Automatic 3D Reconstruction of Urban Areas, by using Epipolar Geometry and Template Matching

José Miguel Salles Dias
Miguel.Dias@iscte.pt

Rodrigo Vicente
Rodrigo.Vicente@iscte.pt

João Correia
Joao.Correia@iscte.pt

Rui Cruz
Rui.Cruz@iscte.pt

Abstract
In this work we describe a novel technique for automatic 3D reconstruction of urban areas, from airborne stereo-pair images whose output is VRML. The main challenge is to compute the relevant information - building’s height, 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, without no knowledge of the camera pose or its intrinsic parameters (other than the camera lens distortion parameters). To achieve this, we have used epipolar geometry, 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, Feature Matching, Template Matching, Airborne Images, Delaunay Triangulation, Epipolar Geometry, Digital Photogrammetry

1. INTRODUCTION
Digital Surface Models (DSM), Digital Terrain Models (DTM) and Orthophotomaps [Kasser 02] of a given geographical are currently available on a commercial basis, and are regarded as an important tools used in Urban Planning tasks. Often these models are obtained from airborne stereo pair photos, using known landmarks specially set in the terrain, manually identifying such landmarks as features in the obtained stereo pair, and applying epipolar Geometry [Trucco 98] 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 Figure 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 94] or laser-based techniques [Nakagawa 02]. With this work, we are aiming to obtain, by automatic 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 (see Figure 2):
- airborne stereoscopic photos (Figure 3);
• 3D DTM (3D points in ASCII File), representing the terrain (Figure 4 top);
• DXF file, representing the CAD entities projected on the DTM (thus in 3D format), representing contours of buildings or rocks (closed polylines), rivers (open polylines), trees (points), etc (Figure 4 top).

The output of our engine is the volumetric representation of the Urban Space, described in VRML (Virtual Relational Model Language) [VRML], representing the urban space the most precise and realistically as possible (Figure 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 functionality of each system module, towards 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.

2. BACKGROUND

In the recent years interesting achievements were made regarding the 3D reconstruction of real spaces. Although most of the works made in this area 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 thesis [Dias 03], “Three Dimensional Reconstruction of Real World Scenes Using Laser and Intensity Data”, Dias presented a system capable of generating complete, high-resolution three-dimensional models of real world scenes combining inputs from passive intensity images and active range sensors. Therefore, Dias work use passive techniques to compensate the limitations of active techniques in specific steps of the reconstruction and vice-versa.

In the SIGGRAPH Course Lecture Notes “Obtaining 3D Models With a Hand-Held Camera”, [Pollefeys 01], 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, 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 urban area model, which enables the user to perform virtual navigation. This model can also be used as an interaction metaphor for a 3D Geographical Information System (GIS).

Figure 4 depicts the reconstructed 3D VRML model, obtained by our system, of the area shown in Figure 3. By using a Boundary Representation (BRP), the VRML model represents the DTM plus all the entities provided by the 3D AutoCAD model, with the height of the buildings computed by our 3D reconstruction technique. Figure 5 shows the modular architecture of our system.

4. SYSTEM DEVELOPMENT
4.1 Input Module
The Input Module reads the geometrical and topological information of the entities of the AutoCAD model and the 3D DTM. Another group of modules (DTM Generator, 3D Reconstruction and Roof Colour Extraction of Entities), is responsible for the core system processing of the integrated with the airborne stereo photos. The Output Module is responsible for the creation of the VRML files with the 3D description of the urban area.

4.2 DTM Surface Generator Module
The development of this module, which generates the DTM, is justified by the fact that the points read in the Input Module, have no specific order. We may consider them as a cloud of points. In the other hand, the Output Module generates triangles, so they can form a BRP mesh. Therefore, this module is responsible for ordering the points in groups of three, formed by their proximity. There is an algorithm that produces these groups, called Delaunay Triangulation. This triangulation algorithm, based on the Voronoi Diagram, performs exactly what is required. The algorithm generates triangles from a cloud of points, which verify the following condition: a triangle

\[ \text{if point } p_k \text{ is in the exterior of the circle defined by } p_i, p_j, \text{ and } p_l \Rightarrow p_k \text{ is a vertex of the triangle.} \]

4.3 3D Reconstruction Module
The proposed 3D reconstruction technique complements the existing 3D CAD information projected onto the DTM, through the computation of the entities height. The reconstruction is accomplished using a stereo pair of airborne images and the available 3D entities’ points and is mathematically based in epipolar geometry. These 3D entities’ points are the vertexes of each entity that reside over the DTM surface (namely, the contours of the buildings over the DTM). The images are two aerial stereo photographs, obtained by the same camera (or by distinct cameras with equal intrinsic parameters) at a given flight altitude.

4.4 Output Module
The Output Module handles the creation of VRML files which integrate the DTM and the 3D urban volumetric models. There are two writers: the DTM Writer that generates the DTM surface, represented in VRML as a mesh of Delaunay triangles using the IndexedFaceSet node; and the 3D Volumetric Writer, responsible for the 3D volumetric representation of the entities. In the specific case of buildings, given that we are only interested in the exterior volumetric description of each one, the associated entity is a parallelepiped.

\[ p_j, p_k, p_l \text{ are points in the exterior of the circle defined by } p_i. \]

\[ p_k \text{ is a vertex of the triangle.} \]

\[ \text{if point } p_k \text{ is in the exterior of the circle defined by } p_i, p_j, \text{ and } p_l \Rightarrow p_k \text{ is a vertex of the triangle.} \]

\[ p_k \text{ is a vertex of the triangle.} \]

1 The intrinsic parameters of the camera are: the focal length, the pixel coordinates of the centre of the image viewport, the pixel scale factors in the two coordinate axes of the image viewport and the lens distortion parameters.
4.5 Epipolar Geometry

With epipolar geometry, we are able to develop a two-camera model where can maintain a relation between world points and their projections on the images planes. In Figure 8, ε and εr, the left and right epipoles, are given by the intersection of the line segment that joins Oi and Or, 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.

![Figure 8 – Two cameras model of a stereo pair](image)

The practical importance of epipolar geometry consists in the fact that epipolar plane defined, in Figure 8, by the three line segments [Oi, Or], [Oi, Pr] and [Oi, Pl] intersects each image plane [I’i] and [I’r] (of the pair) in two lines, respectively, [p, e] and [p, e], called epipolar lines (the line segment [Oi, Or] is also known as the baseline). Therefore, given a projection p 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 (Figure 8). At the same time, we know that the world point P can be found in the line which passes through p_r and Or. If we know Or and Or, and we are able to find the projections p_r and p_l 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 Or and p_r, and, by Or and p_l. This is known as the triangulation technique in 3D reconstruction. As mentioned, this technique requires the evaluation of Or and Or. If we know Or in the world reference frame, Or can be found by applying a translation and a rotation to the former. The parameters that define these two transformations matrices are known as the extrinsic parameters of the stereo camera system.

4.6 The 3D Reconstruction Algorithm

The 3D reconstruction algorithm is divided into six steps:

1. Computation of the Fundamental Matrix, F;
2. Rectification (warping) of the stereo images pair;
3. Projection of the input CAD entities’ world coordinates, that correspond to 3D projections of buildings in the DTM, into the left image pixel coordinates;
4. Extraction of feature points inside the area that correspond to the projection of each CAD entity, in the left image;
5. Matching of those feature points in the right image;
6. Depth computation of each feature point by triangulation and, as a consequence, entity height determination.

4.6.1 Computation of Fundamental Matrix, F

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

\[ u_r = F p_l \]  

Where \( F \) 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:

\[ p_r^T F p_l = 0 \]  

![Figure 9 – Non-warped model](image)
eliminated, $F$ is recalculated to get a better approach. Thus, the most pair of points we give, the higher is the precision of the computation.

### 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. One goal of warping is to be able to make the simplification of the geometry of the stereo pair. In Figure 9 and 10, we see, respectively, the general model of a stereo pair and the rectified (warped) model. Figure 11 depicts the resulting rectification of one of the stereo images used in our case study. In the rectified stereo pair, the image planes $\Pi^1$ and $\Pi^2$ 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 $p_{iL}$ on say, the left image, we need to find the corresponding (matched) feature point, $p_{iR}$, on the right image. From the epipolar geometry (see section 4.5), we know that this matched point $p_{iR}$, 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 $p_{iL}$. 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.

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 Figure 11).

### 4.6.3 Projection of World CAD Entities Coordinates into Image Pixels

The transformation from the CAD model world coordinates into image pixels is required since we need to isolate the area of an entity (for example, a building surrounding) in the image plane. In order to make a detailed stereo study of the entity in focus, this minim area involving the entity is needed, in order to find “good” features and study their matching with the other image of the stereo pair.

For this, we have used the principles of projective geometry, which gives us the relation between an object point $M$ in the world reference frame, and its projection $i$ in the image domain. To achieve this projection, the intrinsic and extrinsic camera parameters must be evaluated. The projection is then obtained with a simple intersection of the ray crossing the camera position and the world point and hitting the image plane. Figure 12 shows this projective geometry case. Given world point $M$ on the DTM, its projection $m=[m_x, m_y]$ on a given projection plane is:

$$
\begin{align*}
    m_x &= f \frac{d_x}{d_z} \\
    m_y &= f \frac{d_y}{d_z}
\end{align*}
$$

(3)

where $f$ is the focal distance and $d=[d_x, d_y, d_z]$, is the model point $M$ coordinates in the camera reference frame. Subsequently, it is necessary to transform from $m$ (in millimetres) in the projection plane, into $p$ (in pixels) in the image plane, given by:

$$
\begin{align*}
    i_x &= \frac{m_x}{S_x} + C_x \\
    i_y &= \frac{m_y}{S_y} + C_y
\end{align*}
$$

(4)

In (4), $S_x$ and $S_y$ are, respectively, the pixel scale values in width and height and, $C_x$ and $C_y$, the coordinates of the image centre.

Our algorithm requires the user to initially and manually define the projection of both cameras in the CAD model in the world reference frame (we are only interested in the $X$ and $Y$, 2D coordinates of such projection), defining:

$$
\begin{align*}
    X_{is}, \text{ for one camera} \\
    Y_{is}
\end{align*}
$$
\[
\begin{align*}
X_{2x} & \text{ for the other camera} \\
X_{2y} & \\
\end{align*}
\]

It requires also the manual specification of two more corresponding points \(i_1\) and \(i_2\) defined in the image pair and the associated CAD model point \(M\), thus defining:

\[
\begin{align*}
\{i_{x1}, i_{y1}\} & \text{ for one stereo pair} \\
\{i_{x2}, i_{y2}\} & \text{ for the other stereo pair} \\
M_x & \text{ for the CAD model} \\
M_y & \\
M_z & \\
\end{align*}
\]

Regarding CAD model point \(M\) (in world coordinates), we have to take a special attention. In fact, if we go from one image to the other in the rectified stereo pair, corresponding points lie in the same scan line (or the same \(y\) in pixel coordinates), since both camera reference frames are parallel. But the CAD model reference frame is not oriented in the same way. Before projecting world points to image points, the world reference frame must be reoriented so that it is parallel to both cameras reference frames. This reduces to rotate the CAD model points in relation to the \(zz'\) axis of one of the cameras reference frame, by an angle alpha, which is computed so that the \(yy'\) axis of both the world reference frame and the camera reference frame, are equal. Figure 14 illustrates this rotation. So after applying this rotation transformation to world point \(M\), we stay with world point \(M'\).

Looking at Figure 12, and by triangle similarity, we can derive the following equations, that, for each image and each camera, evaluate the pixel coordinates \(i_1\) and \(i_2\) as projection of world coordinates \(M'\):

\[
\begin{align*}
\frac{(i_{x1} + C_x)}{S_x} &= \frac{f(X_1 - M'_{x})}{(Z_1 - M'_{z})} \\
\frac{(i_{y1} + C_y)}{S_y} &= \frac{f(Y_1 - M'_{y})}{(Z_1 - M'_{z})} \\
\frac{(i_{x2} + C_x)}{S_x} &= \frac{f(X_2 - M'_{x})}{(Z_2 - M'_{z})} \\
\frac{(i_{y2} + C_y)}{S_y} &= \frac{f(Y_2 - M'_{y})}{(Z_2 - M'_{z})}
\end{align*}
\]

In these equations we have the following knowns and unknowns:

- \(i_1\) and \(i_2\) are known (by user input);
- \(X_1, Y_1, X_2\) and \(Y_2\), respectively, the 2D projections of each camera position in the CAD model in the world reference frame, are known (by user input);
- \(M'_{x}, M'_{y}\) and \(M'_{z}\) are known (point \(M\) is introduced by the user and \(M'\) is just \(M\) after a rotation of alpha);
- \(C_x\) and \(C_y\), the center of the image reference frame (intrinsic camera parameters) are known (is computed as the geometric center of the image);
- From the Stereo Images Rectification (Warping) process (section 4.6.2), the \(Z\) coordinate (height) of each camera position in the in the world reference frame is equal \((Z_1 = Z_2)\) and unknown.
- \(f\), the focal distance of the camera lens (intrinsic camera parameter), is unknown;
- \(S_x, S_y\), are pixel scale factors in width and height (intrinsic camera parameters), which are unknown.

So we have a system of four linear equations with four unknowns \((Z_1 = Z_2, S_x, S_y\) and \(f\)), which can be trivially solved by LUV decomposition.

The precision to the resulting image pixels after world points projection is not perfect, but still satisfactory for...
our feature extraction and matching purposes. It’s worth noticing that, at this stage, we have not yet dealt with the camera lens distortion, and that the manual process of inputting the projections of both cameras in the CAD model and of specifying two more corresponding points $i_1$ and $i_2$, in the image pair, and the associated CAD model point $M$, is prone to human error. The image pixels found are near the correct pixels, with a maximum distance error of 10 pixels, giving that the found cameras’ height is in the order of the 600 meters.

4.6.4 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.

Strong feature points are determined using their eigen-values. The Canny edge detector is used for this purpose [OpenCV]. 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. Figures 14 and 15 are examples of, respectively, good and bad features.

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

Feature matching is the process of finding a feature point in one image, corresponding to the same point in the other stereo image pair. For this correspondence process we have used one version of the template matching technique. The implemented algorithm consists of a patch (an image area that involves the feature point taken from one of the stereo pairs), which is compared across the stereo image where the search is taking place. That comparison is achieved by the taking the square of the sum of differences between all the pixels in the patch, with the pixels of each template area in the other stereo pair. The minimum value corresponds to the template were we find matching.

The templates used for this purpose are not extracted from the entire image, not only to minimise the computation time involved, but also, since success of the algorithm will depend on minimising the similarity that wrong templates may have with the patch. Thus, to simplify the matching process, we have obtained a restricted search area where the correct template should exist. Firstly, by warping, we know that the corresponding feature point must exist on the same scan line, restricting the search to a single image row (see section 4.6.2). To further restrict the search area, we need to compute a measure of the disparity between the stereo pair. By knowing three point correspondences in the pair, we can calculate the mean disparity of these, providing us the required measure. Assuming that the value of the disparity has low variance (for a terrain height that varies smoothly), we can predict a restricted area for a pixel.

Stereo images pair show luminance differences that may hinder the matching process, so we developed an algorithm to normalise the luminous intensity over the stereo pair, which is given by the following equation:

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

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.

Figure 14 – Bad Features

Figure 15 - Good (strong) features
To avoid that outlying pixels with either a very high or very low value severely affect the value of $c$ and $d$, we ignore 5% of highest and lowest pixel values present in image, and set these constants values with the left range of pixels. The results are shown in Figure 16.

### 4.6.6 Dept Computation of each Feature Point and Volumetric Entity High Determination

Each volumetric entity’s height has a single value, because we have considered the simplification that entities are all parallelepiped, with only one roof plane. This way, for each entity, we only need to determine one value to construct the entity volume.

The first step is to compute the depth of each feature previously extracted, by using the equation from [Trucco98]:

$$z = f \left( \frac{fb}{x_2 + C_x - x_1 + C_y} \right) S_x$$

Where:

- $x_1$ and $x_2$ are, respectively, the positions, in pixel, of a corresponding feature in the stereo pair. $x_1 - x_2$ is the disparity.
- $f$ is the focal distance, that is now known (see section 4.6.3);
- $b$ is the baseline, distance between the two camera positions; $C_x$ and $C_y$ is the centre of both images (which is equal for both images since their sizes are the same);
- $S_x$, $S_y$, are pixel scale factors in width and height, computed before
- $z$ is the height of the feature, that is, the distance in the vertical axis from the camera’s altitude to the feature point altitude ($z = H - h$, where $H$ is the altitude of the cameras and $h$ is the altitude of the feature).

Figure 17 illustrates the depth computation.

From the computation of altitudes for each feature of the set, the higher value is selected to certify that this feature corresponds to the roof and not to the floor of the entity. The height of the entity is the difference between the altitude of the higher feature and the altitude of average of the entity’s floor points. These last ones are extracted from the 3D entities points of the CAD Model.

Even without the exact height of the entities, we can conclude that the height computation by our algorithm is an approximated value of the real height.

### 4.5. Entities Colour Extraction Module

In order to give more realism to the VRML model, the real colour of the volumetric entity needs to be extracted. For this, we have used the entity’s interest area. This area

The algorithm then returns the average RGB value of all pixels present in the image containing the desired entity. This value is then applied to the superior plane of each entity in the final VRML model.

In Figure 19, the larger rectangle represents the minim area of the entity, while the smaller rectangle represents the area from where the average color will be extracted.

### 5. RESULTS AND DISCUSSION

#### 5.1. MATCHING TESTES

We have made a test for our template matching algorithm by varying the patch size. For a universe of 10 identified feature points, we have used our matching algorithm by only changing the size of the patch used, for a given fixed search area in the other stereo pair. The minim size used was 4x4 pixels and the maxim was 64x64. By testing for all the possible combinations of width and height, the results are shown below:
<table>
<thead>
<tr>
<th>Patch</th>
<th>Nº of correct matches</th>
</tr>
</thead>
<tbody>
<tr>
<td>4x4</td>
<td>4</td>
</tr>
<tr>
<td>8x8</td>
<td>4</td>
</tr>
<tr>
<td>16x16</td>
<td>10</td>
</tr>
<tr>
<td>32x32</td>
<td>2</td>
</tr>
<tr>
<td>64x64</td>
<td>4</td>
</tr>
<tr>
<td>8x4</td>
<td>2</td>
</tr>
<tr>
<td>16x4</td>
<td>6</td>
</tr>
<tr>
<td>32x4</td>
<td>5</td>
</tr>
<tr>
<td>64x4</td>
<td>5</td>
</tr>
<tr>
<td>16x8</td>
<td>8</td>
</tr>
<tr>
<td>32x8</td>
<td>8</td>
</tr>
<tr>
<td>64x8</td>
<td>5</td>
</tr>
<tr>
<td>32x16</td>
<td>5</td>
</tr>
<tr>
<td>64x16</td>
<td>5</td>
</tr>
<tr>
<td>64x32</td>
<td>3</td>
</tr>
<tr>
<td>4x8</td>
<td>1</td>
</tr>
<tr>
<td>4x16</td>
<td>2</td>
</tr>
<tr>
<td>4x32</td>
<td>2</td>
</tr>
<tr>
<td>4x64</td>
<td>3</td>
</tr>
<tr>
<td>8x16</td>
<td>3</td>
</tr>
<tr>
<td>8x32</td>
<td>2</td>
</tr>
<tr>
<td>8x64</td>
<td>3</td>
</tr>
<tr>
<td>16x32</td>
<td>2</td>
</tr>
<tr>
<td>16x64</td>
<td>4</td>
</tr>
<tr>
<td>32x64</td>
<td>3</td>
</tr>
</tbody>
</table>

We have concluded from this simple study, that the best patch size is a quadrangle patch, with 16x16, since it finds the full 10 feature points, although more tests are needed to obtain some statistical significance. A work reported in [Bastos 04], confirms that a good topology for patches in template matching is a circular one, for situations were we have fast camera motion in rotation, translation and approximation. For our case, were 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. RESULTS
In figures 4 and 19, we present the final result of our reconstruction process. In Figure 4, we depict an overall view of the results of the algorithm to a large urban area. In figure 19 we show the results of the algorithm for a the case of a single house. From left to right, we have (1) the CAD input information; (2) the stereo image pair, and (3) the reconstructed 3D volumetric entity.

Figure 19. Results of 3D reconstruction

6. CONCLUSIONS AND FUTURE WORK
The system presented in this paper describes a three-dimensional reconstruct process which takes only a pair of aerial photos and the respective Digital Terrain Model and CAD model of the area, as inputs, 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 three dimensional 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 calibration, which is a common task in areas such as digital photogrammetry. Regarding future work, our objective is to obtain information about the entities façades extracted from video streams captured in the terrain, again, using stereo pairs, epipolar geometry principles and feature detection by template matching. We are considering two possible approaches to solve this problem. A low level approach where the extracted façades textures (taken from the video stream) would be applied directly onto the entities façades 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. Notice that at the present stage of our research, the roof’s shape are assumed to be planar.

The feature matching achieved with our algorithm is also not ideal yet. The correct feature points are not always found. Since automatic feature matching is an important step in our technique, and should be ideally error-proof, we have concluded that this algorithm must be reviewed, by possibly adding a contour evaluation of the buildings in the image domain.
Figure 20. The Virtual Angra do Heroísmo Model with 109 streets and more than 2004 buildings

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 façade texture evaluation from video (under development), can be identified with the following example. Some of the authors have been involved in the development of the project “Virtual-Urban.com” [Dias 04], which has created a 3D virtual version of the historical zone of the city of Angra do Heroísmo in Azores, Portugal. The system is interactively accessible through the Internet.¹ 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 facades and volumetric descriptions are accurate. With the initial availability of the DTM and 2D CAD representation of the urban area in 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 modeling of all those buildings. With our algorithmic techniques, we envisage to reduce by 75% the total manpower effort currently used to develop 3D models of urban areas, such as this one.

7. ACKNOWLEDGEMENTS

The authors would like to thank José Manuel Fonseca from OA (Oficina de Arquitectura) Portugal, for his constant contributions and discussions about the inputs used in the system (aerial photos and AutoCAD model) and the overall system requirements.

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

8. REFERENCES


[OpenCV] www.intel.com/research/mrl/research/opencv/

