities (growing from green to red)

D. Stereo matching

A GPU implementation of the SORT-SGM algorithm was directly applied to the rectified image pairs.

As one can see the results of the matching were quite satisfying, obtaining a dense stereo image, where correspondences exist, with few erroneous matches (see Fig.4).

Finally the two presented reconstruction methods were applied, the pinhole model, and the linear triangulation algorithm. Several subpixel estimation were tested together with the two algorithms, we present here the best results obtained for a parabolic interpolation method. Although the linear triangulation algorithm has a smaller error in estimating the correct distances, it didn't worked well from the point of view of continuous surfaces with subpixel estimation methods, suggesting for the necessity of future research in finding a proper interpolation method preserving continuity for this type of depth estimation (see Fig. 5).

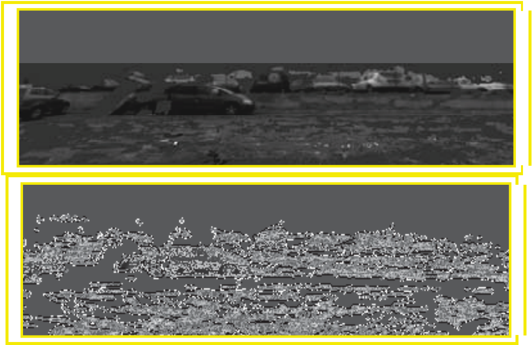

Fig. 5. Depth estimation with the triangular method before (upper image) and after subpixel estimation (the left image rendered in 3D space)

The pinhole method preserves continuity even after subpixel estimation (see Fig. 6.), but while the reconstruction error for the triangular method corresponds to the values on Fig. 2, in case of the pinhole method this error grows significantly worse for large distances.

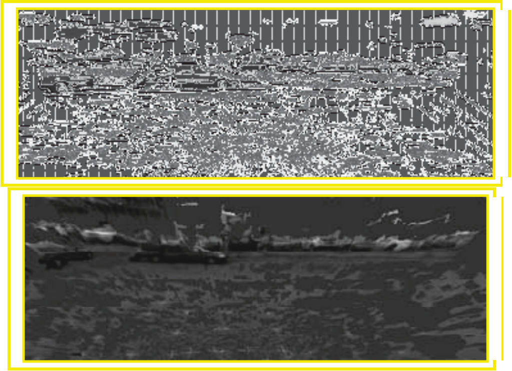

Fig. 6. Reconstruction with the pinhole model before and after subpixel estimation (the left image rendered in 3D space)

VII. Conclusions

We presented in this paper an extensive study about stereo calibration, rectification and reconstruction for omnidirectional images, with a few suggestions for improvement in terms of speed and performance, the most important being the transformation of the two omnidirectional images in a system of 4 pair of perspective ones. The uncalibrated rectification methods provided us with a rectification error comparable to the ones of the perspective cameras, however without the need for a complex algorithm at the time of image stitching. The main problem at the moment consists in the fact that during reconstruction the pinhole reconstruction model has a high reprojection and depth estimation error, while the linear triangulation method does not work well together with the traditional SGM and subpixel estimation methods. We suggest further experiments for improving the reconstruction results.

A first suggestion is changing the rectification method to a calibrated one, creating a canonical configuration such that the pinhole model can be correctly applied with higher precision. This implies a greater rigidity in the hardware configuration, such that there are no changes in the extrinsic parameters (not even minor ones), since they can greatly influence the stereo correspondence process..

A second improvement idea is changing the stereo matching and the subpixel estimation algorithms such that the continuities apply for the linear triangulation model, not only for the pinhole model.

Acknowledgment

This paper is written within PAN-Robots project. 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] B. Micusik, T. Pajdla. "Estimation of omnidirectional camera model from epipolar geometry." *Proceedings of the 13th Scandinavian conference on Image analysis.* Halmstad, Sweden, 2003.

[2] Bjorck, Ake. *Numerical Methods for Least Squares Problems.* Philadelphia: Society For Industrial and Applied Mathematics, 1996.

[3] C. Mei, P. Rives. "Single View Point Omnidirectional Camera Calibration from Planar Grids." *2007 IEEE International Conference on Robotics and Automation.* Rome, 2007. 3945-3950.

[4] Chen, Wang, Wei Xu, Zhihui Xiong, and Maojun Zhang. "View Synthesis for Realistic Virtual Walk Through Based on Omni-directional Images." *The International Journal of Virtual Reality* 8, no. 4 (2009): 87-92.

[5] D. Scaramuzza, A. Martinelli. "A Flexible Technique for Omnidirectional Camera Calibration and Structure from Motion." *Proceedings of IEEE International Conference on Computer Vision Systems.* 2006.

[6] D. Scaramuzza, A. Martinelli, R. Siegwart. "A Toolbox for Easily Calibrating Omnidirectional Cameras." *International Conference on Intelligent Robots and Systems.* 2006.

[7] Geyer, C., and K. Daniilidis. "A Unifying Theory for Central Panoramic Systems and Practical Implications." *6th European Conference on Computer Vision.* Dublin, 2000. 445-461.

[8] Hartley, R., and A. Zissermann. *Multiple View Geometry in Computer Vision.* New York: Cambridge University Press, 2004.

[9] Hirschmuller, H. "Accurate and efficient stereo processing by semi-global matching and mutual information." *IEEE Computer Society Conference on Computer Vision and Pattern Recognition.* 2005.

[10] Huang, Jing-Fu, Lai Shang-Hong, and Chia-Ming Cheng. "Robust Fundamental Matrix Estimation with accurate outlier detection." *Journal of Information Science and Engineering* 23 (2007): 1213-1225.

[11] J. Lei, X. Du, Y.-F. Zhu, J.L.-Liu. "Unwrapping and stereo rectification of omnidirectional images." *Journal of Zhejiang University* 10, no. 8 (2009).

[12] Ma, Y., S. Soatto, J. Kosecka, and S. Sastry. *An Invitation to 3-D vision.* Springer, 2003.

[13] Mei, Christopher. *Omnidirectional Calibration toolbox.* 2007. http://www.robots.ox.ac.uk/~cmei/Toolbox.html.

[14] P. H. S. Torr, D. W. Murray. "The Development and Comparison of Robust Methods for Estimating the Fundamental Matrix." *International Journal of Computer Vision*, 1996.

[15] Pantilie, Cosmin, and Sergiu Nedevschi. "SORT-SGM: SubPixel Optimized Real Time Semi-global Matching for Intelligent Vehicles." *IEEE Transactions on Vehicular Technology* 61, no. 3 (2012).

[16] Scaramuzza, Davide. *OcamCalib toolbox.* 2006. https://sites.google.com/site/scarabotix/ocamcalib-toolbox.

[17] Svoboda, T., and T. Pajdla. "Epipolar Geometry for Central Catadioptric Cameras." *International Journal of Computer Vision* 1, no. 49 (2002): 23-37.

[18] Svoboda, Thomas. *Central Panoramic Cameras, Design, Geometry, Egomotion, Phd Thesis.* Prague: Faculty of Electrical Engineering, Czech Technical University, 1999.

[19] T. Marita, F. Oniga, S. Nedevschi, T. Graf, R. Schmidt. "Camera Calibration Method for Far Range Stereovision Sensors Used in Vehicles." *Intelligent Vehicles Symposium.* Tokyo, 2006.

[20] X. Zhihui, C. Wang, Z. Maojun. "Catadioptric Omni-directional Stereo Vision and Its applications in moving objects detection." *Computer Vision*, 2008.