4D structure from motion: a computational algorithm

Ma Ge, Michael D'Zmura
Department of Cognitive Sciences, UC Irvine, Irvine, California 92697 USA

ABSTRACT

A problem of long standing in vision research is the recovery of three-dimensional (3D) structure from two-dimensional (2D) images. Work on structure from motion has focused on the recovery of 3D structure from multiple views of feature points like the vertices of a cube. Recent work on the perception of four-dimensional (4D) structures has prompted us to determine the circumstances under which 4D structure can be recovered from multiple views of feature points projected onto 2D images. We present a computational algorithm to solve this problem under three assumptions: 1. the correspondence of each feature point over different views is pre-determined; 2. the 4D object undergoes a rigid motion, and 3. the projection from 4D space to 2D images is a orthographic (parallel) one. Four views of five points are required. The algorithm can be generalized to treat the recovery of *n*D structure from *m*D views ($1 \leq m \leq n$). We give some results concerning the minimum number of points and views that are required to recover *n*D structure from *m*D views by this algorithm.

Keywords: structure from motion, algorithm, correspondence, rigidity, orthographic projection, dimension

1. INTRODUCTION

One remarkable ability of our visual systems is the extraction of 3D information from 2D retinal images. It is impossible to recover 3D objects completely from a single image, because all of the information along a single line of sight is projected onto a single retinal location. One way that our visual systems solve this problem is by viewing objects from different viewpoints. For example, when we study an object, we usually move the object and/or move our heads. This way to learn about 3D structure from 2D images is often referred to as 3D structure from motion. Many theoretical analyses and algorithms have been proposed to investigate under what circumstances 3D structure can be recovered successfully from 2D images (rev. Heeger, 1992).

Under what conditions can our visual systems recover high-dimensional object structure from 2D images? We are investigating currently whether humans can be trained to perceive 4D structure from interactive 2D images. A theoretical prerequisite to that investigation is ascertaining that 4D object structure can be recovered, in principle, from 2D images. In this paper, we present a computational algorithm to recover 4D structure from motion under three assumptions.

The first assumption is that the correspondence of each feature point over different images has been pre-determined. In other words, we know which point in an image corresponds to a particular point in another image. With this assumption, we can track the motion of all feature points. Establishing this correspondence is a complex visual procedure which involves the detection and matching of feature points (rev. Aggarwal and Nandhakumar, 1988). Here we assume that the correspondence is pre-determined.

The second assumption is rigidity. The structure of three- or higher-dimensional objects cannot be recovered from motion by itself. Given a finite number of 2D images, there are infinitely many structures which produce the same images. Further constraints are needed to determine structure uniquely. Rigidity is a very important constraint. Many objects in our environment have rigid shapes, and even for non-rigid objects, it is often true that parts of them are approximately rigid. Under the rigidity constraint, Ullman (1979) proved that 3D structure can be recovered uniquely (up to reflection) from three views of four non-planar points. Other researchers have investigated other constraints that our visual system might exploit to recover 3D structure. For example, Hoffman (1982) studied structure from motion under the constraint of planarity: the motion is restricted to a plane. Bennett and Hoffman (1985, 1986) also studied structure from motion under a fixed-axis constraint: all points rotate about a fixed axis. We use the rigidity constraint in the algorithm presented below.

The third assumption is that images are formed through orthographic (parallel) projection. One consequence of orthographic projection is that an object's 2D image does not depend on the distance of the object from the image plane. Another standard projection method is perspective projection, under which an object's 2D image becomes smaller when the object moves away from the image plane. Although perspective projection describes retinal images better than does

Computational Imaging, Charles A. Bouman, Robert L. Stevenson, Editors,
Proceedings of SPIE-IS&T Electronic Imaging, SPIE Vol. 5016 (2003)

orthographic projection, the latter is simpler to analyze and so provides a better starting point for our work on high-dimensional structure from motion.

Under these three assumptions, we have the following 4D structure from motion theorem:

Given four distinct 2D orthographic projections of five non-covolumetric points in a rigid configuration, the 4D structure of the points can be determined up to reflection.

In this theorem, non-covolumetric means that not all five points lie in the same 3D volume. In section 2, we give the proof of this theorem. The proof procedure itself is also an algorithm that can be implemented. In section 3, we generalize the algorithm to higher dimensions, and give some results from this algorithm.

2. ALGORITHM

In this section, first we introduce the basic idea underlying the algorithm. Then we show how to use the algorithm to recover 4D structure from motion.

2.1 Basic idea

We need five non-covolumetric 4D points to recover 4D structure from motion. With these points, one can construct four vectors which originate from the same point (see figure 1a). We denote the origin point as O, and the four vectors as $\mathbf{b}_1$, $\mathbf{b}_2$, $\mathbf{b}_3$, and $\mathbf{b}_4$. The theorem states that we require four 2D views, of which one is shown in Fig. 1b. Each 2D view is represented by two orthogonal axes $\mathbf{a}_1$ and $\mathbf{a}_2$, or $\mathbf{a}_3$ and $\mathbf{a}_4$, and so on. We set the origin of each view to the projection of the origin point O of the structure, so that what we measure in each view are the projections of vectors $\mathbf{b}_1$, $\mathbf{b}_2$, $\mathbf{b}_3$, and $\mathbf{b}_4$.

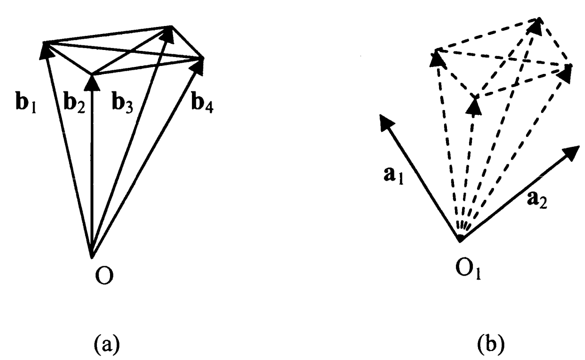

Figure 1: (a) 4D structure. (b) Projection of 4D structure onto 2D views.

The projection of the 4D structure onto the 2D views can be represented by the following matrix equation:

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ \vdots & \vdots & \vdots & \vdots \\ a_{81} & a_{82} & a_{83} & a_{84} \end{pmatrix} \begin{pmatrix} b_{11} & b_{21} & b_{31} & b_{41} \\ b_{12} & b_{22} & b_{32} & b_{42} \\ b_{13} & b_{23} & b_{33} & b_{43} \\ b_{14} & b_{24} & b_{34} & b_{44} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ \vdots & \vdots & \vdots & \vdots \\ c_{81} & c_{82} & c_{83} & c_{84} \end{pmatrix}, \text{ or} \quad (1a)$$

$$\mathbf{AB} = \mathbf{C}. \quad (1b)$$

In Eqn. 1, the row vectors of matrix **A** are the axes of the 2D views, the column vectors of matrix **B** are the vectors of the 4D structure, and the elements in matrix **C** are the projection data. The problem of 4D structure from motion can be stated formally as one of solving structure **B** and 2D views **A**, given projection data **C**.

Each element in the projection data matrix **C** provides a constraint on the values of the elements of **A** and **B**. Yet the number of unknown parameters in matrices **A** and **B** is greater than the number of known elements of **C**. Further constraints are provided by the assumptions that the axes of the 2D views are of unit length, and that the axes of each 2D view are mutually orthogonal, i.e.,

$$\mathbf{a}_i \cdot \mathbf{a}_i = 1, \quad (2)$$

$$\mathbf{a}_i \cdot \mathbf{a}_j = 0, \tag{3}$$

where $\mathbf{a}_i$ and $\mathbf{a}_j$ are the axes of the same 2D view.

Even with a sufficient number of constraints, solving Eqn. 1 is not an easy task, because the equations induced by the unit-length (Eqn. 2) and the orthogonal (Eqn. 3) conditions are not linear ones. However, one can show that, by supposing that the projection data **C** are generated from a specific view matrix of lower-triangular form, the problem can be solved in a basically linear fashion.

We show first that, given a view matrix **A**, there is a lower-triangular matrix **A'** that can be generated from **A** through a series of rotations. Suppose we have a rotation matrix $\mathbf{R}_1$, where

$$\mathbf{R}_1 = \begin{pmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{4}$$

which rotates the coordinate system in the *XY* plane. Applying $\mathbf{R}_1$ to the view matrix **A** and setting $\tan\theta_1 = a_{12}/a_{11}$ (if a_{11} is zero, one can change the order of the row vectors of the matrix **A**), one finds that

$$\mathbf{A}_1 = \mathbf{A}\mathbf{R}_1 = \begin{pmatrix} a_{11}^{(1)} & 0 & a_{13}^{(1)} & a_{14}^{(1)} \\ a_{21}^{(1)} & a_{22}^{(1)} & a_{23}^{(1)} & a_{24}^{(1)} \\ \vdots & \vdots & \vdots & \vdots \\ a_{81}^{(1)} & a_{82}^{(1)} & a_{83}^{(1)} & a_{84}^{(1)} \end{pmatrix}. \tag{5}$$

A second rotation matrix $\mathbf{R}_2$, where

$$\mathbf{R}_2 = \begin{pmatrix} \cos\theta_2 & 0 & -\sin\theta_2 & 0 \\ 0 & 1 & 0 & 0 \\ \sin\theta_2 & 0 & \cos\theta_2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{6}$$

rotates the coordinate system in the *XZ* plane. Applying $\mathbf{R}_2$ to matrix $\mathbf{A}_1$ and setting $\tan\theta_2 = a^{(1)}_{13}/a^{(1)}_{11}$, one has

$$\mathbf{A}_2 = \mathbf{A}_1\mathbf{R}_2 = \begin{pmatrix} a_{11}^{(2)} & 0 & 0 & a_{14}^{(2)} \\ a_{21}^{(2)} & a_{22}^{(2)} & a_{23}^{(2)} & a_{24}^{(2)} \\ \vdots & \vdots & \vdots & \vdots \\ a_{81}^{(2)} & a_{82}^{(2)} & a_{83}^{(2)} & a_{84}^{(2)} \end{pmatrix}. \tag{7}$$

By applying a series of similar rotations, one can transform the view matrix **A** into the following lower-triangular form:

$$\mathbf{A}' = \mathbf{A}\mathbf{R}_1\mathbf{R}_2 \cdots = \begin{pmatrix} a'_{11} & 0 & 0 & 0 \\ a'_{21} & a'_{22} & 0 & 0 \\ a'_{31} & a'_{32} & a'_{33} & 0 \\ a'_{41} & a'_{42} & a'_{43} & a'_{44} \\ a'_{51} & a'_{52} & a'_{53} & a'_{54} \\ \vdots & \vdots & \vdots & \vdots \\ a'_{81} & a'_{82} & a'_{83} & a'_{84} \end{pmatrix}. \tag{8}$$

This lower-triangular form is essential to the algorithm, because it lets one solve the problem in a linear way, and reduces the number of unknown parameters. Let **R** denote this series of rotations, i.e., $\mathbf{R} = \mathbf{R}_1\mathbf{R}_2\ldots$ Applying **R** to Eqn. 1, we have

$$(\mathbf{A}\mathbf{R})(\mathbf{R}^{-1}\mathbf{B}) = \mathbf{C}, \text{ or} \tag{9a}$$

$$\mathbf{A}'\mathbf{B}' = \mathbf{C}, \text{ or} \tag{9b}$$

$$\begin{pmatrix} a'_{11} & 0 & 0 & 0 \\ a'_{21} & a'_{22} & 0 & 0 \\ a'_{31} & a'_{32} & a'_{33} & 0 \\ a'_{41} & a'_{42} & a'_{43} & a'_{44} \\ a'_{51} & a'_{52} & a'_{53} & a'_{54} \\ \vdots & \vdots & \vdots & \vdots \\ a'_{81} & a'_{82} & a'_{83} & a'_{84} \end{pmatrix} \begin{pmatrix} b'_{11} & b'_{21} & b'_{31} & b'_{41} \\ b'_{12} & b'_{22} & b'_{32} & b'_{42} \\ b'_{13} & b'_{23} & b'_{33} & b'_{43} \\ b'_{14} & b'_{24} & b'_{34} & b'_{44} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ \vdots & \vdots & \vdots & \vdots \\ c_{81} & c_{82} & c_{83} & c_{84} \end{pmatrix}, \tag{9c}$$

where $\mathbf{A}' = \mathbf{AR}$, and $\mathbf{B}' = \mathbf{R}^{-1}\mathbf{B}$.

We call **A'** the canonical views and **B'** the rotated structure. When using the projection data **C** and the canonical view matrix **A'** to recover 4D structure, the algorithm produces as its solution the structure **B'**, which is a rotated version of the original structure matrix **B** (note that $\mathbf{R}^{-1}$, which is the inverse matrix of the rotation matrix **R**, is also a rotation matrix). The column vectors of the rotated structure **B'** need not be the same as those of the original structure **B**, but the structural information is the same, because **B'** is a rotated version of **B**.

The basic idea of the algorithm is to suppose that the projection data **C** are generated from the canonical views **A'**, and to solve for the canonical views **A'** and the rotated structure **B'** from the projection data **C**. In the next subsection, we introduce the algorithm for solving the rotated structure **B'** from the projection data **C**. The symbols used in the algorithm are listed in the following table.

A	The original view matrix
A'	The canonical view matrix
$\mathbf{A}'_{4\times4}$	The first four rows of the canonical view matrix **A'**
B	The original structure matrix
B'	The rotated structure matrix
C	The projection data matrix
$\mathbf{C}_{4\times4}$	The first four rows of the projection data matrix **C**
R	The rotation matrix
$\mathbf{a}'_i$	A row vector of the canonical view matrix **A'**
a'_{ij}	An element of the canonical view matrix **A'**
$\mathbf{b}'_i$	A column vector of the rotated structure matrix **B'**
b'_{ij}	An element of the rotated structure matrix **B'**
$\mathbf{c}_i$	A row vector of the projection data matrix **C**
c_{ij}	An element of the projection data matrix **C**
k_{ij}	The linear coefficient between vectors $\mathbf{a}'_i$ or $\mathbf{c}_i$
K	The coefficient matrix
$\mathbf{\Lambda}$	The vector of compound unknown parameters
$\mathbf{\Delta}$	An intermediate data vector

Table 1: The symbols used in the algorithm.

2.2 4D structure from motion

We begin the calculation from the known projection data **C**. If the first four rows $\mathbf{a}'_1$, $\mathbf{a}'_2$, $\mathbf{a}'_3$, and $\mathbf{a}'_4$ of the canonical view matrix **A'** are independent, and the four columns $\mathbf{b}'_1$, $\mathbf{b}'_2$, $\mathbf{b}'_3$, and $\mathbf{b}'_4$ of the rotated structure **B'** are independent, then the first four rows $\mathbf{c}_1$, $\mathbf{c}_2$, $\mathbf{c}_3$, and $\mathbf{c}_4$ of the projection data matrix **C** are also independent. The remaining row vectors $\mathbf{c}_5$, $\mathbf{c}_6$, $\mathbf{c}_7$ and $\mathbf{c}_8$ of the projection data matrix **C** can be represented as linear combinations of the first four vectors:

$$\begin{cases} \mathbf{c}_5 = k_{51}\mathbf{c}_1 + k_{52}\mathbf{c}_2 + k_{53}\mathbf{c}_3 + k_{54}\mathbf{c}_4 \\ \quad\vdots \\ \mathbf{c}_8 = k_{81}\mathbf{c}_1 + k_{82}\mathbf{c}_2 + k_{83}\mathbf{c}_3 + k_{84}\mathbf{c}_4 \end{cases}, \tag{10}$$

where k_{ij} are coefficients to be determined. The first equation in the above set (Eqn. 10) can also be written as

$$\begin{pmatrix} c_{51} \\ c_{52} \\ c_{53} \\ c_{54} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{21} & c_{31} & c_{41} \\ c_{12} & c_{22} & c_{32} & c_{42} \\ c_{13} & c_{23} & c_{33} & c_{43} \\ c_{14} & c_{24} & c_{34} & c_{44} \end{pmatrix} \begin{pmatrix} k_{51} \\ k_{52} \\ k_{53} \\ k_{54} \end{pmatrix}. \tag{11}$$

Because $\mathbf{c}_1$, $\mathbf{c}_2$, $\mathbf{c}_3$, and $\mathbf{c}_4$ are independent, the matrix in Eqn. 11 is full rank, and one can solve for the coefficients k_{51}, k_{52}, k_{53}, and k_{54}. One can determine the other coefficients in Eqn. 10 in like fashion.

The relationship between views $\mathbf{A}'$ and data $\mathbf{C}$ is (from Eqn. 9b)

$$\mathbf{A}' = \mathbf{C}\,(\mathbf{B}')^{-1}, \tag{12a}$$

which, when written out, provides eight equations:

$$\begin{cases} \mathbf{a}'_1 = \mathbf{c}_1 (\mathbf{B}')^{-1} \\ \quad\vdots \\ \mathbf{a}'_8 = \mathbf{c}_8 (\mathbf{B}')^{-1} \end{cases}. \tag{12b}$$

Solving for the row vector $\mathbf{a}'_5$, one finds:

$$\begin{aligned} \mathbf{a}'_5 &= \mathbf{c}_5 (\mathbf{B}')^{-1} \\ &= (k_{51}\mathbf{c}_1 + k_{52}\mathbf{c}_2 + k_{53}\mathbf{c}_3 + k_{54}\mathbf{c}_4)(\mathbf{B}')^{-1} \\ &= k_{51}\mathbf{c}_1(\mathbf{B}')^{-1} + k_{52}\mathbf{c}_2(\mathbf{B}')^{-1} + k_{53}\mathbf{c}_3(\mathbf{B}')^{-1} + k_{54}\mathbf{c}_4(\mathbf{B}')^{-1} \\ &= k_{51}\mathbf{a}'_1 + k_{52}\mathbf{a}'_2 + k_{53}\mathbf{a}'_3 + k_{54}\mathbf{a}'_4 \end{aligned}. \tag{13}$$

Similar expressions obtain for the remaining row vectors of the canonical view matrix:

$$\begin{cases} \mathbf{a}'_6 = k_{61}\mathbf{a}'_1 + k_{62}\mathbf{a}'_2 + k_{63}\mathbf{a}'_3 + k_{64}\mathbf{a}'_4 \\ \mathbf{a}'_7 = k_{71}\mathbf{a}'_1 + k_{72}\mathbf{a}'_2 + k_{73}\mathbf{a}'_3 + k_{74}\mathbf{a}'_4 \\ \mathbf{a}'_8 = k_{81}\mathbf{a}'_1 + k_{82}\mathbf{a}'_2 + k_{83}\mathbf{a}'_3 + k_{84}\mathbf{a}'_4 \end{cases}. \tag{14}$$

Eqns. 13 and 14 show that the relationships among the row vectors of the canonical views $\mathbf{A}'$ are the same as those among the row vectors of the projection data matrix $\mathbf{C}$. Because $\mathbf{a}'_5$, $\mathbf{a}'_6$, $\mathbf{a}'_7$, and $\mathbf{a}'_8$ can be represented as linear combinations of $\mathbf{a}'_1$, $\mathbf{a}'_2$, $\mathbf{a}'_3$, and $\mathbf{a}'_4$, the unknown parameters within the canonical view matrix $\mathbf{A}'$ are the elements in row vectors $\mathbf{a}'_1$, $\mathbf{a}'_2$, $\mathbf{a}'_3$, and $\mathbf{a}'_4$. Bearing in mind the lower-triangular form of $\mathbf{A}'$, the total number of unknown parameters in the canonical view matrix is ten.

Additional conditions are needed to determine the unknown elements of the canonical view matrix $\mathbf{A}'$. These conditions are provided by the equations expressing the unit length of the view vectors:

$$\begin{cases} \mathbf{a}'_1 \cdot \mathbf{a}'_1 = 1 \\ \quad\vdots \\ \mathbf{a}'_8 \cdot \mathbf{a}'_8 = 1 \end{cases}, \tag{15}$$

and their mutual orthogonality:

$$\begin{cases} \mathbf{a}'_1 \cdot \mathbf{a}'_2 = 0 \\ \quad\vdots \\ \mathbf{a}'_7 \cdot \mathbf{a}'_8 = 0 \end{cases}. \tag{16}$$

The first four equations in Eqn. 15, when written out, are

$$\begin{cases} a'^{\,2}_{11} = 1 \\ a'^{\,2}_{21} + a'^{\,2}_{22} = 1 \\ a'^{\,2}_{31} + a'^{\,2}_{32} + a'^{\,2}_{33} = 1 \\ a'^{\,2}_{41} + a'^{\,2}_{42} + a'^{\,2}_{43} + a'^{\,2}_{44} = 1 \end{cases}. \tag{17}$$

The first two equations in Eqn. 16, when written out, are

$$\begin{cases} a'_{11}\, a'_{21} = 0 \\ a'_{31}\, a'_{41} + a'_{32}\, a'_{42} + a'_{33}\, a'_{43} = 0 \end{cases} \quad . \tag{18}$$

The remaining six equations in terms of $\mathbf{a}'_5$, $\mathbf{a}'_6$, $\mathbf{a}'_7$, and $\mathbf{a}'_8$ in Eqns. 15 and 16 may be expressed as follows:

$$\mathbf{K\Lambda} = \mathbf{\Delta}, \tag{19}$$

where

$$\mathbf{K} = \begin{pmatrix} 2k_{51}k_{53} & 2k_{51}k_{54} & 2k_{52}k_{53} & 2k_{52}k_{54} \\ 2k_{61}k_{63} & 2k_{61}k_{64} & 2k_{62}k_{63} & 2k_{62}k_{64} \\ 2k_{71}k_{73} & 2k_{71}k_{74} & 2k_{72}k_{73} & 2k_{72}k_{74} \\ 2k_{81}k_{83} & 2k_{81}k_{84} & 2k_{82}k_{83} & 2k_{82}k_{84} \\ k_{51}k_{63} + k_{53}k_{61} & k_{51}k_{64} + k_{54}k_{61} & k_{52}k_{63} + k_{53}k_{62} & k_{52}k_{64} + k_{54}k_{62} \\ k_{71}k_{83} + k_{73}k_{81} & k_{71}k_{84} + k_{74}k_{81} & k_{72}k_{83} + k_{73}k_{82} & k_{72}k_{84} + k_{74}k_{82} \end{pmatrix}, \tag{20}$$

$$\mathbf{\Lambda} = \begin{pmatrix} a'_{11}\, a'_{31} \\ a'_{11}\, a'_{41} \\ a'_{21}\, a'_{31} + a'_{22}\, a'_{32} \\ a'_{21}\, a'_{41} + a'_{22}\, a'_{42} \end{pmatrix}, \text{ and} \tag{21}$$

$$\mathbf{\Delta} = \begin{pmatrix} 1 - k_{51}^2 - k_{52}^2 - k_{53}^2 - k_{54}^2 \\ 1 - k_{61}^2 - k_{62}^2 - k_{63}^2 - k_{64}^2 \\ 1 - k_{71}^2 - k_{72}^2 - k_{73}^2 - k_{74}^2 \\ 1 - k_{81}^2 - k_{82}^2 - k_{83}^2 - k_{84}^2 \\ -k_{51}k_{61} - k_{52}k_{62} - k_{53}k_{63} - k_{54}k_{64} \\ -k_{71}k_{81} - k_{72}k_{82} - k_{73}k_{83} - k_{74}k_{84} \end{pmatrix}. \tag{22}$$

The matrix equation in Eqn. 19 is a set of six linear equations in terms of four compound unknown parameters $a'_{11}\, a'_{31}$, $a'_{11}\, a'_{41}$, $a'_{21}\, a'_{31} + a'_{22}\, a'_{32}$, and $a'_{21}\, a'_{41} + a'_{22}\, a'_{42}$. Two of these equations are redundant. If the coefficient matrix **K** has full rank (we discuss this problem below), one can determine the four compound parameters. From these four compound parameters, combined with Eqns. 17 and 18, one can find all ten original parameters, namely, the elements in row vectors $\mathbf{a}'_1$, $\mathbf{a}'_2$, $\mathbf{a}'_3$, and $\mathbf{a}'_4$. These ten equations are listed together in Eqn. 23:

$$\begin{cases} a'^{2}_{11} = 1 \\ a'_{11}\, a'_{21} = 0 \\ a'_{11}\, a'_{31} = d_1 \\ a'_{11}\, a'_{41} = d_2 \\ a'^{2}_{21} + a'^{2}_{22} = 1 \\ a'_{21}\, a'_{31} + a'_{22}\, a'_{32} = d_3 \\ a'_{21}\, a'_{41} + a'_{22}\, a'_{42} = d_4 \\ a'^{2}_{31} + a'^{2}_{32} + a'^{2}_{33} = 1 \\ a'_{31}\, a'_{41} + a'_{32}\, a'_{42} + a'_{33}\, a'_{43} = 0 \\ a'^{2}_{41} + a'^{2}_{42} + a'^{2}_{43} + a'^{2}_{44} = 1 \end{cases} , \tag{23}$$

where d_1, d_2, d_3, and d_4 are the elements of the vector $\mathbf{\Lambda}$ solved from Eqn. 19.

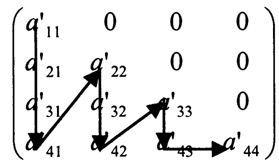

Figure 2. The order of solving for the original parameters

Solving for the view parameters in the order indicated in Fig. 2, we find 16 solutions for the canonical view matrix **A'**. We show below (Section 3.3) that there are, in fact, only two structures related by a reflection which are associated with these 16 solutions.   One such solution is shown in Eqn. 24:

$$\begin{cases} a'_{11}=1 \\ a'_{21}=0 \\ a'_{31}=d_1 \\ a'_{41}=d_2 \\ a'_{22}=1 \\ a'_{32}=d_3 \\ a'_{42}=d_4 \\ a'_{33}=\sqrt{1-d_1^{\,2}-d_3^{\,2}} \\ a'_{43}=\dfrac{-d_1d_2-d_3d_4}{\sqrt{1-d_1^{\,2}-d_3^{\,2}}} \\ a'_{44}=\sqrt{1-d_2^{\,2}-d_4^{\,2}-\dfrac{(d_1d_2+d_3d_4)^2}{1-d_1^{\,2}-d_3^{\,2}}} \end{cases} . \tag{24}$$

With the canonical views **A'** in hand, one may proceed to compute the rotated structure **B'**. Using the first four rows $\mathbf{A'}_{4\times4}$ of the canonical view matrix **A'** and the first four rows $\mathbf{C}_{4\times4}$ of the projection data matrix **C**, one finds:

$$\mathbf{A'}_{4\times4}=\begin{pmatrix} a'_{11} & 0 & 0 & 0 \\ a'_{21} & a'_{22} & 0 & 0 \\ a'_{31} & a'_{32} & a'_{33} & 0 \\ a'_{41} & a'_{42} & a'_{43} & a'_{44} \end{pmatrix}, \tag{25}$$

$$\mathbf{B'}=\begin{pmatrix} b'_{11} & b'_{21} & b'_{31} & b'_{41} \\ b'_{12} & b'_{22} & b'_{32} & b'_{42} \\ b'_{13} & b'_{23} & b'_{33} & b'_{43} \\ b'_{14} & b'_{24} & b'_{34} & b'_{44} \end{pmatrix}, \qquad \text{and} \tag{26}$$

and

$$\mathbf{C}_{4\times4}=\begin{pmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \\ c_{41} & c_{42} & c_{43} & c_{44} \end{pmatrix}. \tag{27}$$

One can thus solve for the rotated structure **B'** using the known projection data $\mathbf{C}_{4\times4}$ and the computed canonical view matrix $\mathbf{A'}_{4\times4}$ as follows:

$$\mathbf{B'} = (\mathbf{A'}_{4\times4})^{-1}\mathbf{C}_{4\times4}. \tag{28}$$

Note, again, that the column vectors of the rotated structure **B'** have the same structural information as the original structure **B**.

To summarize, one can determine 4D structure from 2D views using the following steps:
(1). Solve for the coefficients k_{ij} using the projection data **C** (Eqn. 11);
(2). Construct Eqn. 19 from coefficients k_{ij};
(3). Solve the compound parameters from Eqn. 19;
(4). Solve the canonical view matrix $\mathbf{A}'_{4\times4}$ from Eqn. 23, and
(5). Calculate the rotated structure **B'** from the canonical views $\mathbf{A}'_{4\times4}$ and projection data $\mathbf{C}_{4\times4}$ (Eqn. 28).

3. GENERALIZATION

The algorithm for recovering 4D structure from 2D views introduced in the previous section may be generalized to treat problems of recovering nD structure from motion in which the dimension of the views need not be two, rather any dimension in the range one through n. However, before we generalize the algorithm to higher dimensions, some questions must be considered. First, how many nD points are necessary? Second, how many mD views ($1 \le m \le n$) are needed? Third, do solutions exist and are they unique? We give some results concerning these questions in the following section.

3.1 How many points are necessary?

To solve an nD structure from motion problem, one needs at least n independent column vectors within the structure matrix **B**. Otherwise, the data matrix (Eqn. 11) would not be of full rank, and the coefficients k_{ij} could not be computed. These n column vectors are sufficient: any additional column vectors could be represented as linear combinations of these n vectors, and so provide no extra information to the algorithm. To construct n vectors, one needs at least n+1 points (*e.g.*, Fig. 1a). In a word, to solve for nD structure, one needs a minimum number of points n_p given by:

$$n_p = n+1 . \tag{29}$$

This number is reasonable, because the smallest number of points that can span nD space is n+1. This number of points also can be used to describe the simplest structures in nD space. For example, in the 2D plane, the simplest polygon is a triangle with three vertices; in 3D space, the simplest polyhedron is a tetrahedron with four vertices, while in 4D space, the simplest polytope is a hypertetrahedron with five vertices, and so on.

3.2 How many views are needed?

In the nD structure from motion problem, the canonical view matrix **A'** has the following form:

$$\mathbf{A}' = \begin{pmatrix} a'_{11} & 0 & \cdots & 0 \\ a'_{21} & a'_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a'_{n1} & a'_{n2} & \cdots & a'_{nn} \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix}, \tag{30}$$

which has $(n+1)\,n/2$ unknown parameters. We thus need at least $(n+1)\,n/2$ independent equations to solve for the nD structure.

For the nD structure from 1D views problem, each 1D view gives a single equation. So the minimum number of views n_v is given by:

$$n_v = \binom{n+1}{2} = \frac{(n+1)n}{2} . \tag{31}$$

For the nD structure from mD views problem, each mD view provides m unit-length equations. In addition, each mD view provides $m(m\text{-}1)/2$ orthogonal equations. In total, each mD view provides $(m+1)\,m/2$ equations. Therefore, the minimum number of views n_v must comply with the following condition:

$$n_v \binom{m+1}{2} \ge \binom{n+1}{2} . \tag{32}$$

The minimum number of views given by Eqns. 31 and 32 need not guarantee a full rank for the matrix **K** in Eqn. 20. It may be the case that the minimum number of views provide row vectors in **K** that are not independent. There are two sources for such a lack of independence.

First, if we use a vector $\mathbf{a}'_i$ as an axis of a view, any other vectors of form $\alpha\mathbf{a}'_i$ will produce a dependent row in the matrix **K**. This example is trivial, but shows that one should use views which produce independent rows in the matrix **K**. Because generating the matrix **K** from the elements k_{ij} is a non-linear transformation (Eqn. 20), some views may produce dependent rows in **K**, even when these views themselves are independent.

The second source of dependence is the fact that some of the orthogonal equations may be deduced from others. For example, in the 3D structure from 2D views problem, we need two 2D views according to Eqn. 31. These two 2D views consist of four row vectors, $\mathbf{a}'_1$, $\mathbf{a}'_2$, $\mathbf{a}'_3$, and $\mathbf{a}'_4$ in the canonical view matrix **A'**, where $\mathbf{a}'_1 \perp \mathbf{a}'_2$, $\mathbf{a}'_3 \perp \mathbf{a}'_4$, and $\mathbf{a}'_4 = k_{41}\mathbf{a}'_1 + k_{42}\mathbf{a}'_2 + k_{43}\mathbf{a}'_3$. Then the matrix **K** has two rows that are deduced from $\mathbf{a}'_4 \cdot \mathbf{a}'_4 = 1$ and $\mathbf{a}'_3 \cdot \mathbf{a}'_4 = 0$:

$$\mathbf{K} = \begin{pmatrix} 2k_{41}k_{43} & 2k_{42}k_{43} \\ k_{41} & k_{42} \end{pmatrix}. \qquad (33)$$

This matrix **K** is not full rank. Unlike the first source of dependence, this second source applies to all views, and so cannot be avoided.

More views are needed in both varieties of dependence. It is difficult to find a uniform expression to check the dependence of the matrix **K**. We therefore suggest a practical way to check the rank of matrix **K** step by step. Use the number n_v in Eqn. 31 or 32 as an initial value and then check the rank of the matrix **K**. If the matrix is not full rank, add one more view and check the rank again, and so on until one has a full rank matrix.

The following table gives some results concerning the minimum number of views for determining structure in two through five dimensions. All of the cases have been tested with simulated data using *MatLab*.

Dimension of structure	Dimension of views	Number of views
2	1	3
3	1	6
	2	3
4	1	10
	2	4
	3	3
5	1	15
	2	5
	3	3
	4	3

Table 2: The minimum number of views for the recovery of *n*D structure.

3.3 Existence and uniqueness of solution

The existence of solutions is obvious: because the projection data are generated by projecting a structure onto sub-dimensional views, there exists at least one solution which is the structure itself.

Uniqueness is more complex. When one computes the canonical view matrix $\mathbf{A}'_{4\times4}$ from compound parameters, all of the unknown parameters of the matrix $\mathbf{A}'_{4\times4}$ are solved in a linear way except those along the diagonal. The diagonal elements are solved for in the following way:

$$a'^{\,2}_{nn} = c\,, \qquad (34)$$

where c is a positive constant. For each a'_{nn}, there are two values of opposite sign. Correspondingly, the two solutions are reflections of one other along the *n*th dimension. This can be seen more clearly by transforming the matrix $\mathbf{A}'_{4\times4}$ into a diagonal one:

$$\begin{pmatrix} a'_{11} & 0 & 0 & 0 \\ 0 & a'_{22} & 0 & 0 \\ 0 & 0 & a'_{33} & 0 \\ 0 & 0 & 0 & a'_{44} \end{pmatrix} \begin{pmatrix} b'_{11} & b'_{21} & b'_{31} & b'_{41} \\ b'_{12} & b'_{22} & b'_{32} & b'_{42} \\ b'_{13} & b'_{23} & b'_{33} & b'_{43} \\ b'_{14} & b'_{24} & b'_{34} & b'_{44} \end{pmatrix} = \begin{pmatrix} c'_{11} & c'_{12} & c'_{13} & c'_{14} \\ c'_{21} & c'_{22} & c'_{23} & c'_{24} \\ c'_{31} & c'_{32} & c'_{33} & c'_{34} \\ c'_{41} & c'_{42} & c'_{43} & c'_{44} \end{pmatrix}. \qquad (35)$$

For example, if one takes the opposite sign for a'_{22}, then the second coordinate of structure **B'** will also have the opposite sign. For the *n*D structure from motion problem, there are 2^n solutions if we take different signs for a'_{nn}. Each solution can be represented by a transformation of the original structure, i.e.,

$$\mathbf{B}' = \mathbf{TB}, \tag{36}$$

where **T** is the transformation matrix. The determinant of matrix **T** divides the solutions into two classes. Solutions of the first class have the determinant $|\mathbf{T}|=1$. Solutions of the second class have the determinant $|\mathbf{T}|=-1$. Solutions of the first class are rotations of the original structure. Solutions of the second class are rotations of the reflection of the original structure. In conclusion, the solution is unique up to a reflection.

4. DISCUSSION

In this paper, we present an algorithm for reconstructing 4D structure from motion and find that 4D structure can be determined uniquely up to a reflection by four or more 2D views of five or more 4D points. Generalizing the algorithm to arbitrary dimension lets us determine the minimum number of points and views needed to recover *nD* structure from *m*D views (Table 2).

We do not suppose that the analysis describes human visual function. In the case of recovering 3D structure from 2D views, the algorithm presented here suggests that we need at least three views, a result consistent with that of Ullman (1979). With fewer views, the algorithm will fail. Bennett and Hoffman *et al.* (1989) have proved that, with two orthographic views, there are a family of 3D structures which are compatible with the 2D projection data. Yet psychophysical experiments (Braunstein *et al.* 1987) reveal that one can perceive a single 3D structure when presented fewer views or fewer points. Note also that it is not the case that the rigid body motion in 3D leads necessarily to the perception of rigid body motion; Weiss and Adelson (2000) have shown that the structure of certain 3D rigid objects appears highly nonrigid under rotation.

Standard methods for projecting 4D points onto 2D image planes are detailed by Hollasch (1991). These involve choosing either orthographic or perspective projection from 4D to 3D, followed by a second projection from 3D to 2D which is, again, either orthographic or perspective. In addition, one can take a 3D cross-section of 4D space and project that cross-section onto the 2D image plane, as in our previous work (D'Zmura *et al.*, 2000, 2001; Seyranian, 2001). Because a finite number of 3D cross-sections cannot sample adequately a discrete number of points in 4D space, one cannot pursue with the present methods the theory of 4D structure from motion using 3D cross-sections. We have implemented the 4D-to-3D orthographic projection followed by a 3D-to-2D perspective projection in current psychophysical work on 4D object recognition, and we plan to extend the present results on 4D structure from motion to this case.

ACKNOWLEDGEMENTS

We would like to thank Geoffrey Iverson for his help. This work was supported by NEI EY10014.

REFERENCES

1. J.K. Aggarwal, N. Nandhakumar, "On the computation of motion from sequences of images – a review", *Proc. IEEE*, 76:917-935, 1988.
2. B.M. Bennett, D.D. Hoffman, "The computation of structure from fixed-axis motion: Nonrigid structures", *Biological Cybernetics*, **51**(293-300), 1985.
3. M.L. Braunstein, D.D. Hoffman, L.R. Shapiro, G.J. Anderson, B.M. Bennett, "Minimum points and views for the recovery of three-dimensional structure", *Journal of Experimental Psychology: Human Perception and Performance*, **13**(335-343), 1987.
4. M. D'Zmura, P. Colantoni, G.D. Seyranian, "Virtual environments with four or more spatial dimensions", *Presence* **9**(616-631), 2001.
5. M. D'Zmura, P. Colantoni, G.D. Seyranian, "Visualization of events from arbitrary spacetime perspectives", In R.F. Erbacher, P.C. Chen, J.C. Roberts, C.M. Wittenbrink (Eds.), *Visual Data Exploration and Analysis VII*, **3860**(35-40), 2000.
6. D.J. Heeger, A.D. Jepson, "Subspace methods for recovering rigid motion I: algorithm and implementation", *International Journal of Computer Vision*, **7**, **2**(95-117), 1992.

7. D.D. Hoffman, B.M. Bennett, "The computation of structure from fixed-axis motion: Rigid structures", *Biological Cybernetics*, **54**(71-83), 1986.
8. D.D. Hoffman, B.E. Flinchbaugh, "The interpretation of biological motion", *Biological Cybernetics*, **42, 3**(197-204), 1982.
9. S. Hollasch, *Four-space visualization of 4D objects*, Unpublished M.S. Dissertation, Arizona State University, 1991.
10. G.D. Seyranian, *Human spatial perception in environments with four spatial dimensions*, Unpublished Ph.D. dissertation, University of California, Irvine, 2001.
11. S. Ullman, *The Interpretation of Motion*, MIT Press: Cambridge, MA., 1979.
12. Y. Weiss, E.H. Adelson, "Adventures with gelatinous ellipses: constraints on models of human motion analysis", *Perception* **29**(543-566), 2000.