A Six Point Solution for Structure and Motion

F. Schaffalitzky[1], A. Zisserman[1], and R. I. Hartley[2]

[1] Robotics Research Group, Oxford, UK
[2] G.E. CRD, Schenectady, NY

Abstract. This paper has three main contributions: (1) a "quasi-linear" method for computing structure and motion for $m \geq 3$ views of 6 points; (2) a "quasi-linear" method for computing consistent estimates of the multi-view tensors (fundamental matrix, trifocal tensor and quadrifocal tensor) from n image points; (3) an m view n point robust reconstruction algorithm which uses the 6 point method as a search engine. A minor point is that (1) enables a more concise algorithm, than any given previously, for the reconstruction of 3 views of 6 points.
The new algorithms are evaluated on synthetic and real image sequences, and compared to optimal estimation results (bundle adjustment).

1 Introduction

A large number of methods exist for obtaining 3D structure and motion from correspondences tracked through image sequences. Their characteristics vary from the so-called *minimal* methods [**?**,**?**,**?**] which work with the least data necessary to compute structure and motion, through intermediate methods [**?**,**?**] which may perform mis-match (outlier) rejection as well, to the full-bore *bundle adjustment*.
The minimal solutions are used as search engines in robust estimation algorithms which automatically compute correspondences and tensors over multiple views. For example, the 2 view 7 point solution is used in the RANSAC estimation of the fundamental matrix in [**?**], and the 3 view 6 point solution in the RANSAC estimation of the trifocal tensor in [**?**]. It would seem natural then to use a minimal solution as a search engine in 4 or more views. The problem is that in 4 or more views a solution is forced to include a minimization to account for measurement error (noise). In the '2 view 7 point' and '3 view 6 point' cases there are the same number of measurement constraints as degrees of freedom in the tensor. In both cases 1 or 3 real solutions result (and the duality explanation for this equivalence was given by [**?**]). However, in four views six points provide one more constraint than the number of degrees of freedom in the four view geometry (the quadrifocal tensor). This means than unlike in the two and three view cases where a tensor can be computed which exactly relates the measured points (and also satisfies its internal constraints), this is not possible in the four view case. Instead it is necessary to minimize a measurement error whether algebraic or geometric. The poor estimate which results by using an approach based on minimizing algebraic distance and a standard projective basis for the image is described and demonstrated in section 2.

Here we develop a novel quasi-linear solution for the 6 point $m \geq 3$ case. This solution involves only a SVD and the evaluation of a cubic polynomial in a single variable. This is described in section 3. We also describe a sub-optimal (compared to bundle-adjustment) which minimizes geometric error at the cost of only a 3 parameter minimization.

1.1 Reconstruction for an image sequence

A second part of the paper describes yet another algorithm for computing a reconstruction of cameras and 3D scene points from a sequence of images. The objectives of such algorithms are now well established:

1. **Minimize reprojection error**. A common statistical noise model assumes that measurement error is isotropic and Gaussian in the image. The Maximum Likelihood Estimate in this case involves minimizing the total squared reprojection error over the cameras and 3D points. This is bundle-adjustment.
2. **Cope with missing data**. Structure-from-motion data often arises from tracking features through image sequences and any one track may persist only in few of the total frames.
3. **Cope with mis-matches**. Appearance-based tracking can produce tracks of non-features. A common example is a T-junction which generates a strong corner, but whose pre-image moves slowly between frames.

Bundle adjustment [**?**] is the most accurate and theoretically best justified technique. It can cope with missing data and, with suitable robust statistical cost function, can cope with mis-matches. However, it is expensive to carry out and most significantly requires a good initial estimate.
In the special case of affine cameras, factorization methods [**?**] minimize reprojection error [**?**] and so give the optimal solution found by bundle adjustment. However, factorization cannot cope with mis-matches, and methods to overcome missing data [**?**] lose the optimality of the solution.
In the general case of perspective projection iterative factorization methods have been successfully developed and have recently proved to produce excellent results [**?**,**?**]. The problems of missing data and mis-matches remain though.
Bundle-adjustment will almost always be the final step of a reconstruction algorithm. However, achieving good sub-optimal estimates prior to bundle-adjustment is necessary for the latter to be effective (fewer iterations, and less likely to converge to local minimum.) For practical (in particular automated) applications, mismatches present a real problem. There exist effective methods for estimating structure and motion from data with mismatches for two [**?**] and three [**?**] views (based on RANSAC) and [**?**] based on LMS. These have been put to effective use [**?**] to compute structure and motion by starting from (very reliably) estimated three-view structures and hierarchically coalescing these into sub-sequences of the whole sequence. For four views there is the method in [**?**] for computing the quadrifocal tensor.

Current methods of initializating a bundle-adjustment include factorization [**?**], awf-segments [**?**], duality [**?**,**?**] and the Variable State Dimension Filter (VSDF) [**?**].
In this paper we describe a novel algorithm for computing a reconstruction satisfying the 3 basic objectives above (optimal, missing data, mismatches). It is based on using the 6-pt algorithm as a robust search engine, and is described in section 5.

1.2 Notation

The *standard basis* will refer to the five points in $\mathbb{P}^3$ whose homogeneous coordinates are :

$$\mathbf{E}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \mathbf{E}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \mathbf{E}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \mathbf{E}_4 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \mathbf{E}_5 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

For a 3-vector $\mathbf{v} = (x, y, z)^\top$, we use $[\mathbf{v}]_\times$ to denote the 3×3 skew matrix such that $[\mathbf{v}]_\times \mathbf{u} = \mathbf{v} \times \mathbf{u}$, where $\times$ denotes the vector cross product. For three points in the plane, represented in homogeneous coordinates by $\mathbf{x}, \mathbf{y}, \mathbf{z}$, the incidence relation of collinearity is the vanishing of the bracket $[\mathbf{x}, \mathbf{y}, \mathbf{z}]$ which denotes the determinant of the 3×3 matrix whose columns are $\mathbf{x}, \mathbf{y}, \mathbf{z}$. It equals $\mathbf{x} \cdot (\mathbf{y} \times \mathbf{z})$ where $\cdot$ is the vector dot product.

2 Linear estimation using a duality solution

A method suggested by Carlsson and Weinshall for reconstruction from three views involves a certain duality between points and cameras. In particular, one chooses a projective basis in each image such that the first four points are

$$\mathbf{e}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad \mathbf{e}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \mathbf{e}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad \mathbf{e}_4 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

If in addition one assumes that the corresponding 3D points are $\mathbf{E}_1, \ldots, \mathbf{E}_4$, then the camera matrix may be seen to be of the form

$$\mathtt{P} = \begin{bmatrix} a_i & & -d_i \\ & b_i & -d_i \\ & & c_i \; -d_i \end{bmatrix} \tag{1}$$

Such a camera matrix is called a *reduced camera matrix*. Now, if $\mathbf{X} = (\mathrm{X}, \mathrm{Y}, \mathrm{Z}, \mathrm{T})^\top$ is a 3D point, then one verifies that

$$\begin{bmatrix} a & & -d \\ & b & -d \\ & & c \; -d \end{bmatrix} \begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \\ \mathrm{T} \end{pmatrix} = \begin{bmatrix} \mathrm{X} & & -\mathrm{T} \\ & \mathrm{Y} & -\mathrm{T} \\ & & \mathrm{Z} \; -\mathrm{T} \end{bmatrix} \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \tag{2}$$

Note that the rôles of point and camera are swapped in this last equation. This observation allows us to apply the algorithm for projective reconstruction from two views of many points to solve for six point in many views. The general idea is as follows.

1. Apply a transformation to each image so that the first four points are mapped to the points $\mathbf{e}_i$ of a canonical image basis.
2. The two other points in each view are also transformed by these mappings - a total of two points in each image. Swap the roles of points and views to consider this as a set of two views of several points.
3. Use a projective reconstruction algorithm (based on the fundamental matrix) to solve the two-view reconstruction problem.
4. Swap back the points and camera coordinates as in (2).
5. Transform back to the original image coordinate frame.

The main difficulty with this method is the distortion of the image measurement error distributions by the projective image mapping as illustrated in figure 1. A circular Gaussian distribution is transformed by

Andrew to draw

Fig. 1. Figure to illustrate this - az to draw - which shows that minimizing geometric error (as algebraic error minimization tries to approximate this) in very projectively transformed space pulls back to point away from ellipse centre.

a projective transformation to a distribution that is no longer circular, and not even Gaussian. Common methods of two-view reconstruction are not able to handle such error distributions effectively. One may work very hard to find a solution with minimal residual error with respect to the transformed image coordinates only to find that these errors become very large when the image points are transformed back to the original coordinate system. This is illustrated in figure 2. The method used for reconstruction from the transformed data was a dualization of one of the best methods available for two-view reconstruction ([**?**]) – an iterative method that minimizes algebraic error.

3 Reconstruction from 6 points over m views

This section describes the main algebraic development of the 6 point method. In essence it is quite similar to the development given by Hartley [**?**] and Quan [**?**] for a reconstruction of 6 points from 3 views. The difference is that Quan used a standard projective basis for both the

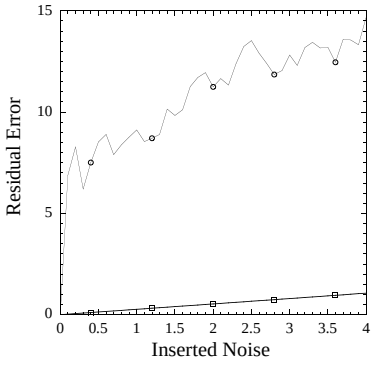

Fig. 2. Residual error as a function of image noise using the duality-based reconstruction algorithm for six points. The results are the average of 100 runs at each error level for randomly chosen synthetic scenes. As may be seen the residual error is extremely high, even for quite low noise levels. It is evident that this method is unusable. In fact the results prove to be unsatisfactory for initializing a bundle adjustment in the original coordinate system.

image and world points, whereas here the image coordinates are not transformed. As described in section 2 the use of a standard basis in the image severely distorts the error that is minimized. The numerical results that follow demonstrate that the method described here produces a near optimal solution.

In the following it will be assumed that we have 6 image points $\mathbf{x}_i$in correspondence over m views. The idea then is to compute cameras for each view such that the scene points $\mathbf{X}_i$ project exactly to their image $\mathbf{x}_i$ for the first five points. Any error minimization required is then restricted to the sixth point in the first instance.

3.1 Pencils of cameras

Each correspondence between a scene point $\mathbf{X}$ and its image $\mathbf{x}$ under a perspective camera $\mathtt{P}$ gives three linear equations for $\mathtt{P}$ whose combined rank is 2. These linear equations are obtained from

$$\mathbf{x} \times \mathtt{P}\mathbf{X} = \mathbf{0} \tag{3}$$

Given only five scene points, assumed to be in general position, it is possible to recover the camera up to a 1-parameter ambiguity. More precisely, the five points generate a linear system of equations for $\mathtt{P}$ which may be written $\mathtt{M}\mathbf{p} = \mathbf{0}$, where $\mathtt{M}$ is a 10×12 matrix formed from two of the linear equations (3) of each point correspondence, and $\mathbf{p}$ is $\mathtt{P}$ written as a 12-vector. This system of equations has a 2-dimensional null-space and thus results in a pencil of cameras.

Suppose that the five world points are the points of the standard projective frame $\mathbf{E}_1, \ldots, \mathbf{E}_5$, so that both $\mathbf{X}_i$ and $\mathbf{x}_i$ $(i = 1, 2, 3, 4, 5)$are now

known. Then the null-space of $\mathtt{M}$ can immediately be computed, and will be denoted from here on by the basis of 3×4 matrices $[\mathtt{A}, \mathtt{B}]$. Then for any choice of the scalars $(s : t) \in \mathbb{P}^1$ the camera in the pencil $\mathtt{P} = s\mathtt{A} + t\mathtt{B}$ exactly projects the standard projective basis to the first five points.
Each camera $\mathtt{P}$ in the pencil has its optical centre located as the null-vector of $\mathtt{P}$ and thus a given pencil of camera gives rise to a 3D curve of possible camera centres. In general (there are degenerate cases) the locus of possible camera centres will be a twisted cubic passing through the five points of the standard projective basis. The five points specify 10 of the 12 degrees of freedom of the twisted cubic, the remaining 2 degrees of freedom are specified by the 2 plane projective invariants of the five image points.

3.2 The quadric constraints

We continue to consider a single camera $\mathtt{P}$ mapping a set of point $\mathbf{X}_1, \ldots, \mathbf{X}_6$ to image points $\mathbf{x}_1, \ldots, \mathbf{x}_6$. Let $[\mathtt{A}, \mathtt{B}]$ be the pencil of cameras consistent with the projections of the first five points. Since $\mathtt{P}$ lies in the pencil, there are scalars $(s : t) \in \mathbb{P}^1$ such that $\mathtt{P} = s\mathtt{A} + t\mathtt{B}$ and so the projection of the sixth world point $\mathbf{X}_6$ is $\mathbf{x}_6 = s\mathtt{A}\mathbf{X}_6 + t\mathtt{B}\mathbf{X}_6$. This means that the three points $\mathbf{x}_6, \mathtt{A}\mathbf{X}_6, \mathtt{B}\mathbf{X}_6$ are collinear in the image, so

$$[\mathbf{x}_6, \mathtt{A}\mathbf{X}_6, \mathtt{B}\mathbf{X}_6] = 0 \ ,$$

which is a quadratic constraint on $\mathbf{X}_6$. Expressing the 3×3 determinant as a triple product gives $(\mathtt{A}\mathbf{X}_6) \cdot (\mathbf{x}_6 \times (\mathtt{B}\mathbf{X}_6)) = 0$, or more neatly ${\mathbf{X}_6}^\top {\mathtt{A}}^\top [\mathbf{x}]_\times \mathtt{B}\mathbf{X}_6 = 0$. To summarize :

Lemma 1. *Let $[\mathtt{A}, \mathtt{B}]$ be the pencil of cameras consistent with the projections of five known points $\mathbf{X}_i$ to image points $\mathbf{x}_i$. Let $\mathbf{x}_6$ be a sixth image point. Then the 3D point $\mathbf{X}_6$ mapping to $\mathbf{x}_6$ must lie on a quadric surface given by*

$$Q = ({\mathtt{A}}^\top [\mathbf{x}_6]_\times \mathtt{B}) sym = ({\mathtt{A}}^\top [\mathbf{x}_6]_\times \mathtt{B}) + ({\mathtt{A}}^\top [\mathbf{x}_6]_\times \mathtt{B})^\top \ .$$

In addition, the known points $\mathbf{X}_1, \ldots, \mathbf{X}_5$ also lie on Q.

One may easily verify that this quadric is unchanged if one replaces either of $\mathtt{A}$ or $\mathtt{B}$ by a linear combination, and hence depends on the pencil only, and not its particular representatives $\mathtt{A}$ and $\mathtt{B}$. It has not yet been shown that the points $\mathbf{X}_1, \ldots, \mathbf{X}_5$ lie on Q. Note however that $\mathtt{A}\mathbf{X}_i = \mathbf{x}_i = \mathtt{B}\mathbf{X}_i$ for $i = 1, \ldots, 5$, and so

$${\mathbf{X}_i}^\top Q \mathbf{X}_i = {\mathbf{X}_i}^\top {\mathtt{A}}^\top [\mathbf{x}_6]_\times \mathtt{B}\mathbf{X}_i = {\mathbf{x}_i}^\top [\mathbf{x}_6]_\times \mathbf{x}_i = 0$$

as required. The last equality holds since $\mathbf{x}_6$ is skew-symmetric. In the particular case where the five points $\mathbf{X}_i$ are the members $\mathbf{E}_i$ of a projective basis, the condition ${\mathbf{X}_i}^\top Q \mathbf{X}_i = 0$ allows us to specify the form of Q simply. From ${\mathbf{E}_i}^\top Q \mathbf{E}_i = 0$ for $i = 1, \ldots, 4$, we deduce that the four

diagonals elements of Q vanish. From ${\mathbf{E}_5}^\top Q\mathbf{E}_5$ it follows that the sum of elements of Q is zero. Thus, we may write Q in the following form

$$Q = \begin{bmatrix} 0 & w_1 & w_2 & -\Sigma \\ w_1 & 0 & w_3 & w_4 \\ w_2 & w_3 & 0 & w_5 \\ -\Sigma & w_4 & w_5 & 0 \end{bmatrix} \tag{4}$$

where $\Sigma = w_1 + w_2 + w_3 + w_4 + w_5$. Let $\mathbf{X}_6 = (p, q, r, s)^\top$ be a point lying on Q. The equation ${\mathbf{X}_6}^\top Q\mathbf{X}_6$ may be written in a vector form as

$$(w_1, w_2, w_3, w_4, w_5) \begin{pmatrix} pq - ps \\ pr - ps \\ qr - qs \\ qs - ps \\ rs - ps \end{pmatrix} = 0 \tag{5}$$

or more briefly, $\mathbf{W}^\top \mathbf{A} = 0$, where $\mathbf{A}$ is the column vector in the above equation.

Solving for the point X

Now consider m views of 6 points and suppose again that the first five world points are in known positions $\mathbf{X}_1, \ldots, \mathbf{X}_5$. To compute projective structure it suffices to find the sixth world point $\mathbf{X}_6$. In the manner described above, each view provides a quadric on which $\mathbf{X}_6$ must lie. For two views the two associated quadrics intersect in a curve, and consequently there is a one parameter family of solutions for $\mathbf{X}_6$ in that case. The curve will meet a third quadric in a finite number of points, so 3 views will determine a finite number (namely $2 \times 2 \times 2 = 8$ by Bézout's theorem) of solutions for $\mathbf{X}_6$. However, five of these points are the points $\mathbf{X}_1, \ldots, \mathbf{X}_5$ which must lie on all three quadrics. Thus there are up to three possible solutions for $\mathbf{X}_6$. With more than three views, a single solution will exist, except for critical configurations ([**?**]).
The general strategy for finding $\mathbf{X}_6$ is as follows. We denote the j-th image of the i-th point by $\mathbf{x}_i^j$ and assume that the first five world points are $\mathbf{E}_1, \ldots, \mathbf{E}_5$.

1. Compute $\mathtt{A}^j$ and $\mathtt{B}^j$ that generate the pencil of cameras that map the basis points onto the first five points in image j.
2. Compute the quadric $Q^j = ({\mathtt{A}^j}^\top [\mathbf{x}_6^j]_\times \mathtt{B}^j)_{\mathrm{sym}}$ and extract the vector $\mathbf{W}^j = (w_1^j, \ldots, w_5^j)^\top$ of its independent entries.
3. For each j formulate the equation ${\mathbf{W}^j}^\top \mathbf{A} = 0$ as in (5) and find the least-squares solution to this set of equations. The solution vector

   $$\mathbf{A} = (a, b, c, d, e)^\top = (pq - ps, pr - ps, qr - ps, qs - ps, rs - ps)^\top$$

   has entries that are quadratic in the entries (p, q, r, s) of the point $\mathbf{X}_6$.
4. Solve for (p, q, r, s) from the now known entries (a, b, c, d, e) of $\mathbf{A}$. Details of this last step will be given later.

Note that solving a set of m quadratic equations in the entries of $\mathbf{X}_6$ has been reduced to solving a set of five simple quadratic equations. In more abstract terms there is a map ψ

$$\psi : \mathbf{X} = \begin{pmatrix} p \\ q \\ r \\ s \end{pmatrix} - \rightarrow \begin{pmatrix} a \\ b \\ c \\ d \\ e \end{pmatrix} = \begin{pmatrix} pq - ps \\ pr - ps \\ qr - ps \\ qs - ps \\ rs - ps \end{pmatrix}$$

which is a (rational) transformation from $\mathbb{P}^3$ to $\mathbb{P}^4$, and maps the quadric $Q \subset \mathbb{P}^3$ into the hyperplane

$$w_1 a + w_2 b + w_3 c + w_4 d + w_5 e = 0 \tag{6}$$

where the (known) coefficients w_i are $Q_{12}, Q_{13}, Q_{23}, Q_{24}, Q_{34}$.
The basic method now is to solve for $\mathbf{A} = (a, b, c, d, e)^\top \in \mathbb{P}^4$ by intersecting hyperplanes in $\mathbb{P}^4$, rather than to solve directly for $\mathbf{X} \in \mathbb{P}^3$ by intersecting quadrics in $\mathbb{P}^3$.

Inverting ψ

Having solved for $\mathbf{A} = (a, b, c, d, e)^\top$ we wish to recover $\mathbf{X} = (p, q, r, s)^\top$. By considering ratios of a, b, c, d, e and their differences, various form of solution can be obtained. In particular it can be shown that $\mathbf{X}$ is a right nullvector of the following 6×4 design matrix :

$$\begin{pmatrix} e-d & 0 & 0 & a-b \\ e-c & 0 & a & 0 \\ d-c & b & 0 & 0 \\ 0 & e-b & a-d & 0 \\ 0 & e & 0 & a-c \\ 0 & 0 & d & b-c \end{pmatrix} \tag{7}$$

This will have nullity ≥ 1 in the ideal noise-free case where the point $\mathbf{A} = (a, b, c, d, e)^\top$ really does lie in the range of ψ. When the point $\mathbf{A}$ does not lie exactly in the image of ψ, the matrix may have nullity 0 and more care has to be taken to recover a meaningful $\mathbf{X}$.

A Cubic constraint

The fact that $\dim \mathbb{P}^3 = 3 < 4 = \dim \mathbb{P}^4$ implies that the image of ψ is not all of $\mathbb{P}^4$. In fact the image is the hypersurface $\mathtt{S}$ cut out by the cubic equation

$$S(a, b, c, d, e) = abd - abe + ace - ade - bcd + bde = \begin{vmatrix} e & e & b \\ d & c & b \\ d & a & a \end{vmatrix} = 0 \tag{8}$$

This can be verified by direct substitution. It may alternatively be derived by observing that all 4×4 subdeterminants of (7) must vanish, since

it is rank deficient. These subdeterminants will be quartic algebraic expressions in a, b, c, d, e which are all multiples of the cubic expression S.
The fact that the image $\psi(\mathbf{X})$ of $\mathbf{X}$ must lie on $\mathtt{S}$ introduces the problem of enforcing this constraint ($S = 0$) numerically. This will be dealt with below.

Solving for 3 views of six points

The linear constraints defined by the three hyperplanes (6) cut out a line in $\mathbb{P}^4$. The line intersects $\mathtt{S}$ in three points (generically) (see figure 3.2). Thus there are three solutions for $\mathbf{X}$. This is a well-known [**?**] minimal solution. Our treatment gives a simpler (than the Quan [**?**] or Carlsson and Weinshall [**?**]) algorithm for computing a trifocal tensor from six points (from a projective reconstruction) because it does not involve changing basis in the images. To be specific, from three views one proceeds as follows :

1. From three views one obtains three equations of the form (5) in the five entries of $\mathbf{A}$. Since this is a homogenous set of equations, the scale in immaterial.
2. One may obtain a set of solutions of the form $\mathbf{A} = s\mathbf{A}_1 + t\mathbf{A}_2$ where $\mathbf{A}_1$ and $\mathbf{A}_2$ are generators of the null space of the 3×5 set of equations.
3. By expanding out the constraint (8) one obtains a homogeneous cubic equation in s and t. There will be either one or three real solutions.
4. Once $\mathbf{A}$ is computed that satisfies the cubic constraint (8), one may solve for $\mathbf{X}_6 = (p, q, r, s)^\top$ as the null-space of the matrix (7).

Missing figure

Fig. 3. The diagram shows a line in 3-space intersecting a surface of degree 3. In the case of a a line in 4-space and a hyper-surface of degree 3, the number of intersections is also 3.

Four or more views

In this case the linear constraints from the hyperplanes alone will (generally) determine a unique solution for $\mathbf{A}$. In the presence of noise, though, this solution will not satisfy the cubic constraint. That is, it does not lie on $\mathtt{S}$; its coordinates do not satisfy $S = 0$. We would like to coerce it to do so. The problem is to perform a "manifold projection" in a non-Euclidean space, with the usual associated problem that we don't know in which direction to project. We will now give a novel solution to this problem.
An (over)determined linear system of equations is often solved using Singular Value Decomposition, by taking as null-vector the singular vector

with the smallest singular value. The justification for this is that the SVD elicits the "directions" of space in which the solution is well determined (small singular values) and those in which it is poorly determined (large singular values). Taking the singular vector with smallest singular value is the usual "linear" solution, but as pointed out, it does not in general lie on S. However, there may still be some information left in the second-smallest singular vector, and taking the space spanned by the two smallest singular vectors gives a line in $\mathbb{P}^4$, which passes through the "linear" solution and must also intersect S in three points (S is cubic). We use these three intersections as our candidates for $\mathbf{A}$. Since they lie exactly on S, recovering their preimages $\mathbf{X}$ under ψ is not a problem.
This, then, is our heuristic. We overcome our manifold projection problems by projecting in the direction of the singular vector with second-smallest singular value. Note that in the case of 4 views, the smallest singular value will actually be 0.

Degeneracies

It is worth noting that if the sixth point in 3-space lies on the twisted cubic through the first five basis points then there is a one parameter family of cameras for each view which will exactly project the six space points to their images. This situation can be detected (in principle) because if the space point lies on the twisted cubic then all 6 image points lie on a conic.

3.3 Minimizing reprojection error

The previous sub-section has described a quasi-linear method involving the following two steps: first, a linear SVD decomposition of a matrix composed of one hyperplane from each view; second, intersecting the line in $\mathbb{P}^4$ (computed from two of the singular vectors) with a cubic surface. The best (most accurate) use of the given data is to minimize total squared image reprojection error over all camera and structure parameters, but that amounts to a full bundle adjustment.
In the current case, we have computed cameras which map the first five points exactly to their measured image points, and rather than jump directly to bundle adjustment, an intermediate case is to minimize total squared reprojection error for the sixth point $\mathbf{X}$ over $\mathbb{P}^3$. This fits in the middle of a spectrum of possible estimates :

1. **Algebraic fit**. The quasi-linear solution minimizes an "algebraic" error by a direct least squares fit on homogeneous coordinates in $\mathbb{P}^4$.
2. **Sub-optimal fit**. Minimizes total squared reprojection error for the sixth point over its position in $\mathbb{P}^3$, mapping the first five points exactly (3 degrees of freedom).
3. **Optimal fit (bundle adjustment)**. Minimizes total squared reprojection error for all points, over all structure and camera parameters ($11m + 3$ degrees of freedom).

The model fitted by the second item is clearly a reduced form of the model fitted by the third item. The cost of executing minimization is

negligible (it has only 3 degrees of freedom), which can be seen as follows. To fit a model with a non-linear Levenberg-Marquardt type minimizer, we need to calculate at the current estimate, $\mathbf{X}$, the fitting residuals and the jacobian of these wrt the current estimate. The latter is obtained (if tediously) from the former, so let us concentrate on the fitting residuals. In each image, fitting error is the distance from the reprojected point $\mathbf{y} = \mathtt{P}\mathbf{X}$ to the measured image point $\mathbf{x} = (u, v, 1)^\top$. The reprojected point will depend both on the position of the sixth world point and on the choice of camera in the pencil for that image. But for a given world point $\mathbf{X}$, and choice of camera $\mathtt{P} = s\mathtt{A} + t\mathtt{B}$ in the pencil, the residual is the 2D image vector from $\mathbf{x}$ to the point $\mathbf{y} = \mathtt{P}\mathbf{X} = s\mathtt{A}\mathbf{X} + t\mathtt{B}\mathbf{X}$ on the line $\mathbf{l}$ joining $\mathtt{A}\mathbf{X}$ and $\mathtt{B}\mathbf{X}$. The optimal choice of s, t for given $\mathbf{X}$ is thus easy to deduce; it must be such as to make $\mathbf{y}$ the perpendicular projection of $\mathbf{x}$ onto this line (figure 3.3). What this means is that explicit minimization over camera parameters is unnecessary and so only the 3 degrees of freedom for $\mathbf{X}$ remain.

Missing figure

Fig. 4. Minimizing reprojection in the reduced model. For a given $\mathbf{X}$, the best choice $\mathtt{P} = s\mathtt{A} + t\mathtt{B}$ of camera in the pencil corresponds to the point $y = s\mathtt{A}\mathbf{X} + t\mathtt{B}\mathbf{X}$ on the line closest to the measured image point $\mathbf{x}$. Hence the image residual is the vector joining x and y.

3.4 Approximating geometric error

We now compare the first item with the second. We have already seen that the components of the line $\mathbf{l}(\mathbf{X}) = \mathtt{A}\mathbf{X} \times \mathtt{B}\mathbf{X}$ are expressible as quadrics in $\mathbf{X}$, and moreover as linear functions of $\mathbf{A} = \psi(\mathbf{X})$:

$$\mathbf{l}(\mathbf{X}) = \mathtt{A}\mathbf{X} \times \mathtt{B}\mathbf{X} = \begin{pmatrix} \mathbf{q}_0\psi(\mathbf{X}) \\ \mathbf{q}_1\psi(\mathbf{X}) \\ \mathbf{q}_2\psi(\mathbf{X}) \end{pmatrix} = \begin{pmatrix} \cdots\mathbf{q}_0\cdots \\ \cdots\mathbf{q}_1\cdots \\ \cdots\mathbf{q}_2\cdots \end{pmatrix}\mathbf{A}$$

for some 3×5 matrix with rows $\mathbf{q}_i$ whose coefficients can be determined from those of $\mathtt{A}$ and $\mathtt{B}$. If the sixth image point is $\mathbf{x} = (u, v, 1)^\top$ then the squared residual is

$$d(\mathbf{x}, l(\mathbf{X}))^2 = \frac{\left|u\mathbf{q}_0\mathbf{A} + v\mathbf{q}_1\mathbf{A} + \mathbf{q}_2\mathbf{A}\right|^2}{\left|\mathbf{q}_0\mathbf{A}\right|^2 + \left|\mathbf{q}_1\mathbf{A}\right|^2}$$

and this is the geometric error minimized in the sub-optimal scheme. Note that this form of the error is amenable to reweighted least squares because, given an initial estimate of $\mathbf{X}$, we can adjust the scale so as to make the denominator close to 1, while putting the numerator into a least squares problem. This expression shows that the minimization of image error over $\mathbf{X} \in \mathbb{P}^3$ can be carried out as a minimization over $\mathbf{A} \in \mathtt{S}$ instead.

The algebraic fitting algorithm which we propose consists of first forming the linear least squares problem which minimizes the sum of squares of $\mathbf{q}_2\mathbf{A}$ over the images. We intersect the 2D SVD nullspace with $\mathtt{S}$ to impose constraints.

As we have presented the algorithm so far, there is an arbitrary choice of scale for each quadric $Q_{\mathtt{A},\mathtt{B}}$, corresponding to the arbitrariness in the choice of representation $[\mathtt{A},\mathtt{B}]$ of the pencil of cameras (in terms of the equation above the algebraic fitting scheme neglects the denominator and just minimizes the residuals defined by the $u\mathbf{q}_0 + v\mathbf{q}_1 + \mathbf{q}_2$), the scale of which depends on the scale of $\mathtt{A},\mathtt{B}$. Which normalization is used matters, and we addres that issue now.

Firstly, by translating coordinates, we may assume that the sixth point is at the origin. This amounts to (pre)multiplying $\mathtt{A},\mathtt{B}$ by a 3×3 translation homography and we assume this has been done (so $u, v = 0$ in the above derivation). Thus the geometric error we want to approximate is

$$\frac{|\mathbf{q}_2\mathbf{A}|^2}{|\mathbf{q}_0\mathbf{A}|^2 + |\mathbf{q}_1\mathbf{A}|^2}$$

Making this assumption on the position of the sixth image point means that the normalization is independent of (*ie* is invariant to) translations of image coordinates. It is desirable that the normalization should be invariant to scaling and rotation as well since these are the transformations which preserve our error model (isotropic Gaussian noise). This requirement rules out many obvious candidates, like normalizing the Frobenius norms of $\mathtt{A},\mathtt{B}$ to 1 or normalizing $\mathbf{q}_2$ to unit norm.

To describe our choice of normalization, we introduce a dot product which is similar to the Frobenius inner product $(\mathtt{A},\mathtt{B})_{\text{Frob}} = \mathbf{trace}(\mathtt{A}^\top\mathtt{B})$. The Frobenius inner product can also be computed as the sum of $\mathtt{A}_{ij}\mathtt{B}_{ij}$ over all indices i, j. Our inner product is the same as the Frobenius inner product, except that the last row is left out :

$$\begin{aligned}(\mathtt{A},\mathtt{B})_{\text{Frob}} &= \sum_{\substack{i=0,1,2\\j=0,1,2,3}} A_{ij}B_{ij}\\ (\mathtt{A},\mathtt{B})_{*} &= \sum_{\substack{i=0,1\\j=0,1,2,3}} A_{ij}B_{ij}\end{aligned}$$

The normalization we use can now be described by saying that the choice of basis of the pencil $[\mathtt{A},\mathtt{B}]$ must be an orthonormal basis wrt $(\cdot,\cdot)_*$. To achieve this, one could start with any basis of the pencil and use the Gram-Schmidt algorithm [**?**] to orthonormalize them.

Scaling image coordinates corresponds to scaling the first two rows of the basis element matrices, which just scales our dot product (but not the Frobenius product). Rotating image coordinates corresponds to applying an orthogonal transformation to the first two rows of the basis elements, and this preserves our dot product. Finally, choosing a different orthonormal basis corresponds to a certain linear basis change in the pencil and the effect on the $\mathbf{q}_i$ is a scaling by the determinant of that basis change. But that basis change must be orthogonal, so it has determinant 1.

3.5 Summary and Results I

It has been demonstrated how to pass from m views of six points in the world to a projective reconstruction in a few steps. The positions of the six world points as well as the camera for each view have been computed. The reconstruction obtained is not the MLE (assuming isotropic Gaussian point localization noise), which optimally distributes measurement error over all the points, but an approximation which puts all the errors on the sixth point.

The steps of the algorithm are :

1. Compute, for each image, the pencil of cameras which map the five standard basis points in the world to the first five image points, using the recommended normalization to achieve invariance to image coordinate changes.
2. Form, from each pencil $[\mathtt{A}, \mathtt{B}]$ the quadric constraint on the sixth world point $\mathbf{X}$ as described in section 3.2.
3. Using the transformation $\psi : \mathbb{P}^3 - \rightarrow \mathbb{P}^4$, convert the quadric intersection problem to a hyperplane intersection problem. Use the SVD to compute a pencil of possible values for $\mathbf{A} = \psi(\mathbf{X})$.
4. Intersect that line with the cubic constraint $S = 0$ to get (up to) three solutions for $\mathbf{A} = \psi(\mathbf{X})$ satisfying the constraint.
5. Use (7) to recover values for the sixth point $\mathbf{X}$ from $\mathbf{A}$. Keep the solution (if there are 3) with the lowest residual.
6. (optional) Minimize reprojection error over the 3 degrees of freedom in the position of $\mathbf{X}$.

In practice, for a given set of six points, the quality of reconstruction can vary depending on which point is last in the basis. We try all six in turn and choose the best one.

We will now give results on synthetic and real image sequences of 6 points in m views. The objective is to compare the performance of three algorithms: quasi-linear; minimizing on the 6th point only; and, bundle adjustment. The three performance measures used are reprojection error, registration error to ground truth, and stability (the algorithm converges). The claim is that the quasi-linear algorithm performs as well as the more expensive variants and can safely be used in practice.

Synthetic data

We first show results of testing the algorithm on synthetic data with varying amounts of pixel localisation noise added; our noise model is isotropic Gaussian noise with standard deviation σ. For each value of σ, the algorithm is run on 100 randomly generated data sets. Each data set is produced by choosing six world points at random uniformly in the cube $[-1, +1]^3$ and six cameras with centres between 4 and 5 units from the origin and principcal rays passing through the cube. After projecting each point under each chosen camera, artificial noise is added. The images are 512×512, with square pixels, and the principal point is at the centre of the image.

Figure 5 summarizes the results.

σ (pixels)	failures	rms (pixels)	max (pixels)	registration rms
0.5	1	0.312	1.271	0.007
1.0	8	0.487	1.936	0.011
1.5	11	0.827	3.314	0.013
2.0	12	1.028	4.228	0.024
2.5	21	1.083	4.362	0.023
3.0	20	1.216	4.867	0.028

Missing figure

σ (pixels)	failures	rms (pixels)	max (pixels)	registration rms
0.5	1	0.215	0.902	0.005
1.0	2	0.463	1.926	0.006
1.5	1	0.672	2.686	0.016
2.0	3	0.813	3.434	0.019
2.5	6	0.968	4.040	0.026
3.0	8	1.114	4.640	0.037

Missing figure

σ (pixels)	failures	rms (pixels)	max (pixels)	registration rms
0.5	1	0.127	0.350	0.003
1.0	3	0.266	0.788	0.010
1.5	7	0.382	1.108	0.013
2.0	6	0.478	1.354	0.020
2.5	4	0.605	1.812	0.022
3.0	12	0.730	2.150	0.024

Missing figure

Fig. 5. Summary of experiments on synthetic data. The tables show, for each estimator, the rms and maximum fitting error, averaged over 100 randomly generated data sets (6 views of 6 points). The last column is the average rms registration error (using a homography minimizing sum of squares in the target space) into the ground truth frame. The graphs display the same information graphically.

The "failures" column count the number of reconstructions for which some reprojection error exceeded 10 pixels. A more plausible error model would be isotropic Gaussian error clamped to a circle of radius, say, 2 pixels and indeed, if this modification is made all the failures disappear. The quality of reconstruction degrades gracefully as the noise is turned up from the slightly optimistic 0.5 to the somewhat pessimistic 3.0; the rms and maximum reprojection error are highly correlated, with correlation coefficient 0.999 in each case (which may also be an indicator of graceful degradation).

Real data

The algorithm is tested on an image sequence consisting of 10 colour images (JPEG, 768×1024) of a turntable. The image sequence is shown in 6. Points were entered and matched by hand using a mouse (estimated accuracy is 2 pixels standard deviation). Ground truth is obtained by measuring the turntable with vernier calipers, and is estimated to be accurate to $0.25mm$. There were 9 tracks, all seen in all views. Of course, in principle any 6 tracks could be used to compute a projective reconstruction, but in practice some bases are much better than others. Examples of poor bases include ones which are almost coplanar in the world or which have points very close together.

Missing figure Missing figure Missing figure Missing figure Missing figure Missing figure Missing figure Missing figure Missing figure

Fig. 6. The nine images of the turntable used for the reconstruction.

The algorithms are compared on this sequence. The table in figure 7 compares the reconstructions.

	basis residuals (pixels)	all residuals (pixels)	registration error (mm)
6 points no optimization	0.363 /2.32	0.750/2.32	0.467/0.676
6 points with optimization	0.358 /2.33	0.744/2.33	0.424/0.596
6 points (and cameras) bundled	0.115 /0.476	0.693/2.68	0.405/0.558
All points (and cameras) bundled	0.334 /0.822	0.409/1.08	0.355/0.521

Fig. 7. Results for the turntable images. There are 9 tracks over 10 views. The reconstruction is compared for the three different algorithms. Residuals (reported as rms/max) are shown for the 6 points which formed the basis (first column) and for all reconstructed points taken as a whole (second column). The last row shows the corresponding residuals when full bundle adjustment (optimize over all points and cameras) is applied to the final reconstruction.

Bundle adjustment achieves the smallest reprojection error over all residuals, because it has greater freedom in distributing the error. Our method minimizes error on the sixth point of a six point basis. Thus it is no surprise that the effect of applying bundle adjustment is to increase the error in column 1 and to decrease the error in column 2. These figures support our claim that the quasi-linear method gives a very good approximation to the optimized method.
Figure 8 shows the reprojected reconstruction in the first and fourth views of the sequence.

Missing figure Missing figure

Fig. 8. Reprojected reconstruction in views 0 and 3. The large white dots are the input points, measured from the images alone. The smaller, dark points are the reprojected points. Note that the reprojected points lie very close to the centre of each white dot. The reconstruction is computed with the 6-point algorithm, optimizing over the position of the sixth point.

4 Estimating multi-view tensors

For two views of 7 points there is a well-known method [**?**,**?**] for recovering the fundamental matrix between the two views. Essentially, each point correspondence $\mathbf{x} \leftrightarrow \mathbf{x}'$ between the two views imposes a single linear constraint $\mathbf{x}'^{\top}\mathtt{F}\mathbf{x} = 0$ and so seven points define a pencil of candidates for $\mathtt{F}$. The requirement that $\mathtt{F}$ be singular imposes a cubic constraint on this pencil and so there are up to three solutions. In geometric terms, the (linear) space of 3×3 matrices can be identified with $\mathbb{P}^8$ and the fundamental matrices lie in a subset of this, namely the locus of singular matrices. Singularity is characterized by the vanishing of the determinant $\det(\mathtt{F}) = 0$, so that the locus of fundamental matrices lies on a cubic hypersurface in $\mathbb{P}^8$. This surface has three intersections with the line cut out in $\mathbb{P}^8$ by the 7 hyperplanes obtained from 7 correspondences.
Given many (n more than 7) correspondences the linear constraints alone will determine a solution, but as before, in the presence of noise, that solution will not satisfy the constraint, i.e. it will not lie on the cubic hypersurface defined by $\det(\mathtt{F}) = 0$). A method similar to that described in section 3.2 can be used to project the linear solution onto the constraint manifold as follows: Use the linear 8-point algorithm as described by Hartley [**?**] (with data normalization) to construct the $n \times 9$ design matrix $\mathtt{A}$. The linear estimate of $\mathtt{F}$ is obtained from $\mathtt{A}$ as the singular vector corresponding to the least singular value. In the orginal algorithm [**?**] Hartley then converts this matrix to one with rank 2 by using the SVD. The alternative proposed here is to compute the pencil of matrix solutions defined by the line joining the singular vectors corresponding to the *two* least singular values, and intersect this pencil with the cubic surface. The result is a rank 2 fundamental matrix "close to" the linear

solution. Note, that the cubic constraint in section 3.2 is on the position of the sixth point, whereas here it is on the elements of the fundamental matrix.

5 Robust Reconstruction Algorithm

In this section we describe a robust algorithm for reconstruction built on the 6-point engine of section 3. The input to the algorithm is a set of point tracks, some of which will contain mismatches. Robustness means that the algorithm is capable of rejecting mismatches, using the RANSAC [**?**] paradigm. It is a straightforward generalization of the corresponding algorithm for 7 points in 2 views [**?**,**?**] and 6 points in 3 views [**?**,**?**].

5.1 Algorithm

The input is a set of measured image projections. A number of world points (usually thousands) have been tracked through a number of images. Some tracks may last for many images, some for only a few (*ie* there may be missing data). There may be mismatches.

1. From the set of tracks which appear in all images, select six at random. This set of tracks will be called a *basis*.
2. Initialize a projective reconstruction using those six tracks. This will provide the world coordinates (of the six points whose tracks we chose) and cameras for all the views (either quasi-linear or with 3 degrees of freedom optimization on 6th point – see below).
3. For all remaining tracks, compute optimal world point positions using the computed cameras by minimizing the reprojection error over all views in which the point appears. This involves a numerical minimization.
4. Reject tracks whose image reprojection errors exceed a threshold. The number of tracks which pass this criterion is used to score the reconstruction.
5. Repeat the above steps as required.

As we have already pointed out, the ordering of the six point basis can sometimes make a difference to the quality of reconstruction, so we try each of the sixth choices for the last point (the ordering of the first five points makes no difference).

The justification for this algorithm is, as always with RANSAC, that once a "good" basis is found it will (a) score highly and (b) provide a reconstruction against which other points can be tested (to reject mismatches).

5.2 Results II

The second sequence is a turntable sequence (*ie* the camera motion is a turntable motion) of a dinosaur model (figure 5.2). The image size is 720×576. Motion tracks were obtained using the fundamental matrix based tracker described in [**?**]. We ran the algorithm with 100 samples

on the subsequence consisting of images 0 to 5. For these 6 views, there were 740 tracks of which only 32 were seen in all views. 127 tracks were seen in 4 or more views. The sequence contains both missing points and mis-matched tracks ** does it ?? **.

For the six point RANSAC basis, linear reconstructions were rejected if any reprojection error exceeded 10 pixels, and the subsequent 3 degrees of freedom optimization was rejected if any reprojection error exceeded a threshold of 5 pixels. These are very generous thresholds and are only intended to avoid spending computation on very bad initializations. The real criterion of quality is how much support an initialization has. When backprojecting tracks to score the reconstruction, only tracks seen in 4 or more views were used and tracks were rejected as mismatches if any residual exceed 1.25 pixels after backprojection.

To assess the performance of our algorithm, we tried three variations. The first mode just uses our quasi-linear algorithm. The second applies the optimization described in section 3.3. The third applies a full bundle adjustment to the 6-point reconstructions. The errors are summarized in figure 9. The last row shows errors after applying bundle adjustment to the final reconstruction (many points, many cameras). Figure 10 shows the tracks accepted by the algorithm, superimposed on the fourth (index 3) image in the sequence. Figure 11 shows the computed model. Remarks

	basis residuals (pixels)	all residuals (pixels)	inlier count
6 points no optimization	0.0443/0.183	0.401/1.24	95
6 points with optimization	0.0443/0.183	0.401/1.24	95
6 points (and cameras) bundled	0.0422/0.127	0.383/1.181	97
All points (and cameras) bundled	0.313 /0.718	0.234/0.925	95

Fig. 9. Dinosaur sequence results. Comparing the three different fitting algorithms (algebraic, reduced, fulll). There were 6 views. For each mode of operation, the number of points marked as inliers by the algorithm is shown in the third column. There were 127 tracks seen in four or more views.

entirely analogous to the ones made about the previous sequence apply to this one, but note specifically that optimizing makes no difference to the residuals at this level of precision (3 significant figures). Applying bundle adjustment to each initial 6-point reconstruction improves the fit somewhat, but the gain in accuracy and support is rather small compared to the extra computational cost (in this example, there was a 7-fold increase in computation time).

Missing figure Missing figure Missing figure Missing figure Missing figure Missing figure

Fig. 10. Tracks used in reconstruction for the dinosaur sequence.

Missing figure | Missing figure
Missing figure | Missing figure

Fig. 11. Dinosaur sequence reconstruction : snapshots of the computed structure with and without cameras shown, from four different positions. The actual camera motion is a turntable sequence with 36 images.

The results shown for view 0 to 5 are typical of results obtained for other segments of 6 consecutive views from this sequence. Decreasing the number of views used has the disadvantage of narrowing the baseline, which generally leads to both structure and cameras being less well determined. The advantage of using only a small number of points (i.e. 6 instead of 7) is that there is a higher probability that sufficient tracks will exist over many views.

6 Conclusion

1. Have shown how to use our 6-point engine to perform robust reconstruction for m views of n points. This reconstruction can now form the basis of a hierarchical method for extended image sequencess. The algorithm in [**?**] builds a hierarchical reconstruction from image triplets. Now can proceed from extended sub-sequences over which at least 6 points tracked.
2. Other minimal cases involving points and lines over $m \geq 4$ views.
3. Multi-view tensor method for 3 and 4 views.