Trilinear Tensor: The Fundamental Construct of Multiple-view Geometry and its Applications

Amnon Shashua

Institute of Computer Science
The Hebrew University of Jerusalem
Jerusalem, 91904, Israel
http://www.cs.huji.ac.il/~ shashua

Abstract. The topic of representation, recovery and manipulation of three-dimensional (3D) scenes from two-dimensional (2D) images thereof, provides a fertile ground for both intellectual theoretically inclined questions related to the algebra and geometry of the problem and to practical applications such as Visual Recognition, Animation and View Synthesis, recovery of scene structure and camera ego-motion, object detection and tracking, multi-sensor alignment, etc.
The basic materials have been known since the turn of the century, but the full scope of the problem has been under intensive study since 1992, first on the algebra of two views and then on the algebra of multiple views leading to a relatively mature understanding of what is known as "multilinear matching constraints", and the "trilinear tensor" of three or more views.
The purpose of this paper is, first and foremost, to provide a coherent framework for expressing the ideas behind the analysis of multiple views. Secondly, to integrate the various incremental results that have appeared on the subject into one coherent manuscript.

1 Introduction

Given that three-dimensional (3D) objects in the world are modeled by point or line sets, then their projection onto a number of distinct image planes produces point or line sets that are related by correspondences. The relationship between a 3D point/line and its 2D projections is easily described by a simple multilinear equation whose parameters consist of the camera location (viewing position).

From an algebraic standpoint, since the 3D-to-2D relationship juxtaposes the variables of object space, variables of image space and variables of viewing position, then one can isolate subsets of these variables and consider their properties:

- Matching Constraints: Given two or more views, it is possible to eliminate the object variables and obtain multilinear functions of image measurements (image variables) and (functions of) viewing variables. In other words, the existence of a correspondence set is an algebraic constraint(s) in disguise whose form becomes explicit via an elimination process and leaves us with a "shape invariant" function.

– Shape constraints: analogously, given a sufficient number of points one can eliminate the viewing variables and obtain functions of image measurements and object variables (known as indexing functions). In other words, it is possible to factor out the role of the changing viewing position and remain with a "view invariant" function.

The application aspect naturally follows the decomposition above and can be roughly divided into two classes:

– Applications for which irrelevant image variabilities are factored out: for example, Visual Recognition of a 3D object under changing viewing positions may use the Matching Constraints to create an equivalence class of the image space generated by all views of the object; or may use the Shape Constraints as an index into a shape library. In both cases, the desire is not to reconstruct the value of unknown variables (say, the shape of the object from image measurements), but rather to find a new representation that will facilitate the matching process between input image to library models. This class of applications includes also Object Tracking by means of image stabilization processing, and Image-based Rendering (a.k.a View Synthesis) of 3D objects directly from a sample of 2D images without recovering object shape.
– Reconstruction Applications: here the goal is to recover the value of unknown variables (shape or viewing positions) from the correspondence set. This line of applications is part of Photogrammetry with origins starting at the turn of the century. Both the Matching Constraints and the Shape Constraints provide simple and linear methods for achieving this goal, but non-linear iterative methods, such as the "block bundle adjustment" of Photogrammetry, are of much use as well.

One of the important ideas that has emerged in the recent years and is related to these issues is the factorization/elimination principles from which the multilinear matching constraints have arisen and consequently the discovery of the Trilinear Tensor which has emerged as the basic building block of 3D visual analysis.

The purpose of this paper is, first and foremost, to provide a coherent framework for expressing the ideas behind the analysis of multiple views. Secondly, to integrate the various incremental results that have appeared on the subject into one coherent manuscript.

We will start with the special case of Parallel Projection (Affine camera) model in order to illuminate the central ideas, and proceed from there to the general Perspective Projection (Projective camera) model and progress through the derivation of the Matching Constraints, the Trilinear Tensor and its properties, the Fundamental matrix, and applications.

2 N-view and n-point Geometry With an Affine Camera

The theory underlying the relationship among multiple affine views is well understood and will serve here to illuminate the goals one wishes to obtain in the general case of perspective views.

An affine view is obtained when the projecting rays emanating from a 3D object are all parallel to each other, and their intersection with an image plane forms the "image" of the object from the vantage point defined by the direction of rays. In general, we are also allowed to take pictures of pictures as well. Let the 3D world consist of n points $P_1, ..., P_n$, whose homogeneous coordinates are $(X_i, Y_i, Z_i, 1)$, $i = 1, ..., n$. Consider N distinct affine views $\psi_1, ..., \psi_N$. If we ignore problems of occlusion (assuming the object is transparent), then each view consists of n points $p_1^j, ..., p_n^j$, $j = 1, ..., N$, with non-homogeneous coordinates (x_i^j, y_i^j), $i = 1, ..., n$.

The relationship between the 3D and 2D spaces is represented by a 2×4 matrix per view:

$$p_i^j = \begin{bmatrix} a_j^\top \\ b_j^\top \end{bmatrix} P_i$$

where a_j, b_j are the rows of the matrix. The goal is to recover P_i (and/or the camera transformations a_j, b_j) from the image measurements alone (i.e., from the set of image points). If the world is undergoing only rigid transformations, then the 2×3 left principle minor of each camera transformation is a principle minor of an orthonormal matrix (rotation in space), otherwise the world may undergo affine transformations. Furthermore, one of the camera matrices may be chosen arbitrarily and, for example, set to $a = (1, 0, 0, 0)$ and $b = (0, 1, 0, 0)$. Note that even in the affine case the task is not straightforward because the camera parameters and the space coordinates (both of which are unknown) are coupled together, hence making the estimation a non-linear problem. But now consider all the measurements stacked together:

$$\begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_n^1 \\ y_1^1 & y_2^1 & \cdots & y_n^1 \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ x_1^N & x_2^N & \cdots & x_n^N \\ y_1^N & y_2^N & \cdots & y_n^N \end{bmatrix} = \begin{bmatrix} a_1^\top \\ b_1^\top \\ \cdot \\ \cdot \\ \cdot \\ a_N^\top \\ b_N^\top \end{bmatrix} \begin{bmatrix} X_1 & \cdots & X_n \\ Y_1 & \cdots & Y_n \\ Z_1 & \cdots & Z_n \\ 1 & \cdots & 1 \end{bmatrix}.$$

Thus, we clearly see that the rank of the $2N \times n$ matrix of image measurements is at most 4 (because the two matrices on the right hand side are at most of rank 4 each). This observation was made independently by Tomasi & Kanade [28] and Ullman & Basri [30] — each focusing on a different aspect of this result. Tomasi & Kanade took this result as an algorithm for reconstruction, namely, the measurement matrix can be factored into two matrices one representing motion and the other representing shape. The factorization can be done via the well known "Singular Value Decomposition" (SVD) method of Linear Algebra and the orthogonality constraints can be employed as well in order to obtain an Euclidean reconstruction. Ullman & Basri focused on the fact that the row space of the measurement matrix is spanned by four rows, thus a view (each view is represented by two rows) can be represented as a linear combination of other views — hence the "linear combination of views" result.

To understand the importance of the rank 4 result further, consider the following. Each column of the measurement matrix is a point in a $2N$ dimensional space, we call "Joint Image Space" (JIS). Each point P in the 3D world maps into a point in JIS. The rank 4 result shows that the entire 3D space lives in a 4-dimensional linear subspace of JIS. Each point in this subspace is linearly spanned by 4 points, and the coefficients of the linear combination are a function (possibly non-linear) of 3D coordinates *alone*. Therefore, the JIS represents a direct connection between 2D and 3D where the camera parameters are *eliminated* altogether. These functions are called "indexing" functions because they allow us to index into a library of 3D objects directly from the image information.

Similarly, each row of the measurement matrix is a point in a n dimensional space, we call "Joint Point Space" (JPS). Each "half" view, i.e., the collection of x or y coordinates, of a set of n points maps to a point in JPS. The rank 4 result shows that all the half views occupy a 4-dimensional linear subspace of JPS[1]. Each point in this subspace is linearly spanned by 4 points, and the coefficients of the linear combination are a function (possibly non-linear) of camera parameters *alone*. Therefore, the JPS represents a direct connection between 2D and camera parameters where the 3D coordinates are *eliminated* altogether. These functions are called "matching constraints" because they describe constraints (in this case linear) across image coordinates of a number of views that must hold uniformly for all points. Finally, JPS and JIS are dual spaces. Fig. 1 illustrates these concepts.

The affine camera case provides the insight of where the goals are in the general case. With perspective views (projective camera) there is an additional coupling between image coordinates and space coordinates, which implies that the subspaces of JIS and JPS are non-linear, they live in a manifold instead. In order to capture these manifolds we must think in terms of *elimination* because this is what actually has been achieved in the example above. The most important distinction between the affine and the general cases is that in the general case we focus on those coefficients that describe the manifolds (the matching constraints or the indexing functions). These coefficients form a tensor, the "trilinear tensor", and the rank 4 result holds again, but not in the JIS or JPS but in the new space of tensors. The linearity thus appears in the general case by focusing on yet another higher level space. The remainder of this paper is about that space, its definition and its relevance to the reconstruction and other image manipulation tasks.

3 Matching Constraints With a General Pin-hole Camera

We wish to transfer the concepts discussed above to the general pin-hole camera. In the Parallel Projection model it was easy to capture both the matching and shape constraints in a single derivation, and which applied (because they were

[1] Jacobs [14] elegantly shows that this subspace is decomposed into two skewed lines, i.e., a 3D model is mapped to two lines in R^n.

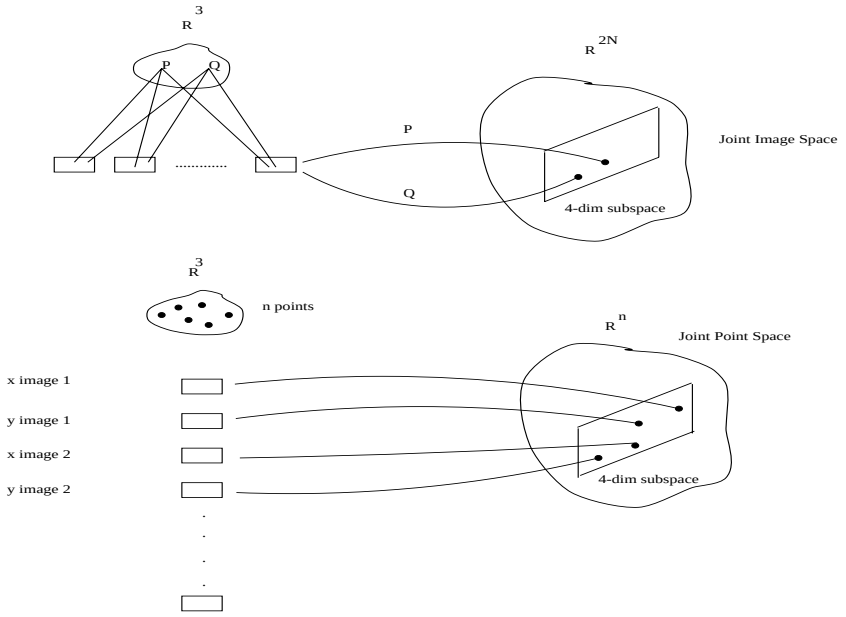

Fig. 1. Illustration of the "rank 4" result in the case of an affine camera. See text for explanation.

linear constraints) to generally N views and n points. The Pin-hole model gives rise to a slightly more complex decomposition, as described below.

A perspective view is obtained when the projecting rays emanating from a 3D object are concurrent and meet at a point known as the "center of projection". The intersection of the projecting rays with an image plane forms the "image" of the object. In general, we are also allowed to take pictures of pictures as well. Let the 3D world consist of n points $P_1, ..., P_n$, whose homogeneous coordinates are $(X_i, Y_i, Z_i, 1)$, $i = 1, ..., n$. Consider N distinct views $\psi_1, ..., \psi_N$. If we ignore problems of occlusion, then each view consists of n points $p_1^j, ..., p_n^j$, $j = 1, ..., N$, with homogeneous coordinates $(x_i^j, y_i^j, 1)$, $i = 1, ..., n$.

The relationship between the 3D and 2D spaces is represented by a 3×4 matrix per view:

$$p_i^j \cong T_j P_i$$

where T_j is the "camera matrix" and $\cong$ defines equality up to scale. In case the world undergoes rigid transformations only, then the left 3×3 principle minor of T_j is orthonormal (rotation matrix), otherwise the world may undergo general projective transformations. Without loss of generality, one of the camera matrices, say T_1, may be chosen as $[I; 0]$ where I is the 3×3 identity matrix and the fourth column is null.

Note that the major difference between the parallel projection and perspective projection models is the additional scale factor hidden in $\cong$. In case of

parallel projection, the 3D-to-2D equation is *bilinear* in the unknowns (space and viewing parameters), thus a single factorization (via SVD) is sufficient for obtaining linear relations between image and viewing variables or image and space variables. However, the perspective 3D-to-2D equation is *trilinear* in the unknowns (space, viewing parameters and the scale factor), thus two steps of factorizations are needed: the first factorization will produce a bilinear structure, and the second factorization will produce a linear structure but not in the image space but in a higher level space. This will become clear in the sequel.

Consider a single point P in space projected onto 4 views with camera matrices $[I;0], T, G, H$. To simplify the indexing notation, the image points of P will be denoted as p, p', p'', p''' in views 1 to 4, respectively. We can eliminate the scale factors as follows. Consider $p' \cong TP$, thus

$$x' = \frac{\boldsymbol{t}_1^\top P}{\boldsymbol{t}_3^\top P} \tag{1}$$

$$y' = \frac{\boldsymbol{t}_2^\top P}{\boldsymbol{t}_3^\top P}, \tag{2}$$

where $\boldsymbol{t}_i$ is the i'th row of T. Note that the third relation x'/y' is linearly spanned by the two above, thus does not add any new information. In matrix form we obtain:

$$\left[\begin{array}{c} x'\boldsymbol{t}_3^\top - \boldsymbol{t}_1^\top \\ y'\boldsymbol{t}_3^\top - \boldsymbol{t}_2^\top \end{array} \right] P = 0, \tag{3}$$

or $MP = 0$. Therefore, every view adds two rows to M whose dimensions become $2N \times 4$. For $N \geq 2$ the vanishing determinant of M ($\mid M \mid = 0$ because $P \neq 0$ is in the null space of M) provides a constraint (Matching Constraint) between the image variables and the viewing parameters — thus the space variables have been eliminated. For $N = 2$ we have exactly 1 such constraint which is bilinear in the image coordinates, for $N = 3$ we have,

$$M = \left[\begin{array}{c} x\boldsymbol{i}_3^\top - \boldsymbol{i}_1^\top \\ y\boldsymbol{i}_3^\top - \boldsymbol{i}_2^\top \\ x'\boldsymbol{t}_3^\top - \boldsymbol{t}_1^\top \\ y'\boldsymbol{t}_3^\top - \boldsymbol{t}_2^\top \\ x''\boldsymbol{g}_3^\top - \boldsymbol{g}_1^\top \\ y''\boldsymbol{g}_3^\top - \boldsymbol{g}_2^\top \end{array} \right], \tag{4}$$

where $\boldsymbol{i}_j$ is the j'th row of $[I;0]$ and every 4×4 minor has a vanishing determinant. Thus, there are 12 matching constraints that include all three views, which are arranged in three groups of 4: each group is obtained by fixing two rows corresponding to one of the three views. For example, the first group consists of $M_{1235}, M_{1236}, M_{1245}, M_{1246}$ where M_{ijkl} is the matrix formed by taking rows i, j, k, l from M. Each constraint has a trilinear form in image coordinates, hence they denoted as "trilinearities".

For $N = 4$ we obtain 16 constraints that include all four views (choose one row from M per view) and which have a quadlinear form in image coordinates.

Clearly, the case of $N > 4$ does not add anything new (because we choose at most subsets of 4 rows at a time).

The bilinear constraint was introduced by Longuett-Higgins [15] in the context of rigid motions and later by Faugeras [6] (see, [7], with references therein) for the general projective model. The trilinearities were originally introduced by Shashua [18] and the derivation adopted here is due to Faugeras & Mourrain [8] (similar derivations also concurrently appeared in [12,29]).

We will focus on the trilinearities because (i) we will show later that the bilinear constraint arises from and is a particular case of the trilinearities, and (ii) the quadlinearities do not add any new information since they can be expressed as a linear combination of the trilinearities and the bilinear constraint [8]. The following questions are noteworthy:

1. How are the coefficients of the trilinearities (per group) arranged? We will show they are arranged as a trivalent tensor with 27 entries.
2. What are the properties of the tensor? The term "properties" contain a number of issues including (i) the geometric objects that the tensor applies onto, (ii) what do contractions (subsets of coefficients) of the tensor represent? (iii) what distinguishes this tensor from a general trivalent tensor? (iv) the connection to the bilinear constraint and other 2-view geometric constructs, and (iv) applications of the Matching Constraints and methods for 3D reconstruction from the tensor.
3. Uniqueness of the solution of the tensor from the correspondence set (image measurements) — the issue of critical configurations.
4. The relationship among tensors — factorization in Tensor space, where the rank 4 constraint we saw in the affine model resurfaces again. This issue includes the notion of "tensorial operators" and their application for rendering tasks.

The first two issues (1,2) will be addressed in the remainder of this paper — the remaining issues can be found in isolation in [21,2,22] or integrated in the full version of this manuscript [19]. We will start with a brief description of notations that will assist the reader with the technical derivations.

4 Primer on Tensorial Notations

We assume that the physical 3D world is represented by the 3D projective space $\mathcal{P}^3$ (object space) and its projections onto the 2D projective space $\mathcal{P}^2$ defines the image space. We use the covariant-contravariant summation convention: a point is an object whose coordinates are specified with superscripts, i.e., $p^i = (p^1, p^2, ...)$. These are called contravariant vectors. An element in the dual space (representing hyperplanes — lines in $\mathcal{P}^2$), is called a covariant vector and is represented by subscripts, i.e., $s_j = (s_1, s_2, ...)$. Indices repeated in covariant and contravariant forms are summed over, i.e., $p^i s_i = p^1 s_1 + p^2 s_2 + ... + p^n s_n$. This is known as a contraction. For example, if p is a point incident to a line s in $\mathcal{P}^2$, then $p^i s_i = 0$. Vectors are also called 1-valence tensors. 2-valence tensors (matrices)

have two indices and the transformation they represent depends on the covariant-contravariant positioning of the indices. For example, a_i^j is a mapping from points to points, and hyperplanes to hyperplanes, because $a_i^j p^i = q^j$ and $a_i^j s_j = r_i$ (in matrix form: $Ap = q$ and $A^\top s = r$); a_{ij} maps points to hyperplanes; and a^{ij} maps hyperplanes to points. When viewed as a matrix the row and column positions are determined accordingly: in a_i^j and a_{ji} the index i runs over the columns and j runs over the rows, thus $b_j^k a_i^j = c_i^k$ is $BA = C$ in matrix form. An outer-product of two 1-valence tensors (vectors), $a_i b^j$, is a 2-valence tensor c_i^j whose i, j entries are $a_i b^j$ — note that in matrix form $C = ba^\top$. An n-valence tensor described as an outer-product of n vectors is a rank-1 tensor. The definition of the rank of a tensor is an obvious extension of the definition of the rank of a matrix: A rank-1 n-valence tensor is described as the outer product of n vectors; the rank of an n-valence tensor is the *smallest* number of rank-1 n-valence tensors with sum equal to the tensor. For example, a rank-1 trivalent tensor is $a_i b_j c_k$ where a_i,b_j and c_k are three vectors. The rank of a trivalent tensor α_{ijk} is the smallest r such that,

$$\alpha_{ijk} = \sum_{s=1}^{r} a_{is} b_{js} c_{ks} \tag{5}$$

5 The Trilinear Tensor

Consider a single point P in space projected onto 3 views with camera matrices $[I; 0], T, G$ with image points p, p', p'' respectively. Note that $P = (x, y, 1, \lambda)$ for some scalar λ. Consider $T = [A; \boldsymbol{v}']$ where A is the 3×3 principle minor of T and $\boldsymbol{v}'$ is the fourth column of T. Consider $p' \cong TP$ and eliminate the scale factor as was done previously:

$$x' = \frac{\boldsymbol{t}_1^\top P}{\boldsymbol{t}_3^\top P} = \frac{\boldsymbol{a}_1^\top p + \lambda v_1'}{\boldsymbol{a}_3^\top p + \lambda v_3'} \tag{6}$$

$$y' = \frac{\boldsymbol{t}_2^\top P}{\boldsymbol{t}_3^\top P} = \frac{\boldsymbol{a}_2^\top p + \lambda v_2'}{\boldsymbol{a}_3^\top p + \lambda v_3'}, \tag{7}$$

where $\boldsymbol{a}_i$ is the i'th row of A. These two equations can be written more compactly as follows:

$$\lambda \boldsymbol{s}'^\top \boldsymbol{v}' + \boldsymbol{s}'^\top A p = 0 \tag{8}$$

$$\lambda \boldsymbol{s}''^\top \boldsymbol{v}' + \boldsymbol{s}''^\top A p = 0 \tag{9}$$

where $\boldsymbol{s}' = (-1, 0, x)$ and $\boldsymbol{s}'' = (0, -1, y)$. Yet in a more compact form consider $\boldsymbol{s}', \boldsymbol{s}''$ as row vectors of the matrix

$$s_j^\mu = \begin{bmatrix} -1 & 0 & x' \\ 0 & -1 & y' \end{bmatrix}$$

where $j = 1, 2, 3$ and $\mu = 1, 2$. Therefore, the compact form we obtain is described below:

$$\lambda s_j^\mu v'^j + p^i s_j^\mu a_i^j = 0, \tag{10}$$

where μ is a free index (i.e., we obtain one equation per range of μ).

Similarly, let $G = [B; \boldsymbol{v}'']$ for the third view $p'' \cong GP$ and let r_k^ρ be the matrix,

$$r_k^\rho = \begin{bmatrix} -1 & 0 & x'' \\ 0 & -1 & y'' \end{bmatrix}$$

And likewise,

$$\lambda r_k^\rho v''^k + p^i r_k^\rho b_i^k = 0, \tag{11}$$

where $\rho = 1, 2$ is a free index. We can eliminate λ from equations 10 and 11 and obtain a new equation:

$$(s_j^\mu v'^j)(p^i r_k^\rho b_i^k) - (r_k^\rho v''^k)(p^i s_j^\mu a_i^j) = 0,$$

and after grouping the common terms:

$$p^i s_j^\mu r_k^\rho (v'^j b_i^k - v''^k a_i^j) = 0,$$

and the term in parenthesis is a trivalent tensor we call the *trilinear tensor*:

$$\boxed{\mathcal{T}_i^{jk} = v'^j b_i^k - v''^k a_i^j. \qquad i, j, k = 1, 2, 3} \tag{12}$$

And the tensorial equations (the 4 trilinearities) are:

$$\boxed{p^i s_j^\mu r_k^\rho \mathcal{T}_i^{jk} = 0}, \tag{13}$$

Hence, we have four trilinear equations (note that $\mu, \rho = 1, 2$). In more explicit form, these trilinearities look like:

$$\begin{aligned}
x''\mathcal{T}_i^{13}p^i - x''x'\mathcal{T}_i^{33}p^i + x'\mathcal{T}_i^{31}p^i - \mathcal{T}_i^{11}p^i = 0, \\
y''\mathcal{T}_i^{13}p^i - y''x'\mathcal{T}_i^{33}p^i + x'\mathcal{T}_i^{32}p^i - \mathcal{T}_i^{12}p^i = 0, \\
x''\mathcal{T}_i^{23}p^i - x''y'\mathcal{T}_i^{33}p^i + y'\mathcal{T}_i^{31}p^i - \mathcal{T}_i^{21}p^i = 0, \\
y''\mathcal{T}_i^{23}p^i - y''y'\mathcal{T}_i^{33}p^i + y'\mathcal{T}_i^{32}p^i - \mathcal{T}_i^{22}p^i = 0.
\end{aligned}$$

Since every corresponding triplet p, p', p'' contributes four linearly independent equations, then seven corresponding points across the three views uniquely determine (up to scale) the tensor $\mathcal{T}_i^{jk}$. Equation 12 was first introduced in [18] and the tensorial derivation leading to Equation 13 was first introduced in [20].

The trilinear tensor has been well known in disguise in the context of Euclidean line correspondences and was not identified at the time as a tensor but as a collection of three matrices (a particular contraction of the tensor as we shall see later) [25,26,32]. The link between the two and the generalization to projective space was identified later by Hartley [10,11].

Before we delve further on the properties of the trilinear tensor, we can readily identify the first application — called *image transfer* in Photogrammetric circles or a.k.a *image reprojection*. Image transfer is the task of predicting the location

of p'' from the corresponding pair p, p' given a small number of basis matching triplets p_i, p'_i, p''_i. This task can be readily achieved using the geometry of two views, simply by intersecting epipolar lines (we will later discuss these concepts) — as long as the three camera centers are not on a line, however. With the trilinearities one can achieve a general result:

Proposition 1. *A triplet of points p, p', p'' is in correspondence iff the four trilinear constraints are satisfied.*

The implication is simple: take 7 triplets p_i, p'_i, p''_i, $i = 1, ..., 7$ and recover linearly the coefficients of the tensor (for each i we have 4 linearly independent equations for the tensor). For any new pair p, p' extract the coordinates of p'' directly from the trilinearities. This will always work without singularities. In practice, due to errors in image measurements and outliers, one uses more advanced techniques for recovering the tensor (cf. [3,16,5]) and exploits further algebraic constraints among its coefficients [9].

6 Properties of the Tensor

The first striking property of the tensor is that it is an object that operates on both point and line correspondences. This becomes readily apparent from Equation 13 that simply tells us that the tensor operates on a point p, on a line passing through p', and on a line passing through p''. To see why this is so, consider $s_j^\mu p'^j = 0$ which means that s_j^1 and s_j^2 are two lines coincident with p' (lines and points in projective plane are duals of one another, thus their scalar product vanishes when they are coincident). Since any line s_j passing through p' can be described as a linear combination of the lines s_j^1 and s_j^2, and any linear combination of two trilinearities is also a trilinearity (i.e. vanishes on p, p', p''), and since the same argument holds for r_k^1 and r_k^2, we have that:

$$p^i s_j r_k \mathcal{T}_i^{jk} = 0 \tag{14}$$

where s_j is *some* line through p' and r_k is *some* line through p''. In other words,

Proposition 2. *A trilinearity represents a correspondence set of a point in the reference image and two lines (not necessarily corresponding) passing through the matching points in the remaining two images, i.e., is a point-line-line configuration. Analogously, in 3D space the configuration consists of a line-plane-plane, where the line is the optical ray of the reference image and the planes are defined by the optical centers and the image lines mentioned above.*

Figure 2 provides a pictorial description of the geometry represented by a trilinearity. The lines in the four trilinearities in Equation 13 are simply the horizontal and vertical scan lines of the image planes — we will call this representation of the trilinearities the *canonical* representation because with it each trilinearity is represented by the minimal number of non-vanishing coefficients (12 instead of

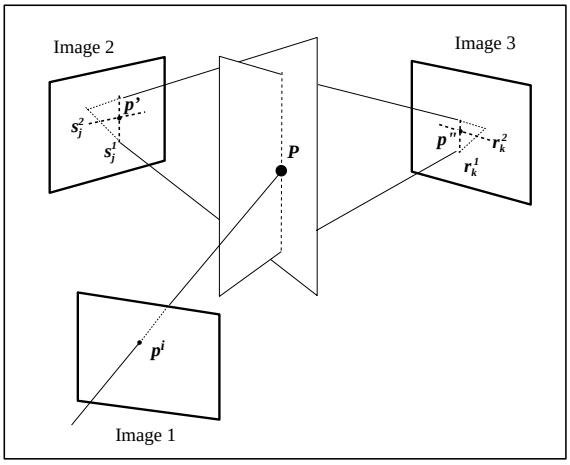

Fig. 2. Each of the four trilinear equations describes a matching between a point p in the first view, some line s_j^μ passing through the matching point p' in the second view and some line r_k^ρ passing through the matching point p'' in the third view. In space, this constraint is a meeting between a ray and two planes (Figure adopted from [2]).

27). The line-plane-plane description was first introduced by Faugeras & Mourrain [8] using Grassmann-Cayley algebra, and the analogous point-line-line description was introduced earlier in the context of Euclidean motion by Spetsakis & Aloimonos [26]. The derivation above, however, is the most straightforward one because it simply comes for free by observing how Equation 13 is organized. Finally, by similar observation one can see that a triplet of matching lines provides 2 trilinearities[2], thus 13 triplets of matching lines are sufficient for solving (linearly) for the tensor.

Before we continue further consider the applications of the point-line-line property. The first application is straightforward, whereas the second requires some elaboration. Consider a polyhedral object, like a rooftop of a house. Say the corner of the roof is visible in the first image, but is occluded in the remaining images (second and third). Thus the image measurements consist of a point-line-line arrangement, and is sufficient for providing a constraint for camera motion (the tensor). In other words, using the remarkable property of the tensor operating on both points and lines one can enrich the available feature space significantly (see for example, [3] for an application of this nature).

The second application, naturally related, is the issue of estimating Structure and Motion directly from image spatio-temporal derivatives, rather than through explicit correspondence set (points or lines). For example, the trilinear constraint (Equation 14) can be replaced by a "model-based brightness constraint" by

[2] two contractions with covariant vectors leaves us with a covariant vector, thus three matching lines provide two linear equations for the tensor elements.

having the lines s_j and r_k become:

$$s_j = \begin{pmatrix} -I_x \\ -I_y \\ I'_t + xI_x + yI_y \end{pmatrix} \qquad r_k = \begin{pmatrix} -I_x \\ -I_y \\ I''_t + xI_x + yI_y \end{pmatrix} \tag{15}$$

where I_x, I_y are the spatial derivatives at location (x, y) and I'_t, I''_t are the temporal derivatives (the image difference) between the reference image and image two and three, respectively. Hence, every pixel with a non-vanishing gradient contributes one linear constraint for camera motion. Stein & Shashua [27] provide the details and an elaborate experimental setup and also show that there are a few subtleties (and open problems) that make a successful implementation of "direct estimation" quite challenging.

6.1 Tensor Contractions

We have discussed so far the manner by which the tensor operates on geometrical entities of points and lines and the applications arising from it. We now turn our attention to similar properties of *subsets* of the tensor arising from contractions of the tensor to bivalent tensors (matrices). There are two kinds of contractions, the first yielding the well known three matrices of line geometry, and the second provides something new in the form of homography matrices. We will start with the latter contraction.

Consider the matrix arising from the contraction,

$$\delta_k \mathcal{T}_i^{jk} \tag{16}$$

which is a 3×3 matrix, we denote by E, obtained by the linear combination $E = \delta_1 \mathcal{T}_i^{j1} + \delta_2 \mathcal{T}_i^{j2} + \delta_3 \mathcal{T}_i^{j3}$ (which is what is meant by a contraction), and δ_k is an *arbitrary* covariant vector. Clearly, if $\delta_k = r_k$ then E maps p onto p' because $p^i r_k \mathcal{T}_i^{jk} \cong p'^j$ (or $Ep \cong p'$). The question of interest, therefore, is whether E has any *general* meaning? The answer is affirmative with the details described below.

Recall that the projection of the space point P onto the second image satisfies $p' \cong TP$ where $T = [A; \boldsymbol{v}']$. Let the three camera centers be denoted by C, C', C'' of the first, second and third views respectively, i.e., $TC' = 0$. The tensor operates on the ray $\overline{CP}$ and two planes one for each image. For the second image, choose the plane $CC'P$, known as the epipolar plane, which is the plane passing through the two camera centers and the space point P. This plane intersects the second image at a point, known as the epipole, which is exactly $\boldsymbol{v}'$. Clearly, the line s_j is simply the epipolar line $p' \times \boldsymbol{v}'$ defined by the vector product of p' and $\boldsymbol{v}'$ (see Figure 3). In the third image, since δ_k is arbitrary, we have a plane that does not contain p''. Let the plane defined by the point C'' and the line δ_k in the third image plane be denoted by π. Since π does not necessarily contain p'', then the intersection of π with the epipolar plane $CC'P$ is some point $\tilde{P}$ on the ray $\overline{CP}$. Clearly, the projection of $\tilde{P}$ onto the second

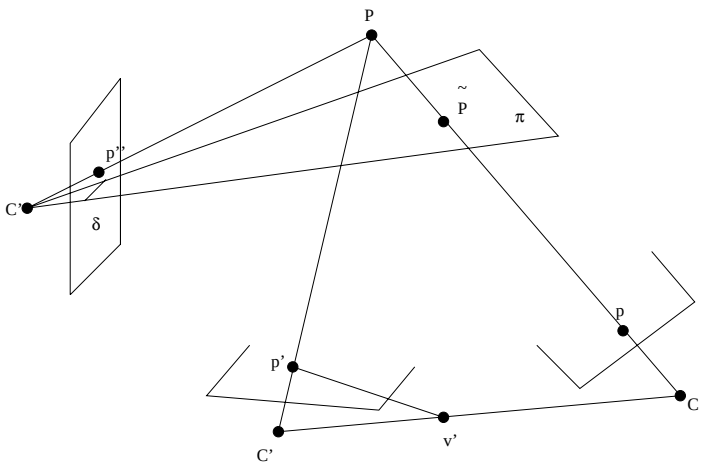

Fig. 3. The contraction $\delta_k \mathcal{T}_i^{jk}$ is a homography matrix due to the plane π determined by the center C'' and the line δ_k. See text for details.

image is a point $\tilde{p}'$ on the epipolar line (the epipolar line is the projection of the ray $\overline{CP}$ onto the second image). Hence,

$$p^i \delta_k \mathcal{T}_i^{jk} \cong \tilde{p}'^j \tag{17}$$

or $Ep \cong \tilde{p}'$. In other words, the matrix E is a 2D projective transformation (a *homography*) from the first to the second image planes via the plane π, i.e., the concatenation of the mappings (i) from first image onto π, and (ii) the mapping from π onto the second image. Stated formally,

Proposition 3. *The contraction $\delta_k \mathcal{T}_i^{jk}$ for some arbitrary δ_k is a homography matrix from image one onto image two determined by the plane containing the third camera center C'' and the line δ_k in the third image plane. Generally, the rank of E is 3. Likewise, the contraction $\delta_j \mathcal{T}_i^{jk}$ is a homography matrix between the reference image and the third image.*

Clearly, one can generate up to three distinct homography matrices because δ_k is spanned by three covariant vectors. Let the *standard* contractions be identified by selecting δ_k be $(1,0,0)$ or $(0,1,0)$ or $(0,0,1)$, thus the three homography matrices are $\mathcal{T}_i^{j1}, \mathcal{T}_i^{j2}$ and $\mathcal{T}_i^{j3}$, and we denote them by E_1, E_2, E_3 respectively. Thus, E_1, E_2 are associated with the planes passing through the horizontal and vertical scan-lines around the origin $(0,0)$ of the third image (and of course containing the center C''), and E_3 is associated with the plane parallel to the third image plane. The matrices E_1, E_2, E_3 were first introduced by Shashua & Werman [24] where further details can be found therein.

The applications of the standard homography contractions include 3D reconstruction of structure and camera motion. The camera motion is simply

$T = [E; \boldsymbol{v}']$ where E is one of the standard contractions or a linear combination of them (we will discuss the recovery of $\boldsymbol{v}'$ later). Similarly, any two homography matrices, say E_1, E_2 define a projective invariant κ defined by,

$$p' \cong E_1 p + \kappa E_2 p.$$

More details on 3D reconstruction from homography matrices can be found in [17,23]. We will encounter further applications of the standard homography contractions later in the paper.

Finally, the contractions $\mathcal{T}_1^{jk}, \mathcal{T}_2^{jk}$ and $\mathcal{T}_3^{jk}$ are the three matrices used by [25,32] to study the structure from motion problem from line correspondences (see [11], for more details).

6.2 The Bilinear Constraint

We wish to reduce the discussion back to the context of two views. We saw in Section 3 that the bilinear and trilinear constraints all arise from the same principle of vanishing determinants of 4×4 minors of M. The question of interest is what form do the coefficients of the bilinear constraint take, and how is that related to the trilinear tensor?

Starting from Equation 12, repeated for convenience below,

$$\mathcal{T}_i^{jk} = v'^j b_i^k - v''^k a_i^j$$

we will consider the case of two views as a *degenerate* instance of $\mathcal{T}_i^{jk}$ in the following way. Instead of three distinct images, we have two distinct images and the third image is a replica of the second image. Thus, the two camera matrices are $[A; \boldsymbol{v}']$ and again $[A; \boldsymbol{v}']$. Substituting A instead of B and $\boldsymbol{v}'$ instead of $\boldsymbol{v}''$ in Equation 12, we obtain a new trivalent tensor of the form:

$$\mathcal{F}_i^{jk} = v'^j a_i^k - v'^k a_i^j. \tag{18}$$

The tensor $\mathcal{F}_i^{jk}$ follows the same contraction properties as $\mathcal{T}_i^{jk}$. For example, the point-line-line property is the same with the exception that the two lines are in the same image:

$$p^i s'_j s''_k \mathcal{F}_i^{jk} = 0,$$

where s'_j and s''_k are any two lines (say the horizontal and vertical scan lines) that intersect at p'. The standard contractions apply here as well: $\delta_k \mathcal{F}_i^{jk}$ is a homography matrix from image one onto image two due to the plane π that passes through the camera center C' and the line δ_k in the second image — but now, since π contains C', it is a rank 2 homography matrix instead of rank 3.

In closer inspection one can note that 9 of the 27 elements of $\mathcal{F}_i^{jk}$ vanish and the remaining 18 are divided into two sets which differ only in sign, i.e., 9 of those elements can be arranged in a matrix F and the other 9 in $-F$, where F satisfies $p'^\top F p = 0$ and $F = [v']_x A$ where $[v']_x$ is the skew-symmetric matrix of vector products ($[v']_x u = v' \times u$); and the contraction $\delta_k \mathcal{F}_i^{jk}$ is the matrix $[\delta]_x F$.

The matrix F is known as the "Fundamental" matrix [15,6], and the constraint ${p'}^{\top} F p = 0$ is the (and only) bilinear constraint. Further details on $\mathcal{F}_i^{jk}$ and its properties can be found in [1].

Finally, given the three standard homography matrices, E_1, E_2, E_3, one can readily obtain F from the following constraint:

$$E_j{}^{\top} F + F^{\top} E_j = 0$$

which yields 18 linear equations of rank 8 for F. Similarly, cross products between columns of two homographies provide epipolar lines which can be used to recover the epipole $\boldsymbol{v}'$ — or simply recover F and then $F^{\top}\boldsymbol{v}' = 0$ will provide a solution for $\boldsymbol{v}'$.

7 Discussion

We have presented the foundations for a coherent and integrated treatment of Multiple View Geometry whose main analysis vehicle is the "trilinear tensor" which captures in a very simple and straightforward manner the basic structures associated with this problem of research.

We have left several issues out of the scope of this paper, and some issues are still an open problem. The issues we left out include (i) uniqueness of solution — the issue of critical configurations [22], (ii) properties of the tensor manifold — relation among tensors across many views, tensorial operators and applications for rendering [21,2], and (iii) Shape Constraints which are the dual of the multilinear matching constraints [31,4,13]. Issues that are still open include the tensorial structure behind the quadlinear matching constraints, the tensor governing the shape constraints and its properties.

References

1. S. Avidan and A. Shashua. Unifying two-view and three-view geometry. Technical report, Hebrew University of Jerusalem, November 1996.
2. S. Avidan and A. Shashua. View synthesis in tensor space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Puerto Rico, June 1997.
3. P. Beardsley, P. Torr, and A. Zisserman. 3D model acquisition from extended image sequences. In *Proceedings of the European Conference on Computer Vision*, April 1996.
4. S. Carlsson. Duality of reconstruction and positioning from projective views. In *Proceedings of the workshop on Scene Representations*, Cambridge, MA., June 1995.
5. R. Deriche, Z. Zhang, Q.T. Luong, and O.D. Faugeras. Robust recovery of the epipolar geometry for an uncalibrated stereo rig. In *Proceedings of the European Conference on Computer Vision*, pages 567–576, Stockholm, Sweden, May 1994. Springer-Verlag, LNCS 800.

6. O.D. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig? In *Proceedings of the European Conference on Computer Vision*, pages 563–578, Santa Margherita Ligure, Italy, June 1992.
7. O.D. Faugeras. Stratification of three-dimensional vision: projective, affine and metric representations. *Journal of the Optical Society of America*, 12(3):465–484, 1995.
8. O.D. Faugeras and B. Mourrain. On the geometry and algebra of the point and line correspondences between N images. In *Proceedings of the International Conference on Computer Vision*, Cambridge, MA, June 1995.
9. O.D. Faugeras and T. Papadopoulo. A nonlinear method for estimating the projective geometry of three views. Submitted, June 1997.
10. R. Hartley. Lines and points in three views — a unified approach. In *Proceedings of the ARPA Image Understanding Workshop*, Monterey, CA, November 1994.
11. R. Hartley. A linear method for reconstruction from lines and points. In *Proceedings of the International Conference on Computer Vision*, pages 882–887, Cambridge, MA, June 1995.
12. A. Heyden. Reconstruction from image sequences by means of relative depths. In *Proceedings of the International Conference on Computer Vision*, pages 1058–1063, Cambridge, MA, June 1995.
13. M. Irani and P. Anandan. Parallax geometry of pairs of points for 3D scene analysis. In *Proceedings of the European Conference on Computer Vision*, LNCS 1064, pages 17–30, Cambridge, UK, April 1996. Springer-Verlag.
14. D.W. Jacobs. Matching 3D models to 2D images. *International Journal of Computer Vision*, 21(1/2):123–153, January 1997.
15. H.C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981.
16. Torr P.H.S., Zisserman A., and Murray D. Motion clustering using the trilinear constraint over three views. In *Workshop on Geometrical Modeling and Invariants for Computer Vision*. Xidian University Press., 1995.
17. A. Shashua. Projective structure from uncalibrated images: structure from motion and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(8):778–790, 1994.
18. A. Shashua. Algebraic functions for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):779–789, 1995.
19. A. Shashua. Trilinear tensor: The fundamental construct of multiple-view geometry and its applications. Submitted for journal publication, June 1997.
20. A. Shashua and P. Anandan. The generalized trilinear constraints and the uncertainty tensor. In *Proceedings of the ARPA Image Understanding Workshop*, Palm Springs, CA, February 1996.
21. A. Shashua and S. Avidan. The rank4 constraint in multiple view geometry. In *Proceedings of the European Conference on Computer Vision*, Cambridge, UK, April 1996.
22. A. Shashua and S.J. Maybank. Degenerate n point configurations of three views: Do critical surfaces exist? Technical Report TR 96-19, Hebrew University of Jerusalem, November 1996.
23. A. Shashua and N. Navab. Relative affine structure: Canonical model for 3D from 2D geometry and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(9):873–883, 1996.
24. A. Shashua and M. Werman. Trilinearity of three perspective views and its associated tensor. In *Proceedings of the International Conference on Computer Vision*, June 1995.

25. M.E. Spetsakis and J. Aloimonos. Structure from motion using line correspondences. *International Journal of Computer Vision*, 4(3):171–183, 1990.
26. M.E. Spetsakis and J. Aloimonos. A unified theory of structure from motion. In *Proceedings of the ARPA Image Understanding Workshop*, 1990.
27. G. Stein and A. Shashua. Model based brightness constraints: On direct estimation of structure and motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Puerto Rico, June 1997.
28. C. Tomasi and T. Kanade. Shape and motion from image streams – a factorization method. *International Journal of Computer Vision*, 9(2):137–154, 1992.
29. B. Triggs. Matching constraints and the joint image. In *Proceedings of the International Conference on Computer Vision*, pages 338–343, Cambridge, MA, June 1995.
30. S. Ullman and R. Basri. Recognition by linear combination of models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-13:992—1006, 1991. Also in M.I.T AI Memo 1052, 1989.
31. D. Weinshall, M. Werman, and A. Shashua. Duality of multi-point and multi-frame geometry: Fundamental shape matrices and tensors. In *Proceedings of the European Conference on Computer Vision*, LNCS 1065, pages 217–227, Cambridge, UK, April 1996. Springer-Verlag.
32. J. Weng, T.S. Huang, and N. Ahuja. Motion and structure from line correspondences: Closed form solution, uniqueness and optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(3), 1992.