Int J Comput Vis (2009) 83: 274–293
DOI 10.1007/s11263-009-0225-1

Reconstruction from Projections Using Grassmann Tensors

Richard I. Hartley · Frederik Schaffalitzky

Received: 24 October 2006 / Accepted: 18 February 2009 / Published online: 19 March 2009
© Springer Science+Business Media, LLC 2009

Abstract In this paper a general procedure is given for reconstruction of a set of feature points in an arbitrary dimensional projective space from their projections into lower dimensional spaces. This extends the methods applied in the well-studied problem of reconstruction of scene points in $\mathcal{P}^3$ given their projections in a set of images. In this case, the bifocal, trifocal and quadrifocal tensors are used to carry out this computation. It is shown that similar methods will apply in a much more general context, and hence may be applied to projections from $\mathcal{P}^n$ to $\mathcal{P}^m$, which have been used in the analysis of dynamic scenes, and in radial distortion correction. For sufficiently many generic projections, reconstruction of the scene is shown to be unique up to projectivity, except in the case of projections onto one-dimensional image spaces (lines), in which case there are two solutions.

Projections from $\mathcal{P}^n$ to $\mathcal{P}^2$ have been considered by Wolf and Shashua (in International Journal of Computer Vision 48(1): 53–67, 2002), where they were applied to several different problems in dynamic scene analysis. They analyzed these projections using tensors, but no general way of defining such tensors, and computing the projections was given. This paper settles the general problem, showing that tensor definition and retrieval of the projections is always possible.

Keywords Grassman tensor · Projective reconstruction · Multiview tensor · Exterior algebra

R.I. Hartley (✉)
National ICT Australia and the Australian National University, Canberra, Australia
e-mail: Richard.Hartley@anu.edu.au

F. Schaffalitzky
Australian National University, Canberra, Australia

1 Introduction

The bifocal tensor (fundamental matrix), trifocal tensor and quadrifocal tensor have been much studied as a means of reconstructing a 3-dimensional scene from its projection in two, three or four images. It is well known that given sufficiently many point (or line) correspondences between the views, it is possible to compute the multiview tensor and subsequently extract from it the original projection matrices of the cameras, up to an unavoidable projective equivalence. There have been too many papers related to this to cite them all, and so we refer here only to the following papers: Hartley and Zisserman (2004), Hartley (1998a), Heyden (1998b). The algorithms previously given for extracting the projection matrices from the bifocal, trifocal and quadrifocal tensor have been quite different, and it was not clear that a general method exists.

Projections from $\mathcal{P}^n$ to $\mathcal{P}^2$ have been considered in work of Wolf and Shashua (2002), where they were applied to several different problems in dynamic scene analysis. Typical of the problems they considered was that of 3D dynamic scene analysis for a configuration of points each moving independently with constant velocity. In this case, each point may be represented by a vector $(\mathrm{X}, \mathrm{Y}, \mathrm{Z}, 1, d\mathrm{X}, d\mathrm{Y}, d\mathrm{Z})^\top$ consisting of its position in homogeneous coordinates and its velocity. The projection of the points into an image may therefore be modelled as a projective mapping from $\mathcal{P}^6$ to $\mathcal{P}^2$. They use tensorial techniques involving 4 views to compute the image projections and the motion of the points. It follows from the present paper that tensors involving any number between 4 and 7 views can be used to solve this problem. However, the applications treated in Wolf and Shashua (2002) are considered largely on a case-by-case basis. No general method is given for defining these tensors, or extracting the projection matrices afterwards.

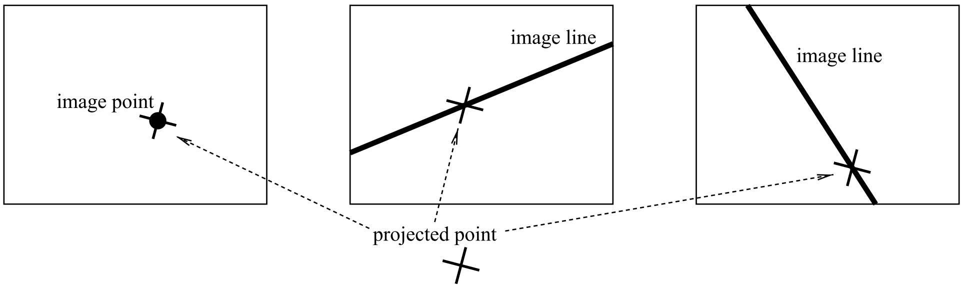

Fig. 1 Illustration of point-line-line situation of trifocal tensor

More recently, such tensors have been used in Hartley and Vidal (2008) to compute non-rigid structure and motion under perspective projection. Assuming that a set of 3D points deform within a space spanned by k independent modes of deformation, the projection of the points into an image may be naturally modelled as a projection from $\mathcal{P}^{3k}$ into $\mathcal{P}^2$. The reconstruction algorithm described in the present paper is used in Hartley and Vidal (2008) to compute the projections $\mathcal{P}^{3k} \rightarrow \mathcal{P}^2$, and hence solve the non-rigid reconstruction problem.

At the other end of the scale, Quan and Kanade (1997), Faugeras et al. (2000) have studied projections between low-dimensional spaces, namely projections from $\mathcal{P}^2$ to $\mathcal{P}^1$, and solve the reconstruction problem using a trifocal tensor. Quan (2001) shows that in this case, there are two possible reconstructions. The paper of Faugeras et al. (2000) pertains to self-calibration of a camera undergoing motion in a plane. The standard 3D to 2D image projection is used to induce a projection onto the horizon line, a 1D projective space. This projection is then analyzed using a 2D trifocal tensor.

Tensors involving projections from $\mathcal{P}^2$ to $\mathcal{P}^1$, and $\mathcal{P}^3$ to $\mathcal{P}^1$ have also been used in the analysis of images with radial distortion, and for radial distortion correction (Thirthala and Pollefeys 2005b, 2005a). The context here is that with unknown radial distortion in an image, the correct projective image of a point in space can not be determined exactly, but may be measured to lie on a radial line through the radial distortion centre (assumed to be the same as the principal point). This projection may be modelled as a mapping from the 3D world onto the 1-dimensional projective space of lines through the radial distortion centre, that is, as a mapping from $\mathcal{P}^3$ to $\mathcal{P}^1$. From four views, a quadrifocal tensor-based method is used for structure and motion recovery and ultimately to compute the radial distortion.

This paper unifies all this previous work by showing that reconstruction from projections of $\mathcal{P}^n$ into arbitrary dimensional projective spaces is always possible, and is almost always projectively unique. The method involves a generalization of the multiview tensors for projections $\mathcal{P}^3 \rightarrow \mathcal{P}^2$ (referred to subsequently as the "classical" tensors). The exceptional case is where all the projections are onto one-dimensional projective spaces (lines). In this case, two reconstructions are always possible.

The Grassmann Tensor We consider a sequence of projections from $\mathcal{P}^n$ to $\mathcal{P}^{m_i}$, for $i = 1, \ldots, r$. Thus, we do not assume that the image space always has the same dimension. For each i, we select integers α_i satisfying $1 \leq \alpha_i \leq m_i$ and $\sum_{i=1}^{r} \alpha_i = n + 1$. These values represent the codimension of affine subspaces to be specified in each of the image spaces. Thus, when $\alpha_i = m_i$, the affine subspace is a point (codimension m_i, hence dimension 0), and when $\alpha_i = 1$, the affine subspace is a codimension-1 hyperplane. A set of affine subspaces with these dimensions are said to *correspond* when there exists at least one point $\mathbf{X}$ in $\mathcal{P}^n$ that maps via each projection to a point in the given affine subspace in the corresponding image space.

For instance in three views we say that $\mathbf{x} \leftrightarrow \mathbf{l}' \leftrightarrow \mathbf{l}''$ is a point-line-line correspondence if there exists a point $\mathbf{X}$ in $\mathcal{P}^3$ that maps to $\mathbf{x}$ in the first image, and to points on the lines $\mathbf{l}'$ and $\mathbf{l}''$ in the other two images. This is illustrated in Fig. 1. Since the projections are all from $\mathcal{P}^3$ to $\mathcal{P}^2$ in this case, we see that $n = 3$ and $m_i = 2$ for all i. In addition, the codimensions α_i of point, line and line are 2, 1 and 1 respectively, and we verify that $\sum \alpha_i = 4 = n + 1$. This configuration leads to the classical trifocal tensor relationship (Hartley and Zisserman 2004) for corresponding point and lines, namely $\sum_{i,j,k} x^i l'_j l''_k T_i^{jk} = 0$.

In the general case now being considered, there also exists a tensor relating the coordinates of a set of corresponding affine subspaces in the set of images. However, to assign coordinates to affine subspaces of arbitrary dimension, we need to use Grassmann coordinates (described later). For points and lines in $\mathcal{P}^2$, the Grassmann coordinates are nothing more than the homogeneous coordinates of the point or line. It is only when we consider image spaces of higher dimension that the correct generalization in terms of Grassmann coordinates becomes apparent and useful. In this case, the tensor relates the Grassmann coordinates of the corresponding affine subspaces in each image. The relationship is of the form

$$\sum_{\sigma_1,\sigma_2,\ldots\sigma_r} |S^1_{\hat{\sigma}_1}||S^2_{\hat{\sigma}_2}|\ldots|S^r_{\hat{\sigma}_r}|\mathcal{A}_{\sigma_1\sigma_2\ldots\sigma_r} = 0.$$

The notation $|S^i_{\hat{\sigma}_i}|$ represents the σ_i-th Grassmann coordinate of the subspace S^i, where the index σ_i is explained later.

Recall that S^i is a subspace of codimension α_i in $\mathcal{P}^{m_i}$. The vector of Grassmann coordinates has dimension $C^{\alpha_i}_{m_i+1}$ (this notation is also known as "(m_i+1)-choose-α_i"), the number of values that the index σ_i can take. The sum is over all combinations of Grassmann coordinates. The notation $\hat{\sigma}_i$ is to be read "not" σ_i. What this means is not made clear until later, but the reader may safely ignore the $\hat{}$, for it is only a notational convenience (or perhaps inconvenience). In some respects the $\hat{}$ sign is analogous to the use of upper and lower indices in the classical tensor notation. We refer to $\mathcal{A}$ as a *Grassmann tensor*.

Computation of the Grassmann Tensor Given a correspondence between subspaces of codimension α_i in each $\mathcal{P}^{m_i}$, we obtain a single linear relationship between the elements of the Grassmann tensor.

If we are given a correspondence between subspaces T^i of greater codimensions, this information is clearly a stronger constraint on the tensor, and by choosing subspaces S^i of codimension α_i such that T^i is contained in S^i we obtain from the S^i a linear condition on the Grassmann tensor as before. Since there are many ways to choose such "enveloping" subspaces S^i we get several linear constraints on the tensor. For example, in the case of the trifocal tensor, a 3-point correspondence $\mathbf{x} \leftrightarrow \mathbf{x}' \leftrightarrow \mathbf{x}''$ has codimensions $(2, 2, 2)$ which is too large in the sense that in the identity $\sum_i \alpha_i = n + 1$ the left-hand side is too large: $2 + 2 + 2 = 6 > 4 = 3 + 1$. But "adjusting" the last two correspondences from points to line leads to four linear relations. These are obtained by choosing any two lines passing through $\mathbf{x}'$ and any two lines passing through $\mathbf{x}''$. Each choice of lines leads to a point-line-line correspondence, from each of which one obtains a linear relation. This same idea allows us to derive linear relations for Grassmann tensors in higher dimension, given a correspondence between subspaces of higher codimension. The exact number of linearly independent correspondences generated in this way is not explored in this paper, though it is well understood in the $\mathcal{P}^3 \to \mathcal{P}^2$ case (Heyden 1998a; Hartley and Zisserman 2004). In any case, given sufficiently many correspondences the Grassmann tensor may be computed linearly.

For clarification, it should be pointed out that for a set of projections $\mathcal{P}^n \to \mathcal{P}^{m_i}$, there may be many different tensors, depending on the choice of the sequence of codimensions $(\alpha_1, \alpha_2, \ldots, \alpha_r)$. The only restrictions are that $1 \le \alpha_i \le m_i$ and $\sum_i \alpha_i = n + 1$. In the well-known case of the trifocal tensor, there are actually three different tensors depending on which of the three images is chosen to have the contravariant index. The three tensors have codimension sequences $(2, 1, 1)$, $(1, 2, 1)$ and $(1, 1, 2)$ respectively. In the general case, we call the sequence of codimensions $(\alpha_1, \alpha_2, \ldots \alpha_r)$ the *profile* of the corresponding tensor. Each such profile corresponds to a different tensor. If we are computing a tensor from point correspondences across several views, then it is necessary to choose in advance which profile to use, since any profile consistent with the dimensions of the image spaces can be used.

Extraction of Projection Matrices Having computed a Grassmann tensor from a set of affine subspace correspondences, we now seek to extract the projection matrices. Ad-hoc techniques for computing the projections from multiview tensors have been proposed in the past, both for the standard case of $\mathcal{P}^3 \to \mathcal{P}^2$ as well as for higher-dimensional cases (Wolf and Shashua 2002). We shall give a general procedure for doing this, and show that (at least for generic projections) the projection matrices are determined uniquely by a Grassmann tensor up to projective equivalence, *except* in the case where each $m_i = 1$. In this latter case, there will always be two projectively non-equivalent solutions, and indeed this represents a basic ambiguity for projective reconstruction from projections onto lines. This ambiguity persists however many point correspondences are involved. The two projective reconstructions are related to each other by a Cremona transform, which is a non-linear transformation of $\mathcal{P}^n$ (Semple and Kneebone 1979).

2 Definition of the Grassmann Tensors

We begin by recalling the derivation of the bifocal tensor (fundamental matrix) and trifocal tensor relating corresponding points in two and three views in the familiar case of projections from $\mathcal{P}^3$ to $\mathcal{P}^2$. This theory is covered in Hartley and Zisserman (2004), but we review it here for convenience of the reader, but also (more importantly) because the formalization will be slightly different, in order to allow generalization to higher dimensions.

2.1 The Fundamental Matrix

We suppose two projection matrices $\mathtt{A}$ and $\mathtt{A}'$ represent projections from $\mathcal{P}^3$ to $\mathcal{P}^2$. Let $\mathbf{x} \leftrightarrow \mathbf{x}'$ be the images of a common point $\mathbf{X}$. We may write this relationship in terms of a single matrix equation

$$\left[\begin{array}{c|c|c} \mathtt{A} & \mathbf{x} & 0 \\ \hline \mathtt{A}' & 0 & \mathbf{x}' \end{array}\right] \begin{pmatrix} \mathbf{X} \\ \lambda \\ \lambda' \end{pmatrix} = \mathbf{0}.$$

From the existence of a solution to this equation set, we deduce that the 6×6 matrix on the left must be singular, and hence have zero determinant. Expanding the determinant down the last two columns we obtain an equation

$$\sum_{i=1}^{3} \sum_{j=1}^{3} x_i x'_j \mathcal{F}_{\hat{i},\hat{j}} = 0. \qquad (1)$$

This notation needs a little explanation. The notation $\hat{}$ is to be read "not". Thus $\hat{i}$ means "not i". Then $\mathcal{F}_{\hat{i},\hat{j}}$ is a 3×3 array indexed by i and j, and each $\mathcal{F}_{\hat{i},\hat{j}}$ is the determinant of a 4×4 matrix made up of two rows from each of $\mathtt{A}$ and $\mathtt{A}'$. Specifically,

$$\mathcal{F}_{\hat{i},\hat{j}} = (-1)^{i+j} \begin{vmatrix} \mathtt{A}_{\hat{i}} \\ \mathtt{A}'_{\hat{j}} \end{vmatrix},$$

where $\mathtt{A}_{\hat{i}}$ means the matrix obtained from $\mathtt{A}$ by deleting the i-th row.

Note here that $\mathbf{x}$ and $\mathbf{x}'$ may be thought of as affine subspaces of $\mathcal{P}^2$ of codimension 2 (that is, dimension 0). Furthermore, each $\mathcal{F}_{\hat{i},\hat{j}}$ is expressed in terms of a matrix containing two rows from each projection matrix. For these reasons, we refer to the fundamental matrix as having a profile $(2, 2)$.

2.2 The Trifocal Tensor

Now, consider three projection matrices $\mathtt{A}$, $\mathtt{A}'$ and $\mathtt{A}''$. The natural correspondence involving the trifocal tensor is a point-line-line correspondence $\mathbf{x} \leftrightarrow \mathbf{l}' \leftrightarrow \mathbf{l}''$, which is to be interpreted to mean that there exists a 3D point $\mathbf{X}$ that maps to the point $\mathbf{x}$ in the first image, and to some points on the lines $\mathbf{l}'$ and $\mathbf{l}''$ in the other two images.

We choose two points $\mathbf{x}'$ and $\mathbf{y}'$ on the line $\mathbf{l}'$, and similarly two points $\mathbf{x}''$ and $\mathbf{y}''$ on the line $\mathbf{l}''$. There must therefore exist constants (λ', μ') such that $\mathbf{X}$ maps to the point $\lambda'\mathbf{x}' + \mu'\mathbf{y}'$ on the line $\mathbf{l}'$. Similar constants (λ'', μ'') exist for the third image. As in the fundamental matrix case, we may express the correspondence relation in a single matrix equation as follows (blank entries are zero).

$$\left[\begin{array}{c|c|c|c} \mathtt{A} & \mathbf{x} & & \\ \hline \mathtt{A}' & & \mathbf{x}'\ \mathbf{y}' & \\ \hline \mathtt{A}'' & & & \mathbf{x}''\ \mathbf{y}'' \end{array}\right] \begin{pmatrix} \mathbf{X} \\ \lambda \\ \lambda' \\ \mu' \\ \lambda'' \\ \mu'' \end{pmatrix} = \mathbf{0}.$$

As before, the matrix here is square, and has zero determinant. We may expand the determinant explicitly as a sum of signed products of matrix elements, each product involving just one element from each row and each column of the matrix. Any such non-zero product term arises from choosing a non-zero entry from each of the last 5 columns. We may then strike out the rows containing the elements so chosen. What remains is a matrix containing 2 rows from $\mathtt{A}$, and one row from each of $\mathtt{A}'$ and $\mathtt{A}''$.

The resulting relation may be written as

$$\sum_{i,j,k} x_i |\mathbf{x}'\mathbf{y}'|_{\hat{j}} |\mathbf{x}''\mathbf{y}''|_{\hat{k}} (-1)^{i+j+k} \begin{vmatrix} \mathtt{A}_{\hat{i}} \\ \mathtt{A}'_{j} \\ \mathtt{A}''_{k} \end{vmatrix}$$
$$= \sum_{i,j,k} x_i |\mathbf{x}'\mathbf{y}'|_{\hat{j}} |\mathbf{x}''\mathbf{y}''|_{\hat{k}} \mathcal{T}_{\hat{i}jk} = 0, \qquad (2)$$

where $\mathcal{T}_{\hat{i}jk}$ is defined by this equality. Here, the expression $|\mathbf{x}'\mathbf{y}'|_{\hat{j}}$ has the following meaning. The two column vectors $\mathbf{x}'$ and $\mathbf{y}'$ together form a 3×2 matrix. Then $|\mathbf{x}'\mathbf{y}'|_{\hat{j}}$ is the value of the 2×2 determinant formed by deleting the j-th row of this matrix.

In this case, the entries of the tensor $\mathcal{T}_{\hat{i}jk}$ are determinants of matrices formed from 2 rows of $\mathtt{A}$ and 1 row from each of $\mathtt{A}'$ and $\mathtt{A}''$. We say that the tensor has profile $(2, 1, 1)$.

Grassmann Coordinates Consider the affine subspace of $\mathcal{R}^n$ spanned by a set of k vectors $\{\mathbf{x}_i\}$, and form an $n \times k$ matrix in which these vectors are the columns. The $k \times k$ minors of this matrix are known as the Grassmann coordinates of the affine space (since $k \leq n$ the matrix is "tall" and so these minors are the maximal minors of the matrix). Thus, in the case above for instance, the $|\mathbf{x}'\mathbf{y}'|_{\hat{j}}$ are (for varying j) the Grassmann coordinates of the affine space spanned by $\mathbf{x}'$ and $\mathbf{y}'$.

In the case of a single point, $\mathbf{x}$, the Grassmann coordinates are of course equal to the coordinates of the point itself. In the case of a line in $\mathcal{P}^2$ spanned by two points $\mathbf{x}$ and $\mathbf{y}$, the Grassmann coordinates are, with correctly chosen sign, simply the homogeneous coordinates of the line. More precisely, if $\mathbf{l} = \mathbf{x} \times \mathbf{y}$, then $\mathbf{l}_k = (-1)^{k+1}|\mathbf{xy}|_{\hat{k}}$. Using this relationship, we could write (2) in the more standard form $x_i l_j l''_k T_i^{jk} = 0$, where T_i^{jk} is the usual trifocal tensor (Hartley and Zisserman 2004), related to $\mathcal{T}_{\hat{i}jk}$ by $T_i^{jk} = (-1)^{j+k}\mathcal{T}_{\hat{i}jk}$.

Thus, (1) and (2) may be interpreted as linear relations between the Grassmann coordinates of corresponding affine subspaces, expressed by the appropriate tensor. This is the correct way of generalizing these results to higher dimensions, where indeed the Grassmann coordinates of the appropriate subspaces are related via a multiview tensor. The distinction between contravariant and covariant indices, which occurs in the case of the trifocal tensor is not relevant in the general case, where the tensors transform under coordinate transforms according to the corresponding transformation of Grassmann coordinates. The covariant transformation rule for indices corresponding to lines is an accident based on the fact that the homogeneous coordinates of the line are the same as the Grassmann coordinates.

2.3 The General Case

A mapping from $\mathcal{P}^n$ to $\mathcal{P}^m$ is represented by a matrix of dimension $(m+1)\times(n+1)$, acting on homogeneous coordinates. We consider a set of r such mappings, where the i-th mapping is from $\mathcal{P}^n$ to $\mathcal{P}^{m_i}$. Thus the dimension of the image of this mapping may be different in each case. The matrix representing the i-th mapping will be denoted by $\mathtt{A}^i$ and has size $(m_i+1)\times(n+1)$; though these matrices may have different numbers of rows they all have $n+1$ columns.

We introduce the concept of an *ordered partition* of $n+1$. This is an ordered tuple of non-negative integers $(\alpha_1,\alpha_2,\ldots,\alpha_r)$ that sum to $n+1$. We are interested in those partitions of n for which each α_i lies in the range 1 to m_i. We will show that for each such ordered partition, there exists an r-view tensor (where r is the length of the partition) relating the coordinates of matched codimension-α_i affine subspaces in r images.

Thus when $n=3$ and each $m_i=2$, the possible ordered partitions of $4=3+1$ are $(2,2)$, $(2,1,1)$, $(1,2,1)$, $(1,1,2)$ and $(1,1,1,1)$. These partitions correspond to the well-known multiview tensors for 2, 3 and 4 views. We see that there is a bifocal tensor (the fundamental matrix) corresponding to the partition $(2,2)$, three trifocal tensors corresponding to the three partitions of length 3, and one quadrifocal tensor, corresponding to the partition $(1,1,1,1)$.

We will call the ordered partition corresponding to a given tensor the profile of the tensor. How the tensor with a given profile is defined will now be explained.

Given $d+1$ points spanning a d-dimensional affine subspace of some projective space, we assemble the points as the columns of a matrix $\mathtt{S}$. The affine subspace is simply the span of the columns of $\mathtt{S}$ and any point in this subspace can be written in the form $\mathtt{S}\mathbf{v}$ for some suitable vector $\mathbf{v}$. We may speak of the matrix $\mathtt{S}$ as *representing* the subspace. The condition for a point $\mathbf{X}$ in $\mathcal{P}^n$ to map into the subspace under a mapping represented by $\mathtt{A}$ is that $\mathtt{A}\mathbf{X}+\mathtt{S}\mathbf{v}=0$ for some $\mathbf{v}$.

Now, choose a set of affine subspaces each of codimension α_i in its projective space $\mathcal{P}^{m_i}$ and let $\mathtt{S}^i$ be the matrix representing the subspace. Thus, $\mathtt{S}^i$ has dimension $(m_i+1)\times(m_i-\alpha_i+1)$. Suppose that there exists a point $\mathbf{X}$ in $\mathcal{P}^n$ that maps under a projection (represented by $\mathtt{A}^i$) to a point lying in the subspace $\mathtt{S}^i$. If this condition holds for all i, we will see that it induces a single constraint on the set of projection matrices $\mathtt{A}^i$.

The fact that this same $\mathbf{X}$ projects into each of the subspaces may be written in one matrix equation as follows.

$$\begin{bmatrix} \mathtt{A}^1 & \mathtt{S}^1 & & & \\ \mathtt{A}^2 & & \mathtt{S}^2 & & \\ \vdots & & & \ddots & \\ \mathtt{A}^r & & & & \mathtt{S}^r \end{bmatrix} \begin{pmatrix} \mathbf{X} \\ \mathbf{v}_1 \\ \mathbf{v}_2 \\ \vdots \\ \mathbf{v}_r \end{pmatrix} = \mathbf{0}. \tag{3}$$

Note that the matrix on the left is square. To check this: the number of rows is equal to $\sum_{i=1}^r (m_i+1)$, whereas the number of columns is equal to

$$(n+1)+\sum_{i=1}^r (m_i+1-\alpha_i) = \sum_{i=1}^r (m_i+1)$$

$$\text{since}\quad \sum_{i=1}^r \alpha_i = (n+1).$$

In order for a non-zero solution to this set of equations to exist, it is necessary that the determinant of the matrix be zero. If the coordinates of the subspaces (the matrices $\mathtt{S}^i$) are given, then this provides a single constraint on the entries of the matrices $\mathtt{A}^i$. To understand the form of this constraint, we need to expand out this determinant, and to do that, we shall need to use Grassmann coordinates.

Grassmann Coordinates Given a matrix $\mathtt{M}$ with q rows and p columns, where $p \le q$, we define its *Grassmann coordinates* to be the sequence of its maximal minors (determinants of all its $p\times p$ submatrices). It is a well known fact that the Grassmann coordinates of a (full-rank) matrix determine its column span. Specifically, the Grassmann coordinates determine the matrix up to right-multiplication by a $p\times p$ matrix with unit determinant. Let σ represent a sequence of p distinct integers in the range 1 to q, in ascending order. Let $|\mathtt{M}_\sigma|$ represent the determinant of the matrix that consists of the rows of $\mathtt{M}$ specified by the sequence σ. Then the values of the determinant $|\mathtt{M}_\sigma|$, as σ ranges over all such sequences, are the Grassmann coordinates of the matrix.

Now, given such a sequence σ indexing the rows of a matrix, let $\hat{\sigma}$ represent the sequence consisting of those integers not in σ. and define $\text{sign}(\sigma)$ to be $+1$ or -1 depending on whether the concatenated sequence $\sigma\hat{\sigma}$, which is a permutation, is even or odd (here we use the standard notion of the sign, or parity, of a permutation).

$$\text{sign}(\sigma) = \text{sign}(\sigma\ \hat{\sigma}).$$

Thus, for example the sequence $\sigma = 125$ has sign $+1$, since $\hat{\sigma} = 34$ and 12534 is an even permutation, being the 3-cycle (3 5 4).

Given a square matrix divided into two blocks, for instance $[\mathtt{A}|\mathtt{B}]$, its determinant may be expressed in terms of the Grassmann coordinates of $\mathtt{A}$ and $\mathtt{B}$. In particular

$$|[\mathtt{A}|\mathtt{B}]| = \sum_\sigma \text{sign}(\sigma)\ |\mathtt{A}_\sigma|\ |\mathtt{B}_{\hat{\sigma}}| \tag{4}$$

where the sum is over all ascending sequences σ of length equal to the number of columns of $\mathtt{A}$. The particular case where $\mathtt{A}$ consists of a single column is just the familiar cofactor expansion of the determinant.

Generalizing the factorization (4), or applying it repeatedly, one may derive a precise formula for the determinant of the matrix on the left of (3), namely

$$\pm \sum_{\sigma_1,\sigma_2,\dots,\sigma_r} \operatorname{sign}(\sigma_1)\cdots\operatorname{sign}(\sigma_r)|\mathtt{A}_{\sigma_1\sigma_2\dots\sigma_r}| \times |\mathtt{S}^1_{\hat{\sigma}_1}||\mathtt{S}^2_{\hat{\sigma}_2}|\cdots|\mathtt{S}^r_{\hat{\sigma}_r}|. \tag{5}$$

In this formula, each σ_i is an ordered sequence of α_i integers in the range from 1 to $m_i + 1$, the length of the sequence being equal to the codimension of the subspace $\mathtt{S}^i$. The notation $|\mathtt{A}_{\sigma_1\sigma_2\dots\sigma_r}|$ means the determinant of the matrix obtained by selecting the rows indexed by σ_i from each $\mathtt{A}^i$. The notation $|\mathtt{S}^i_{\hat{\sigma}_i}|$ represents the determinant of the matrix obtained from $\mathtt{S}^i$ by deleting the rows indexed by σ_i. This is a Grassmann coordinate of the subspace represented by the columns of $\mathtt{S}^i$. The overall sign, whether $+$ or $-$, does not concern us. The set of values

$$\mathcal{A}_{\sigma_1\sigma_2\dots\sigma_r} = \operatorname{sign}(\sigma_1)\operatorname{sign}(\sigma_2)\cdots\operatorname{sign}(\sigma_r)|\mathtt{A}_{\sigma_1\sigma_2\dots\sigma_r}| \tag{6}$$

forms an r-dimensional array whose elements are (up to sign) minors of the matrix $\mathtt{A}$ obtained by stacking the projection matrices $\mathtt{A}^i$. The only minors are ones corresponding to submatrices of $\mathtt{A}$, in which α_i rows are chosen from each $\mathtt{A}^i$. Recalling that the sequence $(\alpha_1,\dots,\alpha_r)$ in which $\sum_{i=1}^r \alpha_i = n+1$ is called a profile, we will call the array $\mathcal{A}_{\sigma_1\sigma_2\dots\sigma_r}$ the *Grassmann tensor* corresponding to the profile $(\alpha_1,\dots,\alpha_r)$.

The tensor $\mathcal{A}$ gives a linear relationship between the Grassmann coordinates of affine subspaces defined in each of the image spaces $\mathcal{P}^{m_i}$:

$$\sum_{\sigma_1,\sigma_2,\dots,\sigma_r} \mathcal{A}_{\sigma_1\sigma_2\dots\sigma_r}|\mathtt{S}^1_{\hat{\sigma}_1}||\mathtt{S}^2_{\hat{\sigma}_2}|\cdots|\mathtt{S}^r_{\hat{\sigma}_r}| = 0. \tag{7}$$

This relationship generalizes the classical bifocal and trifocal relations (Hartley and Zisserman 2004). The classical tensors involve relations between point and line coordinates in $\mathcal{P}^2$. However, the Grassmann coordinates of a single point (a 0-dimensional affine space) are simply the homogeneous coordinates of the point. Similarly, for a line in $\mathcal{P}^2$, the Grassmann coordinates are the same as the homogeneous coordinates, except for sign.

2.3.1 *Coordinate-Free Formulation and Coordinate Changes*

Readers familiar with exterior algebra may appreciate that our construction can be carried out in a coordinate-free setting without reference to minors, determinants or matrices. Readers who are not interested can safely skip this section.

We replace the underlying vector space $\mathcal{R}^{n+1}$ of $\mathcal{P}^n$ with any vector space V of dimension $n+1$ and for each $\mathcal{P}^{m_i}$ we introduce a vector space V_i of dimension $m_i + 1$. The projection matrices $\mathtt{A}^i$ then become linear maps $\mathtt{A}^i : V \to V_i$. The codimension-α_i subspaces S^i of V_i are represented by elements of $\Lambda^{\alpha_i} V_i^*$, the α_i-th exterior power of the dual V_i^* of V_i. If the subspace is cut out by α_i linear forms $\theta_1, \theta_2, \dots \theta_{\alpha_i} \in V_i^*$ then the required element is $\theta_1 \wedge \theta_2 \wedge \cdots \wedge \theta_{\alpha_i} \in \Lambda^{\alpha_i} V_i^*$. Earlier we described our subspaces as given by $m_i + 1 - \alpha_i$ generators $x_1, x_2, \dots, x_{m_i+1-\alpha_i}$ from the original spaces V_i rather than α_i generators from the dual space; in this case one forms the product $S^i = x_1 \wedge x_2 \wedge \cdots \wedge x_{m_i+1-\alpha_i} \in \Lambda^{m_i+1-\alpha_i} V_i$ and uses the isomorphism $* : \Lambda^{m_i+1-\alpha_i} V_i \to \Lambda^{\alpha_i} V_i^*$. (The isomorphism corresponds to a choice of a non-zero element of $\Lambda^{m_i+1} V_i$ and so is only determined up to scale, meaning that there is a 1-parameter family of such isomorphisms; they are all scalar multiples of each other.)

Turning to projection, each linear map $\mathtt{A}^i$ sends an element $X \in V$ to an element $\mathtt{A}^i X \in V_i$ that may or may not lie in the subspace S^i. One way to determine if X projects to the subspace is to "pull back" the linear forms θ cutting S^i out of V_i to the linear forms $\theta \mathtt{A}^i$ on V and test whether $\theta \mathtt{A}^i X = 0$ for each form θ. Another way to express this condition is to note that the pulling back is effected by the transpose (or dual) linear map $\mathtt{A}^{i\top} : V_i^* \to V^*$ and taking exterior powers of this we get a linear map

$$\Lambda^{\alpha_i} \mathtt{A}^{i\top} : \Lambda^{\alpha_i} V_i^* \to \Lambda^{\alpha_i} V^*$$

that maps the element representing S^i to an element representing the pre-image of S^i under $\mathtt{A}^i$, itself an affine subspace of codimension (at most) α_i. Since we have assumed that $\sum_i \alpha_i = n+1$ we see that these subspaces of V have dimensions adding up to the dimension of V and we can answer the question of whether there is a non-zero element X contained in all them by testing for vanishing of the wedge product

$$\Lambda^{\alpha_1} \mathtt{A}^{1\top}(*S^1) \wedge \Lambda^{\alpha_2} \mathtt{A}^{2\top}(*S^2) \wedge \cdots \wedge \Lambda^{\alpha_r} \mathtt{A}^{r\top}(*S^r) \in \Lambda^{n+1} V^*.$$

Because the dualizing operator $*$ is linear, each map $\Lambda^{\alpha_i} \mathtt{A}^{i\top}$ is linear, and the wedge product $\wedge$ is bilinear, the expression is multi-linear in the Grassmann coordinates of the spaces S^i. The expression is just another form of (7) but written without reference to coordinates.

It is not hard to apply this formula to the coordinate case and arrive at the previous formulation. A useful byproduct is that it shows what the correct rule for coordinate transforms of the tensor is. Suppose we wanted to use different coordinates in the ith image, and that we have the (square, invertible) change-of-basis matrix $\mathtt{B}$ for that image; this corresponds to a linear map $\mathtt{B} : V_i \to V_i$. The coordinate-free expression shows that the Grassmann vectors naturally live in the exterior power $\Lambda^{m_i+1-\alpha_i} V_i \cong \Lambda^{\alpha_i} V_i^*$ and so the ith "slot" of the tensor must live in the dual $\Lambda^{\alpha_i} V_i$ of this. Hence the only possible natural way to transform the i-th "slot" of the tensor is to use the (square, invertible) exterior power $\Lambda^{\alpha_i}\mathtt{B}$.

We analyze this in the familiar case of the 2, 1, 1 trifocal tensor, which is an object that lives naturally in the tensor product $V_1^* \otimes V_2 \otimes V_3$:

- The first factor here is V_1^* because the first "slot" of the tensor is for a point (an element of V_1). A coordinate change of $\mathcal{P}(V_1)$ that involves multiplying homogeneous 3-vectors by a 3×3 (homography) matrix $\mathtt{B}$ would act on the tensor in the only possible natural way, by acting along the first index as the coordinate change would act on lines (elements of V_1^*). In matrix terms, this means pre-multiplying the matrices $\mathcal{T}_i^{j1}$, $\mathcal{T}_i^{j2}$, $\mathcal{T}_i^{j3}$ with $\mathtt{B}^{-\top}$, or $\Lambda^2\mathtt{B}$.
- The second factor is V_2 because the second "slot" of the tensor is for a line (an element of V_2^*). A coordinate change that acts as matrix $\mathtt{B}$ on homogeneous coordinate in the second image would act on the second index as $\mathtt{B}$ too. However, in the matrices $\mathcal{T}_i^{j1}$, $\mathcal{T}_i^{j2}$, $\mathcal{T}_i^{j3}$, the second index is written along the rows and so we should post-multiply them with $\mathtt{B}^\top$. The transpose is just an artifact of writing the index of vectors left-to-right instead of top-to-bottom. It is similar to the situation for the fundamental matrix where transposes are used on one side of the matrix but not on the other.
- The third slot is similar to the second, except that the third index runs "across" the three matrices and there is no standard matrix operation to describe the action. It could of course be achieved by relabelling images 2 and 3, post-multiply with $\mathtt{B}^\top$ (as before) and then relabelling again.

2.4 Simplified Notation for Dimension 2

The tensor $\mathcal{A}_{\sigma_1\sigma_2\ldots\sigma_r}$ defined in (6) is indexed by the σ_i, which are increasing sequences of α_i integers in the range $1, \ldots, m_i + 1$. In the case where each $m_i = 2$, for instance the familiar case of projections into a 2-dimensional "image", such sequences σ_i have length either 1 or 2. For simplicity, we denote a sequence of length 1 by the value of the single integer in the sequence. Thus, by a slight abuse of notation, we denote by i the sequence of length 1 consisting of the integer i. Here, i lies in the range $1, \ldots, 3$. Sequences of length 2, will be denoted by the missing element. Thus, $\hat{1}$ represents the sequence $(2, 3)$. Likewise, $\hat{2}$ represents the sequence $(1, 3)$ and $\hat{3}$ represents the sequence $(1, 2)$.

For the trifocal $\mathcal{T}_{\sigma_1,\sigma_2,\sigma_3}$ with profile $(\alpha_1, \alpha_2, \alpha_3) = (2, 1, 1)$, the sequence σ_1 has length 2, whereas σ_2 and σ_3 have length 1. Consequently, the tensor may then be denoted by $\mathcal{T}_{\hat{i},j,k}$, where i, j and k are all integers, representing sequences of length 1. For aesthetic reasons we simplify this notation by omitting the commas, resulting in the notation $\mathcal{T}_{\hat{i}jk}$.

In a similar way, the fundamental matrix is represented by $\mathcal{F}_{\hat{i}\hat{j}}$ and the quadrifocal tensor by $\mathcal{Q}_{ijkl}$. Note that the indices $\hat{i}$ correspond to the covariant indices, and those without $\hat{}$ correspond to the contravariant indices in the usual (Hartley and Zisserman 2004) notation $\mathcal{F}_{ij}$, $\mathcal{T}_i^{jk}$ and $\mathcal{Q}^{ijkl}$ for these tensors.

3 Solving for the Grassmann Tensor

Suppose we have a set of r projections from $\mathcal{P}^n$ to $\mathcal{P}^{m_i}$ and a profile $(\alpha_1, \ldots, \alpha_r)$ such that $\sum_{i=1}^r \alpha_i = n + 1$. Let r affine subspaces be given with codimension α_i, in the respective image spaces, and suppose there exists at least one point $\mathbf{X}$ that maps into all of the affine subspaces. (Such a set of subspaces are said to "correspond".) Then (5) gives a linear relationship between the entries of the Grassmann tensor with profile $(\alpha_1, \ldots, \alpha_r)$.

Given sufficiently many sets of corresponding subspaces, we may solve linearly for the entries of the tensor. In the case of projections from $\mathcal{P}^3$ to $\mathcal{P}^2$, this gives the well-known method of solving for the fundamental matrix, trifocal tensor or quadrifocal tensor.

As in the case of the trifocal tensor (see Hartley and Zisserman 2004) we may obtain linear relationships when different dimensional subspaces are in correspondence. For instance, the trifocal tensor naturally relates correspondences between a point in one image and lines in the other two images, that is a point-line-line correspondence. This is a correspondence with profile (2, 1, 1), corresponding to the codimensions of the subspaces. Suppose that we have instead a point-point-point correspondence. One obtains point-line-line correspondences by choosing arbitrary lines passing through the points in the second and third images. Since we may choose two independent lines passing through any point, we obtain four different point-line-line correspondences in this way. The same idea works in higher dimensions. Thus, given a tensor with a given profile $(\alpha_1, \ldots, \alpha_r)$, and given a subspace with higher codimension than α_i, we may choose a set of independent codimension α_i subspaces that meet in the given subspace, and for each such choice obtain a linear equation in the entries of the tensor. Details of this procedure and investigation of the number of independent equations so obtained are left to the reader. We summarize this discussion as follows.

Theorem 3.1 *Given r projections from $\mathcal{P}^n$ to $\mathcal{P}^{m_i}$ and integers $(\alpha_1, \ldots, \alpha_r)$ satisfying $1 \le \alpha_i \le m_i$ and $\sum_{i=1}^r \alpha_i = n + 1$, there exists an r-dimensional Grassmann tensor $\mathcal{A}$ relating the Grassmann coordinates of corresponding* *codimension-α_i affine subspaces. Thus, each set of corresponding affine subspaces of the $\mathcal{P}^{m_i}$ contributes a linear relationship given by (5) between the entries of the tensor. Given sufficiently many such corresponding subspaces, the tensor may be determined linearly.*

4 Solving for the Projection Matrices

We now consider the problem of determining the projection matrices from a Grassmann tensor. As in the standard case of 3D reconstruction from uncalibrated image measurements, we can not expect to determine the projection matrices more exactly than up to projectivity. In addition, since the projection matrices are homogeneous objects, their overall scale is indeterminate. Thus, we make the following definition:

Definition 4.1 Two sequences of projection matrices $(\mathtt{A}^1, \ldots, \mathtt{A}^r)$ and $(\bar{\mathtt{A}}^1 \ldots, \bar{\mathtt{A}}^r)$ are projectively equivalent if there exists an invertible matrix $\mathtt{H}$ as well as scalars λ_i such that $\bar{\mathtt{A}}^i = \lambda_i \mathtt{A}^i \mathtt{H}$ for all i.

Now, let $\mathtt{A}$ be formed by stacking all the $\mathtt{A}^i$ on top of each other, resulting in a matrix of dimension

$$\left(\sum_{i=1}^{r}(m_i+1)\right) \times (n+1).$$

We accordingly associate the matrices $\mathtt{A}^i$ with successive vertically stacked blocks of the matrix $\mathtt{A}$. Corresponding to definition 4.1, we may define an equivalence relation on matrices with this block structure, as follows.

Definition 4.2 Two matrices $\mathtt{A}$ and $\bar{\mathtt{A}}$ made up of blocks $\mathtt{A}^i$ and $\bar{\mathtt{A}}^i$ of dimension $(m_i+1)\times(n+1)$ are *block projectively equivalent* if there exists an invertible $(n+1)\times(n+1)$ matrix $\mathtt{H}$, and scalar matrices $\lambda_i \mathtt{I}_i$ of dimension $(m_i+1)\times(m_i+1)$ such that

$$\bar{\mathtt{A}} = \mathrm{diag}(\lambda_1 \mathtt{I}_1, \ldots, \lambda_r \mathtt{I}_r)\mathtt{A}\mathtt{H}.$$

It is easily seen that this definition is equivalent to the projective equivalence of the sequences of matrices $(\mathtt{A}^1, \ldots, \mathtt{A}^r)$ and $(\bar{\mathtt{A}}^1 \ldots, \bar{\mathtt{A}}^r)$ as stated in Definition 4.1. It is evident that this is an equivalence relation on matrices with this given block structure.

We note that the notion of block projective equivalence arises from the group action defined by $\mathtt{H}$ and the λ_i and that the Grassmann tensor is invariant to this action.

4.1 The Trifocal Tensor

We give an example of how to retrieve the projection matrices from the trifocal tensor. Since this case involves three views instead of two for the simpler fundamental matrix, the computation is a little involved, and requires the solution of a quadratic equations, as we will see. The case of the fundamental matrix does not display the full range of complications and the reader could do this case as an exercise to test his or her understanding. For concreteness and clarity, we carry out the computation on a specific numerical example.

Let the trifocal tensor be

$$\mathcal{T} = \left\{ \begin{bmatrix} \hat{1}11 & \hat{1}12 & \hat{1}13 \\ \hat{1}21 & \hat{1}22 & \hat{1}23 \\ \hat{1}31 & \hat{1}32 & \hat{1}33 \end{bmatrix} \begin{bmatrix} \hat{2}11 & \hat{2}12 & \hat{2}13 \\ \hat{2}21 & \hat{2}22 & \hat{2}23 \\ \hat{2}31 & \hat{2}32 & \hat{2}33 \end{bmatrix} \begin{bmatrix} \hat{3}11 & \hat{3}12 & \hat{3}13 \\ \hat{3}21 & \hat{3}22 & \hat{3}23 \\ \hat{3}31 & \hat{3}32 & \hat{3}33 \end{bmatrix} \right\}$$
$$= \left\{ \begin{bmatrix} -1 & 8 & -6 \\ 5 & -15 & 10 \\ -5 & 5 & -2 \end{bmatrix} \begin{bmatrix} -2 & 13 & -11 \\ 4 & -12 & 8 \\ -6 & 4 & 2 \end{bmatrix} \begin{bmatrix} 1 & -9 & 7 \\ -6 & 18 & -12 \\ 6 & -6 & 2 \end{bmatrix} \right\}, \qquad (8)$$

where the array in the first line indicates the positions of the different entries $\mathcal{T}_{\hat{i}jk}$ of the tensor.

Step 1. Normalization We begin by choosing a non-zero element of $\mathcal{T}$, and scaling $\mathcal{T}$ so that this element becomes equal to 1. In the discussion below, we assume $\mathcal{T}_{\hat{3}11} = 1$ (as in the numerical example being discussed). If another element of $\mathcal{T}$ is non-zero, we may permute the indices and scale as necessary to ensure that $\mathcal{T}_{\hat{3}11} = 1$. Let $\mathtt{G}$ be the matrix formed from the first two rows of A^1 and the first row of each of A^2 and A^3. The determinant $|\mathtt{G}| = \mathcal{T}_{\hat{3}11} = 1$ (so in particular, $\mathtt{G}$ is invertible). Of course, the matrices $\mathtt{A}^i$ are unknown; it is our purpose to find them. However, whatever they are, multiplying them by $\mathtt{G}^{-1}$ will reduce them to a projectively equivalent set of matrices for which the matrix $\mathtt{G}$ formed as above is the identity matrix. The stacked matrix looks like this:

$$\mathtt{A} = \left[\begin{array}{cc|c|c} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ a & b & c & d \\ \hline 0 & 0 & 1 & 0 \\ e & f & g & h \\ j & k & m & n \\ \hline 0 & 0 & 0 & 1 \\ p & q & r & s \\ t & u & v & w \end{array}\right],$$

where the values $a, b, \ldots, w$ are yet to be determined values. We have also partitioned the columns of the matrix $\mathtt{A}$ according to the profile $\alpha = (2, 1, 1)$. By applying this normalization, at one stroke we have determined most of the matrices $\mathtt{A}^i$. Henceforth, we may assume that they are of the form

$$\mathtt{A}^1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ a & b & c & d \end{bmatrix}, \qquad \mathtt{A}^2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ e & f & g & h \\ j & k & m & n \end{bmatrix},$$

$$\mathtt{A}^3 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ p & q & r & s \\ t & u & v & w \end{bmatrix}.$$

A crucial observation concerns the positions of the 1-entries in this block decomposition of the matrix $\mathtt{A}$. They occur only in the diagonal blocks of $\mathtt{A}$. This observation will be used in the next step of the algorithm.

Step 2. Forming the Reduced Matrix From the previous normalization step, we have reduced some of the rows of the matrices $\mathtt{A}^i$ to simple form, the rows of an identity matrix. These rows will be called the *reduced rows* of the $\mathtt{A}^i$. We already know the form of these rows, and need to determine the entries of the remaining rows of the $\mathtt{A}^i$. Consider the matrix formed by stacking the remaining rows of all the $\mathtt{A}^i$ on top of each other to form a block-partitioned matrix

$$\mathtt{B} = \left[\begin{array}{cc|c|c} a & b & c & d \\ \hline e & f & g & h \\ j & k & m & n \\ \hline p & q & r & s \\ t & u & v & w \end{array}\right] = \begin{bmatrix} \mathtt{B}_{11} & \mathtt{B}_{12} & \mathtt{B}_{13} \\ \mathtt{B}_{21} & \mathtt{B}_{22} & \mathtt{B}_{23} \\ \mathtt{B}_{31} & \mathtt{B}_{32} & \mathtt{B}_{33} \end{bmatrix}. \tag{9}$$

If we can determine the entries of $\mathtt{B}$, then we have the complete matrices $\mathtt{A}^i$, and are finished.

In order to continue, it is essential to understand the relationship between the profile-shaped minors of the matrix $\mathtt{A}$ (i.e. those chosen with α_i rows from each $\mathtt{A}^i$) and certain minors of the matrix $\mathtt{B}$.

Consider a profile-shaped minor of the matrix $\mathtt{A}$. Notice that such a minor of $\mathtt{A}$ inherits the block structure of $\mathtt{A}$ and furthermore has square diagonal blocks. In general, such a minor will consist of some of the reduced rows of the $\mathtt{A}^i$ and some non-reduced rows, which are identical to the rows of $\mathtt{B}$. As an example, consider the profile-shaped minor

$$\mathcal{T}_{\hat{1}12} = -\left|\begin{array}{cc|c|c} 0 & 1 & 0 & 0 \\ a & b & c & d \\ \hline 0 & 0 & 1 & 0 \\ \hline p & q & r & s \end{array}\right|$$

in which the first and third rows are reduced rows of $\mathtt{A}$. In taking the determinant of this matrix, we can eliminate the reduced rows and corresponding columns, as follows:

$$\mathcal{T}_{\hat{1}12} = -\left|\begin{array}{cc|c|c} 0 & 1 & 0 & 0 \\ a & b & c & d \\ \hline 0 & 0 & 1 & 0 \\ \hline p & q & r & s \end{array}\right| = \begin{vmatrix} a & d \\ p & s \end{vmatrix}. \tag{10}$$

The resulting minor on the right of this expression is a minor of the matrix $\mathtt{B}$. Now the *key observation* here is that any 1 entries in the matrix (10) occur, if at all, only in the diagonal blocks. Therefore, when deleting rows and columns from matrix $\mathtt{B}$, we must delete row-column pairs from corresponding row and column blocks. Equivalently, the minors of $\mathtt{B}$ that may occur in this way are precisely those that correspond to a choice of rows from some set of row blocks, and a choice of columns from the correspondingly numbered column blocks. We will call such a minor of $\mathtt{B}$ a *block-principal* minor.

This terminology is derived from the term *principal minor* of a matrix, which is the determinant formed from a square matrix by selecting a sequence σ of rows, the same sequence of columns, and forming the submatrix that consists of elements in the selected rows and columns.

In a *block-principal* minor of a block partitioned matrix, $\mathtt{B} = [\mathtt{B}_{ij}]$, we select rows from a sequence σ^{row} and columns from a sequence σ^{col} such that for each k, σ_k^{row} and σ_k^{col} belong to the same numbered row and column blocks of $\mathtt{B}$.

We thus have the important observation:

The entries of the tensor $\mathcal{A}$ are equal up to well-determined sign to the block-principal minors of the reduced matrix $\mathtt{B} = [\mathtt{B}_{ij}]$.

Thus, if we are able to determine the matrix $\mathtt{B}$ from its block-principal minors, then we are able to determine the projection matrices $\mathtt{A}^i$ from the tensor, which is our goal. As it turns out, this is not quite possible, but it is true up to a certain equivalence, as we shall see. From now on, we concentrate on the computation of the reduced matrix $\mathtt{B}$, given knowledge of its block-principal minors. Our goal will be to fill in the entries of $\mathtt{B}$ one by one.

Step 3. Using 1×1 Determinants The 1×1 block-principal minors of $\mathtt{B}$ are simply the entries in the diagonal blocks of $\mathtt{B}$. Hence we are able to determine these values a, b, g, m, s, w directly from the corresponding entries of the tensor. In particular, we have

$$a = -\mathcal{T}_{\hat{1}11} = 1,$$

$$b = -\mathcal{T}_{\hat{2}11} = 2,$$

$$g = -\mathcal{T}_{\hat{3}21} = 6,$$

$$m = +\mathcal{T}_{\hat{3}31} = 6,$$

$$s = -\mathcal{T}_{\hat{3}12} = 9,$$

$$w = +\mathcal{T}_{\hat{3}13} = 7.$$

So far, we have identified the diagonal blocks of $\mathtt{B}$, which now have the following form.

$$\mathtt{B} = \left[\begin{array}{cc|c|c} 1 & 2 & c & d \\ \hline e & f & 6 & h \\ j & k & 6 & n \\ \hline p & q & r & 9 \\ t & u & v & 7 \end{array}\right].$$

Step 4. Using 2×2 *Determinants* We use the 2×2 block-principal minors of $\mathtt{B}$ next to determine other entries. The most representative case is the computation of the unknown elements in the last two row and column blocks, namely the elements h, n, r and v. To do this, we use as before 2×2 block-principal minors of $\mathtt{B}$ consisting of rows and columns from the last two row and column blocks. In this way, we obtain four equations, bilinear in (h, n) and (r, v), namely

$$hr = 36; \quad hv = 30; \quad nr = 48; \quad nv = 40.$$

As an example, the first of these equations arises from the 2×2 minor of $\mathtt{B}$

$$\begin{vmatrix} g & h \\ r & s \end{vmatrix} = \begin{vmatrix} 6 & h \\ r & 9 \end{vmatrix} = 54 - hr = \mathcal{T}_{\hat{3}22} = 18.$$

The remaining three equations are similarly obtained by considering $\mathcal{T}_{\hat{3}23}$, $\mathcal{T}_{\hat{3}32}$ and $\mathcal{T}_{\hat{3}33}$. The set of 4 equations can be written as a single matrix equation

$$\begin{bmatrix} h \\ n \end{bmatrix} \begin{bmatrix} r & v \end{bmatrix} = \begin{bmatrix} 36 & 30 \\ 48 & 40 \end{bmatrix} = \begin{bmatrix} 3 \\ 4 \end{bmatrix} \begin{bmatrix} 12 & 10 \end{bmatrix}.$$

The matrix on the right has rank 1. In the presence of noise, this might not be the case, but we can use the Singular Value Decomposition (SVD) to enforce this condition, and find a solution. The solution is determined only up to an unknown scale. In this case, we get $(h, n) = \nu(3, 4)$ and $(r, v) = \nu^{-1}(12, 10)$ for some constant ν.

In this way, we were able to find the entries in a single pair of symmetrically opposite blocks of $\mathtt{B}$. In the same way we are able to compute the other symmetrically placed blocks, each up to a different unknown scale factor, obtaining the following form for $\mathtt{B}$ (so far).

$$\mathtt{B} = \left[\begin{array}{cc|c|c} 1 & 2 & \lambda & \mu \\ \hline 1/\lambda & 8/\lambda & 6 & 3\nu \\ 1/\lambda & 6/\lambda & 6 & 4\nu \\ \hline 1/\mu & 5/\mu & 12/\nu & 9 \\ 1/\mu & 3/\mu & 10/\nu & 7 \end{array}\right].$$

Step 5. Reciprocal Scale Equivalence At this point we may exercise the freedom of choosing one solution among the family of equivalent ones, to fix the value of some of the unknown scales. We are free to multiply each $\mathtt{A}^i$ on the right by a diagonal matrix $\operatorname{diag}(\lambda_1 I_{2\times 2}, \lambda_2 I_{1\times 1}, \lambda_3 I_{1\times 1})$ and each $\mathtt{A}^i$ on the left by the constant λ_i^{-1}, where λ_1, λ_2 and λ_3 are arbitrary constants. (In the general case with profile $\alpha = (\alpha_1, \ldots, \alpha_r)$ we would multiply on the right by the matrix $\operatorname{diag}(\lambda_1 I_{\alpha_1 \times \alpha_1}, \ldots, \lambda_r I_{\alpha_r \times \alpha_r})$. This operation preserves the normalized rows of the $\mathtt{A}^i$.

It may be verified that the effect of this operation on the block matrix (9) is to multiply the i-th block column $[\mathtt{B}_{1i}; \mathtt{B}_{2i}; \mathtt{B}_{3i}]$ by λ_i and the corresponding block row $[\mathtt{B}_{i1}, \mathtt{B}_{i2}, \mathtt{B}_{i3}]$ by λ_i^{-1}. This operation leaves the diagonal blocks unchanged, and (as one may verify) also leaves unchanged the values of any block-principal minors. The effect of this operation on an arbitrary block $\mathtt{B}_{ij}$ is to multiply it by $\lambda_i \lambda_j^{-1}$.

We may now exercise this freedom to select the constants that appear in the first row-block. Thus, in our example, we may choose $\lambda = \mu = 1$, leading to a solution

$$\mathtt{B} = \left[\begin{array}{cc|c|c} 1 & 2 & 1 & 1 \\ \hline 1 & 8 & 6 & 3\nu \\ 1 & 6 & 6 & 4\nu \\ \hline 1 & 5 & 12/\nu & 9 \\ 1 & 3 & 10/\nu & 7 \end{array}\right].$$

The value of ν can not be set to a chosen value, but must remain indeterminate at this point. In the next step, we will determine the value of ν.

Step 6. Using 3×3 *Determinants* We want to find the unknown value ν to complete the computation of the matrix $\mathtt{B}$, and hence (up to equivalence) the three camera matrices $\mathtt{A}^i$. To do this, we use 3×3 block-principal minors of $\mathtt{B}$, consisting of one row and column from each block. There are various choices. One such minor is obtained by using rows $(1, 3, 4)$ and columns $(1, 3, 4)$ of $\mathtt{B}$. This gives a minor

$$\begin{vmatrix} 1 & 1 & 1 \\ 1 & 6 & 4\nu \\ 1 & 12\nu^{-1} & 9 \end{vmatrix} = -9 + 4\nu + 12/\nu = \mathcal{T}_{\hat{1}32} = 5.$$

This leads to a quadratic equation $4\nu + 12/\nu - 14 = 0$, or $4\nu^2 - 14\nu + 12 = 0$, which has solutions $\nu = 2$ or $\nu = 3/2$.

Another choice of rows $(1, 3, 5)$ and column $(1, 3, 4)$ gives equations

$$\begin{vmatrix} 1 & 1 & 1 \\ 1 & 6 & 4\nu \\ 1 & 10/\nu & 7 \end{vmatrix} = -11 + 4\nu + 10\nu^{-1} = -\mathcal{T}_{\hat{1}33} = 2$$

which has solutions $\nu = 2$, $\nu = 5/4$. The common solution to these two equations is $\nu = 2$, which must therefore be the correct value. This leads to the final value of

$$\mathtt{B} = \begin{bmatrix} 1 & 2 & 1 & 1 \\ 1 & 8 & 6 & 6 \\ 1 & 6 & 6 & 8 \\ 1 & 5 & 6 & 9 \\ 1 & 3 & 5 & 7 \end{bmatrix}$$

and finally the projection matrices

$$\mathtt{A}^1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 2 & 1 & 1 \end{bmatrix}, \qquad \mathtt{A}^2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 8 & 6 & 6 \\ 1 & 6 & 6 & 8 \end{bmatrix},$$

$$\mathtt{A}^3 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 5 & 6 & 9 \\ 1 & 3 & 5 & 7 \end{bmatrix}.$$

In this computation, we found the value of ν as the common root of two quadratic equations. An alternative is to consider the pair of equations involved here as a pair of simultaneous "linear" equations in the two variables ν and ν^{-1}, ignoring the fact that they are reciprocals. In the above example:

$$\begin{bmatrix} 4 & -14 & 12 \\ 4 & -13 & 10 \end{bmatrix} \begin{bmatrix} \nu \\ 1 \\ \nu^{-1} \end{bmatrix} = 0.$$

The 2×3 matrix has a 1D null-space and dividing by the 2nd entry of any generator allows us to recover ν (and ν^{-1}). In the presence of noise, the two quadratic equations do not have an exact common root, whereas the two linear equations have a solution for ν (though the solution for ν^{-1} may not be the exact reciprocal). A further advantage of this linear solution is that there are (in this case) 8 possible 3×3 minors giving equations in ν and ν^{-1}, corresponding to the 8 trifocal tensor entries $\mathcal{T}_{\hat{i}jk}$ where $i \in \{1, 2\}$ and $j, k \in \{2, 3\}$. The set of 8 equations can then be solved using linear least-squares techniques. This finds the solution that requires the least-squares adjustment to the entries of the tensor (but ignores the condition that ν and ν^{-1} are reciprocals).

Obviously, this example has been carefully chosen for numerical simplicity, but the techniques are generally applicable.

4.2 Projections onto $\mathcal{P}^1$

The techniques developed above for the trifocal tensor are sufficient to solve most cases. Unfortunately there is a further complication in the case where all the projections are onto $\mathcal{P}^1$.

We choose an example of four projections $\mathcal{P}^3 \to \mathcal{P}^1$, with projection matrices equivalent to

$$\mathtt{A}^1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 2 & 3 & 4 \end{bmatrix}; \qquad \mathtt{A}^2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 8 & 6 & 6 \end{bmatrix};$$

$$\mathtt{A}^3 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 6 & 6 & 8 \end{bmatrix}; \qquad \mathtt{A}^4 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 5 & 3 & 9 \end{bmatrix}.$$

In this case there is only one possible profile (1, 1, 1, 1) and the corresponding tensor $\mathcal{A}_{ijkl}$ has $2 \times 2 \times 2 \times 2 = 16$ elements.

The reduced matrix B corresponding to these projection matrices $\mathtt{A}^i$ is equal to

$$\mathtt{B} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 8 & 6 & 6 \\ 1 & 6 & 6 & 8 \\ 1 & 5 & 3 & 9 \end{bmatrix} \tag{11}$$

and all the blocks of this matrix have dimensions 1×1. The elements of the tensor are equal (up to sign) to the block-principal minors of B. Since all the blocks consist of a single element the block-principal minors are just the standard principal minors in this case.

In our problem, we do not know this matrix (11), but we do know the corresponding tensor, whose entries (up to sign) are the principal minors of this matrix. Thus, we assume that we know any of the principal minors of (11), and our task is to find the matrix B itself.

Let us start with a symbolic value for B, given by

$$\mathtt{B} = \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ j & k & l & m \\ n & p & q & r \end{bmatrix}.$$

Proceeding as with the first steps of the algorithm of Sect. 4.1, we arrive at a form for B as follows.

$$\mathtt{B} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 8 & g & h \\ 3 & 36/g & 6 & m \\ 4 & 30/h & 24/m & 9 \end{bmatrix}.$$

In the reciprocal scaling step (step 5) of the algorithm, we made a choice of the scale factors. We assume that the choice was made so as to set the first row of B to all 1s, though other choices are possible, and as can be seen, this choice leads to a different solution from (11).

Using the 3×3 *Minors* To establish the values of the remaining values g, h and m, we consider 3×3 principal minors. The first minor consisting of the first 3 rows and columns gives rise to a quadratic equation $-24 + 3g + 72g^{-1} = 6$. We obtain two solutions: $g = 6$ and $g = 4$. Unfortunately, we are not able to obtain a second equation for g, as we did in the trifocal tensor case. The difference is that all the blocks of $\mathtt{B}$ have dimension 1×1; we need a block with at least two elements to obtain a second equation.

There is no way to choose between the two solutions, so we need to consider them both. First, consider the solution $g = 4$. This leads to

$$\mathtt{B} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 8 & 4 & h \\ 3 & 9 & 6 & m \\ 4 & 30/h & 24/m & 9 \end{bmatrix}.$$

We may obtain three other equations involving the remaining variables h and m by taking the 3×3 principal minors of $\mathtt{B}$ omitting the first, second and third rows and columns. The three equations and their solutions are:

$$348 - 216(m/h)^{-1} - 120(m/h) = 0$$

$$\Rightarrow \quad m/h = 2; \quad m/h = 9/10,$$

$$36 - 72m^{-1} - 4m = 0 \quad \Rightarrow \quad m = 3; \quad m = 6,$$

$$32 - 60/h - 4h = 0 \quad \Rightarrow \quad h = 3; \quad h = 5.$$

Each of these equations has two solutions, but there is only one consistent solution, namely $h = 3$; $m = 6$. This gives the following solution for $\mathtt{B}$.

$$\mathtt{B}_1 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 8 & 4 & 3 \\ 3 & 9 & 6 & 6 \\ 4 & 10 & 4 & 9 \end{bmatrix}. \tag{12}$$

The other solution $g = 6$ leads via a similar set of equations to solutions $h = 5$; $m = 3$, and hence a second solution

$$\mathtt{B}_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 8 & 6 & 5 \\ 3 & 6 & 6 & 3 \\ 4 & 6 & 8 & 9 \end{bmatrix}. \tag{13}$$

The Transpose Solution Note that the two solutions are apparently dissimilar, and neither is the same as the original "true" value of $\mathtt{B}$. However, note that the second solution (13) is the transpose of the original value given in (11). In addition, multiplying $\mathtt{B}_1$ given in (12) on the right by $\mathrm{diag}(1, 2, 3, 4)$ and on the left by its inverse yields the original "true" value of $\mathtt{B}$ in (11). Thus, the two solutions that we have obtained are equivalent to (11) and its transpose.

In fact, it is clear that a matrix and its transpose yield the same set of principal minors. But we demonstrated in the above computation that there are just two possible equivalence classes of solutions for $\mathtt{B}$ given the set of all its principal minors. These must therefore be the matrix itself, and its transpose. It may seem optimistic to call the above arguments a proof, but in fact, the above computation contains all the ingredients of a complete proof. The general case will be considered more formally (and briefly) later. Effectively, we have shown the following result.

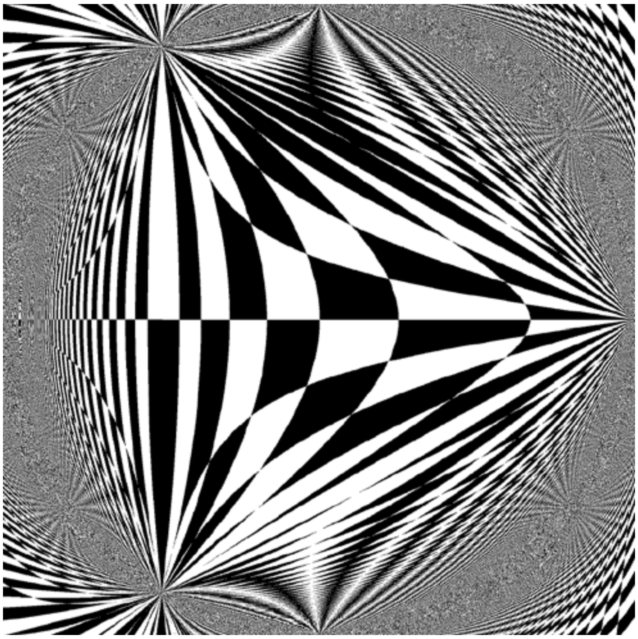

Fig. 2 An alternative reconstruction of a simple checker-board pattern, reconstructed from projections of the plane onto three 1-dimensional images. The two reconstructions (this and the checker-board) are related via a Cremona transformation

Theorem 4.3 *For a generic square matrix, the set of principal minors determines the matrix up to transposition and reciprocal scaling.*

2D Example For a simpler example, consider a set of three projections from $\mathcal{P}^2$ to $\mathcal{P}^1$ given by matrices

$$\mathtt{A}^1 = \begin{bmatrix} 1 & 0 & 0 \\ a & b & c \end{bmatrix}; \qquad \mathtt{A}^2 = \begin{bmatrix} 0 & 1 & 0 \\ d & e & f \end{bmatrix};$$

$$\mathtt{A}^3 = \begin{bmatrix} 0 & 0 & 1 \\ g & h & j \end{bmatrix}$$

and an alternative set

$$\mathtt{A}'^1 = \begin{bmatrix} 1 & 0 & 0 \\ a & d & g \end{bmatrix}; \qquad \mathtt{A}'^2 = \begin{bmatrix} 0 & 1 & 0 \\ b & e & h \end{bmatrix};$$

$$\mathtt{A}'^3 = \begin{bmatrix} 0 & 0 & 1 \\ c & f & j \end{bmatrix}.$$

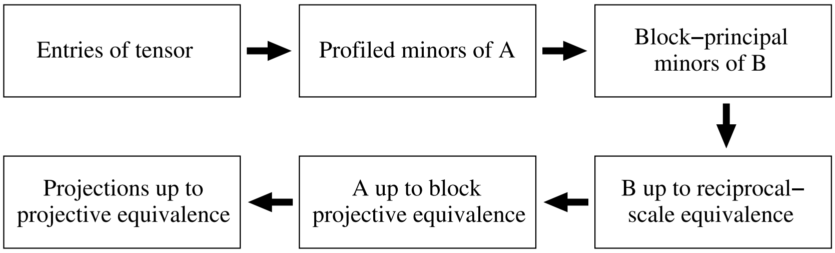
Entries of tensor
Profiled minors of A
Block−principal minors of B
B up to reciprocal−scale equivalence
A up to block projective equivalence
Projections up to projective equivalence

Fig. 3 The stages of the recipe presented in this paper, for retrieving the projection matrices from the Grassmann tensor

Since the corresponding reduced matrices are

$$\mathtt{B}=\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & j \end{bmatrix}$$

and its transpose, the corresponding trifocal tensors for these two sets of matrices are the same, and the two configurations can not be distinguished by the corresponding point projections. In fact, one may easily verify that if $\mathbf{X}=(x,y,z)$ in homogeneous coordinates, and

$$\mathbf{X}'=\begin{bmatrix} x(dg\,x+dh\,y+fg\,z) \\ y(gb\,x+hb\,y+ch\,z) \\ z(cd\,x+fb\,y+cf\,z) \end{bmatrix}$$

then $\mathtt{A}^i\mathbf{X}=\mathtt{A}'^i\mathbf{X}'$ for all i. The mapping from $\mathbf{X}$ to $\mathbf{X}'$ is a quadratic mapping, or Cremona transformation. There are in general three fundamental (singular) points, namely the centers of projection in $\mathcal{P}^2$. Figure 3 shows the effect of applying a mapping of this type to a simple checker-board pattern. The resulting pattern is an alternative interpretation of the checker-board pattern, derived from reconstruction of the plane from point correspondences in three views.

Genericity We state our assumption of genericity. All projections from $\mathcal{P}^n$ to $\mathcal{P}^m$ are assumed to be "generic", which means in effect that improbable special cases are ruled out. Any polynomial expression in the coordinate representation of the projections, or related points may be assumed to be non-zero, unless it is always (identically) zero. Thus the results we prove will hold, except on a set of positive co-dimension (hence of measure zero).

The result of Theorem 4.3 is true only for generic matrices, that is for all matrices except for some exceptions lying in a nowhere dense exceptional set. Exceptions do exist for which more than two solutions are possible. For instance, all upper triangular matrices with the same diagonal have the same set of principal minors. However, upper triangular matrices form an $(n(n+1)/2)$-dimensional subspace of the n^2-dimensional space of all $n\times n$ matrices.

Throughout these proofs or examples, in arguing the existence of a single solution (or a pair of solutions), we relied on the fact that certain elements of the tensor were non-zero, or that sets of polynomials did not by chance have more than one common solution. In all such cases, the tensor entries or polynomials involved relied on independent elements of the matrix $\mathtt{B}$. Thus, accidents such as zero-elements of the tensor, or polynomials with more than one common solution could be avoided by small perturbations of elements of $\mathtt{B}$, and hence the projection matrices $\mathtt{A}^i$. Thus, the results hold for generic though not all projections.

It is outside the scope of this paper to investigate exceptional configurations where more than the generic number of solutions exist.

5 General Proof

In this section we give a more general description of the method for reconstruction presented in this paper. The recipe is summarized in Fig. 3 which shows the stages of the construction.

We assume that there are sufficiently many such projections that $\sum_{i=1}^r m_i\geq n+1$. Let $(\alpha_1,\dots,\alpha_r)$ be an ordered partition of $(n+1)$ with the property that $1\leq\alpha_i\leq m_i$. We may form square matrices of dimension $(n+1)\times(n+1)$ by selecting exactly α_i rows from each matrix $\mathtt{A}^i$. We may then take the determinant of such a square matrix. Of course, we may select α_i rows from each $\mathtt{A}^i$ in many different ways—to be exact, there are $\prod_{i=1}^r C_{m_i+1}^{\alpha_i}$ ways of doing this, and that many such subdeterminants of $\mathtt{A}$ corresponding to the given partition.

We now state the main theorem of this part of the paper.

Theorem 5.1 *Let $\mathtt{A}$ be a generic matrix with blocks $\mathtt{A}^i$; $i=1,\dots,r$, of dimension $(m_i+1)\times(n+1)$, and let $(\alpha_1,\dots,\alpha_r)$ be any fixed ordered partition of $n+1$. If at least one m_i is greater than one, then the matrix $\mathtt{A}$ is determined up to block projective equivalence by the collection of all its minors, chosen with α_i rows from each $\mathtt{A}^i$. If all $m_i=1$, then there are two equivalence classes of solutions.*

We refer to the partition $(\alpha_1,\dots,\alpha_r)$ as the *profile* of the minors. Thus, the theorem states that the matrix $\mathtt{A}$ is determined up to projective equivalence by its collection of minors with a given fixed profile.

Proof The strategy of the proof is to provide an explicit method by which $\mathtt{A}$ may be reconstructed (up to equivalence) from knowledge of its α-minors.

By assumption, there exists at least one non-zero minor, and without loss of generality (by rearranging rows if required), this may be chosen to belong to the submatrix of $\mathtt{A}$ in which the first α_i rows are chosen from each $\mathtt{A}^i$. Let this submatrix be denoted by $\mathtt{G}$. Choosing $\mathtt{H} = \mathtt{G}^{-1}$, we may transform any solution $\mathtt{A}$ into an equivalent solution $\mathtt{AH}$ in which the matrix $\mathtt{G}$ is replaced by the identity matrix.

After this transformation, the form of the matrix $\mathtt{A}$ is simplified. The first α_i rows from each block are known, consisting of zeros, except for one unit element in each such row. We refer to these rows of $\mathtt{A}$ as the *reduced* rows. The elements of the remaining rows of $\mathtt{A}$ are still to be determined. We show that they can be determined (up to block projective equivalence) from other minors of the matrix.

We consider a finer block decomposition of the matrix $\mathtt{A}$ into blocks indexed by (i, j) where the block $\mathtt{A}^{ij}$ has dimension $(m_i + 1) \times \alpha_i$ as shown:

$$\begin{bmatrix} \mathtt{A}^{11} & \dots & \mathtt{A}^{1r} \\ \vdots & \ddots & \vdots \\ \mathtt{A}^{r1} & \dots & \mathtt{A}^{rr} \end{bmatrix}.$$

The first α_i rows of each such $\mathtt{A}^{ij}$ are reduced, so

$$\mathtt{A}^{ii} = \begin{bmatrix} \mathtt{I} \\ \mathtt{B}^{ii} \end{bmatrix} \quad \text{and} \quad \mathtt{A}^{ij} = \begin{bmatrix} 0 \\ \mathtt{B}^{ij} \end{bmatrix} \quad \text{for } i \neq j.$$

The reduced rows of $\mathtt{A}$ form an identity matrix, having unit determinant. Let $\mathtt{B}$ be the matrix obtained from $\mathtt{A}$ by removing the reduced rows. Then $\mathtt{B}$ has the same type of block structure as $\mathtt{A}$. We investigate the relationship between minors of $\mathtt{A}$ and those of $\mathtt{B}$.

By α-minors of $\mathtt{A}$, we mean those chosen according to the given profile, with α_i rows from each $\mathtt{A}^i$. These minors are in one-to-one correspondence with (and equal up to a well-determined sign to) the block-principal minors of $\mathtt{B}$.

Corresponding to the concept of projective equivalence of the set of projection matrices $\mathtt{A}^i$, we define a different type of equivalence relation on block matrices of the form $\mathtt{B} = [\mathtt{B}^{ij}]$, where $i, j = 1, \dots, r$.

Definition 5.2 Two matrices $\mathtt{B} = [\mathtt{B}^{ij}]$ and $\bar{\mathtt{B}} = [\bar{\mathtt{B}}^{ij}]$ will be called *reciprocal-scale equivalent* if there exist non-zero scalars λ_i such that $\bar{\mathtt{B}}^{ij} = \lambda_i \lambda_j^{-1} \mathtt{B}^{ij}$.

We will show that the block-principal minors of $\mathtt{B}$ determine $\mathtt{B}$ up to reciprocal-scale equivalence. This is sufficient, however, because reciprocal-scale equivalence of two matrices $\mathtt{B}$ and $\bar{\mathtt{B}}$ implies that the corresponding matrices $\mathtt{A}^i$ and $\bar{\mathtt{A}}^i$ are projectively equivalent, which is all we need. This is true because

$$\bar{\mathtt{A}}^i = \lambda_i \mathtt{A}^i \mathrm{diag}(\lambda_1^{-1} \mathtt{I}_{\alpha_1}, \dots, \lambda_r^{-1} \mathtt{I}_{\alpha_r})$$

follows from the fact that $\bar{\mathtt{B}}^{ij} = \lambda_i \lambda_j^{-1} \mathtt{B}^{ij}$.

The proof of Theorem 5.1 will be completed therefore by proving the following lemma.

Lemma 5.3 *A matrix $\mathtt{B} = [\mathtt{B}^{ij}]$ is determined up to reciprocal-scale equivalence by its collection of block-principal minors.*

In fact, it will be sufficient to consider only minors up to dimension 3 × 3.

Step 1. The 1×1 *Minors* The 1×1 block-principal minors are nothing other than the elements of the diagonal blocks $\mathtt{B}^{ii}$, and hence these 1×1 minors determine the diagonal blocks.

Step 2. The 2×2 *Minors* As shown in the example of Sect. 4.1, the 2×2 block-principal minors of $\mathtt{B} = [\mathtt{B}^{ij}]$ determine the entries of the blocks $\mathtt{B}^{ij}$ and $\mathtt{B}^{ji}$ for $i \neq j$ up to multiplication by scale factors μ_{ij}, where $\mu_{ij} = \mu_{ji}^{-1}$.

The only thing that can go wrong here is that, when solving the bilinear problems, the right-hand sides are all zero and so one factor is zero while the other is undetermined. But this a non-generic case, which we have excluded.

Step 3. Normalization of the First Row and Column Blocks The requirement to reconstruct $\mathtt{B}$ only up to reciprocal-scale equivalence allows us to set the unknown scale factors in the first row and column blocks of $\mathtt{B}$. It remains then to determine the scale factors for the other blocks.

Step 4. Consistent Choice of Scale Factors Once we have chosen the scale factors μ_{1j} and μ_{i1}, then the rest of the scale factors μ_{ij}, and hence the matrix $\mathtt{B}$, will be determined uniquely. The exception is when $m_i = 1$ for all i, that is all the projections are onto $\mathcal{P}^1$. In this case, there are two solutions. Consider block-principal 3×3 minors of $\mathtt{B}$, in which one row is taken from the first row block, and one row each from each of two other row blocks. Columns are chosen from the corresponding blocks. The 3×3 minor is of the form

$$\begin{vmatrix} a & b & c \\ d & e & \mu f \\ g & \mu^{-1} h & k \end{vmatrix}. \tag{14}$$

Equating this determinant with its known value (the block-principal minor of $\mathtt{B}$, assumed known) gives a quadratic equation in μ, for which there are generically two solutions for μ. In most situations we may obtain a second equation for the same μ which will also have two solutions, but there will be only one solution common to both equations.

To see this, we need to make an assumption that the first projection matrix $\mathtt{A}^1 = [\mathtt{A}^{11} \; \ldots \; \mathtt{A}^{1r}]$ has more than two rows—its dimension is $m_1 + 1 \geq 3$. This is possible without loss of generality provided that there exists at least one projection matrix with at least three rows, for it may be chosen as the first. The number of rows chosen from $\mathtt{A}^1$ is α_1 which is in the range $1 \leq \alpha_1 \leq m_1$. Now, suppose that the rows and columns of (14) are chosen from the row and column blocks numbered 1, i and j. Thus, the entries of (14) are drawn from the block matrix

$$\begin{bmatrix} \mathtt{B}^{11} & \mathtt{B}^{1i} & \mathtt{B}^{1j} \\ \mathtt{B}^{i1} & \mathtt{B}^{ii} & \mathtt{B}^{ij} \\ \mathtt{B}^{j1} & \mathtt{B}^{ji} & \mathtt{B}^{jj} \end{bmatrix} . \tag{15}$$

Now, the dimension of the matrix $\mathtt{B}^{ij}$ is $(m_i + 1 - \alpha_i) \times \alpha_j$. Specifically, $\mathtt{B}^{i1}$ has dimension $(m_i + 1 - \alpha_i) \times \alpha_1$, and $\mathtt{B}^{1j}$ has dimension $(m_1 + 1 - \alpha_1) \times \alpha_j$. However, since $1 \leq \alpha_1 \leq m_1$ and $m_1 > 1$, it must follow that either $\alpha_1 > 1$ or $m_1 + 1 - \alpha_1 > 1$. Thus, either $\mathtt{B}^{i1}$ has at least two columns, or $\mathtt{B}^{1j}$ has at least two rows. In either case, there is more than one way of selecting rows and columns from (15) to obtain a submatrix of the form (14). Each such choice will give a different equation for μ. Generically, there will only be one common solution to the two equations.

The exception to this argument is the case where each $m_i = 1$. In this case, it is possible to find only one quadratic equation in any one of the unknown scale factors μ_{ij}. Recall that we may assume (calling upon our freedom afforded by reciprocal-scale equivalence) that $\mu_{1j} = \mu_{j1} = 1$ for all i and j. As shown in the example of Sect. 4.1 there will then be two possible values for any other given μ_{ij}. However, once one of these values (say μ_{23}) is chosen from the two possible values, then the choice of all the other values μ_{ij} is forced by the requirement of consistency among the solutions arising from the other 3 × 3 principal minors. As was shown in Sect. 4.2, the two possible solutions for $\mathtt{B}$ in this case are (up to reciprocal-scale equivalence) transposes of each other.

The proof is now complete, since we have given a specific method for reconstructing the matrix $\mathtt{B}$ (up to reciprocal-scale equivalence) given its set of block-principal minors. □

6 Practical Algorithmic Considerations

The main result of this paper is the theoretical existence proof of the possibility of reconstruction from projections between projective spaces. Nevertheless, the algorithm for reconstruction outlined in Sect. 4.1 represents the basis for a practical method for reconstruction. In this section, we describe some of the engineering decisions necessary to make this into a good quality functioning algorithm.

6.1 Normalization

As with any of the linear tensor-based algorithms for projective reconstruction, it is necessary to normalize the point coordinates before computing the tensor. In the most important case, the projections will be from a space $\mathcal{P}^n$ into $\mathcal{P}^2$, a normal 2-dimensional image. In this case, we use the simple normalization technique described in Hartley (1997). This involves applying a scale change and translation to image coordinates, and subsequently correcting the computed projected matrices accordingly.

6.2 Row Permutation

In step 1 of the algorithm, we choose an element of the tensor and scale the tensor so that this entry is 1. It is important that the element of $\mathcal{A}$ we choose should be non-zero. In the description of the algorithm, it was assumed that this element could be conveniently chosen so that it corresponded to the leading rows of the camera matrices $\mathtt{A}^i$, and that hence the matrix formed by these rows could be inverted. If another element is chosen, it is simplest to rearrange the tensor, and the projection matrices so that this will be the case. For instance, for the example given with profile (2, 1, 1), the tensor should be rearranged so that the non-zero chosen element is $\mathcal{A}_{\hat{3}11}$. In doing this, it is important (and tricky) to get the signs correct.

Let π be a permutation of the integers $1, \ldots, m$, and τ a sequence of integers $(i_1, i_2, \ldots, i_r)$, in the range $1, \ldots, m$, We may define the sequence $\pi(\tau) = (\pi(i_1), \ldots, \pi(i_r))$. Now, suppose that the rows of each projection matrix $\mathtt{A}^i$ are rearranged according to a permutation π_i to form a new matrix $\mathtt{A}'^i$ defined by $\mathtt{A}'^i_k = \mathtt{A}^i_{\pi(k)}$, where $\mathtt{A}'^i_k$ means the k-th row of $\mathtt{A}'^i$. According to (6), the tensor corresponding to the new set of matrices is given by

$$\begin{aligned} \mathcal{A}'_{\tau_1 \ldots \tau_r} &= \mathrm{sign}(\tau_1) \ldots \mathrm{sign}(\tau_r) \; |\mathtt{A}'_{\tau_1 \ldots \tau_r}| \\ &= \mathrm{sign}(\tau_1) \ldots \mathrm{sign}(\tau_r) \; |\mathtt{A}_{\pi(\tau_1) \ldots \pi(\tau_r)}| \, . \end{aligned}$$

The elements of each permutation $\pi(\tau_i)$ are not in general in ascending order. We define σ_i to be the sequence of integers $\pi(\tau_i)$ rearranged into ascending order. Now, we correspondingly rearrange the rows of the matrix $\mathtt{A}_{\pi(\tau_1) \ldots \pi(\tau_r)}$. This will change the sign of the determinant in the above expression. Continuing the calculation, we get

$$\mathcal{A}'_{\tau_1 \ldots \tau_r} = \prod_{i=1}^{r} \mathrm{sign}(\tau_i) \mathrm{sign}(\pi(\tau_i)) \mathrm{sign}(\sigma_i) |\mathtt{A}_{\sigma_1 \ldots \sigma_r}|$$

$$= \prod_{i=1}^{r} \mathrm{sign}(\tau_i)\mathrm{sign}(\pi(\tau_i))\mathcal{A}_{\sigma_1\ldots\sigma_r}. \qquad (16)$$

This gives a formula for the tensor corresponding to the rearranged projection matrices in terms of the entries of the original tensor. It suggests an algorithm to compute the projection matrices, starting from the tensor $\mathcal{A}$, as follows:

1. Select permutations π_i for $i = 1, \ldots, r$ and compute the *reordered tensor* $\mathcal{A}'$ using (16).
2. Use the algorithm of Sect. 4.1 to compute the corresponding projection matrices $\mathtt{A}'^i$ from tensor $\mathcal{A}'$.
3. Rearrange the rows of each $\mathtt{A}'^i$ to get the projection matrices $\mathtt{A}^i$, according to the formula $\mathtt{A}^i_{\pi(k)} = \mathtt{A}'^i_k$.

The result of this algorithm depends significantly on the permutation being used, but in a somewhat unpredictable way. The quality of the reconstruction may be evaluated by computing the associated algebraic or geometric error. We have found that best results are achieved by trying a few different permutations, and choosing the best result.

6.3 Algebraic Minimization

An improved estimate of the projection matrices may be obtained using a technique described in Hartley (1998a) for computing the quadrifocal tensor. This involves finding the geometrically consistent tensor that minimizes algebraic error, as defined in Hartley (1998b).

Specifically, the linear method of computing the tensor (either the classical trifocal or quadrifocal tensors, or the Grassmann tensors discussed here) involves finding the least-squares solution of a set of linear equations of the form $\mathtt{M}\mathbf{t} = 0$, where $\mathtt{M}$ is a matrix computed from the point or subspace correspondences, and $\mathbf{t}$ is a vector containing the elements of the tensor. The solution is the vector $\mathbf{t}$ of unit norm that minimizes $\|\mathtt{M}\mathbf{t}\|$.

As is well known, the tensor $\mathcal{A}$ corresponding to the solution $\mathbf{t}$ is not a *well-formed* tensor, in the sense that there exist projection matrices $\mathtt{A}^i$ corresponding to $\mathcal{A}$ according to the relation (6). The main weakness of the tensor-based reconstruction methods is that the tensor obtained by linear methods does not satisfy the constraints implied by this relationship. Improved results may be obtained by minimizing the algebraic error $\|\mathtt{M}\mathbf{t}\|$ over all $\mathbf{t}$ corresponding to well-formed tensors $\mathcal{A}$. This is a non-linear optimization problem. In order to carry out this optimization, we need a method of parameterizing the set of all well-formed Grassmann tensors for a given profile. This is provided from the entries of the reduced matrix (9). Consider the function f defined as follows.

$$f : \mathtt{B} \mapsto \mathtt{A} \mapsto \mathcal{A} \mapsto \mathbf{t} \mapsto \mathtt{M}\mathbf{t}/\|\mathbf{t}\|.$$

Here, the reduced matrix $\mathtt{B}$ is completed to form the stack of all projection matrices, $\mathtt{A}$. The corresponding tensor $\mathcal{A}$ is computed according to (6) and its elements are made into a vector $\mathbf{t}$. Finally, $\mathbf{t}$ is normalized to unit length and multiplied by the matrix $\mathtt{M}$ to give a vector consisting of the algebraic errors corresponding to each correspondence. The optimization task is to minimize $\|\mathtt{M}\mathbf{t}\|/\|\mathbf{t}\|$ over all the entries of matrix $\mathtt{B}$. This is accomplished using the Levenberg-Marquardt algorithm, starting from the initial reduced matrix $\mathtt{B}$ computed using the non-iterative algorithm described in this paper.

One small refinement of this algorithm will save some computation time. We replace the matrix $\mathtt{M}$ by its orthogonal row-reduction $\mathtt{M}'$, This is computed by taking the singular value decomposition $\mathtt{M} = \mathtt{UDV}^\top$ and then defining $\mathtt{M}' = \mathtt{DV}^\top$. It is easily seen that $\mathtt{M}^\top\mathtt{M} = \mathtt{M}'^\top\mathtt{M}'$, and so $\|\mathtt{M}\mathbf{t}\| = \|\mathtt{M}'\mathbf{t}\|$ for any vector $\mathbf{t}$. If the number of correspondences is large, then $\mathtt{M}'$ will be considerably smaller than $\mathtt{M}$, and so the amount of computation is substantially decreased.

6.4 Bundle Adjustment

A final bundle adjustment will give the best achievable solution. This involves a large-scale non-linear optimization over the parameters of the reduced projection matrix $\mathtt{B}$ (which provide a convenient parametrization of the projection matrices), as well as the coordinates of the points $\mathbf{X}_j$ being matched. The initial coordinates of these points may be computed using a linear triangulation procedure.

Non-linear optimization seeks to minimize the reprojection error in each of the images. In the general case, this is the Euclidean distance between the projected point and an affine subspace in $\mathcal{P}^{m_i}$, represented by $\mathtt{S}_i$.

The suggested overall method for carrying out the reconstruction consists of three steps:

1. From affine subspace correspondences, compute the Grassmann tensor linearly, and extract the projection matrices $\mathtt{A}^i$ using the non-iterative method outlined in this paper (Sect. 4.1). For best performance two or three iterations of the permutation method of Sect. 6.2 should be used and the best result retained.
2. Refine this initial solution using non-linear optimization to minimize the algebraic error, as in Sect. 6.3.
3. Obtain a best-achievable solution through full-scale bundle-adjustment minimizing geometric reprojection error.

The second step may be skipped here, but we have found that it leads to better results by providing a better initial solution for the final bundle adjustment. In addition, it is easy to implement, and runs fast.

7 Experimental Results

The algorithm of this paper has been implemented for projections between spaces of arbitrary dimension. Some results have been given in Hartley and Vidal (2008) in the context of the application to dynamic perspective structure and motion. In this paper, we show results for projections from $\mathcal{P}^3$ to $\mathcal{P}^2$, and compare against the standard algorithms involving the bifocal, trifocal and quadrifocal tensors. Each of these problems has been studied carefully and good quality methods are available. The algorithm given in this paper is a general one, not optimized to any of the particular problems addressed by the classical tensor algorithms. Nevertheless, it will be seen that it performs nearly as well, and has the advantage of offering a single approach that works in for all dimensions, and any number of views within the scope of tensor-based reconstruction.

The experiments we carry out are for synthetic data generated as follows. A set of 50 points are randomly generated within a cube centred at the origin, with half-side length equal to one unit. Between two and four cameras are placed approximately three units distance from the origin, and oriented roughly towards the origin (but with non-incident principal rays). Thus, the points subtend an angle of around 60° in each image—roughly equivalent to the field of view of a 35 mm camera. The angle between the directions of view varies from 0 to about 40° with average about 30°. Thus, the imaging geometry approximates a set of snapshots taken from different viewing directions ranging over about 40°.

The experiment is carried out with 50 different random configurations, and the calculation is repeated 50 times with different instantiations of Gaussian image noise with varying variance. We report results for a noise level of 0.01 × focal-length. Thus, with the given viewing geometry, if the image is 1000 × 1000 pixels, then the noise level is about 10 pixels. Thus, we consider very high noise levels. Despite this, the algorithm is shown to perform close to perfectly in most cases.

The usual way to report the results of such reconstructions is to show graphs of the root-mean-squared (RMS) reprojection error as a function of noise. In this case such graphs are not informative, since even with high levels of noise, the algorithm performs perfectly for most cases. Occasionally, the algorithm fails to find the best minimum, in which case the reprojection error can be quite large, thus badly skewing the RMS error estimate. Instead, we focus on one noise level (0.01 × focal-length) and run 2500 tests. In each graph, we show the results of one algorithm, and plot the RMS reprojection error (over 50 points) for each of the 2500 tests. The horizontal axis represents the test number, and the vertical axis the RMS reprojection error. The tests are ordered according to the resulting RMS reprojection error.

Expected Reprojection Error One may compute the expected residual error for the optimum solution to these reconstruction problems using the formula

$$E(\delta_{\text{opt}}) = \sigma\ \sqrt{1 - d/N}$$

where N is the number of measurements and d is the number of degrees of freedom. In this case with p points, r views, this gives $N = 2pr$ and $d = 11r + 3p - 15$. Thus, with $\sigma = 0.01$, we expect $E(\delta_{\text{opt}})$ to equal 0.0046, 0.0066 and 0.0074 for $r = 2, 3, 4$.

7.1 Two-View Tests

We compare various two-view algorithms, described in Hartley and Zisserman (2004):

1. 2–8-point. The normalized 8-point algorithm (Hartley and Zisserman 2004, Algorithm 11.1, p. 282).
2. 2-algebraic. The algebraic minimization algorithm is described in Hartley and Zisserman (2004), Sect. 11.3 and consists of the first two steps of Algorithm 11.2.
3. 2-iterative. This algorithm is described in Hartley and Zisserman (2004), Algorithm 11.2. It finds the fundamental matrix that minimizes algebraic error as defined in Sect. 6.3, while satisfying the singularity constraint $\det(\mathtt{F}) = 0$.
4. 2-Grassmann. This is the non-iterative algorithm described in this paper.
5. 2-Grassmann-algebraic. This starts from the 2-Grassmann algorithm, and follows it with iteration to minimize algebraic error, as described in Sect. 6.3.

The algorithm 2-iterative is most closely comparable with 2-Grassmann-algebraic, since both seek to minimize the algebraic error over all well-formed fundamental matrices. We also compare the results of bundle-adjustment initiated by the results of all these algorithms.

The results of the comparison are shown in Fig. 4. If the algorithm achieves the optimal solution, then the graph of residuals takes on the characteristic shape appearing in the right-hand graph of Fig. 4. The residuals will be distributed about the optimal expected value (in this case 0.0046).

7.2 Three-View Tests

We compare the algorithm of this paper with a number of different 3-view algorithms described in Hartley and Zisserman (2004), pp. 395–397. The algorithms tested are as follows:

1. 3-linear. This is the “Algebraic Minimization” algorithm described in Hartley and Zisserman (2004), p. 395. It is a two-step linear algorithm that computes first the epipoles, and then the remaining elements of the camera matrices from the trifocal tensor. It minimizes the algebraic error constrained by the initial estimated positions of the epipoles.

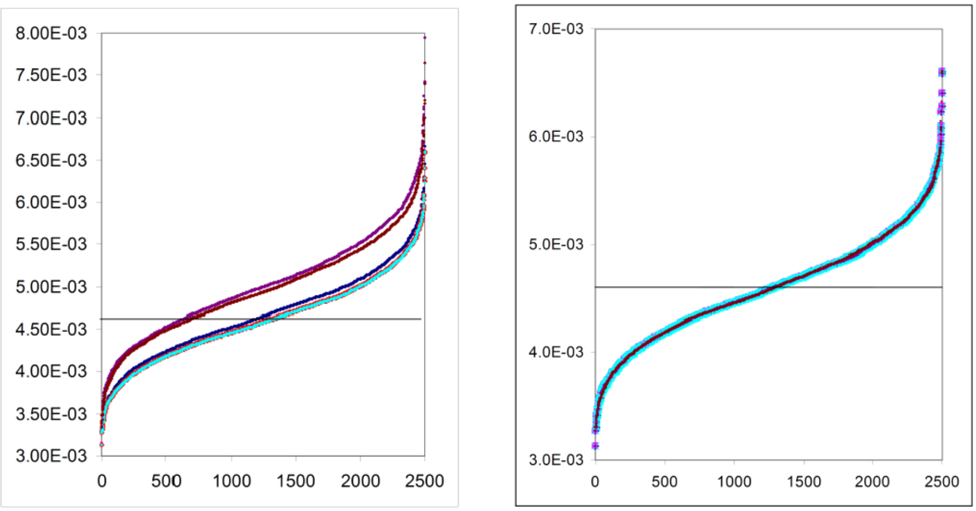

Fig. 4 Comparison of two-view algorithms. On the *left* are the results of the different algorithms without final bundle adjustment. On the *right* are the results after bundle adjustment. The graphs show the reprojection error for 2500 separate tests, and the results are sorted (separately for each method) according to increasing reprojection error. On the *left*, the algorithms are, from top to bottom, 2-Grassmann, 2-Grassmann-algebraic, followed by the other algorithms, largely indistinguishable. After bundle-adjustment (*right*) there is no distinguishable difference between the algorithms. All algorithms achieve the global minimum. The horizontal line indicates the expected RMS error of 0.0046

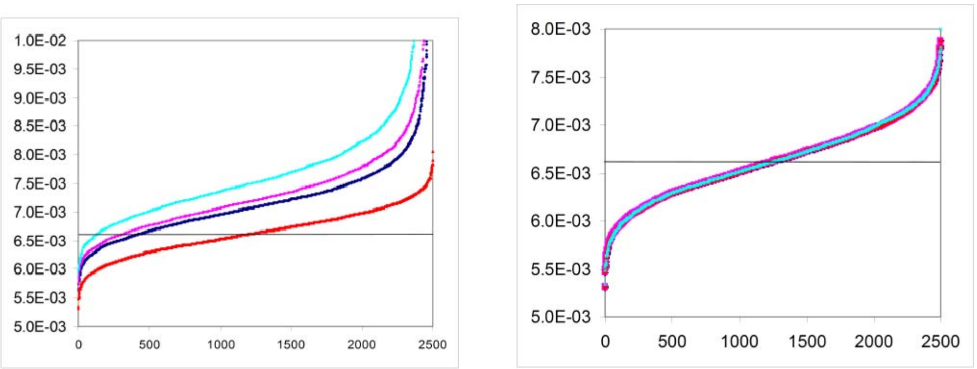

Fig. 5 Comparison of three-view algorithms. *Left* before, *right* after bundle adjustment. From top to bottom (*left*) are 3-Grassmann-algebraic, 3-linear, 3-iterative and for reference, 3-iterative-b, which represents the optimal bundle-adjusted solution. The residual errors for the 3-Grassmann-algebraic algorithm are about 5% to 10% larger than those for the specialized trifocal tensor algorithms. Note also that all three algorithms result in large residuals (go off the top of the graph) for some small percentage of cases. Nevertheless, all the algorithms find solutions close enough to the minimum to give optimal results after bundle adjustment

2. 3-iterative. This is an iterative algorithm (Hartley and Zisserman 2004, p. 396) that minimizes algebraic error by adjusting the positions of the epipoles. This algorithm, followed by bundle-adjustment is recommended in Hartley and Zisserman (2004) as the preferred "gold-standard" method for computing the trifocal tensor.
3. 3-Grassmann and 3-Grassmann-algebraic are the same as for the two-view case.

In Fig. 5 are shown the results of the comparison of the different 3-view algorithms.

In Fig. 6. we also show results illustrating the effect of the permutation (Sect. 6.2) and algebraic minimization (Sect. 6.3) methods. These results show that the algebraic minimization step is essential for good quality results.

7.3 Four-View Tests

We compare our Grassmann algorithms with various classical quadrifocal tensor algorithms, described in Hartley (1998a).

1. 4-linear. This algorithm is a non-iterative algorithm for computation of the quadrifocal tensor, given in Hartley (1998a), Sect. 3.1.
2. Two iterative algorithms are described in Hartley (1998a), Sect. 3.2. We refer to them as 4-iterative-1, 4-iterative-2.

The second of these algorithms minimizes algebraic error over all reduced quadrifocal tensors. The first one, algorithm 4-iterative-1, is faster but does not do a full minimization of algebraic error. The algorithm 4-iterative-2, followed by bundle-adjustment is the best algorithm we know for computing the quadrifocal tensor.

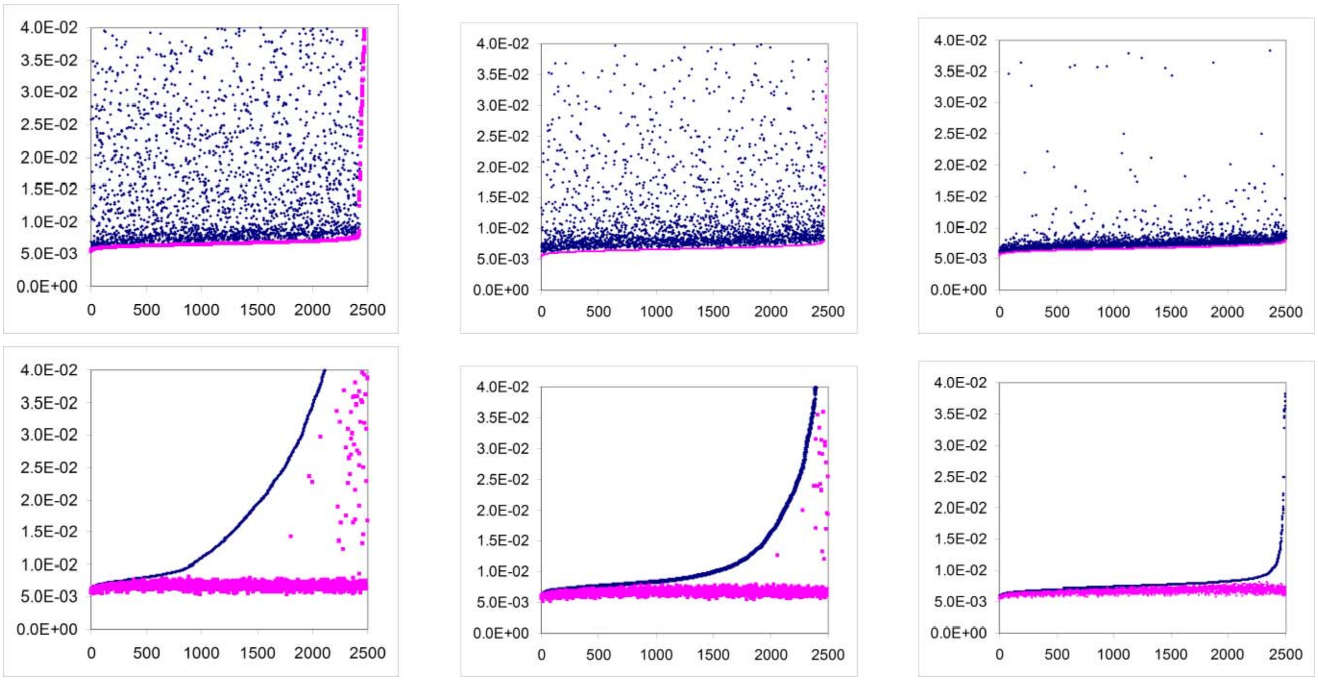

Fig. 6 (Color online) Different versions of the Grassmann algorithms. Each graph shows the reprojection error (*blue*) before and (*red*) after bundle-adjustment. Three different versions are shown: *left*, the non-iterative Grassmann algorithm, 3-Grassmann; *middle*, best of 5 permutations of 3-Grassmann (see Sect. 6.2); *right*, algorithm 3-Grassmann-algebraic, initialized by 2 permutations of 3-Grassmann. Each point on the horizontal axis represents one test out of 2500. In the *top row* of graphs, the tests are ordered according to increasing error after bundle adjustment. Thus each *blue dot* represents one test, and bundle adjustment reduces the error to the near optimal value represented by the point on the *red curve* directly below. This provides a visualization of the improvement effected by bundle adjustment. The *bottom row* of graphs shows the same results, but with the tests ordered according to increasing error before bundle adjustment. This allows a more qualitative visualization of the results of each test. Thus, from the *left-bottom graph*, we see that the 3-Grassmann algorithm results in quite large errors for all but the best 1000 out of 2500 tests. Nevertheless, in most cases, the optimum is achieved after bundle adjustment (represented by the broad *red band* below). *Isolated red dots* correspond to tests in which the optimal reprojection error was not achieved. Multiple row-permutation (*middle graphs*) is not sufficient to ensure an optimal solution in all tests, but does improve the results. However, the 3-Grassmann-algebraic algorithm (*right*) achieves close to the optimal for at least 90% of the tests, and in all cases, the optimal is achieved after bundle adjustment

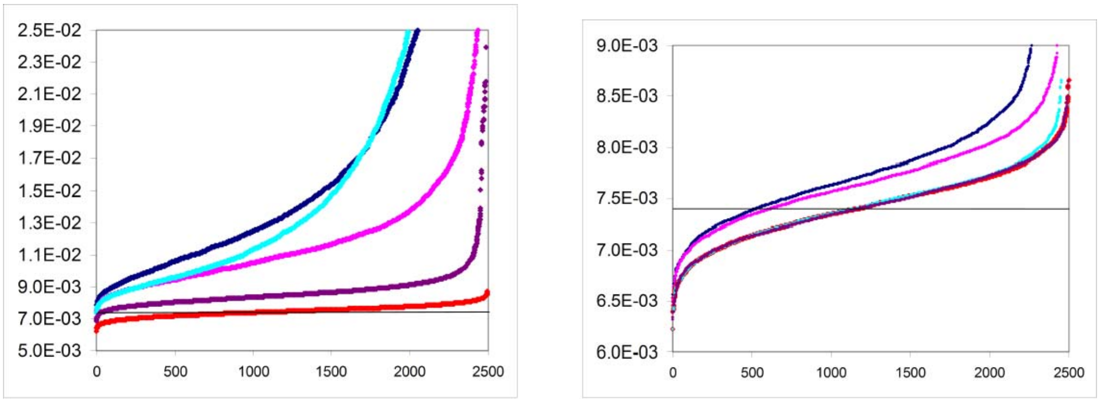

Fig. 7 (Color online) Comparison of four-view algorithms. Graphs show the reprojection error for each of 2500 separate tests (each curve sorted individually). On the *left*, results before bundle-adjustment, the algorithms are from the top: 4-linear, 4-Grassmann, 4-iterative-1, 4-Grassmann-algebraic and 4-iterative-2. After the bundle adjustment (*right*), from the top, 4-linear, 4-iterative-1 and the other three algorithms, 4-Grassmann, 4-Grassmann-algebraic and 4-iterative-2, are almost indistinguishable—all find the global optimal in almost all cases. The exception is 4-Grassmann, which fails to find the optimal in about 50 (2%) cases (see *right end* of the graph, *light blue curve*)

3. We also consider algorithms 4-Grassmann and 4-Grassmann-algebraic, which are analogous to the similarly named three-view algorithms.

The algorithm 4-iterative-2 is most closely comparable with 4-Grassmann-algebraic, since both seek to minimize the algebraic error over all well-formed quadrifocal tensors. We also compare the results of bundle-adjustment initiated by the results of these algorithms.

The results of the comparison are shown in Fig. 7.

8 Discussion of Experiments and Conclusion

This paper shows how the classical multiview tensor extends to higher dimensions, and allows reconstruction of a scene from projections in any dimension. The solution is (generically) unique except in the case of projections onto lines. The basic algorithm of this paper gives a unified method for reconstruction from projections between arbitrary dimension, and is in fact the only known algorithms to do this.

From a practical point of view, the basic algorithm, when enhanced by iterative methods, performs nearly as well as the best known 2, 3 and 4-view algorithms. In the 4-view case, only the most involved of previously known quadrifocal tensor algorithms performs better. The results of the Grassmann algorithms are sufficiently close to the optimal solution that subsequent bundle-adjustment converges to the optimal solution.

The papers of Wolf and Shashua (2002) and Hartley and Vidal (2008) describe applications of higher-dimensional projections, where the methods described in this paper may be utilized.

Acknowledgements We thank Rene Vidal for making available his implementation of the algorithm described in this paper, and for useful discussions concerning applications.

This research has been supported partly by National ICT Australia (NICTA). NICTA is funded by the Australian Government as represented by the Department of Broad-band, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

References

Faugeras, O. D., Quan, L., & Sturm, P. (2000). Self-calibration of a 1D projective camera and its application to the self-calibration of a 2D projective camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*(10), 1179–1185.

Hartley, R. I. (1997). In defense of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19*(6), 580–593.

Hartley, R. I. (1998a). Computation of the quadrifocal tensor. In *Lecture notes in computer science: Vol. 1406. Proceedings of the 5th European conference on computer vision* (pp. 20–35). Freiburg, Germany. Berlin: Springer.

Hartley, R. I. (1998b). Minimizing algebraic error. *Philosophical Transactions of the Royal Society of London, Series A*, *356*(1740), 1175–1192.

Hartley, R., & Vidal, R. (2008). Perspective nonrigid shape and motion recovery. In *Proceedings of the European conference on computer Vision* (pp. 276–289).

Hartley, R. I., & Zisserman, A. (2004). *Multiple view geometry in computer vision* (2nd ed.). Cambridge: Cambridge University Press.

Heyden, A. (1998a). A common framework for multiple-view tensors. In *Proceedings of the 5th European conference on computer vision* (pp. 3–19). Freiburg, Germany.

Heyden, A. (1998b). Reduced multilinear constraints: Theory and experiments. *International Journal of Computer Vision*, *30*(1), 5–26.

Quan, L. (2001). Two-way ambiguity in 2d projective reconstruction from three uncalibrated 1d images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *23*(2), 212–216.

Quan, L., & Kanade, T. (1997). Affine structure from line correspondences with uncalibrated affine cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19*(8), 834–845.

Semple, J. G., & Kneebone, G. T. (1979). *Algebraic projective geometry*. Oxford: Oxford University Press.

Thirthala, S., & Pollefeys, M. (2005a). Multi-view geometry of 1d radial cameras and its application to omnidirectional camera calibration. In *Proceedings of the 10th international conference on computer vision* (pp. 1539–1546). Beijing, China.

Thirthala, S., & Pollefeys, M. (2005b). The radial trifocal tensor: A tool for calibrating the radial distortion of wide-angle cameras. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (Vol. 1, pp. 321–328) San Diego.

Wolf, L., & Shashua, A. (2002). On projection matrices $\mathcal{P}^k \to \mathcal{P}^2, k = 3, \ldots, 6$, and their applications in computer vision. *International Journal of Computer Vision*, *48*(1), 53–67.