

Reconstruction of Three Dimensional Models of Environments with a Mobile Robot*

Janelcy Alferes

*CARTIF Technology Center
Parque Tecnológico de Boecillo, parcela 205
Boecillo, Spain
janalf@cartif.es*

Eduardo Zalama and Jaime Gómez-García-Bermejo

*Department of Systems Engineering and Control
University of Valladolid
Paseo del Cause s/n 47011. Spain
ezalama@eis.uva.es jaigom@eis.uva.es*

Abstract - - In this paper a method is presented for the three-dimensional reconstruction of environment, using a mobile platform equipped with an onboard laser scanner (for the scene geometry acquisition), and a webcam (for the texture registration). The used algorithm is based on calculating the 3D planes in the scene upon the measured 3D points, through the normal calculation, the projection of the points onto the found planes, and the 2D triangulation of the projected points. This way, the plane regions and the corresponding contours in the model are obtained. Concerning the texture, images are assigned to the plane regions, and the complete scene is built through a stitching algorithm. Images and 3D points are projected onto a sphere, in order the information loss on free-form objects, and the distortion caused by points improperly assigned to planes, to be reduced.

Index Terms – Scanner laser, image mosaics, 3D models.

I. INTRODUCTION

2D maps of the environment are suitable in many applications. However, a more precise and detailed information is some times required. 3D maps are advantageous in these cases. For example, localization and navigation of mobile robots, architectural design, and automatic environment reconstruction for augmented reality benefit from 3D maps. 3D reconstruction upon artificial vision [7],[4],[3],[20],[2] often leads to errors in the resulting models. The use of wheeled mobile robots with onboard laser scanners, and cameras for texture registration, is an alternate solution [25],[1],[24],[7]. In [9], a mobile platform equipped with two scanners is used. In [23] and [22], a platform with a 2D scanner and a servomotor is used, so that a more dense 3D measurement is attained, but the proposed algorithm requires some scene information, and objects are approached through boxes. In [15], the techniques developed in [23] are combined with an ICP (iterative closest points) based algorithm, but the obtained information on the scene details is poor. In [16], RANSAC (Random Sample Consensus) and ICP methods are combined, but the model understanding requires previous knowledge about the scene. Recently, a method, which uses a motorized laser scanner and a webcam, has been developed in the CARTIF Tech. Center [5][14]. However, free-form object information is lost. A method for reducing free-form object distortion, based on the projection of 3D points and texture on a sphere, is proposed in present paper.

The paper is organized as follows. The acquisition platform is first described. The proposed approach is described in section III. Experimental results are discussed in section IV, and conclusions are summarized in section V.

II. DESCRIPTION OF THE MOBILE PLATFORM

The wheeled mobile robot platform is shown in figure 1. A Sick laser scanner [10] is used for 3D measurement. The scanner provides the distance “ r ” and the corresponding angle “ θ ” to points within a 180° plane field-of-view. A complete 3D swept of the scene is obtained through attaching the Sick to a stepper motor-driven, 90°-tilt platform. Texture information is provided by a webcam, mounted on a 180°-tilt platform attached to the Sick.



Fig. 1 Mobile Platform used for the data acquisition

III. ALGORITHM FOR ENVIRONMENT MODELLING

The modelling process is presented in figure 2.

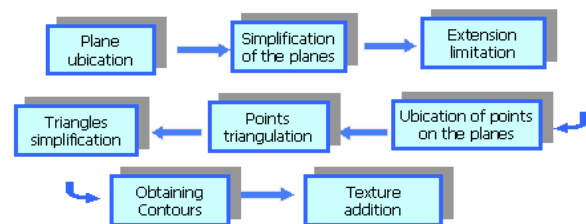


Fig. 2 Algorithm to obtain the model of environment.

This process comprises the geometry reconstruction and the texture addition.

A. Geometry Reconstruction

The scene geometric model is obtained from the 3D point cloud measured by the laser. The scene is modelled through a set of plane regions [14]. The algorithm provides the surface normal direction at each measured point (i.e. the “local normal”). This direction is obtained from the smallest variance direction of points in a size-adaptive, square box (centred at the current point).

* This work is partially supported by Ministry of Science and Technology (project number DPI2002-04377-C02-01), Fomento ministry (project number FOC-002-4) and the Local Government in Castilla y León (Spain).

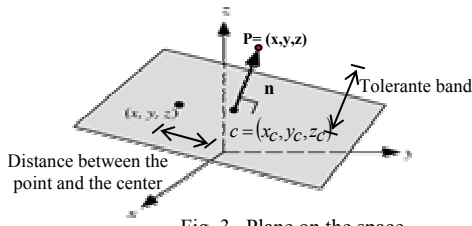


Fig. 3. Plane on the space.

The Hessian Normal form of the corresponding plane (see figure 3) is

$$n_x x + n_y y + n_z z - d_n = 0 \quad (1)$$

where $\vec{n} = (n_x, n_y, n_z)$ is the normal vector, and d_n is the distance from the plane to the coordinate system center. A single least-square fitting procedure is used to estimate the position of the plane

$$\min \left\{ \sum_{i=1}^N (n_x x_i + n_y y_i + n_z z_i - d_n)^2 \right\} \quad (2)$$

This leads to

$$\begin{bmatrix} S_{xx} & S_{xy} & S_{xz} \\ S_{xy} & S_{yy} & S_{yz} \\ S_{xz} & S_{yz} & S_{zz} \end{bmatrix} - \frac{1}{N} \begin{bmatrix} S_x & S_y & S_z \end{bmatrix} \begin{bmatrix} S_x \\ S_y \\ S_z \end{bmatrix} \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} = 0 \quad (3)$$

$$S_x = \sum_{i=1}^N x_i \quad S_{xx} = \sum_{i=1}^N x_i^2 \quad S_{xy} = \sum_{i=1}^N x_i y_i \text{ etc.}$$

which can be expressed in terms of the media and the variance (by dividing (3) over the number of points N):

$$\frac{E(X \cdot X) - E(X) \cdot \vec{n}}{\sigma^2(X)} \quad (4)$$

The obtained matrix, i.e. the ‘‘Space Variance Matrix’’, allows the measurement error to be modelled as a white noise. The three eigenvalues of this matrix are the main variances. If points belong to a plane, the ellipsoid generated by $\sigma^2(X)$ is plane-shaped, because $\sigma_3 \ll \sigma_2 \leq \sigma_1$ (given that σ_3 is the smallest eigenvalue). The associated eigenvector represents the most probable normal direction at the current point. The other two variances are associated to the two main plane direction. The obtained normal directions are used to group points into planes, by checking the vicinity, the plane occupation and the discarded points. The different plane regions within such planes are then obtained by a applying a common 2D Delaunay triangulation [6]. The contours of said regions are obtained by only keeping the boundary of the resulting meshes. Result is finally coded in VRML standard format in order to ease further operations.

B. Texture Addition

The texture addition algorithm undoes the image perspective and assigns regions within images, to the plane regions of the 3D model, upon a previous calibration process. Then, a stitching procedure (‘‘image mosaics’’)

allows the texture corresponding to each plane region to be obtained.

1) Camera Calibration

Camera calibration process [5] [9] involves the estimation of a set of intrinsic parameters (focal distance (f), image center position (cx, cy), distortion coefficients (k1,k2)); and a set extrinsic parameters (translation and rotation of the camera with respect to the world reference, i.e. T_x, T_y, T_z and angles α, β, γ respectively). The procedure is basically performed through (1) finding image coordinates of a set of known 3D control points; and (2) estimating intrinsic and extrinsic parameters through a fitting procedure. The well-known Tsai method [5] has been used in this work (see fig 4).

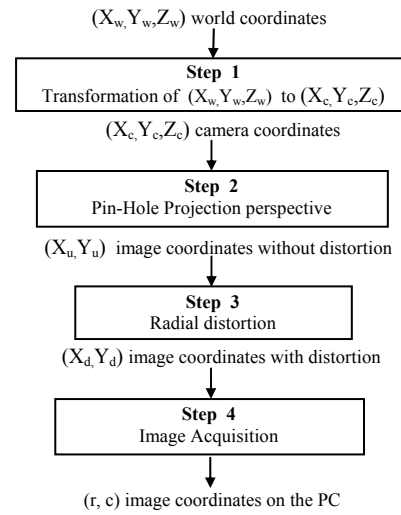


Fig 4. Camera Calibration

- Transformation from (X_w, Y_w, Z_w) to (X, Y, Z) is:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = R \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + T \quad R = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix} \quad T = \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} \quad (5)$$

- Transformation from (X, Y, Z) to (X_u, Y_u, Z_u) , using the Pinhole model [5],[9] becomes:

$$X_u = f \frac{X_c}{Z_c}, \quad Y_u = f \frac{Y_c}{Z_c} \quad (6)$$

- Transformation from (X_u, Y_u, Z_u) to (X_d, Y_d, Z_d) is:

$$X_d + d_x = X_u \quad ; \quad Y_d + d_y = Y_u \quad (7)$$

where:

$$d_x = X_d(k_1 r^2 + k_2 r^4 + \dots)$$

$$d_y = Y_d(k_1 r^2 + k_2 r^4 + \dots) \quad r = \sqrt{X_d^2 + Y_d^2}$$

- Transformation from (X_d, Y_d, Z_d) to (r, c) , the highest order terms in (7) being rejected, is:

$$r = r_0 + N_x \left(f \frac{X_c}{Z_c} \right) = r_0 + K_1 \left(\frac{X_c}{Z_c} \right) \quad ; \quad c = c_0 + N_y \left(f \frac{Y_c}{Z_c} \right) = c_0 + K_2 \left(\frac{Y_c}{Z_c} \right) \quad (8)$$

N_x, N_y are the scale factors. The calibration matrix is then obtained from (5) and (8):

$$\begin{bmatrix} nr \\ nc \\ n \end{bmatrix} = \begin{bmatrix} (r_0r_7 + k_1r_1) & (r_0r_8 + K_1r_2) & (r_0r_9 + K_1r_3) & (r_0T_3 + K_1T_1) \\ (c_0r_7 + K_2r_4) & (c_0r_8 + K_2r_5) & (c_0r_9 + K_2r_6) & (c_0T_3 + K_2T_2) \\ r_7 & r_8 & r_9 & T_3 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (9)$$

Calibration Matrix M

A fitting procedure allows the calibration matrix M to be obtained, from a set of N control points. (A set of black squares drawn on a white target, placed at different positions, has been used.) Extrinsic parameters are then updated according to the pan-tilt platform pose at each image take.

2) Perspective elimination

It is necessary to apply certain prosecution to the images to carry out the adjustment of the images to the model [21],[13]. In the present work, this is done by locating a virtual camera along the normal direction of the plane regions, so that perspective is corrected [5].

3) Image Mosaics

Fine alignment and merging of two or more images corresponding to a given plane region can be automatically performed (when common areas are present). The alignment parameters are found though a 8 parameter transformation matrix, obtained from the optimization “Levenberg-Marquardt” algorithm [21],[13]. In the case of two images A and B, 4 corresponding points are first determined, as shown in figure 5.

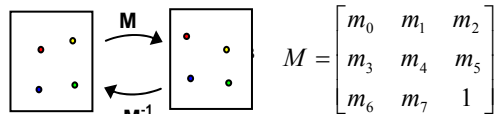


Fig 5. Transformation Image

The transformation of a (x,y) point in image “A” to (x',y') point in image “B” is obtained through

$$\underbrace{\begin{bmatrix} x & y & 1 & 0 & 0 & 0 & -x*x' & -y*y' \\ 0 & 0 & 0 & x & y & 1 & -x*y' & -y*y' \\ x & y & 1 & 0 & 0 & 0 & -x*x' & -y*y' \\ 0 & 0 & 0 & x & y & 1 & -x*y' & -y*y' \\ x & y & 1 & 0 & 0 & 0 & -x*x' & -y*y' \\ 0 & 0 & 0 & x & y & 1 & -x*y' & -y*y' \\ x & y & 1 & 0 & 0 & 0 & -x*x' & -y*y' \\ 0 & 0 & 0 & x & y & 1 & -x*y' & -y*y' \end{bmatrix}}_b \underbrace{\begin{pmatrix} m0 \\ m1 \\ m2 \\ m3 \\ m4 \\ m5 \\ m6 \\ m7 \end{pmatrix}}_A = \underbrace{\begin{pmatrix} x' \\ y' \\ x2' \\ y2' \\ x3' \\ y3' \\ x4' \\ y4' \end{pmatrix}}_p \quad (10)$$

Parameters in this equation are obtained through $\vec{p} = [A]^{-1} \cdot \vec{b}$, by a common SVD algorithm [14]. Fine alignment is then found by minimizing the luminance quadratic error along the overlapped pixels in A and B:

$$\sum_i e_i^2 = \sum_i [A'(x,y) - B(x,y)]^2 \quad (11)$$

where A' is the transformed image, and B the reference image. The intensity value A(x,y) is obtained through M⁻¹ followed by a bilinear interpolation. The Hessian matrix “A” and a gradient vector “b” are calculated by solving the following linear system, for the m parameters.

$$(A + \lambda I) \Delta m = b \quad (12)$$

Coefficients a_{kl} and b_k are normalized over the pixel number (n) of the overlapped region:

$$a_{kl} = \frac{\sum_i \frac{\partial e_i}{\partial m_k} \frac{\partial e_i}{\partial m_l}}{n}, \quad b_k = \frac{-\sum_i \frac{\partial e_i}{\partial m_k}}{n} \quad (13)$$

Finally, both images are mixed by using a pyramidal weighting function, according to [5],[21]

$$w(x,y) = \left(1 - \left|1 - \frac{2*invX}{ancho}\right|\right) \left(1 - \left|1 - \frac{2*invY}{alto}\right|\right) \quad (14)$$

(invX ,invY) are the inverse coordinates of image which determine the location of each pixel in the original image (before the stitching correction).

4) Detail of Free Form objects in the Scene

The above described procedure works suitably for giving general information about the environment (e.g. for obtaining 3D maps). However, some times improved realism is required. The said procedure fails in these cases, since free-form objects are not correctly modelled. In concrete, the algorithm deals with points assigned to planes. Points, which do not belong to any plane, are either simply discarded, or assigned to a near plane.

Given that the camera and the SICK describe spherical movements around a central point, a sphere is the most suited projection surface. So, the projection of the 3D points and the acquired images onto a sphere, is proposed. Projection at each point is defined by the optical center of the camera, (V_x,V_y,V_z) and the horizontal and vertical angles (θ, φ), as shown in figure 6.

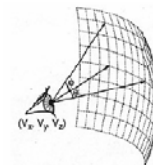


Fig 6. Viewpoint on a sphere

The points of the scene are imaged (by the camera) onto the image plane, located at a distance “P” from (V_x,V_y,V_z). Then, images are projected onto a sphere [11][26][18] for obtaining the corresponding texture, as shown in figure 7.

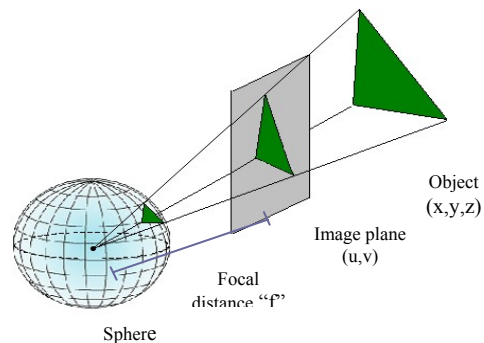


Fig 7. Projection of the 3D and 2D points on the sphere

Subsequently, the 3D points are grouped into triangles [6] which are projected on the same sphere. This allows a direct assignment of 2D texture coordinates to the 3D coordinates of the object to be carried out, thus minimizing perspective distortion. (The spherical geometric form lead 3D points to be less precisely mapped near the “poles” of the sphere.) The model is obtained when the stitching is carried out on the sphere. This approach results into a notable improvement of object perspective. Figure 8 shows an example on a fire extinguisher 3D model. We have selected the gnomonic projection [8][10] because it has the propriety that the grid straight lines, correspond to circles on the grid sphere, so there is a direct correspondence between the points.

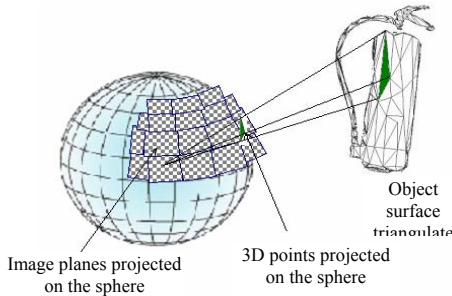


Fig 8. Projection of 3D and 2D extinguisher points onto the sphere.

5) 2D Images mapping on the sphere.

Any point on a sphere is defined by the radius (R), the horizontal sweep angle θ and the vertical sweep angle ϕ . We have to find the relations between these parameters, and the image points. First, we will assume that the optical axis crosses the image at $\theta = 0$; $\phi = 0$. Planes “xy” and “yz” showed in figure 9 are obtained.

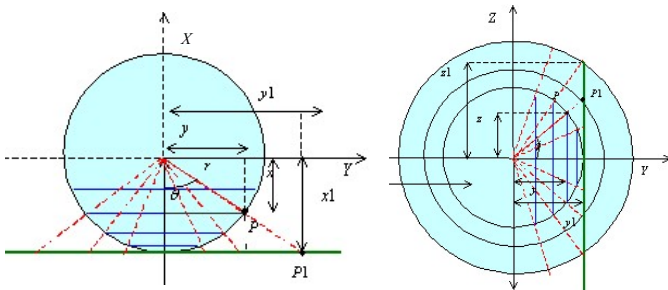


Fig.9 Plane xy and plane yz.

From information in planes xy and yz, a “mapped spatial” function set is obtained, which provide coordinates on the resulting image. Furthermore, horizontal (αh) and vertical (αv) angles of the used pan-tilt platform have to be added, thus leading to:

$$\begin{aligned} \phi &= a \tan \left[\frac{(R \cdot \sin(\alpha v) + z \cdot \cos(\alpha v)) \cdot \cos(\theta)}{R} \right] \\ \theta &= a \tan \left[\frac{(x \cdot \sin(\alpha h)) + (y \cdot \cos(\alpha h))}{R} \right] \end{aligned} \quad (15)$$

An interpolation algorithm is required to determinate the intensity level on the final image; we have used the bilinear interpolation [27]. Moreover, the image registration, i.e. finding the eight transformation parameters detailed in previous sections, is performed through “Levenberg-Marquardt” procedure. Stitching is carried out on $(\theta-\phi)$ planes obtained upon image projection on the sphere (see fig 10).

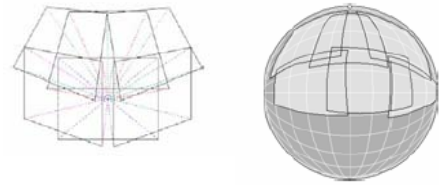


Fig 10. Concetric Images. Spherical Mosaic.

6) 3D point mapping on the sphere.

Texture mapping on a 3D surface requires 2D image points to be assigned to surface vertex. Points within 3D triangles can be readily mapped onto plane texture triangles. However, in present work we have to deal with spherical triangles, i.e. the region enclosed by three great circles on the sphere (a great circle is a circle on the sphere whose center is at the center of the sphere).

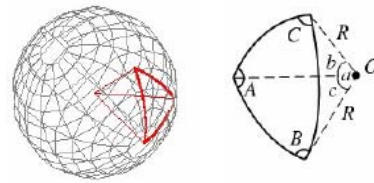


Fig11. Spherical triangle.

For R-radius spherical triangle, centred in $O=(0,0,0)$, with vertexes A, B and C, the following relations can be applied.

$$\begin{aligned} \cos A &= -\cos B \cdot \cos C + \sin B \cdot \sin C \cdot \cos a \\ \cos B &= -\cos C \cdot \cos A + \sin C \cdot \sin A \cdot \cos b \\ \cos C &= -\cos A \cdot \cos B + \sin A \cdot \sin B \cdot \cos c \\ \frac{\sin A}{\sin a} &= \frac{\sin B}{\sin b} = \frac{\sin C}{\sin c} \end{aligned} \quad (16)$$

This equation is the base for the 3D points triangulation on the sphere. This step of the algorithm is currently under development.

IV. EXPERIMENTAL RESULTS

We have tested the proposed method on real data. In the following experiment, a Cartif corridor has been modelled. Data have been acquired by placing the robot at two different locations: the beginning of the corridor, and 4 meters away along the axis robot direction (roughly parallel to walls). 3D has been measured through 0 to 1000 Sick tilt steps, a step corresponding to 0.09 degrees. Thus, 0 to 90° has been swept. The obtained point cloud is shown in figure 12.



Fig. 13. Point Cloud from Cartif Corridor

Figure 14 shows the first model, after having found the planes within the point cloud, and the corresponding plane region contours. The size of the normal line plots is proportional to the numbers of points belonging to each plane. The statistical data of the process are shown in table I.

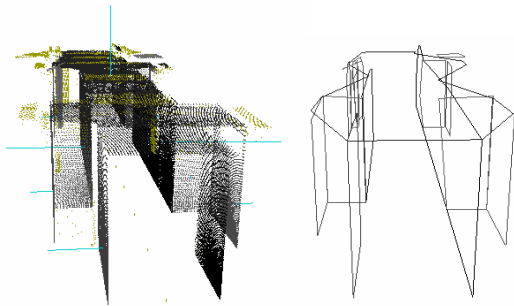


Fig. 14 Surfaces and Contours of the scene.

TABLE I
STATISTICAL DATA

Stage	Parameter		
Initial	Scene points	32400	
	Initial planes	56	
	Descarted points	6510 (40%)	
Simplification	Vicinity	N° planes: 18	
	Occupation	N° planes: 15	
	Final result	Located planes	14
		Descarted points	4120 (25,4%)

The real environment is shown in figure 15. 18 planes have been found, including the corridor floor and roof. The algorithm has located about 80% of the planes. The dimension error of the found planes is less than 4 cm (in comparison with the real environment).



Fig. 15. Real environment

Figure 16 shows the final, textured model. The texture image map corresponding to each plane has been found upon several image stitching.



Fig16. 3D Model with texture.

Suitable results are obtained for plane surfaces, when navigating along the obtained model. However, large perspective distortion may happen at free-form objects. In fact, free-form objects appear embedded in walls, and large distortion happens when the selected view point differs from that of the texture acquisition. For example, the fire extinguisher exhibits low perspective distortion from view

point selected in figure 17a (close to that of the texture image acquisition). However, large perspective distortion occurs when watching the same object from other viewpoints, as is the case in figures 17b, 17c.



(a) Frontal view point. (b) Left Lateral (c) Right Lateral

Fig.17. View points of the extinguisher

We have tried the proposed method on different images, in order to study the distortion effect. We have first mapped a regular (θ, ϕ) discrete grid, on a regular image (through the inverse transformation). RGB values at the resulting image have been calculates through bilinear interpolation. Results for different radius-to-image size, i.e. R-to-A, ratios are shown in figure 18. Better results are obtained for large R/A values.

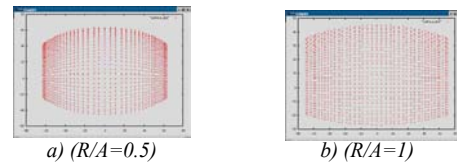


Fig 18. Plane $\theta-\phi$

Results obtained by applying the direct transformations to a group of images in a scene, taken from a given optical center, are shown in figure 19.

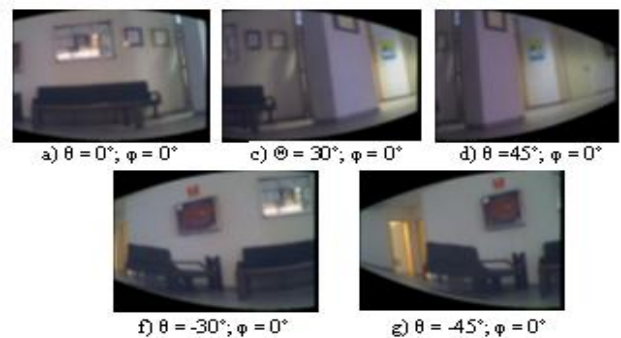


Fig. 19 Image Planes projected on the sphere.

Figure 20 shows the result obtained after the corresponding stitching.



Fig. 20 Stitching of the image planes on the sphere.

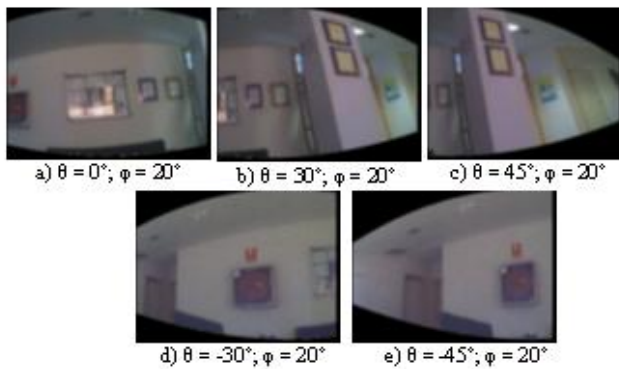


Fig. 21 More Image Planes projected on the sphere.

Results obtained on a further image set are shown in figures 21 and 22.



Fig. 22 Stitching of the image planes on the sphere

Next step consists on projecting the measured 3D points on the sphere, in order to establish the texture to 3D correspondence, which is currently under development.

V. CONCLUSIONS

A method for reconstructing 3D models of environment, using the 3D data measured by a laser scanner and the images acquired by a camera, onboard a mobile robot, has been described. Results have been found suitable for plane surfaces. However, large distortion may occur at free-form objects, since their points are projected on planes. A method for reducing the perspective distortion at free-form objects has then been proposed. This method consists on projecting the 3D points within the model, and the 2D points within the texture images, onto a spherical surface. The obtained results have allowed the distortion in the said projection to be analysed. It has been found that some parameters largely affect results, such as the sphere radius, and the image discretization.

An immediate future research line will focus towards projecting the triangles of the 3D model, onto the $\theta-\phi$ plane, in order the texture to be mapped onto the model.

REFERENCES

[1] Andreozzi, L; Barnobi, L; Giuffrida, A; Santagati, C. (2002). "New Frontiers to Architectural Survey: Laser Scanner 3D". University of Catania.

[2] Axel, W; Amir, M. (2000). Monloulouk, "Multisensoric active spatial exploration and modeling" in *Dynamische Perzeption: Workshop der GI-Fachgruppe 1.0.4 Bildverstehen*, Ulm. Berlin.

[3] Bajcsy, R; Kamberova, G; Nocera, L. (2000) "3D Reconstruction of Environments for Virtual Reconstruction". 4th IEEE Workshop on Applications of Computer Vision.

[4] Debevec, P; Taylor, C. (1996) "Modeling and Rendering Architecture from Photographs: A Hybrid Geometry-and Image-based approach". *Computer Graphics*, vol 30. pp: 11-20.

[5] Diez, J.; Zalama, E; Gómez, J. "Adding texture to 3D models using a mobile robot". Technical Report. ETSI Ingenieros Industriales. University of Valladolid. 2004.

[6] Francés, A; Lambán, L; Rubio, A. (1998). "Diagramas de Voronoy y Topología Digital". Universidad de Zaragoza. España.

[7] Frintop, S; Rome, E.; Nuchter, A.; Surmann, H. (2003) "An Attentive Multi-modal Laser Eye". In: *Proc. of 3rd International Conference on Computer Vision Systems (ICVS 2003)*, J. Crowley, J.H. Piater, M. Vincze, and L. Paletta (eds), pp. 202-211, Springer, Berlin. Conference: Graz, Austria, April 1-2, 2003

[8] GFD Dennou Library. "Map Projections".

[9] González, J. (1999). "Visión por Computador". Editorial Paraninfo. España.

[10] Howells, L. (2002). "The Design of an Automated Data Analysis System for Robotic Telescopes and its Application to be Stars".

[11] "Introduction to Computer Vision". CS/ECE 181B. 2003.

[12] Itkowitz, B. (1998). "Image Mosaics". CAS CS585.

[13] Lensch, He, Heidrich, Wo. (2001). "A Silhouette-Based Algorithm for Texture Registration and Stitching". *Graphics Models*. Vol 63, issue 4. pp. 245-262.

[14] Martín, M; Gómez, J; Zalama, E. "Obtaining 3D Models of Indoor Environments with a Mobile Robot by Estimating Local Surface Directions". *Robotics and Autonomous Systems*, Elsevier, vol 48/2-3 pp. 131-143. 2004.

[15] Nuchter, A (2003). "Planning Robot Motion for 3D Digitalization of Indoor Environments". Alemania.

[16] Nuchter, A; Surmann, H. (2003). "Kurt3D - An Autonomous Mobile Robot for Modelling the World in 3D". *Ercim News* No. 55. http://www.ercim.org/publication/Ercim_News/enw55/nuechter.html

[17] Pardiñas, A. (2002) "Operaciones Básicas en el Dominio Espacial". 2002.

[18] "Project One: Spherical Virtual Image Space" www.missouri.edu/w117a/Course/vision.

[19] Sheffer, A; Gotsman C.; Dyn, N. (2003). "Robust Spherical Parametrization of Triangular Meshes". *Computing*, 72(1-2):185-193, 2004.

[20] Shum, H; Han, M. (1998). "Interactive Construction of 3D Models from Panoramic Mosaics". *International Conference on Computer Vision and Pattern Recognition (CVPR)*.

[21] Shum, H; Szalinski, R. (1998) "Creating Full View Panoramic Image Mosaics and Environment Maps". *International Conference on Computer Vision and Pattern Recognition (CVPR)*.

[22] Surmann, H.; Lingemann, Kai; Nuchter, Andreas; Hertzberg, Joachim. (2001) "Fast acquiring and analysis of three dimensional laser range data". 6th International workshop Vision, Modeling and Visualization. Stuttgart, Germany.

[23] Surmann, H. (2001). "A 3D laser finger for autonomous mobile robots" in *Proceedings of the 32nd ISR (International Symposium on Robotics)*, Alemania,

[24] Surmann, H; Lingemann, K; Nuchter, A. (2001). "A 3D Laser Range Finger for Autonomous Mobile Robots". *Proceedings of the 32nd ISR (International Symposium on Robotics)*.

[25] Thrun, Se; Burgard, W (2000). "A Real Time Algorithm for Mobile Robot Mapping with Applications to Multi-Robot and 3D Mapping". *IEEE International Conference on Robotics and Automation*. San Francisco.

[26] Tumblin, J. (2002). "New Direction: Applying Projections". CS 395/495-26 Spring.