Camera calibration using reflections in planar mirrors and object reconstruction using volume carving method

N Martins[a] and **J Dias**[b]*
[a]Departamento de Informática e Sistemas, Instituto Superior de Engenharia de Coimbra, Instituto Politécnico de Coimbra, Portugal
[b]Instituto de Sistemas e Robótica, Departamento de Engenharia Electrotécnica, Universidade de Coimbra, Portugal

Abstract: A calibration and three-dimensional (3D) reconstruction method is presented based on images reflected from planar mirrors and acquired with one camera. The geometric model of the camera–mirror set and a method of calibrating it are described. The calibration technique computes the model parameters using linear equations, and it is proved that the calibration is possible with the knowledge of only six 3D points. The reconstruction method is based on a volumetric representation. The 3D reconstruction is based on a space carving algorithm and the calibration method described in the paper. The results of the calibration and reconstruction method show the efficiency of both techniques. This set-up enables a simple and inexpensive multi-ocular system to be built to recover the 3D structure of volumes.

Keywords: camera calibration, epipolar geometry, three-dimensional reconstruction, space carving

NOTATION

P	three-dimensional (3D) point, m
p	two-dimensional (2D) point, pixel
P_{M}	3D point on the mirror, m
p_{m}	projection of point P_{M}, pixel
${}^{W}\boldsymbol{P}$	vector with the homogeneous coordinates of a 3D point, expressed in the world referential $\{W\}$
${}^{I}\boldsymbol{p}$	vector with the homogeneous coordinates of a 2D point, expressed in the image referential $\{I\}$
$\mathbf{R}$	matrix
${}^{W}\boldsymbol{l}$	3D line expressed in the referential $\{W\}$

1 INTRODUCTION

The present paper presents a calibration and three-dimensional (3D) reconstruction method based on images reflected from planar mirrors and acquired with one camera. The geometric model of the camera–mirror set is described and also a method of calibrating it. The reconstruction technique is based on volumetric representation. The volumetric 3D reconstruction uses a space carving algorithm and the previous calibration results. This camera–mirror set-up allows a simple and inexpensive multi-ocular system to be built for 3D reconstruction of a scene.

The volumetric representation of a surface uses a volumetric element, known as a voxel, and it is based on the concept of volumetric occupation. Surfaces can be described by different geometric representations, and an accepted approach is the description of the surface geometrical structure by a set of previously defined surface primitives of different shapes. Although there are several techniques for solving the surface reconstruction problem from images, these solutions are numerically stable and restricted to well-defined problems. Most approaches to surface recovery employ as many assumptions as needed to ensure that reconstruction from stereo, shading or occluding

The MS was accepted for publication on 10 January 2004.
* *Corresponding author: Instituto de Sistemas e Robótica, Departamento de Engenharia Electrotécnica, Universidade de Coimbra, Portugal; e-mail: jorge@isr.uc.pt*

contours is well-posed when considered in isolation. Examples include the use of smoothness constraints for regularization, the use of small stereo baselines for minimizing the effect of occlusions, and the use of textureless surfaces for contour-based reconstruction.[1–3] Unfortunately, since no a priori surface information about the scene is available when these assumptions are made, it is impossible to predict their effect on the final reconstruction. An alternative for these solutions is the volumetric reconstruction based on space carving.

The space carving approach demonstrated that very complex structures could be recovered, based on this concept.[4,5] The technique gives a general solution for the surface reconstruction from images. To apply the technique, it is necessary to define, at the beginning, a volumetric space where the surface to be reconstructed fits. This volumetric space is then carved until the resulting surface is consistent with the acquired images. This means that every visible point within the volumetric space must be consistent with those images.

Generally, the images used for all these 3D reconstruction solutions must be calibrated, i.e. it is necessary to know the camera parameters associated with them. The results presented in the present paper are obtained using the camera calibration process described in section 2.

The extraction of 3D information from images is a computer vision problem which has been studied for a long time, and the solutions differ, owing to the image acquisition process to the computational algorithm used on the 3D reconstruction.

According to the image acquisition process, one can classify the image capture systems in two categories: the systems that use refracting elements (lenses), and the systems that use refracting and reflecting elements.[6]

The use of mirrors during the image acquisition phase has many advantages over the conventional use of systems using only lenses. One of these is the possibility of capturing views of the scene with only one camera.

As mirrors reduce the number of cameras, usually to one, the relation between the images is assisted, because the lens, sensor and digitizer parameters (such as lens distortions, focal length, pixel size, image centre, etc.) are identical. Furthermore, the estimation of some of those parameters becomes easier.[7] The minimization of the occlusion problem is another advantage of systems using reflection with a mirror.[8] The utilization of mirrors with different shapes has been studied by many researchers.[7–12] The shapes of the mirrors were planar or curved (ellipsoidal, hyperbolic, paraboloid, spherical and convex).[7,10] The curved mirrors have been used primarily to increment the field of view of the cameras.

The epipolar geometry of these systems was studied by Svoboda and others.[11] Sameer and Nayar[7] described an image stereo system based on reflective elements which preserve a single viewpoint. They demonstrated the use of a single camera and two or more planar mirrors to acquire stereo data in a single image.[13] Zhuang[14] constructed a model for computing the mirror centre offset, and took its sensitivity into account. Tamura[15] proposed a method for correcting errors of two-axis mirror scanner parameters using a coarse–fine parameter search. Since they have dealt mainly with systems with one or two mirrors, Kim and Cho[8] generalized the use of multiple mirrors. The results of those experiments demonstrate the viability of the use of mirrors.

Nayar[10] demonstrated the possibility of triangulating and computing depth from images obtained from reflections, generalizing the concept to include n reflecting elements of arbitrary shapes. Other possibilities for reconstructing a scene using reflected images of a scene have also been shown.[9,12]

The precision of mirror-based systems depends on the accurate estimation of the position and orientation of the mirrors and correct calibration of the camera parameters. The accurate estimation of these parameters is not easy, owing to dimensional uncertainties caused by manufacturing tolerances and distortions and aberrations of the lens. This results in uncertainties in: the parameters associated with the mirrors (positions and orientations); the camera's parameters (extrinsic and intrinsic); and the recovered 3D information. Those parameters can be measured directly, one by one, using precise instruments. However, during this operation, they could suffer from unexpected disturbance or intermittent adjustments (such as auto-focus or mirror relocation). Such a technique needs manual intervention, high mechanical precision and a large amount of computation. It is a slow and cumbersome method but it has the advantage that the acquired data and, subsequently, the result of its use are of high-resolution and precision data.

The present paper describes a linear calibration method for camera–mirror systems where the system parameters are computed automatically and without the use of high mechanical precision devices. It deals

with the linear extraction of some important parameters of a camera–mirror system, composed by one camera and planar mirrors. It shows how the use of a mirror simplifies the estimation of the epipolar geometry of the camera–mirror system and the computation of the extrinsic and intrinsic parameters of the camera.

The paper is organised as follows. Section 2 describes a closed-form solution to the camera calibration problem. Section 3 explains the shape and texture recovered, based on the space carving algorithm. Sections 2 and 3 also provide the experimental results. Finally, section 4 concludes the paper with the perspective of this work.

2 CAMERA CALIBRATION

2.1 Introduction

Camera calibration is a necessary step in the process of the extraction of 3D information from images. Different solutions have been extensively studied. In the photogrammetry area,[16,17] there are several examples but, more recently, the computer vision community has developed other techniques.[18–26]

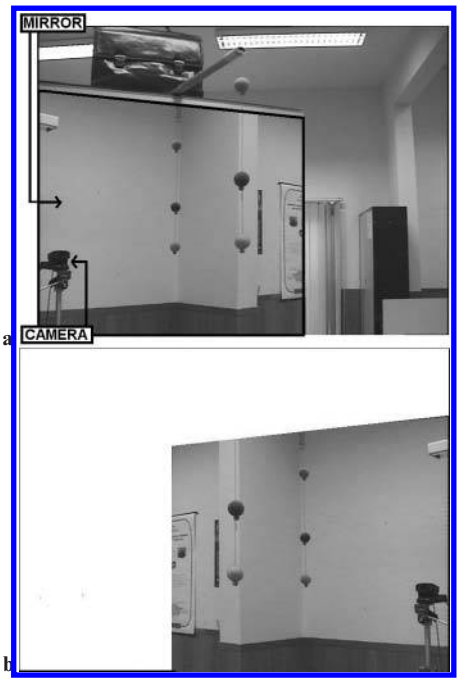
Zhang[27] stratifies the calibration techniques generically, according to the dimension of the calibration objects. The first technique (3D calibration) is performed observing a calibration object whose geometry, in 3D space, is known with very good precision.[28] The calibration object usually consists of two or three planes orthogonal to each other. Sometimes, a plane undergoing a precisely known translation is also used, which equivalently provides 3D reference points. In spite of the efficiency of this approach, it requires expensive calibration apparatus and an elaborate set-up. The second technique (2D calibration) is performed observing a planar pattern shown at a few different orientations.[29] The third technique (1D calibration) is performed by observing objects composed of a set of collinear points (one of them fixed) from several different positions (minimum six).[27] To calibrate the relative geometry between images, it is necessary to observe the same points (three at least) simultaneously. It is hardly possible to achieve this with 3D or 2D calibration, but it is not a problem for this calibration. Another calibration approach, very effective in multiple camera calibration, can be performed moving a camera in a static scene, and using the rigidity of the scene and the image information alone.[30] Although no calibration objects are necessary, a large number of parameters need to be estimated, resulting in a much harder mathematical problem. These calibration approaches must use several images, but that is the consequence of using so little information.

There are other techniques which are difficult to classify. Some examples are vanishing points for orthogonal directions[30,31] and calibration from pure rotation.[30,32] Kim[8] classified the calibration methods into six categories: techniques involving full-scale nonlinear optimization, techniques involving computation of a perspective transformation matrix and use of linear equation solving, the two-plane method, the two stage method and adaptive self-calibration.

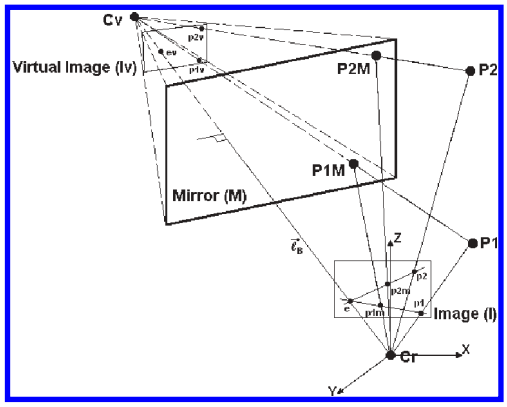
To calibrate the camera–mirror set, the geometry resulting from the planar mirror and camera position is explored. This calibration technique is simple and robust, since two perspectives are combined in just one image acquired by the camera (see Fig. 1). Figure 2 shows a ray-trace model of this image acquisition. This image acquisition simulates a multi-ocular geometry, with a virtual camera and a real one observing the same scene from two different viewpoints.

The calibration process has three phases:

1. the estimation of an epipole point;
2. the calculation of the camera's optical centre;
3. the estimation of its pose.



1 (a) Image from real camera; (b) image from virtual camera



2 Camera–mirror geometric model: P_1 and P_2 are two generic 3D points; $\mathbf{C}_r$ and $\mathbf{C}_r$ are real and virtual (generated by use of mirror) optical centres; mirror's reflections are P_{1M} and P_{2M}. Their image projections are p_1, p_2, p_{1m} and p_{2m}, respectively. Intersection of $\overline{p_1 p_{1m}}$ and $\overline{p_2 p_{2m}}$ defines epipole point in image. That point is image of intersection of segment $\overline{C_v C_r}$ with mirror plane

For the first two phases, the geometry of the acquisition system as illustrated in Fig. 2 is explored.

2.2 Camera model

The present paper refers to $P = (X, Y, Z)$ as a 3D world point and $p = (x, y)$ its projection in the image plane. When using homogeneous coordinates, those points will be represented, respectively, by ${}^W\boldsymbol{P} = [X \quad Y \quad Z \quad 1]^T$ and ${}^I\boldsymbol{p} = [x \quad y \quad 1]^T$.

The image projection is modelled using perspective projection. The relationship between P and p, in homogeneous coordinates, will be given by

$$\begin{aligned} k'^I\boldsymbol{p} &= \begin{bmatrix} fk_x & \gamma & c_x \\ 0 & fk_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} {}^W\boldsymbol{P} \\ &= \mathbf{C}\begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{r}_3 & \mathbf{t} \end{bmatrix} {}^W\boldsymbol{P} = \mathbf{C}[\mathbf{R} \quad \mathbf{t}]\,{}^W\boldsymbol{P} \\ &= \mathbf{CAL}_{(3\times4)}\,{}^W\boldsymbol{P} \end{aligned} \tag{1}$$

where k is an arbitrary scale factor, non-null, due to projective projection, $\mathbf{R}$ and $\mathbf{t}$ are, respectively, the rotation and translation, which relates the world coordinate system to the camera coordinate system, $\mathbf{C}$ is the matrix which contains the camera intrinsic parameters, $\mathbf{R}$ and $\mathbf{t}$ contain the camera extrinsic parameters. Note that $\mathbf{R}$ belongs to a $SO(3)$ group, with $\det(\mathbf{R}) = +1$, and it is always assumed ($\|r_1\| = \|r_2\| = \|r_3\| = 1$).

The intrinsic parameter f is the focal length, measured in the world metric system. The values (c_x, c_y) are the image coordinates of the principal point, and k_x and k_y are, respectively, the horizontal and vertical relationship between pixel units in the image and metre units (m).

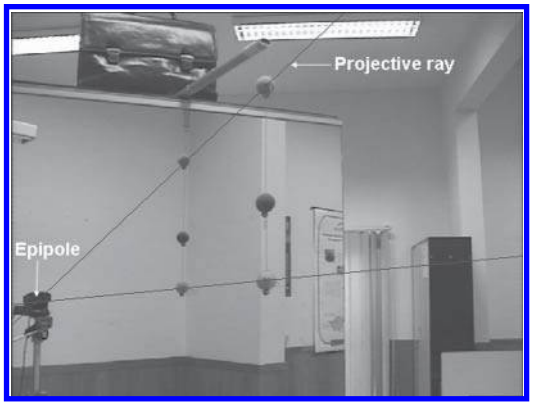
2.3 Epipole

With any two images from the same scene, the epipole in one image is the projection of the perspective centre of the other image. The two perspective centres and any point in the scene define a plane, called the epipolar plane, which intersects the two image planes defining epipolar lines. The epipole point is a unique point which belongs to all epipolar lines.

The usual techniques for estimating the epipoles requires first the estimation of the epipolar geometry from a stereo image set. This can be obtained by computing the fundamental matrix.[33] These techniques are usually very sensitive to small errors on the input data and image quality. Using a camera–mirror set gives a robust solution for the epipolar geometry and camera parameters.

The solution of the camera calibration problem was based on a so-called epipole 'by construction' method, following the geometric explanation in Fig. 2. The process uses the definition of image points from scenes similar to that illustrated in Fig. 3. A dedicated graphics software interface was designed to assist in the process.

To obtain the desired epipole **e** in the image, projections of at least two 3D points in the scene not belonging to the mirror plane were chosen. Those 3D



3 Definition of epipole point 'by construction': referential in world is defined in mirrors plane; XY plane ($Z = 0$) coincides with mirror plane, and Z axis with its normal

world points P and perspective centre define projection rays which intersect the planar mirror at points P_M. The epipole **e**, as shown in Fig. 3, is defined by the intersection of the two projection rays. Since epipole **e** corresponds to the projection of the virtual centre perspective $\boldsymbol{C}_v$, these two projection rays must intersect in the image of the camera.

The image points from the real camera with a corresponding point in the image from the virtual camera must belong to a line called the epipolar line.

2.4 Centre of perspective

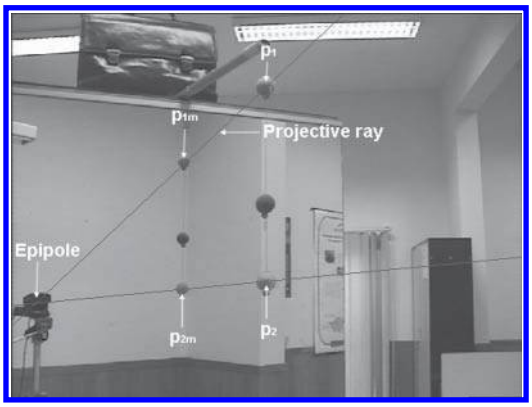
To recover the 3D coordinates of the optical centre, it is assumed that all rays of light passing through the camera optical centre are represented by 3D lines. If the intersection of one pair of those 3D lines is computed, a 3D point is obtained corresponding to the camera's optical centre. To pursue this approach, one needs to know the 3D coordinates of at least two points belonging to them. If the reflected point in the planar mirror and its image projection are used, it is necessary to know only one point on each 3D line (see Fig. 4).

Considering, with no loss of generality, the mirror plane is in the XY plane ($Z = 0$), 3D points in the mirror $P_{iM} = (X_M, Y_M, 0)$ could be related to image points $p_{im} = (x_m, y_m)$. A generic point in homogeneous coordinates in the mirror surface is given by

$$^{W}P_{iM} = k'[X \quad Y \quad 0 \quad 1]^T$$

with k' an arbitrary scale factor. Using equation (1), the images of those points are given by

$$k'^{I}\boldsymbol{P}_{im} = \mathbf{CAL}_{(3\times4)}{}^{W}\boldsymbol{P}_{iM} = \begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{bmatrix}\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \mathbf{T}_{XY}\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \tag{2a}$$



4 Definition of 3D lines

or in a more compact form

$$\frac{1}{k'}\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \mathbf{T}_{XY}^{-1}{}^{I}\boldsymbol{P}_{im} \tag{2b}$$

Thus, knowing $\mathbf{T}_{xy}$ and $^{I}\boldsymbol{p}_{im}$, one obtains $^{W}\boldsymbol{P}_{im}$. Rewriting equation (2b), one obtains

$$\mathbf{A}_i\mathbf{t}_{xy} = 0 \tag{3}$$

where

$$\boldsymbol{A}_i = \begin{bmatrix} X_M & Y_M & 1 & 0 & 0 & 0 & -X_M x_m & -Y_M x_m & -x_m \\ 0 & 0 & 0 & X_M & Y_M & 1 & -X_M y_m & -Y_M y_m & -y_m \end{bmatrix}$$

and

$$\mathbf{t}_{xy} = [t_{11} \quad t_{12} \quad t_{13} \quad t_{21} \quad t_{22} \quad t_{23} \quad t_{31} \quad t_{32} \quad t_{33}]$$

Analysis of equation (3) leads to the conclusion that it is necessary to know at least four 3D points on the mirror and its projections in the image to estimate $\mathbf{t}_{xy}$. Notice that each 3D point establishes two equations. The estimation of $\mathbf{t}_{xy}$ corresponds to solving the equation $\mathbf{A}\mathbf{t}_{xy} = 0$, where matrix **A** is composed of sub-matrices $\mathbf{A}_i$, which results in the use of different pairs of points ($^{I}\boldsymbol{p}_{im}$, $^{W}\boldsymbol{P}_{iM}$). Numerically, $\mathbf{t}_{xy}$ can be estimated using the eigenvector associated with the smallest eigenvalue of the matrix $\mathbf{A}^T\mathbf{A}$ to compute the fundamental matrix.[34]

To estimate $\mathbf{t}_{xy}$ in equation (3), it will be necessary to know the 3D coordinates of points in the scene. Assume, with no loss of generality, the origin of the world referential $\{W\}$ is a point in the mirror and its Z axis coincides with a normal to mirror plane. Thus, the mirror is the XY plane of $\{W\}$ and the 3D coordinates of points are easily obtained using marks or special characteristics of the 3D point (e.g. a corner of the mirror). The marks or characteristics are used only for this purpose and are recognisable in the image, where its 2D correspondents are manually obtained. Based on this procedure, a set of image point pairs $^{I}\boldsymbol{p}_{1m}$ and $^{I}\boldsymbol{p}_{2m}$, are defined and used to compute equation (3).

Knowing any two 3D points of the scene $^{W}\boldsymbol{P}_1$ and $^{W}\boldsymbol{P}_2$ not belonging to the mirror, and their correspondent image points $^{I}\boldsymbol{p}_{1m}$ and $^{I}\boldsymbol{p}_{2m}$, one can define the 3D lines corresponding to their projective rays, respectively $^{W}\boldsymbol{l}_1$ and $^{W}\boldsymbol{l}_2$. These lines are given by

$$^{w}\boldsymbol{l}_1 \equiv {}^{w}\boldsymbol{P}_1 - {}^{w}\boldsymbol{P} + \lambda\frac{{}^{w}\boldsymbol{P}_{1M} - {}^{w}\boldsymbol{P}_1}{\left\|{}^{w}\boldsymbol{P}_{1M} - {}^{w}\boldsymbol{P}_1\right\|}$$

and

$$^{w}\boldsymbol{l}_2 \equiv {}^{w}\boldsymbol{P}_2 - {}^{w}\boldsymbol{P} + \lambda\frac{{}^{w}\boldsymbol{P}_{2M} - {}^{w}\boldsymbol{P}_2}{\left\|{}^{w}\boldsymbol{P}_{2M} - {}^{w}\boldsymbol{P}_2\right\|}$$

Points ${}^{W}\boldsymbol{P}_{1M}$ and ${}^{W}\boldsymbol{P}_{2M}$ are 3D points on the mirror with $Z = 0$, and they are obtained from equation (2b) when it is applied to the image points ${}^{I}\boldsymbol{p}_{1m}$ and ${}^{I}\boldsymbol{p}_{2m}$, respectively.

The intersection of these two lines ${}^{W}\boldsymbol{l}_i$ and ${}^{W}\boldsymbol{l}_j$ defines the virtual centre perspective $\boldsymbol{C}_v$. From the reflective properties of planar mirrors, the centre of perspective $\boldsymbol{C}_r$ is given by the X and Y coordinates of the $\boldsymbol{C}_v$ point, with the symmetric Z coordinate. This come out directly from the geometry of the camera–mirror set (see Fig. 2).

The perspective centre (optical centre of the camera) $\boldsymbol{C}_r$ is given in metres, since the coordinates of the world points are expressed in that metric.

The line between $\boldsymbol{C}_r$ and $\boldsymbol{C}_v$ is defined as the baseline ${}^{W}\boldsymbol{l}_B$.

2.5 Intrinsic and extrinsic parameters

Matrix $\mathbf{T}_{xy}$ given by equation (2) models the projection between the image plane (measured in pixels) and the mirror plane (measured in metres). So, in order to know completely the elements of equation (1), which relates the coordinates of 3D points in the scene and their 2D image points, one needs to estimate more than three parameters to fulfil $\mathbf{CAL}_{(3\times4)}$.

2.5.1 *Estimation of* $\mathbf{t}$ *and* $\mathbf{r}_3$

Since the origin of the world coordinate system is on the mirror (see section 2.3) the real perspective centre $\boldsymbol{C}_r$ is equivalent to the translation $\mathbf{t}$ of equation (1). The translation $\mathbf{t}$ is estimated by the following procedure.

To compute $\mathbf{r}_3$, consider, as before, the mirror is in the XY plane ($Z = 0$). From the knowledge of $\mathbf{T}_{xy}$, $\boldsymbol{C}_r$ and equations (1) and (2)

$$k'\mathbf{T}_{XY} = \mathbf{C}[\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{t}] \tag{4}$$

for points belonging to the planar mirror surface. Expanding equation (4), the scale factor k' can be computed

$$k'\begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{bmatrix} = \begin{bmatrix} fk_x & \gamma & c_x \\ 0 & fk_y & c_y \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} r_{11} & r_{12} & t_x \\ r_{21} & r_{22} & t_y \\ r_{31} & r_{32} & t_z \end{bmatrix}$$

$$= \begin{bmatrix} r_{11}fk_x + r_{21}\gamma + r_{31}c_x & r_{12}fk_x + r_{22}\gamma + r_{32}c_x & t_xfk_x + t_y\gamma + t_zc_x \\ r_{21}fk_y + r_{31}c_y & r_{22}fk_y + r_{32}c_y & t_yfk_y + t_zc_y \\ r_{31} & r_{32} & t_x \end{bmatrix}$$

From the equality between the element in position (3, 3), of each matrix, from equation (4) one obtains

$$k' = \frac{t_z}{t_{33}} \tag{5}$$

Using this result in the equality for the elements in position (3, 1), one obtains

$$r_{31} = \frac{t_z t_{31}}{t_{33}}$$

For the elements in positions (3, 2), one obtains

$$r_{32} = \frac{t_z t_{32}}{t_{33}}$$

Since rotation $\mathbf{R}$ belongs to a $SO(3)$ group and ($\|r_1\| = \|r_2\| = \|r_3\| = 1$), the internal product between the same vector of the rotation is unitary. Using this, one obtains

$$r_{33} = \pm\sqrt{1 - r_{31}^2 - r_{32}^2}$$

Element r_{33} belongs to the unit vector representing the camera optical axis. That unit vector is used in the equations which model all projective rays, and the sign for r_{33} is defined according to the sign of the scale factor—see next section.

Using this procedure, the elements $\mathbf{t}$ and $\mathbf{r}_3$ of $\mathbf{R}$ are computed.

2.5.2 *Estimation of* (c_x, c_y), k_x, k_y *and* f

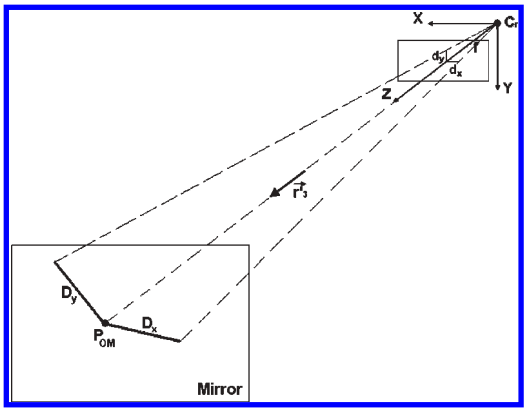
The rotation matrix $\mathbf{R}$ expresses the orientation of the world referential in the camera referential. From the properties of the group $SO(3)$, it is known that $\mathbf{R}^{-1} = \mathbf{R}^T$. This means that the last row of $\mathbf{R}$, $\mathbf{r}'_3 = [r_{31} \quad r_{32} \quad r_{33}]^T$ is the unit vector normal to the image plane and parallel to the optical axis. Any point in the optical vector belongs to the 3D line

$${}^{W}\boldsymbol{l}_o = \mathbf{C}_r + \gamma\mathbf{r}'_3$$

Computing the intersection of ${}^{W}\boldsymbol{l}_o$ with the mirror plane ($Z = 0$), one obtains a point P_{oM} (see Fig. 5). The projection of this point in the image plane will define the principal point $p_{om} = (c_x, c_y)$. Equation (2) allows the coordinates (c_x, c_y) to be computed from P_{oM}.

To know the metric relations between horizontal and vertical pixels and metres, k_x and k_y, respectively, the process described in Fig. 5 is followed.

Thus, to obtain the values for parameters k_x and k_y, two distances are defined on the image plane and two distances on the mirror plane. In the image, each distance, in pixels, is defined in vertical and horizontal directions of the image d_y and d_x, respectively. The two distances d_y and d_x have equivalent distances D_y and D_x, respectively, on the mirror plane, defined in metres (see Fig. 5). Those distances have values that can be computed by equation (2b).



5 Geometric model used to compute principal point and metric relations between horizontal/vertical pixels and metres, respectively, k_x and k_y

Parameters k_x and k_y are expressed by

$$k_x = \frac{d_x}{D_x} \qquad k_y = \frac{d_y}{D_y}$$

where

$$d_x = \overline{p_1 - p_0} \qquad d_y = \overline{p_2 - p_0}$$

and, using equation (2b), one obtains

$$D_x = \overline{T_{XY}^{-1}\, p_1 - T_{XY}^{-1}\, p_0}$$
$$D_y = \overline{T_{XY}^{-1}\, p_2 - T_{XY}^{-1}\, p_0}$$

In spite of the image coordinate points (p_0, p_1 and p_2) being in pixels, distances D_y and D_x are in metres owing to the use of matrix $\mathbf{T}_{xy}$. This scale issue is in the matrix because its estimation was done with world points (measured in metres) and image points (measured in pixels).

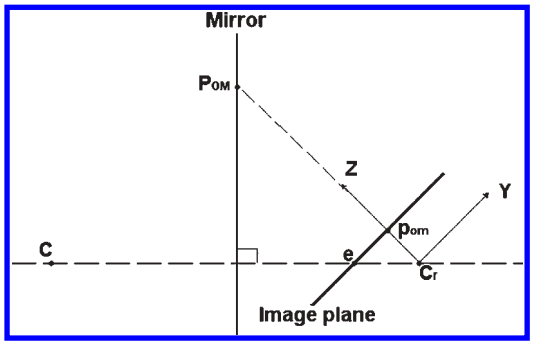
In practice, to avoid distortion effects from the lens, points p_0, p_1 and p_2 are centred on the image.

The information estimated so far enables the focal length f to be estimated based on the similarity between triangles (see Fig. 6). Based on this similarity, the focal length is given by

$$f = \frac{t_z(\overline{p_{\mathrm{om}}e})}{\overline{P_{\mathrm{oM}} - P_{\mathrm{EM}}}}$$

Point p_{om} is the principal point in the image; the point P_{oM} is the intersection of $\boldsymbol{l}_{\mathrm{o}}$ with the mirror plane, **e** is the estimated epipole, and P_{EM} is the intersection of the baseline $\overline{\mathbf{C}_{\mathrm{r}}\mathbf{C}_{\mathrm{v}}}$ with the planar mirror. Notice the distance $\overline{\mathbf{C}_{\mathrm{r}}\mathbf{C}_{\mathrm{v}}}$ is perpendicular to the mirror, and $\overline{\mathbf{C}_{\mathrm{r}}\mathbf{P}_{\mathrm{oM}}}$ is perpendicular to the image plane.

To have all the measurements in the same metric, it is necessary to divide the x and y coordinates of the image points by k_x and k_y, respectively.



6 Geometric model used to compute focal length

2.5.3 *Estimation of* fk_y

A technique to reduce error propagation in the calibration process is to compute fk_y on behalf of k_y. Using equation (4) and the equality between the elements (2, 3), fk_y can be estimated, where

$$fk_y = \frac{t_z t_{23} - c_y t_x t_{33}}{t_y t_{33}}$$

However, the previous computations of k_x and k_y are usually used.

2.5.4 *Estimation of* $\mathbf{r}_1$ *and* $\mathbf{r}_2$

Based on the previously obtained parameters and equations (4) and (5), it is possible to compute parameters $\mathbf{r}_1$ and $\mathbf{r}_2$ from $\mathbf{R}$.

Using equation (4) and the equality between elements (2, 1) of each matrix, one obtains

$$r_{21} = \frac{k' t_{21} - c_y r_{31}}{fk_y}$$

Now using f, k_x, k_y, c_x, c_y and equation (5), from the equality between the element in position (2,2) of each matrix in equation (4), one gets

$$r_{22} = \frac{k' t_{22} - c_y r_{32}}{fk_y}$$

Usng the same process for the element in position (1, 3) of each matrix in equation (4), one obtains

$$\gamma = \frac{k' t_{13} - c_x t_z - fk_x t_x}{t_y}$$

Finally, using all the parameters estimated so far

$$r_{11} = \frac{k' t_{11} - c_x t_{31} - \gamma r_{21}}{fk_x}$$
$$r_{12} = \frac{k' t_{12} - c_x r_{32} - \gamma r_{22}}{fk_x}$$

is calculated.

This process enables parameters $\mathbf{r}_1$ and $\mathbf{r}_2$ to be estimated from $\mathbf{R}$. From $\mathbf{r}_1$ and $\mathbf{r}_2$ and using the properties of rotation matrix $\boldsymbol{R}$, the previous values for $\mathbf{r}_3$ can be validated from

$$\mathbf{r}_3 = \mathbf{r}_1 \wedge \mathbf{r}_2 \tag{6}$$

2.6 Calibration procedure

A summary of the camera–mirror calibration is as follows:

1. Define the epipole **e**—section 2.3.
2. Estimate the transformation $\mathbf{T}_{xy}$ between the image plane and the planar mirror—section 2.4.
3. Estimate the centre of perspective of the real camera and virtual camera $\boldsymbol{C}_{\mathrm{r}}$ and $\boldsymbol{C}_{\mathrm{v}}$, respectively—section 2.4.
4. Estimate **t** and $\mathbf{r}_3$—section 2.5.
5. Estimate (c_x, c_y), k_x, k_y and f—section 2.5.
6. Estimate $\mathbf{r}_1$ and $\mathbf{r}_2$—section 2.5.

2.7 Camera calibration results

To exemplify the camera–mirror calibration process, images similar to those in Fig. 1 were used. From that image and the camera calibration process described above, the intrinsic parameters of the camera (matrix **C**) are obtained, given by

$$\mathbf{C} = \begin{bmatrix} 1079.0 & -235.3 & 345.3 \\ 0 & 1054.0 & 746.6 \\ 0 & 0 & 1 \end{bmatrix}$$

The extrinsic parameters for each chosen position (relative orientation) are

$$\mathbf{R}_{\mathrm{a}} = \begin{bmatrix} -7.240e-2 & 9.974e-1 & 7.025e-2 \\ 9.974e-1 & 7.241e-2 & -6.500e-3 \\ -7.013e-3 & 6.511e-3 & 9.999e-1 \end{bmatrix}$$

$$\mathbf{t}_{\mathrm{a}} = \begin{bmatrix} 5.555e+2 \\ 5.528e+2 \\ 1.456e+3 \end{bmatrix}$$

Another interesting example was to see the behaviour of the calibration algorithm when the skew parameter was set to zero ($\gamma = 0$). So, for the same data used above one gets

$$\mathbf{C} = \begin{bmatrix} 1079.0 & 0 & 322.8 \\ 0 & 1054.0 & 516.0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{R}_{\mathrm{a}} = \begin{bmatrix} -7.240e-2 & 9.974e-1 & 7.025e-2 \\ 9.974e-1 & 7.241e-2 & -6.500e-3 \\ -7.013e-3 & 6.511e-3 & 9.999e-1 \end{bmatrix}$$

$$\mathbf{t}_{\mathrm{a}} = \begin{bmatrix} 5.555e+2 \\ 5.528e+2 \\ 1.456e+3 \end{bmatrix}$$

Both the results are acceptable because, when they are used in known 3D points, the corresponding known image points are obtained. It is noticeable that the translation is the same, because the camera does not move. The differences between the rotation and calibration matrices are the readjustment due the skew.

To verify the relative magnitude of errors that arise in the present calibration method (its robustness), uniform noise was randomly added to the set of points used to compute the calibration parameters. First, the calibration parameters were obtained without introducing noise, and then those parameters were estimated with different noise levels. The results in Table 1 are the ratio between the parameters obtained with noise and the parameters without noise.

Table 1 shows that the method gives acceptable parameters $\leq 5\%$ of noise. Beyond that percentage, some of the results did not make any practical sense. As illustrated in the table, the method started to be more unstable with additive noise $> 5\%$ on the points used as input for the method. These results are mainly due to data flow of the method where the previous computation of some parameters is used to compute the others (error propagation).

This method is also compared with Zhang's method.[29] Thus, for the same camera, the same scene was used and three more images were captured from different positions. Using the minimum number of images, the resulting camera's intrinsic parameters (matrix **C**) are given by

Table 1 Relative magnitude of error for specific noise level in used points (%)

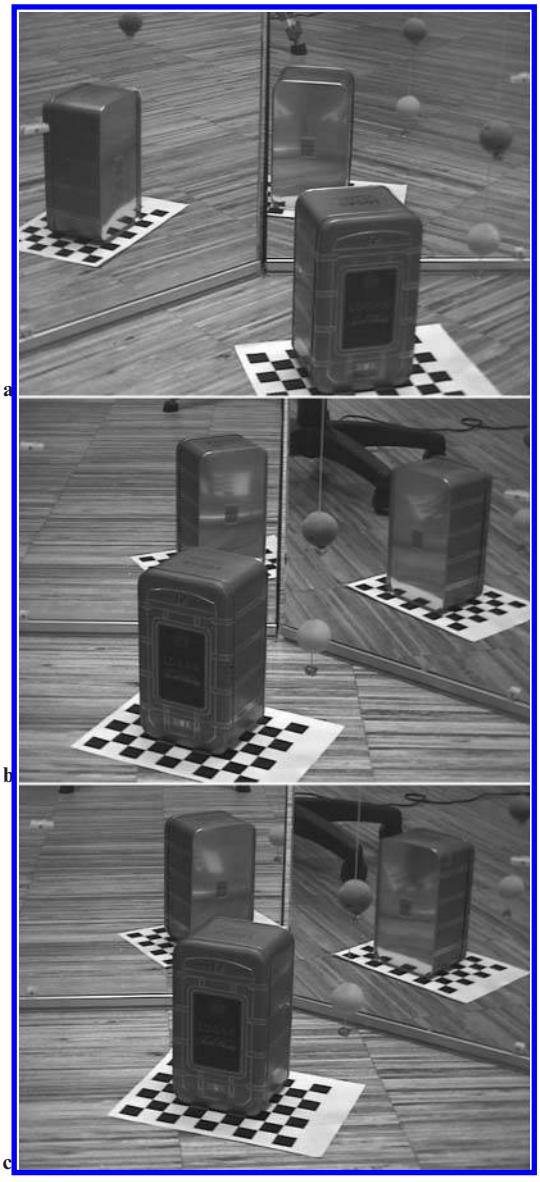
	Noise level				
	1%	5%	6%	7%	10%
c_x (pixel)	–2	–14	38	81	—
fk_x (pixel)	15	55	—	—	—
fk_y (pixel)	10	40	—	—	—
k_y (m)	0.0003	–0.0011	–0.0032	–0.0041	–0.005

— values not available.

$$\mathbf{C} = \begin{bmatrix} 1079.0 & 0 & 322.8 \\ 0 & 1054.0 & 516.0 \\ 0 & 0 & 1 \end{bmatrix}$$

The comparison shows that, in spite of the differences, the parameters follow the same pattern for the same relative additive noise in the data points used to obtain Table 1. However, this method uses only a simple image instead of the four in Zhang's method. This method is performing well mainly owing the geometric properties of the reflected images which are useful for camera calibration.

Finally, as a last example, important for use in the following sections of paper, images were acquired in three different positions, as illustrated in Fig. 7.



7 Images used in surface reconstruction process

The intrinsic parameters of the camera are

$$\mathbf{C} = \begin{bmatrix} 4.712e-3 & 0 & -3.851e-9 \\ 0 & 1.842e-3 & 2.701e-10 \\ 0 & 0 & 1 \end{bmatrix}$$

Since the camera is the same, matrix **C** is equal to all three positions.

The extrinsic parameters for each chosen position (relative orientation) are

Position (a)

$$\mathbf{R}_a = \begin{bmatrix} -9.967e-1 & -5.666e-2 & -5.754e-2 \\ 3.372e-3 & 2.942e-1 & -9.558e-1 \\ 1.456e-7 & -4.664e-7 & -4.014e-7 \end{bmatrix}$$

$$\mathbf{t}_a = \begin{bmatrix} 4.142e+2 \\ 3.657e+2 \\ 1.885e-3 \end{bmatrix}$$

Position (b)

$$\mathbf{R}_b = \begin{bmatrix} -9.865e-1 & -1.614e-1 & -2.953e-2 \\ -1.642e-2 & 1.429e-2 & -9.998e-1 \\ -1.046e-6 & -2.551e-6 & -2.054e-7 \end{bmatrix}$$

$$\mathbf{t}_b = \begin{bmatrix} 4.443e+2 \\ 3.305e+2 \\ 2.277e-3 \end{bmatrix}$$

Position (c)

$$\mathbf{R}_c = \begin{bmatrix} 9.930e-1 & -1.173e-1 & 1.043e-2 \\ 1.220e-1 & -2.675e-1 & 9.558e-1 \\ 8.999e-7 & -1.960e-6 & 7.646e-8 \end{bmatrix}$$

$$\mathbf{t}_c = \begin{bmatrix} -5.029e+2 \\ -5.310e+2 \\ -1.790e-3 \end{bmatrix}$$

3 RECONSTRUCTION

The projection matrices obtained by the calibration method described above will be used in a reconstruction process. The 3D reconstruction results are dependent on the results from the calibration method, and they are a good assessment for the calibration process.

First, the algorithm used to reconstruct from images, which is based on the technique from volumetric modelling known as space carving,[4] is described. Then, some reconstruction results are presented, one using the images in Fig. 7 and its projection matrices.

It should be noted that the reconstruction process does not take advantage of the use of mirrors and can be used with other sets of calibrated images obtained from different viewpoints. The reconstruction method described in the paper only needs to have calibrated images of the 3D object to be reconstructed, and some of those images can be images reflected from mirrors.

As described in section 1, the technique is applied to a volume previously defined. That volume is carved until the resulting surface is consistent with the acquired images. One voxel (a volumetric element of the 3D world) is consistent with an image when its radiation value is similar to its colour projection in the image. So, in the excavation process there is a radiation function which attributes its radiation value to a voxel in space. The consistence criterion is estimated for the voxel and, if it is satisfied, the radiation value is attributed accordingly. Otherwise, the voxel will be eliminated, since it does not belong to the volume being reconstructed. This process is repeated until no more voxels can be eliminated.

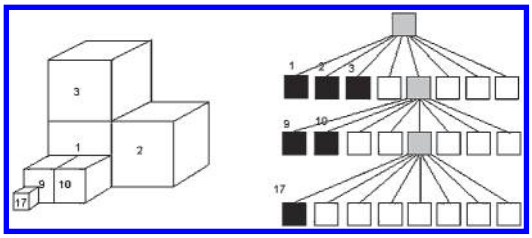
This approach follows Wong's work[5] to obtain the shape and texture of the object in the scene. Thus, the reconstruction algorithm creates an octree from the images and uses a variation of the consistence test. Instead of a function that models the scene radiation, silhouettes from the object are used. These silhouettes are obtained from silhouettes extracted by an *edge* filter. This variation improves the test of consistence for each voxel.

The surface texture is computed in parallel with the volume carving. Since each image from a surface point defines a set of colours, the final texture will be generated from that set of colours, as described in section 3.3.

3.1 Octree representation

An octree[35] is a data structure with the shape of a tree, in which every node has, at most, eight ramifications. It is generally used in graphic computation for a volumetric representation of objects, where every node on the tree represents a part of the object surface, called a voxel (volume element), in 3D space.

The representation of the objects is achieved as follows. The root node of the octree consists of a sole voxel of large dimension, which defines the involving volume of the object. The octree is built, subdividing recursively every voxel of the tree into eight sub-voxels, which are represented by eight resulting nodes. To every node of the tree is attributed one of three colours (black, grey or white), according to the occupation of the voxels with the object (occupation rate). A black node represents a completely filled voxel, a grey node represents a partially filled voxel, and a white node represents a completely empty voxel. It is noticeable that neither the black nor the white nodes have ramifications. The grey nodes have ramifications that could have one of the three colours. Figure 8 shows a simple volume represented by an octree.



8 Representation of simple volume by octree

More details concerning the building and manipulation of the octree can be found in the work of Jackins and Tanimoto[35] and Chen and Huang.[36]

In the present implementation, the voxels are cubes and every sub-division generates eight identical sub-cubes. Every node in the octree stores the voxel colour, its length and the coordinates of its centre. It also contains information about its ramifications, if there are any. The voxel colour represents its occupation level with the object. So, it will be black if completely filled, grey if partially filled or white if completely empty.

3.2 Space carving

The algorithm, based on silhouettes, initially establishes the root node of the octree as a grey node which completely enfolds the object. From this node new levels are formed, subdividing every grey node from the previous level into eight sub-nodes. Neither the white nor the black nodes need to be subdivided because all their possible ramification nodes would have the same classification as the original ones. To test whether a voxel must be divided, the consistence test is applied by projecting the voxels in every used image and, with the resulting points, verifying whether they belong to the silhouette in those images.

The algorithm for the 3D reconstruction, using calibrated images of the same scene, collected from several different positions, is given by the following pseudo code:

```
Set the size of the cube that contains the object;
Set the number of levels for the octree;
```

```
Initialise the root node of the octree as grey;
While maximum level is not reached do,
  If no grey node in the current level then,
    End the process;
  For each grey node of the current level do,
    Subdivide it into eight sub nodes;
    For each sub node do,
      Set its colour to black;
      For each image in the sequence do,
        Project the cube onto the image;
        If the projection lies completely outside the silhouette
        then,
          Set the colour of the correspondent node to white;
          Ignore the rest of the images;
        Else,
          If the projection lies partially inside the silhouette
          then,
            Set the colour of the correspondent node to grey;
        Else,
            Keep the colour of the node;
```

In this algorithm, the bigger the division of the initial volume, the better the reconstruction resolution will be. However, for high-resolution volumes, the process will be computationally slower, since the number of voxels to test will increase by 8^{NL}, NL being the number of the octree level.

After completing the process of removing the voxels, the remnants correspond to the 3D reconstructed surface, which ensures geometric and photometric consistency of the images of the surface, as illustrated in Fig. 9.

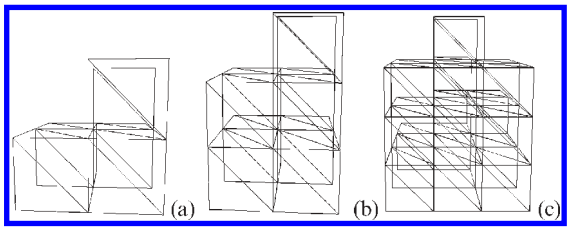
A graphic library, OpenGL in this case, is used to render the information included in the octree.

3.3 Textures

The textures are obtained in the same phase where the voxels are tested. To recover the texture of a voxel, the projection of the voxel on each the image is computed. The final texture is a linear combination of the colour set visible from each image. In the limit, the projection of a voxel, on each image, is confined to a unique point. In that case, the colour of the point in the image will be colour of the voxel.

In practice, as several images of the 3D surface are used and these images are captured from different positions, the colour in each image, related to the same 3D point, can be different. The solution of this problem is to define the colour of the voxel by averaging the colours of its projection in all the images.

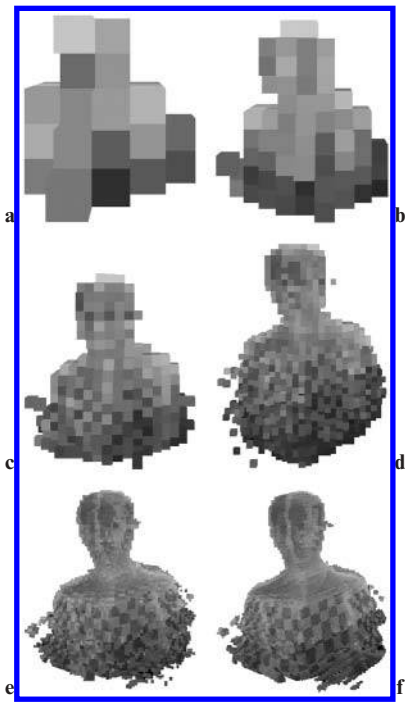
In high-resolution volumes, more voxel points are used and better results will be achieved, as illustrated in Fig. 10.



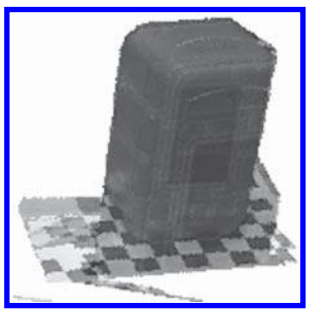
9 Different levels of octree: (a) 2; (b) 3; (c) 4

3.4 Reconstruction results

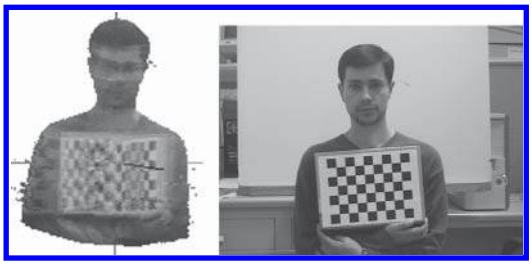
The results from the reconstruction process are presented in Fig. 11, using the calibration results presented in section 2 for images illustrated in Fig. 7. This result is supported by a 10-level octree.



10 Textured results by levels: (a) 4; (b) 5; (c) 6; (d) 7; (e) 8; (f) 9



11 Reconstruction result, using space carving process, of scene presented in Fig. 7

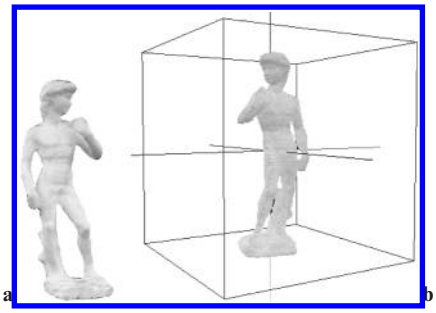


12 Reconstruction results of person: figure results from rendering application which enables metric characteristics of volume to be measured

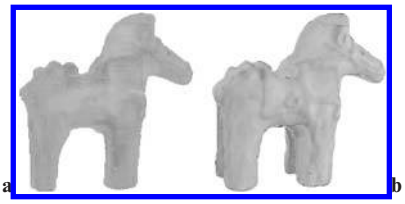
In order to show these method capabilities, a 3D reconstruction of a person is presented in Fig. 12. This result is supported by a nine-level octree, with more accuracy of texture recovery than the results in Fig. 10. Instead of four points of the voxel's projections, more points were used to calculate the final texture of the voxel. One of the images used which produced that result is presented to serve as ground truth in the process of validating this method. For more dense details, more octree levels can be used.

To compare the processing time and results with other techniques, the Octcarve[5] application, available on the Internet,[37] was used. For comparisons, some of the sets of calibrated images, downloaded from the Internet,[37] were used, namely the 'David' and 'Haniwa' sets. For the 'David' images, the results from the Octcarve application are presented in Fig. 13a, obtained with a processing time of 1 min 15 s. With the present application, the result presented in Fig. 13b was obtained with a processing time of 0 min 56 s. Both results are obtained by parameterisation of an octree with 10 levels.

For the images from 'Haniwa', the results from the Octcarve application are presented in Fig. 14a, obtained with a processing time of 3 min 53 s. With the present application, the result presented in Fig. 14b



13 Reconstruction results of 'David': (a) from Octcarve; (b) from present program



14 Reconstruction results of 'Haniwa': (a) from Octcarve; (b) from present program

was obtained with a processing time of 3 min 02 s. Both results are obtained with a parameterisation of an octree with 10 levels.

The results verify that, in spite of the present program being faster, the results from the Octcarve application provide more detailed reconstruction. To achieve similar results with the present application, more octree levels need to be used.

4 CONCLUSIONS

The present paper developed a flexible new technique for calibrating the camera, based on mirrors, applying it to the surface reconstruction problem. This calibration method begins with Sameer's idea[7] and expands it in order to obtain all the information needed concerning the intrinsic camera parameters and the camera relative orientation. It is a very simple method because it requires the knowledge of only four points in a planar mirror, two 3D points between the camera and the planar mirror, and the projections of all 3D points in each captured image. Although the points were not measured with a precision device and data normalisation was not used on the different estimation processes, the present method obtains good reconstruction results, as shown in Figs 11 and 12. These results can be improved using more levels in the octree with a subsequent increase in computing time.

REFERENCES

1 Cipolla, R. and Blake, A. Surface shape from the deformation of apparent contours. *Int. J. Comput. Vision*, 1992, **9**(2), 83–112.

2 Kanade, T., Narayanan, P. J. and Rander, P. W. Virtualized reality: concepts and early results. In Proceedings of Workshop on *Representations of Visual Scenes*, 1995, IEEE Computer Society, Washington, DC, pp. 69–76.

3 Poggio, T., Torre, V. and Koch, C. Computational vision and regularization theory. *Nature*, 1985, **317(**26), 314–319.

4 Kutulakos, K. N. and Seitz, S. M. A theory of shape by space carving. In Proceedings of the 7th International Conference on *Computer Vision*, Kerkyra, Greece, September 1999, IEEE Computer Society, Washington, DC, pp. 307–314.

5 Wong, K.-Y. K. and Cipolla, R. Structure and motion from silhouettes. In Proceedings of the 8th International Conference on *Computer Vision*, Vancouver, Canada, 2001, Vol. 2, IEEE Computer Society, Washington, DC, 217–222.

6 Hecht, E. and Zajac, A. *Optics*, 1974 (Addison Wesley, Reading, MA).

7 Nene, S. A. and Nayar, S. K. Stereo with mirrors. In Proceedings of the 6th International Conference on *Computer Vision (ICCV'98)*, Bombay, India, January, 1998, IEEE Computer Society, Washington, DC.

8 Kim, W. S. and Cho, H. S. A novel omnidirectional image sensing system for assembling parts with arbitrary cross-section shapes. *IEEE/ASME Trans. Mechatron.*, 1998, **3**.

9 Mitsumoto, H., Tamura, S., Okazaki, K., Kajimi, N. and Fukui, Y. 3d reconstruction using mirror images based on a plane symmetry recovery method. *IEEE Trans. Pattern Anal. Machine Intell.*, 1992, **9**, 941–945.

10 Nayar, S. K. Sphereo: recovering depth using a single camera and two specular spheres. In *Optics, Illumination, and Image Sensing for Machine Vision II*, Proceedings of SPIE, November, 1988.

11 Svoboda, T., Pajdla, T. and Hlavac, V. Epipolar geometry for panoramic cameras, In Fifth European Conference on *Computer Vision*, 1998, Springer, Heidelberg, pp. 218–232.

12 Fonseca, I. and Dias, J. Exploring spherical image properties for robot navigation. In Proceedings of IROS'98-IEEE/RSJ International Conference on *Intelligent Robots and Systems*, Victoria, BC, Canada, 13–17 October, 1998, IEEE Computer Society, Washington, DC.

13 Mathieu, H. and Devernay, F. Système de miroirs pour la stéréoscopie. Technical Report 0172, INRIA Sophia-Antipolis, 1993.

14 Zhuang, H. Modeling gimbal axis misalignments and mirror centre offset in a single-beam laser tracking measurement system. *Int. J. Robotics Res.*, 1995, **14**, 211–224.

15 Tamura, S., Kim, E. K. and Sato, Y. Error correction in laser scanner three-dimensional measurement by two-axis model and coarse-fine parameter search. *Pattern Recognition*, 1994, **27**, 331–338.

16 Brown, D. C. Close-range camera calibration. *Photogramm. Eng.*, 1971, **37**, 855–866.

17 Wong, K. W. Mathematical formulation and digital analysis in close-range photogrammetry. *Photogramm. Eng. Remote Sensing*, 1975, **41**, 1355–1373.

18 Toscani, G. and Faugeras, O. The calibration problem for stereo. In Proceedings of the IEEE Conference on *Computer Vision and Pattern Recognition*, Miami Beach, FL, June 1986, IEEE Computer Society, Washington, DC, pp. 15–20.

19 Faugeras O., Luong, T. and Maybank, S. Camera self-calibration: theory and experiments. In Proceedings of the 2nd ECCV, *Lecture Notes in Computer Science*, Vol. 588, 1992, pp. 321–334 (Springer, Berlin).

20 Ganapathy, S. Decomposition of transformation matrices for robot vision. *Pattern Recognition Lett.*, 1984, **2**, 401–412.

21 Gennery, D. Stereo-camera calibration. In Proceedings of the *Image Understanding* Workshop, Los Angeles, CA, 1979, Science Applications, Arlington, VA, pp. 101–108.

22 Maybank, S. J. and Faugeras, O. D. A theory of self-calibration of a moving camera. *Int. J. Comput. Vision*, 1992, **8**(2), 123–152.

23 Tsai, R. Y. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE J. Robotics Automat.*, 1987, **3**, 323–344.

24 Han, M. H. and Rhee, S. R. Camera calibration for three-dimensional measurement, *Pattern Recognition*, 1992, **25**, 155–164.

25 Wei, G. Q. and Ma, S. D. A complete two-plane camera calibration method and experimental comparisons. In Proceedings of the 4th International Conference on *Computer Vision*, Berlin, May, 1993, IEEE Computer Society, Washington, DC, pp. 439–446.

26 Weng, J. P., Cohen, M. and Herniou, M. Camera calibration with distortion models and accuracy evaluation. *IEEE Trans. Pattern Anal. Machine Intell.*, 1992, **10**(14), 965–980.

27 Zhang, Z. *Camera Calibration with One-dimensional Objects*, 2002, pp. 161–174 (ECCV, Springer, Berlin).

28 Faugeras, O. *Three-Dimensional Computer Vision: a Geometric Viewpoint*, 1993 (MIT Press, Cambridge, MA).

29 Zhang, Z. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Machine Intell.*, 2000, **11**(22), 1330–1334.

30 Hartley, R. Self-calibration from multiple views with a rotating camera. Proceedings of the 3rd European Conference on *Computer Vision*, *Lecture Notes in Computer Science*, Vol. 800–801, 1994, pp. 471–478 (Springer, Berlin).

31 Liebowitz, D. and Zisserman, A. Metric rectification for perspective images of planes. In Proceedings of the IEEE Conference on *Computer Vision and Pattern Recognition*, IEEE Computer Society, Santa Barbara, CA, June,

1998, IEEE Computer Society, Washington, DC, pp. 482–488.

32 Stein, G. Accurate internal camera calibration using rotation, with analysis of sources of error. In Proceedings of the 5th International Conference on *Computer Vision*, Cambridge, MA, June, 1995, IEEE Computer Society, Washington, DC, pp. 230–236.

33 Hartley, R. and Zisserman, A. *Multiple View Geometry*, 2000, ISBN 0521623049 (Cambridge University Press, Cambridge).

34 Zhang, Z. Determining the epipolar geometry and its uncertainty: a review. *Int. J. Comput. Vision*, 1998, **27**(2), 161–195.

35 Jackins, C. L. and Tanimoto, S. L. Octrees and their use in representing three-dimensional objects. *Comput. Graph. Image Process.*, 1980, **14**(3), 249–270.

36 Chen, H. H. and Huang, T. S. A survey of construction and manipulation of octrees. *Comput. Vision Graph. Image Process.*, 1988, **43**, 409–431.

37 http://www.csis.hku.hk/~kykwong/