A simple method for interactive 3D reconstruction and camera calibration from a single view

Akash M Kushal   Vikas Bansal   Subhashis Banerjee

Department of Computer Science and Engineering
Indian Institute of Technology, Delhi
New Delhi 110016
{akash,vikas,suban}@cse.iitd.ernet.in

Abstract

We present a simple and intuitive method for interactive 3D reconstruction and camera calibration from a single image of a structured scene. The method is based on manual registration of two world planes. We present experimental results on some test images.

1. Introduction

Recently the problem of interactive 3D reconstruction from a single view of a scene has attracted considerable attention [2, 5, 3, 6, 8, 7, 1, 9]. Such a reconstruction, without any knowledge of camera parameters, may find wide use in a variety of geometric modeling applications. 3D reconstruction from a single image must necessarily be through an interactive process, where the user provides information and constraints about the scene structure. Such information may be in terms of vanishing points or lines [5, 1, 6, 7], coplanarity [8], spatial inter-relationship of features [2, 3] and camera constraints [9].

In this paper we assume the customary pin-hole camera model [4] and present a method for 3D reconstruction from a single image based on the registration of two planes. The method is simple and intuitive and is applicable for structured scenes. The method is based on computation of homographies [4] from two world planes to the image, which can be accomplished more easily and reliably than computing vanishing points and lines. The user needs to identify the two planes by clicking a rectangle on each.

We show that the interactive registration of two planes facilitate 3D reconstruction of structured scenes using simple similar triangles and that it is also possible to recover the camera position and the internal parameters of the camera. We deal with cases when (i) the two planes are orthogonal and the X axis of the two planes are parallel to each other, (ii) the two planes are orthogonal and but the coordinate system of one plane is rotated with respect to the other, and (iii) the two planes are not orthogonal.

The organization of the rest of the paper is as follows. In Section 2 we present the basic method of camera calibration and 3D reconstruction from two planes. In Section 3 we work out some extensions of the basic method to make it practically applicable. In Section 4 we present some results and finally, in Section 5, we conclude the paper.

2. The basic case

Consider first the situation of Figure 1 where the two world planes are orthogonal and their common axis of intersection is visible in the image. Let the world coordinate system

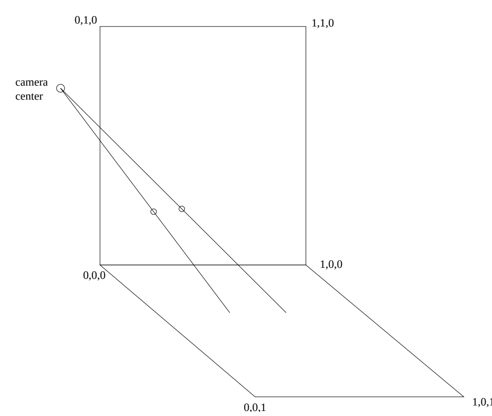


Figure 1: The basic case

of the X-Y (vertical) and X-Z (horizontal) planes be as indicated in the figure. It is not necessary that three axes are of same scale. Let $\mathbf{H}$ and $\mathbf{G}$ be the homographies from the world X-Z and X-Y planes to the image respectively. These homographies may be computed by hand-picking the six defining points in the image in a standard way [4].

In what follows we establish two crucial properties of $\mathbf{H}$ and $\mathbf{G}$:

1. *The third column of both the homographies are equal (up to scale):* The last column of both the homo-

graphies is the image of the $(0,0)$ point for both the planes. Now, since the $(0,0,0)$ point in the world has coordinates $(0,0)$ in both the X-Y and X-Z planes, both the homographies will map it to the image of the world coordinate system's origin. Hence, both the homographies have their last column equal (up to scale). Since homographies are themselves defined only up to a scale, the last columns can be taken to be the same. This will fix the relative scale between the two homographies.

2. *The first column of both the homographies are equal (again up to scale):* The first columns of the homographies are the image of the point at infinity in the X direction in the world. Also, if we fix the relative scale of the homographies (by making the last columns equal in the previous part) then since the unit length in the X direction for both the homographies is the same, the image of the (1,0) point for both the homographies will be the same (i.e. the image of the (1,0,1) point in the world). Hence the first columns would also have to be equal (including scale being equal).

So to obtain consistent homographies $\mathbf{H}$ and $\mathbf{G}$ we can solve for both of the homographies together and decrease the number of independent variables from 16 independent variables (8 for each of the homographies) to just 11 independent variables (8 for any one and only 3 for the other since only the 2^{nd} column of the second is independent). These 11 degrees of freedom are exactly those required to define the full projection matrix (3×4) for the camera [4].

2.1. Camera calibration

The 3×4 projection matrix of the camera can be read out from the columns of the homographies $\mathbf{H} = [\mathbf{H}(1), \mathbf{H}(2), \mathbf{H}(3)]$ and $\mathbf{G} = [\mathbf{G}(1), \mathbf{G}(2), \mathbf{G}(3)]$. The projection matrix $\mathbf{P}$ for the camera is

$$\mathbf{P} = \begin{bmatrix} \mathbf{H}(1) & \mathbf{G}(2) & \mathbf{H}(2) & \mathbf{H}(3) \end{bmatrix}$$

This is because the last column of the projection matrix is the projection of the world origin and the first three columns of $\mathbf{P}$ are the vanishing points in the respective directions. The vanishing points in the X direction are same for both the homographies and the unit lengths are also equal. So the relative scale of the 4 columns is also correct in the world coordinate sytem that we had specified.

But, if we want to calibrate the camera (i.e. determine the matrix of camera internals, $\mathbf{K}$) we require to know the relative scales of each of the axis of the world coordinate system that we had initially specified. To get the projection matrix in a coordinate system with unit lengths of the axis equal we can just multiply the initial $\mathbf{P}$ matrix with a diagonal matrix as follows

$$\mathbf{P_{new}} = \mathbf{P_{old}} * \begin{pmatrix} \alpha & 0 & 0 & 0 \\ 0 & \beta & 0 & 0 \\ 0 & 0 & \gamma & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1}$$

where $1/\alpha$ is the unit length in the X-direction, $1/\beta$ is the unit length in the Y-direction and $1/\gamma$ is the unit length in the Z direction.

The projection matrix $\mathbf{P}$ obtained above can be written as $\mathbf{K}[\mathbf{R}|\mathbf{t}]$ where the 3×3 matrix $\mathbf{K}$ is the matrix of camera internals, $\mathbf{R}$ is the 3×3 rotation matrix from the world coordinate system to the camera coordinate system and $-\mathbf{R}^t\mathbf{t}$ are the coordinates of the camera center [4]. Thus, the first 3×3 of $\mathbf{P}$ is $\mathbf{KR}$ (call it $\tilde{\mathbf{P}}$). Clearly $\tilde{\mathbf{P}} * \tilde{\mathbf{P}}^t = \mathbf{KK}^t$.

The camera internal matrix has a general form [4]

$$\mathbf{K} = \begin{pmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{pmatrix} \tag{2}$$

So $\mathbf{KK}^t$ (call it $\mathbf{X}$) will have the form

$$\mathbf{X} = \mathbf{KK^t} = \begin{pmatrix} \alpha_u^2 + u_0^2 + s^2 & \alpha_v s + u_0 v_0 & u_0 \\ \alpha_v s + u_0 v_0 & \alpha_v^2 + v_0^2 & v_0 \\ u_0 & v_0 & 1 \end{pmatrix} \tag{3}$$

from which we can directly obtain the values of the camera internals. We first normalize $\mathbf{X}$ and make X_{33} equal to 1. Now,

$$u_0 = X_{31}\ ,\ v_0 = X_{32}\ ,\alpha_v = -\sqrt{X_{22} - v_0^2}$$
$$s = (X_{21} - u_0 v_0)/\alpha_v\ ,\ \alpha_u = \sqrt{X_{11} - s^2 - u_0^2}$$

Once $\mathbf{K}$ has been obtained, $\mathbf{R}$ and $\mathbf{t}$ can also be simply obtained. $\mathbf{R} = \mathbf{K^{-1}} * \tilde{\mathbf{P}}$ where $\tilde{\mathbf{P}}$ is $\mathbf{KR}$ (the first 3×3 of $\mathbf{P}$). $\mathbf{t} = \mathbf{K}^{-1} * \mathbf{Kt}$ where $\mathbf{Kt}$ is the last column of $\mathbf{P}$.

2.2. Localization of the camera center

Once $\mathbf{R}$ and $\mathbf{t}$ are computed as above, the camera center can be computed as $-\mathbf{R}^t\mathbf{t}$.

The camera center can also be directly computed from the homographies $\mathbf{H}$ and $\mathbf{G}$ defined above in the following way.

Clearly, $\mathbf{T} = \mathbf{G}^{-1}\mathbf{H}$ is the homography which maps from the X-Z plane to the X-Y plane as shown Figure 1. Let the coordinates of the camera center in the world frame be (C_x, C_y, C_z). It can easily be verified geometrically that the homography that maps points on the X-Z plane to the X-Y plane is

$$\mathbf{T} = \mathbf{G}^{-1}\mathbf{H} = \lambda \begin{pmatrix} -C_z & C_x & 0 \\ 0 & C_y & 0 \\ 0 & 1 & -C_z \end{pmatrix} \tag{4}$$

So we can now get the coordinates of the camera center. First we normalize the homography $\mathbf{T}$ by making $T_{32} = 1$. Then the camera center is

$$C = (C_x, C_y, C_z) = (T_{12}, T_{22}, -T_{11})$$

This process can be geometrically understood as just taking two pairs of corresponding points on the 2 planes and shooting lines through each pair of points. The intersection these 2 lines is the camera center. The camera center's coordinates (C_x, C_y, C_z) will be obtained in the world coordinate system. Here, the scales of X, Y and Z need not be identical.

2.3. Computing the 3D coordinates

Once the camera center coordinates have been determined, the coordinates of 3D points in the scene can be computed using simple geometry. The user clicks a point (call it the *head*) and its projection on the X-Z plane (call it the *foot*) in the image, for example the head and the foot of a person (see Figure 2). Now using the homography $\mathbf{H^{-1}}$ (from the image plane to the X-Z horizontal plane) we can project the head and the foot of the point on the horizontal (X-Z plane). Now we can use simple similar triangles to calculate the height (Y coordinate) of the point.

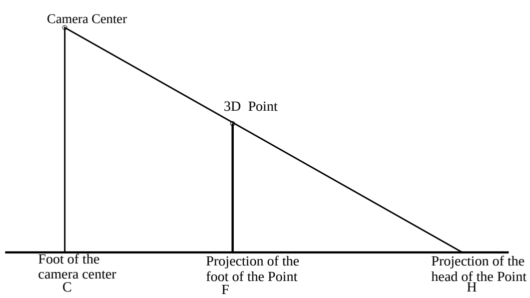


Figure 2: Determining the Height

The height can be computed as h = Camera height * $\frac{FH}{CH}$

Note that the height can also be obtained relative to the height of another object (with its head and foot given) by eliminating the camera height from the two similar triangle equations. The X and Z coordinates come out directly using the homography $\mathbf{H}^{-1}$. Thus we can get all the 3 affine coordinates of points in the scene coordinate system by simple geometry.

Of course, the above procedure can also be applied relative to the X-Y plane to compute the width of objects as well.

Also, in case the foot of the object (on the ground plane) cannot be seen clearly in the image (eg. when the object is placed on a horizontal table), then one can first find out the height h of the table by clicking the head and foot of any point on the table and then get the homography from the image to the horizontal (table) plane by setting $Y = h$ in the matrix $\mathbf{P}$. Then again we can use a similar approach as above just taking the projection of the head on the table plane and clicking the foot on the table plane. Then we can use similar triangles as before to calculate the height. In this case the height of the camera center will obviously be $C_y - h$ (taken from the table).

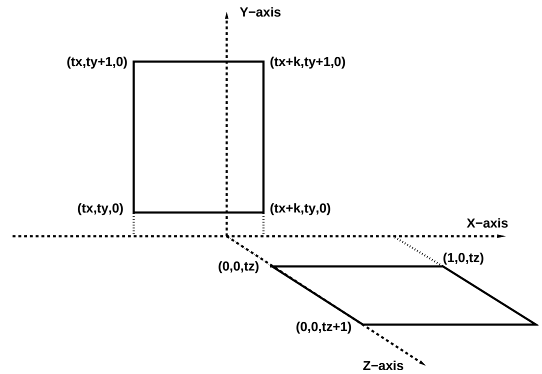


Figure 3: Planes orthogonal, but X axis not visible

3. Possible extensions

In what follows we consider various possible extensions of the method.

3.1. The two planes are orthogonal but the common X axis is not visible

Consider the situation of Figure 3. Let us assume that the world X axis is not visible but we have four Euclidean points on each plane to define homographies $\mathbf{H}'$ and $\mathbf{G}'$ between the X-Z and X-Y planes and the image respectively. Let us assume the X dimension of the ground plane rectangle to be the unit length of the world. Let $1/\gamma$ be the Z dimension of the ground plane rectangle and $1/\alpha$ and $1/\beta$ be the X and Y dimensions of the vertical plane rectangle. Also let t_x, t_y and t_z be the translations of the rectangles as shown in Figure 3.

Clearly, the homographies $\mathbf{H}'$ and $\mathbf{G}'$ are related to the homographies $\mathbf{H}$ and $\mathbf{G}$ defined before as follows:

$$\begin{aligned} \mathbf{H} = \mathbf{H}' * \begin{pmatrix} 1 & 0 & 0 \\ 0 & \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} * \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -t_z \\ 0 & 0 & 1 \end{pmatrix} \\ = \mathbf{H}' * \begin{pmatrix} 1 & 0 & 0 \\ 0 & \gamma & -\gamma t_z \\ 0 & 0 & 1 \end{pmatrix} \\ \mathbf{G} = \mathbf{G}' * \begin{pmatrix} \alpha & 0 & -\alpha t_x \\ 0 & \beta & -\beta t_y \\ 0 & 0 & 1 \end{pmatrix} \end{aligned} \tag{5}$$

Now substituting from Equation 5 in to Equation 4 and rewriting we have,

$$\mathbf{H}' \begin{pmatrix} 1 & 0 & 0 \\ 0 & \gamma & -\gamma t_z \\ 0 & 0 & 1 \end{pmatrix} = \lambda * \mathbf{G}' * \begin{pmatrix} \alpha & 0 & -\alpha t_x \\ 0 & \beta & -\beta t_y \\ 0 & 0 & 1 \end{pmatrix} * \begin{pmatrix} -C_z & C_x & 0 \\ 0 & C_y & 0 \\ 0 & 1 & -C_z \end{pmatrix} \tag{6}$$

Note that there are no hidden scales as we have written them all explicitly. Let

$$\mathbf{H}' = \begin{pmatrix} \mathbf{h_1} & \mathbf{h_2} & \mathbf{h_3} \end{pmatrix}, \mathbf{G}' = \begin{pmatrix} \mathbf{g_1} & \mathbf{g_2} & \mathbf{g_3} \end{pmatrix}$$

Expanding Equation 6 we obtain

$$\begin{pmatrix} \mathbf{h_1} & \gamma\mathbf{h_2} & -\mathbf{h_2}\gamma t_z + \mathbf{h_3} \end{pmatrix} = \lambda \left(\mathbf{X_1} \;\; \mathbf{X_2} \;\; \mathbf{X_3}\right) \tag{7}$$

where,

$$\mathbf{X_1} = \begin{pmatrix} -\mathbf{g_1}\alpha C_z \end{pmatrix}$$
$$\mathbf{X_2} = \begin{pmatrix} \mathbf{g_1}\alpha(C_x - t_x) + \mathbf{g_2}\beta(C_y - t_y) + \mathbf{g_3} \end{pmatrix}$$
$$\mathbf{X_3} = \begin{pmatrix} C_z(\mathbf{g_1}\alpha t_x + \mathbf{g_2}\beta t_y - \mathbf{g_3}) \end{pmatrix}$$

Note that since the first columns of both $\mathbf{H}'$ and $\mathbf{G}'$ is the vanishing point in the X direction so they will be equal up to scale. This has to be enforced during the computation of $\mathbf{H}'$ and $\mathbf{G}'$. Now,

1. If one of αt_x, βt_y or γt_z is known then the others can be found out easily by comparing the entries of the third column of Equation 7. We can also obtain λC_z from these 3 equations. Then by comparing the first columns we can compute $-\lambda\alpha C_z$. Now, from these two values we can also figure out the value of α.

2. Instead, if α is known then the ratio λC_z can be obtained from the first column. Plugging it in to the third column we have three equations in three unknowns (t_x, βt_y and γt_z) and we can solve in general. It is indeed not surprising that the knowledge of α should enable us to figure out translations/depths. The scaling in the image of two parallel world vectors of the same length (or known relative length) seperated in depth indeed gives us information about the perspective effect.

Once we have obtained λC_z, αt_x, βt_y and γt_z we can compute the other unknowns as follows. Multiplying column 2 of both sides with C_z and plugging in the values of αt_x, βt_y and γt_z we can solve for αC_x, βC_y and γC_z. Note that β and γ cannot be determined separately (unless given). They can be set to 1 without loss of generality and t_y, t_z, C_y and C_z can computed up to scale. That is the scales between the unit lengths of the X, Y, and Z axis is not known. If β or γ is known then we know the scale between the unit lengths of the corresponding axis and so we get the lengths up to a common scale.

Once these parameters are computed we can use Equation 5 to compute $\mathbf{H}$ and $\mathbf{G}$. Since we have already computed the camera center we can proceed as in Section 2 to compute the camera internals and 3D coordinates.

3.2. Rotation of the coordinate system of the horizontal plane

In addition to the situation in the previous section, consider a rotation of the coordinate system of the horizontal plane (about the Y axis). To account for this rotation, the first equation in Equation 5 will have to be changed to

$$\mathbf{H} = \mathbf{H}' * \begin{pmatrix} 1 & 0 & 0 \\ 0 & \gamma & 1 \\ 0 & 0 & 1 \end{pmatrix} * \begin{pmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} * \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -t_z \\ 0 & 0 & 1 \end{pmatrix} \tag{8}$$

Now the left hand side of Equation 7 changes to

$$\begin{pmatrix} \mathbf{Y_1} & \mathbf{Y_2} & \mathbf{Y_3} \end{pmatrix}$$

where,

$$\mathbf{Y_1} = \begin{pmatrix} cos\theta\mathbf{h_1} - sin\theta\gamma\mathbf{h_2} \end{pmatrix}$$
$$\mathbf{Y_2} = \begin{pmatrix} sin\theta\mathbf{h_1} + cos\theta\gamma\mathbf{h_2} \end{pmatrix}$$
$$\mathbf{Y_3} = \begin{pmatrix} -\mathbf{h_1}\gamma t_z sin\theta - \mathbf{h_2}\gamma t_z cos\theta + \mathbf{h_3} \end{pmatrix}$$

We know that the first column of the LHS matrix is a linear combination of the first and second columns of the original homography since it represents a point at infinity in the XZ plane and so must lie on the line passing through the points at infinity defined by the first and second columns of the original homography.

If γ and α are known then by comparing the first columns we can get the value of the $\gamma tan\theta$ and so get θ. From the first column itself we will also get $\lambda\alpha C_z$. Now, if we know α then we can get λC_z which can be used when we compare the third columns to get 3 equations and solve for t_x, βt_y and t_z. Now we can compare the second column and use the previous information to get C_x, βC_y and C_z.

If γ and one of αt_x, βt_y or t_z are known then we can first compare the first columns and get θ as in the previous case. Then using the third column we can get the others of αt_x, βt_y and t_z and also get λC_z. Now, from this and $\lambda\alpha C_z$ obtained from the first column we can get α also. Now again as in the previous case we can compare the second columns to get C_x, βC_y and C_z.

If θ is known then γ can also be calculated from the first column. Now if we also know α then we can also get λC_z from the first column itself. Using these we can get the translations by comparing the third column and then proceed as before.

If all the parameters can be computed we can obtain $\mathbf{H}$ and $\mathbf{G}$ and proceed as in Section 2.

3.3. Rotation of the coordinate system of the vertical plane

This case is identical to that of the previous section. Either of the vertical or the horizontal plane coordinate system can be taken as the reference.

3.4. The two planes not orthogonal

This situation is depicted in Figure 4 which shows a cross-section. Instead of the XY plane, measurements are now made with respect to the XY' plane which makes an angle ϕ with the vertical. $\mathbf{G}'$ and $\mathbf{G}$ are now homographies from the XY' plane to the image. Since $\mathbf{H}'$ and $\mathbf{H}$, and $\mathbf{G}'$ and $\mathbf{G}$

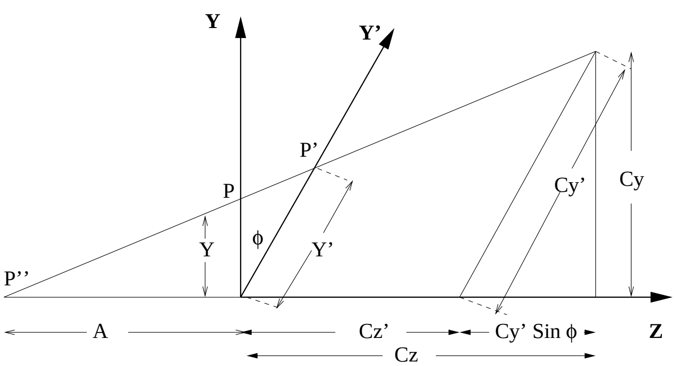


Figure 4: Planes not orthogonal

are related by homography relationships on their respective planes, all the analysis of the previous two sub-sections still apply and proceeding as before we can compute the camera center as (C_x, C'_y, C'_z) instead of (C_x, C_y, C_z). However, for 3D reconstruction, we need to now click the *head* and *foot* parallel to the XY' plane.

For 3D reconstruction when the *head* and *foot* are available in the standard way parallel to the XY plane, we need to know ϕ. If ϕ is known then we can compute (C_y, C_z) in the orthogonal coordinate system and subsequently compute Y using similar triangles as before.

S.No.	Measured height		Average	Error (in %)
	1st Edge	2nd Edge		
1	16.2094	16.066	16.1377	0.860625
2	15.7446	15.8174	15.781	1.36875
3	15.7977	15.8479	15.8228	1.1075
4	15.6799	16.0096	15.84475	0.9703125
5	16.1058	16.2843	16.19505	1.2190625
6	15.9151	15.8073	15.8612	0.8675
7	16.1429	16.0521	16.0975	0.609375
8	15.8025	16.0452	15.92385	0.4759375
9	16.1466	16.0452	16.0959	0.599375
10	15.8957	15.7984	15.84705	0.9559375

Table 1: Experimental measurements of the height of the small box

4 Results

We experimented with 10 images of a laboratory scene with the method of Section 3.1. Two of these are shown in Figure 5. The height of the smaller box was measured from 10

Figure 5: Laboratory images

images independently (with different orientations of camera and the box) and the results obtained are shown below. The heights of both the end points of the box were measured and the average value was taken. The errors in each measurement is also shown in the following table. The true height of the floppy disk box was 16.0 cm. The average error in the calculation of the height was 0.9%. We obtain similar accuracy in the computation of the height of the bigger Intel Motherboard box. The average error in the calculation of the height was 0.89%

In Figure 6 we show two views of a VRML model generated from the first image. See `http://www.cse.iitd.ernet.in/vglab/demo/single-view/2plane` for the VRML results. The $\mathbf{K}$ calculated for one of the laboratory images is shown below.

$$\mathbf{K} = \begin{pmatrix} 1029.8 & 0.1 & 324.4 \\ 0 & -1033.6 & 245.4 \\ 0 & 0 & 1 \end{pmatrix} \qquad (9)$$

Another experiment was conducted with the image of Figure 7 taken from [1], in which the height of a person was

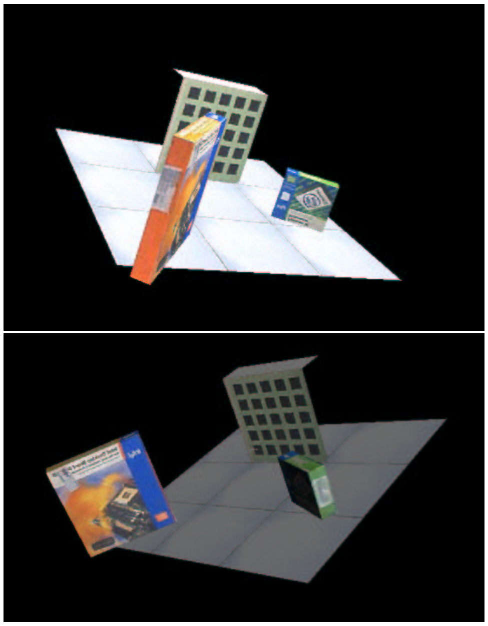
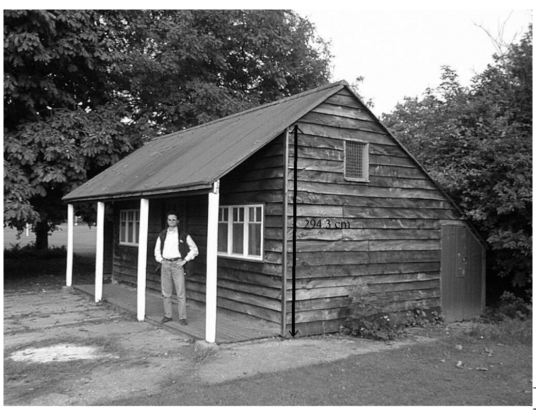


Figure 6: Novel views generated from a VRML model

measured from the image shown. Three trials were conducted and the results are displayed below. The actual height of the person was 180 cm and the reference height for the height calculation was taken as shown. The average error is 0.38%.



]

Figure 7: The hut image from [1]

5. Conclusion

We have presented a simple method for 3D reconstruction and camera calibration from a single image. The method is based on interactive registration of two world planes and is an alternative to the method proposed in [1]. It is to be noted that methods proposed in [1] regarding measurements on parallel planes can be directly imported in to our framework. Our experimental results demonstrate the applicability and robustness of the method.

References

[1] A. Criminisi, I. Reid, and A. Zisserman. Single view metrology. *IJCV*, 40(2):123–148, 2000.

[2] P. E. Debevec, C. J. Taylor, and J. Malik. Modeling and rendering architecture from photographs: a hybrid geometry and image based approach. In *In Proc. SIGGRAPH*, 1996.

[3] E. Grossmann, D. Ortin, and J. Santos-Victor. Single and multi-view reconstruction of structured scenes. In *In Proc. ACCV*, 2002.

[4] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[5] Y. Horry, K. .Anjyo, and K. Arai. Tour into the picture: using a spidery mesh interface to make animation from single image. In *In Proc. SIGGRAPH*, pages 225–232, 1997.

[6] D. Leibowitz, A. Criminisi, and A. Zisserman. Creating architectural models from images. In *In Proc. Eurographics*, pages 39–50, 1999.

[7] R.Cipolla and D. Robertson. 3d models of architectural scenes from uncalibrated images and vanishing points. In *In Proc. 10th IAPR*, pages 824–829, 1999.

[8] P. F. Strum and S. J. Maybank. A method for ineractive 3d reconstruction of piecewise planar objects from single views. In *In Proc. BMVC*, pages 265–274, 1999.

[9] M. Wilczkowiak, E. Boyer, and P. Strum. Camera calibration and 3d reconstruction from single images using parallelepipeds. In *In Proc. ICCV*, pages 142–148, 2001.