Geometric Method for Projective Reconstruction of Shape and Motion Using n Uncalibrated Cameras

Eduardo Bayro-Corrochano and Vladimir Banarer

Christian Albrechts University,
Computer Science Institute,
Preußerstraße 1-9, 24105 Kiel, Germany,
edb,vlb@ks.informatik.uni-kiel.de

Abstract

The paper focus on the analysis and computing of the projective structure and motion using geometric invariants. This work relates current approaches in the geometric algebra framework as a result the approach gains geometric transparency and elegance. The papers presents experiments regarding projective reconstruction of shape and motion using both simulated and real images.

1 Introduction

In this paper we present a geometric approach for the computation of shape and motion using invariant theory in the geometric algebra framework. In the last years researchers have developed methods to compute projective invariants using n uncalibrated cameras [4, 5, 3, 1]. Projective reconstruction has been done using the projective depth [7], the kinematic depth [8], projective invariants [5] and factorization methods [9, 6, 10]. Since the projective factorization methods require the scalar factor of projective depth, the use of projective invariants to compute these scalars can help to initialize the projective reconstruction of shape and motion. In this paper we present a method to compute the projective depth using projective invariants depending of the trifocal tensor. With these projective depth we can initialize the projective reconstruction of structure and motion. The paper presents experiments for projective reconstruction of shape and motion using both and simulated and real images.

2 Computing Projective Invariant of Points Using Two Uncalibrated Cameras

A 3D projective invariant can be formed from a set of six points as follows

$$Inv = \frac{[\mathbf{X}_1\mathbf{X}_2\mathbf{X}_3\mathbf{X}_4][\mathbf{X}_4\mathbf{X}_5\mathbf{X}_2\mathbf{X}_6]}{[\mathbf{X}_1\mathbf{X}_2\mathbf{X}_4\mathbf{X}_5][\mathbf{X}_3\mathbf{X}_4\mathbf{X}_2\mathbf{X}_6]}. \quad (1)$$

In [3] it is shown that the bracket of these 4 points (in R^4) can be equated as

$$S_{1234} = [\mathbf{X}_1\mathbf{X}_2\mathbf{X}_3\mathbf{X}_4] \equiv [\mathbf{A}_0\mathbf{B}_0\mathbf{A}'_{1234}\mathbf{B}'_{1234}]. \quad (2)$$

Expanding the bracket in equation (2) by expressing the intersection points in terms of the $\mathbf{A}$'s and $\mathbf{B}$'s ($\mathbf{A}'_i = \alpha_{ij}\mathbf{A}_j$ and $\mathbf{B}'_i = \beta_{ij}\mathbf{B}_j$) and defining a matrix $\tilde{\boldsymbol{F}}$ such that

$$\tilde{F}_{ij} = [\mathbf{A}_0\mathbf{B}_0\mathbf{A}_i\mathbf{B}_j] \quad (3)$$

and the vectors $\boldsymbol{\alpha}_{1234} = (\alpha_{1234,1}, \alpha_{1234,2}, \alpha_{1234,3})$ and $\boldsymbol{\beta}_{1234} = (\beta_{1234,1}, \beta_{1234,2}, \beta_{1234,3})$ we can write $S_{1234} = \boldsymbol{\alpha}^T{}_{1234}\tilde{\boldsymbol{F}}\boldsymbol{\beta}_{1234}$ [4]. The ratio

$$Inv = \frac{(\boldsymbol{\alpha}^T{}_{1234}\tilde{\boldsymbol{F}}\boldsymbol{\beta}_{1234})(\boldsymbol{\alpha}^T{}_{4526}\tilde{\boldsymbol{F}}\boldsymbol{\beta}_{4526})}{(\boldsymbol{\alpha}^T{}_{1245}\tilde{\boldsymbol{F}}\boldsymbol{\beta}_{1245})(\boldsymbol{\alpha}^T{}_{3426}\tilde{\boldsymbol{F}}\boldsymbol{\beta}_{3426})} \quad (4)$$

is therefore seen to be an invariant using two cameras. Note that equation (4) is invariant whatever values of the γ_4 components of the vectors $\mathbf{A}_i, \mathbf{B}_i, \mathbf{X}_i$ etc. are chosen. A confusion arises if we attempt to express the Inv of Eq. (4) in terms of what we actually observe, i.e. the 3D image coordinates and the fundamental matrix calculated from these image coordinates. In order to avoid that it is necessary to transfer the computations of Eq. (4) carried out in R^4 to 3D. Let us explain now this procedure.

If we define $\tilde{\boldsymbol{F}}$ by

$$\tilde{F}_{kl} = (\mathbf{A}_k \cdot \gamma_4)(\mathbf{B}_l \cdot \gamma_4)F_{kl}, \quad (5)$$

then it follows using the relationships $\alpha_{ij} = \frac{\mathbf{A}'_i\gamma_4}{\mathbf{A}_j\gamma_4}\delta_{ij}$ and $\beta_{ij} = \frac{\mathbf{B}'_i\gamma_4}{\mathbf{B}_j\gamma_4}\epsilon_{ij}$ that

$$\alpha_{ik}\tilde{F}_{kl}\beta_{il} = (\mathbf{A}'_i \cdot \gamma_4)(\mathbf{B}'_i \cdot \gamma_4)\delta_{ik}F_{kl}\epsilon_{il}. \quad (6)$$

0-7803-5655-1/99/$10.00 © 1999 IEEE

Proceedings of the First Workshop on Robot Motion and Control, RoMoCo '99 : J
RoMoCo <1, 1999, Kiekrz>
1999
0-7803-5655-1

According to the above, we can write the invariant as

$$Inv_2 = \frac{(\delta^T{}_{1234}F\epsilon_{1234})(\delta^T{}_{4526}F\epsilon_{4526})\phi_{1234}\phi_{4526}}{(\delta^T{}_{1245}F\epsilon_{1245})(\delta^T{}_{3426}F\epsilon_{3426})\phi_{1245}\phi_{3426}}, \quad (7)$$

where $\phi_{pqrs} = (\mathbf{A}'_{pqrs} \cdot \gamma_4)(\mathbf{B}'_{pqrs} \cdot \gamma_4)$. We see therefore that the ratio of the terms $\delta^T F\epsilon$ which resembles the expression for the invariant in R^4, but uses only the observed coordinates and the estimated fundamental matrix, will not be an invariant. Instead, we need to include the factors ϕ_{1234} etc., which do not cancel. It is relatively easy to show [3] that these factors can be formed as follows. Since a'_3, a'_4 and a'_{1234} are collinear we can write $a'_{1234} = \mu_{1234}a'_4 + (1-\mu_{1234})a'_3$. Then, by expressing $\mathbf{A}'_{1234}$ as the intersection of the line joining $\mathbf{A}'_1$ and $\mathbf{A}'_2$ with the plane through $\mathbf{A}_0$, $\mathbf{A}'_3$, $\mathbf{A}'_4$ we can use the projective split and equate terms to give

$$\frac{(\mathbf{A}'_{1234} \cdot \gamma_4)(\mathbf{A}'_{4526} \cdot \gamma_4)}{(\mathbf{A}'_{3426} \cdot \gamma_4)(\mathbf{A}'_{1245} \cdot \gamma_4)} = \frac{\mu_{1245}(\mu_{3426} - 1)}{\mu_{4526}(\mu_{1234} - 1)}. \quad (8)$$

The values of μ are readily obtainable from the images. The factors $\mathbf{B}'_{pqrs} \cdot \gamma_4$ are found in a similar way so that if $b'_{1234} = \lambda_{1234}b'_4 + (1 - \lambda_{1234})b'_3$ etc., the overall expression for the invariant becomes

$$I_2 = \frac{(\delta^T{}_{1234}F\epsilon_{1234})(\delta^T{}_{4526}F\epsilon_{4526})}{(\delta^T{}_{1245}F\epsilon_{1245})(\delta^T{}_{3426}F\epsilon_{3426})} \frac{\mu_{1245}(\mu_{3426} - 1)}{\mu_{4526}(\mu_{1234} - 1)}$$

$$Inv_2 = I_2 \frac{\lambda_{1245}(\lambda_{3426} - 1)}{\lambda_{4526}(\lambda_{1234} - 1)}. \quad (9)$$

Concluding given the coordinates of a set of 6 corresponding points in the two image planes (where these 6 points are projections from arbitrary world points but with the assumption that they are not coplanar) we can form 3D projective invariants provided we have some estimate of F. See [1] for a more detailed discussion on this issue.

3 Projective Invariant of Points Using Three Un–calibrated Cameras

The technique used to form the 3D projective invariants for two views can be straightforwardly extended to give expressions for invariants of three views. Consider four world points, $\{\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3, \mathbf{X}_4\}$ (or two lines and $\mathbf{X}_3 \wedge \mathbf{X}_4 = (\mathbf{A}_0 \wedge L^A{}_{34}) \vee (\mathbf{C}_0 \wedge L^C{}_{34})$. Once again, we can combine the above expressions to give an equation for the 4-vector $\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_3 \wedge \mathbf{X}_4$:

$$\begin{aligned} &\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_3 \wedge \mathbf{X}_4 = \\ &= [(\mathbf{A}_0 \wedge L^A{}_{12}) \vee (\mathbf{B}_0 \wedge L^B{}_{12})] \wedge [(\mathbf{A}_0 \wedge L^A{}_{34}) \vee (\mathbf{C}_0 \wedge L^C{}_{34})] \\ &= (\mathbf{A}_0 \wedge \mathbf{A}_{1234}) \wedge [(\mathbf{B}_0 \wedge L^B{}_{12}) \vee (\mathbf{C}_0 \wedge L^C{}_{34})]. \end{aligned} \quad (10)$$

Writing the lines L^B_{12} and L^C_{34} in terms of the line coordinates we have $L^B_{12} = l^B_{12,j} L^B_j$ and $L^C_{34} = l^C_{34,j} L^C_j$. It has been shown in section two that the components of the trilinear tensor (which plays the role of the fundamental matrix for 3 views), can be written in geometric algebra as

$$T_{ijk} = (\mathbf{A}_0 \wedge \mathbf{A}_i) \wedge [(\mathbf{B}_0 \wedge L^B_j) \vee (\mathbf{C}_0 \wedge L^C_k)] \quad (11)$$

so that equation (10) reduces to

$$\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_3 \wedge \mathbf{X}_4 = T_{ijk}\alpha_{1234,i} l^B{}_{12,j} l^C{}_{34,k}. \quad (12)$$

The invariant Inv_3 can then be expressed as

$$Inv_3 = \frac{(T_{ijk}\alpha_{1234,i} l^B{}_{12,j} l^C{}_{34,k})(T_{mnp}\alpha_{4526,m} l^B{}_{26,n} l^C{}_{45,p})}{(T_{qrs}\alpha_{1245,q} l^B{}_{12,r} l^C{}_{45,s})(T_{tuv}\alpha_{3426,t} l^B{}_{26,u} l^C{}_{34,v}} \quad (13)$$

noting that the factoring must be done so that the same line factorizations occur in both the numerator and denominator – as discussed in section 2. We therefore have an expression for invariants in three views which is a direct extension of the invariants for 2 views. When we form the above invariant from *observed* quantities we note, as before, that some correction factors will be necessary – equation (13) is given above in terms of R^4 quantities. Fortunately this is quite straightforward. Regarding the results of section 2 we can simply consider the $\alpha's$ terms in equation (13) as not observable quantities, conversely the line terms like $l^B_{12,j} l^C{}_{34,k}$ are indeed observed quantities. As a result, the expression has to be modified using partially the coefficients computed in section 3 and for the unique four combinations of three cameras their invariant equations read

$$I_{ABC} = \frac{(T^{ABC}_{ijk}\alpha_{1234,i} l^B{}_{12,j} l^C{}_{34,k})(T^{ABC}_{mnp}\alpha_{4526,m} l^B{}_{26,n} l^C{}_{45,p})}{(T^{ABC}_{qrs}\alpha_{1245,q} l^B{}_{12,r} l^C{}_{45,s})(T^{ABC}_{tuv}\alpha_{3426,t} l^B{}_{26,u} l^C{}_{34,v})}$$

$$Inv_{ABC} = I_{ABC} \frac{\mu_{1245}(\mu_{3426} - 1)}{\mu_{4526}(\mu_{1234} - 1)}, \quad (14)$$

similar expressions for I_{ABD}, I_{CAD}, I_{BCD}. We noticed that first two have the same scalar coefficient

Extensive simulations with Maple confirmed that the use of this kind of coefficients in the four invariants is fully correct.

4 Camera Self–localization

Using the invariant theory approach we can determine the changes of the 3–D coordinates of a moving uncalibrated camera. According three invariants we can determine the coordinates of a 3–D point [5]. We select as a projective basis five fixed points in the 3–D space $\boldsymbol{X}_1$, $\boldsymbol{X}_2$, $\boldsymbol{X}_3$, $\boldsymbol{X}_4$, $\boldsymbol{X}_5$ and consider the unknown point $\boldsymbol{X}_6$ as the optical center of the moving camera, see Figure 4.

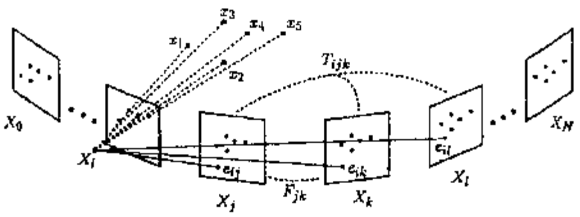

Figure 1: Computing the center of views of a moving camera

We can then compute the moving optical centers using two cameras

$$I_x^F = \frac{X_6}{W_6} = \frac{(\boldsymbol{\delta}^T{}_{2346}\boldsymbol{F}\boldsymbol{\epsilon}_{2346})(\boldsymbol{\delta}^T{}_{1235}\boldsymbol{F}\boldsymbol{\epsilon}_{1235})}{(\boldsymbol{\delta}^T{}_{2345}\boldsymbol{F}\boldsymbol{\epsilon}_{2345})(\boldsymbol{\delta}^T{}_{1236}\boldsymbol{F}\boldsymbol{\epsilon}_{1236})} \cdot \frac{\mu_{2345}\lambda_{2345}\mu_{1236}\lambda_{1236}}{\lambda_{2346}\lambda_{2346}\lambda_{1235}\lambda_{1235}} \quad (15)$$

or using three cameras

$$I_x^T = \frac{X_6}{W_6} = \frac{(T_{ijk}^{ABC}\alpha_{2346,i}l^B{}_{23,j}l^C{}_{46,k})}{(T_{qrs}^{ABC}\alpha_{2345,q}l^B{}_{23,r}l^C{}_{45,s})} \cdot \frac{(T_{mnp}^{ABC}\alpha_{1236,m}l^B{}_{12,n}l^C{}_{36,p})}{(T_{tuv}^{ABC}\alpha_{1236,t}l^B{}_{12,u}l^C{}_{36,v})}\frac{\mu_{2345}\mu_{1236}}{\mu_{2346}\mu_{1235}}. \quad (16)$$

Similarly permuting the six points we compute I_y^F, I_y^T and I_z^F, I_z^T. The compensating coefficients for the invariants I_y and I_z vary due to the permuted points.

positions of a moving camera. These curves show that

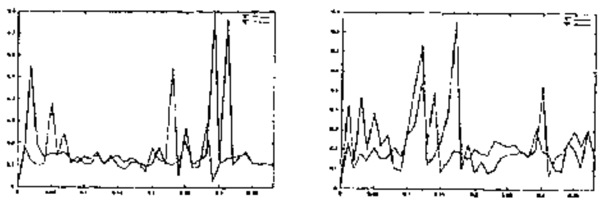

Figure 2: Performance of the computing of any two center of views using F and T

the trinocular computation renders more accurate results as the binocular case. The Euclidean coordinates of the optical centers are gained applying the transformation which relates the projective basis to its a priori known Euclidean basis.

5 Projective Depth

In a geometric sense the projective depth can be seen as the relation between the distance regarding the view center of a 3–D point $\boldsymbol{X}_i$ and the focal distance f. We can derive the projective depth from a projective mapping. According to the pinhole model this projective mapping in a matrix representation reads

$$\lambda \boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix} \quad (17)$$
$$= \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{R}_{3x3} & \boldsymbol{t}_{3x1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ 1 \end{bmatrix},$$

where we call λ a projective scale factor. Note that the projective mapping is further expressed in terms of a f, rotation and translation components. Let us attach the world coordinates at the view center of the camera, the resultant projective mapping becomes

$$\lambda_j x = \begin{bmatrix} 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_3 \\ 1 \end{bmatrix} = PX. \quad (18)$$

We can then compute straightforward

$$\lambda = X_3. \quad (19)$$

Using this result we can say that the projective depth α fulfills the following relation

$$\alpha f = \lambda = X_3. \quad (20)$$

The way how we compute the projective depth α of a 3–D point appears simple using invariant theory. For that we select a basis system taking four 3–D points in general position $\boldsymbol{X}_1$, $\boldsymbol{X}_2$, $\boldsymbol{X}_3$, $\boldsymbol{X}_5$, as the four point $\boldsymbol{X}_4$ the optical center of camera at the new position, and as unknown 3–D point the point $\boldsymbol{X}_6$. This is depicted in Figure 3.

For that we select as projective basis in P^3 points in general position $\boldsymbol{X}_1$, $\boldsymbol{X}_2$, $\boldsymbol{X}_3$, $\boldsymbol{X}_5$ as $\boldsymbol{X}_4$ the view center of the moving camera and as the point to be reconstructed $\boldsymbol{X}_6$. Since we use the mapped points, we consider as the four point the epipole or mapping of the current view center and the mapped sixth point as the point with unknown depth. The other mapped basis points remain constant during the procedure.

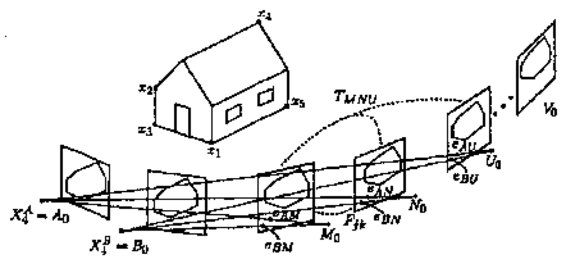

Figure 3: Computing the projective depths of n cameras

The tensor based invariant expression for computing the third coordinate or projective depth of a point

$$\cdot \frac{(T^{ABC}_{mnp} \alpha_{2416,m} l^B{}_{24,n} l^C{}_{16,p})}{(T^{ABC}_{tuv} \alpha_{2415,t} l^B{}_{24,u} l^C{}_{15,v})} \frac{\mu_{2316}\mu_{2415}}{\mu_{2315}\mu_{2416}}. \quad (21)$$

In this way we can successively compute the projective depths λ_{ij} of the j–points referred to the i–camera. The λ_{ij} will be used in next section for the 3–D reconstruction using the join image concept and the SVD method.

Since this kind of invariant can be also expressed in terms of the quadrifocal tensor [2], we can compute the projective depth based on four cameras.

6 Shape and Motion

The orthographic and paraperspective factorization method for structure and motion using the affine camera model was developed by Tomasi, Kanade and Poelman [9, 6]. This method works for cameras viewing small and distance scenes, thus all scale factors of projective depth λ_{ij}=1. For the case of perspective images the scale factors λ_{ij} are unknown. According Triggs [10] all λ_{ij} satisfy a set of consistency reconstruction equations of the so called *join image* and they can be computed using the epipolar constraint.

In the previous section we presented a procedure for the computing of λ_{ij} using an invariant based on the trifocal tensor. Since this kind of invariant can be also expressed using the quadrifocal tensor [2] we could also compute the projective depths via an invariant involving the quadrifocal tensor.

6.1 The join image

The joint image $\mathcal{J}$ is nothing else as the intersections of optical rays and planes at the points or lines in the 3D projective space. The interrelated geometry can be linearly expressed by the fundamental tensor, trifocal and quadrifocal tensors.

In order to take into account the interrelated geometry, the projective reconstruction procedure should

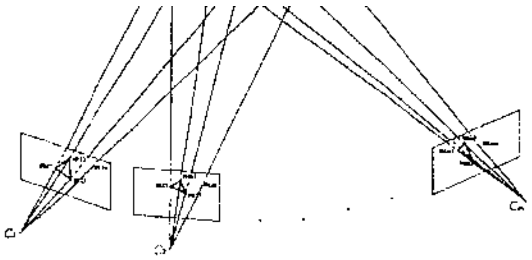

Figure 4: The geometry of the join image

put together all the data of the individual images in a geometrically coherent manner. The way to do that is considering the observations of the points $\boldsymbol{X}_j$ regarding each i–camera

$$\lambda_{ij}\boldsymbol{x}_j = P_i\boldsymbol{X}_j \tag{22}$$

as the i–row of a matrix of rank 4. For m cameras and n points the 3m×n matrix $\mathcal{J}$ of the joint image is given by

$$\mathcal{J} = \begin{pmatrix} \lambda_{11}\boldsymbol{x}_{11} & \lambda_{12}\boldsymbol{x}_{12} & \dots & \lambda_{1n}\boldsymbol{x}_{1n} \\ \lambda_{21}\boldsymbol{x}_{21} & \lambda_{22}\boldsymbol{x}_{22} & \dots & \lambda_{2n}\boldsymbol{x}_{2n} \\ \lambda_{31}\boldsymbol{x}_{31} & \lambda_{32}\boldsymbol{x}_{32} & \dots & \lambda_{3n}\boldsymbol{x}_{3n} \\ . & . & \dots & . \\ . & . & \dots & . \\ . & . & \dots & . \\ \lambda_{m1}\boldsymbol{x}_{m1} & \lambda_{m2}\boldsymbol{x}_{m2} & \dots & \lambda_{mn}\boldsymbol{x}_{mn}. \end{pmatrix} \tag{23}$$

For the affine reconstruction procedure the matrix is of rank 3. The matrix $\mathcal{J}$ of the joint image is amenable to a singular value decomposition for finding the shape and motion.

6.2 The SVD method

The application of SVD to $\mathcal{J}$ gives

$$\mathcal{J}_{3m\times n} = U_{3m\times r}S_{r\times r}V^T_{n\times r}, \tag{24}$$

where the columns of matrix $V^T_{n\times r}$ and $U_{3m\times r}$ are orthonormal base for the input (co-kernel) and output (range) spaces of $\mathcal{J}$. In order to get a decomposition

$$\mathcal{J}_{3m\times n} = (U_{3m\times r}S^{\frac{1}{2}}_{r\times r})(S^{\frac{1}{2}}_{r\times r}V^T_{n\times r}) =$$
$$(P_1^T P_2^T P_3^T \dots P_m^T)^T_{3m\times 4}(\boldsymbol{X}_1\boldsymbol{X}_2\boldsymbol{X}_3\dots\boldsymbol{X}_n)_{4\times n}. \tag{25}$$

This way to divide $S_{r\times r}$ is not unique. Since the rank of $\mathcal{J}$ is 4 we should take for $S_{r\times r}$ the first four biggest singular values. The matrices P_i correspond to the projective mappings or "motion" from the projective space to the individual images and the point structure or "shape" is given by the $\boldsymbol{X}_i$. We test our approach using a simulations program written in Maple. Using the method of section 5 firstly we computed the projective depth of the points of a wire house observed with 9 cameras and then using the SVD projective reconstruction method we gained the shape and motion. The reconstructed house after the Euclidean readjustment for the presentation is shown in Figure 6.

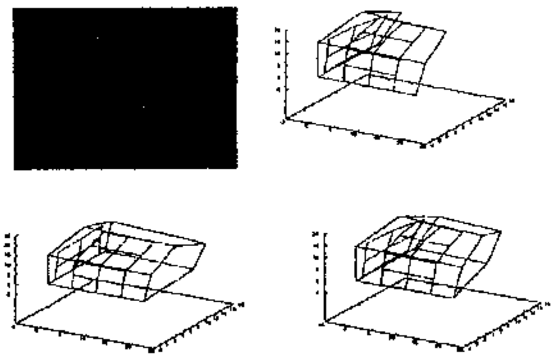

Figure 5: a) One of the three images, b) reconstructed incomplete house using 3 images c) extending the join image d) completing in the 3–D space

We notice that the reconstruction keeps quite well the original form of the model. The next section will show how using geometric expressions in terms of the operators of algebra of incidence $\vee$ (meet) and $\bigwedge$ (join) and particular tensor based invariants we can improve the shape of the reconstructed model.

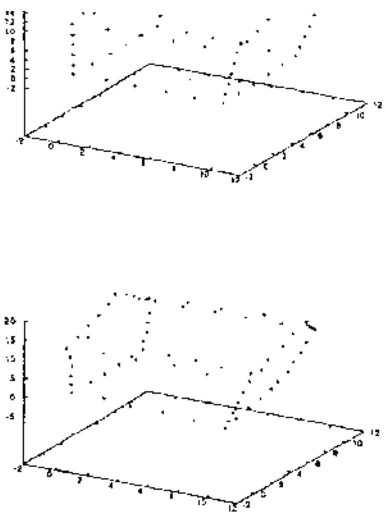

Figure 6: Reconstructed house using a) noise-free observations and b) noisy observations

6.3 Completion of the 3–D shape using geometric invariants

The projective structure can be improved in two ways: completing points on the images, expanding the join image and and then call the SVD procedure or after the reconstruction complete points like occluded points in the 3D space. Both approaches can use on the one hand geometric inference rules based on symmetries or concrete knowledge about the scene. Using three real views of a similar model house with its most right lower corner missing, see Figure 7.b, we compute in each image the virtual image point of this 3–D point. Then we reconstruct the scene as shown in Figure 7.c. As opposite using geometric incidence operations we completed the house employing the space points as depicted in Figure 7.d. We can see that creating points in the images yields a better reconstruction of the occluded point. Note that in the reconstructed image we transformed the projective shape to an Euclidean one for the presentation of the re-

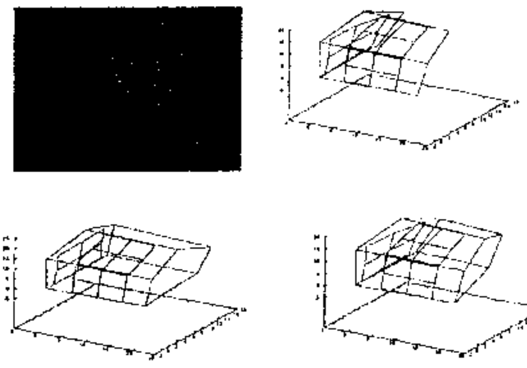

Figure 7: a) One of the three images, b) reconstructed incomplete house using 3 images c) extending the join image d) completing in the 3–D space

Similarly we proceeded using 9 images, as presented in in Figure 8.a–d. We can see that the resulting reconstructed point is almost similar in both procedures. As a result we can draw the following conclusion: when we have few views we should extend the join image using virtual image points and in the case of several images we should extend the point structure in the 3–D space.

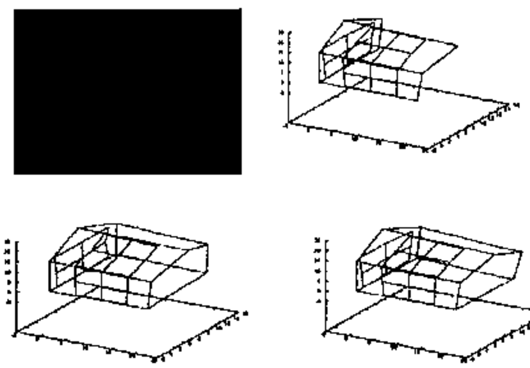

Figure 8: a) One of the nine images, b) reconstructed incomplete house using 9 images c) extending the join image d) completing in the 3–D space

variants based on the trifocal tensor. We developed a method to compute the projective depth using this kind of invariants. With these projective depth we can initialize the projective reconstruction of structure and motion. The papers presents experiments regarding projective reconstruction of shape and motion using both simulated and real images. This work relates current approaches in the geometric algebra framework, as a result the approach gains in geometrically transparency and elegance. However the authors believe that more work have to be done in order to improve the computational algorithms so that the use of projective invariants will be more and more attractive for real time systems with noisy data.

Acknowledgments Eduardo Bayro Corrochano was supported by the project SO-201 of the Deutsche Forschungsgemeinschaft.

References

[1] Bayro-Corrochano, E. and Lasenby, J. 1998. Geometric techniques for the computation of projective invariants using n uncalibrated cameras. In *Proceedings of the Indian Conference on Computer Vision and Image Processing*, New delhi, India, 21–23 December, pp. 95–100.

[2] Bayro-Corrochano, E. and Lasenby, J. 1999. Analysis and the computation of projective invariants using n uncalibrated cameras. Submitted to the Journal of the Royal Society.

[3] Lasenby, J., Bayro-Corrochano, E., A.N. Lasenby and Sommer, G. 1996. A new methodology for computing invariants in computer vision. *Proceedings of the International Conference on Pattern Recognition (ICPR'96), Vienna, August 1996.*, Vol.I, pp334-338.

[4] Carlsson, S. 1994. The Double Algebra: and effective tool for computing invariants in computer vision. *Applications of Invariance in Computer Vision*, Lecture Notes in Computer Science 825; Proceedings of 2nd-joint Europe-US workshop, Azores, October

[illegible] dimensional project invariants from a pair of images using the Grassmann–Cayley algebra. *Journal of Image and Vision Computing*, 16, pp. 3-12.

[6] Poelman C.J. and Kanade T. 1994. A paraperspective factorization method for shape and motion recovery. In J–O. Eklundh, editor, *European Conference on Computer Vision*, Stockholm, pp. 97–108.

[7] Shashua, A. 1994. Projective structure from uncalibrated images: structure from motion and recognition PAMI, 16(8), 778:790, August.

[8] Sparr Gunnar. 1994. Kinetic depth. In J–O. Eklundh, editor, *European Conference on Computer Vision*, Stockholm.

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

[10] Triggs, W. 1995. Matching Constraints and the Joint Image. In proceedings of *ICCV'95*, MIT.