Tensorial Transfer: Representation of $N > 3$ Views of 3D scenes *

Shai Avidan
Institute of Computer Science
The Hebrew University of Jerusalem
91904 Jerusalem, Israel

Amnon Shashua
Department of Computer Science
Technion - Israel Institute of Technology
32000 Haifa, Israel

Abstract

The recently discovered "trilinear tensor" has been shown to play a similar role to the "fundamental" matrix, but with three instead of two perspective views, in representing the relative motion parameters of an observer moving in the 3D projective world. The issue of a general representation for any number of views remains an open problem. The main result presented here is captured by two operators on tensors that together enable a closed-form concatenation of any number of views. The concatenation operation shows that N tensors are sufficient for representing $N+2$ views, i.e., the tensor of any triplet of views from the $N+2$ set, can be generated in closed-form from an arbitrary collection of N tensors. These concatenation operators may be useful in the context of image synthesis, image mosaicing and video compression.

1 Introduction

In this paper we propose to represent a multitude of perspective 2D views of a 3D scene by the *minimal* number of trilinear tensors. We show what is that minimal number and derive new operators that enable a simple concatenation of trilinear tensors.

This paper continues the basic paradigm that seeks to establish a framework for representing and manipulating a multitude of views without ever going through a 3D model of the scene. In other words, if one seeks to synthesize novel views from a given small number of model views (namely, create an image from other images of the same object), it is not necessary to calibrate the cameras and to model the shape of the 3D scene. One can instead rely on certain algebraic relations that hold across the views in order to directly create any new image of the scene. The basic algebraic relations are known as the "fundamental" matrix (of two views) and the "trilinear tensor" (of three views). Hence, if two model images with correspondences are given, and a few number of point correspondences are given with a novel view, then the entire novel view can be generated by concatenating two fundamental matrices [1; 6] or by the use of the trilinear tensor [12] (see also [2] for comparative study as well).

The question of interest is how to generalize this concept to more than three views. Namely, if $N > 3$ views are given what is the best way to represent the algebraic relations among them? One possibility is to seek higher-order tensors beyond four views — however, that has been recently ruled out by [5; 13; 10]. Another possibility is based on the recent finding that tensors across certain arrangements of triplets of views occupy a low-dimensional subspace of the space of all tensors [14] — however, that result is more about finding a measure of consistency over many views, than a representation of a collection of views.

The approach closest to ours is that of [8] that suggested to concatenate the $N(N-1)/2$ fundamental matrices. We, instead, consider the concatenation of trilinear tensors.

This papers addresses the following questions: (i) given N views there are $N(N-1)(N-2)$ possible tensors (all ordered triplets of views), what is the minimal number of tensors that represent the entire set? and (ii) how can one generate new tensors without using further image measurements, i.e., what are the operators required on tensors that generate new tensors? A particular case of the latter is how does a tensor transform with the permutation of the three underlying views? We will see that the transformation is not linear, but is obtained in *closed form* using simple matrix calculations.

The answers we provide are complete, in the sense that they cannot be improved further. We show that these results provide basic tools for a relatively wide range of applications ranging from (i) better estimation of the tensors themselves, (ii) representation of a multitude of views, and (iii) manipulation of views for purposes of animation, recognition and compression.

2 Trilinear Tensor: Preliminaries

Consider two perspective views ψ, ψ' of a 3D scene. Let P be a point in 3D projective space projecting onto matching points $p \in \psi, p' \in \psi'$ in 2D projective plane. The relationship between the 3D and 2D spaces is represented by the 3×4 matrices, $[I, 0]$, $[A, v']$, i.e.,

$$p = [I, 0]P$$

*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.

$$p' \quad \cong \quad [A, v']P \tag{1}$$

We may adopt the convention that $p = (x, y, 1)^\top$, $p' = (x', y', 1)^\top$, and therefore $P = (x, y, 1, \rho)$. The coordinates $(x, y), (x'y')$ are matching points (with respect to some arbitrary image origin — say the geometric center of each image plane). The vector v' is the translational component of camera motion and is the view of the center of projection of the first camera in view ψ'. The matrix A is a 2D projective transformations (collineation, homography matrix) from ψ to ψ' induced by *some* plane in space (the plane $\rho = 0$). In a calibrated camera setting the plane $\rho = 0$ is the plane at infinity and A is the rotational component of camera motion and $\rho = 1/z$ where z is the depth of the point P in the first camera coordinate frame. For more details on the representation and methods for projective reconstruction see [4; 7; 11; 15; 9; 3].

Let s_k^l be the elements of the matrix

$$s = \begin{bmatrix} -1 & 0 & x' \\ 0 & -1 & y' \end{bmatrix}.$$

It can be verified by inspection that Eq. 1 can be represented by the following two equations ($l = 1, 2$):

$$\rho s_k^l v'^k + p^i s_k^l a_i^k = 0, \tag{2}$$

with the standard summation convention that an index that appears as a subscript and a superscript is summed over (known as a contraction). Superscripts denote contravariant indices (representing points in the 2D plane, like v') and subscripts denote covariant indices (representing lines in the 2D plane, like the rows of A). Thus, a_i^k is the element of the k'th row and i'th column of A, and v'^k is the k'th element of v'.

Similarly, the camera transformation between views ψ and ψ'' is

$$p' \cong [B, v'']P.$$

Likewise, let r_j^m be the elements of the matrix

$$r = \begin{bmatrix} -1 & 0 & x'' \\ 0 & -1 & y'' \end{bmatrix},$$

and likewise,

$$\rho r_j^m v''^j + p^i r_j^m b_i^j = 0, \tag{3}$$

Note that k and j are dummy indices (are summed over) in Equations (2) and (3), respectively. We used different dummy indices because now we are about to eliminate ρ and combine the two equations together. Likewise, l, m are free indices, therefore in the combination they must be separate indices. We eliminate ρ and obtain a new equation:

$$(s_k^l v'^k)(p^i r_j^m b_i^j) - (r_j^m v''^j)(p^i s_k^l a_i^k) = 0,$$

and after grouping the common terms:

$$s_k^l r_j^m p^i (v'^k b_i^j - v''^j a_i^k) = 0,$$

and the term in parenthesis is the trilinear tensor:

$$\boxed{\alpha_i^{jk} = v'^k b_i^j - v''^j a_i^k. \qquad i, j, k = 1, 2, 3} \tag{4}$$

And the tensorial equations (the trilinearities) are:

$$\boxed{s_k^l r_j^m p^i \alpha_i^{jk} = 0}, \tag{5}$$

Hence, we have four trilinear equations (note that $l, m = 1, 2$). In more explicit form, these functions (referred to as "trilinearities") are:

$$\begin{aligned}
x''\boldsymbol{\alpha}_i^{13}p^i - x''x'\boldsymbol{\alpha}_i^{33}p^i + x'\boldsymbol{\alpha}_i^{31}p^i - \boldsymbol{\alpha}_i^{11}p^i &= 0, \\
y''\boldsymbol{\alpha}_i^{13}p^i - y''x'\boldsymbol{\alpha}_i^{33}p^i + x'\boldsymbol{\alpha}_i^{32}p^i - \boldsymbol{\alpha}_i^{12}p^i &= 0, \\
x''\boldsymbol{\alpha}_i^{23}p^i - x''y'\boldsymbol{\alpha}_i^{33}p^i + y'\boldsymbol{\alpha}_i^{31}p^i - \boldsymbol{\alpha}_i^{21}p^i &= 0, \\
y''\boldsymbol{\alpha}_i^{23}p^i - y''y'\boldsymbol{\alpha}_i^{33}p^i + y'\boldsymbol{\alpha}_i^{32}p^i - \boldsymbol{\alpha}_i^{22}p^i &= 0.
\end{aligned}$$

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 α_i^{jk}. More details and applications can be found in [12].

Another detail that will be useful later is that certain contractions of the tensor yield collineations (homography matrices) as follows: The three 3×3 matrices $E_1 = \alpha_i^{1k}$ (the contraction $e_j \alpha_i^{jk}$ where $e = (1, 0, 0)^\top$), $E_2 = \alpha_i^{2k}$, and $E_3 = \alpha_i^{3k}$ are collineations from ψ to ψ' induced by three distinct planes (whose orientation is determined by B, v''), recovered up to a global common scale factor. Similarly, the matrices $W_k = \alpha_i^{jk}$ are three collineations from ψ to ψ'' of three distinct planes (whose orientation is determined by A, v'). For example, the "fundamental" matrix F between ψ and ψ' can be linearly determined from the tensor by:

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

which yields 18 linear equations of rank 8 for F. More details can be found in [16].

The final detail we will need is related to invariance of two views. Given two collineations — homography matrices A_{π_1} and A_{π_2} — of two distinct planes π_1, π_2, we define an invariant κ, referred to as "projective depth", that satisfies:

$$p' \cong (A_{\pi_1} + \kappa A_{\pi_2})p \tag{6}$$

In the case the same two planes are fixed, i.e., for any two views we calculate the collineations due to π_1 and π_2, then κ is invariant to all motion parameters of both views and reflects a projective measurement of the scene. In particular if we switch the two views and recompute the homography matrices (of the same two planes), κ will remain unchanged. Further details can be found in [11].

3 Tensorial Transfer Operators

Given views ψ_i, ψ_j and ψ_k denote by $T_{(i,j,k)}$ the corresponding tensor. Note that we are recycling the indices i, j, k used before as the indices of the tensor.

Consider the transformation of the tensor as we change the order of views. It is common knowledge, and clearly seen from the way the tensor is derived, that if we switch between views ψ_j and ψ_k, i.e., the tensor $T_{(i,k,j)}$, then the tensor $T_{(i,j,k)}$ simply undergoes a rearrangement of the indices, i.e., the coefficients α_i^{jk} are replaced with α_i^{kj}.

However, if we change the role of the first view, then the resulting tensor is no longer a rearrangement of the original. It appears we have three separate sets of 27 coefficients, one for each image playing the role of the first image [5]. What we show next is the extension to the three views case, namely that there is a simple, closed form, transformation among the three sets of 27 coefficients (i.e., among the three tensors). This goes as follows.

Assume we wish to compute the tensor $T_{(2,1,3)}$ from tensor $T_{(1,2,3)}$. We have seen in the previous section that each tensor can be rearranged (contracted) to yield three homography matrices, denoted by E_1, E_2, E_3. Let E_j denote the homography matrices of $T_{(1,2,3)}$ and G_j the homography matrices of tensor $T_{(2,1,3)}$. Thus each E_j is a collineation from ψ_1 onto ψ_2 associated with a plane whose orientation is determined by ψ_3 (the exact manner in which the orientation depends on ψ_3 can be found in [16], but is not important here). Likewise, each G_j is a collineation from ψ_2 onto ψ_1, but since ψ_3 has not changed, these collineations correspond to the same planes as E_j do. Therefore,

$$G_j \cong E_j^{-1} \qquad j = 1, 2, 3.$$

What is left is to find the undetermined scale factors. Since tensors are defined up to a global scale, we can set $G_1 = E_1^{-1}$, and we need to find two scale factors. In order to do that we use the "projective depth" invariance described in the previous section, as follows: Let $p \in \psi_1$ and $p' \in \psi_2$. Then,

$$p' \cong (E_1 + \kappa E_2)p,$$

and

$$p \cong (G_1 + \lambda\kappa G_2)p',$$

where λ is the desired scale factor. Thus,

$$\lambda = \frac{\frac{(E_1 p \times p')^T (p' \times E_2 p)}{\|p' \times E_2 p\|^2}}{\frac{(E_1^{-1} p' \times p)^T (p \times E_2^{-1} p')}{\|p \times E_2^{-1} p'\|^2}}$$

We can repeat this process between E_1 and E_3 and find the scale factor for G_3. Finally, the pair p, p' need not come from a real correspondence, but can be generated: for example, let $p = (1,1,1)^T$ be some arbitrary point in the first image, then the point $p' = (E_1 + E_2 + E_3)p$ constitutes a legitimate matching point. This is true because a linear combination of homography matrices is an homography matrix [14].

We have, therefore, described a closed-form formula for transforming a tensor $T_{(i,j,k)}$ to tensor $T_{(j,i,k)}$. And we have,

Theorem 1 (Tensor Permutation) *The tensor $T_{(j,i,k)}$ can be obtained from tensor $T_{(i,j,k)}$ via transformation of its coefficients alone.*

We are ready now to define two unitary operators on tensors:

Definition 1 *Let O_{12} be the operator that applies to a tensor $T_{(i,j,k)}$ and returns the tensor $T_{(j,i,k)}$. Similarly, let O_{23} be the operator that applies to a tensor $T_{(i,j,k)}$ and returns the tensor $T_{(i,k,j)}$. Namely,*

$$O_{12}(T_{(i,j,k)}) = T_{(j,i,k)}$$
$$O_{23}(T_{(i,j,k)}) = T_{(i,k,j)}$$

We derive next a binary operator on tensors which will constitute our last operator. First, definition:

Definition 2 (Tensorial Transfer) *Let $\times_j$ be a binary operator that applies to $T_{(i,j,k)}$ as the first operand and to $T_{(j,k,l)}$ as the second operand, and returns the tensor $T_{(i,k,l)}$. Namely,*

$$T_{(i,j,k)} \times_j T_{(j,k,l)} = T_{(i,k,l)}.$$

The binary operator $\times_j$, referred to as the "tensorial transfer" operator, is derived very similarly to the way Theorem 1 was derived. The homography matrices of tensor $T_{(i,l,k)}$ are from ψ_i onto ψ_l, where the planes are determined by ψ_k. These are the product of the corresponding homography matrices of $T_{(i,j,k)}$ (from ψ_i to ψ_j) and of the homography matrices of $O_{23}(T_{(j,k,l)})$ (from ψ_j onto ψ_l). The planes are fixed because they are determined by ψ_k which hasn't changed. The scale factors can be determined using the projective-depth invariance, as before. Finally, $T_{(i,k,l)} = O_{23}(T_{(i,l,k)})$.

We are ready now for our main result which shows how to concatenate $\times_j$ across $N+2$ views and thereby obtain a complete representation with N tensors (instead of $N(N-1)(N-2)$ tensors).

Theorem 2 *Given $N+2$ views, N tensors of arbitrary triplets (no two of which are identical), are sufficient to generate all other tensors of the set of views by means of the three operators O_{12}, O_{23} and $\times_j$.*

Proof: We prove by induction on $n = N+2$. When $n = 1$ (three views), we have six possible tensors all of which can be obtained from one of the tensors by the two unitary operators O_{12} and O_{23}.

Base Step (n=2): Let the four views be numbered $1, ..., 4$ and let $T_{(1,2,3)}$ and $T_{(2,3,4)}$ be the two given tensors. Since we can generate any internal permutation of three views by the unitary operators, we only need to show that we can generate $T_{(1,3,4)}$ and $T_{(1,2,4)}$:

$$T_{(1,2,3)} \times_2 T_{(2,3,4)} = T_{(1,3,4)}$$
$$O_{23}(T_{(1,2,3)}) \times_3 O_{12}(T_{(2,3,4)}) = T_{(1,2,4)}.$$

A similar procedure can be applied for every other pair of tensors.

Induction Step: Assume induction principle holds for n, we wish to prove the claim for $n+1$. We are given an $(n+1)$'th tensor $T_{(i,j,n+1)}$ for some arbitrary ψ_i, ψ_j, $0 \leq i, j \leq n$. We wish to generate all the new tensors $T_{(x,y,n+1)}$ for all $0 \leq x, y \leq n$ (the rest are generated by internal permutations using unitary operators). This is done as follows:

$$T_{(y,i,j)} \times_i T_{(i,j,n+1)} = T_{(y,j,n+1)}$$
$$T_{(x,i,j)} \times_i T_{(i,j,n+1)} = T_{(x,j,n+1)}$$
$$O_{23}(O_{12}(T_{(y,j,n+1)})) = T_{(j,n+1,y)}$$
$$T_{(x,j,n+1)} \times_j T_{(j,n+1,y)} = T_{(x,n+1,y)}$$
$$O_{23}(T_{(x,n+1,y)}) = T_{(x,y,n+1)},$$

where the existence of $T_{(y,i,j)}$ and $T_{(x,i,j)}$ come from the induction principle on n, because $i, j, x, y \leq n$. $\square$

4 Applications

The unitary and binary tensorial operators derived in the previous section enables one to store a single tensor for each additional view, and to have all additional computations done in the "tensor space" rather than in the image space. For example, for the sake of numerical stability it is recommended to compute tensors of views that are as far apart as possible (large base-line), however, for several tasks of interest one is interested in accumulating a "sliding window" of views, i.e., compute tensors $T_{(1,2,3)}, T_{(2,3,4)}, T_{(4,5,6)}$..... The tensorial operators allow us to compute tensors from farther apart views and then derive from them the sliding-window tensor arrangement. One advantage of a sliding-window is the possibility to perform an extended Kalman filter (EKF) on the contribution of new views.

Another application of these operators is the possibility to create tensors of views that have few or none matching points in common. This situation arises in practice when one wishes to process a large sequence of views, for purposes of video animation, video indexing, and video compression. Most, if not all, of the methods for handling 3D-from-2D geometry would not be suitable for long sequences (unless one is willing to "chain" image measurements along the sequence which will inevitable become numerically unstable). Therefore, the tensorial operators are potentially a strong tool in this context because the inter-relation between distant views may be captured without the necessary "book-keeping" of what image point is seen in what view.

We leave further details and implementation of these applications for future work. Preliminary tests have so far demonstrated the feasibility of these applications.

5 Summary

This paper has presented unitary and binary operators on trilinear tensors for purposes of concatenating the relative geometry of triplet of views. As a byproduct we have shown that given $N+2$ views, N tensors associated with arbitrary triplets of views are sufficient to generate all other tensors of the set of views.

The material presented in this paper is important for the task of image synthesis from a set of model images of a 3D scene — a topic with growing interest in the current literature. Previous work in this area is either limited to three or four views, or is limited to concatenation of fundamental matrices. Therefore, this paper presents first results on extending the manner in which a sequence of N views can be internally represented for visual processing.

References

[1] E.B. Barrett, M.H. Brill, N.N. Haag, and P.M. Payton. Invariant linear methods in photogrammetry and model-matching. In J.L. Mundy and A. Zisserman, editors, *Applications of invariances in computer vision.* MIT Press, 1992.

[2] E.B. Barrett, P.M. Payton and G. Gheen. Robust algebraic invariant methods with applications in geometry and imaging. In *Proceedings of the SPIE on Remote Sensing*, San Diego, CA, July 1995.

[3] P.A. Beardsley, A. Zisserman, and D.W. Murray. Navigation using affine structure from motion. In *Proceedings of the European Conference on Computer Vision*, pages 85–96, Stockholm, Sweden, May 1994.

[4] 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.

[5] 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.

[6] O.D. Faugeras and L. Robert. What can two images tell us about a third one? In *Proceedings of the European Conference on Computer Vision*, pages 485–492, Stockholm, Sweden, May 1994.

[7] R. Hartley. Projective reconstruction and invariants from multiple images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(10):1036–1040, 1994.

[8] S. Leveau and O.D. Faugeras. 3-D scene representation as a collection of images and fundamental matrices. Technical report, INRIA, Feb. 1994.

[9] Q.T. Luong and T. Vieville. Canonic representations for the geometries of multiple projective views. In *Proceedings of the European Conference on Computer Vision*, pages 589–599, Stockholm, Sweden, May 1994. Springer Verlag, LNCS 800.

[10] 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.

[11] 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.

[12] A. Shashua. Algebraic functions for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):779–789, 1995.

[13] B. Triggs. Matching constraints and the joint image. In *Proceedings of the International Conference on Computer Vision*, pages 338–343, Cambridge, MA, June 1995.

[14] A. Shashua and S. Avidan. The rank 4 constraint in multiple (≥ 3) view geometry. Technical report, Technion, CS Dept., October 1995.

[15] A. Shashua and N. Navab. Relative affine structure: Theory and application to 3D reconstruction from perspective views. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 483–489, Seattle, Washington, 1994.

[16] A. Shashua and M. Werman. On the trilinear tensor of three perspective views and its underlying geometry. In *Proceedings of the International Conference on Computer Vision*, June 1995.