Shape Tensors for Efficient and Learnable Indexing

Daphna Weinshall, Michael Werman and Amnon Shashua*
Institute of Computer Science, The Hebrew University of Jerusalem
91904, Jerusalem, Israel
tl: 972-2-658-6298
email: daphna@cs.huji.ac.il

Abstract

Multi-point geometry: *The geometry of 1 point in N images under perspective projection has been thoroughly investigated, identifying bilinear, trilinear, and quadrilinear relations between the projections of 1 point in 2, 3 and 4 frames respectively. The* **dual** *problem - the geometry of N points in 1 image - has been studied mostly in the context of object recognition, often assuming weak perspective or affine projection. We provide here a complete description of this problem. We employ a formalism in which multi-frame and multi-point geometries appear in symmetry: points and projections are interchangeable. We then derive bilinear equations for 6 points (dual to 4-frame geometry), trilinear equations for 7 points (dual to 3-frame geometry), and quadrilinear equations for 8 points (dual to the epipolar geometry). We show that the quadrilinear equations are dependent on the the bilinear and trilinear equations, and we show that adding more points will not generate any new equation.*

Applications to reconstruction and recognition: *The new equations are used to design new algorithms for the reconstruction of shape from many frames, and for learning invariant relations for indexing into a data-base. We describe algorithms which require matching 6 (or more) corresponding points from at least 4 images, 7 (or more) points from at least 3 images, or 8 (or more) points from at least 2 images. Unlike previous approaches, the equations developed here lead to* direct *and* linear *solutions without going through the cameras' geometry. Our final linear shape computation uses all the available data – all points and all frames simultaneously: it uses a factorization of the matrix of invariant relations into 2 components of rank 4, a shape matrix and a coordinate-system matrix.*

1 Introduction

The geometry of multiple primitives in multiple frames, where a $3D$ model consisting of many primitives is projected to a sequences of images via unknown cameras, has two inherent unknowns: the camera geometry, and the shape geometry. That is, we do not know in advance the parameters of the projection from $3D$ to $2D$ in each image, and we do not know the parameters of the $3D$ shape of each point in the model. Depending on the application at hand, we may want to compute shape, or camera geometry, or both. For example, tracking requires the knowledge of camera geometry, whereas object recognition requires the knowledge of shape.

Very often, the natural sequence of computations had been argued to be the following: First, compute the camera geometry (by camera geometry we refer here to both explicit representations, e.g., rotation and translation matrices, or implicit representations, e.g., the fundamental matrix). Second, compute the shape using the known projection geometry. This order of events makes particular sense when the first task that one needs to solve is the tracking of features across frames.

But the reverse order of computations is also sensible: first compute shape, then compute camera geometry if necessary. This order makes sense mostly when only shape is needed, for example, in a pure recognition task. If this is the case, it makes sense to compute the shape first, rather than rely in the shape computation on a pre-processing step (the computation of camera geometry), which may not always be reliable.

If one needs to know both camera geometry and shape, for example - if one wants to accomplish both feature tracking and recognition, then either order makes sense. Thus a complete system may approach the problem in both directions simultaneously.

Most of the literature on the subject took the first approach, namely, the computation of camera geometry is done first. In particular, the relations between the projections of a single point in 2, 3, and 4 images are now completely known under orthographic, weak perspective, and perspective projection models (there are no relations involving 5 or more images that cannot be factored into relations involving fewer images). For the case of uncalibrated perspective cameras, the following has been shown:

2 images: This case is described by the epipolar geometry: the fundamental matrix represents the camera geometry, and the image measurements appear in the equation in a bilinear form [5, 6].

3 images: A few $3\times3\times3$ tensors represent the camera geometry, and the image measurements appear in the equations in trilinear forms [8, 9, 11, 4].

*This research was sponsored by ARPA through the U.S. Office of Naval Research under Grant N00014-93-1-1202, R&T Project Code 4424341—01.

4 images: A few $3 \times 3 \times 3 \times 3$ tensors represent the camera geometry, and the image measurements appear in the equations in quadrilinear forms [13, 17, 3].

[3] provides a simple derivation of the same results, using crisp geometrical intuition and no more than linear algebra.

Much less is known about multi-point geometry under perspective projection with uncalibrated cameras, and this gap is filled by our paper. We present here a complete analysis of the multi-point geometry, describing relations between the projections of many points in an image (each relation has a dual relation in multi-camera geometry). This analysis provides the foundation for a computation where shape is computed first, and camera geometry second. More specifically, in Section 2 we show dual results to the ones listed above (for 4 basis points + n other points):

(4+)2 points: A 3×3 matrix (dual to the fundamental matrix) represents the $3D$ shape, and the image measurements appear in the equation in a bilinear form.

(4+)3 points: A few $3 \times 3 \times 3$ tensors represent the shape, and the image measurements appear in the equations in trilinear forms.

(4+)4 points: A few $3{\times}3{\times}3{\times}3$ tensors represent the $3D$ shape, and the image measurements appear in the equations in quadrilinear forms.

We prove these results by modifying the analysis described in [3] to apply to the dual multi-point case. This analysis gives the geometrical insight into what is going on. We first observe that a projection matrix from a $3D$ projective world to a $2D$ projective image is really a point in $\mathcal{P}^3$. We then observe that the relations between models, projections, and images can be written in a symmetrical form where models and projections are interchangeable. Using these observations, for every known relation between images and projection matrices we can derive a dual relation between images and models. In particular, we easily get all the results stated above.

In addition to the pleasing elegance of this duality, the implication is simple and powerful: we get new algorithms to compute shape directly, without first computing camera geometry. In particular, we describe a linear algorithm to compute the shape of 6 points from at least 4 images, a linear algorithm to compute the shape of 7 points from at least 3 images, and a linear algorithm to compute the shape of 8 points from at least 2 images. The shape representations that we compute with these 3 algorithms are related to the usual description of shape like the fundamental matrix is related to external and internal camera parameters: the homogeneous coordinates of points are uniquely defined by any of these descriptions.

The computed shape descriptions are sufficient for the identification of novel images of the same object. Moreover, the description provides a natural index from an image to the database. Thus object recognition applications can directly benefit from these representations. A previous paper [15] described the use of such model-based invariant relations between models and images in object recognition, for indexing and database building, assuming the scaled orthographic (weak perspective) projection model. A general description of an algorithm to optmize such indexing schemes can be found in [18].

So far we have discussed shape descriptors of a minimal number of points. In Section 3 we show how to enhance the shape computation to include many points simultaneously. This is achieved by building a matrix $\mathtt{M}$ of the shape descriptions of different minimal sets of the points (as described in Section 2.5). This matrix $\mathtt{M}$ is then decomposed into two matrices: one is the 3D projective basis of the points, and the other matrix is the homogeneous coordinates of the non-basis points. This results in a straightforward linear algorithm to compute shape that uses multiple points and multiple images, as describe in Section 4.

We made the distinction above between algorithms that compute camera geometry first and shape next, and algorithms that compute shape first and camera geometry next. Under the weak perspective projection, the distinction does not have to be that sharp. In [12] (see also [14, 15]), Tomasi & Kanade described a factorization method for the weak perspective projection model, where the data is linearly divided into shape components and camera geometry components. This factorization does not seem possible for the more general perspective projection model; however, the symmetrical formalism described below, where camera parameters are fully interchangeable with $3D$ model parameters, gives us a non-linear factorization of the data into shape and camera components.

To proceed, we cannot use a global method such as SVD decomposition as proposed in [12]; instead, we can isolate and compute each component separately, as proposed in [15, 16]. The algorithms described in this paper, together with the algorithms developed for the multi-camera geometry, allow us to accomplish this goal. The results described below for multi-point geometry show how to compute shape linearly from trajectories of points across many images; using the known results for multi-camera geometry, the camera geometry is computed linearly from many points. Hopefully, the 2 approaches can be combined to obtain better results in both computations.

2 Algebraic and Geometrical derivation of Results

In this section we derive the bilinear, trilinear, and quadrilinear results described in the introduction, using a formalism analogous to [3]. We show duality between multi-frame and multi-point geometries: every relation between the vector coordinates of one point in many images has an almost identical equivalent here, a relation between the vector coordinates of many points in 1 image. This duality follows from the employment of a symmetrical formalism to describe multi-point and multi-camera geometry, where $3D$ coordinates of points and projection parameters are interchangeable (see Section 2.2).

2.1 Problem definition

Consider a model which includes more than 5, $3D$ points, where each point is represented by its vector of homogeneous coordinates $\mathbf{M}_i = [X_i, Y_i, Z_i, W_i] \in \mathcal{P}^3$. W.l.o.g. (assuming uncalibrated cameras) we choose a coordinate system where the first 5 points are the standard projective basis, and $\mathbf{M}_i$ denotes the coordinates of the (i+5) point. Thus the shape matrix of the model $\mathtt{M}$, a $4 \times n$ matrix is:

$$\mathtt{M} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & X_1 & \dots & X_n \\ 0 & 1 & 0 & 0 & 1 & Y_1 & \dots & Y_n \\ 0 & 0 & 1 & 0 & 1 & Z_1 & \dots & Z_n \\ 0 & 0 & 0 & 1 & 1 & W_1 & \dots & W_n \end{pmatrix} \quad (1)$$

Similarly and again w.l.o.g, we choose the image coordinates of the first 4 points to be the standard projective basis in $\mathcal{P}^2$. Let $\mathbf{m}_i = [a_i, b_i, c_i] \in \mathcal{P}^2$ denote the vector of homogeneous coordinates of the $(i+5)$ point in the image. The projected shape matrix $\mathtt{m}$, a $3 \times n$ matrix, is:

$$\mathtt{m} = \begin{pmatrix} 1 & 0 & 0 & 1 & a_0 & \dots & a_n \\ 0 & 1 & 0 & 1 & b_0 & \dots & b_n \\ 0 & 0 & 1 & 1 & c_0 & \dots & c_n \end{pmatrix}$$

Since the image shape matrix $\mathtt{m}$ is a projection of the shape matrix $\mathtt{M}$, there exists a 3×4 projection matrix $\mathtt{P}$ such that the following equality holds in $\mathcal{P}^2$:

$$\mathtt{P} \cdot \mathtt{M} = \mathtt{m} \quad (2)$$

Given our particular selection of projective bases, and using the fact that the first 4 points are transformed from a basis in $\mathcal{P}^3$ to a basis in $\mathcal{P}^2$, the projection matrix $\mathtt{P}$ must be of the form [2]:

$$\mathtt{P} = \begin{bmatrix} \alpha & 0 & 0 & \delta \\ 0 & \beta & 0 & \delta \\ 0 & 0 & \gamma & \delta \end{bmatrix}$$

We define a corresponding projection vector in $\mathcal{P}^3$

$$\mathbf{p} = (\,\alpha \quad \beta \quad \gamma \quad \delta\,)$$

2.2 Multi-frame and multi-point geometry

We start by writing down (2) for the $5+i$ model point $\mathbf{M}_i$ and the j-th frame obtained by the projection camera $\mathbf{p}_j$, producing the image point $(x_{ji}, y_{ji}, 1)$:

$$x_{ji} \;=\; \frac{\alpha_j X_i + \delta_j W_i}{\gamma_j Z_i + \delta_j W_i}\;, \quad y_{ji} \;=\; \frac{\beta_j Y_i + \delta_j W_i}{\gamma_j Z_i + \delta_j W_i}$$

Clearly these equations are completely symmetrical with respect to $\mathbf{M}_i$ and $\mathbf{p}_j$: if we interchange the 2 vectors, we will get exactly the same image point. With k frames and $n+4$ points, we get the $2k \times n$ measurements matrix $\mathtt{W}$ whose ji element is x_{ji} for $j \leq k$, and $y_{(j-k)i}$ for $k < j \leq 2k$ (cf. [12, 16]). Now if we read $\mathtt{W}$ by columns, the i-th column gives us multi-camera geometry[1]; if we read it by rows, the j-th and $(j+k)-th$ rows give us multi-point geometry in a single image.

We use this symmetry to obtain dual relations to those described in the introduction for multi-camera geometry, by reading the data-matrix by rows instead of by columns; there are the following role changes:

- The excluded vector in $\mathcal{P}^3$ is a projection operator $\mathbf{p}_j$ in multi-point geometry, and a $3D$ point $\mathbf{M}_i$ in multi-camera geometry.
- In multi-point geometry the data vectors are the 2 row-vectors $[x_{1j}, \dots, x_{nj}]$ and $[y_{1j}, \dots, y_{nj}]$ - the image coordinates in the j-th frame of all the points. In multi-frame geometry, the data vector is the column-vector $[x_{i1}, \dots, x_{ik}, y_{i1}, \dots, y_{ik}]$ - the trajectory of the $i+5$ point.

A geometrical interpretation and derivation leading to the same observation can be also found in [1].

2.3 Multi-point geometry

From now on we fix the frame and ignore the subscript j. Every $3D$ point $\mathbf{M}_i = [X_i, Y_i, Z_i, W_i]$ from the 5th on $(i \geq 0)$, which is projected to an image point $[a_i, b_i, c_i]$, defines 2 additional constraints on the projection matrix $\mathtt{P}$. We write these as linear homogeneous equations constraining the projection vector $\mathbf{p}$, $\forall\ i \geq 0$:

$$\begin{bmatrix} c_i\, X_i & 0 & -a_i\, Z_i & c_i\, W_i - a_i\, W_i \\ 0 & c_i\, Y_i & -b_i\, Z_i & c_i\, W_i - b_i\, W_i \end{bmatrix} \cdot \mathbf{p} = 0 \quad (3)$$

(note that $X_0 = Y_0 = Z_0 = W_0 = 1$). Using these equations, we obtain relations between models and images.

2.4 Bilinear equations

We start with 6 points, including the 4 image basis points and 2 extra points; thus $n = 1$. Each additional point gives 2 equations as shown in (3). The 4 equations, using points 5 and 6, can be written as:

$$\begin{bmatrix} c_0 & 0 & -a_0 & c_0 - a_0 \\ 0 & c_0 & -b_0 & c_0 - b_0 \\ c_1\, X_1 & 0 & -a_1\, Z_1 & c_1\, W_1 - a_1\, W_1 \\ 0 & c_1\, Y_1 & -b_1\, Z_1 & c_1\, W_1 - b_1\, W_1 \end{bmatrix} \cdot \mathbf{p} = 0$$

The 4×4 matrix on the left is a linear function of $\mathbf{m}_0$ and $\mathbf{m}_1$ (the projections of the 5th and 6th points).

[1] In multi-camera geometry, when single columns are considered, it is too restrictive to assume that the number of matched points is larger than 4. Therefore it is necessary to consider the columns of a measurement matrix where the $3D$ and $2D$ coordinate systems are general, using a general 3×4 projection matrix and not a point in $\mathcal{P}^3$. This is the derivation described in [3].

In order for these points to be instances of the model $\mathtt{M}$, the 4×4 matrix must be of rank 3, namely, its determinant must be 0. This gives us the following equation:

$$
\begin{aligned}
&(-a_0\, b_1 + a_0\, c_1)\,(W_1\, X_1 - Y_1\, Z_1)\ + \\
&(a_1\, b_0 - b_0\, c_1)\,(W_1\, Y_1 - Y_1\, Z_1)\ + \\
&(-a_1\, c_0 + b_1\, c_0)\,(W_1\, Z_1 - Y_1\, Z_1)\ + \\
&(-a_0\, c_1 + b_0\, c_1)\,(X_1\, Y_1 - Y_1\, Z_1)\ + \\
&(a_0\, b_1 - b_1\, c_0)\,(X_1\, Z_1 - Y_1\, Z_1)\ =\ 0
\end{aligned}
\tag{4}
$$

The shape representation is therefore the following shape vector[2]:

$$\mathbf{V6}\ =\ [W_1\, X_1,\ W_1\, Y_1,\ W_1\, Z_1,\ X_1\, Y_1,\ X_1\, Z_1] - Y_1\, Z_1 \mathbf{1}_5 \tag{5}$$

This representation can be computed from 4 pictures (or from the 3D model), and can then be used for reconstruction or object recognition.

We can rewrite (4) in a matrix form, in a dual way to the use of the fundamental matrix to describe the epipolar geometry (cf. [7]):

$$\mathbf{m}_0^T G_{01} \mathbf{m}_1 = [a_0, b_0, c_0] G_{01} \begin{bmatrix} a_1 \\ b_1 \\ c_1 \end{bmatrix} = 0$$

where

$$G_{01} = \begin{bmatrix} 0 & X_1\, Z_1 - W_1\, X_1 & -X_1\, Y_1 + W_1\, X_1 \\ W_1\, Y_1 - Y_1\, Z_1 & 0 & X_1\, Y_1 - W_1\, Y_1 \\ Y_1\, Z_1 - W_1\, Z_1 & -X_1\, Z_1 + W_1\, Z_1 & 0 \end{bmatrix}$$

Whereas the fundamental matrix depends on the camera calibration, the matrix G_{01} here depends on the $3D$ shape of the object. Unlike the fundamental matrix, it has only 4 degrees of freedom in it (up to scale), since its elements sum up to 0 (this is because we chose the special bases).

2.5 Trilinear equations:

We continue by adding 3 points to the 4 image basis points, altogether 7 points; thus $n = 2$. Now we have an overdetermined linear system of the form:

$$\begin{bmatrix} c_0 & 0 & -a_0 & c_0 - a_0 \\ 0 & c_0 & -b_0 & c_0 - b_0 \\ c_1\, X_1 & 0 & -a_1\, Z_1 & c_1\, W_1 - a_1\, W_1 \\ 0 & c_1\, Y_1 & -b_1\, Z_1 & c_1\, W_1 - b_1\, W_1 \\ c_2\, X_2 & 0 & -a_2\, Z_2 & c_2\, W_2 - a_2\, W_2 \\ 0 & c_2\, Y_2 & -b_2\, Z_2 & c_2\, W_2 - b_2\, W_2 \end{bmatrix} \cdot \mathbf{p} = 0$$

The 6×4 matrix on the left is a linear function of $\mathbf{m}_0$, $\mathbf{m}_1$, $\mathbf{m}_2$ (the projections of the 5th, 6th and 7th points). In order for these points to be instances of the model $\mathtt{M}$, the 6×4 matrix must be of rank 3, namely, the determinant of every 4×4 sub-matrix must be 0. We get $\binom{6}{4} = 15$ constraints. Using algebraic tools, we show that there are only 7 independent equations: 3 bilinear equations involving subsets of 6 points, and 4 new trilinear equations. The 4 new constraints give us the following set of equations:

$$\begin{bmatrix} 0 & 0 \\ -a_0\, a_2\, c_1 + a_2\, c_0\, c_1 & 0 \\ a_0\, a_2\, c_1 - a_0\, c_1\, c_2 & 0 \\ 0 & -a_0\, c_1\, c_2 + b_0\, c_1\, c_2 \\ 0 & -a_2\, b_0\, c_1 + a_2\, c_0\, c_1 \\ 0 & a_2\, b_0\, c_1 - b_0\, c_1\, c_2 \\ a_0\, a_1\, c_2 - a_1\, c_0\, c_2 & a_0\, b_1\, c_2 - b_1\, c_0\, c_2 \\ 0 & 0 \\ -a_1\, a_2\, c_0 + a_1\, c_0\, c_2 & -a_2\, b_1\, c_0 + b_1\, c_0\, c_2 \\ -a_0\, a_1\, c_2 + a_0\, c_1\, c_2 & -a_0\, b_1\, c_2 + a_0\, c_1\, c_2 \\ 0 & 0 \end{bmatrix}^T \begin{pmatrix} X_1\, Y_2 - W_1\, Z_2 \\ X_1\, Z_2 - W_1\, Z_2 \\ X_1\, W_2 - W_1\, Z_2 \\ Y_1\, X_2 - W_1\, Z_2 \\ Y_1\, Z_2 - W_1\, Z_2 \\ Y_1\, W_2 - W_1\, Z_2 \\ Z_1\, X_2 - W_1\, Z_2 \\ Z_1\, Y_2 - W_1\, Z_2 \\ Z_1\, W_2 - W_1\, Z_2 \\ W_1\, X_2 - W_1\, Z_2 \\ W_1\, Y_2 - W_1\, Z_2 \end{pmatrix} = 0$$

$$\tag{6}$$

$$\begin{bmatrix} -a_0\, c_1\, c_2 + b_0\, c_1\, c_2 & 0 \\ a_0\, b_2\, c_1 - b_2\, c_0\, c_1 & 0 \\ -a_0\, b_2\, c_1 + a_0\, c_1\, c_2 & 0 \\ 0 & 0 \\ 0 & -b_0\, b_2\, c_1 + b_2\, c_0\, c_1 \\ 0 & b_0\, b_2\, c_1 - b_0\, c_1\, c_2 \\ 0 & 0 \\ -a_1\, b_0\, c_2 + a_1\, c_0\, c_2 & b_0\, b_1\, c_2 - b_1\, c_0\, c_2 \\ a_1\, b_2\, c_0 - a_1\, c_0\, c_2 & -b_1\, b_2\, c_0 + b_1\, c_0\, c_2 \\ 0 & 0 \\ a_1\, b_0\, c_2 - b_0\, c_1\, c_2 & -b_0\, b_1\, c_2 + b_0\, c_1\, c_2 \end{bmatrix}^T \begin{pmatrix} X_1\, Y_2 - W_1\, Z_2 \\ X_1\, Z_2 - W_1\, Z_2 \\ X_1\, W_2 - W_1\, Z_2 \\ Y_1\, X_2 - W_1\, Z_2 \\ Y_1\, Z_2 - W_1\, Z_2 \\ Y_1\, W_2 - W_1\, Z_2 \\ Z_1\, X_2 - W_1\, Z_2 \\ Z_1\, Y_2 - W_1\, Z_2 \\ Z_1\, W_2 - W_1\, Z_2 \\ W_1\, X_2 - W_1\, Z_2 \\ W_1\, Y_2 - W_1\, Z_2 \end{pmatrix} = 0$$

The shape representation is now the following shape vector:

$$
\begin{aligned}
\mathbf{V7}\ =\ &[X_1\, Y_2,\ X_1\, Z_2,\ X_1\, W_2,\ Y_1\, X_2,\ Y_1\, Z_2,\ Y_1\, W_2, \\
&Z_1\, X_2,\ Z_1\, Y_2,\ Z_1\, W_2,\ W_1\, X_2,\ W_1\, Y_2] - W_1\, Z_2 \mathbf{1}_{11}
\end{aligned}
\tag{7}
$$

As in multi-camera geometry, we can write each new trilinear constraint in a tensor form, using a $3 \times 3 \times 3$ shape tensor T. More specifically, we have:

$$\sum_{i,j,k} \mathsf{T}_{ijk} (\mathbf{m}_0)_i (\mathbf{m}_1)_j (\mathbf{m}_2)_k\ =\ 0$$

Each of the 4 constraints has a different shape tensor. For example, the tensor of the first constraint is:

$$T\ =\ \begin{bmatrix} 0 & 0 & Z_1\, X_2 - W_1\, X_2 \\ 0 & 0 & 0 \\ 0 & 0 & -X_1\, W_2 + W_1\, X_2 \end{bmatrix}$$

[2] $\mathbf{1}_n$ deontes the vector of length n, whose elements are all 1.

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$
$$\begin{bmatrix} 0 & 0 & -Z_1 X_2 + Z_1 W_2 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

However, there are only 11 unknowns in all 4 tensors, and writing the constraints as in (6) allows us to compute the shape vectors from 3 images or more.

2.6 Quadrilinear equations:

Finally we add 4 points to the 4 image basis points, altogether 8 points; thus $n = 3$. We get an overdetermined linear system of the form:

$$\begin{bmatrix} c_0 & 0 & -a_0 & c_0 - a_0 \\ 0 & c_0 & -b_0 & c_0 - b_0 \\ c_1 X_1 & 0 & -a_1 Z_1 & c_1 W_1 - a_1 W_1 \\ 0 & c_1 Y_1 & -b_1 Z_1 & c_1 W_1 - b_1 W_1 \\ c_2 X_2 & 0 & -a_2 Z_2 & c_2 W_2 - a_2 W_2 \\ 0 & c_2 Y_2 & -b_2 Z_2 & c_2 W_2 - b_2 W_2 \\ c_3 X_3 & 0 & -a_3 Z_3 & c_3 W_3 - a_3 W_3 \\ 0 & c_3 Y_3 & -b_3 Z_3 & c_3 W_3 - b_3 W_3 \end{bmatrix} \cdot \mathbf{p} = 0$$

The 8×4 matrix on the left is a linear function of $\mathbf{m}_0$, $\mathbf{m}_1$, $\mathbf{m}_2$, $\mathbf{m}_3$ (the projections of the 5th, 6th, 7th and 8th points). In order for these points to be instances of the model $\mathtt{M}$, the 8×4 matrix must be of rank 3, namely, the determinant of every 4×4 submatrix must be 0. We get $\binom{8}{4} = 70$ constraints. Using algebraic tools, we show that there are only 15 independent equations: 3 bilinear equations involving subsets of 6 points, and 12 trilinear equations involving 7 points.

However, there are new quadrilinear equations that define 22 independent constraints on a new 41-dimensional shape vector. One of the new 22 equations are given in Appendix A. The shape representation now is the following shape vector:

$$\begin{aligned} \mathbf{V8} = [\ & W_1 W_2 Z_3,\ W_1 X_2 Z_3,\ W_1 X_2 W_3,\ W_1 Y_2 X_3, \\ & Z_1 W_2 Y_3,\ Z_1 W_2 Z_3,\ Z_1 W_2 W_3,\ W_1 X_2 Y_3, \\ & Z_1 X_2 W_3,\ Z_1 Y_2 X_3,\ W_1 Z_2 W_3,\ W_1 W_2 X_3, \\ & W_1 W_2 Y_3,\ W_1 Y_2 Z_3,\ W_1 Y_2 W_3,\ W_1 Z_2 X_3, \\ & W_1 Z_2 Y_3,\ W_1 Z_2 Z_3,\ Y_1 Z_2 X_3,\ Y_1 Z_2 Z_3, \\ & Y_1 Z_2 W_3,\ Y_1 W_2 X_3,\ Y_1 W_2 Z_3,\ Y_1 W_2 W_3, \qquad (8) \\ & Z_1 X_2 Y_3,\ Z_1 X_2 Z_3,\ Y_1 X_2 Z_3,\ Y_1 X_2 W_3, \\ & Z_1 Z_2 Y_3,\ Z_1 Z_2 W_3,\ Z_1 W_2 X_3,\ Z_1 Y_2 Z_3, \\ & Z_1 Y_2 W_3,\ Z_1 Z_2 X_3,\ X_1 Y_2 W_3,\ X_1 Z_2 Y_3, \\ & X_1 Z_2 Z_3,\ X_1 Z_2 W_3,\ X_1 W_2 Y_3,\ X_1 W_2 Z_3, \\ & X_1 W_2 W_3] \ - \ X_1 Y_2 Z_3 \mathbf{1}_{41} \end{aligned}$$

As in multi-camera geometry, we can write each quadrilinear constraint in a tensor form, using a $3 \times 3 \times 3 \times 3$ shape tensor Q:

$$\sum_{i,j,k,l} \mathsf{Q}_{ijkl} (\mathbf{m}_0)_i (\mathbf{m}_1)_j (\mathbf{m}_2)_k (\mathbf{m}_3)_l \ = \ 0$$

Each of the 22 constraints has a different tensor. However, there are only 41 unknowns in all 22 tensors, and writing the constraints as in (9) allows us to compute all shape tensors from 2 images or more (see Appendix A).

2.7 Projective shape from shape tensors

An explicit representation of projective shape, as in (1), can be uniquely derived from any of the shape vectors above. Note that only "projective shape", namely, the shape up to a linear transformation in the projective space $\mathcal{P}^3$, is computed; with uncalibrated cameras, this is the most we can compute [2]. Clearly there are less degrees of freedom in the explicit representation as compared to any of the shape vectors. This is because there are additional non-linear constraints on the elements of the shape vectors, constraints which we have until now ignored. If these constraints are ignored and the shape vector is computed linearly, fitting an explicit representation to it may not be straightforward.

Ignoring for now this technical difficulty, the homogeneous $3D$ coordinates of each point can be recovered from the shape vectors $\mathbf{V6}$, $\mathbf{V7}$ as follows:

6 model points:

$$\frac{X_1}{W_1} = \frac{\mathbf{V6}[4] - \mathbf{V6}[5]}{\mathbf{V6}[2] - \mathbf{V6}[3]}$$
$$\frac{Y_1}{W_1} = \frac{\mathbf{V6}[4]}{\mathbf{V6}[1] - \mathbf{V6}[3]}$$
$$\frac{Z_1}{W_1} = \frac{\mathbf{V6}[5]}{\mathbf{V6}[1] - \mathbf{V6}[2]}$$

7 model points:

$$\frac{X_1}{W_1} = \frac{\mathbf{V7}[1] - \mathbf{V7}[2]}{\mathbf{V7}[11]}$$
$$\frac{Y_1}{W_1} = \frac{\mathbf{V7}[4] - \mathbf{V7}[5]}{\mathbf{V7}[10]}$$
$$\frac{Z_1}{W_1} = \frac{\mathbf{V7}[7] - \mathbf{V7}[8]}{\mathbf{V7}[10] - \mathbf{V7}[11]}$$

$$\frac{X_2}{W_2} = \frac{\mathbf{V7}[4] - \mathbf{V7}[7]}{\mathbf{V7}[6] - \mathbf{V7}[9]}$$
$$\frac{Y_2}{W_2} = \frac{\mathbf{V7}[1] - \mathbf{V7}[8]}{\mathbf{V7}[3] - \mathbf{V7}[9]}$$
$$\frac{Z_2}{W_2} = \frac{\mathbf{V7}[2] - \mathbf{V7}[5]}{\mathbf{V7}[3] - \mathbf{V7}[6]}$$

Similar derivation can be obtained from $\mathbf{V8}$.

2.8 Geometrical interpretation:

A geometrical interpretation of what's going on here is the following: A projection transformation from a $3D$ world to a $2D$ image is mapped to a point in $\mathcal{P}^3$. Every model point, from the 5th on, defines a line in $\mathcal{P}^3$ which goes through that point:

- Given 6 model points, the projection is defined by the intersection of 2 lines.
- Given 7 model points, the projection is also defined by the intersection of a line and 2 planes in $\mathcal{P}^3$: for example, the line is defined by the 5th model point, and the 2 planes are defined by the 6th and 7th points.
- Given 8 model points, the projection is also defined by the intersection of 4 planes in $\mathcal{P}^3$, where each plane is defined by one point.

It follows that by adding more points we cannot generate new constraints, and thus using 9 or more points give no additional benefit other than the redundancy that can be used to cancel noise when using this method.

3 Full reconstruction from Multiple Views and Points

W.l.o.g., consider the trilinear shape vector[3] $\mathbf{V7}$ defined in (7). As shown in Section 2.5, $\mathbf{V7}$ can be recovered linearly from three arbitrary views, or by linear least-squares methods when more views are provided. We describe next a relation among the shape vectors of different sets of 7 points. The relation has a simple and convenient form:

Result 1: *[rank 4] Given $6+n$ points, the $11 \times n$ matrix whose i-th column is the shape vector* $\mathbf{V7}$ *of the points $<1,2,\ldots,6,6+i>$, $i=1,\ldots,n$, is of rank 4.*

In other words, we first select 6 fixed points and recover the shape vectors of the sets $<1,\ldots,6,7>$, $<1,\ldots,6,8>,\ldots,<1,\ldots,6,6+n>$. We then stack the shape vectors into a $11 \times n$ matrix denoted $\mathtt{W}$. Our claim is that the resulting matrix $\mathtt{W}$ is of rank 4 (instead of 11). The proof of this result proceeds as follows:

Let $\mathbf{V7}_i$ denote the shape vector of the set of points $<1,\ldots,6,6+i>$. It follows from (7) that:

$$\mathtt{W} \;=\; [\mathbf{V7}_1 \;\ldots\; \mathbf{V7}_n] =$$

$$\begin{bmatrix} 0 & X_1 & -W_1 & 0 \\ 0 & 0 & X_1 - W_1 & 0 \\ 0 & 0 & -W_1 & X_1 \\ Y_1 & 0 & -W_1 & 0 \\ 0 & 0 & Y_1 - W_1 & 0 \\ 0 & 0 & -W_1 & Y_1 \\ Z_1 & 0 & -W_1 & 0 \\ 0 & Z_1 & -W_1 & 0 \\ 0 & 0 & -W_1 & Z_1 \\ W_1 & 0 & -W_1 & 0 \\ 0 & W_1 & -W_1 & 0 \end{bmatrix} \begin{bmatrix} X_2 & \ldots & X_n \\ Y_2 & \ldots & Y_n \\ Z_2 & \ldots & Z_n \\ W_2 & \ldots & W_n \end{bmatrix}$$

[3] A similar derivation can be described using $\mathbf{V8}$.

which proves that $\mathtt{W}$ is of rank 4[4].

The rank 4 constraint provides a convenient tool for integrating many views and many points into one scheme. This is done as follows: first, each shape vector of the set $<1,\ldots,6,6+i>$ is recovered using all available views (at least three views); second, the rank 4 of the shape matrix, whose columns are these shape vectors, is enforced via SVD, producing a new shape matrix where the errors are minimized (in least-squares sense).

More specifically, the SVD decomposition of W is:

$$\mathtt{W} = O_1 \Sigma O_2$$

where O_1, O_2 are orthonormal matrices of dimensions $11 \times n$, and $n \times n$, respectively. The matrix Σ is diagonal containing the n singular values. Because $\mathtt{W}$ should be of rank 4, we can ignore the last $n-4$ singular values. Therefore, we may replace the last $n-4$ singular values by zeros and obtain a new shape matrix $\tilde{\mathtt{W}}$. Each column of $\tilde{\mathtt{W}}$ describes the shape vector $\mathbf{V7}_i$ that is consistent with the shape vectors of all other points; thus the shape description is computed after taking into account all the available frames and all the available points.

4 Algorithm: shape from many points and many frames

In Section 2.5 we described a linear computation of shape vectors, describing the shape of 7 points, using all the available frames. In Section 3 we showed how the shape vectors of different subsets of 7 points can be put together into a low rank matrix, which can be decomposed into 2 rank-4 components: a matrix describing the shape of points $1\ldots6$, and a matrix describing the shape of points $7\ldots n$. These 2 steps can be put together into a single linear aglorithm, which computes the shape matrix $\mathtt{M}$ defined in (1), from all the given data: all frames and all points.

Consider an object with $n+6$ points:

1. Choose a subset of 6 "good" points (an algorithm on how to choose good basis points in described in [16]).
2. For every additional point M_i, $i \geq 7$:
   (a) Using Eq. (6), compute the shape vector $\mathbf{V7}_i$ of the set of 7 points $<1,2,3,4,5,6,6+i>$ using all available frames (but at least 3).
3. Define the $11 \times n$ matrix $\mathtt{W}$, whose i-th column is the shape vector $\mathbf{V7}_i$. From Result 1, the rank of $\mathtt{W}$ is 4. Let $\tilde{\mathtt{W}}$ denote the matrix of rank 4 which is the closest (in least squares) to $\mathtt{W}$; $\tilde{\mathtt{W}}$ is computed

[4] On the duality between the trilinear camera and shape tensors, we note that this result is dual to the recent result stating that all trilinear camera tensors having two fixed views satisfy a rank constraint [10]. Note also that 2 fixed views are described by 24 parameters, and 6 fixed points are again described by 24 parameters – completing the duality between the camera and shape trilinear tensors.

using SVD factorization of W (see Section 3), with the following decomposition:

$$\mathtt{W} \approx \tilde{\mathtt{W}} = \mathtt{U} \cdot \mathtt{V}$$

where U is a 11×4 matrix, and V is a $4 \times n$ matrix.

4. Notice that

$$\tilde{\mathtt{W}} = \mathtt{UTT}^{-1}\mathtt{V}$$

for every non-singular 4×4 matrix T. Compute T such that

$$\mathtt{UT} = \begin{bmatrix} 0 & X_1 & -W_1 & 0 \\ 0 & 0 & X_1 - W_1 & 0 \\ 0 & 0 & -W_1 & X_1 \\ Y_1 & 0 & -W_1 & 0 \\ 0 & 0 & Y_1 - W_1 & 0 \\ 0 & 0 & -W_1 & Y_1 \\ Z_1 & 0 & -W_1 & 0 \\ 0 & Z_1 & -W_1 & 0 \\ 0 & 0 & -W_1 & Z_1 \\ W_1 & 0 & -W_1 & 0 \\ 0 & W_1 & -W_1 & 0 \end{bmatrix}$$

for some constants $X_1,\ Y_1,\ Z_1,\ W_1$. This system can only be solved in a least-squares sense, as there are 40 equations with only 16 unknowns (the elements of T).

5. (a) $\mathtt{T}^{-1}\mathtt{V}$ is the shape matrix of point $7, \ldots, 6+n$
   (b) the coordinates of point 6 are obtained from UT and the equality above

5 Summary and discussion

We described relations between models and images, providing dual results to those previously obtained between camera geometry and images. We showed how these results can be used to design new shape reconstruction algorithms, which can be implemented completely linearly if non-linear constraints are ignored. The new relations can also be used as indices in a direct recognition algorithm, and for verification. This is because every image of the points must satisfy (4), (6), or (9); these equations give model-based invariants which can be used to identify novel images of the object. More specifically, each equation can be used to verify that a given image is an instance of a particular model (in which case the equation is satisfied).

The existence of the shape vectors provides a way to obtain shape without recovering camera geometry (i.e., shape without motion). Consider again the trilinear shape tensor. The tensor describes the shape of two points relative to a basis of 5 points, regardless of the camera geometry. Thus any collection of three or more views are sufficient for a linear solution of the shape vector — providing the means for removing the effects of noise by using multiple views. The rank 4 result (described in Section 3) provides the means for removing noise by using many points. Taken together, both the multiplicity of views (three or more) and the multiplicity of points (7 or more) contribute to a linear least-squares solution of projective shape, in the final algorithm described in Section 4. To date, this idea is the closest in spirit to the factorization method of [12] for scaled orthographic projection.

Appendix

A A quadrilinear equation:

(The quadrilinearity is seen in the shape vector, where all 8 points take part in each term; the factorization out of common denominators makes the constraint matrix appear locally trilinear.)

$$\begin{bmatrix} 0 \\ -a_0 a_1 c_2 + a_0 c_1 c_2 \\ 0 \\ 0 \\ 0 \\ -a_1 a_2 c_0 + a_1 c_0 c_2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ a_1 a_2 c_0 - a_2 c_0 c_1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ a_0 a_1 c_2 - a_1 c_0 c_2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -a_0 a_2 c_1 + a_2 c_0 c_1 \\ 0 \\ 0 \\ a_0 a_2 c_1 - a_0 c_1 c_2 \\ 0 \end{bmatrix}^T \begin{bmatrix} W1\,W2\,Z3 - X_1 Y_2 Z_3 \\ W_1 X_2 Z_3 - X_1 Y_2 Z_3 \\ W_1 X_2 W_3 - X_1 Y_2 Z_3 \\ W_1 Y_2 X_3 - X_1 Y_2 Z_3 \\ Z_1 W_2 Y_3 - X_1 Y_2 Z_3 \\ Z_1 W_2 Z_3 - X_1 Y_2 Z_3 \\ Z_1 W_2 W_3 - X_1 Y_2 Z_3 \\ W_1 X_2 Y_3 - X_1 Y_2 Z_3 \\ Z_1 X_2 W_3 - X_1 Y_2 Z_3 \\ Z_1 Y_2 X_3 - X_1 Y_2 Z_3 \\ W_1 Z_2 W_3 - X_1 Y_2 Z_3 \\ W_1 W_2 X_3 - X_1 Y_2 Z_3 \\ W_1 W_2 Y_3 - X_1 Y_2 Z_3 \\ W_1 Y_2 Z_3 - X_1 Y_2 Z_3 \\ W_1 Y_2 W_3 - X_1 Y_2 Z_3 \\ W_1 Z_2 X_3 - X_1 Y_2 Z_3 \\ W_1 Z_2 Y_3 - X_1 Y_2 Z_3 \\ W_1 Z_2 Z_3 - X_1 Y_2 Z_3 \\ Y_1 Z_2 X_3 - X_1 Y_2 Z_3 \\ Y_1 Z_2 Z_3 - X_1 Y_2 Z_3 \\ Y_1 Z_2 W_3 - X_1 Y_2 Z_3 \\ Y_1 W_2 X_3 - X_1 Y_2 Z_3 \\ Y_1 W_2 Z_3 - X_1 Y_2 Z_3 \\ Y_1 W_2 W_3 - X_1 Y_2 Z_3 \\ Z_1 X_2 Y_3 - X_1 Y_2 Z_3 \\ Z_1 X_2 Z_3 - X_1 Y_2 Z_3 \\ Y_1 X_2 Z_3 - X_1 Y_2 Z_3 \\ Y_1 X_2 W_3 - X_1 Y_2 Z_3 \\ Z_1 Z_2 Y_3 - X_1 Y_2 Z_3 \\ Z_1 Z_2 W_3 - X_1 Y_2 Z_3 \\ Z_1 W_2 X_3 - X_1 Y_2 Z_3 \\ Z_1 Y_2 Z_3 - X_1 Y_2 Z_3 \\ Z_1 Y_2 W_3 - X_1 Y_2 Z_3 \\ Z_1 Z_2 X_3 - X_1 Y_2 Z_3 \\ X_1 Y_2 W_3 - X_1 Y_2 Z_3 \\ X_1 Z_2 Y_3 - X_1 Y_2 Z_3 \\ X_1 Z_2 Z_3 - X_1 Y_2 Z_3 \\ X_1 Z_2 W_3 - X_1 Y_2 Z_3 \\ X_1 W_2 Y_3 - X_1 Y_2 Z_3 \\ X_1 W_2 Z_3 - X_1 Y_2 Z_3 \\ X_1 W_2 W_3 - X_1 Y_2 Z_3 \end{bmatrix} = 0 \qquad (9)$$

References

[1] S. Carlsson. Duality of Reconstruction and Positioning from Projective Views In these Proceedings.

[2] O. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig? In *Proceedings of the 2nd European Conference on Computer Vision*, pages 563–578, Santa Margherita Ligure, Italy, 1992. Springer-Verlag.

[3] O. Faugeras and B. Mourrain. On the geometry and algebra of the point and line correspondences between N images. In *Proceeding of the Europe-China Workshop on geometrical modeling and invariant for computer vision*. Xidian University Press, 1995.

[4] R. Hartley. Lines and points in three views - an integrated approach. In *Proceedings Image Understanding Workshop*, pages 1009–10016, San Mateo, CA, 1994. Morgan Kaufmann Publishers, Inc.

[5] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981.

[6] Q.-T. Luong and O. Faugeras. The fundamental matrix: theory, algorithms, and stability analysis. *International Journal of Computer Vision*, 1995. in press.

[7] L. Quan. Invariants of 6 points from 3 uncalibrated images. In *Proceedings of the 3rd European Conference on Computer Vision*, pages 459–470, Sweden, 1994. Springer-Verlag.

[8] A. Shashua. Trilinearity in visual recognition by alignment. In *Proceedings of the 1st European Conference on Computer Vision*, Stockholm, Sweden, May 1994.

[9] A. Shashua. Algebraic functions for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1995. in press.

[10] A. Shashua and S. Avidan. N>4 camera geometry part I: rank consistency across trilinear tensors, 1995. in preparation.

[11] A. Shashua and M. Werman. Trilinearity of three perspective views and its associated tensor. In *Proceedings of the 5th International Conference on Computer Vision*, Cambridge, MA, 1995. IEEE, Washington, DC.

[12] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9(2):137–154, 1992.

[13] B. Triggs. The geometry of projective reconstruction I: matching constraints and the joint image. Technical report, LIFIA, INRIA Rhone Alpes, 1994.

[14] S. Ullman and R. Basri. Recognition by linear combinations of models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:992–1006, 1991.

[15] D. Weinshall. Model-based invariants for 3D vision. *International Journal of Computer Vision*, 10(1):27–42, 1993.

[16] D. Weinshall and C. Tomasi. Linear and incremental acquisition of invariant shape models from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):512–517, 1995.

[17] M. Werman and A. Shashua. Elimination: An approach to the study of 3D-from-2D. In *Proceedings of the 1st International Conference on Computer Vision*, June 1995.

[18] M. Werman and D. Weinshall. Complexity of indexing: Efficient and learnable large database indexing. Technical Report 95-7, Institute of Computer Science, Hebrew University, Jerusalem, 1995.