PhotoBuilder – 3D Models of Architectural Scenes from Uncalibrated Images

Roberto Cipolla, Duncan Robertson and Edmond Boyer
Department of Engineering
University of Cambridge
Cambridge CB2 1PZ

Abstract

We address the problem of recovering 3D models from uncalibrated images of architectural scenes. We propose a simple, geometrically intuitive method which exploits the strong rigidity constraints of parallelism and orthogonality present in indoor and outdoor architectural scenes. We show how these simple constraints can be used to calibrate the cameras and to recover the projection matrices for each viewpoint.

The projection matrices are used to recover partial 3D models of the scene and these can be used to visualise new viewpoints. Our approach does not need any *a priori* information about the cameras being used.

A working system called *PhotoBuilder* had been designed and implemented to allow a user to interactively build a VRML model of a building from uncalibrated images from arbitrary viewpoints.

1 Introduction

Considerable efforts have been made to recover photorealistic models of the real world. The most common *geometric* approach is to attempt to recover 3D models from calibrated stereo images [15] or uncalibrated extended image sequences [21, 1, 16] by triangulation and exploiting epipolar [13] and trilinear constraints [9, 20]. An alternative approach consists of visualisation from image-based representations of a 3D scene. This has been successfully used to to generate an intermediate viewpoint image given two nearby viewpoints and has the advantage that it does not need to make explicit a 3D model of the scene [22, 18, 8, 11, 6].

Facade [5] – one of the most successful systems for modelling and rendering architectural buildings from photographs – consists of a hybrid geometric and image-based approach. Unfortunately it involves considerable time and effort from the user in decomposing the scene into prismatic blocks, followed by the estimation of the pose of these primitives. However the high quality of the results obtained with the Facade system has encouraged others to design interactive systems.

In this paper we propose a much simpler approach to construct a 3D model and generate new viewpoint images by exploiting strong constraints present in the scenes to be modelled. In the context of architectural environments, the constraints which can be used are parallelism and orthogonality. These constraints lead to very simple and geometrically intuitive methods to calibrate the intrinsic and extrinsic parameters of the cameras and to recover Euclidean models of the scene from only two images from arbitrary positions. Our approach is similar to another interactive system [19] but exploits vanishing points [2] directly to recover the projection matrices.

2 Outline of the algorithm

A 3D model can be recovered from two or more uncalibrated images in the following four stages (see figure 1).

1. The user selects a set of image edges which are either parallel or perpendicular in the world. These primitives are precisely localised in the image using the image gradient information.

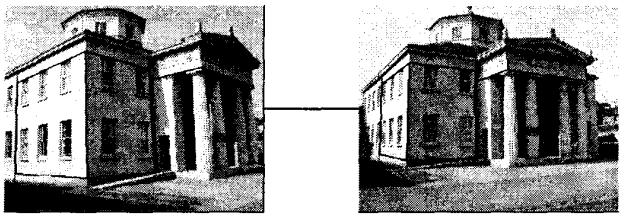
2. The next step concerns the camera calibration: the intrinsic parameters of the camera are determined for each image. This is done by determining the vanishing points associated with parallel lines in the world. Three mutually orthogonal directions are exploited to give three intrinsic parameters and the orientation of each viewpoint.

3. A projection matrix for each viewpoint is computed from the image edges and vanishing points. These matrices are further refined by exploiting epipolar constraints and additional matches to give the motion (a rotation and a translation) between the viewpoints.

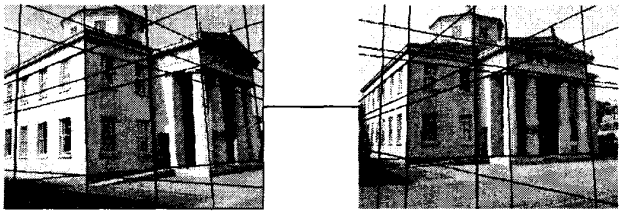
0-7695-0253-9/99 $10.00 © 1999 IEEE

1. Original uncalibrated photographs

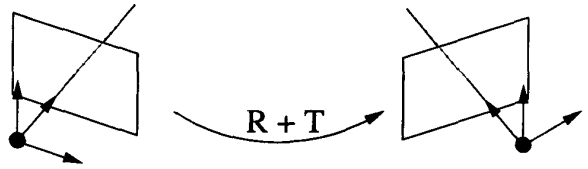
2. Primitive definition and localisation



3. Finding vanishing points and camera calibration



4. Computation of projection matrices and camera motion



5. Triangulation, 3D reconstruction and texture mapping

Figure 1: Outline of the algorithm. The user interactively labels a few parallel and perpendicular edges in the images. These are then localised precisely and used to compute the projection matrices for the viewpoints. Triangulation and texture mapping is used to produce a 3D VRML model.

4. The last step consists in using these projection matrices to find more correspondences between the images and then to compute 3D textured triangles that represent a model of the scene.

A working application called *PhotoBuilder* has been designed and implemented to allow the user to interactively build a 3D VRML model from a pair of uncalibrated images from arbitrary viewpoints in less than 15 minutes.

3 Geometric Framework

3.1 Review and notation

For a pin-hole camera, perspective projection from Euclidean 3-space to an image can be conveniently represented in homogeneous coordinates by a 3×4 camera projection matrix, $\mathbf{P}$:

$$\lambda_i \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \quad (1)$$

The projection matrix has 11 degrees of freedom and can be decomposed into the orientation and position of the camera relative to a the world co-ordinate system (a 3×3 rotation matrix $\mathbf{R}$ and a 3×1 translation vector $\mathbf{T}$):

$$\mathbf{P} = \mathbf{K} \begin{bmatrix} \mathbf{R} & \mathbf{T} \end{bmatrix} \quad (2)$$

and a 3×3 camera calibration matrix, $\mathbf{K}$, corresponding to the following image plane transformation:

$$\mathbf{K} = \begin{bmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (3)$$

where α_u, α_v are scale factors; s is a skew parameter; and u_0, v_0 are the pixel coordinates of the principal point (the intersection of the optical axis with the image plane).

3.2 Approach

In our approach the vanishing points corresponding to three mutually orthogonal directions can be used to determine for each viewpoint:

1. the camera calibration matrix, $\mathbf{K}$ under the assumption of zero skew and known aspect ratio.

2. the rotation matrix $\mathbf{R}$.

3. the direction of translation, $\mathbf{T}$.

We show that the 8 degrees of freedom of the projection matrix for this special case can be determined from three vanishing points corresponding to the projections of 3 points at infinity and a reference point. The projection matrix can thus be recovered from the projection of at least one arbitrary cuboid. Applying the algorithm to two views allows the Euclidean reconstruction of all visible points up to an arbitrary scale.

Using vanishing points

From (1) and considering the points at infinity corresponding to the three orthogonal directions we can derive simple constraints on the elements of the projection matrix:

$$\begin{bmatrix} \lambda_1 u_1 & \lambda_2 u_2 & \lambda_3 u_3 \\ \lambda_1 v_1 & \lambda_2 v_2 & \lambda_3 v_3 \\ \lambda_1 & \lambda_2 & \lambda_3 \end{bmatrix} = \mathbf{P} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad (4)$$

where λ_i are initially unknown scaling factors. This equation can be rearranged and expressed in terms of the camera calibration matrix $\mathbf{K}$ and camera orientation (rotation matrix), $\mathbf{R}$:

$$\begin{bmatrix} u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} = \mathbf{KR} \quad (5)$$

Camera calibration and recovery of orientation

By exploiting the properties of the rotation matrix, $\mathbf{R}$, we can rerrange (5) to recover constraints on the intrinsic parameters of the camera and the unknown scaling parameters, λ_i. In particular:

$$\begin{bmatrix} u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \lambda_1^2 & 0 & 0 \\ 0 & \lambda_2^2 & 0 \\ 0 & 0 & \lambda_3^2 \end{bmatrix} \begin{bmatrix} u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \\ 1 & 1 & 1 \end{bmatrix}^T = \mathbf{K}\mathbf{K}^T \quad (6)$$

Under the assumption of known aspect ratio and zero skew, (6) can be rewritten as 6 linear equations (from six elements of the symmetric matrix) and can be solved to recover 3 intrinsic parameters and the unknown scale factors, λ_i^2.

Geometric interpretation

Note that these equations can also be rearranged to derive the curious geometric interpretation (see figure

2) that the orthocentre of the image triangle formed by the three vanishing points is the principal point. This was first shown by Caprile and Torre [2]. In addition to this property, we can show that the scale factors λ_i have the geometric interpretation shown in figure 2.

Recovery of projection matrix

The solution of (6) and substitution into (5) leads to the recovery of the 3×3 sub-matrix of the projection matrix, **KR**, which can then be easily decomposed into the rotation matrix **R**.

The fourth column of the projection matrix depends on the position of the world co-ordinate system relative to the camera co-ordinate system. An arbitrary reference point can be chosen as the origin. Its image co-ordinates fix the translation, **T**, up to an arbitrary scale factor, λ_4:

$$\lambda_4 \begin{bmatrix} u_4 \\ v_4 \\ 1 \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \mathbf{KT} \quad (7)$$

In a single viewpoint and with no metric information this scale is indeterminate and can be arbitrarily set, e.g. $\lambda_4 = 1$. For additional views the image correspondences of a fifth point is required to fix this scale factor in the additional views. This is equivalent to fixing the epipoles from the translational component of image motion under known rotation – two point correspondences are required to recover the direction of translation.

4 Finding Vanishing Points

The key step in the algorithm to recover the projection matrices requires finding the vanishing points of parallel lines with known orientations. Image segments can be interactively defined by a user and care must be taken to find the corresponding vanishing points.

A vanishing point corresponds to the projection of the intersection of parallel lines at infinity. A number of approaches have been proposed to localise precisely this intersection, from the simple calculation of a weighted mean of pairwise intersections [2] to more elaborate approaches involving noise assumption and non-linear criteria [4, 10]. However, all these approaches are based on criteria which take into account image distances. Though trying to minimise the distance from the vanishing point to the image lines is geometrically correct, it appears to be numerically unstable in the presence of noise. In contrast, our approach is based on a linear criterion which optimises a three dimensional direction, the dual of a vanishing point in 3D space.

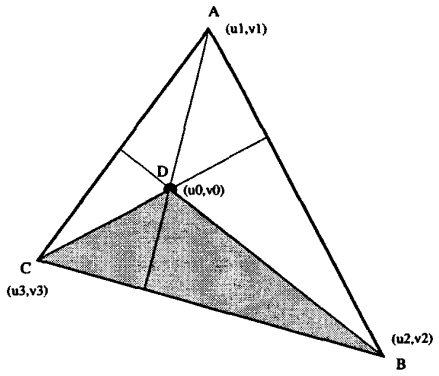


Figure 2: Geometric interpretation. The orthocentre, **D**, of the triangle formed by the three vanishing points (**A**, **B** and **C**) can be shown [2] to be the principal point, (u_0, v_0). The scale-factor, λ_1^2, associated with the vanishing point $\mathbf{A} = (u_1, v_1)$, can similarly be shown to be the area of the triangle **BCD** normalised by the total area of the triangle **ABC**. A similar result applies to the other two factors and vanishing points, **B** and **C**. The scale factor α_u can also be inferred from the triangle.

Let $l_1, .., l_n$ be a set of image lines corresponding to parallel space lines and let ν be the vanishing point defined by these lines. Then if we suppose that l_i and ν are expressed in homogeneous form in the image coordinate system (u, v, w), we have the following relation:

$$[l_1, .., l_n]^T \nu = 0. \quad (8)$$

and the null vector of the $3 \times n$ matrix $[l_1, .., l_n]$ gives a solution $\hat{\nu}$ for the vanishing point position. However, the minimised criterion does not necessarily have a physical meaning. By choosing an appropriate normalisation for $(l_1, .., l_n)$, the solution $\hat{\nu}$ can minimise the sum of the Euclidean distances from $\hat{\nu}$ to $[l_1, .., l_n]$. As said before, this solution will not be completely satisfactory in the presence of noise due to poor conditioning of the matrix $[l_1, .., l_n]$.

Now suppose that the intrinsic parameters of the camera are known. Then image lines and points can be expressed using *normalised image coordinates*, that is to say in the coordinate system (x, y) associated with the camera retinal plane (see figure 3). Let

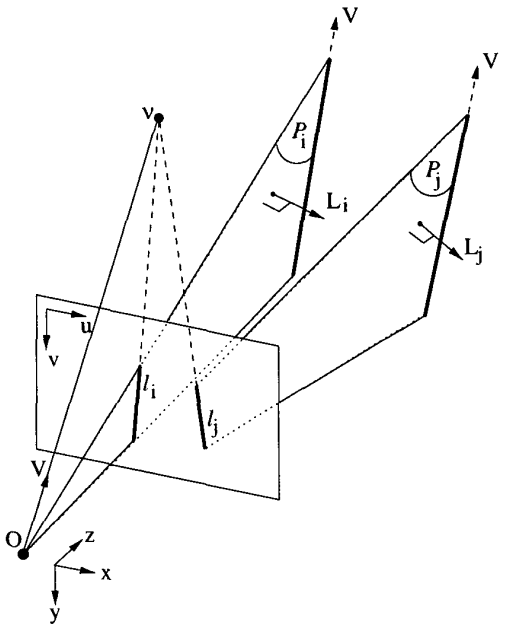


Figure 3: Vanishing point determination. The idea is to find the viewing direction V of the vanishing point ν. This direction should belong to all the planes $\mathcal{P}_i$.

$(L_1, .., L_n)$ and V be the homogeneous representations of $(l_1, .., l_n)$, ν in the retinal plane. Since V belongs to all the image lines, we still have the relation:

$$[L_1, .., L_n]^T V = 0. \quad (9)$$

However, this relation has also an interpretation in the three dimensional coordinate system (x, y, z) associated with the camera (see figure 3). Indeed, L_i is a vector orthogonal to the plane $\mathcal{P}_i$ spanned by the camera projection centre O and the image line l_i, and V is a three dimensional vector. This gives a physical meaning to the null vector $\hat{V}$ of $[L_1, .., L_n]^T$ which is the space direction closest to all planes $(\mathcal{P}_1, .., \mathcal{P}_n)$.

Finally, we compute the space direction V as the null vector of the matrix $[L_1, .., L_n]^T$, where $|L_i| = 1$. This is done using a singular value decomposition of $[L_1, .., L_n]^T$ [17]. Experiments show that estimating a space direction is more robust to noise perturbations than estimating an image point. We use this method in the calibration routine described in the next section.

In the case where the camera's intrinsic parameters are not known, an intermediate solution consists of using (8) with prenormalized image point coordinates [9] in order to improve the condition number of the matrix $[l_1, .., l_n]$.

5 Projection Matrices

Having found the vanishing points we can now recover the intrinsic parameters (and λ_i scale factors) by solving equation (6) using singular value decomposition. In practice the principal point is very sensitive to error and is assumed known (i.e. at the centre of the image). The other parameters are extremely reliable and the vanishing points and directions are then used to obtain the orientation and position of the cameras. Note that the vanishing points are used directly to estimate the rotation (since these are independent of camera translation). The translation is then computed from pairs of corresponding points using the epipolar constraint.

These motions combined with the intrinsic parameters allow us to compute projection matrices for the different views involved. From these projection matrices, we can determine the epipolar geometry to help find more point correspondences and then the 3D point positions. These points are then used in association with an image point triangulation to obtain 3D structure. This structure is rendered afterwards using a standard texture mapping procedure and the final model is stored in standard VRML format.

Experiments on synthetic and real data have been conducted. Figure 4–6 show some preliminary results for real images of the Downing College library in the University of Cambridge. These images were obtained by an Olympus digital camera. The calibration for these images was performed using 3 pairs of parallel edges shown in Figure 4 to determine the vanishing points and hence calibrate the two cameras. The geometry was then computed using the vanishing points the 3D co-ordinates were then recovered. Figure 5 shows an example of the VRML mdoel with texture. This model consists of of forty textured triangles and was produced in less than 15 minutes.

6 Conclusions

The techniques presented have been successfully used to interactively build models of architectural scenes from pairs of uncalibrated photographs. The simple but powerful constraints of parallelism and orthogonality in architectural scenes can be used to recover very precise projection matrices with only a few point and line correspondences.

We plan to use these initial estimates of the projection matrices (and hence the epipolar geometry) to automatically match additional features and then to optimise the parameters of the camera motion and 3D structure by standard ray-bundle adjustment. The *PhotoBuilder* system has been use to build 3D models of numerous buildings and is presently being used to

buid a VRML model of the University of Cambridge. Preliminary results are extremely impressive.

References

[1] P. Beardsley, P. Torr and A. Zisserman. 3D Model Acquisition from Extended Image Sequences. In *Proc. 4th European Conf. on Computer Vision*, Cambridge (April 1996); LNCS 1065, volume II, pages 683–695, Springer–Verlag, 1996.

[2] B. Caprile and V. Torre. Using Vanishing Points for Camera Calibration. *International Journal of Computer Vision*, 4: 127–139, 1990.

[3] R. Cipolla and E. Boyer 3D model acquisition from uncalibrated images. In *Proc. IAPR Workshop on Machine Vision Applications*, Chiba, Japan, pages 559–568, (November) 1998.

[4] R.T. Collins and R.S. Weiss. Vanishing point calculation as a statistical inference on the unit sphere. In Proc. Third Int. Conference on Computer Vision, pages 400-403, Osaka, (December) 1990.

[5] P.E. Debevec, C.J. Taylor, and J. Malik. Modeling and Rendering Architecture from Photographs: A Hybrid Geometry- and Image-Based Approach. In *ACM Computer Graphics (Proceedings SIGGRAPH)*, pages 11–20, 1996.

[6] O. Faugeras and S. Laveau. Representing Three-Dimensional Data as a Collection of Images and Fundamental Matrices for Image Synthesis. In *Proceedings of the 12th International Conference on Pattern Recognition, Jerusalem (Israel)*, pages 689–691, 1994.

[7] O. Faugeras, S. Laveau, L. Robert, G. Csurka and C. Zeller. 3D reconstruction of urban scenes from sequences of images. Computer Vision and Image Understanding, 69(3):292–309, 1998.

[8] S.J. Gortler, R. Grzeszczuk, R. Szeliski, and M.F. Cohen. The Lumigraph. In *ACM Computer Graphics (Proceedings SIGGRAPH)*, pages 43–54, 1996.

[9] R.I. Hartley. Lines and points in three views and the trifocal tensor. *International Journal of Computer Vision*, 22(2): 125–140, 1996.

[10] K. Kanatani. *Statistical Optimization for Geometric Computation: Theory and Practice.* Lecture Note, Gunma University (Japan) 1995.

[11] M. Levoy and P. Hanrahan. Light Field Rendering. In *ACM Computer Graphics (Proceedings SIGGRAPH)*, pages 31–42, 1996.

[12] H.C. Longuet-Higgins. A computer program for reconstructing a scene from two projections. *Nature*, 293: 133–135, September 1981.

[13] Q.T. Luong and O. Faugeras. The Fundamental Matrix: Theory, Algorithms and Stability Analysis. *International Journal of Computer Vision*, 17(1): 43–75, 1996.

[14] L. McMillan and G. Bishop. Plenoptic modeling: an image-based rendering system. In *ACM Computer Graphics (Proceedings SIGGRAPH)*, pages 39–46, 1995.

[15] P.J. Narayanan, P.W. Rander, and T. Kanade. Constructing Virtual Worlds Using Dense Stereo. In *Proc. of Sixth IEEE Intl. Conf. on Computer Vision, Bombay (India)*, pages 3–10, January 1998.

[16] M. Pollefeys, R. Koch and L. Van Gool. Self-calibration and metric reconstruction inspite of varying and unknown internal camera parameters. In *Proc. of Sixth IEEE Intl. Conf. on Computer Vision, Bombay (India)*, pages 90–95, January 1998.

[17] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes in C, The Art of Scientific Computing, Second Edition.* Cambridge University Press, 1992.

[18] S.M. Seitz and C.R. Dyer. Toward Image-Based Scene Representation Using View Morphing. In *Proc. of Intl. Conf. on Pattern Recognition, Vienna (Austria)*, January 1996.

[19] H-Y. Shum, M. Han and R. Szeliski. Interactive construction of 3D models from Panoramic Mosaics. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 427–433, Santa Barbara, (June) 1998.

[20] A. Shashua. Trilinearity in Visual Recognition by Alignment. In *Proceedings of Third European Conference on Computer Vision, Stockholm, (Sweden)*, pages 479–484, January 1994.

[21] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9(2):137–154,1990.

[22] T. Werner, R.D. Hersh, and V. Hlavac. Rendering Real-World Objects Using View Interpolation. In *Proceedings of 5th International Conference on Computer Vision, Boston (USA)*, pages 957–962, January 1995.

[23] A. Zisserman, D. Liebowitz and M. Armstrong. Resolving ambiguities in auto-calibration. In *Phil. Trans. Royal Society*, A356:1193–1211, 1998.

Figure 4: Original pair of photographs taken with different camera zoom setting. Six edges in each image are used to obtain the camera intrinsic parameters and the orientation and position of the viewpoints.

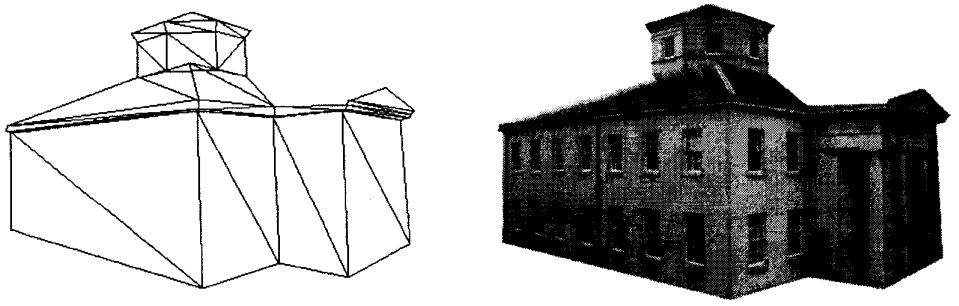


Figure 5: 3D wireframe model with texture from original images. Correspondences are given manually and processed to refine the motion between the viewpoints. Triangles that are visible in both views can be renconstructed.

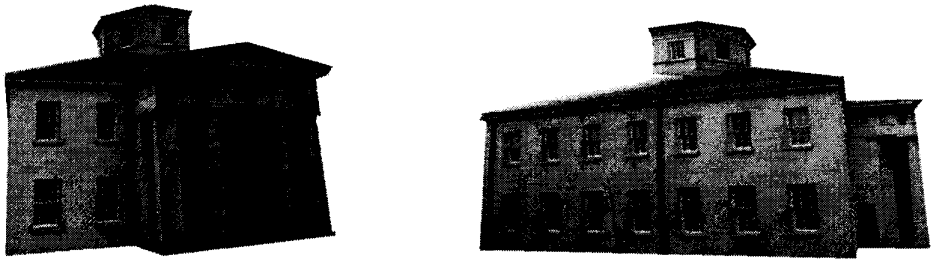


Figure 6: The 3D model can be output in VRML format and viewed from new viewpoints.