Automatic 3D Model Construction for Turn-Table Sequences

Andrew W. Fitzgibbon, Geoff Cross and Andrew Zisserman
{awf,geoff,az}@robots.ox.ac.uk

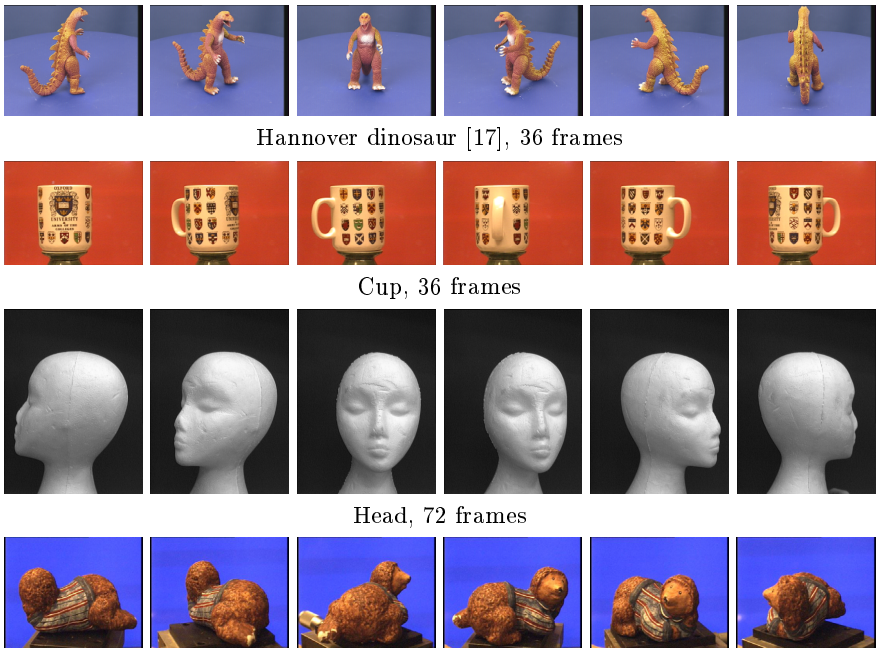
Robotics Research Group, Department of Engineering Science,
University of Oxford, 19 Parks Road, Oxford OX1 3PJ, United Kingdom
http://www.robots.ox.ac.uk/~vgg

Abstract. As virtual worlds demand ever more realistic 3D models, attention is being focussed on systems that can acquire graphical models from real objects. This paper describes a system which, given a sequence of images of an object rotating about a single axis, generates a textured 3D model fully automatically. In contrast to previous approaches, the technique described here requires no prior information about the cameras or scene, and does not require that the turntable angles be known (or even constant through the sequence).
From an analysis ofthe projective geometry of the situation, it is shown that the rotation angles may be determined unambiguously, and that camera calibration, camera positions and 3D structure may be determined to within a two parameter family. An algorithm has been implemented to compute this reconstruction fully automatically. The two parameter reconstruction ambiguity may be removed by specifying, for example, camera aspect ratio and parallel scene lines. Examples are presented on four turn-table sequences.

1 Introduction

Numerous graphics and computer vision papers have dealt with the construction of 3D solid models by volume intersection from multiple views. As pointed out by Ponce [19] the idea dates back to Baumgart [2] in 1974. Well engineered systems built on this idea have yielded 3D texture mapped graphical models of impressive quality [5, 19]. A good example is the system of Hannover [17] where, as is usual for such systems, the object is rotated on a turntable against a background which can easily be removed by image segmentation. Such systems are generally completely calibrated, i.e. the camera internal parameters, rotation angles, distance to the rotation axis etc are all accurately known.

In this paper we develop the projective geometry of single axis rotation and describe its automatic and optimal estimation from an image sequence with no other *a priori* information supplied. It is shown that 3D structure and cameras can be estimated (including auto-calibration) up to an overall two-parameter ambiguity. The angle of rotation between views is not ambiguous. This geometry is described in section 2, and an algorithm to automatically estimate this geometry from an image sequence is given in section 3.



Hannover dinosaur [17], 36 frames

Cup, 36 frames

Head, 72 frames

Freiburg, 36 frames

Fig. 1. Some example sequences.

We then describe a modelling system based on this estimated geometry. The input is a turn-table image sequence, the output is the set of cameras and a 3D VRML texture mapped model of the object, with all processing automatic. Other than the estimation of the camera geometry, the system is much the same as the calibrated Hannover system, and involves: volume intersection; representation of the surface as a triangulated network; triangle grouping; and texture mapping. This is described in section 4. The output models are of equal quality to those of fully calibrated sequences—a fact demonstrated on a sequence supplied by Hannover and shown in figure 1.

General uncalibrated multiple-view geometry Before specializing to single axis rotation, consider first the general case of reconstruction from multiple pinhole cameras viewing a 3D scene [10]. 3D points $\mathbf{X}$ in the scene are represented as homogeneous 4-vectors $[\mathrm{X}, \mathrm{Y}, \mathrm{Z}, 1]^\top$, while their 2D projections $\boldsymbol{x}$ are represented as homogeneous 3-vectors $[x, y, 1]^\top$. The action of each camera is represented by a 3×4 projection matrix $\mathtt{P}$:

$$\boldsymbol{x}_{ij} = \mathtt{P}_i \mathbf{X}_j$$

The m cameras are indicated by $\mathtt{P}_i$ while the n 3D points are $\mathbf{X}_j$.

In L. Van Gool and R. Koch, Editors, *Structure and Motion from Multiple Images in Large-Scale Environments*, Lecture Notes in Computer Science, Springer 1998.

In the case where m different cameras view a scene, there is no relationship between the $\mathtt{P}_i$. Therefore $11m$ parameters are required to specify all the cameras. When the cameras have identical internal parameters, such as when a camera is moved through a static scene without any change in focus or zoom, the internal parameters are constant over the sequence. This reduces the number of parameters required to specify the cameras from $11m$ to $6m + 5$.

Uncalibrated single-axis multiple-view geometry In the single-axis case, we shall see that the number of parameters is reduced to $m + 8$, and that estimation is relatively straightforward. It is worth contrasting the reduction in the number of parameters that occurs in this special motion case, with a popular alternative which is to reduce the number of parameters by approximating the perspective camera as a weak perspective camera [8, 18]. Such "affine" cameras are an approximation of the geometry, and under imaging conditions which typically apply in close-range model acquisition, this approximation can be quite poor. However, the advantage of this approximation is a simple, non-iterative estimation algorithm [23]. In contrast, specializing the motion to single axis is an exact model of the geometry, not an approximation, yet it admits a closed-form solution. Previous investigations of turn-table sequences [12, 20, 22] have not fully exploited the special motion to simplify camera recovery.

2 The projective geometry of single axis motion

A single axis motion consists of a set of Euclidean actions on the world such that the relative motion between the scene and camera can be described by rotations about a single fixed axis. In the language of screw decompositions, any Euclidean action can be decomposed as a rotation about a screw axis (which is parallel to the Euclidean rotation axis) together with a translation along the screw axis. In the case of single axis motions there is zero translation along the screw axis, and the screw axes of each Euclidean action coincide.

There are many cases of this motion commonly occurring in computer vision applications. The most common, and the one that is used here, is the case of a static camera viewing an object rotating on a turntable. A second case is that of a camera rotating about a fixed axis. For example, imagine a QuickTime VR acquisition device where the camera is offset along its principal axis, so that it does not rotate about its centre. A third case is that of a camera viewing a rotating mirror.

It will be helpful in the following to consider that the object is fixed and that the camera rotates about it. The camera internal parameters are fixed. To aid visualisation, we assume that the rotation axis is vertical, so that the camera rotates in a horizontal plane.

We now describe the camera and image geometry arising from this constrained motion, particularly the fixed entities of the motion, which play an important rôle. It will be seen that the fundamental matrix, $\mathtt{F}$, trifocal tensor $\mathcal{T}$ and camera matrices $\mathtt{P}$ all have additional properties, and that the multiple

In L. Van Gool and R. Koch, Editors, *Structure and Motion from Multiple Images in Large-Scale Environments*, Lecture Notes in Computer Science, Springer 1998.

view tensors ($\mathtt{F}$ and $\mathcal{T}$) determine a two-parameter family of camera matrices. This ambiguity is removed using internal and external constraints.

2.1 3D Geometry

Under a single axis rotation the camera centre describes a circle in a horizontal plane π_h. The geometry is illustrated in figure 2. There are a number of geometric entities which are fixed under this motion, including:

- The (vertical) rotation axis denoted $\mathbf{L}_s$ ("s" for "screw" axis). This is a line of fixed points.
- The plane π_h, and indeed the pencil of horizontal planes. Each plane is fixed as a set.

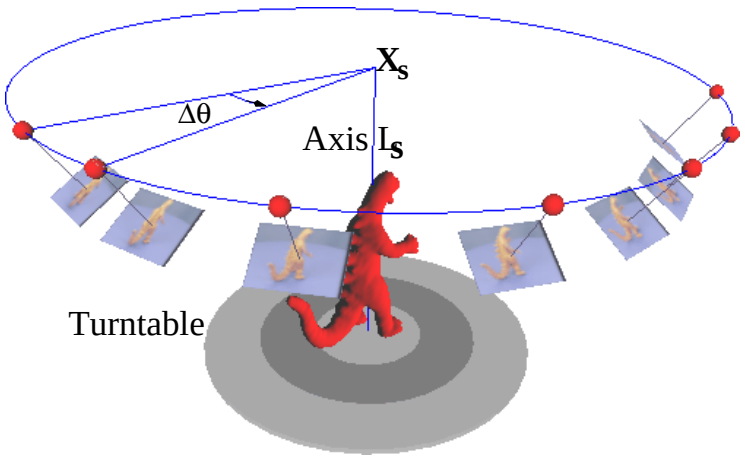


Fig. 2. 3D geometry. The cameras are indicated by their centres (spheres), and image planes. The point $\mathbf{X}_s$ is the intersection of the plane, π_h, containing the camera centres with the rotation axis $\mathbf{L}_s$.

2.2 Image fixed entities

The 3D fixed entities are *sequence invariants* since they are imaged at the same position in every view. Their images include:

- The line $\boldsymbol{l}_s$ which is the image of the rotation axis $\mathbf{L}_s$. Since points on $\mathbf{L}_s$ are fixed under the motion, their images are also fixed under the motion.
- The line $\boldsymbol{l}_h$ in which π_h intersects each image plane. It is the vanishing line of π_h (and indeed of all planes parallel to π_h).

In L. Van Gool and R. Koch, Editors, *Structure and Motion from Multiple Images in Large-Scale Environments*, Lecture Notes in Computer Science, Springer 1998.

– The point $\boldsymbol{x}_s$ which is the image of the fixed point $\mathbf{X}_s$.
– The point $\boldsymbol{v}$ which is the vanishing point of the rotation axis.

These sequence invariants are illustrated schematically in Figure 3a, and the two fixed lines are illustrated in Figure 4 on a real sequence.

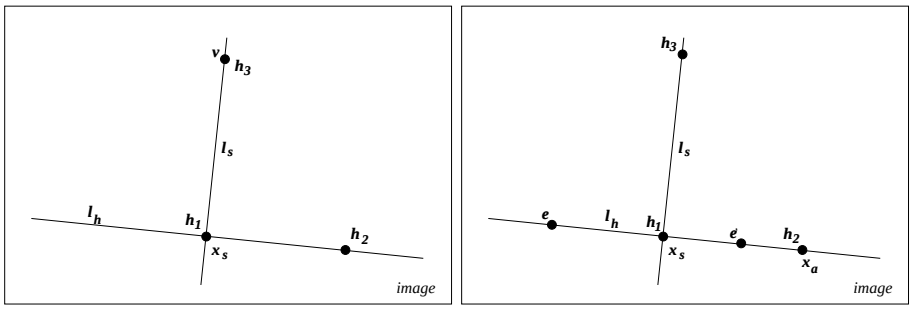


Fig. 3. (a) **Fixed image entities** over the sequence, and their relation to the columns $\boldsymbol{h}_i$ of H. (b) **Two-view entities**. The entities which can be determined from F and their relation to the columns of H. The symmetric part of F is a degenerate conic consisting of the two lines $\boldsymbol{l}_s$ and $\boldsymbol{l}_h$. The anti- symmetric part is represented by the point $\boldsymbol{x}_a$. Points $\boldsymbol{x}_s$ and $\boldsymbol{x}_a$ have fixed position over all view pairs. The position of the epipoles depends on the angle of rotation $\Delta\theta_i$ between views.

2.3 Camera matrices

We have the freedom to choose the world coordinate system so that the rotation axis is aligned with the world Z axis, and the first camera centre is at position $\boldsymbol{t}$ on the X axis. Thus the first camera may be written

$$\mathtt{P}_0 = \mathtt{H}[\mathtt{I} \mid \boldsymbol{t}]$$

where H is a homography representing the camera internal parameters and rotation about the camera centre, and $\boldsymbol{t} = (t, 0, 0)^\top$. A rotation of the camera by θ about the Z axis is achieved by post-multiplying $\mathtt{P}_0$ by

$$\begin{bmatrix} R_Z(\theta) & \mathbf{0} \\ \mathbf{0}^\top & 1 \end{bmatrix}$$

yielding the camera $\mathtt{P}_\theta = \mathtt{H}[R_Z(\theta) \mid \boldsymbol{t}]$. In detail, with $\boldsymbol{h}_i$ the columns of H:

$$\mathtt{P}_\theta = \begin{bmatrix} & & \\ \boldsymbol{h}_1 & \boldsymbol{h}_2 & \boldsymbol{h}_3 \\ & & \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 & t \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{1}$$

Fig. 4. Fixed image lines. The fixed image lines are shown overlaid on images from the head sequence. The almost vertical line is $\boldsymbol{l}_s$, the horizontal line is $\boldsymbol{l}_h$ (see also figure 3a). The eyebrow of the mannequin, which is approximately coplanar with $\boldsymbol{\pi}_h$, remains tangent to $\boldsymbol{l}_h$ as the object rotates. These fixed lines are automatically computed from the images using the algorithm of §3.

This division of the internal and external parameters means that $\mathtt{H}$ and $\boldsymbol{t}$ are fixed over the sequence, only the angle of rotation, θ_i, about the $\mathtt{Z}$ axis varies for each camera $\mathtt{P}_i$. Given this parametrization, the estimation problem can now be precisely stated: we seek the common matrix $\mathtt{H}$ and the angles θ_i in order to estimate the set of cameras $\mathtt{P}_i$ for the sequence. Thus a total of $8+m$ parameters must be estimated for m views, where 8 is the number of degrees of freedom of the homography $\mathtt{H}$. Note, the magnitude of translation only determines the overall scaling and need not be considered further as we are interested only in a similarity reconstruction. The relative angle between views i and $i+1$ is denoted $\Delta\theta_i$.

We now relate the columns of $\mathtt{H}$ to the fixed image entities:

- $\boldsymbol{x}_s$ is the image of $\mathbf{X}_s = (0,0,0,1)^\top$, so under any $\mathtt{P}_\theta$, $\boldsymbol{x}_s = \mathtt{H}(t,0,0)^\top = t\boldsymbol{h}_1$.
- $\boldsymbol{v}$ is the image of the direction of the world $\mathtt{Z}$ axis $(0,0,1,0)^\top$, giving $\boldsymbol{v} = \boldsymbol{h}_3$.
- $\boldsymbol{l}_s = \boldsymbol{h}_1 \times \boldsymbol{h}_3$.
- $\boldsymbol{l}_h = \boldsymbol{h}_1 \times \boldsymbol{h}_2$.

These relations are shown in figure 3a. To see that $\boldsymbol{l}_s$, the image of the $\mathtt{Z}$ axis, is given by $\boldsymbol{l}_s = \boldsymbol{h}_1 \times \boldsymbol{h}_3$, consider a general point on $\mathtt{Z}$, $(0,0,u,v)^\top$. Its projection by any P_θ is $\mathtt{H}(tv,0,u)^\top = tv\boldsymbol{h}_1 + u\boldsymbol{h}_3$, a point on the line through $\boldsymbol{h}_1$ and $\boldsymbol{h}_3$. Similar consideration of a general point on $\boldsymbol{\pi}_h$ leads to $\boldsymbol{l}_h = \boldsymbol{h}_1 \times \boldsymbol{h}_2$.

The columns of $\mathtt{H}$ are the vanishing points of an orthogonal triad of directions. This triad rotates with the camera such that these vanishing points are related to the fixed entities. $\boldsymbol{h}_2$ is the vanishing point of the direction orthogonal to those corresponding to $\boldsymbol{h}_1$ and $\boldsymbol{h}_3$.

The procedure from here on is to determine the columns of $\mathtt{H}$ from the multiple view tensors $(\mathtt{F}, \mathcal{T})$. We first consider the reconstruction ambiguity, where it will be seen that from the multiple view tensors (i.e. from image measurements alone) $\mathtt{H}$ is not determined uniquely, but is restricted to a two-parameter family.

In L. Van Gool and R. Koch, Editors, *Structure and Motion from Multiple Images in Large-Scale Environments*, Lecture Notes in Computer Science, Springer 1998.

2.4 Reconstruction ambiguity

It is well known [6,11] that if nothing is known of the calibration of 2 or more cameras, nor their relative placement, then the reconstruction of the scene and cameras is determined only up to an arbitrary projective transformation of 3-space. For if $\mathtt{T}$ is any 4×4 invertible matrix, representing a projective transformation of $\mathbb{P}^3$, then replacing points $\mathbf{X}_j$ by $\mathtt{T}\mathbf{X}_j$ and cameras $\mathtt{P}_i$ by $\mathtt{P}_i\mathtt{T}^{-1}$ does not change the image points since $\boldsymbol{x}_{ij} = \mathtt{P}_i\mathbf{X}_j = \mathtt{P}_i\mathtt{T}^{-1}\mathtt{T}\mathbf{X}_j$.

In the case of single axis rotation we know that the cameras $\mathtt{P}_i$ have the restricted form (1), so we may ask the question: suppose we determine a reconstruction with a set of cameras of the form (1), how are these cameras related to the actual cameras?

To answer this question [16], consider the class of transformations $\mathtt{T}$ which preserve the form (1). Suppose we have two reconstructions with sets of cameras $\mathtt{P}_i = \mathtt{H}[R_Z(\theta_i) \mid \boldsymbol{t}]$ and $\mathtt{P}'_i = \mathtt{H}'[R_Z(\theta'_i) \mid \boldsymbol{t}']$ of the correct form. Then, $\mathtt{T}$ is an admissible transformation if the sets of cameras are related as:

$$\mathtt{P}'_i = \mathtt{H}'[R_Z(\theta'_i) \mid \boldsymbol{t}'] = \mathtt{H}[R_Z(\theta_i) \mid \boldsymbol{t}]\,\mathtt{T} \qquad \forall i \tag{2}$$

over at least 3 views (i.e. $m \geq 3$). We require that both $\mathtt{H}$ and $\mathtt{H}'$ are full rank 3×3 matrices independent of θ and θ', and $\boldsymbol{t} = (t, 0, 0)^\top, \boldsymbol{t}' = (t', 0, 0)^\top$. Since we are not concerned with the Euclidean transformation part of the ambiguity, $\mathtt{T}$ may be written as

$$\mathtt{T} = \begin{bmatrix} \mathtt{U} & \mathbf{0} \\ \boldsymbol{a}^\top & 1 \end{bmatrix}$$

where $\mathtt{U}$ is an upper triangular matrix. It can be shown that that (2) has a solution provided: $\theta'_i = \theta_i$ and

$$\mathtt{T} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \alpha & 0 \\ 0 & 0 & \beta & 1 \end{bmatrix} \tag{3}$$

with α and β arbitrary scalars.

This shows: (i) from image measurements alone the camera matrices can be recovered only up to a two parameter ambiguity parametrized by α and β. Note that the angle θ is not ambiguous; (ii) the actual cameras lie in this two parameter family, so the reconstruction is also related to the actual cameras by (3); (iii) the matrix $\mathtt{H}$ is only determined up to this ambiguity. To see this, note that

$$\begin{aligned} \mathtt{P}' &= \begin{bmatrix} & & \\ \boldsymbol{h}_1 & \boldsymbol{h}_2 & \boldsymbol{h}_3 \\ & & \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 & t \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \alpha & 0 \\ 0 & 0 & \beta & 1 \end{bmatrix} \\ &= \begin{bmatrix} & & \\ \boldsymbol{h}_1 & \boldsymbol{h}_2 & \boldsymbol{h}_3 \\ & & \end{bmatrix} \begin{bmatrix} 1 & 0 & \beta t \\ 0 & 1 & 0 \\ 0 & 0 & \alpha \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 & t \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{aligned}$$

This means that the last column of $\mathtt{H}$ can only be determined to within a 2-parameter ambiguity from image measurements alone. We will see this ambiguity arising when computing $\mathtt{H}$ from $\mathtt{F}$ and $\mathcal{T}$ in the following sections, and return in section 2.7 to methods of resolving the ambiguity.

2.5 Two-view geometry

The 2-view geometry of single axis rotation is identical to that of planar motion, for which many of the following properties of $\mathtt{F}$ have been derived [1, 4, 24]. In the planar motion case, however, the axis $\mathbf{L}_s$ varies between view pairs, i.e. it is not fixed over the sequence.

The fundamental matrix may be parametrized in terms of the fixed image lines and an image point $\boldsymbol{x}_a$ as

$$\mathtt{F} = \mu\,[\boldsymbol{x}_a]_\times + \tan\frac{\Delta\theta}{2}\left(\boldsymbol{l}_s\boldsymbol{l}_h{}^\top + \boldsymbol{l}_h\boldsymbol{l}_s{}^\top\right) \qquad \text{with} \quad \boldsymbol{x}_a{}^\top\boldsymbol{l}_h = 0 \tag{4}$$

where the 3-vectors $\boldsymbol{x}_a, \boldsymbol{l}_h, \boldsymbol{l}_s$ are scaled to unit norm.

Once $\mathtt{F}$ is estimated from two view correspondences then the points $\boldsymbol{x}_a$ and $\boldsymbol{x}_s$ and lines $\boldsymbol{l}_s$ and $\boldsymbol{l}_h$ are known. Their relation to $\mathtt{H}$ is shown in figure 3b, and also can be read off from the expression for $\mathtt{F}$ in terms of $\mathtt{H}$ and $\Delta\theta$:

$$\mathtt{F} = [\boldsymbol{h}_2]_\times - \frac{1}{(\det\mathtt{H})}\tan\frac{\Delta\theta}{2}\left((\boldsymbol{h}_1\times\boldsymbol{h}_3)(\boldsymbol{h}_1\times\boldsymbol{h}_2)^\top + (\boldsymbol{h}_1\times\boldsymbol{h}_2)(\boldsymbol{h}_1\times\boldsymbol{h}_3)^\top\right)$$

Taking account of the unknown scaling of the homogeneous 3-vectors, the columns of $\mathtt{H}$ are determined from $\mathtt{F}$ (i.e. the 2-view geometry), to within the 3-parameter family

$$\mathtt{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3] = [\boldsymbol{x}_s, \mu\boldsymbol{x}_a, \nu\boldsymbol{x}_s + \omega\boldsymbol{d}] \tag{5}$$

parametrized by the as yet undetermined scalars μ, ν, and ω, where $\boldsymbol{d}$ is an (arbitrary) point on $\boldsymbol{l}_s$, which may be chosen as $\boldsymbol{d} = \boldsymbol{l}_s \times (0,0,1)^\top$. In detail the columns are determined by the following procedure:

1. Extract $\boldsymbol{x}_a$ from the antisymmetric part of $\mathtt{F}$, $\mathtt{F} - \mathtt{F}^\top = [\boldsymbol{x}_a]_\times$.
2. Extract epipoles $\boldsymbol{e}$ and $\boldsymbol{e}'$, and compute $\boldsymbol{l}_h = \boldsymbol{e}\times\boldsymbol{e}'$.
3. Compute $\boldsymbol{l}_s = (2\boldsymbol{l}_h{}^\top\boldsymbol{l}_h\mathtt{I} - \boldsymbol{l}_h\boldsymbol{l}_h{}^\top)(\mathtt{F} + \mathtt{F}^\top)\boldsymbol{l}_h$.
4. Compute $\boldsymbol{x}_s = \boldsymbol{l}_s \times \boldsymbol{l}_h$.
5. Set $\mathtt{H}$ according to (5).

Although the ratio k_i, where $\mu = k_i \tan\frac{\Delta\theta_i}{2}$ may be computed from $\mathtt{F}$, the value of μ is unknown. This means that $\Delta\theta_i$ cannot be computed from two views.

2.6 Three view geometry

From three views, it can be shown that the trifocal tensor may be written as a pencil of tensors, parametrized by μ:

$$\mathcal{T} = \mu^2\mathcal{K} + \mathcal{K}'$$

In L. Van Gool and R. Koch, Editors, *Structure and Motion from Multiple Images in Large-Scale Environments*, Lecture Notes in Computer Science, Springer 1998.

where the elements of the tensors $\mathcal{K}$ and $\mathcal{K}'$ are computed from the two-view quantities k_i and $\mathtt{H}$. Thus, one three-view point correspondence allows μ, and hence the $\Delta\theta_i$, to be recovered uniquely. The only remaining ambiguity is in the third column $\boldsymbol{h}_3$ of $\mathtt{H}$. As shown in section 2.4 this ambiguity cannot be reduced further by the single axis motion constraint alone.

2.7 Removing the reconstruction ambiguity

The reconstruction ambiguity of (3) is the following [25]: metric structure is recovered in planes perpendicular to the axis of rotation; there is an unknown 1D projective transformation along the axis. The ambiguity may be written:

$$\begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \end{pmatrix} \rightarrow \begin{pmatrix} \mathrm{X}/(\beta'\mathrm{Z}+1) \\ \mathrm{Y}/(\beta'\mathrm{Z}+1) \\ \alpha'\mathrm{Z}/(\beta'\mathrm{Z}+1) \end{pmatrix}$$

Note that since metric structure is determined in planes perpendicular to the axis, the angle of rotation between views is known. Figure 5 illustrates this projective ambiguity.

Fig. 5. Projective ambiguity: With no information about the camera or scene, there is a 1D projective ambiguity in the z direction. Five models of the cup with different choices for $\boldsymbol{h}_3$.

To this point no information on the internal calibration of the camera, or on the 3D shape of the object has been used. Internal constraints are provided, for example, by that fact that the image pixels have zero-skew, and known aspect ratio. Often the zero-skew constraint is not useful in practice because it does not resolve the ambiguity [26]. For example, if the image plane is parallel to the rotation axis then all members in the family of solutions for the calibration matrix will already satisfy the zero-skew constraint, so it does not provide any additional information.

It can be shown that specifying the aspect ratio places a quartic constraint on the parameters α, β.

The easiest method of resolution is to use a vanishing point in the scene to identify the plane at infinity (we already have the vanishing line of $\boldsymbol{\pi}_h$), for example by identifying two or more parallel scene lines. This determines $\boldsymbol{h}_3$ up to scale (i.e. the ratio $\alpha : \beta$), and the only remaining ambiguity is then a relative

In L. Van Gool and R. Koch, Editors, *Structure and Motion from Multiple Images in Large-Scale Environments*, Lecture Notes in Computer Science, Springer 1998.

scaling of the Z and plane directions:

$$\begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \end{pmatrix} \rightarrow \begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ \alpha'' \mathrm{Z} \end{pmatrix}$$

Given $\boldsymbol{h}_3$ up to scale, the internal aspect ratio then determines α and β uniquely (up to sign). Alternatively, the aspect ratio of the object can be used to resolve the ambiguity.

3 Estimation of camera matrices

This section describes the implementation of the algebra developed in the previous section. From a raw input sequence we wish to compute the P matrices and 3D point structure. We first summarize the algebraic procedure of the previous section, with the estimation steps then described in more detail below.

3.1 Algorithm summary

Robust point tracks are computed *a priori* using our general-motion trifocal tensor based system [7].

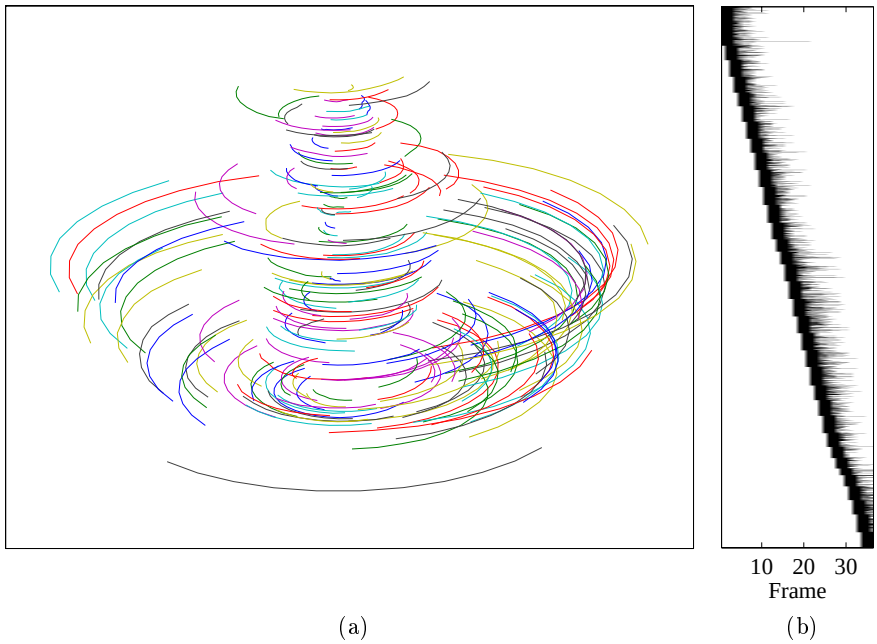
1. For each pair of views fit the planar-motion fundamental matrix (eq. 4).
2. From one of the Fs determine H up to a 3-parameter ambiguity.[1]
3. From each $\mathcal{T}_i$ determine μ and the two angles $\Delta\theta_i$ and $\Delta\theta_{i+1}$.
4. Average μ over the sequence, and angles from overlapping triplets.
5. Bundle adjust, varying H, θ_i and 3D points $\mathbf{X}_j$ to minimize reprojection error $\sum_{i,j} d^2(\boldsymbol{x}_{ij}, \mathtt{H}[R_Z(\theta_i)|\boldsymbol{t}]\mathbf{X}_j)$.

3.2 Point tracking

This is achieved by tracking interest points (Harris corners [9]) through the sequence. Tracking is easily achieved by our current general motion system [3, 7], based on the trifocal tensor. This functionality is used unchanged in the current system, although some speed improvements would certainly accrue if this process were also modified to make use of the specialized geometry. Example point tracks and track lifetimes are shown in figure 6. Typically, about 150 points are tracked in each image triplet, with 2000–3000 points appearing through a sequence.

[1] In the special case where the between-view angles $\Delta\theta_i$ are known to be identical, F is estimated from all 2-view correspondences (typically thousands). Then H is extracted from this F. Similarly $\mathcal{T}$ is fitted to all triplets.

In L. Van Gool and R. Koch, Editors, *Structure and Motion from Multiple Images in Large-Scale Environments*, Lecture Notes in Computer Science, Springer 1998.



(a) (b)

Fig. 6. (a) **Point tracks**: Some point tracks from the dinosaur sequence. For clarity, only the 200 tracks which survived for longer than 7 successive views are shown. In total, 3070 points were tracked for 3 or more views. (b) **Track lifetimes** for dinosaur sequence: Each horizontal bar corresponds to a single point track, extending from the first to last frame in which the point was seen. The measurement matrix is relatively sparse, and few points survived longer than 15 frames.

3.3 F estimation

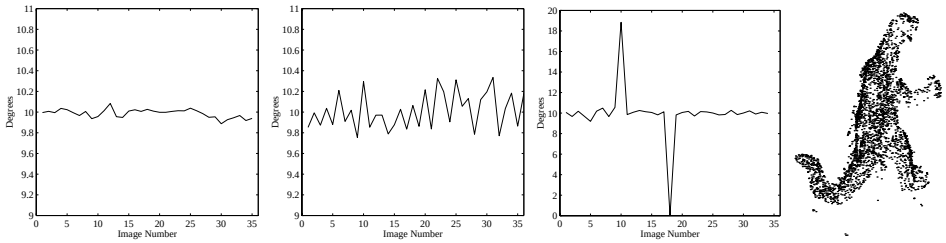
The fundamental matrix is estimated by first fitting a general-motion F to the points. Then the symmetric part of F is truncated to rank 2, and decomposed to recover $\boldsymbol{l}_s$ and the epipoles. This provides a starting point for the special parametrized form (4), which is fitted by minimizing the distance of points to epipolar lines. The average number of point matches per view pair varied from 137 for the Head sequence to 399 for the Dinosaur. The average distance from points to epipolar lines is about 0.3 pixels.

3.4 $\mathcal{T}$ estimation

The trifocal tensor is used only to determine μ from three views. From the special-form fundamental matrices for the views, the two tensors $\mathcal{K}$ and $\mathcal{K}'$ are computed. Then single point correspondences provide candidates for μ. The median of the candidates yields the estimate of μ for the triplet.

3.5 Bundle adjustment

The two and three view geometry provides an (excellent) initial estimate for the camera matrices. In order to determine the maximum likelihood estimate, we assume that errors in the positions of the 2D points are normally distributed. An optimal estimate is then obtained by nonlinear minimization of the distances between the reprojected 3D points and the 2D corners [10]. Typical results for geometry estimation are shown in figure 7. These results are of comparable quality with those of [22] where the camera matrices were determined using a calibration pattern. Convergence is generally achieved in 8 iterations, reducing the RMS reprojection error from 0.3 pixels to 0.1 pixels. For 2000 points, compute time per iteration is of the order of 10 seconds on a 300MHz UltraSparc. The radius of convergence is large, the correct minimum being achieved from initial estimates where the θ_i are in error by up to a factor of 2, although of course many more iterations (about 100) are required.



(a) Dinosaur (b) Head (c) Freiburg (d)

Fig. 7. Geometry estimation. The graphs show the recovered angles between successive views for each of three sequences. (a) Object rotated by a mechanical turntable with a resolution of 1 millidegree. The RMS difference between the angle recovered by our algorithm and the nominal value is 40 millidegrees. This demonstrates the accuracy of the angle recovery. (b) (c) Turn-table rotated by hand. The angle increment is irregular and unknown *a priori*. Variation is up to 20° due to missing and repeated views. (d) 3D points for dinosaur sequence.

4 Space carving and surface rendering

The object is computed as the intersection of the outline cones back-projected from all views. The outline in each image is determined by blue-screening. The surface of the object is determined very efficiently by an octree based algorithm.

Octree Growing The octree is initialised as a cube bounding the object, and is recursively subdivided to determine the surface. Each cube has one of three

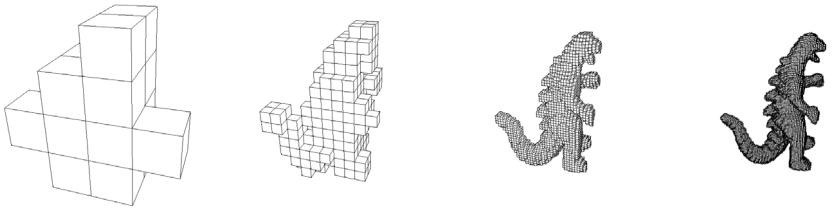


Fig. 8. Octree generation: the dinosaur octree is grown from a single bounding box. The images above show the octree after (*left* to *right*) 3, 5, 7 and 8 subdivisions, given 36 images of the dinosaur.

labels [21] depending on whether it lies entirely inside; entirely outside; or partially intersects the surface. The former two cases are not of interest, and the nodes are not subdivided. The subdividing is stopped at a preset depth. The label of a cube is determined by successively projecting it into each image in the sequence. An example of the octree "surface" developing is shown in figure 8.

Surface Generation The standard marching cubes algorithm[15] provides an initial consistent surface which is then smoothed using a localised surface decimation algorithm. Examples are shown in figures 9 through 11.

5 Conclusions

This paper has demonstrated that uncalibrated structure recovery systems based on the single-axis motion constraint can produce models of equivalent quality to fully calibrated systems, making *a-priori* calibration and expensive control of the viewing environment unnecessary.

Also of interest are the results of volume intersection as a means of producing fully 3D models of arbitrary topologies. Although the "visual hull" effect[14] might be expected to severely limit the range of models that can be acquired, the dinosaur and cup experiments (see especially Figure 11) show that surprisingly complex models can be acquired. However, it is on the model acquisition phase that most plans for future work are centred—given the excellent camera geometry, more advanced techniques [13] can be applied. Particularly, correlation of the surface texture is expected to allow true super-resolution texture mapping, and simultaneously get "inside" the visual hull.

Acknowledgements We are grateful for permission to use the dinosaur sequence supplied by the University of Hannover, and for financial support from EU ACTS Project Vanguard.

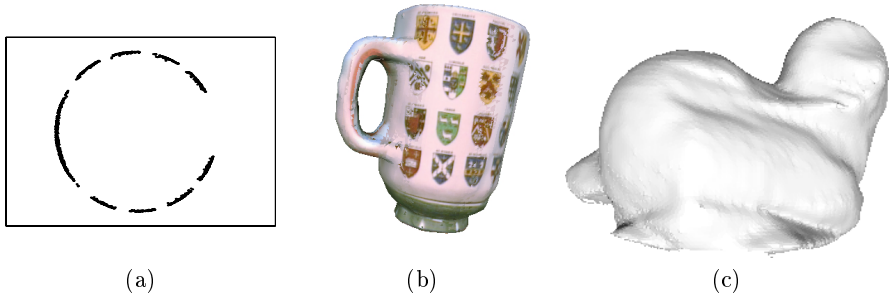
In L. Van Gool and R. Koch, Editors, *Structure and Motion from Multiple Images in Large-Scale Environments*, Lecture Notes in Computer Science, Springer 1998.

References

1. M. Armstrong, A. Zisserman, and R. Hartley. Self-calibration from image triplets. In *Proc. ECCV*, LNCS 1064/5, pages 3–16. Springer-Verlag, 1996.
2. B. G. Baumgart. *Geometric Modelling for Computer Vision*. PhD thesis, Standford University, Palo Alto, 1974.
3. P. Beardsley, P. Torr, and A. Zisserman. 3D model acquisition from extended image sequences. In *Proc. ECCV*, LNCS 1064/1065, pages 683–695. Springer-Verlag, 1996.
4. P. Beardsley and A. Zisserman. Affine calibration of mobile vehicles. In Mohr, R. and Chengke, W., editors, *Europe-China workshop on Geometrical Modelling and Invariants for Computer Vision*, pages 214–221. Xidan University Press, Xi'an, China, 1995.
5. E. Boyer. Object models from contour sequences. In *Proc. ECCV*, pages 109–118, 1996.
6. O. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig? In *Proc. ECCV*, LNCS 588, pages 563–578. Springer-Verlag, 1992.
7. A. W. Fitzgibbon and A. Zisserman. Automatic camera recovery for closed or open image sequences. In *Proc. ECCV*, pages 311–326, 1998.
8. C. J. Harris. Structure-from-motion under orthographic projection. In *Proc. ECCV*, pages 118–123, 1990.

Fig. 9. Texture-mapped dinosaur model. From 36 input images, a 256^3 resolution volumentric model was generated containing 34752 triangles.

In L. Van Gool and R. Koch, Editors, *Structure and Motion from Multiple Images in Large-Scale Environments*, Lecture Notes in Computer Science, Springer 1998.



(a) (b) (c)

Fig. 10. (a) Top view of reconstructed cup points (no points were detected on the handle). RMS difference from a fitted cylinder is 0.004 of the diameter. (b) Texture-mapped cup model. (c) Shaded Freiburg model. The visual hull effect is apparent here, with too few views to penetrate to the object surface.

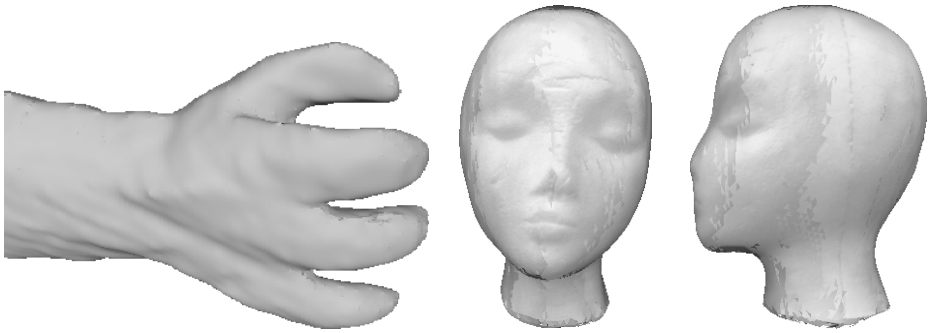


Fig. 11. Closeup: High-resolution model of dinosaur hand, showing the fine detail recoverable using volume intersection. **Head**: Texture-mapped model. The shades of grey indicate the view from which each texture was taken—for each triangle, the view in which it has largest visible area is chosen as the texture source.

9. C. J. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conf.*, pages 147–151, 1988.
10. R. I. Hartley. Euclidean reconstruction from uncalibrated views. In J. Mundy, A. Zisserman, and D. Forsyth, editors, *Applications of Invariance in Computer Vision*, LNCS 825, pages 237–256. Springer-Verlag, 1994.
11. R. I. Hartley, R. Gupta, and T. Chang. Stereo from uncalibrated cameras. In *Proc. CVPR*, 1992.
12. S. B. Kang. Quasi-euclidean recovery from unknown but complete orbital motion. Technical Report CRL 97/10, Digital CRL, 1997.
13. K. N. Kutalakos and S. M. Seitz. A theory of shape by space carving. Technical Report CSTR 692, University of Rochester, 1998.
14. A. Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE T-PAMI*, 16(2):150–162, Feb 1994.
15. W. Lorensen and H. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM Computer Graphocs*, 21(24):163–169, July 1987.
16. J. Mundy and A. Zisserman. Repeated structures: Image correspondence constraints and ambiguity of 3D reconstruction. In J. Mundy, A. Zisserman, and D. Forsyth, editors, *Applications of invariance in computer vision*, pages 89–106. Springer-Verlag, 1994.
17. W. Niem and R. Buschmann. Automatic modelling of 3d natural objects from multiple views. In *European Workshop on Combined Real and Synthetic Image Processing for Broadcast and Video Production, Hamburg, Germany*, 1994.
18. L. S. Shapiro, A. Zisserman, and M. Brady. 3d motion recovery via affine epipolar geometry. *IJCV*, 16(2):147–182, 1995.
19. S. Sullivan and J. Ponce. Automatic model construction, pose estimation, and object recognition from photographs using triangular splines. In *Proc. ICCV*, 1998.
20. R. Szeliski. Shape from rotation. In *Proc. CVPR*, pages 625–630, 1991.
21. R. Szeliski. Rapid octree construction from image sequences. *CVGIP*, 58(1):23–32, July 1993.
22. R. Szeliski and S. B. Kang. Recovering 3D shape and motion from image streams using non-linear least squares. Technical Report CRL 93/3, DEC Cambridge Research Lab, Mar 1993.
23. C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization approach. *IJCV*, 9(2):137–154, November 1992.
24. T. Vieville and D. Lingrand. Using singular displacements for uncalibrated monocular vision systems. Technical Report 2678, I.N.R.I.A., 1995.
25. A. Zisserman, P. Beardsley, and I. Reid. Metric calibration of a stereo rig. In *IEEE Workshop on Representation of Visual Scenes, Boston*, pages 93–100, 1995.
26. A. Zisserman, D. Liebowitz, and M. Armstrong. Resolving ambiguities in autocalibration. *Phil. Trans. R. Soc. Lond. A*, 356:1193–1211, 1998.

In L. Van Gool and R. Koch, Editors, *Structure and Motion from Multiple Images in Large-Scale Environments*, Lecture Notes in Computer Science, Springer 1998.