Robust Recovery of Camera Rotation from Three Frames *

B. Rousso S. Avidan A. Shashua† S. Peleg‡
Institute of Computer Science
The Hebrew University of Jerusalem
91904 Jerusalem, Israel
e-mail : rousso@cs.huji.ac.il

Abstract

Computing camera rotation from image sequences can be used for image stabilization, and when the camera rotation is known the computation of translation and scene structure are much simplified as well. A robust approach for recovering camera rotation is presented, which does not assume any specific scene structure (e.g. no planar surface is required), and which avoids prior computation of the epipole.

Given two images taken from two different viewing positions, the rotation matrix between the images can be computed from any three homgraphy matrices. The homographies are computed using the trilinear tensor which describes the relations between the projections of a 3D point into three images. The entire computation is linear for small angles, and is therefore fast and stable. Iterating the linear computation can then be used to recover larger rotations as well.

1 Introduction

Recovering camera rotation is one of the basic steps in many image sequence applications, such as electronic image stabilization. Most existing methods take one of the following two approaches. One approach is to compute the camera rotation only after computing the camera translation (the epipole) [20, 4, 9, 13]. The second approach assumes a specific 3D scene structure, e.g. assuming the existence and the detection of 3D planes in the scene [9, 19, 14, 11].

We propose a new method to recover rotations using three homography matrices, without using the epipoles and without assuming any specific 3D model. A homography is a transformation that maps the image of a 3D plane in one frame into its image in the second frame, and it can be represented as a 3×3 matrix (see Section 2). The homographies used do not correspond to any physical planes that have to be present in the scene, and therefore there is no restriction on the scene structure. The benefits of our method are increased accuracy, as epipoles are believed to be a source for error [27], and more general applicability as no specific 3D model is assumed.

A theoretical background to the presented approach is the property that homography matrices between two images form a linear space of rank 4 (See [18]). We further show that assuming small-motion ,which is typically the case in video sequence processing, the rotational component of camera motion (which is the homography due to the plane at infinity) is spanned by only three homography matrices.

There are several existing approaches for finding the three homography matrices needed to compute the camera rotation [8, 1, 2, 11, 14, 19]. Most of these methods assume that each homography corresponds to a physical planar surface in the scene. Even when several planar surfaces do exist in the scene, the accuracy of these methods decreases as more homographies are extracted.

We found that the best method for computing the needed homographies is by using the trilinear tensor between three images [16, 4, 20, 5]. The trilinear tensor is relatively accurate since it is computed from three frames, rather than only two, and no 3D scene structure is assumed. We then obtain a simple, closed-form solution for the rotational component of the camera motion from the trilinear tensor.

2 The Homography Matrix

In this section we will briefly define the homography matrices, and prove that all homography matrices between two images form a linear space of rank 4. For more detailed information on homography matrices in 3D-from-2D geometry see [3, 24, 15, 6, 19, 14, 10, 12, 9], and for more details on the rank-4 result see [18].

Let P be a point in 3D space projecting onto images Ψ, Ψ'. Let $p \in \Psi$ and $p' \in \Psi'$ be the matching points in the two image planes described by:

$$p \cong M[I;0]P$$
$$p' \cong M'[R;T]P$$

where $\cong$ denotes equality up to scale, and M denotes the transformation from the coordinates in the image plane to the camera coordinate system of the first camera, and M' is similar for the second camera. In

*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 and by the ACTS consortium under the 4th framework R&D program of the European Community, Project No. AC074 "Vanguard".

†A. Shashua is with the Dept. of Computer Science, Technion, Haifa, Israel.

‡S. Peleg is visiting David Sarnoff Research Center, Princeton, NJ 08540.

1063-6919/96 $5.00 © 1996 IEEE

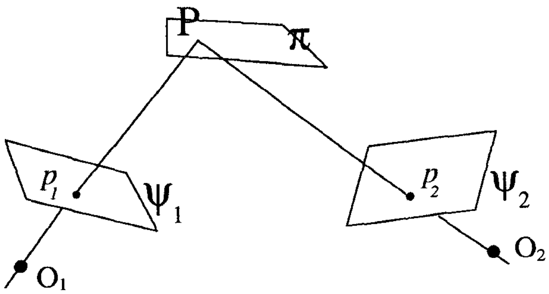


Figure 1: The homography induced by the plane π maps p to p'. p and p' are the perspective projections of any point P in the 3D plane π on the image planes Ψ and Ψ'.

the simplest configuration, M is of the form:

$$M = \begin{bmatrix} f & 0 & x_o \\ 0 & f & y_o \\ 0 & 0 & 1 \end{bmatrix}$$

where f denotes the focal length of the camera and x_o, y_o is the origin of the image plane (known as the "principle point"). The rigid camera motion is represented by the rotation matrix R and translation vector T. Taken together, $P = (x, y, 1, 1/z)^\top$, where x, y are the image coordinates of the first view, z is the depth of the point, and f is assumed to be 1, i.e., $zM^{-1}p$ are the Euclidean coordinates of the point P in the first camera frames, and:

$$p' \cong M'RM^{-1}p + \frac{1}{z}M'T.$$

When the points P live on a plane π, then $n_\pi^\top(zM^{-1}p) = d_\pi$ where n_π (normal vector) and d_π (scalar) are the parameters of π in the first camera coordinate system. We obtain $\frac{1}{z} = \frac{1}{d_\pi}n_\pi^\top(M^{-1}p)$, and therefore,

$$\begin{aligned} p' &\cong M'RM^{-1}p + \frac{1}{d_\pi}M'T(n^\top M^{-1}p) \\ &= M'(R + \frac{1}{d_\pi}Tn^\top)M^{-1}p \\ &= H_\pi p. \end{aligned}$$

In other words, the homography matrix associated with π is

$$H_\pi \cong M'(R + \frac{Tn^\top}{d_\pi})M^{-1}. \quad (1)$$

Therefore, A homography H_π is a transformation associated with the two images Ψ and Ψ', and with the 3D plane π. For any point P in π, the homography H_π maps p to p' (see Fig. 1).

For a fixed pair of cameras (M, M', T and R are constant), given a homography matrix H_π of some 3D plane π, all other homography matrices can be described by

$$\lambda H_\pi + Tn^\top \quad (2)$$

for some scale factor λ and a normal to some plane n, since the homographies differ only in scale and in the plane parameters.

Consider homography matrices $H_1, H_2, ..., H_k$ each as a column vector in a $9 \times k$ matrix. Let $H_i = \lambda_i H_\pi + Tn_i^\top$. The following can be verified by inspection:

$$\begin{aligned} &\begin{bmatrix} \; \end{bmatrix}_{9\times k} = [\lambda_1 H_\pi \cdots \lambda_k H_\pi]_{9\times k} + \\ &\begin{bmatrix} T & 0 & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{bmatrix}_{9\times 3} [n_1 \cdots n_k]_{3\times k} = \\ &= \begin{bmatrix} H_\pi & \begin{matrix} T & 0 & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{matrix} \end{bmatrix}_{9\times 4} \begin{bmatrix} \lambda_1 \cdots \lambda_k \\ n_1 \cdots n_k \end{bmatrix}_{4\times k} \end{aligned} \quad (3)$$

We have thus proven that the space of all homography matrices between two fixed views is embedded in a 4 dimensional linear subspace of R^9.

3 The Rotation Matrix

Given a sequence of images taken by a camera moving in a static scene, we would like to recover the rotation parameters of the camera. The rotation matrix R is an orthonormal matrix (up to scale). The orthonormality is the only constraint on R, which can generate five non-linear constraints on the elements of R. Note also that $M'RM^{-1}$ is the homography of the plane at infinity (letting d_π go to infinity in Eq. 1). Since solving non-linear equations is in general less stable and harder to compute than linear equations, we will first examine the case of small rotations. In addition we will assume that $M = M' = I$, i.e., that the internal parameters of the cameras are known. We will show that in addition to using linear constraints, the case of small rotations also places a strong constraint on the family of admissible homographies: the only skew-symmetric homography matrix corresponds to the plane at infinity.

When small rotations are involved, the rotation matrix R can be approximated by a matrix having the following skew-symmetric form (up to scale):

$$R \approx \hat{R} = \begin{bmatrix} 1 & \Omega_Z & -\Omega_Y \\ -\Omega_Z & 1 & \Omega_X \\ \Omega_Y & -\Omega_X & 1 \end{bmatrix}. \quad (4)$$

In this representation, the vector $\Omega = (\Omega_X, \Omega_Y, \Omega_Z)^\top$ is the rotation axis, and the magnitude of the vector is the magnitude of the rotation around this axis. Likewise, the family of all approximate homography matrices $\hat{H}_\pi$ is defined by:

$$H_\pi \approx \hat{H}_\pi \equiv \hat{R} + \frac{1}{d_\pi}Tn_\pi^\top, \quad (5)$$

where d_π is the distance from the origin to the plane π, $n_\pi^\top$ is the unit vector perpendicular to the plane toward the origin, and $T = (T_X, T_Y, T_Z)^\top$ is the camera's translation vector. Note that when $d_\pi \to \infty$, we have $\hat{H}_\pi = \hat{R}$. Therefore, the approximate rotation matrix $\hat{R}$ is also an approximate homography matrix of the plane at infinity. There are two main advantages of this framework. First, as we shall see next, the non-linear constraints associated with orthonormal matrices is replaced with a skew-symmetric condition on the family of approximate homographies — which, moreover, leads to a unique choice for the appropriate rotation (unlike the general discrete case in which the space of homography matrices is represented by a 4 parameter family). Second, shown later, is that one can obtain a direct link between the trilinear tensor and the rotation matrix, thereby avoiding the computation of the translational component of camera motion (the epipolar geometry).

We show next that the skew-symmetric form of $\hat{R}$ is unique.

Lemma 1 *The only approximate homography matrix that has an skew-symmetric form*

$$\begin{bmatrix} \xi & \nu & -\sigma \\ -\nu & \xi & \tau \\ \sigma & -\tau & \xi \end{bmatrix} \tag{6}$$

is the one associated with the plane at infinity.

Proof: if $\hat{H}_\pi$ has the skew-symmetric form (6), then $Tn_\pi^\top = d_\pi(\hat{H}_\pi - \hat{R})$ (using Eq. 5) also has the skew-symmetric form (6), because $\hat{R}$ has this skew-symmetric form (Eq. 4). However, the skew-symmetric form (6) has rank 3 (or rank 2 if $\xi = 0$), while the matrix $Tn_\pi^\top$ has rank 1, and we got a contradiction. $\Box$

This Lemma together with the rank 4 result implies that the matrix $\hat{R}$ we are looking for is *the* skew-symmetric matrix spanned by four approximate homography matrices. This can be expressed as

$$\begin{bmatrix} 1 & \Omega_Z & -\Omega_Y \\ -\Omega_Z & 1 & \Omega_X \\ \Omega_Y & -\Omega_X & 1 \end{bmatrix} = \hat{R} = c_1\hat{E}_1 + c_2\hat{E}_2 + c_3\hat{E}_3 + c_4\hat{E}_4 \tag{7}$$

where c_i, $i = 1..4$, are scalars, and $\hat{E}_i$, $i = 1..4$, are given aproximated homography matrices.

We can take the small-angle assumption a step further to show that in fact one needs elements of only three homography matrices in order to linearly span the small-angle rotation matrix:

Lemma 2 *The subset of all homography matrices that have a skew-symmetric form can be linearly spanned by elements of three homogrpahy matrices.*

Proof: Recall Eq. 2, all homography matrices H have the form

$$\lambda H_\pi + Tn^\top \ .$$

Any two columns of a skew-symmetric matrix contain all the different entries of the matrix (Eq. 6). The first two columns can be written as

$$\lambda H_{\pi(2)} + Tn_{(2)}^\top \ ,$$

where $H_{\pi(2)}$ represent the first two columns of H_π, and $n_{(2)}^\top$ represent the first two elements of $n^\top$. Consider the first two columns of k homography matrices, denoted by $H_{1_{(2)}}, H_{2_{(2)}}, ..., H_{k_{(2)}}$, each as a 6-element column vector in a $6 \times k$ matrix. The following can be verified by inspection:

$$\begin{aligned} \begin{bmatrix} \ & \ \end{bmatrix}_{6\times k} &= \left[\lambda_1 H_{\pi(2)} \cdots \lambda_k H_{\pi(2)}\right]_{6\times k} + \\ & \begin{bmatrix} T & 0 \\ 0 & T \end{bmatrix}_{6\times 2} \left[n_{1(2)} \cdots n_{k(2)}\right]_{2\times k} = \\ &= \begin{bmatrix} H_{\pi(2)} & \begin{matrix} T & 0 \\ 0 & T \end{matrix} \end{bmatrix}_{6\times 3} \begin{bmatrix} \lambda_1 \cdots \lambda_k \\ n_{1(2)} \cdots n_{k(2)} \end{bmatrix}_{3\times k} \end{aligned} \tag{8}$$

We have thus proven that the first two columns of the skew-symmetric matrix can be recoved as a linear combination of three aproximated homography matrices, as follows:

$$\begin{bmatrix} 1 & \Omega_Z \\ -\Omega_Z & 1 \\ \Omega_Y & -\Omega_X \end{bmatrix} = \hat{R}_{(2)} = c_1\hat{E}_{1_{(2)}} + c_2\hat{E}_{2_{(2)}} + c_3\hat{E}_{3_{(2)}} \ . \Box \tag{9}$$

In the next section we will see that the small-angle assumption together with the rank-3 result above gives rise to a direct method for obtaining the small-angle rotation from the trilinear tensor without recovering the translational component of camera motion and without imposing any constraint on the 3D structure of the scene.

4 Computing Small-Angle Rotation

We have seen that since the small-angle rotation matrix is skew-symmetric, its elements can be spanned by three approximate homography matrices. We have seen also that the skew-symmetric constraint provides a unique solution. The unknowns are the three coefficients that satisfy the three linear constraints of $\hat{R}_{(2)}$ (Eq. 9). Therefore, one can symbolycally solve for the coefficients, and obtain a closed-form formula expressing $\Omega_X, \Omega_Y, \Omega_z$ as a function of elements of the three given homogrpahy matrices.

What is left is to find a general way of obtaining three homography matrices without imposing further motion constraints, without recovering the translational component of camera motion, and without imposing constraints on the structure of the 3D scene. Instead we add a third view and recover the trilinear tensor associated with the three views (in a particular order).

The trilinear tensor is described in Appendix A. The tensor α_i^{jk} contains 27 entries (coefficients) as $i, j, k = 1, 2, 3$. The coefficients can be recovered linearly from at least 7 matching points across three views (see Section 5.1). The homography matrcies can be recovered directly from the tensor by setting $E_j = \alpha_i^{jk}$ (see Appendix A).

By spanning the first two columns of the rotation matrix (Eq. 9) and eliminating the linear coefficients of the combination, we obtain a very simple, closed-form, expression relating the tensor α_i^{jk} and $\Omega_X, \Omega_Y, \Omega_Z$:

$$\begin{aligned} \Omega_X &= det \begin{pmatrix} \alpha_2^{j3} \\ \alpha_2^{j3} + \alpha_3^{j2} \\ \alpha_3^{j3} - \alpha_2^{j2} \end{pmatrix} / K \\ \Omega_Y &= det \begin{pmatrix} -\alpha_1^{j3} \\ \alpha_2^{j3} + \alpha_3^{j2} \\ \alpha_3^{j3} - \alpha_2^{j2} \end{pmatrix} / K \\ \Omega_Z &= det \begin{pmatrix} \alpha_1^{j2} \\ \alpha_2^{j3} + \alpha_3^{j2} \\ \alpha_3^{j3} - \alpha_2^{j2} \end{pmatrix} / K \\ K &= det \begin{pmatrix} \alpha_2^{j2} \\ \alpha_2^{j3} + \alpha_3^{j2} \\ \alpha_3^{j3} - \alpha_2^{j2} \end{pmatrix} \end{aligned} \tag{10}$$

where α_2^{j2} stands for $(\alpha_2^{12}, \alpha_2^{22}, \alpha_2^{32})$, etc. This expression recovers directly and simply small rotations from the trilinear tensor. This result can be extended to handle also the case of general rotations by using iterations. For details, see Appendix B.

5 Video Stabilization

In this section we present a two-step algorithm for video stabilization. The first step is to compute the trilinear tensors, and the second step is to compute the camera rotations and perform derotation on the frames.

5.1 Computing the Trilinear Tensor

Following are the steps we performed to compute the trilinear tensor from a set of three images. For computation stability, all coordinates are normalized to the range of (-1, 1).

- **Selection of Corresponding Points**
  Optical flow is computed between all three possible image pairs of the three images. As corresponding points we select only those points having a high gradient, and for which all the pairwise optical flow vectors are consistent. This process results in the selection of several hundred points as corresponding in all three frames ("corresponding triplets").

- **Robust Estimation**
  From all corresponding triplets computed in the previous step, several hundred subsets of ten triplets are randomly selected [26]. For each subset the trilinear tensor is computed using Eq. 13. Each computed tensor is then applied to all matching triplets, and the single tensor for which the maximal number of triplets satisfy Eq. 13 is selected.

- **Least Square Step**
  As a final step we use all the points which satisfied the selected tensor in the previous step to solve the tensor again from Eq. 13 using a least squares method. From this tensor the homography matrices will be computed.

5.2 Derotation of Video Frames

The image sequence is stabilized with regard to the first image. This is done by selecting an arbitrary image to serve as the third image and sequentially going through the images. For each such triplet of images the trilinear tensor is computed as described above. From the compute tensor we compute $\Omega_X, \Omega_Y, \Omega_Z$ using Eq. 10. The image is then warped back to cancel the rotation, thus getting a stable sequence.

6 Experimental Results: Stabilization

Given a sequence of images, a trilinear tensor is recovered from the first frame to all other frames, using an arbitrary frame as the third frame. The rotation from the first frame to all other images is then recovered by using the the tensor values in Eq. 10. By warping back every image using the calculated rotation, we obtain a new sequence of images having no rotation compared to the first image. The remaining motion in the new sequence is only due to the original translation, thus the new sequence is smooth and clear, as can be observed in the average images.

The method was tested on outdoor scene (Fig. 2), indoor scene (Fig. 4), and on a scene with objects that are very close to the camera (Fig. 3). The method proved to be robust and efficient.

7 Concluding Remarks

A new robust method to recover the rotation of the camera was described. The main contribution to the robustness is the fact that we do not have to recover the epipoles, and the rotation is computed *directly* from three homography matrices assuming small rotations. The homography matrices are obtained from three images using the trilinear tensor parameters, and the recovery process does not assume any 3D model.

A Appendix: The Trilinear Tensor

In this section we briefly present the trilinear tensor and give an example to measure its quality. The trilinear tensor is an extension of the fundamental matrix of two views and the point and line geometry of three views [25, 22, 21], and it describes the spatial relation of three cameras. The derivation described here is taken from [17], previous work and more details can be found in [16, 20, 4, 5, 23, 7].

Let P be a point in 3D projective space projecting onto p, p', p'' three views Ψ, Ψ', Ψ'' represented by the two dimensional projective space. The relationship between the 3D and the 2D spaces is represented by the 3×4 matrices, $[I, 0]$, $[A, v']$ and $[B, v'']$, i.e.,

$$\begin{aligned} p &= [I, 0]P \\ p' &\cong [A, v']P \\ p'' &\cong [B, v'']P \end{aligned}$$

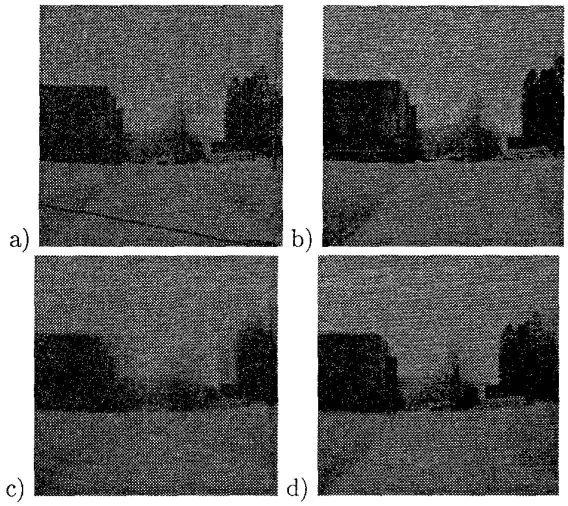


Figure 2: First sequence - outdoor scene.
a) First original frame.
b) Last (fourth) original frame. The camera was rotating while moving forward.
c) average of the four original images.
d) average of the four images after rotation cancellation. The remaining motion is only due to the original translation. The sequence looks as if it was taken using a stabilized camera.

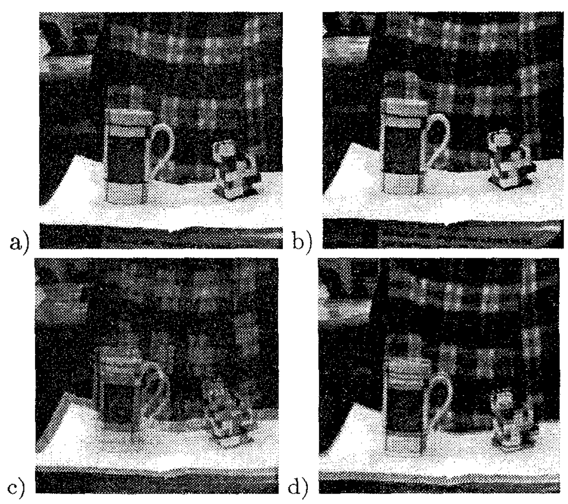


Figure 3: Second sequence - close objects.
a) First original frame.
b) Second original frame. The camera was moving and rotating around the objects.
c) average of the two original images.
d) average of the two images after rotation cancelation. The remaining motion is only due to the original translation.

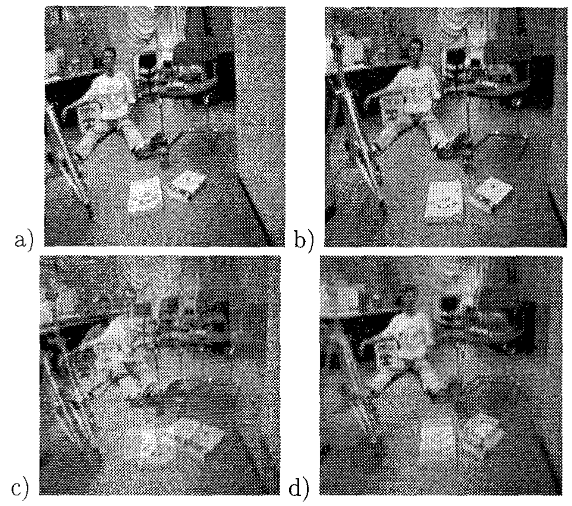


Figure 4: Third sequence - indoor scene.
a) First original frame.
b) Second original frame. The unstable camera was moving towards the man.
c) average of the two original images. The rotations make the average image unclear.
d) average of the two images after rotation cancellation. The remaining motion is only due to the original translation. The sequence looks as if it was taken using a stabilized camera.

We may adopt the convention that $p = (x, y, 1)^\top$, $p' = (x', y', 1)^\top$ and $p'' = (x'', y'', 1)^\top$, and therefore $P = [x, y, 1, \rho]$. The coordinates $(x, y), (x'y'), (x'', y'')$ are matching points (with respect to some arbitrary image origin — say the geometric center of each image plane). The matrices A and B homography matrices from Ψ to Ψ' and Ψ'', respectively, induced by *some* plane in space (the plane $\rho = 0$). The vectors v' and v'' are known as epipolar points (the projection of O, the center of projection of the first camera, onto views Ψ' and Ψ'', respectively).

The trilinear tensor is an array of 27 entries:

$$\alpha_i^{jk} = v'^k b_i^j - v''^j a_i^k. \qquad i, j, k = 1, 2, 3 \qquad (11)$$

where 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'. The tensor α_i^{jk} forms the set of coefficients of certain trilinear forms that vanish on any corresponding triplet p, p', p'' (i.e., functions of views that are invariant to object structure). These functions have the following form: let s_k^l be the matrix,

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

and, similarly, let r_j^m be the matrix,

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

Then, the tensorial equations are:

$$s_k^l r_j^m p^i \alpha_i^{jk} = 0, \tag{12}$$

with the standard summation convention that an index that appears as a subscript and superscript is summed over (known as a contraction). Hence, we have four trilinear equations (note that $l, m = 1, 2$). In more explicit form, these functions (referred to as "trilinearities") are:

$$x''\alpha_i^{13}p^i - x''x'\alpha_i^{33}p^i + x'\alpha_i^{31}p^i - \alpha_i^{11}p^i = 0,$$
$$y''\alpha_i^{13}p^i - y''x'\alpha_i^{33}p^i + x'\alpha_i^{32}p^i - \alpha_i^{12}p^i = 0,$$
$$x''\alpha_i^{23}p^i - x''y'\alpha_i^{33}p^i + y'\alpha_i^{31}p^i - \alpha_i^{21}p^i = 0,$$
$$y''\alpha_i^{23}p^i - y''y'\alpha_i^{33}p^i + y'\alpha_i^{32}p^i - \alpha_i^{22}p^i = 0.$$

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}.

The connection between the tensor and homography matrices comes from contraction properties as follows: for any vector $s_j = (s_1, s_2, s_3)$, the matrix $s_j\alpha_i^{jk}$ is a homography matrix from view Ψ to view Ψ', where s describes the orientation of the associated plane (similarly, $s_k\alpha_i^{jk}$ is a homography matrix from view Ψ to view Ψ''). In particular, when $s = (1,0,0), (0,1,0)$ and $(0,0,1)$ we get our three independent homography matrices $E_j = \alpha_i^{jk}$ [20].

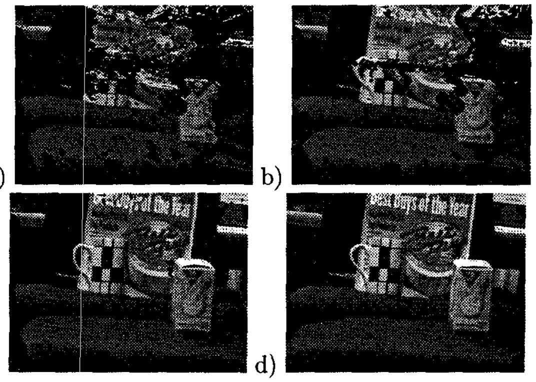
Using the homography matrices one can obtain the "fundamental" matrix F (the tensor produces 18 linear equations of rank 8 for F, for details see [20]). Fig. 5 shows an example of image reprojection (transfer) using the trilinearities, compared to using the epipolar geometry (recovered using INRIA code or using F recovered from the tensor). One can see that the best results are obtained from the trilinearities directly.

B Appendix: General Rotations Case

In the case of general (large) rotations, the following iteration scheme is proposed:

1. Compute the trilinear tensor, and use Eq. 10 to find the rotation. Results are not accurate as it was assumed that the rotation is small.

2. Derotate the second image towards the first one using the inverse of the *general rotation matrix* of the resulting angles. The rotation angles between the first image and the derotated second image will be smaller than the rotation angles between the two original images.



a) b) c) d)

Figure 5: Reprojection in different methods.
a) Reprojection using epipolar line intersection. Fundamental Matrices computed with code distributed by INRIA.
b) Reprojection using epipolar line intersection. Fundamental Matrices computed from tensor.
c) Reprojection using the tensor equations.
d) Original third image. Presented for comparison.

3. Repeat the process with the derotated second image instead of the original second image.

4. The sum of the intermediate angles will give the general (large) rotation between the two images.

This approach can be computationally expensive if in each iteration the tensor is computed and and the second image is derotated. To reduce the computational complexity we show that only manipulations of the elements of the trilinear tensor is needed in each iteration.

Let α_i^{jk} be the tensor of views $< 1, 2, 3 >$ and R be a rotation matrix. Then the tensor

$$\beta_i^{jk} = R_l^k \alpha_i^{jl} \tag{13}$$

is the trilinear tensor of views $< 1, 2', 3 >$, where view $2'$ results from rotating the second camera (view 2) around its coordinate axes by R.

Proof: α_i^{jk} can be written as

$$\alpha_i^{jk} = v'^k R''^j_i - v''^j R'^k_i \ .$$

Therefore,

$$\beta_i^{jk} = R_l^k \alpha_i^{jl} = R_l^k (v'^l R''^j_i - v''^j R'^l_i) =$$
$$(v'^l R_l^k) R''^j_i - v''^j (R_l^k R'^l_i)$$

It is therefore clear that β_i^{jk} is a tensor $< 1, \psi, 3 >$ for some view ψ. Note that $R_l^k R'^l_i$ is the combined rotation RR' and $v'^l R_l^k$ is Rv' which, taken together, means that the view ψ is a result of rotating the second image plane by R. □

Eq. 13 is mathematically equivalent to the derotation and tensor recovery steps. Using this result, we can apply the inverse of the *general rotation matrix* of the computed rotation angles to the trilinear tensor, avoiding the need to derotate an image, and recompute the tensor, in each iteration. As before, the final rotation angles are the sum of the incremental rotation angles.

References

[1] J.R. Bergen, P. Anandan, K.J. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *ECCV*, pages 237–252, Santa Margarita Ligure, May 1992.

[2] O.D. Faugeras. *Three-Dimensional Computer Vision, A Geometric Viewpoint.* MIT Press, 1993.

[3] O.D. Faugeras and F. Lustman. Let us suppose that the world is piecewise planar. In O. D. Faugeras and Georges Giralt, editors, *International Symposium on Robotics Research*, pages 33–40. MIT Press, Cambridge, MA, 1986.

[4] O.D. Faugeras and B. Mourrain. On the geometry and algebra of the point and line correspondences between n images. In *ICCV*, pages 951–956, Cambridge, MA, June 1995.

[5] R. Hartley. A linear method for reconstruction from lines and points. In *ICCV*, pages 882–887, Cambridge, MA, June 1995.

[6] R. Hartley and R. Gupta. Computing matched-epipolar projections. In *CVPR*, pages 549–555, New York, NY, 1993.

[7] A. Heyden. Reconstruction from image sequences by means of relative depths. In *ICCV*, pages 1058–1063, Cambridge, MA, June 1995.

[8] M. Irani, B. Rousso, and S. Peleg. Detecting and tracking multiple moving objects using temporal integration. In *ECCV*, pages 282–287, Santa Margarita Ligure, May 1992.

[9] M. Irani, B. Rousso, and S. Peleg. Recovery of ego-motion using image stabilization. In *CVPR*, pages 454–460, Seattle, June 1994.

[10] R. Kumar and P. Anandan. Direct recovery of shape from multiple views: A parallax based approach. In *ICPR*, Jerusalem, Israel, October 1994.

[11] J.M. Lawn and R. Cipolla. Robust egomotion estimation from affine motion parallax. In *ECCV*, pages 205–210, May 1994.

[12] Q.T. Luong and T. Vieville. Canonic representations for the geometries of multiple projective views. In *ECCV*, pages 589–599, Stockholm, Sweden, May 1994.

[13] S. Negahdaripour and S. Lee. Motion recovery from image sequences using first-order optical flow information. In *IEEE Workshop on Visual Motion*, pages 132–139, Princeton, NJ, October 1991.

[14] Harpreet Sawhney. 3d geometry from planar parallax. In *CVPR*, June 1994.

[15] A. Shashua. Projective structure from uncalibrated images: structure from motion and recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(8):778–790, 1994.

[16] A. Shashua. Algebraic functions for recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17:779–789, 1995.

[17] A. Shashua and P. Anandan. Trilinear constraints revisited: Generalized trilinear constraints and the tensor brightness constraint. In *ARPA IU Workshop*, Palm Springs, CA, February 1996.

[18] A. Shashua and S. Avidan. The rank4 constraint in multiple view geometry. In *ECCV*, Cambridge, UK, April 1996.

[19] A. Shashua and N. Navab. Relative affine structure: Theory and application to 3D reconstruction from perspective views. In *CVPR*, pages 483–489, Seattle, Washington, 1994.

[20] A. Shashua and M. Werman. Trilinearity of three perspective views and its associated tensor. In *International Conference on Computer Vision*, pages 920–925, Cambridge, MA, 1995.

[21] M.E. Spetsakis and J. Aloimonos. A unified theory of structure from motion. In *ARPA IU Workshop*, 1990.

[22] M.E. Spetsakis and J. Aloimonos. Structure from motion using line correspondences. *International Journal of Computer Vision*, 4(3):171–183, 1990.

[23] B. Triggs. Matching constraints and the joint image. In *ICCV*, pages 338–343, Cambridge, MA, June 1995.

[24] R. Tsai and T.S. Huang. Estimating three-dimensional motion parameters of a rigid planar patch, II: singular value decomposition. *IEEE Trans. on Acoustic, Speech and Signal Processing*, 30, 1982.

[25] J. Weng, T.S. Huang, and N. Ahuja. Motion and structure from line correspondences: Closed form solution, uniqueness and optimization. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14(3), 1992.

[26] P.H.S Torr, A. Zisserman, and D. Murray. Motion clustering using the trilinear constraint over three views. In *Workshop on Geometrical Modeling and Invariants for Computer Vision*. Xidian University Press., 1995.

[27] A. Zisserman and S. Maybank. A case againt epipolar geometry. In J.L. Mundy, A. Zisserman, and D. Forsyth, editors, *Applications of Invariance in Computer Vision*, pages 35–50, Ponta Delgada, Azores, 1994. Springer.