Repositório ISCTE-IUL

Semi-Automatic 3D Reconstruction of Urban Areas, by using

Epipolar Geometry and Template Matching

José Miguel Salles Dias[1,2]

midias@microsoft.com

João Correia[2]

João.Correia@iscte.pt

Rafael Bastos[2]

Rafael.Bastos@iscte.pt

Rodrigo Vicente[2]

Rodrigo.Vicente@iscte.pt

MLDC, Microsoft Language Development Center Portugal, Av. Prof. Doutor Aníbal Ca-
vaco Silva, Edifício Qualidade C1, C2, Tagus Park, 2744-010 Porto Salvo, Portu-
gal, www.microsoft.com/portugal/mldc

ADETTI/ISCTE, Associação para o Desenvolvimento das Telecomunicações e Técnicas
de Informática, Edifício ISCTE, 1600-082 Lisboa, Portugal, www.adetti.pt

Abstract

In this work we describe a novel technique for semi-automatic 3D reconstruction of urban areas, from airborne stereo-pair images whose output is VRML or DXF. The main challenge is to compute the relevant information - building's height and volume, roof's description and texture – algorithmically, since it is very time consuming and thus expensive to produce it manually for large urban areas. The algorithm requires some initial calibration input and is able to compute the above mentioned building characteristics from the stereo pair and the availability of the 2D CAD and the Digital Elevation Model of the same area, with no knowledge of the camera pose or its intrinsic parameters. To achieve this, we have used epipolar geometry, homography computation, automatic feature extraction and we have solved the feature correspondence problem in the stereo pair, by using template matching.

Keywords

Urban Spaces, 3D Reconstruction, VRML, Digital Terrain Model, Homography, Cany Edge Detection, Feature Matching, Template Matching, Airborne Images, Epipolar Geometry, Digital Photogrammetry

1. INTRODUCTION

Digital Surface Models (DSM), Digital Terrain Models (DTM) and Orthophotomaps, (Kasser, 2002) of a given area are currently available on a commercial basis and are regarded as important tools used in Urban Planning tasks. More re-

Fig. 1. One of the aerial stereo images pair (Moledo region in the West of Portugal).

cently, this type of geo-referenced information, linked with Geographical Information System – GIS mappings, have been made available on a global scale (Google Hearth, 2005). Often these models are obtained from airborne stereo pair photos (Fig. 1), using known landmarks specially set in the terrain, manually identifying such landmarks as features in the obtained stereo pair and, applying *epipolar* Geometry (Trucco, 1998) to derive the DSM, DTM and orthophotomap. This procedure is also valid for Urban Areas, although the aim is to obtain only the DTM and orthophotomap. Volumetric models of the built space are not widely obtained by this procedure. In Urban Areas, the available format is generally 2D, as depicted in Fig. 1 (described by means of proprietary formats such as DWG or DXF from AutoCAD). 3D reconstruction of the exterior volumes of the built space is also possible, but only for limited areas, using manual, image-based (Faugeras, 1994) or laser-based techniques (Nakagawa, 2002). With this work, we are aiming to obtain, by automatic procedures, coupled with some interactive procedures, the volumetric description of large Urban Areas. Our aim is to make that type of information easier to produce, at low cost, and capable of being altered or added without having all the work of collecting the inputs again. Our algorithmic technique receives the following inputs related to the same geographical area:

- airborne stereoscopic photos (Fig. 1);

- a DXF file, representing the 2D CAD entities representing contours of buildings or rocks (closed polylines), rivers (open polylines), trees (points), etc. (Fig. 2);

- a 3D Digital Elevation Model (3D points in an ASCII file), representing the terrain.

Fig. 2. 2D Geometrical Model of an Urban Area (Moledo region in the West of Portugal).

The output of our engine (Fig. 3) is the volumetric representation of the Urban Space, described in VRML (Virtual Relational Model Language) (VRML) or DXF, representing the urban space with sufficient precision and realism (Fig. 4). This VRML file can afterwards be published over the internet and accessed with browsers using a VRML plug-in.

The paper is organized as it follows: in section 2, we provide a background overview on the issue of 3D Reconstruction. In section 3, we present a system overview. Section 4 covers the system development and details the algorithmic techniques of each system module, towards semi-automatic 3D reconstruction of urban areas. In section 5, test results and discussion are presented and finally, in section 6, conclusions and future directions of research are given.
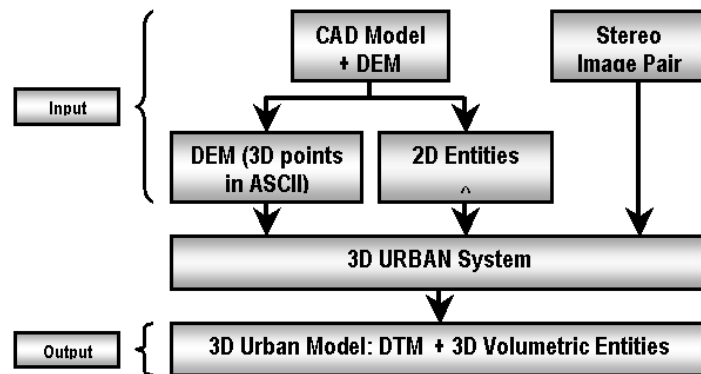

Fig. 3. 3D Urban System datapath.

2. RELATED WORK

International initiatives, such as the Source for Environmental Representation and Interchange (SEDRIS), are providing today key enabling technologies to represent and exchange without loss, environmental geo-referenced data (terrain, ocean, air and space). This initiative, which has ongoing efforts of standardization through the International Organization for Standardization (ISO), has generated the technological framework, were our work can be integrated in the near future. In the recent years interesting achievements have been made also regarding the 3D reconstruction of real urban spaces (Bernardes 1996). Although most of the works made in this area are being supported by *epipolar* geometry, they implement different combination of techniques in order to improve the precision and performance of the reconstruction system.

In his Phd thesis, Dias (Dias, 2003), has presented a system capable of generating complete, high-resolution three-dimensional models of real word scenes combining inputs from passive intensity images and active range sensors. Therefore, Dias work uses passive techniques to compensate the limitations of active techniques in specific steps of the reconstruction and vice-versa.

In (Pollefeys, 2001), Pollefeys explains *"how a 3D surface model can be obtained from a sequence images",* without real-time requirements and without prior knowledge of the camera motion and camera settings. The system presented by Pollefeys adopts the *epipolar* geometry, uses full perspective cameras and does not require prior models. According to Pollefeys (2001), in the last few years, researchers working in three-dimensional reconstruction area *"tried to reduce the requirements for calibration and augment the automation of the acquisition"*. This matches the aims of our work, where we are focused in the automation of the reconstruction process with the minimum number of inputs. The main difference between Pollefeys work and ours, resides in the fact that the former applies his 3D reconstruction techniques to any sequence of images, whereas ours is specifically oriented to stereo pairs.

3. SYSTEM OVERVIEW

As mentioned, the aim of our system is to derive a complete virtual 3D urban model, from the availability of the following input data of the same geographical area:

• Stereo pairs with sufficient superposition of the area in question, such that, all relevant entities (such as buildings), are at least partially visible in both images;

• 3D DEM - Digital Elevation Model;

• 2D CAD model representing the contours of the urban model.

The system is able to compute the 3D urban model associated to the area of superposition of the stereo pairs. A full 3D urban area model of a large area enables the user to perform virtual navigation and can also be used as an interaction metaphor for a 3D Geographical Information System (GIS), much like in the GoogleEarth system (GoogleEarth 2005).

Fig. 4 depicts the reconstructed 3D VRML model obtained by our system, of the area shown in Fig. 1. In light blue we see zones of no superposition of the stereo pair, where the algorithm is unable to compute the heights of CAD entities. By using a Boundary Representation (BRep) model, the VRML or DXF output represents the DTM plus all the entities of the buildings layer (obtained from the input CAD model), but with their heights and volumetric description calculated by our 3D reconstruction technique.
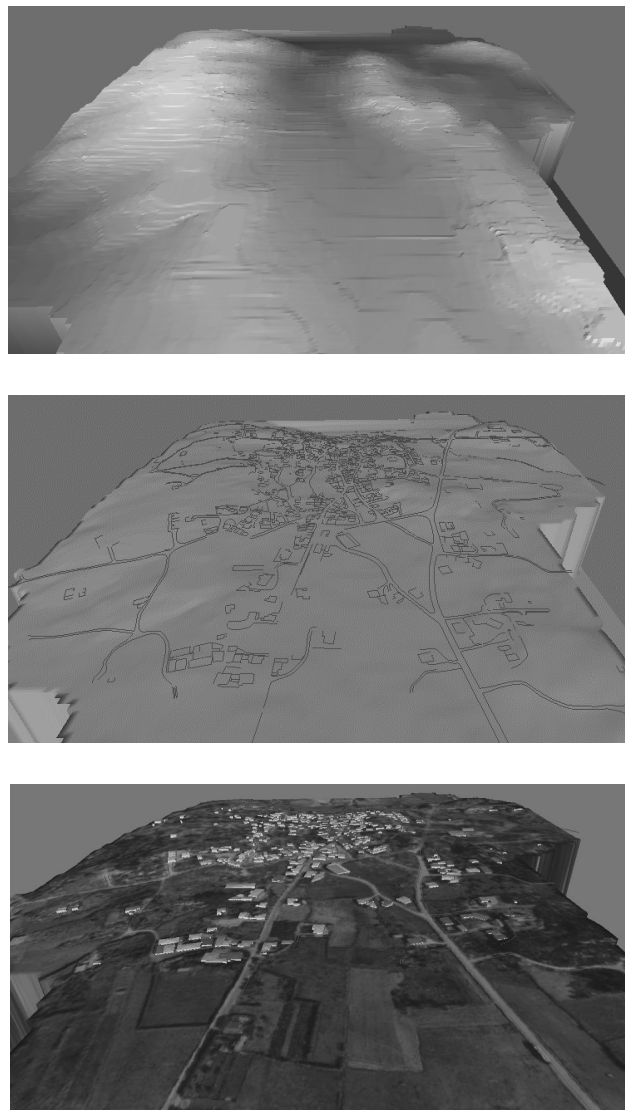

Fig. 4. Top: Generated DTM; Middle: DTM with projected CAD entities; Bottom: General view of the 3D urban model with texture overlay, computed by our technique (Moledo region in the West of Portugal).

4. 3D-URBAN SYSTEM ARCHITECTURE

Fig. 5 shows the modular architecture of our system, referred to as 3D Urban (3D-URBAN). The Input Module reads the geometrical and topological information of the entities of the 2D CAD model, the airborne stereo pair and the 3D DEM. Another group of modules (DTM Generator, 3D Reconstruction and Roof Colour Extraction) is responsible for the core system processing. The Output Module is responsible for the creation of the VRML or DXF files with the 3D description of the urban area.
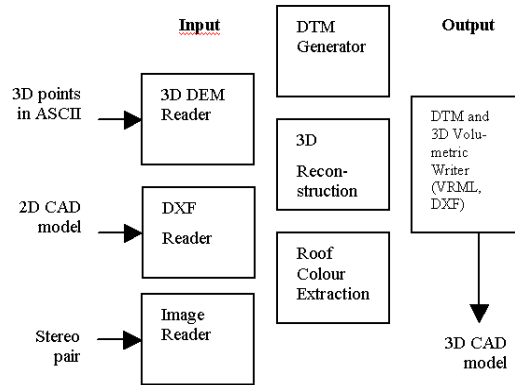
Fig. 5. 3D Urban System Architecture.

4.1 Input Module

The Input Module encapsulates three readers:

1. The 3D DEM Points reader, which reads an ASCII file with 3D point information, defining the DEM and loads it into a vector of coordinates;

2. The DXF CAD 2D layer reader, which is responsible for reading all the layers of the CAD model (see Fig. 2);

3. The stereo image pair reader (see Fig. 1).

The DXF reader parses a series of entities for each CAD layer, with each entity being represented by a closed 2D polyline, with an arbitrary number of points (Fig. 2). The CAD layers have some semantic meaning and describe, for example, the contours of the buildings, the roads, the telecom systems, the walkways, etc. Our technique only applies to entities described by closed polylines.

The input images are two aerial stereo photographs, obtained by the same camera (or by distinct cameras with equal intrinsic parameters[1]) at a given flight altitude.

4.2 DTM Generator

The development of this module, which generates the DTM - Digital Terrain Model, from the DEM - Digital Elevation Model, is justified by the fact that the points of the DEM have neither a specific order, nor a topology. We may consider them as a cloud of points. For our 3D processing and visualization requirements, we need to generate a mesh of triangles. Therefore, this module is responsible for ordering the points in groups of three, formed by their proximity, thus generating Triangles. We have initially adopted a Delaunay Triangulation technique, based on the Voronoy Diagram (Baker 1989), whose result can be seen in Fig. 6 - top. However, we have realised that for our 3D reconstruction needs, of generating elevations along the CAD entities polylines with a step in the order of 1 meter, a regular mesh of triangles was better suited, as depicted in Fig. 6 - bottom. This approach generates more triangles, but it brings, as advantage, better control of the DTM triangle grid resolution.
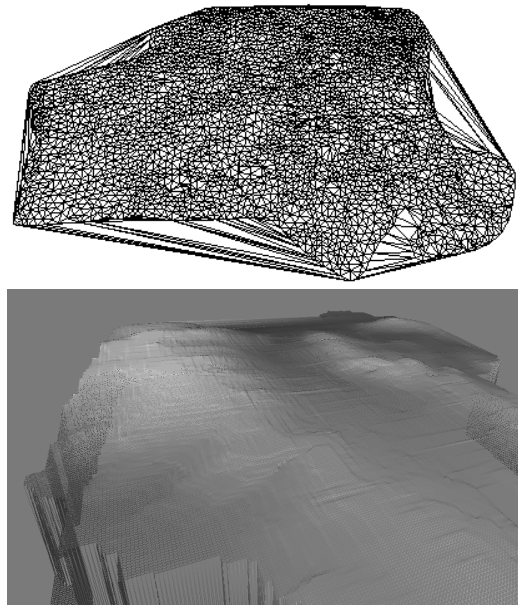
Fig. 6. A BRep model of the Digital Terrain Model: Top –
Delaunay triangulation; Bottom: regular triangulation.

4.3 3D Reconstruction

The proposed 3D reconstruction technique complements the input 2D CAD model, by adding unavailable 3D volumetric information to the model. Additionally, the technique requires a stereo pair of airborne images and the DEM of the geographical zone. Our technique is mathematically based in concepts used in Computer Vision and 3D Computer Graphics, such as:

---

[1] The intrinsic parameters of the camera are: the focal length, the pixel coordinates of the centre of the image, the pixel scale factors in

- *epipolar* geometry, which deals with stereoscopy and projective geometry:

- *homography*, a planar geometrical transformation here applied in the computation for the stereo pair camera pose;

- *feature extraction*, in one image of the stereo pair;

- *feature correspondence* discovery (or *feature matching*) in the other image of the pair, using *template matching*, for precise disparity and depth computation of the interior of CAD entities.

Our novel reconstruction algorithm computes the CAD entities height using a semi-automatic technique. The volume of each entity is found by applying an extrusion, with the found height as a parameter, to the entity contour, available from the input module.

4.4 Output Module

The Output Module handles the creation of output files (in DXF and VRML formats) which integrate the DTM and the 3D urban volumetric models. There are two writers:

- The **DTM Writer**, which generates the DTM surface (Fig. 1, top). In the case of the VRML format, the DTM is represented as a mesh of triangles using the IndexedFaceSet node;

- The **3D Volumetric Writer**, responsible for the 3D volumetric representation of the entities.

In the specific case of buildings, we are not yet addressing the problem of reconstructing their roofs topology with a sufficient level of detail. In fact, in this paper, we evaluate the heights of each CAD entity, with the simplified assumption that the roofs have all the same topology: a simple pyramid, with its apex in the geometrical centre of the roof contour (Fig. 4).

4.5 *Epipolar* Geometry

With *epipolar* geometry (Trucco 1998), we are able to develop a two-camera stereo model that can maintain a relation between world points and their projections on the images planes of a stereo pair. In Fig. 7, e_l and e_r, the left and right *epipoles*, are given by the intersection of the line segment that joins O_l and O_r, respectively, the left and right virtual camera positions (that match the real camera positions), known as the baseline, with the left and right images planes. The *epipoles* represent in fact, the projection of the camera position in the other image. When the images are parallel the *epipoles* are at the infinite.

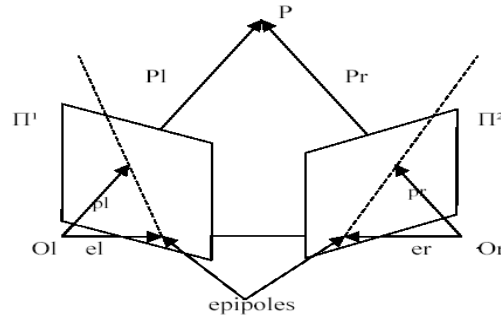the two coordinate axis of the image and the lens distortion parameters.

Fig. 7. The two cameras model of a stereo pair.

The practical importance of *epipolar* geometry consists in the fact that the *epipolar* plane defined, in Fig. 7, by the three line segments $[O_l \, O_r]$, $[O_l \, P]$ and $[O_r \, P]$ intersects each image plane Π^1 and Π^2 (of the pair) in two lines, respectively, $[p_l \, e_l]$ and $[p_r \, e_r]$, called *epipolar* lines (the line segment $[O_l \, O_r]$ is also known as the *baseline*). Therefore, given a projection p_l of a world point P on an image (for example, on the left image), it is known that p_r, the projection of the same point in the other image (in this case, in the right image) lays on the *epipolar* line (Fig. 7). At the same time, we know that the world point P can be found in the line which passes through p_l and O_l. If we know O_l and O_r, and we are able to find the projections p_l and p_r in each of the image pair through Computer Vision techniques, we can derive the world point P, by intersecting (under a certain distance tolerance) the lines that pass, respectively, by O_l and p_l and, by O_r and p_r. This is known as the triangulation technique in 3D reconstruction. As mentioned, this technique requires the evaluation of O_l and O_r. If we know O_l in the world reference frame, O_r can be found by applying a translation and a rotation to the former. The parameters that define these two transformations matrixes are known as the extrinsic parameters of the stereo camera system.

4.6 The 3D Reconstruction Algorithm

Our 3D reconstruction algorithm is divided into six main steps:

1. Computation of the Fundamental Matrix, F, a basic concept of *epipolar* geometry in stereoscopy, which is required for the stereo rectification process (Dennis 1996);

2. Rectification (warping) of the stereo images pair, to aid the point correspondence problem (Arbouche, 1999);

3. Computation of the 3D heights (in world coordinates) along the polylines of the 2D entities, belonging to the input CAD model, to aid the computation of the 3D volume of such entities;

4. Computation of the absolute cameras pose of the rectified stereo pair as well as the camera intrinsic parameters, assuming a "real" distorted model for the camera lens, to establish a relation between the (CAD) world reference frame and each rectified image pair reference frame.

5. Depth Computation of the highest point inside each CAD entity and determination of its 3D height and volume, assuming simple pyramidal roofs, using *projective* geometry, *epipolar* geometry of the rectifies stereo system, *feature extraction* and *template matching* along the *epipolar* lines.

6. Generation of the 3D DTM with the 3D volumetric entities (buildings) for visualisation and 3D interaction purposes.

These algorithmic steps are described in more detail, one by one, in the following sections.

4.6.1 Computation of the Fundamental Matrix, F

The Fundamental Matrix establishes a relation between two images of a stereo pair through *epipolar* geometry. This relation is demonstrated by the following equation:

$$u_r = F \, p_l \quad (1)$$

In (1) F is the 3 x 3 Fundamental Matrix (with nine unknowns), u_r is an *epipolar* line in the right image and p_l is the projection in the left image of the world point P. This equation establishes a relation between points and the corresponding projective *epipolar* lines. This relation is useful, for example, in our feature matching problem and, for the rectification of the image pair, as will be seen in the next section. For each corresponding pair of projection points p_r and p_l in pixel coordinates, the Fundamental matrix F, satisfies the equation:

$$p_r{}^T F \, p_l = 0 \quad (2)$$

To obtain one unique result of *F*, it is necessary to substitute eight corresponding points p_r and p_l in the previous equation and solve a linear system of eight equations. These points must be as far from each other as possible. To compute the Fundamental Matrix, using this technique, we have used the OpenCV - Open Source Computer Vision Library, namely the function *cvFindFundamentalMatrix* (OpenCV). This function calculates the Fundamental Matrix from eight pairs of correspondent points in both images, according to the RANSAC (*Random Sample and Consensus*) 8-point algorithm (Fischler 1981). The advantage of the RANSAC over other 8-point algorithms (like LMedS – *Least-Median-Squares* (Longuet-Higgins 1981), is the way it determinates the best *F* matrix. Although both techniques eliminate outliers due to bad point localization and the presence of noise in the image segmentation, in the RANSAC's case, the chosen *F* maximizes de numbers of inliers. Once the outliers are eliminated, *F* is recalculated to get a better approach. Thus, the more pair of points we provide to the algorithm, the higher is the precision of the computation.
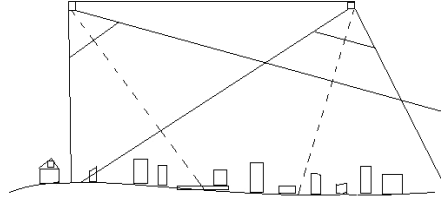
Fig. 8. Non-warped stereo model.

4.6.2 Stereo Images Rectification (Warping)

A pair of stereo images is rectified (or warped) when their *epipolar* lines are horizontal and parallel to the *baseline* (see Fig. 7).
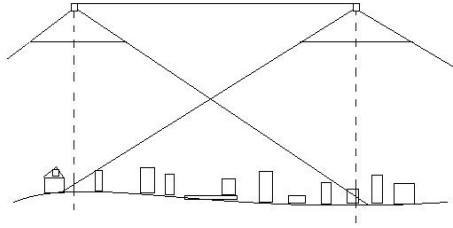

Fig. 9. Warped stereo model.

One goal of warping is to be able to make the simplification of the geometry of the stereo pair. In Fig. 8 and 9, we illustrate, respectively, the general model of a stereo pair and the rectified (warped) model. Fig. 11 depicts the resulting rectification of the stereo images used in our case study. In the rectified stereo pair, the image planes $\prod^1$ and $\prod^2$ are parallel to each other, which forces the correspondent *epipolar* lines to lie on the same scan line of both images.

This is useful for the feature matching problem, where, given a feature (projected) point p_{1l} on say, the left image, we need to find the corresponding (matched) feature point, p_{1r}, on the right image. From the *epipolar* geometry (see section 4.5), we know that this matched point p_{1r}, must lie in an *epipolar* line that satisfies equation (1). But since the stereo pair is rectified, this *epipolar* line lies on the same scan line of p_{1l}. We conclude that the feature matching search problem reduces from 2D to 1D. Additionally, in the wrapped stereo pair, the projection lines from both camera positions are perpendicular to the *baseline*, which is very useful for the entities depth computation, since it reduces the triangulation problem (see section 4.5) from 3D to 2D.
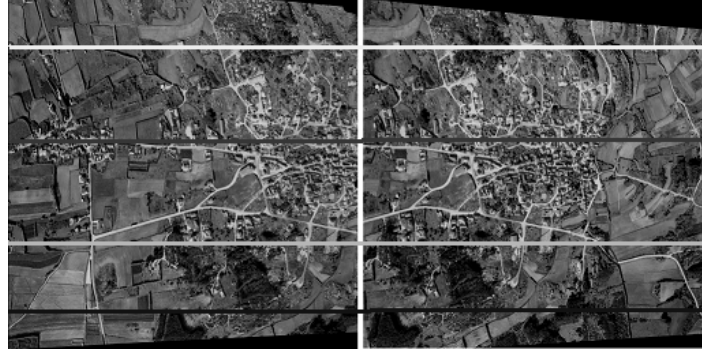
Fig. 10. The rectified stereo pair showing tree *epipolar* lines which correspond to three common scan lines of the images. Note that, although rectified, the images are not yet corrected for the radial and tangential lens distortion**.**

To warp a pair of stereo images, we have used the *epipolar* geometry principles and the OpenCV warping algorithm (*cvWarpImage*). We have extracted all the corresponding *epipolar* lines from the two images, which are then transformed into collinear horizontal lines (see result in Fig. 10).

4.6.3 Computation of the 3D heights of the entities edges

As mentioned, the input CAD model, includes various layers, each one aggregating a number of entities described by 2D polylines. These polylines define the contour of buildings, roads and other elements of an urban plan. In our algorithm, we need first to compute the 3D height of the polylines vertexes and then to estimate the 3D height along the polyline edges. The computation of the 3D height of the polylines vertexes is straightforward, since the generated DTM is in the same reference frame as the input 2D CAD model (Fig. 2). The 3D height of each polyline vertex is easily done by finding the nearest DTM triangle and, inside it, the nearest triangle vertex and by setting the height of the polyline to the height of the found vertex. This gives us a maximum error in the horizontal plane of 1 meter (the planar resolution of the triangle mesh). To estimate the height along polylines edges, we perform linear interpolation of height along the edges with a step of 1 meter (Fig. 4 – Middle).

4.6.4 Computation of the absolute cameras pose of the stereo pair

At this stage, we have the following knowledge of the stereo system:

1. We know the relation between points projected in the image planes of the stereo pair and, specifically, we know that, given one projected point in one of the images of the pair, the corresponding projected point lies in a known *epipolar* line in the other image of the pair, which is in fact a common scan line of both images (since they are rectified);

2. We know the DTM of the area in and the projected 3D entities polylines in that DTM, in world coordinates;

Now we need to obtain the absolute cameras pose of the stereo pair to establish a relation between the world reference frame and each rectified image pair reference frame.
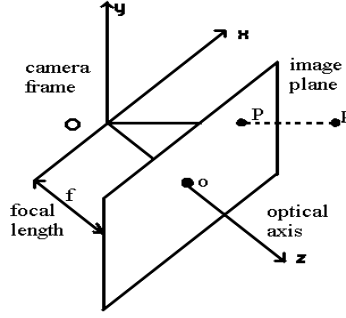

Fig. 11. The Pinhole camera model.

To understand this problem, we need to introduce the classical pinhole camera model and principles of *projective* geometry (Trucco 1998), which gives us the relation between a world point P, and its projection p in the image plane.

In this model, the virtual camera 3D reference frame (Fig. 11) is translated and oriented, in relation to a given 3D world reference frame. The so-called camera *extrinsic* parameters, are structured in a rotation matrix R (with a total of nine independent parameters) and a translation vector T (totalling tree independent parameters), which transform one reference frame into the other. The concatenation of these matrixes T and R, generates a new resultant matrix, which transforms object points P, defined in the world reference frame, into object points P_{cam}, in the camera reference frame, with the following form:

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}, T = \begin{bmatrix} t_{11} \\ t_{21} \\ t_{31} \end{bmatrix} \qquad (3)$$

P and P_{cam} are related through:

$$P_{cam} = [R\ T]\,P \qquad (4)$$

3D object points P_{cam}, in the camera reference frame are then transformed, into (ideal) 2D projected points p in the image plane, in pixel coordinates, by using *projective* geometry. The parameters that define such transformation are called the *intrinsic* parameters of the camera model and are structured in the *intrinsic* matrix A, which relates 3D object points P_{cam} in the camera reference frame with 2D projected points p in the image, in pixel coordinates (Zhang 1999):

$$A = \begin{bmatrix} f/S_x & 0 & c_x \\ 0 & f/S_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \qquad (5)$$

These parameters are:

- The focal length f (the distance between the camera lens and the view plane) in millimetres;

- The image centre location (c_x, c_y) in pixel coordinates;

- The effective pixel size (S_x, S_y) in millimetres;

- The lens distortion, which can be expressed by using four coefficients: two radial distortion coefficients k_1, k_2, and two tangential ones t_1, t_2.

The 2D projected points, p, in the image plane and the 3D object points, P_{cam}, in the camera reference frame (Fig. 11), are related through:

$$p = A P_{cam} \qquad (6)$$

By replacing (6) in (4), we can find a relation between any image point p (in pixel) and a world point P of the CAD model, of the following form:

$$p = A [R\,T] P \qquad (7)$$

If *intrinsic* matrix A, and *extrinsic* matrixes R and T are unknown, we may find these analytically, given at least 8 (p, P) points correspondences (Trucco 1998).
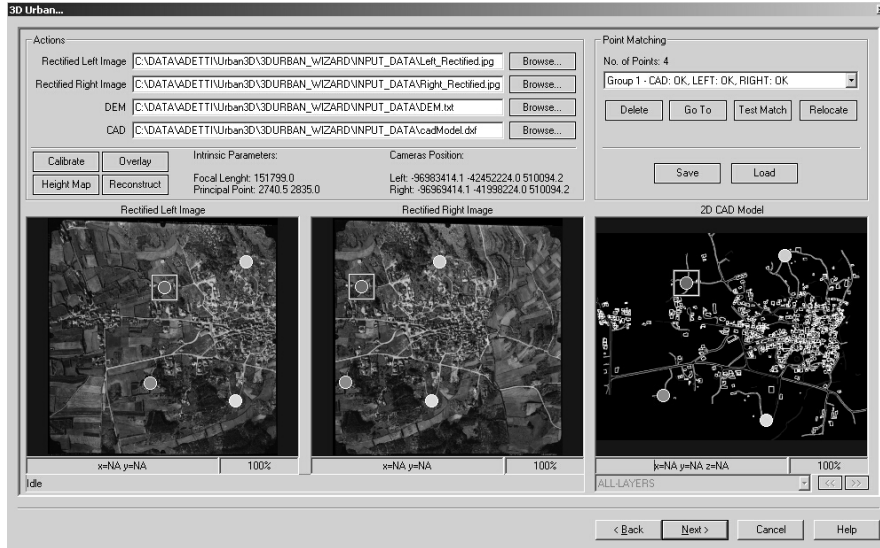
Fig. 12. Picking corresponding points for the homography, respectively, for the left image, right image and the CAD model.
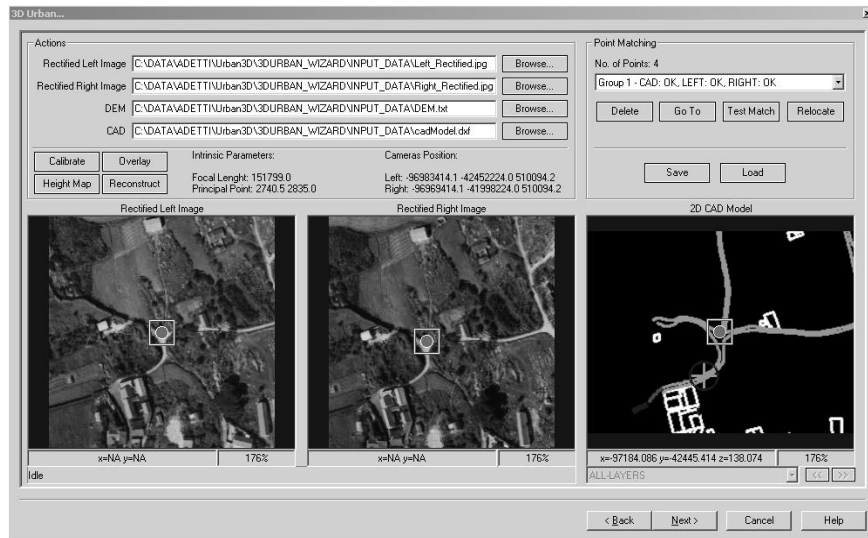
Fig. 13. Picking corresponding points for the *homography*: - zoom in**.**

However, our problem is a sub-set of the more general problem stated before. In our case, we need to find the transformation between the world plane reference frame of the CAD model (if we define if just as planar problem of 2D CAD geometry) and the left and right camera view planes. Mathematically this problem is called a *homography*, a geometric transformation between the reference frames associated to two arbitrary planes. This reduces our problem to a 2D perspective transformation:

Given a set of 2D point correspondences (p_i, p_i'), where p_i belongs to one of the planes and p_i', belongs to the other, we need to estimate the projective transformation *homography* matrix *H*, of the form:

$$p_i = \begin{bmatrix} x_i \\ y_i \\ w_i \end{bmatrix} \qquad p_i' = \begin{bmatrix} x_i' \\ y_i' \\ w_i' \end{bmatrix} \qquad (8)$$

$$p_i' = H p_i \qquad (9)$$

$$\begin{bmatrix} x_i' \\ y_i' \\ w_i' \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ w_i \end{bmatrix} \qquad (10)$$

By knowing at least 4 point correspondences, the Direct Linear Transformation - DLT algorithm can be applied (Abdel-AzizKarara 1971), using Singular Value Decomposition (SVD) (Golub 1993), to find the unknown matrix H. In our case, we need to evaluate two *homographies*: one between the 2D CAD model plane and the rectified left image and, the other, between the 2D CAD model plane and the rectified right image. In Fig. 12, we depict an interactive session of picking the corresponding points for the *homographies*, respectively, for the left image, right image and the CAD model. While finding points correspondences in the stereo pairs, the system is able to snap automatically to the point where a better match is found, by using the *template matching* technique (see section 4.6.6 - Extraction of Feature Points from the Entities Area on the Image). In Fig. 13, we depict the *zoom in* function, which aids the interactive picking of point correspondences.

After the computation of matrix H, the camera *extrinsic* parameters (matrixes T and R), can be trivially extracted from this *homography* (Abdel-AzizKarara 1971). We also minimize the re-projection error via gradient descend (Brakke 2004), iterating to a maximum of 20 times using a refining threshold value of 1e-10. However, there is a restriction on the use of the DLT algorithm: three of the four points used cannot be collinear or else it will result in undetermined H matrix. This algorithm requires the previous knowledge of the mentioned camera *intrinsic* parameters. These can be obtained by using the same 4 point correspondences and applying the technique by Z. Zhang described in "Flexible camera calibration by viewing a plane from unknown orientations" (Zhang 1999), which, in our case, correspond to two camera poses. In Fig. 14 we project the 2D CAD model in both image planes of the stereo pair, using the found camera poses. We can notice that the overlay result is correct and precise (see Fig. 17 for a zoom in view of one of the projections.
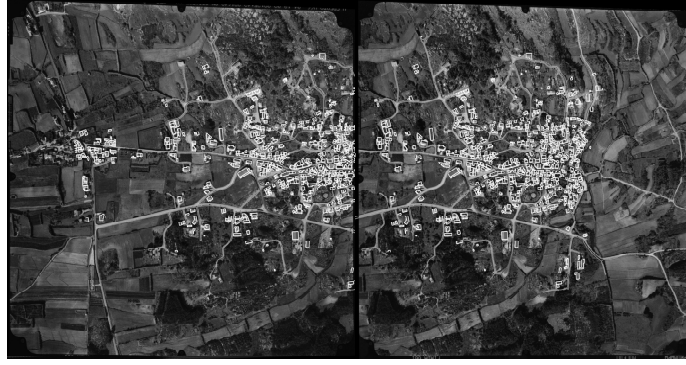
Fig. 14. Re-projection of the CAD model in the Left and Rights images, using the evaluated camera poses of the stereo pair**.**

4.6.5 Depth computation of each feature point and determination of the 3D height and volume of each CAD entity

At this stage of our technique, we have the added to our knowledge corpus, the relation between arbitrary points in the 2D CAD model reference frame and points in image coordinates of the stereo pair, since we know precisely both cameras *extrinsic* and *intrinsic* parameters.

Now is time to search inside both images areas of the projected 2D CAD entities, for feature points and for precise point correspondences, in order to derive point disparities and proportional point depths.

But first, we need now to go back again to the *epipolar* geometry of the rectified stereo pair (see Fig. 16), so that we are able to derive mathematically the heights and corresponding 3D volume of each CAD entity.
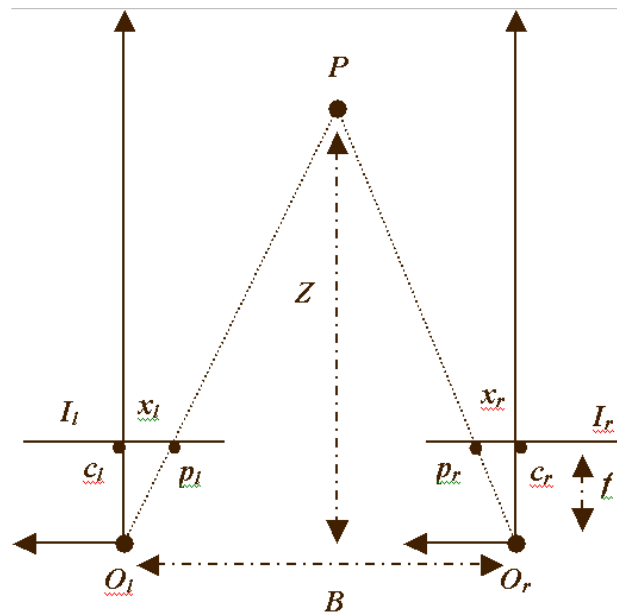

Fig. 15. Analysis of the rectified stereo pair geometric problem.

Given a stereo pair, a point P in world coordinates and its projections p_l and p_r, respectively, in the left and right images, the problem statement can de defined as follows (Trucco 1998): "Recover the depth Z of P from its projections p_l and p_r". In Fig. 16, we have:

- x_l and x_r are the coordinates, in pixel, of p_l and p_r, the positions of corresponding features in the stereo pair in relation to the images centres, c_l and c_r;

- f is the focal length in *mm* (computed by the technique described in 4.6.4);

- B is the baseline, in *mm*, of the rectified stereo system: the distance between the two camera positions (our problem is know in the literature as the "high baseline problem" (Trucco 1998), thus quite sensitive to errors in the evaluation of B);

- Z is the depth of the point P, in *mm*, that is, the distance in the vertical axis from the (aligned) cameras centre altitude to the feature point altitude;

- $d = S_x (x_r - x_l)$ is the stereo disparity of point P in *mm*;

- S_x, is the effective pixel size in *mm*, in the width direction (computed by the technique described in 4.6.4);;

From the similarity of triangles in Fig. 15 ($O_l P\ O_r$) and ($p_l P\ p_r$), we have:

$$\frac{B + S_x\left(x_l - x_r\right)}{Z - f} = \frac{B}{Z} \qquad (11)$$

Equation (11) can be simplified to:

$$Z = f\frac{B}{d} \qquad (12)$$

Fig. 16 illustrates the depth computation. We see that the depth Z is related with the altitude of the airplane, H (622 meters, in our case) and, with the altitude of point P of the CAD entity, h, by:

$$Z = H - h \qquad (13)$$

In equations (11) and (12), we see that the depth Z of a point in the entity roof is very sensitive to B (baseline), found by *homography*, and d (disparity), found by point correspondences and *template matching*. From equation (13), we see also that an input error given in the altitude of the airplane, H, propagates to h, the altitude of point P of the CAD entity.

Finally, for each point P of the CAD entity, we can find its height by subtracting from h, the DTM height at that point (which was evaluated in section *4.6.3 Computation of the 3D heights of the entities edges*):

$$CAD\,entity\,height\big|_P = h\big|_P - DTM\,\,height\big|_P \qquad (14)$$
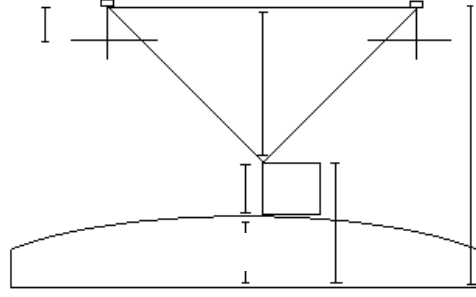

Fig. 16. Depth computation in the rectified stereo pair.

In order to apply this method of depth computation and CAD entity height evaluation to our case, we need to find a sufficient number of image point correspondences in the stereo pair. As mentioned in section 4.6.2, Stereo Images Rectification (Warping), these point correspondences can be found along the *epipolar* lines of the rectified stereo pair, inside the image area were the CAD entities are projected. We are interested in obtaining the point with the highest depth *Z* of the interior of each CAD entity (such as buildings). This higher value is selected to certify that this feature corresponds to the roof and not to the floor of the entity. As referred before, we are not yet addressing the problem of reconstructing with a sufficient level of detail, the topology of the roofs of the buildings. In fact, in this paper, we evaluate the heights of each CAD entity, assuming that the roofs are simple pyramidal objects.

Our method of finding a sufficient number of image point correspondences in the stereo pair, goes as follows.

1. With the knowledge of the left and right cameras pose (see section 4.6.4, Computation of the absolute cameras pose of the stereo pair), we project the vertexes of each CAD entity into each rectified stereo pair image, in order to create a projected convex hull of each entity, in the image domain (see Fig. 21a).

2. For each entity projected convex hull in both the left and right images, we find feature points inside these image areas, by using the Cany Edge Detector (Heath 1997), which finds "good features" in one image that need to be matched in the other, namely in the corners and roof edges of the buildings.

3. For each feature found inside the projected convex hull of the left image, we find the correspondences in the projected convex hull of the right image, which lie in the same *epipolar* line and within a nearby region of the feature points. To find such correspondences we have used a *template matching* technique.

4. For each point correspondence found, we compute the stereo disparity d and evaluate the proportional depth Z of the corresponding entity world point, as well as its height, by using equations (12 and (13). We are then able to determine the height of all known world points inside each CAD entity.

5. From this knowledge, we can find an estimate of the highest world point of each entity, so that the 3D volume of each entity can be found, assuming a simple topology for the roofs.-l
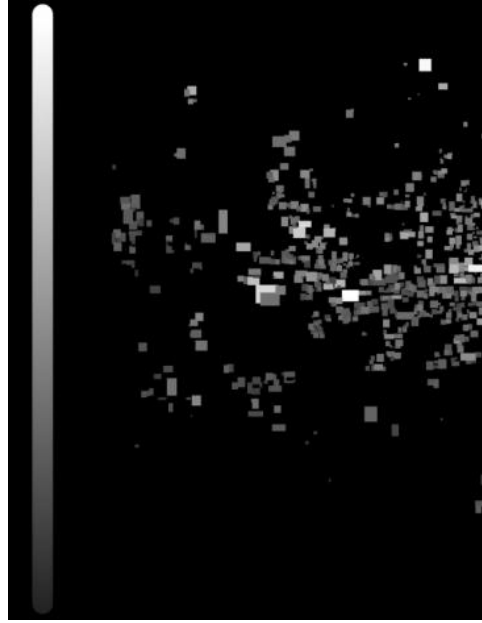

Fig. 17. Grey-scale height map of the CAD entities – a brighter value means a taller height and a darker value, a shorter one.

In Fig. 17, we depict the computed height map of the CAD entities of our problem. In the case of buildings whose height was not computed by finding feature point correspondences inside the projected entities convex hulls, which may be caused by the lack of stereo superposition in the area (which accounts for most of the cases), or by lack of success in the *template matching* (a minority of cases) when finding point correspondences, we simply project the known 2D polylines of the entities that have failed our basic algorithm, respectively, in the left image using the left camera parameters and, in the right image, using the right camera parameters (see Table 1 for the 3D reconstruction results of the Moledo Urban area). Then, we evaluate the stereo disparity along those polylines and, consequently, derive the depth and the height of those CAD entities along their polylines, using equations (12) and (13).

4.6.6 Extraction of feature points from the entities area on the image

The feature points are characteristic points on the image, such as corners of buildings and abrupt changes in luminance. Our technique requires that, given a number of features in one of the stereo images pair, there is the need to find the

matching features in the other image pair. Strong feature points are those that indicate a strong luminance variation, having high *eigen values*. This type of points is used to make the feature matching process easier, since to proceed with feature matching using random feature points, would maximize the probability of finding incorrect matches in the stereo pair, because of their lack of differentiation from other neighbour pixels. This way, we force the points used in matching to be strong features, which are easily detected in their neighbourhood area.

The Canny edge detector is used to detect strong feature points. We require the algorithm to find feature points with high *eigen-value* surrounded by low *eigen-values*. This way, we may find a notable point in the middle of non-notable points. This point is then a transition point between two regions of distinct luminance intensities (Fig. 18).

4.6.7 Matching Feature Points in the other Image of the Stereo Pair

Feature matching is the process of finding a feature point from one image of the stereo pair, in the other image. For this correspondence process we have adopted the *Template Matching* technique (Owen 2002). This technique is used to de-

Fig. 18. "Good" (strong) features found by the Canny Edge Detector.

termine the level of similarity between two images and consists of a *patch* or *template* (a rectangular or circular area of the image that involves the feature point, taken, in our case, from one of the stereo pairs), which is convoluted across the image where the search for successful matches is taking place, computing a specific measure. A minimization or a maximization of such measure corresponds to a found template match. From the literature review (Laganiére 2002), we know that there are various possible methods to accomplish this technique. One of such methods is the Mean Square Error (MSE), which is mostly used when measuring image degradation. In this model we measure the dissimilarity $c(I, P)$, between the template image P and the current image I, were we are searching for matches:

$$c(I,P) = \left(\sum_x \sum_y \left(I(x,y) - P(x,y) \right)^2 \right)^{\frac{1}{2}} \qquad (15)$$

When using the MSE measure, similarity converges to a value of 0. By adding the fact that this method is not luminance invariant, a better approach to achieve effective *Template Matching* is the evaluation of the cross-correlation coefficient ρ. When computing this coefficient, we must pre-evaluate two parameters for each image patch: its mean μ and standard deviation σ. In fact, the patch/template image mean and standard deviation parameters can be pre-computed off-line to improve system performance. The mean expressions for both the template image P and the current image frame I are:

$$\begin{cases} \mu_I = \dfrac{1}{xy} \sum_x \sum_y I(x,y) \\[2mm] \mu_P = \dfrac{1}{xy} \sum_x \sum_y P(x,y) \end{cases} \qquad (16)$$

The standard deviation expressions are:

$$\begin{cases} \sigma_I = \left(\sum_x \sum_y (I(x,y) - \mu_I)^2 \right)^{\frac{1}{2}} \\[3mm] \sigma_P = \left(\sum_x \sum_y (P(x,y) - \mu_P)^2 \right)^{\frac{1}{2}} \end{cases} \qquad (17)$$

The cross-correlation coefficient can be computed as follows:

$$\rho = \frac{\sum_x \sum_y (I(x,y) - \mu_I)(P(x,y) - \mu_P)}{\sigma_I \sigma_P} \qquad (18)$$

This method is used to obtain a degree of similarity, which is noise and luminance invariant and its complete description can be found in (Owen 2002).

In our case, the templates obtained from say, the left image of the stereo pair, are not matched in the entire right image, not only to minimise the computation time involved, but also because success of the algorithm will depend on minimising the similarity that wrong templates may have with the source patch (that is, minimizing false positives in the template matching process). Thus, to simplify the search and matching process, we can use a restricted search area where the correct template should exist. From the *epipolar* geometry principles, we know that, after the stereo images rectification process (section 4.6.2), the corresponding feature point must exist on the same *epipolar* (scan) line, restricting the search to a single image row, which provides the desired efficiency in our technique.

Fig. 19. Image normalization: Left - before; Right – after.

A stereo images pair shows luminance differences that may hinder the matching process, so we have normalised the luminous intensity over the stereo pair, which is given by the following equation:

$$P_{out} = (P_{in} - c)\left(\frac{b - a}{d - c}\right) + a \qquad (19)$$

where:

- P_{out} and P_{in} are the normalized and original pixels respectively;

- a and b are the minimum and maximum pixel values that the image type allows (in our case, 0 and 255 are the lower and upper limits for grey level images);

- c and d are the lowest and highest pixel values present in the image.

To avoid that outlying pixels, either with a very high or with a very low value, severely affect the value of c and d, we ignore 5% of highest and lowest pixel values in the image. The results are shown in Fig. 19.

We have assumed that the image stereo pair is taken in the same flight, by the same camera, but with a difference in the time they are made. This creates the problem that, moving objects that may be associated to "good features to track", might be captured in both images of the stereo pair, whereas
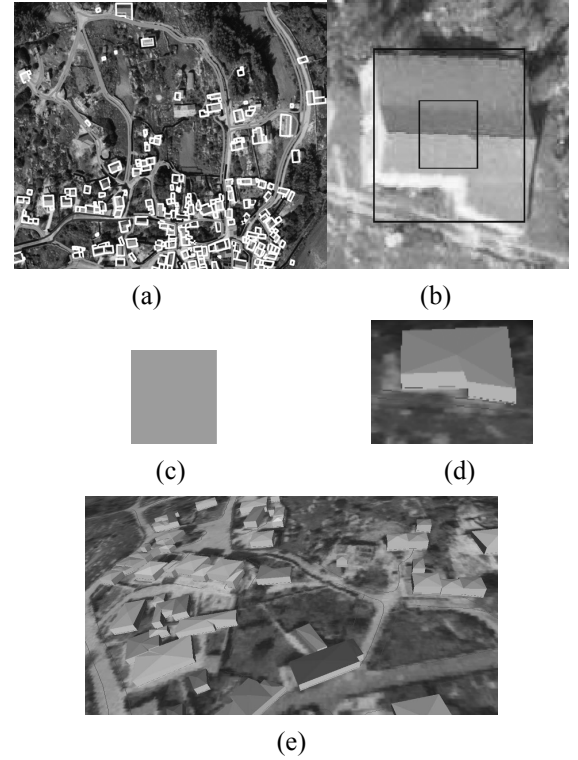

Fig. 20. Entities Roof Colour Extraction: from top to bottom and left to right: (a) CAD entities projection on the image; (b) close view of the image area were the roof colour is extracted; (c) resulting colour; (d) 3D reconstructed model of a building; (e) view of several reconstructed buildings.

our technique requires that those features correspond to fixed objects. However, in our technique we are only interested in finding feature correspondences that lay in the interior of the CAD entities contours, that is, in the interior of buildings roofs as seen from the air. Therefore, in our approach, we neglect the occurrence of moving objects, for those cases, which we think is a fair assumption.

4.7 Entities Roof Colour Extraction and Texture Reconstruction (Panorama)

In order to give more realism to the resulting 3D model of the each CAD entity, an approximation to the real colour of the volumetric entity roof needs to be extracted, as well as the texture to be mapped on the DTM of 3D final model. For the roof colour extraction, we have used the entity's projected convex hull. This area is obtained by the technique described in section 4.6.4, (Computation of the absolute cameras pose of the stereo pair), by re-projecting each CAD entity polyline into one of the images of the stereo pair. We then return the average RGB value of all pixels present in the image, whish lie in the interior of the projected entity, which sets the entity roof colour (see Fig. 20). In Fig. 20, the smaller rectangle represents the area from where the average colour will be extracted.

For the panoramic texture evaluation, we have used a back-projection technique, based in the prior knowledge of the rectified image pair and of the CAD model of the area in study. Given the unknown panoramic texture, for each one of its 2D texel (texture elements) coordinates m (s, t), where s, is the horizontal dimension and t, the vertical direction, we need to find a mapping between each unknown texel m (s, t) and each observed distorted pixel p (u', v'), where u´ and v´ are distorted image coordinates, belonging to the left or right rectified image of the stereo pair. When we are able to find such mapping, we will only need to replace the RGB value of m (s, t) by the found RGB value at p (u', v'). The steps to complete this algorithm are as follows:

 Step 1:

Given the 2D CAD model of the area in study, compute its minimum and maximum x and y coordinates, whose integer difference (maximum – minimum), give us, respectively, L, the texture width in millimetres and H, the texture height in millimetres of the panoramic image.

Step 2:

For each texel m (s, t), compute its 4D homogeneous coordinate correspondence point P (x, y, z, w), in world homogeneous coordinates:

$$\begin{cases} x = T_x\, s - L\!\big/\!2 \\ y = -T_y\, t + H\!\big/\!2 \\ z = 0 \\ w = 1 \end{cases} \qquad (20)$$

In the previous expressions, T_x and T_y are the effective texel size in millimetres respectively in the horizontal and vertical directions, L is the texture width in millimetres and H the texture height in millimetres.

Step 3:

Transform $P\,(x, y, z, 1)$, defined in the world reference frame, into $P_{cam}\,(x_c,\, y_c,\, z_c,\, 1)$, defined in the camera reference frame, using the camera extrinsic camera parameters obtained for each camera view ($M_{ext} = [R\ T]$):

$$P_{cam} = M_{ext}\, P \qquad (21)$$

Step 4:

Obtain true pixel image coordinates $p\,(u, v)$, that is, coordinates with ideal projection, from $P_{cam}\,(x_c,\, y_c,\, z_c,\, 1)$ (equation 6), by:

$$u = -\frac{f \cdot x_c}{S_x \cdot z_c} + c_x$$
$$v = -\frac{f \cdot y_c}{S_y \cdot z_c} + c_y \qquad (22)$$

Since all cameras exhibit considerable lens distortion, mainly of radial nature (Trucco 1998), we have to tackle this issue. Lens distortion can be expressed using four coefficients: two radial distortion coefficients k_1, k_2, and two tangential ones p_1, p_2. If we consider (u',v') as being the corresponding real observed distorted pixel image coordinates and (x',y') the ideal distortion-free image physical coordinates in millimetres, we have:

$$x' = \frac{S_x(u - c_x)}{f}$$
$$y' = \frac{S_y(v - c_y)}{f} \qquad (23)$$

and,

$$r^2 = x'^2 + y'^2$$
$$u' = u + (u - c_x)\left[k_1 r^2 + K_2 r^4 + 2p_1 y' + p_2\left(\frac{r^2}{x'} + 2x' \right) \right] \qquad (24)$$
$$v' = v + (v - c_y)\left[k_1 r^2 + K_2 r^4 + 2p_2 x' + p_1\left(\frac{r^2}{y'} + 2y' \right) \right]$$

Since we need to take in consideration lens distortion, so that the back projected image maintains consistency with the real observed camera image, we need to apply equation (24) to the ideal (distortion-free) pixel image coordinates (u, v), leading to projected and distorted image coordinates (u', v').

Step 5:

Inspect p (u', v') in the left rectified image of the stereo pair, to see if it is found there or if it violates the limits of that image. In this last case, inspect the right rectified image of the same pair, to find p (u', v'). Replace the RGB value of m (s, t) by the found RGB value at p (u', v'). The end result is depicted in Fig. 22 Top-Right.

5. RESULTS AND DISCUSSION

5.1. MATCHING TESTES

We have made a test for our *template matching* algorithm in order to derive an optimal the patch size, by varying this parameter. For a universe of 10 identified feature points, we have used our *template matching* algorithm by only changing the size of the patch used, for a given fixed search area in the other stereo pair. The minimum size used was 4x4 pixel and the maxim size was 64x64 pixel. By testing for all the possible combinations of width and height, we came up with the results of the number of correct matches shown in Fig. 21. We have concluded from this simple study, that the best path size is 16x16 pixel, since with this type of patch it we are able to find the full 10 feature points, although more tests would be needed to obtain some statistical significance.
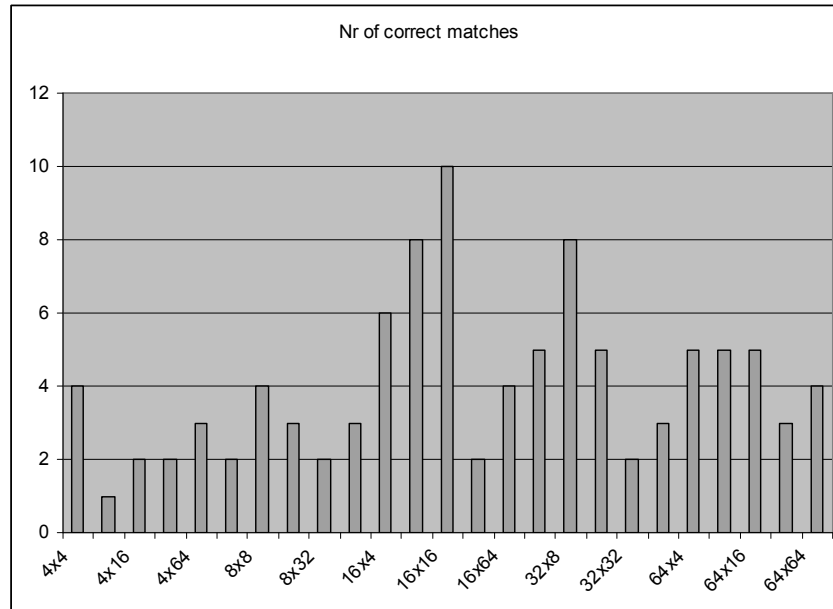

Fig. 21. Patch size versus Nr of correct matches.

A work reported in (Bastos, 2004), has confirmed that a good topology for patches in *Template Matching* is a circular one, for situations were there is fast camera motion in rotation, translation and approximation. For our case, where the camera motion is essentially translation, we obtain good results using quadrangle 16x16 patches. We would like to point out also that, initially, since the search problem in *template matching* has been classified, in our case, as being just 1D, we have designed horizontal patches. However, the number of correct matches was not as good as with the 16x16 patches.

5.2. 3D RECONSTRUCTION RESULTS

In table 1, we present the quantitative results of our 3D reconstruction algorithm, applied to an urban area in Moledo, a region in the West of Portugal, whose results have already been presented in Fig. 4. We see that out of 992 CAD entities representing buildings, 833 (83,97% of the total urban area) were identified in an area of stereo superposition, where our technique is applicable. In this area, all the input 2D CAD entities have been processed and their 3D height reconstructed, although only 695 (70,06% of the total), have had their height evaluated by our complete algorithm, based in *epipolar geometry*, *homography* and *template matching*. These 70,06% of entities, account for the cases were stereo superposition exists, the entities are closed polylines and point correspondences discovery in the stereo pair via *template matching*, has succeeded. For the remaining cases of CAD entities (138 or 13,91% of the total of entities), we have not used the *template matching* technique in the 3D reconstruction process (see section 4.6.5). Fig. 22 depicts the final result of our reconstruction process and Fig. 23, shows an overall view of the results of the algorithm applied to the Moledo Region.

	Number of CAD entities	Reconstruction Rate
Total Number of CAD entities (buildings) of input CAD model	992	100,00%
Number of CAD entities (buildings) without stereo superposition	159	16,02%
Number of CAD entities (buildings) reconstructed (with stereo superposition)	833	83,97%
Number of 3D reconstructed CAD entities (with stereo superposition) from *epipolar geometry + homography + template matching*	695	70,06%
Number of 3D reconstructed CAD entities (with stereo superposition) from *homography* only	138	13,91%

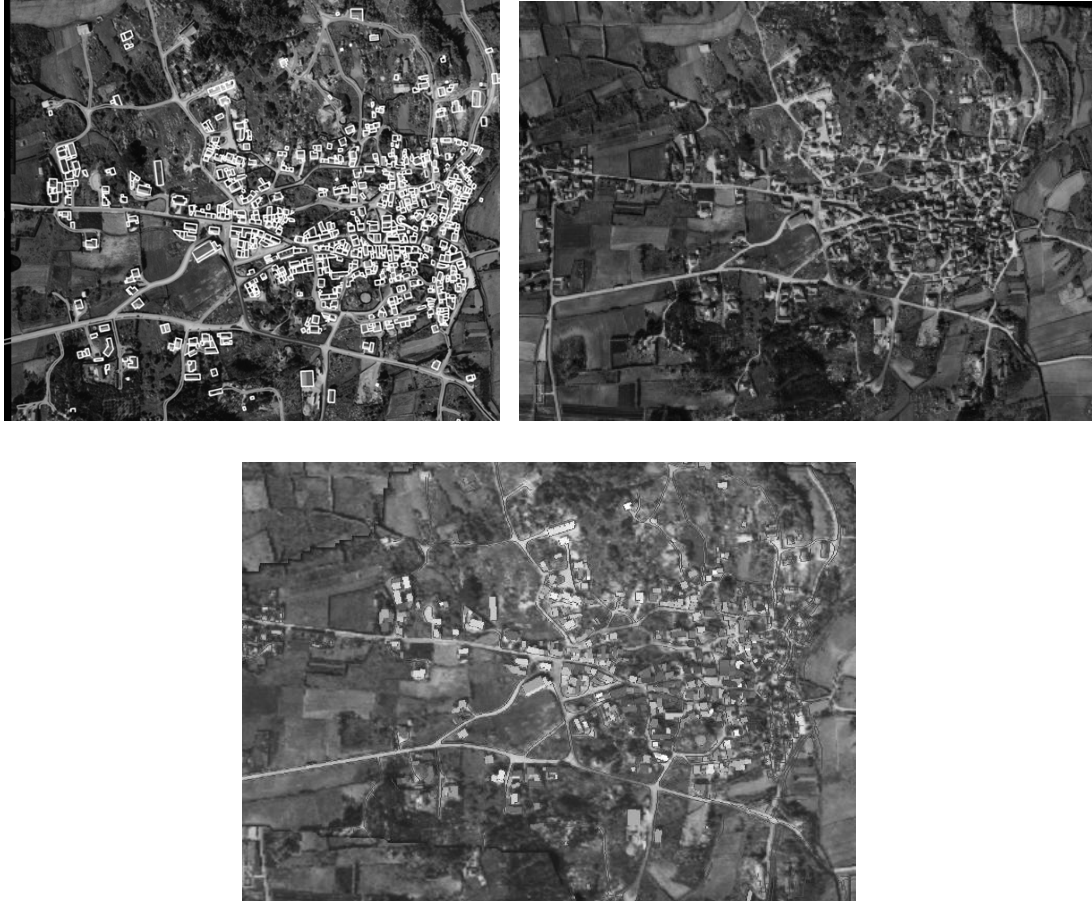**Table 1 -** 3D reconstruction results of the Moledo Urban area.

Fig. 22. Top-Left: 2D CAD entities projected on the left image; Top-right: reconstructed panoramic texture of the urban area; Bottom: orthogonal view of the 3D model generated by our system, using the reconstructed texture.

Fig. 23. A perspective of the 3D reconstructed Urban Plan.

6. CONCLUSIONS AND FUTURE WORK

The technique presented in this paper presents a contribution to the problem of generating 3D models of large urban areas, which has applications in the simulation of such large virtual environments. We have described a three-dimensional

reconstruction process, which takes only a pair of aerial photos, the respective DEM and 2D CAD model of the area, as input data, plus some limited required user input information. The described system does not require prior knowledge of the urban area and the necessary manual work to design a 3D urban model is substantially reduced. Therefore, we are able to build low-cost 3D models of any urban space using a reduced number of input information for image rectification and camera calibration, which is a common task in areas such as digital photogrammetry.

Regarding future work, we aim at integrating our technique in the SEDRIS framework. We want also to broaden our scope and apply our method to other related problems, such as to obtain information about the buildings facades extracted from video streams captured on the urban site. We envisage again the use of stereo pairs, *epipolar* geometry principles and feature detection by *template matching*, following a similar approach as developed by Debevec (Debevec 1998), which, from a shaded 3D model and a set of photographs, taken from known positions, of the real object that corresponds to the model, obtains a textured mapped 3D model. We are considering two possible approaches to solve this problem. A low level approach where the extracted facades textures (taken from the video stream) would be applied directly onto the entities facades assumed to be planner or, alternatively, a more complex approach where we would be able to change the entities facade shapes according to the automatic depth calculation of the facades points. Another line of work is the possibility to reconstruct the volumetric entities roof shape based on the aerial stereo image pair and the assumption of certain typologies of geometric templates for the roof shape. Notice that at the present stage of our research, the roof shape is assumed to be of a simple pyramidal topology, which limits the generation of realistic volumetric descriptions of buildings of the urban area.

The feature extraction and matching achieved with our algorithm is also not ideal yet. Feature points are not always found and false positive matches sometimes occur. Since automatic feature matching is an important step in our technique, we plan to review the algorithm. A considerable amount of work is already being done in this research area, namely the Scale Invariant Feature Transform (SIFT) (Lowe, 2004). This solution allows robust and precise matching from a sparse collection of pre-processed features.
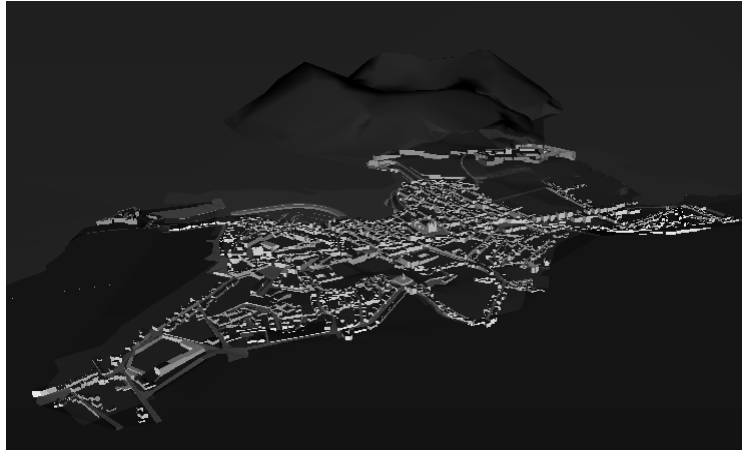
Fig. 24. The Virtual Angra do Heroísmo Model with 109 streets and more than 2004 buildings (Dias, 2004).

The interest of combining these techniques of automatic volumetric building and roof reconstruction (taken from DTM, CAD models and airborne stereo pairs), with the algorithm of automatic facade texture evaluation from video (under development), can be illustrated with the following example. Some of the authors have been involved in the development of the project "Virtual-Urban.com" (Dias, 2004), which has created a 3D virtual version of the historical zone of the city of Angra do Heroísmo in Azores, Portugal (Fig. 24). The system is interactively accessible through the Internet[2]. The model of this urban area includes five "Freguesias" (Conceição, São Bento, Sé, São Pedro and Santa Luzia), with 109 streets and more than 2400 buildings, whose textured façcdes and volumetric descriptions are accurate. With the initial availability of the DTM and the 2D CAD representation o the urban area subject to study, the development of the Virtual Angra model has required 5000 man-hours, including the on-site data acquisition stage (topographical survey and systematic facade photography for more than 2400 buildings) and the exterior (simplified) volumetric modelling of all those buildings. With our semi-automatic algorithmic techniques, we envisage to reduce by 75% the total man-power effort currently used to develop 3D models of urban areas, such as this one.

7. ACKNOWLEDGEMENTS

---

[2] Virtual Angra do Heroísmo can be accessed interactively in www.urban-virtual.com

8. REFERENCES

Abdel-Aziz, Y.I. and Karara, H.M. (1971), "Direct Linear Transformation into Object Space Coordinates in Close-Range Photogrammetry", in Procedures of Symposium of Close-Range Photogrammetry, January, pp 1-18.

Arbouche, S., (1999), "Feature Point Correspondence: A Matching Constraints Survey", in M.C.S Thesis, University of Ottawa.

Baker, Timothy J. (1989), "Automatic Mesh Generation for Complex Three-Dimensional Regions Using a Constrained Delaunay Triangulation", Engineering with Computers, Springer-Verlag, Nr 5, pp.161-175.

Bernardes, P., Laurentina, S., Sérgio C., Rodrigues, R., Diogo, A., Teixeira, J., (1996), "Virtual Reality, Urban Planning and Historical Town Centers", COMPUTER GRAPHIK topics, Volume 8, No. 4/96, August.

Bastos, R., Dias, J. M. S. (2004), "Fully Automated Texture Tracking Based on Natural Features Extraction and Template Matching", ISMAR 04 Demo, 3rd IEEE and ACM International Symposium on Mixed and Augmented Reality, in Arlington, VA, USA, Nov. 2-5.

Brakke, K. A. (2004), "Surface Evolver Manual", in Mathematics Department, Susquehanna University, Selinsgrove, PA 17870, 2.24 edition, October.

Dennis, T, Papadimitiou D. (1996), "Epipolar line estimation and rectification for stereo image pairs", IEEE, Trans. Image Processing, pp. 672-676.

Debevec, Paul E., Borshukov, George, and Yu, Yizhou (1998). Efficient View-Dependent Image-Based Rendering with Projective Texture-Mapping. In 9th Eurographics Rendering Workshop, Vienna, Austria, June.

Dias, P. (2003), "Three dimensional Reconstruction of Real World Scenes Using Laser and Intensity Data", in PhD Thesis, Universidade de Aveiro.

Dias, J.M.S., (2004) "3D Interaction via the Internet in the Virtual Angra do Heroísmo Model", JOIN'04, 19-21th April (in Portuguese).

Website: www.urban-virtual.com visited in 1.11.2005

Faugeras, O., Deriche, R., Zhang, Z. (1994), "Recovery of the Epipolar Geometry for an Uncalibrated Stereo Rig", ECCV, pp. 567-756.

Fischler, M.A., Bolles, R.C (1981), Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM, Vol. 24, No. 6, pp. 381-395.

Golub, G. H., and Van Loan, C. F. (1993), "Matrix Computations", in 2nd ed. Johns Hopkins University Press,

Google Hearth, "A 3D interface to the planet", Website: earth.google.com, visited in 1.11.2005

Heath, M., Sarkar, S., Sanocki, T. and Bowyer, K.W. (1997), "A Robust Visual Method for Assessing the Relative Performance of Edge-Detection Algorithms", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 12, December, pp. 1338-1359

Kasser, M., Ebels, Y. (2002), "Digital Photogrammetry", pp. 212-217, Tayler and Francis ed.

Laganiére, R. (2002), "A Comparative Study of Some Matching Strategies", University of Ottawa.

Longuet-Higgins, H.C. (1981), "A computer algorithm for reconstructing a scene from two projections," Nature, vol. 293, September, pp. 133–135.

Lowe, David G. (2004), "Distinctive Image Features from Scale-Invariant Keypoints", in International Journal of Computer Vision, 60, 2, pp 91-110.

Nakagawa, M., Shibasaki R., Kagawa Y. (2002), "Fusing stereo linear CCD image and laser range data for building 3D urban model, ISPRS Commisssion IV, WG IV/7.

OpenCV, Intel Open Computer Vision Library, Website: www.intel.com/research/mrl/research/opencv/ visited in 1.11.2005

Owen, C. B., Xiao, F. and Middlin, P. (2002), "What is the best fiducial?", in The First IEEE International Augmented Reality Toolkit Workshop, Darmstadt, Germany, 29th September.

Pollefeys, M. (2001), "Obtaining 3D Models With a Hand-held Camera], in Lecture Notes SIGGRAPH Course, Sunday 12th August.

SEDRIS, The Source for Environmental Representation and Interchange, Website: http://www.sedris.org/ visited in 1.11.2005

Trucco, E., Veri, A. (1998), "Introductory Techniques for 3D Computer Vision", Prentice Hall, New Jersey.

VRML, Website: http://www.web3d.org/x3d/specifications/vrml/ISO_IEC_14772-All/ visited in 1.11.2005

Zhang, Z. (1999), "Flexible Camera Calibration By Viewing a Plane From Unknown orientations", in International Conference on Computer Vision (ICCV'99), Corfu, Greece, September, pp 666-673.

3D URBAN, "Sistema de Gestão e Visualização 3D de Planos e Projectos URBANos – Projecto POSI 2010 (in Portuguese), Website: http://www.3d-urban.com visited in 1.11.200.