Model-based Brightness Constraints: on Direct Estimation of Structure and Motion*

G. P. Stein

Artificial Intelligence Laboratory
MIT
Cambridge, MA 02139
gideon@ai.mit.edu

A. Shashua

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

Abstract

We describe a new direct method for estimating structure and motion from image intensities of multiple views. We extend the direct methods of [9] to three views. Adding the third view enables us to solve for motion, and compute a dense depth map of the scene, directly from image spatio-temporal derivatives in a linear manner without first having to find point correspondences or compute optical flow.

We describe the advantages and limitations of this method which are then verified with experiments using real images.

1. Introduction

We present a new method for computing ego-motion and dense structure from three views. This method can be viewed as an extension of the 'direct methods' of Horn & Weldon [9] from two views (one motion) to three views (two motions). These methods are dubbed 'direct methods' because they do not require prior computation of optical flow. Within a coarse-to-fine implementation our method can handle displacements averaging up to 50 pixels for 640×480 resolution images.

By combining the constant brightness constraint [8] with the trilinear tensor [13, 5, 18] we obtain a model-based brightness constraint that describes the relationship between the spatio-temporal derivatives at each pixel in the image with the ego-motion parameters modeled by the 27 coefficients of the trilinear tensor. Hence, the model-based brightness constraint provides a linear constraint per pixel in the image which results in a highly over-constrained set of equations. We then proceed through a hierarchy of reduced motion models first by assuming calibrated cameras and then by assuming the Longuett-Higgins & Prazdny small motion model [10] resulting in reduced model-based brightness constraints for those motion models.

The advantage of our method over both optical flow methods [10, 11] and feature based methods [17] is that we obtain a linear system of equations that combines the information from all the pixels in the image. Hence, we avoid the aperture problem — thereby considerably increasing the density of reliable correspondences — and avoid the need to explicitly detect feature points.

These advantages are highlighted in a scene such as in Fig. 3. The scene contains a background of straight bars, a plaster bust in the foreground, and an oblique line cutting the bars. The background bars provide an unreliable source of information for point-to-point correspondence because of the aperture problem, thus an optical flow method of the standard variety will not perform well there; the feature points at the intersection of the bars and the line do not correspond to physical features in 3D because the line is not coincident with the background plane; and, finally, the texture on the plaster bust varies smoothly thereby giving rise only to a relatively small number of reliable features to track. Yet, our method produces a full 3D model of the scene as demonstrated in Fig. 3 *(e,f)*.

1.1. Previous Work

The 'direct methods' were pioneered by Horn & Weldon in [9]. Using only a single image pair one has N equations in $N+6$ unknowns, where N is the number of points in the image, thus additional constraints are needed. Negahdaripour & Horn [12] present a closed form solution assuming a planar or quadratic surface. Heel [6] constructs a Kalman filter to build up a structure model from more than one image pair but the core computation is fundamentally the same single image pair computation. More recently, Fermuller & Aloimonos [4] describe global geometric properties of the flow-field that give rise to direct relationships between the measurement of normal flow and the ego-motion parameters

*We acknowledge US-IS BSF contract 94-00120 and the European ACTS project AC074. General support for G.S. comes from DARPA contracts N00014-94-01-0994 and 95009-5381.

solved by means of search techniques.

This work is based on the work of Shashua & Hanna [16]. Here we describe the results of implementing these ideas in practice. During the course of implementation various subtleties and limitations were discovered.

2. Mathematical Background

Let $\boldsymbol{x}$ be a point in 3D space and its projection in a pair of images be p and p'. Then $p = [I; 0]\boldsymbol{x}$ and $p' \cong A\boldsymbol{x}$. The left 3×3 minor of A stands for a 2D projective transformation of the chosen plane at infinity and the fourth column of A stands for the epipole (the projection of the center of camera 0 on the image plane of camera 1). In particular, in a calibrated setting the 2D projective transformation is the rotational component of camera motion and the epipole is the translational component of camera motion.

We will occasionally use tensorial notations as described next. 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. An outer-product of two 1-valence tensors (vectors), $a_i b^j$, is a 2-valence tensor (matrix) c_i^j whose i, j entries are $a_i b^j$ — note that in matrix form $C = ba^\top$.

Matching image points across three views will be denoted by p, p', p''; the homogeneous coordinates will be referred to as p^i, p'^j, p''^k, or alternatively as non-homogeneous image coordinates $(x, y), (x', y'), (x'', y'')$ — hence, $p^i = (x, y, 1)$, etc.

Three views, $p = [I; 0]\boldsymbol{x}, p' \cong A\boldsymbol{x}$ and $p'' \cong B\boldsymbol{x}$, are known to produce four trilinear forms whose coefficients are arranged in a tensor representing a bilinear function of the camera matrices A, B:

$$\alpha_i^{j\,k} = t'^j b_i^k - t''^k a_i^j \tag{1}$$

where $A = [a_i^j, t'^j]$ (a_i^j is the 3×3 left minor and t' is the fourth column of A) and $B = [b_i^k, t''^k]$. The tensor acts on a triplet of matching points in the following way:

$$p^i s_j^\mu r_k^\rho \alpha_i^{j\,k} = 0 \tag{2}$$

where s_j^μ are any two lines (s_j^1 and s_j^2) intersecting at p', and r_k^ρ are any two lines intersecting p''. Since the free indices are μ, ρ each in the range 1,2, we have 4 trilinear equations (unique up to linear combinations). These constraints first became prominent in [13] and the underlying theory has been studied intensively in [17, 5, 15, 3, 19, 7, 14].

3. Model-based Brightness Constraints

Geometrically, a trilinear matching constraint is produced by contracting the tensor with the point p of image 0, *some* line coincident with p' in image 1, and *some* line coincident with p'' in image 2. In particular, we may use the tangent to the iso-brightness contour at p' and p'', respectively, and thus one can recover in principle the camera matrices across three views in the context of the "aperture" problem, as suggested by [18]. However, there still remains the problem of finding those matching tangents in the first place.

Alternatively, if we wish to use the spatio-temporal derivatives of the images at point p in the first image, we obtain two parallel lines coincident with points p' and p'' whose orientation is normal to the tangent to the iso-brightness contour at p. Thus, as a first-order approximation to the camera motion model, we obtain a constraint equation involving the unknowns $\alpha_i^{j\,k}$ and the spatio-temporal derivatives at each pixel — the constraint is linear in the unknowns. This constraint was introduced by [16] under the name "Tensor Brightness Constraint". This is briefly derived next.

Let $ax' + by' + \lambda = 0$ define a line coincident with p'. Substitute $x' = x + u'$ and $y' = y + v'$, where u', v' are the displacement component vector, to obtain $au' + bv' + c = 0$ where $c = \lambda + ax + by$. Therefore, a line s' coincident with p', i.e., $s'_j p'^j = 0$, can be described as the vector $(a, b, c - ax - by)$. In particular, the scalars a, b, c can be substituted by I_x, I_y, I'_t, respectively, given the constant brightness equation due to [8]:

$$u' I_x + v' I_y + I'_t = 0. \tag{3}$$

I_x, I_y are the components of the image gradient at location (x, y), and I'_t is the discrete temporal derivative at (x, y), i.e., $I_1(x, y) - I_0(x, y)$ where I_1 and I_0 are the image intensity values of the second and first images, respectively. Thus, the line

$$s' = \begin{pmatrix} I_x \\ I_y \\ I'_t - xI_x - yI_y \end{pmatrix} \tag{4}$$

is coincident with p' under the assumption of small motion. The orientation of s' is normal to the tangent to the iso-brightness contour at p (not p'!). Similarly, the line s'' coincident with p'' is parallel to s' but is shifted to pass through (x'', y''), hence has the form:

$$s'' = \begin{pmatrix} I_x \\ I_y \\ I''_t - xI_x - yI_y \end{pmatrix} \tag{5}$$

where I''_t is the temporal derivative between images 0 and 2. The *tensor brightness constraint* is therefore:

$$\boxed{s''_k s'_j p^i \alpha_i^{j\,k} = 0.} \tag{6}$$

We have one such equation for each point on the image where s''_k and s'_j can be computed from the image gradients and $p^i = (x, y, 1)$ are the (projective) image coordinates of the point in image 0. We wish to solve for α_i^{jk} which combines the motion and camera parameters.

Every pixel with a non-vanishing gradient contributes one linear equation to the 27 unknown parameters comprising α_i^{jk}. One can prove, however, that a configuration of a point in the first image and two *parallel* lines in images two and three, can provide a unique solution to 21 of the parameters and the remaining 6 parameters of α_i^{jk} can be determined up to an 8-fold ambiguity (8 distinct solutions):

Theorem 1 *Every point-line-line matching triplet across three images provides one linear constraint to α_i^{jk}. In case all the matching triplets have the two lines parallel to each other, then α_i^{jk} can be determined up to an 8-fold ambiguity.*

We will not prove this result here. Therefore, employing the constant brightness equation within a general three-frame sequence of images provides a linear system of equations but which is not fully constrained to solve for the tensor uniquely.

Starting from the general projective motion model, we investigate two further reduced models. The next model is defined for small-angle rotations i.e., $A = [I + [w']_x; t']$ and $B = [I + [w'']_x; t'']$ where w', w'' are the angular velocity vectors and $[\cdot]_x$ is the skew-symmetric matrix of vector products. We obtain the following model-based brightness constraint as a reduction of the tensor brightness constraint:

$$I''_t s'^{\top} t' - I'_t s''^{\top} t'' + s'^{\top}[t' w''^{\top}] v'' - s''^{\top}[t'' w'^{\top}] v' = 0, \tag{7}$$

where

$$v' = p \times s' = \begin{pmatrix} -I_y + y(I'_t - xI_x - yI_y) \\ I_x - x(I'_t - xI_x - yI_y) \\ xI_y - yI_x \end{pmatrix}$$

and

$$v'' = p \times s'' = \begin{pmatrix} -I_y + y(I''_t - xI_x - yI_y) \\ I_x - x(I''_t - xI_x - yI_y) \\ xI_y - yI_x \end{pmatrix}$$

The 24-parameter model contains 20 linearly independent parameters (instead of 27), yet still is not sufficiently constrained to produce a unique solution for the reduced tensor parameters (not proven here).

A unique solution is obtained when we reduce the motion model further to include infinitesimal translational motion with the model introduced by Longuett-Higgins & Prazdny [10]:

$$u' = \frac{1}{z}(t'_1 - xt'_3) - w'_3 y + w'_2(1 + x^2) - w'_1 xy \tag{8}$$
$$v' = \frac{1}{z}(t'_2 - yt'_3) + w'_3 x - w'_1(1 + y^2) + w'_2 xy$$

from which we obtain the 15-parameter model-based brightness constraint:

$$I''_t s^T t' - I'_t s^T t'' + s^T [t' w''^T - t'' w'^T] v = 0 \tag{9}$$

where s, v are defined below:

$$s = \begin{pmatrix} I_x \\ I_y \\ -xI_x - yI_y \end{pmatrix} \tag{10}$$

and

$$v = \begin{pmatrix} -I_y - y(xI_x + yI_y) \\ I_x + x(xI_x + yI_y) \\ xI_y - yI_x \end{pmatrix} \tag{11}$$

Given the solution to the ego-motion parameters (to be described later), one can recover the dense depth map from the following equations [9]:

$$k s^{\top} t' + v^{\top} w' + I'_t = 0 \tag{12}$$
$$k s^{\top} t'' + v^{\top} w'' + I''_t = 0 \tag{13}$$

where $K = \frac{1}{z}$ denotes the inverse depth at each pixel location. Finally, the pure translational small-motion model takes the simplest form:

$$I''_t s^T t' - I'_t s^T t'' = 0 \tag{14}$$

4. Solving the Ego-motion Parameters

The ego-motion parameters from eqn. 9 are embedded in a 15 parameter model: 6 translation parameters and the 9 outer-product terms $[t' w''^T - t'' w'^T]$. Each pixel provides one linear equation, thus N pixels provide a system $Ac = 0$ where A is an $N \times 15$ measurement matrix and c is the 15-vector of unknown parameters. The least squares solution is the eigenvector of $A^{\top} A$ corresponding to the smallest eigenvalue. The solution is unique if the rank of A is 14. However, since $s^{\top} v = s^{\top}(p \times s) = 0$ the vector

$$c_0 = (0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1)^T \tag{15}$$

is in the null space of A. Since c_0 is not an admissible solution, the rank of A is 13, and the null space of A is two-dimensional. Let b_0 be another vector of the null-space (say c_0 and b_0 are the two eigenvectors of $A^{\top} A$ corresponding to the two smallest eigenvalues), then the desired solution vector b is a linear combination of the two:

$$b = b_0 + \alpha c_0 \tag{16}$$

In order to find α given c_0 and b_0 we enforce the constraint that the matrix $[t'w''^T - t''w'^T]$ is of rank 2.

Clearly the choice of α will have no affect the first 6 elements of the vector b. Let us arrange the last 9 elements of b, b_0 and c_0 into the corresponding 3×3 matrices B, B_0 and C_0. We are now looking for an α such that:

$$Rank(B_0 - \alpha C_0) = 2 \tag{17}$$

Since C_0 in our case is the identity matrix, the solution for α is given by the eigenvalues of B_0. We choose the one with the smallest absolute value.

Based on the previous arguments the algorithm for finding the motion parameters is as follows:

1. Compute the $N \times 15$ matrix A from equation (9).
2. Find the eigenvector corresponding to the second smallest eigenvalue of the matrix $A^T A$. This is b_0. (The vector c_0 corresponds to the smallest.)
3. The first 6 elements of b_0 are the translations, t',t''.
4. Arrange the last 9 elements of b_0 into a $3x3$ matrix B_0.
5. Construct a rank 2 matrix B from B_0:
$$B = B_0 - \alpha I \tag{18}$$
6. Solve: $[t'w''^T - t''w'^T] = B$,
for w' and w'', given t' and t'' from step (3).

5. Implementation details

5.1. Iterative refinement and coarse to fine processing

The constant brightness constraint is a linearized form of the Sum Square Difference (SSD) criteria. The linear solution can be thought of as a single iteration of Newton's method applied to the problem. Iterative refinement is performed as follows: First one calculates motion and depth using eqns. 9,12,13. Then, using the depth and motion, images 1 and 2 are warped towards image 0. A correction to the depth and motion is computed using the warped images. In the ideal case, as the final result the warped images should appear nearly identical to image 0.

One must be careful not to compute the incremental model from the previous iteration because as the iterations proceed the system of equations of the incremental model will become badly conditioned. Thus, one must compute the full model (previous iteration plus the incremental change to the model). This is described below.

Let Ψ_0, Ψ_1 and Ψ_2 be the three image. Assume we have $\hat{k}$, $\hat{t}^j$, $\hat{w}^j$, from the previous iteration. The image motions $\hat{u}^j$ and $\hat{v}^j$ can be computed using $\hat{K}$, $\hat{t}^j$ and $\hat{w}^j$ in equation (8). These are then used to warp images Ψ_1 to $\hat{\Psi}_1$ and Ψ_2 to $\hat{\Psi}_2$. After warping, the images satisfy the brightness constraint equation:

$$\begin{aligned} I_x du' + I_y dv' + \hat{I}'_t = 0 \\ I_x du'' + I_y dv'' + \hat{I}''_t = 0 \end{aligned} \tag{19}$$

where the temporal derivatives at each pixel are given by:

$$\begin{aligned} \hat{I}'_t = \hat{\Psi}_1 - \Psi_0 \\ \hat{I}''_t = \hat{\Psi}_2 - \Psi_0 \end{aligned} \tag{20}$$

and du^j, dv^j are the (still unknown) differences between computed image motions and the real image motions:

$$\begin{aligned} du^j &= u^j - \hat{u}^j \\ dv^j &= v^j - \hat{v}^j \end{aligned} \tag{21}$$

Let:

$$\alpha^j = I_x \hat{u}^j + I_y \hat{v}^j \tag{22}$$

which can also be written as:

$$\alpha^j = \hat{k} s^T \hat{t}^j + v^T \hat{w}^j \tag{23}$$

Substituting equations (22) and (21) in equation (19) we get:

$$\begin{aligned} I_x u + I_y v + (\hat{I}'_t - \alpha') = 0 \\ I_x u + I_y v + (\hat{I}''_t - \alpha'') = 0 \end{aligned} \tag{24}$$

Substituting equation (8) in equation (24) we get modified versions of the equations (12) and (13)

$$\begin{aligned} k s^T t' + v^T w' + (I'_t - \alpha') = 0 \\ k s^T t'' + v^T w'' + (I''_t - \alpha'') = 0 \end{aligned} \tag{25}$$

We start our first iteration with $\hat{k}$, $\hat{t}^j$, $\hat{w}^j$ all zero and therefore $\alpha = 0$ as well.

In order to deal with image motions larger than 1 pixel we use a Gaussian pyramid for coarse to fine processing [1][2].

5.2. Computing the depth, smoothing and interpolation.

After recovering the camera motions we use equations (12) and (13) to compute depth at every point where $s^T t'$ or $s^T t''$ are non zero. To combine the information from both images and to interpolate over areas where image gradients are small we used Local Weighted Regression.

Equation (26) shows the cost function used to compute the depth at a given point:

$$\min_K \arg \sum_{x,y \in R} \sum_j \beta(x,y) |s^T t^j|^p \left(k s^T t^j + v^T w^j + I_t^j \right)^2 \tag{26}$$

The sum is over a region R and over the two motions $j = 1, \ldots, 2$. The windowing function $\beta(x, y)$ allows one to increase the weight of the closer points. We used a function created by convolving two box filters together. It is a crude approximation to a Gaussian. The $|S^T t^j|^p$ term reduces the weight of points which have a small gradient or where the gradient is perpendicular to that camera motion since these cases are highly affected by noise. We used $p = 2$. The size of the region R depends on the amount of smoothing and interpolation required. During the iteration process we used a region of 5×5 or 9×9. In order to get 'prettier' results, after the last iteration, we reduced the region R to 3×3 but added a multigrid membrane interpolation scheme.

6. Experiments with real images

6.1. Experimental procedure

The images were taken with a Pulnix TM9701 $\frac{1}{2}inch$ CCD video camera and an $8.5mm$ lens (focal length of 630 $pixels$). Image capture was performed using an SGI Indy Workstation at $640 \times 480 pixel$ resolution.

In the two image sequences the vertical motion was $10mm$ and the horizontal motion $5mm$. In Fig. (1) we created a scene with few strong feature points and edges. We also added rotation around the Y axis in conjunction with the horizontal motion. The depths in (1) range from $40cm$ to $17cm$. For pure translation image motions were on average $15pixels$. With $4.0degrees$ of rotation around the Y axis the horizontal pixels motion averaged $50pixels$.

In Fig. (3) we created a scene which, although it has strong edge features, none of the point features correspond to true 3D features and the horizontal lines correspond to curved lines in 3D. The depths in (1) range from $62cm$ to $31cm$ and image motions were on average $9pixels$. The diagonal bar is balanced on the head a bit closer to the camera on image left. The front of the right hand cylinder was at the same depth as the tip of the nose.

6.2. Results

The images were processed using 5 levels of coarse to fine processing with 4 iterations at each level. Varying the number of iterations from 1 through 4 had no qualitative impact but using a single iteration caused a small change in the resulting motion estimates. A 9×9 region was used for the local constant depth fit at all levels.

Fig. (1e) shows the recovered depth map for the first sequence. Fig. (1f) shows a 3D rendering of the surface. Image (1a) was used as the texture map. The Z values have been scaled by 18000. The rendering uses an orthographic projection which creates a slight distortion in the images.

In order to get smoother and more visually pleasing results we performed some smoothing using a membrane interpolation model used for the final stage (figures 2a, 2b). The recovered translation estimates were qualitatively correct even when rotation around the vertical (Y) axis was added to the horizontal motion.

Fig. (2c) shows the recovered rotation estimates for true rotations ranging from $-4.0degrees$ to $0.8degrees$. Outside this range the program did not converge. Negative rotation angles induced image motion that was in the same direction as the image motion induced by the translation. There seems to be a bad scaling error in the rotation estimates. Fig. (3d) shows the recovered depth map for the second sequence. The results are qualitatively correct. In Fig. (3f) the texture mapping was removed for clarity.

7. Summary

We have derived a general relationship between the spatio-temporal derivatives of three frames and the ego-motion parameters of the two motions. This relationship was derived first for the general case where the ego-motion model comprises of the 27 coefficients of the trilinear tensor, and then for the small-motion model of Longuett-Higgins & Prazdny.

On practical grounds, these relationships are model-based brightness constraints which provide a linear constraint per pixel in the image — thereby providing a direct structure and motion estimation that cuts through the aperture problem and without prior detection of feature points.

The implementation details of these model-based brightness constraints are important and include three critical elements: (i) embedding of the computations within a coarse-to-fine (Gaussian Pyramid) framework, (ii) Newton iterations over the brightness constraint equation, and (iii) post-processing of smoothness and surface interpolation for obtaining visually pleasing results.

The algorithm was tested on a set of challenging real image situations which contain very little "good features" (local regions with significant variability of gradient direction) which are necessary for optical flow and discrete point matching algorithms. Yet, we obtained a faithful dense depth map of the scene due to the fact that the algorithm is not hindered by the presence of aperture effects.

References

[1] J.R. Bergen, P. Anandan, K.J. Hanna and R. Hingorani. Hierarchical model-based motion estimation. In *Proceedings EECV*, Santa Margherita Ligure, Italy, June (1992)

[2] P.J. Burt and E.H. Adelson. The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, 31:532-540, (1983)

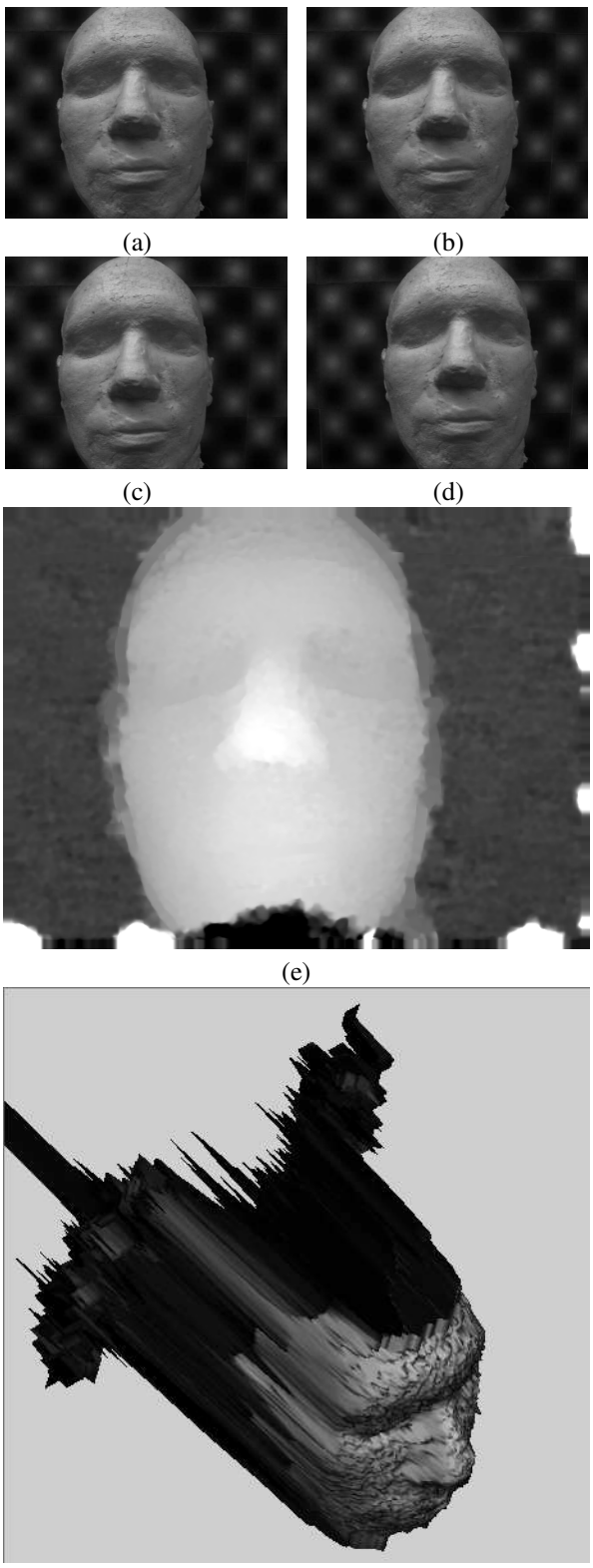

Figure 1. *The three input images (a,b,c) with pure translation. (d) another image taken at the same location as (b) with $-4.0degrees$ rotation. The estimated depth map (e) and 3D rendering of the surface (f). Uses 9×9 region and a local constant depth model.*

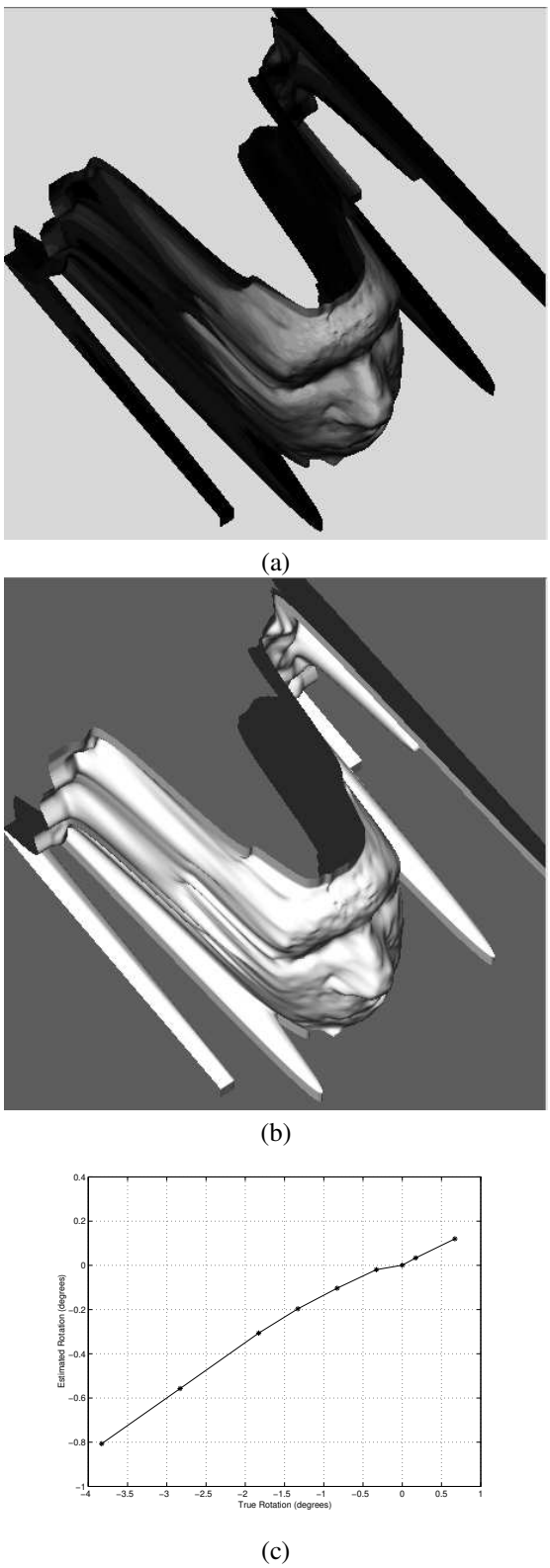

Figure 2. *(a)(b)3D rendering of the estimated surface (scaled in Z) with interpolation for smoothing. (c) Estimated rotation as a function of true rotation. Although there is the correct linear relationship there also appears to be a significant scale error.*

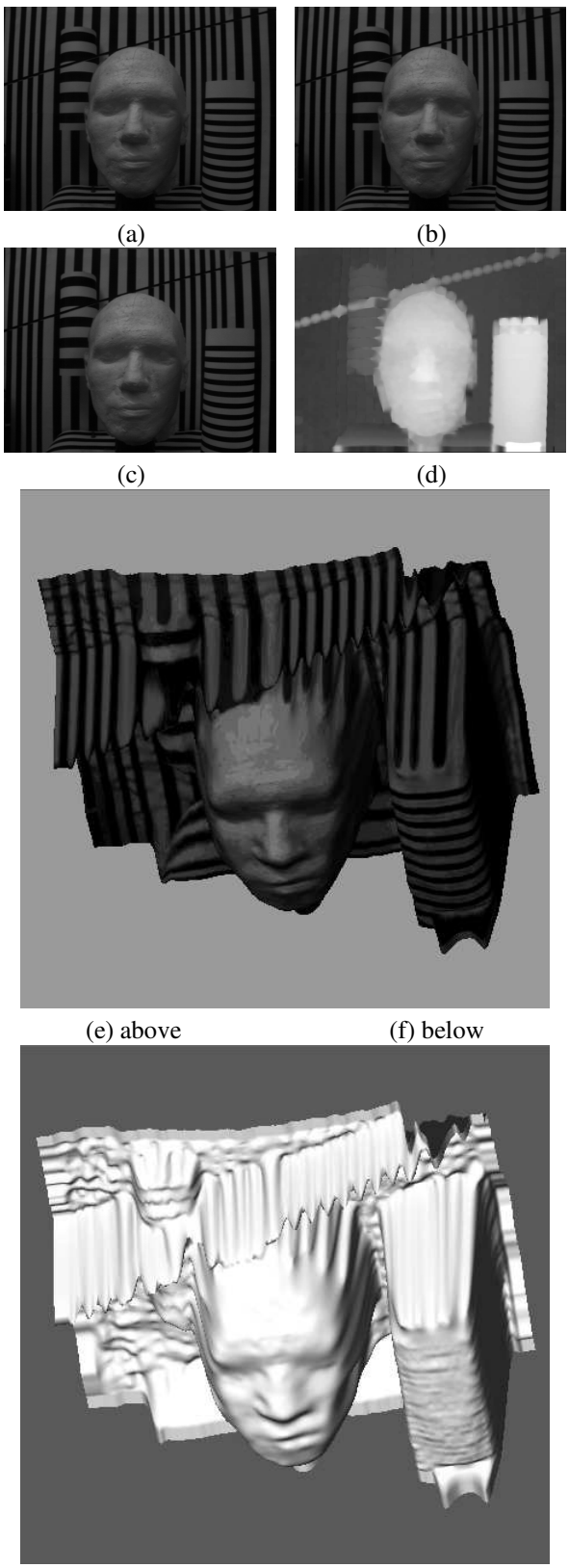

Figure 3. *The three input images (a,b,c) and the estimated depth map (d). 3D rendering of the estimated surface (scaled in Z) with interpolation for smoothing (e)(f).*

[3] O. Faugeras and B. Mourrain. On the geometry and algebra of the point and line correspondences between N images. In *Proceedings ICCV*, 951-956, Cambridge, MA, June (1995)

[4] C. Fermuller and Y. Aloimonos. Global rigidity constraints in image displacement fields. In *Proceedings ICCV*, 245-250, Cambridge, MA, June (1995)

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

[6] J. Heel. Direct Estimation of Structure and Motion from Multiple Frames. AI memo 1190, March (1990)

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

[8] B.K.P. Horn and B.G. Schunk. Determining optical flow. *Artificial Intelligence*, 17:185-203 (1981)

[9] B.K.P. Horn and E.J. Weldon. Direct methods for recovering motion *IJCV* 2:51-76, (1988)

[10] H.C. Longuett-Higgins and K. Prazdny. The interpretation of a moving retinal image. *Proceedings of the Royal Society of London B*, 208:385-397,(1980)

[11] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings IJCAI*, 674-679, Vancouver, (1981)

[12] S. Negahdaripour and B.K.P. Horn. Direct passive navigation. *PAMI*, 9(1):168-176,(1987)

[13] A. Shashua. Algebraic functions for recognition. *PAMI*, 17(8):779–789, 1995.

[14] A. Shashua and S. Avidan. The rank 4 constraint in multiple view geometry. In *Proceedings ECCV*, Cambridge, UK, April 1996

[15] A. Shashua and P. Anandan. Trilinear Constraints Revisited: Generalized Trilinear Constraints and the Tensor Brightness Constraint. *Proceedings ARPA IU Workshop*, February 1996, Palm Springs, CA.

[16] A. Shashua and K.J. Hanna. The Tensor Brightness Constraints: Direct Estimation of Motion Revisited. Technion Technical Report, Haifa, Israel, November (1995)

[17] A. Shashua and M. Werman. Trilinearity of Three Perspective Views and its Associated Tensor. In *Proceedings of ICCV 95*, 920-925 Boston, MA, USA, June (1995)

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

[19] B. Triggs. Matching constraints and the Joint Image. In *Proceedings ICCV*, 338-343, Cambridge, MA, June (1995)