Journal of Mathematical Imaging and Vision 16: 131–154, 2002

A Geometric Approach for the Theory and Applications of 3D Projective Invariants

EDUARDO BAYRO-CORROCHANO
Centro de Investigación y de Estudios Avanzados, Cinvestav, Apartado Postal 31-438, Plaza la Luna Guadalajara, Jal. 44550, Mexico
edb@gdl.cinvestav.mx

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

Abstract. A central task of computer vision is to automatically recognize objects in real-world scenes. The parameters defining image and object spaces can vary due to lighting conditions, camera calibration and viewing position. It is therefore desirable to look for geometric properties of the object which remain invariant under such changes in the observation parameters. The study of such geometric invariance is a field of active research. This paper presents the theory and computation of projective invariants formed from points and lines using the geometric algebra framework. This work shows that geometric algebra is a very elegant language for expressing projective invariants using n views. The paper compares projective invariants involving two and three cameras using simulated and real images. Illustrations of the application of such projective invariants in visual guided grasping, camera self-localization and reconstruction of shape and motion complement the experimental part.

Keywords: computer vision, invariants, Clifford (geometric) algebra, projective geometry, 3D projective invariants, visual guided robotics

1. Introduction

The concept of invariance has been of interest in many areas of computer vision. The scope of *geometric invariance* was captured in the volume [25]. Invariance has been widely used for object recognition, matching and reconstruction [24, 28]. Indeed, the currently fashionable topic of camera self-calibration can be cast in terms of looking for entities which are invariant under the class of similitudes. Thus, the study of invariants remains one of fundamental interest in computer vision. In this paper we will outline the use of geometric algebra (GA) in establishing a framework in which invariants can be derived and calculated. An important point to note here is that the *same* framework and approach can be used for extensions such as differential invariants, Lie algebra approaches, etc., although only projective invariants from points in 1, 2 and 3D will be discussed here.

Geometric algebra is a coordinate-free approach to geometry based on the algebras of Grassmann [10] and Clifford [7]. The algebra is defined on a space whose elements are called *multivectors*; a multivector is a linear combination of objects of different type, e.g. scalars and vectors. It has an associative and fully invertible product called the *geometric* or Clifford product. The existence of such a product and the calculus associated with the geometric algebra give the system tremendous power. Some preliminary applications of geometric algebra in the field of computer vision have already been given. [2, 16, 19], and here we would like to extend the discussion of geometric invariance

given in [3, 17, 20, 21]. Geometric algebra provides a very natural language for projective geometry and has all the necessary equipment for the tasks which the Grassmann-Cayley algebra is currently used for. The Grassmann-Cayley or double algebra [1, 5] is as system for computations with subspaces of finite-dimensional vector spaces. While it expresses the ideas of projective geometry, such as the meet and join, very elegantly, it lacks an inner (regressive) product (and indeed often goes to great lengths to 'create' an inner product) and some other key concepts which we will discuss later.

The paper explains the geometric meaning of invariants in terms of areas and volumes, and using the projective split, it relates the projective invariants of the 3D projective space to the invariants of the image plane. This is the proper way to show that what is measured is not what it is projected; the paper clarifies this issue. In our opinion, this matter was not satisfactorily explained by Carlsson and Csurka [5, 9]. Another contribution of the paper is to extend the computation of the projective invariants for the cases of three and four uncalibrated cameras. This has been done so far only using two views [5, 9]. The use of the projective split is crucial in this kind of computation and it is one essential difference of the geometric, algebra approach to the approaches based on Grassmann-Cayley or double algebras [5, 9]. Unfortunately, using Grassmann-Cayley or double algebras it is not possible to resort to the projective split for various promising applications. Another contribution of the paper is the extension of the computation of the projective depth done by Triggs [31] who used only two cameras; in this paper we present a method of using invariants involving three uncalibrated cameras. The geometric interpretation of the projective invariants as invariant relation of areas or volumes appears very useful for tasks of object manipulation or robot navigation where we can compute the invariants using three uncalibrated cameras. In this regard the paper extends the work of Rotwel et al [28], and Colios and Trahanias [8], who presented nicely how invariants using monocular vision can be helpful for 3D object recognition, manipulation and robot navigation. The same work can be done using our approach, with the difference that now we can use invariants involving three or four cameras. This is a much robuster method, as our experimental part in Section 8 shows: the efficiency of the projective invariants increases when more camera views are involved.

The organization of the papers is as follows: section two presents a brief introduction in geometric algebra, section three outlines the projective geometry and the protective split and section four algebra of incidence. Section five explains the geometry of two and three uncalibrated views. Sections six and seven are devoted to the derivation of projective invariants using one, two and three cameras respectively. Section eight compares experimentally these projective invariants using simulated and real images. Section nine presents the applications: visual guided grasping and camera self-localization. Section ten shows the computing of projective depth and section eleven the computation of 3D shape and motion. Finally, section twelve presents the conclusions. In this paper vectors will be bold quantities (except for basis vectors) and multivectors will not be bold. Lower case is used to denote vectors in 3D Euclidean space and upper case to denote vectors in 4D projective space.

2. Geometric Algebra: An Outline

The algebras of Clifford and Grassmann are well known to pure mathematicians, but were long ago abandoned by physicists in favour of the vector algebra of Gibbs—which is still most commonly used today. The approach to Clifford algebra we adopt here was pioneered in the 1960's by David Hestenes who has, since then, worked on developing his version of Clifford algebra—which will be referred to as *geometric algebra* (GA)—into a unifying language for mathematics and physics [14].

2.1. *The Geometric Product and Multivectors*

Let $\mathcal{G}_n$ denote the geometric algebra of n-dimensions—this is a graded linear space. As well as vector addition and scalar multiplication we have a non-commutative product which is associative and distributive over addition—this is the geometric or Clifford product. A further distinguishing feature of the algebra is that any vector squares to give a scalar. The geometric product of two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ is written $\boldsymbol{ab}$ and can be expressed as a sum of its symmetric and antisymmetric parts

$$\boldsymbol{ab} = \boldsymbol{a} \cdot \boldsymbol{b} + \boldsymbol{a} \wedge \boldsymbol{b}, \tag{1}$$

where the inner product $\boldsymbol{a} \cdot \boldsymbol{b}$ and the outer product $\boldsymbol{a} \wedge \boldsymbol{b}$ are defined by

$$\boldsymbol{a} \cdot \boldsymbol{b} = \frac{1}{2}(\boldsymbol{ab} + \boldsymbol{ba}) \qquad \boldsymbol{a} \wedge \boldsymbol{b} = \frac{1}{2}(\boldsymbol{ab} - \boldsymbol{ba}). \tag{2}$$

Since the associative geometric algebra is defined by the anti-commutative bilinear form given by Eq. (1), it comprises of symmetric and antisymmetric algebras. The Grassmann-Cayley algebras are only antisymmetric algebras of signature zero, as a result they do lack of spinors, which are very useful, for example, to compute of homograpries in 2D and 3D. Geometric algebra has many algebraic advantages for other tasks beyond the projective geometry. Grassmann-Cayley algebra, double algebra or bracket algebra have not the inner (regressive) product, thus they can no handle projection to subspaces. In contrast, in geometric algebra using the projective split we can reduce the computational complexity by projecting to lower dimension spaces spanned by vector-, bivector or trivector basis.

The inner product of two vectors is the standard *scalar* or *dot* product and produces a scalar. The outer or wedge product of two vectors is a new quantity we call a *bivector*. We think of a bivector as a directed area in the plane containing $\boldsymbol{a}$ and $\boldsymbol{b}$, formed by sweeping $\boldsymbol{a}$ along $\boldsymbol{b}$—see Fig. 1a. Thus, $\boldsymbol{b} \wedge \boldsymbol{a}$ will have the opposite orientation making the wedge product anti-commutative as given in Eq. (2). The outer product is immediately generalizable to higher dimensions—for example, $(\boldsymbol{a} \wedge \boldsymbol{b}) \wedge \boldsymbol{c}$, a *trivector*, is interpreted as the oriented volume formed by sweeping the area $\boldsymbol{a} \wedge \boldsymbol{b}$ along vector $\boldsymbol{c}$—see Fig. 1b. The outer product of k vectors is a k-vector or k-blade, and such a quantity is said to have *grade* k. A *multivector* is made up of a linear combination of objects of different grade, i.e. scalar plus bivector etc. GA provides a means of manipulating multivectors which allows us to keep track of different grade objects simultaneously—much as one does with complex number operations. For a general multivector X, the notation $\langle X \rangle$ will mean *take the scalar part of* X. The highest grade element in a space is called the *pseudoscalar*. The unit pseudoscalar is denoted by I and is crucial when discussing duality.

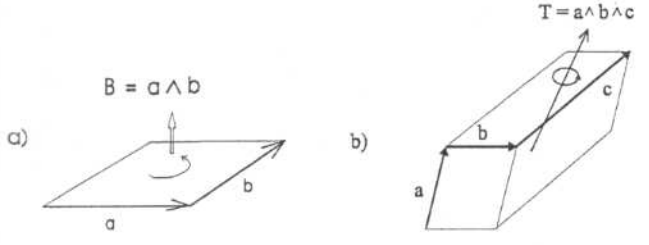


Figure 1. (a) The directed area, or bivector, $\boldsymbol{a} \wedge \boldsymbol{b}$. (b) The oriented volume, or trivector, $\boldsymbol{a} \wedge \boldsymbol{b} \wedge \boldsymbol{c}$.

We now end this introductory section by giving a very brief review of the geometric algebra approach to linear algebra. A more detailed review is found in [14].

Consider a linear function f which maps vectors to vectors in the same space. We can extend f to act linearly on multivectors via the *outermorphism*, $\underline{f}$, defining the action of $\underline{f}$ on blades by

$$\begin{aligned} &\underline{f}(\boldsymbol{a}_1 \wedge \boldsymbol{a}_2 \wedge \cdots \wedge \boldsymbol{a}_r) \\ &\quad = f(\boldsymbol{a}_1) \wedge f(\boldsymbol{a}_2) \wedge \cdots \wedge f(\boldsymbol{a}_r). \end{aligned} \tag{3}$$

We use the term outermorphism because $\underline{f}$ preserves the grade of any r-vector it acts on. We therefore know that the pseudoscalar of the space must be mapped onto some multiple of itself. The scale factor in this mapping is the *determinant* of $\underline{f}$;

$$\underline{f}(I) = \det(\underline{f})I. \tag{4}$$

This is much simpler than many definitions of the determinant enabling one to establish most properties of determinants with little effort.

2.2. *Geometric Algebras for the Image Plane and Projective Space*

For the modeling of the image plane, we use the geometric algebra of the 3D Euclidean space $\mathcal{G}_{3,0,0}$, which has the standard *Euclidean signature* and is generated by $2^3 = 8$ multivector elements given by:

$$\underbrace{1}_{\text{scalar}}, \underbrace{\{\sigma_1, \sigma_2, \sigma_3\}}_{\text{vectors}}, \underbrace{\{\sigma_1\sigma_2, \sigma_2\sigma_3, \sigma_3\sigma_1\}}_{\text{bivectors}}, \underbrace{\{\sigma_1\sigma_2\sigma_3\} \equiv I}_{\text{trivector}}. \tag{5}$$

Here, bivectors can be interpreted as oriented areas and trivectors as oriented voumes. Note that we will not use bold for these basis vectors. The highest grade element is a trivector called the unit *pseudoscalar*. It can easily be verified that the pseudoscalar $I = \sigma_1\sigma_2\sigma_3$ squares to -1 and commutes with all multivectors (a multivector is a general linear combination of any of the elements in the algebra) in the 3D space. In a three-dimensionsal space we can construct a trivector $\boldsymbol{a} \wedge \boldsymbol{b} \wedge \boldsymbol{c}$, but no 4-vectors exist, since there is no possibility of sweeping the volume element $\boldsymbol{a} \wedge \boldsymbol{b} \wedge \boldsymbol{c}$ over a fourth dimension.

If we choose to map between projective space and 3D Euclidean space via the projective split, we are forced to use the 4D geometric algebra $\mathcal{G}_{1,3,0}$ for $\mathcal{P}^3$ with Lorentzian metric.

The $\mathcal{G}_{1,3,0}$ geometric algebra has as its vector basis $\gamma_1, \gamma_2, \gamma_3, \gamma_4$, where $\gamma_4^2 = +1$ and $\gamma_k^2 = -1$ for $k = 1, 2, 3$. This then generates the following multivector basis:

$$\underbrace{1}_{\text{scalar}}, \underbrace{\gamma_k}_{4\ \text{vectors}}, \underbrace{\gamma_2\gamma_3, \gamma_3\gamma_1, \gamma_1\gamma_2, \gamma_4\gamma_1, \gamma_4\gamma_2, \gamma_4\gamma_3}_{6\ \text{bivectors}},$$
$$\underbrace{I\gamma_k}_{4\ \text{trivectors}}, \underbrace{I}_{\text{pseudoscalar}}. \tag{6}$$

The pseudoscalar is $I = \gamma_1\gamma_2\gamma_3\gamma_4$, with

$$I^2 = (\gamma_1\gamma_2\gamma_3\gamma_4)(\gamma_1\gamma_2\gamma_3\gamma_4) = -(\gamma_3\gamma_4)(\gamma_3\gamma_4) = -1. \tag{7}$$

Note that we use the same symbol for pseudoscalars of different geometric algebras, because it is understood that the pseudoscalar changes its dimension and signature for $\mathcal{G}_{1,3,0}$ and I^3 for $\mathcal{G}_{1,3,0}$.

The fourth basis vector, γ_4, can also be seen as a selected direction for the *projective split* [17] operation in 4D. We will see shortly that by carrying out the geometric product via γ_4, we can associate bivectors of our 4D space with vectors of our 3D space. The role and use of the projective split operation will be treated in more detail in next sections.

3. Projective Geometry and the Projective Split

Since about the mid 1980's most of the computer vision literature discussing geometry and invariants has used the language of projective geometry (see appendix of [25]). As any point on a ray from the optical centre of a camera will map to the same point in the camera image plane it is easy to see why a 2D view of a 3D world might well be best expressed in projective space. In classical projective geometry one defines a 3D space, $\mathcal{P}^3$, whose points are in $1-1$ correspondence with lines through the origin in a 4D space, R^4. Similarly, k-dimensional subspaces of $\mathcal{P}^3$ are identified with $(k+1)$-dimensional subspaces of R^4. Such projective views can provide very elegant descriptions of the geometry of incidence (intersections, unions etc.). The projective space, $\mathcal{P}^3$, has no metric, the basis and metric are introduced in the associated 4D space. In this 4D space a coordinate description of a projective point is conventionally brought about by using *homogeneous coordinates*. Here we will briefly outline how projective geometry looks in the GA framework.

The basic projective geometry operations of meet and join are easily expressible in terms of standard operations within the geometric algebra. Firstly, to introduce the concepts of duality which are so important in projective geometry, we define the dual A^* of an r-vector A as

$$A^* = AI^{-1}. \tag{8}$$

In an n-dimensional geometric algebra one can define the *join* $J = A \wedge B$ of an r-vector, A, and an s-vector, B, by

$$J = A \wedge B \quad \text{if } A \text{ and } B \text{ are linearly independent.} \tag{9}$$

If A and B are not linearly independent the join is not given simply by the wedge but by the subspace that they span. J can be interpreted as a *common dividend of lowest grade* and is defined up to a scale factor. It is easy to see that if $(r+s) \geq n$ then J will be the pseudoscalar for the space. In what follows we will use $\wedge$ for the join only when the blades A and B are not linearly independent, otherwise we will use the ordinary exterior product, $\wedge$.

If A and B have a common factor (i.e. there exists a k-vector C such that $A = A'C$ and $B = B'C$ for some A', B') then we can define the 'intersection' or *meet* of A and B as $A \vee B$ where [15]

$$(A \vee B)^* = A^* \wedge B^*. \tag{10}$$

That is, the dual of the meet is given by the join of the duals (a familiar result from classical projective geometry). The dual of $(A \vee B)$ is understood to be taken with respect to the *join* of A and B. In most cases of practical interest this join will be the whole space and the meet is therefore easily computed. A more useful expression for the meet is as follows

$$A \vee B = (A^* \wedge B^*)I = (A^* \wedge B^*)(I^{-1}I)I = (A^* \cdot B) \tag{11}$$

We therefore have the very simple and readily computed relation of $A \vee B = (A^* \cdot B)$. The above concepts are discussed further in [15].

Points in real 3D space will be represented by vectors in $\mathcal{E}^3$, a 3D space with a Euclidean metric. As mentioned earlier, we find it useful to associate a point in $\mathcal{E}^3$ with a line in a 4D space, R^4. In these two distinct but related spaces we define basis vectors: $(\gamma_1, \gamma_2, \gamma_3, \gamma_4)$

in R^4 and $(\sigma_1, \sigma_2, \sigma_3)$ in $\mathcal{E}^3$. We identify R^4 and $\mathcal{E}^3$ with the GAs of 4 and 3 dimensions, $\mathcal{G}_{(1,3,0)}$ and $\mathcal{G}_{(3,0,0)}$ (here $\mathcal{G}_{(p,q,r)}$ is a $p+q+r$-dimensional GA in which p, q and r basis vectors square to $+1$, -1 and 0 respectively). We require that vectors, bivectors and trivectors in R^4 will represent points, lines and planes in $\mathcal{E}^3$. Suppose we choose γ_4 as a selected direction in R^4, we can then define a mapping which associates the bivectors $\gamma_i\gamma_4$, $i = 1, 2, 3$, in R^4 with the vectors σ_i, $i = 1, 2, 3$, in $\mathcal{E}^3$;

$$\sigma_1 \equiv \gamma_1\gamma_4, \qquad \sigma_2 \equiv \gamma_2\gamma_4, \qquad \sigma_3 \equiv \gamma_3\gamma_4. \tag{12}$$

To preserve the Euclidean structure of the spatial vectors $\{\sigma_i\}$ (i.e. $\sigma_i^2 = +1$) we are forced to assume a non-Euclidean metric for the basis vectors in R^4. We choose to use $\gamma_4^2 = +1$, $\gamma_i = -1$, $i = 1, 2, 3$. This process of associating the higher and lower dimensional spaces is an application of what Hestenes calls the *projective split*.

For a vector $\mathbf{X} = X_1\gamma_1 + X_2\gamma_2 + X_3\gamma_3 + X_4\gamma_4$ in R^4 the projective split is obtained by taking the geometric product of $\mathbf{X}$ and γ_4;

$$\begin{aligned}\mathbf{X}\gamma_4 &= \mathbf{X}\cdot\gamma_4 + \mathbf{X}\wedge\gamma_4 = X_4\left(1 + \frac{\mathbf{X}\wedge\gamma_4}{X_4}\right)\\ &\equiv X_4(1+\boldsymbol{x}).\end{aligned} \tag{13}$$

Note that $\boldsymbol{x}$ contains terms of the form $\gamma_1\gamma_4$, $\gamma_2\gamma_4$, $\gamma_3\gamma_4$ or, via Eq. (12), terms in $\sigma_1, \sigma_2, \sigma_3$. We therefore associated the vector $\boldsymbol{x}$ in $\mathcal{E}^3$ with the bivector $\mathbf{X}\wedge\gamma_4/X_4$ in R^4.

If we start with a vector $\boldsymbol{x} = x_1\sigma_1 + x_2\sigma_2 + x_3\sigma_3$ in $\mathcal{E}^3$, we can represent this in R^4 by the vector $\mathbf{X} = X_1\gamma_1 + X_2\gamma_2 + X_3\gamma_3 + X_4\gamma_4$ such that

$$\begin{aligned}\boldsymbol{x} &= \frac{\mathbf{X}\wedge\gamma_4}{X_4} = \frac{X_1}{X_4}\gamma_1\gamma_4 + \frac{X_2}{X_4}\gamma_2\gamma_4 + \frac{X_3}{X_4}\gamma_3\gamma_4\\ &= \frac{X_1}{X_4}\sigma_1 + \frac{X_2}{X_4}\sigma_2 + \frac{X_3}{X_4}\sigma_3,\end{aligned} \tag{14}$$

$\Rightarrow x_1 = \frac{X_1}{X_4}$, for $i = 1, 2, 3$. This process can therefore be seen to be equivalent to using *homogeneous coordinates*, $\mathbf{X}$, for $\boldsymbol{x}$. Thus, in this GA formulation we postulate distinct spaces in which we represent ordinary 3D quantities and their 4D projective counterparts, together with a well-defined way of moving between these spaces.

4. Algebra in Projective Space

Having introduced duality, defined the operations of meet and join, and given the geometric approach to linear algebra, we are now ready to carry out geometric computations using the algebra of incidence.

Consider three non-collinear points, P_1, P_2, P_3, represented by vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3$ in $\mathcal{E}^3$ and by vectors $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3$ in R^4. The line L_{12} joining points P_1 and P_2 can be expressed in R^4 by the bivector

$$L_{12} = \mathbf{X}_1 \wedge \mathbf{X}_2. \tag{15}$$

Any point P, represented in R^4 by $\mathbf{X}$, on the line through P_1 and P_2, will satisfy the equation

$$\mathbf{X} \wedge L_{12} = \mathbf{X} \wedge \mathbf{X}_1 \wedge \mathbf{X}_2 = 0. \tag{16}$$

This is therefore the equation of the line in R^4. In general, such an equation is telling us that $\mathbf{X}$ belongs to the subspace spanned by $\mathbf{X}_1$ and $\mathbf{X}_2$—that is, that

$$\mathbf{X} = \alpha_1\mathbf{X}_1 + \alpha_2\mathbf{X}_2 \tag{17}$$

for some α_1, α_2. In computer vision we can use this equation as a geometric constraint to test whether a point $\mathbf{X}$ lies on L_{12}.

The plane Φ_{123} passing through points P_1, P_2, P_3 is expressed by the following trivector in R^4:

$$\Phi_{123} = \mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_3. \tag{18}$$

In 3D space there are generally three types of intersections we wish to consider: the intersection of a line and a plane, a plane and a plane, and a line and a line. To compute these intersections, we will make use of the following general formula [14], which gives the inner product of an r-blade, $A_r = \boldsymbol{a}_1 \wedge \boldsymbol{a}_2 \wedge \cdots \wedge \boldsymbol{a}_r$, and an s-blade, $B_s = \boldsymbol{b}_1 \wedge \boldsymbol{b}_2 \wedge \cdots \wedge \boldsymbol{b}_s$ (for $s \le r$):

$$\begin{aligned}&B_s \cdot (\boldsymbol{a}_1 \wedge \boldsymbol{a}_2 \wedge \cdots \wedge \boldsymbol{a}_r)\\ &\quad = \sum_{\boldsymbol{j}} \epsilon(j_1 j_2 \ldots j_r) B_s \cdot (\boldsymbol{a}_{j_1} \wedge \boldsymbol{a}_{j_2} \wedge \cdots \wedge \boldsymbol{a}_{j_s})\\ &\qquad \boldsymbol{a}_{j_s+1} \wedge \cdots \wedge \boldsymbol{a}_{j_r}\end{aligned} \tag{19}$$

In the equation, we sum over all the combinations $\boldsymbol{j} = (j_1, j_2, \ldots, j_r)$ such that no two j_k's are the same. If j is an even permutation of $(1, 2, 3, \ldots, r)$, then the expression $\epsilon(j_1 j_2 \ldots j_r) = +1$, and it is an odd permutation if $\epsilon(j_1 j_2 \ldots j_r) = -1$.

4.1. Intersection of a Line and a Plane

In the space R^4, consider the line $A = \mathbf{X}_1 \wedge \mathbf{X}_2$ intersecting the plane $\Phi = \mathbf{Y}_1 \wedge \mathbf{Y}_2 \wedge \mathbf{Y}_3$. We can compute the intersection point using a *meet* operation, as follows:

$$\begin{aligned} A \cap \Phi &= (\mathbf{X}_1 \wedge \mathbf{X}_2) \cap (\mathbf{Y}_1 \wedge \mathbf{Y}_2 \wedge \mathbf{Y}_3) \\ &= A \cap \Phi = A^* \cdot \Phi. \end{aligned} \tag{20}$$

Here, we have used Eq. (11), and we note that in this case the join covers the entire space.

Note also that the pseudoscalar I_4 in $\mathcal{G}_{1,3,0}$ for R^4 squares to -1, that it commutes with bivectors but anticommutes with vectors and trivectors, and that its inverse is given by $I_4^{-1} = -I_4$. Therefore, we can claim that

$$A^* \cdot \Phi = (AI^{-1}) \cdot \Phi = -(AI) \cdot \Phi. \tag{21}$$

Now, using Eq. (20), we can expand the meet, such that

$$\begin{aligned} A \cap \Phi &= -(AI) \cdot (\mathbf{Y}_1 \wedge \mathbf{Y}_2 \wedge \mathbf{Y}_3) \\ &= -\{(AI) \cdot (\mathbf{Y}_2 \wedge \mathbf{Y}_3)\}\mathbf{Y}_1 \\ &\quad + \{(AI) \cdot (\mathbf{Y}_3 \wedge \mathbf{Y}_1)\}\mathbf{Y}_2 \\ &\quad + \{(AI) \cdot (\mathbf{Y}_1 \wedge \mathbf{Y}_2)\}\mathbf{Y}_3. \end{aligned} \tag{22}$$

Noting that $(AI) \cdot (\mathbf{Y}_i \wedge \mathbf{Y}_j)$ is a scalar, we can evaluate Eq. (22) by taking scalar parts. For example, $(AI) \cdot (\mathbf{Y}_2 \wedge \mathbf{Y}_3) = \langle I(\mathbf{X}_1 \wedge \mathbf{X}_2)(\mathbf{Y}_2 \wedge \mathbf{Y}_3)\rangle = I(\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{Y}_2 \wedge \mathbf{Y}_3)$. From the definition of the bracket given earlier, we can see that if $P = \mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{Y}_2 \wedge \mathbf{Y}_3$, then $[P] = (\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{Y}_2 \wedge \mathbf{Y}_3)I_4^{-1}$. If we therefore write $[\mathbf{A}_1\, \mathbf{A}_2\, \mathbf{A}_3\, \mathbf{A}_4]$ as a shorthand for the magnitude of the pseudoscalar formed from the four vectors, then we can readily see that the meet reduces to

$$\begin{aligned} A \cap \Phi &= [\mathbf{X}_1\mathbf{X}_2\mathbf{Y}_2\mathbf{Y}_3]\mathbf{Y}_1 + [\mathbf{X}_1\mathbf{X}_2\mathbf{Y}_3\mathbf{Y}_1]\mathbf{Y}_2 \\ &\quad + [\mathbf{X}_1\mathbf{X}_2\mathbf{Y}_1\mathbf{Y}_2]\mathbf{Y}_3, \end{aligned} \tag{23}$$

thus giving the intersection point (vector in R^4).

4.2. Intersection of Two Planes

The *line of intersection of two planes*, $\Phi_1 = \mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_3$ and $\Phi_2 = \mathbf{Y}_1 \wedge \mathbf{Y}_2 \wedge \mathbf{Y}_3$, can be computed via the meet of Φ_1 and Φ_2:

$$\Phi_1 \cap \Phi_2 = (\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_3) \cap (\mathbf{Y}_1 \wedge \mathbf{Y}_2 \wedge \mathbf{Y}_3). \tag{24}$$

As in the previous section, this expression can be expanded as

$$\begin{aligned} \Phi_1 \cap \Phi_2 &= \Phi_1^* \cdot (\mathbf{Y}_1 \wedge \mathbf{Y}_2 \wedge \mathbf{Y}_3) \\ &= -\{(\Phi_1 I) \cdot \mathbf{Y}_1\}(\mathbf{Y}_2 \wedge \mathbf{Y}_3) \\ &\quad + \{(\Phi_1 I) \cdot \mathbf{Y}_2\}(\mathbf{Y}_3 \wedge \mathbf{Y}_1) \\ &\quad + \{(\Phi_1 I) \cdot \mathbf{Y}_3\}(\mathbf{Y}_1 \wedge \mathbf{Y}_2). \end{aligned}$$

Once again, the join covers the entire space and so the dual is easily formed. Following the arguments of the previous section, we can show that $(\Phi_1 I) \cdot \mathbf{Y}_i \equiv -[\mathbf{X}_1\mathbf{X}_2\mathbf{X}_3\mathbf{Y}_i]$, so that the meet is

$$\begin{aligned} \Phi_1 \cap \Phi_2 &= [\mathbf{X}_1\mathbf{X}_2\mathbf{X}_3\mathbf{Y}_1](\mathbf{Y}_2 \wedge \mathbf{Y}_3) \\ &\quad + [\mathbf{X}_1\mathbf{X}_2\mathbf{X}_3\mathbf{Y}_2](\mathbf{Y}_3 \wedge \mathbf{Y}_1) \\ &\quad + [\mathbf{X}_1\mathbf{X}_2\mathbf{X}_3\mathbf{Y}_3](\mathbf{Y}_1 \wedge \mathbf{Y}_2), \end{aligned} \tag{25}$$

thus producing a line of intersection or bivector in R^4.

4.3. Intersection of Two Lines

Two lines will intersect only if they are coplanar. This means that their representations in R^4, $A = \mathbf{X}_1 \wedge \mathbf{X}_2$, and $B = \mathbf{Y}_1 \wedge \mathbf{Y}_2$, will satisfy the equation

$$A \wedge B = 0. \tag{26}$$

This fact suggests that the computation of the intersection should be carried out in the 2D Euclidean space which has an associated 3D projective counterpart R^3. In this plane, the intersection point is given by

$$\begin{aligned} A \cap B &= A^* \cdot B = -(AI_3) \cdot (\mathbf{Y}_1 \wedge \mathbf{Y}_2) \\ &= -\{((AI_3) \cdot \mathbf{Y}_1)\mathbf{Y}_2 - ((AI_3) \cdot \mathbf{Y}_2)\mathbf{Y}_1\}, \end{aligned} \tag{27}$$

where I_3 is the pseudoscalar for R^3. Once again, we evaluate $((AI_3) \cdot \mathbf{Y}_i)$ by taking scalar parts:

$$\begin{aligned} (AI_3) \cdot \mathbf{Y}_i &= \langle \mathbf{X}_1\mathbf{X}_2I_3\mathbf{Y}_i\rangle = I_3\mathbf{X}_1\mathbf{X}_2\mathbf{Y}_i \\ &= -[\mathbf{X}_1\mathbf{X}_2\mathbf{Y}_i]. \end{aligned} \tag{28}$$

The meet can therefore be written as

$$A \cap B = [\mathbf{X}_1\mathbf{X}_2\mathbf{Y}_1]\mathbf{Y}_2 - [\mathbf{X}_1\mathbf{X}_2\mathbf{Y}_2]\mathbf{Y}_1, \tag{29}$$

where the bracket $[\mathbf{A}_1\mathbf{A}_2\mathbf{A}_3]$ in R^3 is understood to mean $[\mathbf{A}_1 \wedge \mathbf{A}_2 \wedge \mathbf{A}_3]I_3^{-1}$. This equation is often an

impractical means of performing the intersection of two lines. See [4] for a complete treatment of the incidence relations between points, lines, and planes in the n-affine plane.

5. Visual Geometry of Uncalibrated Cameras

In this section we will analyze the constraints related to the geometry of uncalibrated cameras. For two and three views, the epipolar geometry is defined in terms of bilinear and trilinear constraints. Since the constraints are based on the coplanarity of lines, we will only be able to define relationships expressed by a single tensor for up to four cameras. For more than four cameras, the constraints are linear combinations of bilinearities, trilinearities, and quadrilinearities.

5.1. *Geometry of Two Views*

In this and subsequent sections we will work in projective space R^4, although a return to 3D Euclidean space will be necessary when we discuss invariants in terms of image coordinates; this will be done via the projective split. Figure 2 shows a world point $\mathbf{X}$ projecting onto points $\mathbf{A}'$ and $\mathbf{B}'$ in the two image planes ϕ_A and ϕ_B, respectively.

The so-called epipoles $\boldsymbol{E}_{AB}$ and $\boldsymbol{E}_{BA}$ correspond to the intersections of the line joining the optical centers with the image planes. Since the points $\mathbf{A}_0, \mathbf{B}_0, \mathbf{A}', \mathbf{B}'$ are coplanar, we can formulate the bilinear constraint

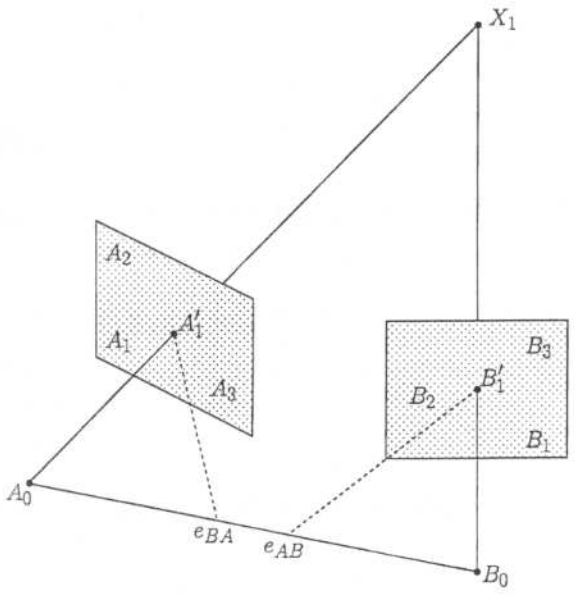


Figure 2. Sketch of binocular projection of a world point.

by taking advantage of the fact that the outer product of these four vectors must disappear. Thus,

$$\mathbf{A}_0 \wedge \mathbf{B}_0 \wedge \mathbf{A}' \wedge \mathbf{B}' = 0. \tag{30}$$

Now, if we let $\mathbf{A}' = \alpha_i \mathbf{A}_i$ and $\mathbf{B}' = \beta_j \mathbf{B}_j$, then equation (30) can be written as

$$\alpha_i \beta_j \{\mathbf{A}_0 \wedge \mathbf{B}_0 \wedge \mathbf{A}_i \wedge \mathbf{B}_j\} = 0. \tag{31}$$

Defining $\tilde{F}_{ij} = \{\mathbf{A}_0 \wedge \mathbf{B}_0 \wedge \mathbf{A}_i \wedge \mathbf{B}_j\} I^{-1} \equiv [\mathbf{A}_0 \mathbf{B}_0 \mathbf{A}_i \mathbf{B}_j]$ gives us

$$\tilde{F}_{ij} \alpha_i \beta_j = 0, \tag{32}$$

which corresponds in R^4 to the well-known relationship between the components of the *fundamental matrix* [23] or the *bilinear constraint* in E^3, F, and the image coordinates [23]. This suggests that $\tilde{F}$ can be seen as a linear function mapping two vectors onto a scalar:

$$\tilde{F}(\mathbf{A}, \mathbf{B}) = \{\mathbf{A}_0 \wedge \mathbf{B}_0 \wedge \mathbf{A} \wedge \mathbf{B}\} I^{-1}, \tag{33}$$

so that $\tilde{F}_{ij} = \tilde{F}(\mathbf{A}_i, \mathbf{B}_j)$. Note that viewing the fundamental matrix as a linear function means that we have a coordinate-independent description. Now, if we use the projective split to associate our point $\mathbf{A}' = \alpha_i \mathbf{A}_i$ in the image plane with its $\mathcal{E}^3$ representation $\boldsymbol{a}' = \delta_i \boldsymbol{a}_i$, where $\boldsymbol{a}_i = \frac{\mathbf{A}_i \wedge \gamma_4}{\mathbf{A}_i \cdot \gamma_4}$, it is not difficult to see that the coefficients are expressed as follows:

$$\alpha_i = \frac{\mathbf{A}' \cdot \gamma_4}{\mathbf{A}_i \cdot \gamma_4} \delta_i. \tag{34}$$

Thus, we are able to relate our 4D fundamental matrix $\tilde{F}$ to an *observed* fundamental matrix F in the following manner:

$$\tilde{F}_{kl} = (\mathbf{A}_k \cdot \gamma_4)(\mathbf{B}_l \cdot \gamma_4) F_{kl}, \tag{35}$$

so that

$$\alpha_k \tilde{F}_{kl} \beta_l = (\mathbf{A}' \cdot \gamma_4)(\mathbf{B}' \cdot \gamma_4) \delta_k F_{kl} \epsilon_l, \tag{36}$$

where $\boldsymbol{b}' = \epsilon_i \boldsymbol{b}_i$, with $\boldsymbol{b}_i = \frac{\mathbf{B}_i \wedge \gamma_4}{\mathbf{B}_i \cdot \gamma_4}$. F is the standard fundamental matrix that we would form from observations.

5.2. *Geometry of Three Views*

The so-called trilinear constraint captures the geometric relationships existing between points and lines in three camera views. Figure 3 shows three image planes ϕ_A, ϕ_B, and ϕ_C with bases $\{\mathbf{A}_i\}$, $\{\mathbf{B}_i\}$, and $\{\mathbf{C}_i\}$ and optical centers $\mathbf{A}_0$, $\mathbf{B}_0$, $\mathbf{C}_0$. Intersections of two world points $\mathbf{X}_i$ with the planes occur at points $\mathbf{A}'_i$, $\mathbf{B}'_i$, $\mathbf{C}'_i$, $i = 1, 2$. The line joining the world points is $L_{12} = \mathbf{X}_1 \wedge \mathbf{X}_2$, and the projected lines are denoted by L'_A, L'_B, and L'_C.

We first define three planes:

$$\Phi'_A = \mathbf{A}_0 \wedge \mathbf{A}'_1 \wedge \mathbf{A}'_2, \qquad \Phi'_B = \mathbf{B}_0 \wedge \mathbf{B}'_1 \wedge \mathbf{B}'_2,$$
$$\Phi'_C = \mathbf{C}_0 \wedge \mathbf{C}'_1 \wedge \mathbf{C}'_2. \tag{37}$$

It is clear that L_{12} can be formed by intersecting Φ'_B and Φ'_C:

$$L_{12} = \Phi'_B \cap \Phi'_C = (\mathbf{B}_0 \wedge L'_B) \cap (\mathbf{C}_0 \wedge L'_C). \tag{38}$$

If $L_{A_1} = \mathbf{A}_0 \wedge \mathbf{A}'_1$ and $L_{A_2} = \mathbf{A}_0 \wedge \mathbf{A}'_2$, then we can easily see that L_1 and L_2 intersect with L_{12} at $\mathbf{X}_1$ and $\mathbf{X}_2$, respectively. We therefore have

$$L_1 \wedge L_{12} = 0 \quad \text{and} \quad L_2 \wedge L_{12} = 0, \tag{39}$$

which can then be written as

$$(\mathbf{A}_0 \wedge \mathbf{A}'_i) \wedge \{(\mathbf{B}_0 \wedge L'_B) \cap (\mathbf{C}_0 \wedge L'_C)\} = 0 \quad \text{for } i = 1, 2. \tag{40}$$

This suggests that we should define a linear function T which maps a point and two lines onto a scalar as follows:

$$T(\mathbf{A}', L'_B, L'_C) = (\mathbf{A}_0 \wedge \mathbf{A}') \wedge \{(\mathbf{B}_0 \wedge L'_B) \cap (\mathbf{C}_0 \wedge L'_C)\}. \tag{41}$$

Now, using the line bases of the planes B and C, we can write

$$\mathbf{A}' = \alpha_i \mathbf{A}_i, \quad L'_B = l^B_j L^B_j \quad L'_C = l^C_k L^C_k. \tag{42}$$

If we define the components of a tensor as $T_{ijk} = T(\mathbf{A}_i, L^B_j, L^C_k)$, and if $\mathbf{A}'$, L'_B, and L'_C are all derived from projections of the same two world points then

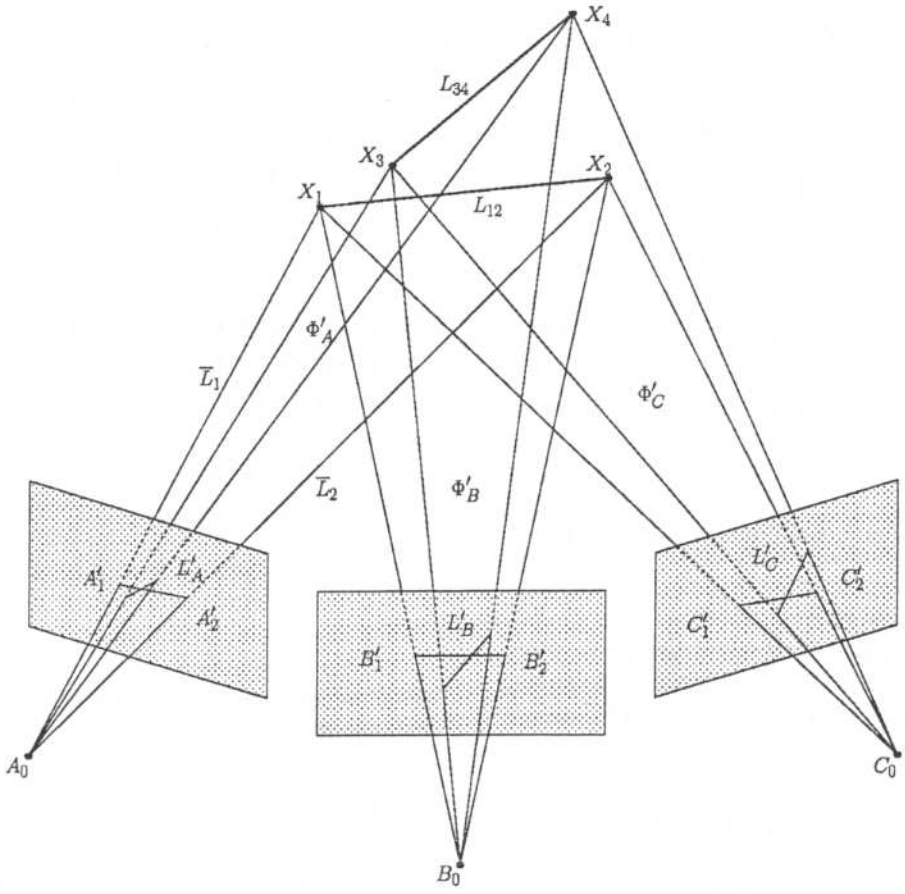


Figure 3. Model of the trinocular projection of the visual 3D space.

Eq. (40) tells us that we can write

$$T_{ijk}\alpha_i l_j^B l_k^C = 0. \tag{43}$$

T is the *trifocal tensor* [12, 29] and Eq. (43) is the *trilinear constraint*. In [11, 29] this constraint was arrived at by consideration of camera matrices; hare, however, Eq. (43) is arrived at from purely geometric considerations, namely, that two planes intersect in a line, which in turn intersects with another line. To see how we relate the three projected *lines*, we express the line in image plane ϕ_A joining $\mathbf{A}'_1$ and $\mathbf{A}'_2$ as the intersection of the plane joining $\mathbf{A}_0$ to the world line L_{12} with the image plane $\Phi_A = A_1 \wedge A_2 \wedge A_3$:

$$L'_A = \mathbf{A}'_1 \wedge \mathbf{A}'_2 = (\mathbf{A}_0 \wedge L_{12}) \cap \Phi_A. \tag{44}$$

Considering L_{12} as the meet of the planes $\Phi'_B \vee \Phi'_C$ and using the expansions of L'_A, L'_B, and L'_C given in Eq. (42), we can rewrite this equation as

$$\begin{aligned} l_i^A \boldsymbol{L}_i^A = \big((\boldsymbol{A}_0 \wedge \boldsymbol{A}_i) \wedge l_j^B l_k^C \big\{ \big(\mathbf{B}_0 \wedge L_j^B\big) \\ \cap \big(\mathbf{C}_0 \wedge L_k^C\big)\big\}\big) \cap \Phi_A. \end{aligned} \tag{45}$$

Using the expansion of the meet given in Eq. (25), we have

$$\begin{aligned} l_i^A L_i^A = \big[(\mathbf{A}_0 \wedge \mathbf{A}_i) \wedge l_j^B l_k^C \big\{ \big(\mathbf{B}_0 \wedge L_j^B\big) \\ \cap \big(\mathbf{C}_0 \wedge L_k^C\big)\big\}\big] L_i^A, \end{aligned} \tag{46}$$

which, when we equate coefficients, gives

$$l_i^A = T_{ijk} l_j^B l_k^C. \tag{47}$$

Thus, we obtain the familiar equation which relates the projected lines in the three views.

6. 3-D Projective Invariants from Multiple Views

This section presents the point and line projective invariants computable by means of n uncalibrated cameras. First we introduce the computation of the cross-ratio in 2D and 3D using the projective split. The use of the projective split, which is based in the inner product, is an advantage of the geometric algebra framework, this cannot be done using Grassmann-Cayley algebra or double algebra. We also explain how we can generate geometric invariants using the Plücker-Grassmann quadratic relation. We give a geometric interpretation of the cross-ratio in the 3-D space and in the image plane. Finally, we compute projective invariants using two and three cameras and applying the projective split.

6.1. 2D Generalization of the Cross-Ratio

When we consider points in a plane, we once move up to a space with one higher dimension, which we shall call R^3. Let a point P in the plane M be described by the vector $\boldsymbol{x}$ in $\mathcal{E}^2$, where $\boldsymbol{x} = x\sigma_1 + y\sigma_2$. In R^3 this point will be represented by $\mathbf{X} = X\gamma_1 + Y\gamma_2 + Z\gamma_3$, where $x = X/Z$ and $y = Y/Z$. We can define a general projective transformation via a linear function $\underline{f}_2$ by mapping vectors to vectors in R^3, such that

$$\begin{aligned} \underline{f}_2(\gamma_1) &= \alpha_1\gamma_1 + \alpha_2\gamma_2 + \tilde{\alpha}\gamma_3 \\ \underline{f}_2(\gamma_2) &= \beta_1\gamma_1 + \beta_2\gamma_2 + \tilde{\beta}\gamma_3 \\ \underline{f}_2(\gamma_3) &= \delta_1\gamma_1 + \delta_2\gamma_2 + \tilde{\delta}\gamma_3. \end{aligned} \tag{48}$$

Now, consider three vectors (representing non-collinear points) $\mathbf{X}_i$, $i = 1, 2, 3$, in R^3, and form the trivector

$$\mathcal{S}_2 = \mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_3 = \lambda_2 I_3, \tag{49}$$

where $I_3 = \gamma_1\gamma_2\gamma_3$ is the pseudoscalar for R^3. As before, under the projective transformation given by $\underline{f}_2$, $\mathcal{S}_2$ transforms to $\mathcal{S}'_2$, where

$$\mathcal{S}'_2 = \det \underline{f}_2 \mathcal{S}_2. \tag{50}$$

Therefore, the ratio of any trivector is invariant under $\underline{f}_2$. To project down into $\mathcal{E}^2$, assuming that $\mathbf{X}_i\gamma_3 = Z_i(1 + \boldsymbol{x}_i)$ under the projective split, we then write

$$\begin{aligned} \mathcal{S}_2 I_3^{-1} &= \langle \mathbf{X}_1\mathbf{X}_2\mathbf{X}_3 I_3^{-1} \rangle \\ &= \langle \mathbf{X}_1\gamma_3\gamma_3\mathbf{X}_2\mathbf{X}_3\gamma_3\gamma_3 I_3^{-1} \rangle \\ &= Z_1Z_2Z_3 \langle (1 + \boldsymbol{x}_1)(1 - \boldsymbol{x}_2)(1 + \boldsymbol{x}_3)\gamma_3 I_3^{-1} \rangle, \end{aligned} \tag{51}$$

where the $\boldsymbol{x}_i$ represent vectors in $\mathcal{E}^2$. We can only get a scalar term from the expression within the brackets by calculating the product of a vector, two spatial vectors, and I_3^{-1}, i.e.,

$$\begin{aligned} \mathcal{S}_2 I_3^{-1} &= Z_1Z_2Z_3 \langle (\boldsymbol{x}_1\boldsymbol{x}_3 - \boldsymbol{x}_1\boldsymbol{x}_2 - \boldsymbol{x}_2\boldsymbol{x}_3)\gamma_3 I_3^{-1} \rangle \\ &= Z_1Z_2Z_3 \{(\boldsymbol{x}_2 - \boldsymbol{x}_1) \wedge (\boldsymbol{x}_3 - \boldsymbol{x}_1)\} I_2^{-1}. \end{aligned} \tag{52}$$

It is therefore clear that we must use multiples of the ratios in our calculations, so that the arbitrary scalars Z_i cancel. In the case of four points in a plane, there are only four possible combinations of $Z_i Z_j Z_k$ and it is not possible to cancel all the Z's by multiplaying two ratios of the form $\mathbf{X}_i \wedge \mathbf{X}_j \wedge \mathbf{X}_k$ together. For five coplanar points $\{\mathbf{X}_i\}$, $i = 1, \ldots, 5$, however, there are several ways of achieving the desired cancellation. For example,

$$Inv_2 = \frac{(\mathbf{X}_5 \wedge \mathbf{X}_4 \wedge \mathbf{X}_3) I_3^{-1} (\mathbf{X}_5 \wedge \mathbf{X}_2 \wedge \mathbf{X}_1) I_3^{-1}}{(\mathbf{X}_5 \wedge \mathbf{X}_1 \wedge \mathbf{X}_3) I_3^{-1} (\mathbf{X}_5 \wedge \mathbf{X}_2 \wedge \mathbf{X}_4) I_3^{-1}}.$$

According to Eq. (52), we can interpret this ratio in $\mathcal{E}^2$ as

$$Inv_2 = \frac{(\boldsymbol{x}_5 - \boldsymbol{x}_4) \wedge (\boldsymbol{x}_5 - \boldsymbol{x}_3) I_2^{-1} (\boldsymbol{x}_5 - \boldsymbol{x}_2) \wedge (\boldsymbol{x}_5 - \boldsymbol{x}_1) I_2^{-1}}{(\boldsymbol{x}_5 - \boldsymbol{x}_1) \wedge (\boldsymbol{x}_5 - \boldsymbol{x}_3) I_2^{-1} (\boldsymbol{x}_5 - \boldsymbol{x}_2) \wedge (\boldsymbol{x}_5 - \boldsymbol{x}_4) I_2^{-1}} = \frac{A_{543} A_{521}}{A_{513} A_{524}}, \tag{53}$$

where $\frac{1}{2} A_{ijk}$ is the area of the triangle defined by the three vertices $\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_k$. This invariant is regarded as the 2D generalization of the 1D cross-ratio.

6.2. 3D Generalization of the Cross-Ratio

For general points in $\mathcal{E}^3$, we have seen that we move up one dimension to compute in the 4D space R^4. For this dimension, the point $\boldsymbol{x} = x\sigma_1 + y\sigma_2 + z\sigma_3$ in $\mathcal{E}^3$ is written as $X = X\gamma_1 + Y\gamma_2 + Z\gamma_3 + W\gamma_4$ where $x = X/W$, $y = Y/W$, $z = Z/W$. As before, a nonlinear projective transformation in $\mathcal{E}^3$ becomes a linear transformation, described by the linear function $\underline{f}_3$ in R^4.

Let us consider 4-vectors in R^4, $\{\mathbf{X}_i\}$, $i = 1, \ldots, 4$ and form the equation of a 4-vector:

$$\mathcal{S}_3 = \mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_3 \wedge \mathbf{X}_4 = \lambda_3 I_4, \tag{54}$$

where $I_4 = \gamma_1\gamma_2\gamma_3\gamma_4$ is the pseudoscalar for R^4. As before, $\mathcal{S}_3$ transforms to $\mathcal{S}_3'$ under $\underline{f}_3$:

$$\mathcal{S}_3' = \mathbf{X}_1' \wedge \mathbf{X}_2' \wedge \mathbf{X}_3' \wedge \mathbf{X}_4' = \boldsymbol{det}\, \underline{f}_3 \mathcal{S}_3. \tag{55}$$

The ratio of any two 4-vectors is therefore invariant under $\underline{f}_3$ and we must take multiples of these ratios to ensure that the arbitrary scale factors W_i cancel. With five general points we see that there are five possibilities for forming the combinations $W_i W_j W_k W_l$. It is then a simple matter to show that one cannot consider multiples of ratios such that the W factors cancel. It is, however, possible to do this if we have six points. One example of such an invariant might be

$$Inv_3 = \frac{(\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_3 \wedge \mathbf{X}_4) I_4^{-1} (\mathbf{X}_4 \wedge \mathbf{X}_5 \wedge \mathbf{X}_2 \wedge \mathbf{X}_6) I_4^{-1}}{(\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_4 \wedge \mathbf{X}_5) I_4^{-1} (\mathbf{X}_3 \wedge \mathbf{X}_4 \wedge \mathbf{X}_2 \wedge \mathbf{X}_6) I_4^{-1}}. \tag{56}$$

Using the arguments of the previous sections, we can now write:

$$(\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_3 \wedge \mathbf{X}_4) I_4^{-1} \equiv W_1 W_2 W_3 W_4 \{(\boldsymbol{x}_2 - \boldsymbol{x}_1) \wedge (\boldsymbol{x}_3 - \boldsymbol{x}_1) \wedge (\boldsymbol{x}_4 - \boldsymbol{x}_1)\} I_3^{-1}. \tag{57}$$

We can therefore see that the invariant Inv_3 is the 3D equivalent of the 1D cross-ratio and consists of ratios of volumes,

$$Inv_3 = \frac{V_{1234} V_{4526}}{V_{1245} V_{3426}}, \tag{58}$$

where V_{ijkl} is the volume of the solid formed by the four vertices $\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_k, \boldsymbol{x}_l$.

Conventionally, all of these invariants are well known, but we have outlined here a general process which is straightforward and simple for generating projective invariants in any dimension.

6.3. Generation of Geometric Projective Invariants

We choose for the visual projective space P^3 the geometric algebra $\mathcal{G}_{1,3,0}$ and for the image or projective plane P^2 the geometric algebra $\mathcal{G}_{3,0,0}$. Any 3D point is written in $\mathcal{G}_{1,3,0}$ as $\mathbf{X}_n = X_n\gamma_1 + Y_n\gamma_2 + Z_n\gamma_3 + W_n\gamma_4$ and its projected image point in $\mathcal{G}_{3,0,0}$ as $\boldsymbol{x}_n = x_n\sigma_1 + y_n\sigma_2 + z_n\sigma_3$, where $x_n = X_n/W_n$, $y_n = Y_n/W_n$, $z_n = Z_n/W_n$. The 3-D projective basis consists of four basis points and a fifth one for normalization: $\boldsymbol{X}_1 = [1, 0, 0, 0]^T, \boldsymbol{X}_2 = [0, 1, 0, 0]^T, \boldsymbol{X}_3 = [0, 0, 1, 0]^T, \boldsymbol{X}_4 = [0, 0, 0, 1]^T, \boldsymbol{X}_5 = [1, 1, 1, 1]^T$ and the 2-D projective basis comprises three basis points and one for normalization: $\boldsymbol{x}_1 = [1, 0, 0]^T$, $\boldsymbol{x}_2 = [0, 1, 0]^T, \boldsymbol{x}_3 = [0, 0, 1]^T, \boldsymbol{x}_4 = [1, 1, 1]^T$. Using them we can express in terms of brackets for any 3D point its 3D projective coordinates X_n, Y_n, Z_n and its

2D projected coordinates x_n, y_n as well

$$\frac{X_n}{W_n} = \frac{[234n][1235]}{[2345][123n]}, \quad \frac{Y_n}{W_n} = \frac{[134n][1235]}{[1345][123n]},$$
$$\frac{Z_n}{W_n} = \frac{[124n][1235]}{[1245][123n]}. \tag{59}$$

$$\frac{x_n}{w_n} = \frac{[23n][124]}{[234][12n]}, \quad \frac{y_n}{w_n} = \frac{[13n][124]}{[134][12n]}. \tag{60}$$

These equations are projective invariants relations and they will be used to compute the position of a moving camera in subsection 9.2.

The projective structure and its projection on the 2-D image is related according the following geometric constraint

$$\begin{pmatrix} 0 & w_5Y_5 & -y_5Z_5 & (y_5-w_5)W_5 \\ w_5X_5 & 0 & -x_5Z_5 & (x_5-w_5)W_5 \\ 0 & w_6Y_6 & -y_6Z_6 & (x_5-w_5)W_5 \\ 0 & w_6Y_6 & -y6Z_6 & (y_6-w_6)W_6 \\ w_6X_6 & 0 & -x_6Z_6 & (x_6-w_6)W_6 \\ 0 & w_7Y_7 & -y7Z_7 & (y_7-w_7)W_7 \\ w_7Y_7 & 0 & -x_7Z_7 & (x_7-w_7)W_7 \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \end{pmatrix} \begin{pmatrix} X_0^{-1} \\ Y_0^{-1} \\ Z_0^{-1} \\ W_0^{-1} \end{pmatrix} = 0, \tag{61}$$

where X_0, Y_0, Z_0, W_0 are the coordinates of the view point. Since the matrix is of rank < 4, any determinant of four rows becomes a zero. Considering as a normalizing point $(X_5, Y_5, Z_5, W_5) = (1, 1, 1, 1)$ and taking the determinant formed by the first four rows of Eq. (61) we get the geometric constraint equation involving six points which was pointed out by Quan [27]

$$\begin{aligned}&(w_5y_6 - x_5y_6)X_6Z_6 + (x_5y_6 - x_5w_6)X_6W_6 \\ &+ (x_5w_6 - y_5w_6)X_6Y_6 + (y_5x_6 - w_5x_6)Y_6Z_6 \\ &+ y_5w_6 - y_5x_6)Y_6W_6 + (w_5x_6 - w_5y_6)Z_6W_6 = 0\end{aligned} \tag{62}$$

Carlsson [6] showed that the Eq. (62) can be also derived using the *Plücker-Grassmann relations*. This can be computed as the *Laplace expansion* of the 4×8 rectangular matrix involving the same six points as above

$$\begin{aligned}&[X_1, X_2, X_3, X_4, X_5, X_5, X_6, X_7] \\ &= [X_0, X_1, X_2, X_3][X_4, X_5, X_6, X_7] \\ &\quad - [X_0, X_1, X_2, X_4][X_3, X_5, X_6, X_7] \\ &\quad + [X_0, X_1, X_2, X_5][X_3, X_4, X_6, X_7] \\ &\quad - [X_0, X_1, X_2, X_6][X_3, X_4, X_5, X_7] \\ &\quad + [X_0, X_1, X_2, X_7][X_3, X_4, X_5, X_6] = 0.\end{aligned} \tag{63}$$

Using four functions like Eq. (63) in terms of permutations of eight points, we get an expression where the brackets having two identical points vanish

$$\begin{aligned}&[0152][1345] - [0153][1245] + [0154][1235] = 0, \\ &[0216][2346] - [0236][1246] + [0246][1236] = 0, \\ &[0315][2345] + [0325][1345] + [0345][1235] = 0, \\ &[0416][2346] + [0426][1346] - [0436][1246] = 0,\end{aligned} \tag{64}$$

Here the points are indicated only with their subindices. It is easy to show that the brackets of image points can be written in the form $[\boldsymbol{x}_i\boldsymbol{x}_j\boldsymbol{x}_k] = w_iw_jw_k[K][X_0X_iX_jX_k]$, where [K] is the matrix of the intrinsic parameters [23]. Now if in Eq. (64) we express all the brackets which have the point X_0 in terms of the brackets of image points and organize all the bracket products as a 4×4 matrix we get the singular matrix

$$\begin{pmatrix} 0 & [125][1345] & [135][1245] & [145][1235] \\ [216][2346] & 0 & [236][1246] & [246][1236] \\ [315][2345] & [325][1345] & 0 & [345][1235] \\ [416][2346] & [426][1346] & [436][1246] & 0. \end{pmatrix} \tag{65}$$

Note that the scalars $w_iw_jw_k[K]$ in each matrix entry cancell themselves. Now after taking the determinat of these matrix and rearrange the terms conveniently, we obtain the following useful bracket polynom

$$\begin{aligned}&[125][346][1236][1246][1345][2345] \\ &\quad - [126][345][1235][1245][1346][2346] \\ &\quad + [135][246][1236][1245][1346][2345] \\ &\quad - [136][245][1235][1246][1345][2346] \\ &\quad + [145][236][1235][1246][1346][2345] \\ &\quad - [146][235][1236][1245][1345][2346] = 0.\end{aligned} \tag{66}$$

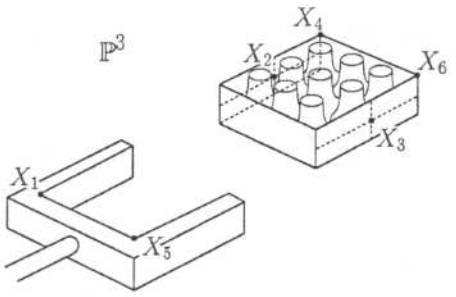


Figure 4. Grasping a box.

Surprisingly this bracket expression is exactly the *shape constraint* for six points given by Quan [27]

$$i_1 I_1 + i_2 I_2 + i_3 I_3 + i_4 I_4 + i_5 I_5 + i_6 I_6 = 0, \quad (67)$$

where the $i_1 = [125][346]$, $i_2 = [126][345]$, $i_3 = \ldots$ and and $I_1 = [1236][1246][1345][2345]$, $I_2 = \ldots$ are the relative linear invariants in P^2 and P^3 respectively. Using the shape constraint we are now ready to derive invariants for different purpose. Let us illustrate this with and example. According the Fig. 4 there is a configuration of six points which indicates whether or not the end-effector is grasping properly. To test this situation we can use an invariant generated from the constraint of Eq. (66). In this particular situation we recognize two planes, thus $[1235] = 0$ and $[2346] = 0$. Substituting these six points in Eq. (66) some brackets vanish reducing the equation to

$$\begin{aligned}
&[125][346][1236][1246][1345][2345] \\
&\quad - [135][246][1236][1245][1346][2345] = 0 \\
&[125][346][1246][1345] \\
&\quad - [135][246][1245][1346] = 0
\end{aligned} \quad (68)$$

or

$$\begin{aligned}
&Inv \\
&= \frac{(\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_4 \wedge \mathbf{X}_5) I_4^{-1} (\mathbf{X}_1 \wedge \mathbf{X}_3 \wedge \mathbf{X}_4 \wedge \mathbf{X}_6) I_4^{-1}}{(\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_4 \wedge \mathbf{X}_6) I_4^{-1} (\mathbf{X}_1 \wedge \mathbf{X}_3 \wedge \mathbf{X}_4 \wedge \mathbf{X}_5) I_4^{-1}} \\
&= \frac{(x_1 \wedge x_2 \wedge x_5) I_3^{-1} (x_3 \wedge x_4 \wedge x_6) I_3^{-1}}{(x_1 \wedge x_3 \wedge x_5) I_3^{-1} (x_2 \wedge x_4 \wedge x_6) I_3^{-1}}.
\end{aligned} \quad (69)$$

In this equation any bracket of P^3 after the projective mapping fulfills

$$\begin{aligned}
&(\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_4 \wedge \mathbf{X}_5) I_4^{-1} \\
&\quad \equiv W_1 W_2 W_4 W_5 \{(x_2 - x_1) \wedge (x_4 - x_1) \wedge \\
&\quad (x_5 - x_1)\} I_3^{-1},
\end{aligned} \quad (70)$$

The constraint (66) makes always sure that the $W_i W_j W_k W_l$ constants are canceled. Furthermore, according Eqs. (53–58), we can interpret nicely the invariant *Inv*, the equivalent of the 1-D *cross-ratio*, in P^3 as ratios of volumes and in P^2 as rations of triangle areas

$$Inv = \frac{V_{1245} V_{1346}}{V_{1246} V_{1345}} = \frac{A_{125} A_{346}}{A_{135} A_{246}}. \quad (71)$$

In other words We can also see this invariant in P^3 as the relation of 4-vectors or volumes build by points lying on a quadric which projected in P^2 represents an invariant build by areas of triangles encircled by conics.

For example utilizing this invariant we can check whether or not the grasper is holding the box correctly, see Fig. 4. Note that using the observed 3-D points in the image we can compute this invariant and see if the relation of the triangle areas corresponds with the appropriate relation for firm grasping, i.e. if the grasper is away the invariant has different value than the invariant value if the points X_1, X_5 of the grasper as required are near to the objects points X_2, X_3.

7. 3D Projective Invariants from Multiple Views

In the previous section the projective invariant was explained within the context of homogeneous projective coordinates derived from a single image. Since, in general, objects in 3D space are observed from different positions, it would be convenient to be able to extend the projective invariant in terms of the linear constraints imposed by the geometry of two, three, or more cameras.

7.1. *Projective Invariants Using Two Views*

Let us consider a 3D *projective invariant* derived from Eq. (66):

$$\boldsymbol{Inv_3} = \frac{[\mathbf{X}_1 \mathbf{X}_2 \mathbf{X}_3 \mathbf{X}_4][\mathbf{X}_4 \mathbf{X}_5 \mathbf{X}_2 \mathbf{X}_6]}{[\mathbf{X}_1 \mathbf{X}_2 \mathbf{X}_4 \mathbf{X}_5][\mathbf{X}_3 \mathbf{X}_4 \mathbf{X}_2 \mathbf{X}_6]}. \quad (72)$$

The computation of the bracket

$$\begin{aligned}
[1234] &= (\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_3 \wedge \mathbf{X}_4) I_4^{-1} \\
&= ((\mathbf{X}_1 \wedge \mathbf{X}_2) \wedge (\mathbf{X}_3 \wedge \mathbf{X}_4)) I_4^{-1}
\end{aligned}$$

of four points from R^4, mapped onto camera-images with optical centers $\mathbf{A}_0$ and $\mathbf{B}_0$, suggests the use of

a binocular model based on incidence algebra techniques, as discussed in [4]. Defining the lines

$$L_{12} = \mathbf{X}_1 \wedge \mathbf{X}_2 = \left(\mathbf{A}_0 \wedge \boldsymbol{L}_{12}^A\right) \vee \left(\mathbf{B}_0 \wedge \boldsymbol{L}_{12}^B\right)$$
$$L_{34} = \mathbf{X}_3 \wedge \mathbf{X}_4 = \left(\mathbf{A}_0 \wedge \boldsymbol{L}_{34}^A\right) \vee \left(\mathbf{B}_0 \wedge \boldsymbol{L}_{34}^B\right),$$

where lines $\boldsymbol{L}_{ij}^A$ and $\boldsymbol{L}_{ij}^B$ are mappings of the line $\mathbf{L}_{ij}$ onto the two image planes, results in the expression

$$[1234] = [\mathbf{A}_0\mathbf{B}_0\mathbf{A}'_{1234}\mathbf{B}'_{1234}]. \tag{73}$$

Here, $\mathbf{A}'_{1234}$ and $\mathbf{B}'_{1234}$ are the points of intersection of the lines $\boldsymbol{L}_{12}^A$ and $\boldsymbol{L}_{34}^A$ or $\boldsymbol{L}_{12}^B$ and $\boldsymbol{L}_{34}^B$, respectively. These points, lying on the image planes, can be expanded using the mappings of three points $\mathbf{X}_i$, say $\mathbf{X}_1$, $\mathbf{X}_2$, $\mathbf{X}_3$, to the image planes. In other words, considering $\mathbf{A}_j$ and $\mathbf{B}_j$, $j = 1, 2, 3$, as projective bases, we can expand the vectors

$$\mathbf{A}'_{1234} = \alpha_{1234,1}\mathbf{A}_1 + \alpha_{1234,2}\mathbf{A}_2 + \alpha_{1234,3}\mathbf{A}_3$$
$$\mathbf{B}'_{1234} = \beta_{1234,1}\mathbf{B}_1 + \beta_{1234,2}\mathbf{B}_2 + \beta_{1234,3}\mathbf{B}_3.$$

Then, using Eq. (36), we can express

$$[1234] = \sum_{i,j=1}^{3} \tilde{F}_{ij}\alpha_{1234,i}\beta_{1234,j} = \boldsymbol{\alpha}_{1234}^T\tilde{F}\boldsymbol{\beta}_{1234}, \tag{74}$$

where $\tilde{F}$ is the fundamental matrix given in terms of the projective basis embedded in R^4, and $\boldsymbol{\alpha}_{1234} = (\alpha_{1234,1}, \alpha_{1234,2}, \alpha_{1234,3})$ and $\boldsymbol{\beta}_{1234} = (\beta_{1234,1}, \beta_{1234,2}, \beta_{1234,3})$ are corresponding points.

The ratio

$$\boldsymbol{Inv_{3F}} = \frac{\left(\boldsymbol{\alpha}_{1234}^T\tilde{F}\boldsymbol{\beta}_{1234}\right)\left(\boldsymbol{\alpha}_{4526}^T\tilde{F}\boldsymbol{\beta}_{4526}\right)}{\left(\boldsymbol{\alpha}_{1245}^T\tilde{F}\boldsymbol{\beta}_{1245}\right)\left(\boldsymbol{\alpha}_{3426}^T\tilde{F}\boldsymbol{\beta}_{3426}\right)} \tag{75}$$

is therefore seen to be an invariant using the views of two cameras [5]. Note that Eq. (75) is invariant for whichever values of the γ_4 components of the vectors $\mathbf{A}_i, \mathbf{B}_i, \mathbf{X}_i$, etc., are chosen. If we attempt to express the invariant of Eq. (75) in terms of what we actually observe, we may be tempted to express the invariant in terms of the homogeneous Cartesian image coordinates $\boldsymbol{a}'_i s, \boldsymbol{b}'_i s$ and the fundamental matrix F calculated from these image coordinates. In order to avoid this, it is necessary to transfer the computations of Eq. (75) carried out in R^4 to R^3. Thus,

if we define $\tilde{F}$ by

$$\tilde{F}_{kl} = (\mathbf{A}_k \cdot \gamma_4)(\mathbf{B}_l \cdot \gamma_4)F_{kl} \tag{76}$$

and consider the relationships $\alpha_{ij} = \frac{\mathbf{A}'_i \cdot \gamma_4}{\mathbf{A}_j \cdot \gamma_4}\boldsymbol{a}_{ij}$ and $\beta_{ij} = \frac{\mathbf{B}'_i \cdot \gamma_4}{\mathbf{B}_j \cdot \gamma_4}\boldsymbol{b}_{ij}$, we can claim

$$\alpha_{ij}\tilde{F}_{kl}\beta_{il} = (\mathbf{A}'_i \cdot \gamma_4)(\mathbf{B}'_i \cdot \gamma_4)\boldsymbol{a}_{ik}F_{kl}\boldsymbol{b}_{il}. \tag{77}$$

If F is subsequently estimated by some method, then $\tilde{F}$ as defined in Eq. (76) will also *act as a fundamental matrix* or *bilinear constraint* in R^4. Now, let us look again at the invariant $\boldsymbol{Inv_{3F}}$. As we demonstrated ealier, we can write the invariant as

$$\boldsymbol{Inv_{3F}} = \frac{\left(\boldsymbol{a}_{1234}^T F\boldsymbol{b}_{1234}\right)\left(\boldsymbol{a}_{4526}^T F\boldsymbol{b}_{4526}\right)\phi_{1234}\phi_{4526}}{\left(\boldsymbol{a}_{1245}^T F\boldsymbol{b}_{1245}\right)\left(\boldsymbol{a}_{3426}^T F\boldsymbol{b}_{3426}\right)\phi_{1245}\phi_{3426}} \tag{78}$$

where $\phi_{pqrs} = (\mathbf{A}'_{pqrs} \cdot \gamma_4)(\mathbf{B}'_{pqrs} \cdot \gamma_4)$. Therefore, we see that the ratio of the terms $\boldsymbol{a}^T F\boldsymbol{b}$, 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. They are formed as follows (see [17, 19]): Since $\boldsymbol{a}'_3$, $\boldsymbol{a}'_4$, and $\boldsymbol{a}'_{1234}$ are collinear, we can write $\boldsymbol{a}'_{1234} = \mu_{1234}\boldsymbol{a}'_4 + (1 - \mu_{1234})\boldsymbol{a}'_3$. 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, so that

$$\frac{(\mathbf{A}'_{1234} \cdot \gamma_4)(\mathbf{A}'_{4526} \cdot \gamma_4)}{(\mathbf{A}'_{3426} \cdot \gamma_4)(\mathbf{A}'_{1245} \cdot \gamma_4)} = \frac{\mu_{1245}(\mu_{3426} - 1)}{\mu_{4526}(\mu_{1234} - 1)}. \tag{79}$$

Note that 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 $\boldsymbol{b}'_{1234} = \lambda_{1234}\boldsymbol{b}'_4 + (1 - \lambda_{1234})\boldsymbol{b}'_3$, etc., the overall expression for the invariant becomes

$$\boldsymbol{Inv_{3F}} = \frac{\left(\boldsymbol{a}_{1234}^T F\boldsymbol{b}_{1234}\right)\left(\boldsymbol{a}_{4526}^T F\boldsymbol{b}_{4526}\right)}{\left(\boldsymbol{a}_{1245}^T F\boldsymbol{b}_{1245}\right)\left(\boldsymbol{a}_{3426}^T F\boldsymbol{b}_{3426}\right)} \cdot \frac{\mu_{1245}(\mu_{3426} - 1)\lambda_{1245}(\lambda_{3426} - 1)}{\mu_{4526}(\mu_{1234} - 1)\lambda_{4526}(\lambda_{1234} - 1)}. \tag{80}$$

In conclusion, given the coordinates of a set of six corresponding points in two image planes, where these six points are projections of arbitrary world points in general position, we can form 3D projective invariants, provided we have some estimate of F.

7.2. *Projective Invariant of Points Using Three Uncalibrated 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. Considering four world points $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3, \mathbf{X}_4$ or two lines $\mathbf{X}_1 \wedge \mathbf{X}_2$ and $\mathbf{X}_3 \wedge \mathbf{X}_4$ projected onto three camera planes, we can write

$$\mathbf{X}_1 \wedge \mathbf{X}_2 = (\mathbf{A}_0 \wedge \boldsymbol{L}_{12}^A) \cap (\mathbf{B}_0 \wedge \boldsymbol{L}_{12}^B)$$
$$\mathbf{X}_3 \wedge \mathbf{X}_4 = (\mathbf{A}_0 \wedge \boldsymbol{L}_{34}^A) \cap (\mathbf{C}_0 \wedge \boldsymbol{L}_{34}^C).$$

Once again, we can combine the above expressions so that they give an equation for the 4-vector $\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_3 \wedge \mathbf{X}_4$,

$$\begin{aligned}\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_3 \wedge \mathbf{X}_4 &= \left((\mathbf{A}_0 \wedge \boldsymbol{L}_{12}^A) \cap (\mathbf{B}_0 \wedge \boldsymbol{L}_{12}^B)\right) \\ &\quad \wedge \left((\mathbf{A}_0 \wedge \boldsymbol{L}_{34}^A) \cap (\mathbf{C}_0 \wedge \boldsymbol{L}_{34}^C)\right) \\ &= (\mathbf{A}_0 \wedge \mathbf{A}_{1234}) \wedge \left((\mathbf{B}_0 \wedge \boldsymbol{L}_{12}^B) \right. \\ &\quad \left. \cap (\mathbf{C}_0 \wedge \boldsymbol{L}_{34}^C)\right). \end{aligned} \tag{81}$$

Then, by rewriting the lines $\boldsymbol{L}_{12}^B$ and $\boldsymbol{L}_{34}^C$ in terms of the line coordinates, we get $\boldsymbol{L}_{12}^B = \sum_{j=1}^{3} l_{12,j}^B \boldsymbol{L}_j^B$ and $\boldsymbol{L}_{34}^C = \sum_{j=1}^{3} l_{34,j}^C \boldsymbol{L}_j^C$. As has been shown in subsection 5.2, the components of the *trifocal tensor* (which takes the place of the fundamental matrix for three views) can be written in geometric algebra as

$$\tilde{T}_{ijk} = \left[(\mathbf{A}_0 \wedge \mathbf{A}_i) \wedge \left((\mathbf{B}_0 \wedge \boldsymbol{L}_j^B) \cap (\mathbf{C}_0 \wedge \boldsymbol{L}_k^C)\right)\right], \tag{82}$$

so that by using Eq. (81) we can derive:

$$\begin{aligned}[\mathbf{X}_1 \wedge \mathbf{X}_2 \wedge \mathbf{X}_3 \wedge \mathbf{X}_4] &= \sum_{i,j,k=1}^{3} \tilde{T}_{ijk} \alpha_{1234,i} l_{12,j}^B l_{34,k}^C \\ &= \tilde{T}(\alpha_{1234}, \boldsymbol{L}_{12}^B, \boldsymbol{L}_{34}^C). \end{aligned} \tag{83}$$

The invariant $\boldsymbol{Inv}_3$ can then be expressed as

$$\boldsymbol{Inv}_{3T} = \frac{\tilde{T}(\alpha_{1234}, \boldsymbol{L}_{12}^B, \boldsymbol{L}_{34}^C)\tilde{T}(\alpha_{4526}, \boldsymbol{L}_{25}^B, \boldsymbol{L}_{26}^C)}{\tilde{T}(\alpha_{1245}, \boldsymbol{L}_{12}^B, \boldsymbol{L}_{45}^C)\tilde{T}(\alpha_{3426}, \boldsymbol{L}_{34}^B, \boldsymbol{L}_{26}^C)}. \tag{84}$$

Note that the factorization must be done so that the same line factorizations occur in both the numerator and denominator. We have thus developed an expression for invariants in three views that is a direct extension of the expression for invariants using two views.

In calculating the above invariant from observed quantities, we note, as before, that some correction factors will be necessary: Eq. (84) is given above in terms of R^4 quantities. Fortunately, this correction is quite straightforward. By extrapolating from the results of the previous section, we simply consider the $\alpha's$ terms in Eq. (84) as unobservable quantities, and conversely the line terms, such as $\boldsymbol{L}_{12}^B$, $\boldsymbol{L}_{34}^C$ are indeed observed quantities. As a result, the expression must be modified, by using to some extent the coefficients computed in the previous section. Thus, for the unique four combinations of three cameras their invariant equations can be expressed as

$$\begin{aligned}&\boldsymbol{Inv}_{3T} \\ &= \frac{T(a_{1234}, l_{12}^B, l_{34}^C)T(a_{4526}, l_{25}^B, l_{26}^C)\mu_{1245}(\mu_{3426} - 1)}{T(a_{1245}, l_{12}^B, l_{45}^C)T(a_{3426}, l_{34}^B, l_{26}^C)\mu_{4526}(\mu_{1234} - 1)}. \end{aligned} \tag{85}$$

8. Experimental Analysis of Projective Invariants

In this section we present results for the formation of 3D projective invariants from two or three views on both simulated and real data. The simulations (carried out in Maple) involve generating four different sets, S_i $i = 1, \ldots, 4$, of 6 points;

$$S_i = \{\mathbf{X}_1^i, \mathbf{X}_2^i, \mathbf{X}_3^i, \mathbf{X}_4^i, \mathbf{X}_5^i, \mathbf{X}_6^i\}$$

within some spatial volume. The volume was a spherical region whose dimensions were around a tenth of the distance of the center of the volume from the camera's optical center. These sets of points are then observed from four different viewpoints, so that the four sets of image coordinates for set S_i are given by s_{ij}, $j = 1, \ldots, 4$;

$$s_{ij} = \{x_{j1}^i, x_{j2}^i, x_{j3}^i, x_{j4}^i, x_{j5}^i, x_{j6}^i\}$$

For each set of 6 points the three linearly independent invariants I^1, I^2, I^3 are formed, where these are the standard invariants given as follows

$$\begin{aligned} I^1 &= \frac{[\mathbf{X}_1\mathbf{X}_2\mathbf{X}_3\mathbf{X}_4][\mathbf{X}_4\mathbf{X}_5\mathbf{X}_2\mathbf{X}_6]}{[\mathbf{X}_1\mathbf{X}_2\mathbf{X}_4\mathbf{X}_5][\mathbf{X}_3\mathbf{X}_4\mathbf{X}_2\mathbf{X}_6]} \\ I^2 &= \frac{[\mathbf{X}_1\mathbf{X}_2\mathbf{X}_3\mathbf{X}_5][\mathbf{X}_4\mathbf{X}_5\mathbf{X}_2\mathbf{X}_6]}{[\mathbf{X}_1\mathbf{X}_2\mathbf{X}_4\mathbf{X}_5][\mathbf{X}_3\mathbf{X}_5\mathbf{X}_2\mathbf{X}_6]} \\ I^3 &= \frac{[\mathbf{X}_1\mathbf{X}_2\mathbf{X}_3\mathbf{X}_6][\mathbf{X}_6\mathbf{X}_5\mathbf{X}_2\mathbf{X}_4]}{[\mathbf{X}_1\mathbf{X}_2\mathbf{X}_6\mathbf{X}_5][\mathbf{X}_3\mathbf{X}_6\mathbf{X}_2\mathbf{X}_4]} \end{aligned} \tag{86}$$

Invariants using F

0	0.590	0.670	0.460	0.063	0.650	0.750	0.643	0.148	0.600	0.920	0.724	0.900	0.838	0.690	0.960
	0	0.515	0.68		0.67	0.78	0.687		0.60	0.96	0.755		0.276	0.693	0.527
		0.59	0			0.86	0.145			0.71	0.97			0.98	0.59
			0.69				0.531				0.596				0.663

Invariants using T

0	0.590	0.310	0.630	0.044	0.590	0.326	0.640	0.031	0.100	0.352	0.660	0.000	0.640	0.452	0.700
	0	0.63	0.338		0	0.63	0.376		0.031	0.337	0.67		0.063	0.77	0.545
		0.134	0.67			0.192	0.67			0.31	0.67			0.321	0.63
			0.29				0.389				0.518				0.643

Figure 5. The distance matrices (using F upper row and using T lower row) show the performance of the invariants with increasing Gaussian noise σ (from left to right): 0.005, 0.015, 0.025 and 0.04.

These I's are formed using a) views 1 and 2, b) views 2 and 3, c) views 1, 2 and 3 and d) views 2, 3 and 4-in a) and b) the fundamental matrix is calculated by linear means and in c) and d) the trifocal tensor is derived also from a simple linear algorithm. Although these simple linear methods do not enforce the necessary constraints on F and T, the resulting estimates were adequate for the purposes of the experiments shown here.

These invariants of each set were represented as 3D vectors, $\mathbf{v}_i = [I_{1,i}, I_{2,i}, I_{3,i}]^T$. The comparison of the invariants was done using 'Euclidean distances' of the vectors, $d(\boldsymbol{v}_i, \boldsymbol{v}_j)$, where

$$d(\boldsymbol{v}_i, \boldsymbol{v}_j) = \left[1 - \left|\frac{\boldsymbol{v}_i \cdot \boldsymbol{v}_j}{\|\boldsymbol{v}_i\| \|\boldsymbol{v}_j\|}\right|\right]^{\frac{1}{2}} \tag{87}$$

For any $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ the distance $d(\boldsymbol{v}_i, \boldsymbol{v}_j)$ lies between 0 and 1 and it does not vary when $\boldsymbol{v}_i$ or $\boldsymbol{v}_j$ is multiplied by a nonzero constant—this follows Hartley's analysis given in [13].

Figure 5 shows two sets of tables. The (i, j)th entry in the top left-hand box shows the distance, $d(\boldsymbol{v}_i, \boldsymbol{v}_j)$, between invariants for set S_i formed from views 1 and 2 and invariants for set S_j formed from views 2 and 3, when Gaussian noise of $\sigma = 0.005$ was added to the image points. The next boxes show the same thing for increasing σ. The lower row shows the equivalent for invariants formed from three views using the expression given in Section 3; here the (i, j)th entry in the top right-hand box shows the distance, $d(\boldsymbol{v}_i, \boldsymbol{v}_j)$, between invariants for set S_i formed from views 1–3, and invariants for set S_j formed from views 2–4. Clearly, we would like the diagonal elements to be as close as possible to zero since the invariants should be the same in all views in the zero noise case. The off-diagonal elements give some indication of the usefulness of the invariants in distinguishing between sets of points (we would like these to be as close to 1 as possible). We can see that the performance of the invariants based on trilinearities is rather better than those based on bilinearities.

In the case of real images we use a sequence of images taken by a moving robot equipped with a binocular head. Figure 6 shows an example of images taken with the left and right eyes. Image pairs, one from the left sequence and one from the right sequence were taken to form invariants using F. For the formation of invariants using T, two from the left and one from the right sequence were used. 38 points were semi-automatically taken and 6 sets of 6 general points were selected. The vector of invariants for each set was formed using both F and T and the set of distances found are shown in Fig. 7 We again see that computing the invariants from 3 views is more robust and useful than computing them from 2 views—one would expect this from a theoretical viewpoint.

9. Applications

This section present a practical use of projective invariants using three views. The results will show that despite certain noise sensitivity the projective invariants they can be used for various tasks regardless camera calibration and ignoring a home coordinate system.

9.1. Visual Guided Grasping

Let us now apply simple geometric rules using meet or join operations, invariants and points at infinity to the task of grasping as depicted in Fig. 8(a). The grasping procedure uses only image points and it consists basically of the following four steps.

9.1.1. Parallel Orienting. Let us assume that the 3D points of Fig. 8 are observed by three cameras A, B, C. The mapped points in the three cameras are $\{\boldsymbol{o}_{A_i}\}, \{\boldsymbol{g}_{A_i}\}, \{\boldsymbol{o}_{B_i}\}, \{\boldsymbol{g}_{B_i}\}$ and $\{\boldsymbol{o}_{C_i}\}, \{\boldsymbol{g}_{C_i}\}$. In the projective 3D space P^3 the three points at infinity V_x, V_y, V_z

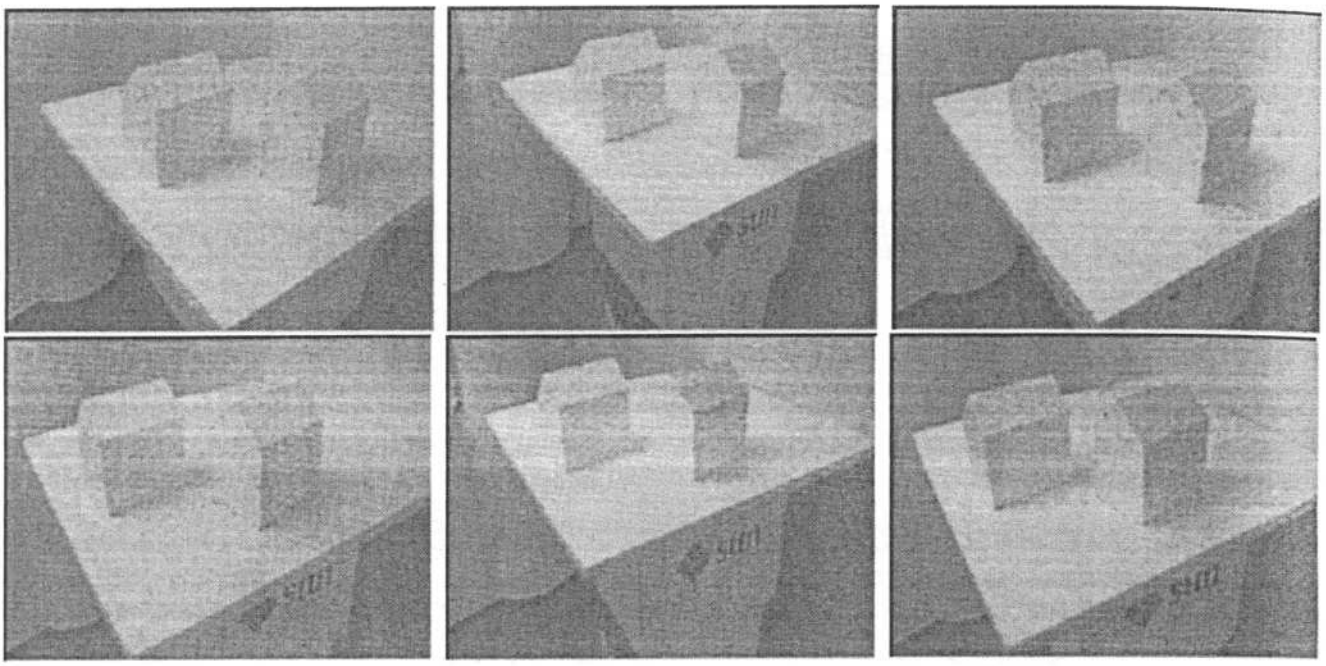


Figure 6. Image sequence taken during navigation by the binocular head of a mobile robot. The upper and lower rows shows the left and right eye images respectively.

using F

0.04	0.79	0.646	0.130	0.679	0.89
	0.023	0.2535	0.278	0.268	0.89
		0.0167	0.723	0.606	0.862
			0.039	0.808	0.91
				0.039	0.808
					0.039

using T

0.021	0.779	0.346	0.930	0.759	0.81
	0.016	0.305	0.378	0.780	0.823
		0.003	0.83	0.678	0.97
			0.02	0.908	0.811
				0.008	0.791
					0.01

Figure 7. The distance matrices show the performance of the computed invariants using bilinearities (left) and trilinearities (right) for the real image sequence.

for the orthogonal corners of the object can be computed as the meet of two parallel lines and similarly in the images planes the vanishing points v_x, v_y, v_z are computed as the meet of the two projected parallel lines as well

$$\begin{aligned} V_x &= (O_1 \wedge O_2) \vee (O_5 \wedge O_6) \\ v_{j_x} &= (o_{j_1} \wedge o_{j_2}) \vee (o_{j_5} \wedge o_{j_6}), \\ V_y &= (O_1 \wedge O_5) \vee (O_2 \wedge O_6) \\ v_{j_y} &= (o_{j_1} \wedge o_{j_5}) \vee (o_{j_2} \wedge o_{j_6}), \\ V_x &= (O_1 \wedge O_4) \vee (O_2 \wedge O_3) \\ v_{j_z} &= (o_{j_1} \wedge o_{j_4}) \vee (o_{j_2} \wedge o_{j_3}), \end{aligned} \tag{88}$$

where $j \in \{A, B, C\}$. The parallelism in the projective space P^3 can be checked in two ways:

a) If the orthogonal edges of the grasper are parallel with the edges of the object, then

$$\begin{aligned} &(G_1 \wedge G_8) \wedge V_x = 0, \qquad (G_1 \wedge G_9) \wedge V_y = 0, \\ &(G_1 \wedge G_2) \wedge V_z = 0. \end{aligned} \tag{89}$$

The conditions (89) using the points of a single camera can be expressed as

$$\begin{aligned} &[g_{i_1} g_{i_8} v_{i_x}] = 0, \qquad [g_{i_1} g_{i_9} v_{i_y}] = 0, \\ &[g_{i1} g_{i2} v_{i_z}] = 0. \end{aligned} \tag{90}$$

b) If the perpendicular planes of the grasper and those of the object are parallel, then

$$\begin{aligned} &[G_1 G_8 O_1 O_2] = 0, \qquad [G_{15} G_{16} O_5 O_8] = 0, \\ &[G_{12} G_{13} O_3 O_4] = 0. \end{aligned} \tag{91}$$

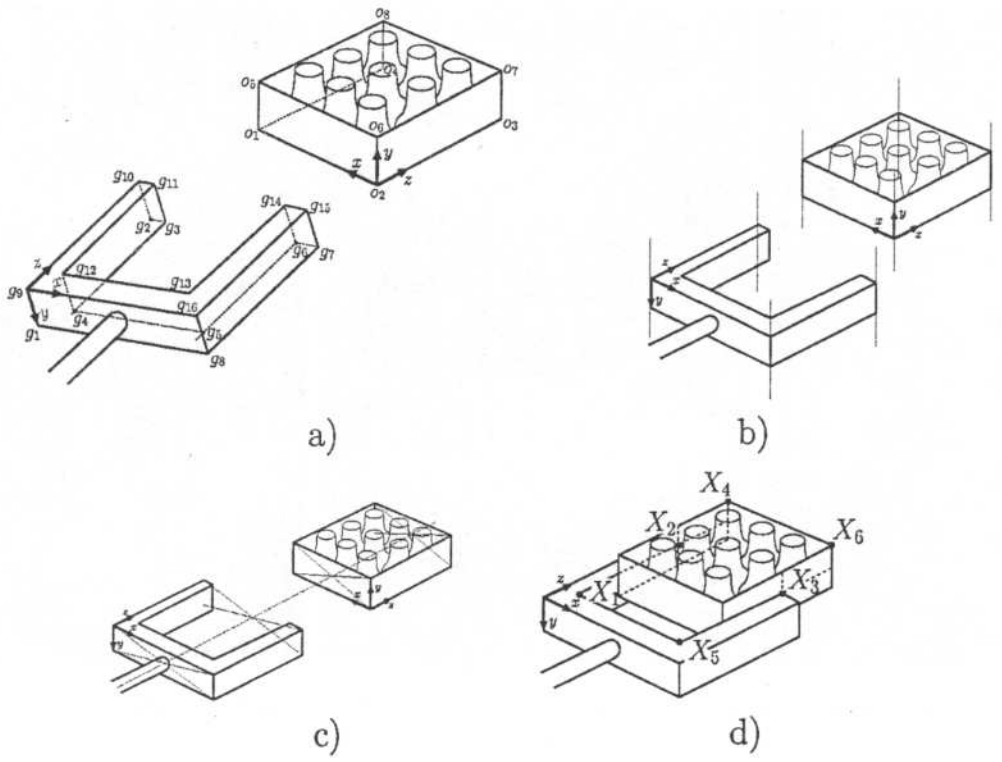


Figure 8. Grasping an object: a) Arbitrary position for grasping b) parallel orienting c) centering d) optimal grasping.

The conditions (91) can be expressed en terms of image coordinates either using two cameras (bilinear constraint) or three cameras (trifocal tensor)

$$\begin{aligned} \boldsymbol{x}^T_{j_{g_1 g_8 o_1 o_2}} F_{ij} \boldsymbol{x}_{i_{g_1 g_8 o_1 o_2}} &= 0, \\ \boldsymbol{x}^T_{j_{g_{15} g_{16} o_5 o_8}} F_{ij} \boldsymbol{x}_{i_{g_{15} g_{16} o_5 o_8}} &= 0, \\ \boldsymbol{x}^T_{j_{g_{12} g_{13} o_3 o_4}} F_{ij} \boldsymbol{x}_{i_{g_{12} g_{13} o_3 o_4}} &= 0, \end{aligned} \tag{92}$$

$$\begin{aligned} T_{ijk} x_{i_{g_1 g_8 o_1 o_2}} \boldsymbol{l}_{j_{g_1 g_8}} \boldsymbol{l}_{k_{o_1 o_2}} &= 0, \\ T_{ijk} x_{i_{g_{15} g_{16} o_5 o_8}} \boldsymbol{l}_{j_{g_{15} g_{16}}} \boldsymbol{l}_{k_{o_5 o_8}} &= 0, \\ T_{ijk} x_{i_{g_{12} g_{13} o_3 o_4}} \boldsymbol{l}_{j_{g_{12} g_{13}}} \boldsymbol{l}_{k_{o_3 o_4}} &= 0. \end{aligned} \tag{93}$$

If the trinocular geometry is known, it is always more accurate to use Eq. (93).

9.1.2. Centering. After a first movement the grasper should be parallel and centered in front of the object, see Fig. 8(b). The center points of the grasper and object can be computed as follows

$$\begin{aligned} \boldsymbol{C}_o &= (\boldsymbol{O}_1 \wedge \boldsymbol{O}_6) \vee (\boldsymbol{O}_2 \wedge \boldsymbol{O}_5), \\ \boldsymbol{C}_g &= (\boldsymbol{G}_1 \wedge \boldsymbol{G}_{16}) \vee (\boldsymbol{G}_8 \wedge \boldsymbol{G}_9). \end{aligned} \tag{94}$$

We can then check whether the line crossing these center points encounters the point at infinity $\boldsymbol{V}_z$ which is the intersecting point of the parallel lines $\boldsymbol{O}_{j_1} \wedge \boldsymbol{O}_{j_4}$ and $\boldsymbol{O}_{j_2} \wedge \boldsymbol{O}_{j_3}$. For that we use the constraint: a point lies on a line if it is true that

$$\boldsymbol{C}_o \wedge \boldsymbol{C}_g \wedge \boldsymbol{V}_z = 0. \tag{95}$$

This equation computed using image points of a single camera is given by

$$\left[\boldsymbol{c}_{i_o} \boldsymbol{c}_{i_g} \boldsymbol{v}_{i_z}\right] = 0. \tag{96}$$

9.1.3. Grasping. We can detect the grasping situation when the plane of the grasper touches the plane of the object. This can be checked using the coplanar plane condition as follows

$$[\boldsymbol{C}_o \boldsymbol{C}_g \boldsymbol{o}_1 \boldsymbol{o}_2] = 0. \tag{97}$$

Since we want to use image points we can compute this bracket straightforward either using two or three cameras employing the bilinear or trilinear constraint respectively

$$\begin{aligned} \boldsymbol{x}^T_{j_{c_o c_g o_1 o_2}} F_{ij} \boldsymbol{x}_{i_{c_o c_g o_1 o_2}} &= 0, \\ T_{ijk} \boldsymbol{x}_{i_{c_o c_g o_1 o_2}} \boldsymbol{l}_{j_{c_o c_g}} \boldsymbol{l}_{k_{o_1 o_2}} &= 0. \end{aligned} \tag{98}$$

If the epipolar or trinocular geometry is known, it is always more accurate to use Eq. (98).

9.1.4. Holding the Object. The final step is to held the object correctly, see Fig. 8(d). This can be checked using the invariant in terms of the trifocal tensor given by Eq. (85). In this particular problem, the perfect condition will be when the invariant has the approximated value of $\frac{3}{4}$, for the case when the grasper is a bit away of the control point X_2, X_3 the invariant becomes the values for example of $\frac{6}{8}$ or $\frac{5}{6}$. Note that the invariant elegantly relates volumes indicating a particular relationship between the points of the grasper and of the object.

9.2. Camera Self-Localization

We will use now the Eq. (59) to compute the 3-D coordinates of a moving uncalibrated camera. For that, we select first as a projective basis five fixed points in the 3-D space X_1, X_2, X_3, X_4, X_5 and consider the unknown point X_6 as the optical center of the moving camera, see Fig. 9. Assuming that the camera does not moves on a plane the projection of the optical center X_6 of the first camera position corresponds to the epipole in any of the subsequent views. We can then compute the moving optical center using two cameras,

$$I_x^F = \frac{X_6}{W_6} = \frac{\left(\delta_{2346}^T \boldsymbol{F} \varepsilon_{2346}\right)\left(\delta_{1235}^T \boldsymbol{F} \varepsilon_{1235}\right)\mu_{2345}\lambda_{2345}\mu_{1236}\lambda_{1236}}{\left(\delta_{2345}^T \boldsymbol{F} \varepsilon_{2345}\right)\left(\delta_{1236}^T \boldsymbol{F} \varepsilon_{1236}\right)\mu_{2346}\lambda_{2346}\mu_{1235}\lambda_{1235}}. \tag{99}$$

or using three cameras

$$I_x^T = \frac{X_6}{W_6} = \frac{\left(T_{ijk}^{ABC}\alpha_{2346,i}l^B{}_{23,j}l^C{}_{46,k}\right)\left(T_{mnp}^{ABC}\alpha_{1235,m}l^B{}_{12,n}l^C_{35,p}\right)\mu_{2345}\mu_{1236}}{\left(T_{qrs}^{ABC}\alpha_{2345,q}l^B{}_{23,r}l^C{}_{45,s}\right)\left(T_{iuv}^{ABC}\alpha_{1236,t}l^B{}_{12,u}l^C{}_{36,v}\right)\mu_{2346}\mu_{1235}}. \tag{100}$$

Similarly permuting the six points like Eq. (60) 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. We carried out simulations by increasing noise. Figure 10 shows using two views or three views the deviation of the true optical center for three consecutive positions of a moving camera. These curves show that 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.

10. Projective Depth

In a geometric sense *projective depth* can be defined as the relation between the distance from the view-center of a 3D point $\boldsymbol{X}_i$ and the focal distance f, as depicted in Fig. 11.

We can derive projective depth from a projective mapping of 3D points. According to the pinhole model, the coordinates of any point in the image plane are obtained from the projection of the 3D point to the three optical planes $\phi_A^1, \phi_A^2, \phi_A^3$. They are spanned by a trivector basis $\gamma_i, \gamma_j, \gamma_k$ and the coefficients t_{ij}. This projective mapping in a matrix representation reads as

$$\lambda x = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \phi_A^1 \\ \phi_A^2 \\ \phi_A^3 \end{bmatrix} X = \begin{bmatrix} t_{11} & t_{12} & t_{13} & t_{14} \\ t_{21} & t_{22} & t_{23} & t_{24} \\ t_{31} & t_{32} & t_{33} & t_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \tag{101}$$

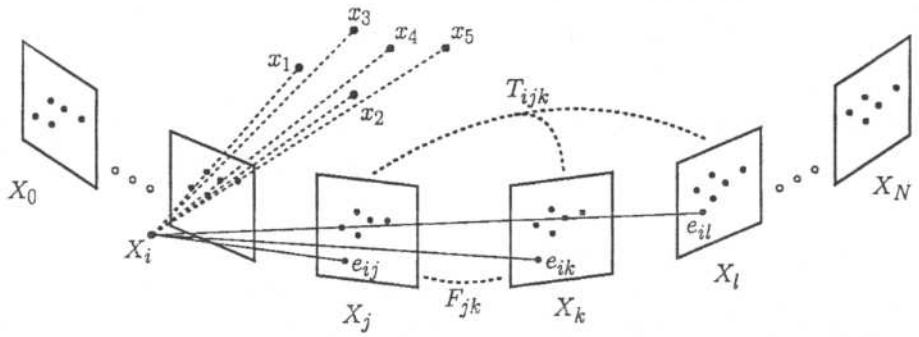
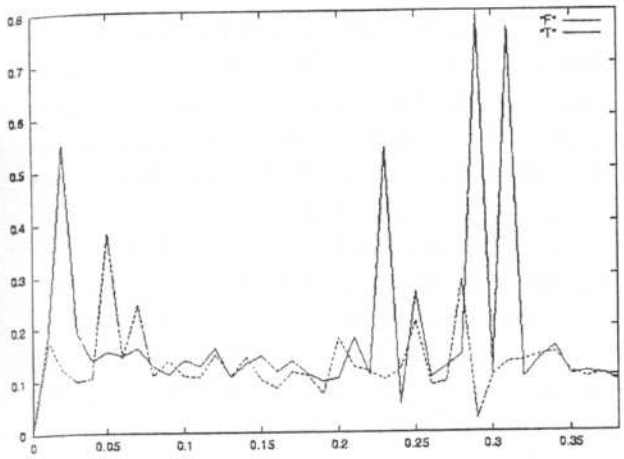
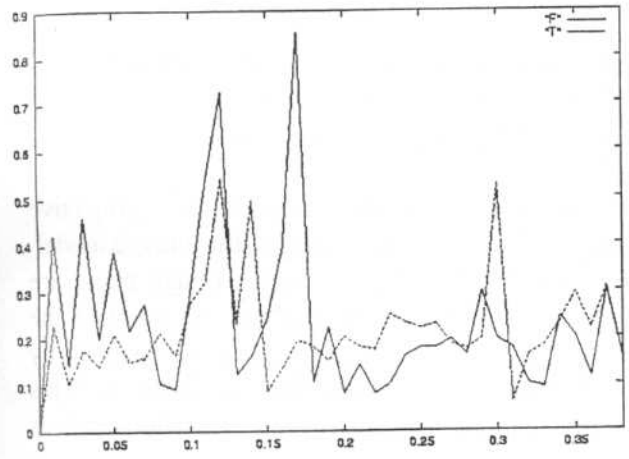


Figure 9. Computing the center of views of a moving camera.





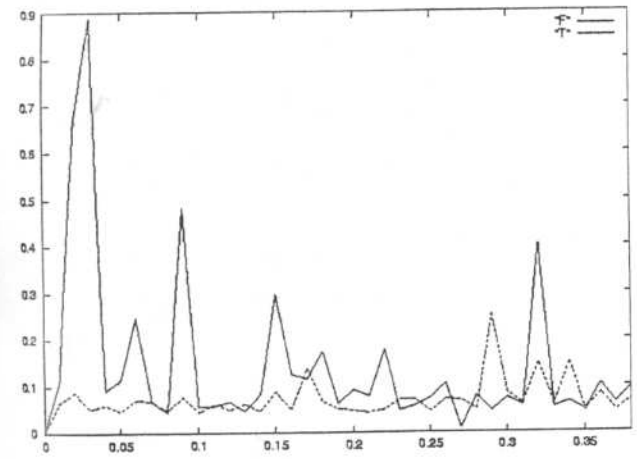


Figure 10. Performance of the computing of any three center of view using F (higher spikes) and T. Range of the additive noise 0 to 0.4 of pixel.

where the projective scale factor is called λ. Note that the projective mapping is further expressed in terms of a f, rotation, and translation components. Let us attach the world coordinates to the view-center of the

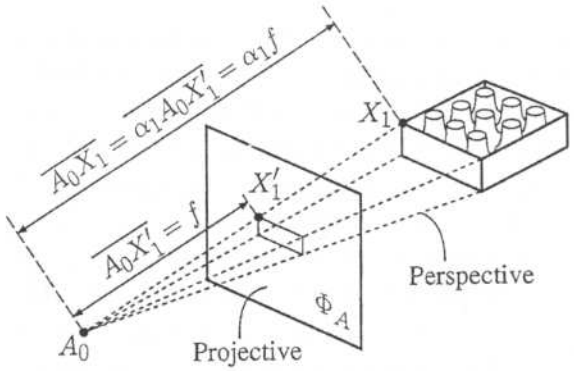


Figure 11. Geometric interpretation of projective depth.

camera. The resultant projective mapping becomes

$$\lambda \boldsymbol{x} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \equiv P\boldsymbol{X}. \tag{102}$$

We can then straightforwardly compute

$$\lambda = Z. \tag{103}$$

The method for computing the projective depth ($\equiv \lambda$) of a 3D point appears simple using invariant theory, namely, using Eq. (59). For this computation, we select a basis system, taking four 3D points in general position $\boldsymbol{X}_1, \boldsymbol{X}_2, \boldsymbol{X}_3, \boldsymbol{X}_5$, and the optical center of camera at the new position as the four point $\boldsymbol{X}_4$, and $\boldsymbol{X}_6$ as the 3D point which has to be reconstructed. This process is shown in Fig. 12.

Since we are using mapped points, we consider the *epipole* (mapping of the current view-center) to be the fourth point and the mapped sixth point to be the point with unknown depth. The other mapped basis points remain constant during the procedure.

According to Eq. (59), the tensor-based expression for the computation of the third coordinate, or projective depth, of a point $\boldsymbol{X}_j (= \boldsymbol{X}_6)$ is given by

$$\begin{aligned} \lambda_j &= \frac{Z_j}{W_j} \\ &= \frac{T\left(\boldsymbol{a}_{124j}, l^B_{12}, l^C_{4j}\right) T\left(\boldsymbol{a}_{1235j}, l^B_{12}, l^C_{35}\right)}{T\left(\boldsymbol{a}_{1245}, l^B_{12}, l^C_{45}\right) T\left(\boldsymbol{a}_{123j}, l^B_{12}, l^C_{3j}\right)} \cdot \frac{\mu_{1245}\mu_{123j}}{\mu_{124j}\mu_{1235}}. \end{aligned} \tag{104}$$

In this way, we can successively compute the projective depths λ_{ij} of the j-points relating to the i-camera. We will use λ_{ij} in Section 11, in which we

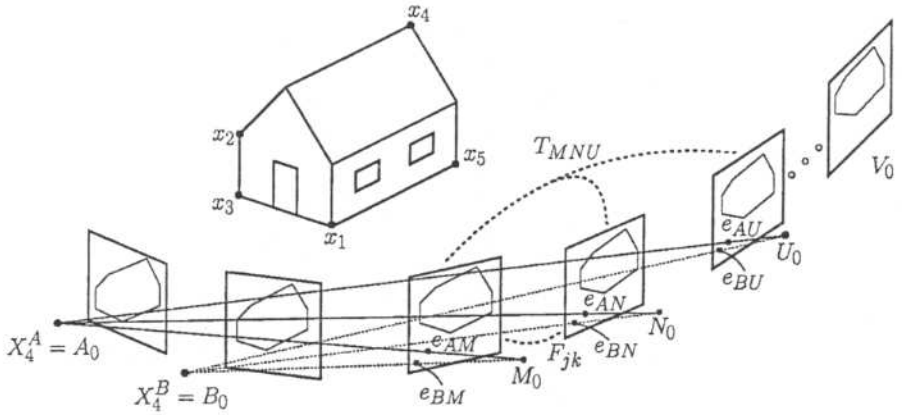


Figure 12. Computing the projective depths of n cameras.

employ the join-image concept and singular value decomposition (SVD) for *singular value decomposition* 3D reconstruction.

Since this type of invariant can also be expressed in terms of the quadrifocal tensor [17], we are also able to compute projective depth based on four cameras.

11. Shape and Motion

The orthographic and paraperspective *factorization method for shape and motion* using the affine camera model was developed by Tomasi, Kanade, and Poelman [26, 30]. This method works for cameras viewing small and distant scenes, and thus for all scale factors of projective depth $\lambda_{ij} = 1$. In the case of perspective images, the scale factors λ_{ij} are unknown. According to Triggs [31], all λ_{ij} satisfy a set of consistency reconstruction equations of the so-called *join-image*. One way to compute λ_{ij} is by using the epipolar constraint. If we use a matrix representation, this is given by

$$F_{ik}\lambda_{ij}\boldsymbol{x}_{ij} = \boldsymbol{e}_{ik} \wedge \lambda_{kj}\boldsymbol{x}_{kj}, \tag{105}$$

which, after computing an inner product with $\boldsymbol{e}_{ik} \wedge \boldsymbol{x}_{kj}$, gives the relation of projective depths for the j-point between camera i and k:

$$\lambda'_{kj} = \frac{\lambda_{kj}}{\lambda_{ij}} = \frac{(\boldsymbol{e}_{ik} \wedge \boldsymbol{x}_{kj})F_{ik}\boldsymbol{x}_{ij}}{\|(\boldsymbol{e}_{ik} \wedge \boldsymbol{x}_{kj})\|^2}. \tag{106}$$

Considering the i-camera as a reference, we can normalize λ_{kj} for all k-cameras and use λ'_{kj} instead. If that is not the case, we can normalize between neighbor images in a chained relationship [31].

In Section 10, we presented a better procedure for the computing of λ_{ij} involving three cameras. An extension of Eq. (106), however, in terms of the trifoca or quadrifocal tensor is awkward and unpractical.

11.1. The Join-Image

The *Join-image* $\mathcal{J}$ is nothing more than the intersec tions of optical rays and planes with points or lines ir 3D projective space, as depicted in Fig. 13. The inter related geometry can be linearly expressed by the fun damental matrix and trifocal and quadrifocal tensors.

In order to take into account the interrelated geom etry, the *projective reconstruction* procedure shoulc bring together all the data of the individual images in a geometrically coherent manner. We do this by consid ering the points $\boldsymbol{X}_j$ for each i-camera,

$$\lambda_{ij}\boldsymbol{x}_{ij} = P_i\boldsymbol{X}_j, \tag{107}$$

as the i-row points of a matrix of rank 4. For m camera and n points, the $3m \times n$ matrix $\mathcal{J}$ of the join-image is given by

$$\mathcal{J} = \begin{pmatrix} \lambda_{11}\boldsymbol{x}_{11} & \lambda_{12}\boldsymbol{x}_{12} & \lambda_{13}\boldsymbol{x}_{13} & \cdots & \lambda_{1n}\boldsymbol{x}_{1n} \\ \lambda_{21}\boldsymbol{x}_{21} & \lambda_{22}\boldsymbol{x}_{22} & \lambda_{23}\boldsymbol{x}_{23} & \cdots & \lambda_{2n}\boldsymbol{x}_{2n} \\ \lambda_{31}\boldsymbol{x}_{31} & \lambda_{32}\boldsymbol{x}_{32} & \lambda_{33}\boldsymbol{x}_{33} & \cdots & \lambda_{3n}\boldsymbol{x}_{3n} \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \lambda_{m1}\boldsymbol{x}_{m1} & \lambda_{m2}\boldsymbol{x}_{m2} & \lambda_{m3}\boldsymbol{x}_{m3} & \cdots & \lambda_{mn}\boldsymbol{x}_{mn} \end{pmatrix}. \tag{108}$$

For the ffine reconstruction procedure, the matrix is o rank 3. The matrix $\mathcal{J}$ of the join-image is therefore

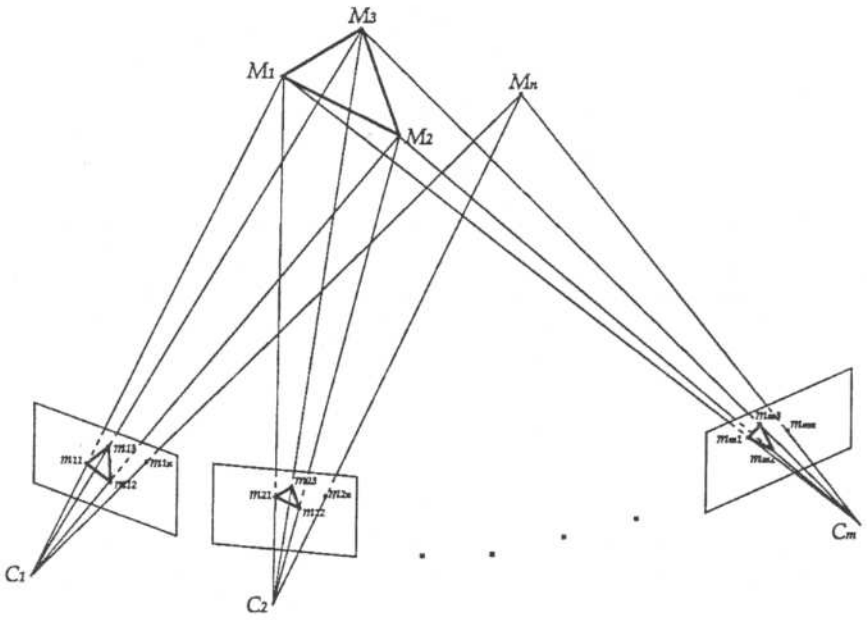


Figure 13. Geometry of the join-image.

amenable to a singular value decomposition for the computation of the shape and motion [26, 30].

11.2. The SVD Method

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

$$\mathcal{J}_{3m\times n} = U_{3m\times r} S_{r\times r} V^T_{n\times r}, \tag{109}$$

where the columns of matrix $V^T_{n\times r}$ and $U_{3m\times r}$ constitute the orthonormal base for the input (co-kernel) and output (range) spaces of $\mathcal{J}$. In order to get a decomposition in motion and shape of the projected point structure, $S_{r\times r}$ can be absorbed into both matrices, $V^T_{n\times r}$ and $U_{3m\times r}$, as follows:

$$\begin{aligned}\mathcal{J}_{3m\times n} &= \left(U_{3m\times r} S^{\frac{1}{2}}_{r\times r}\right)\left(S^{\frac{1}{2}}_{r\times r} V^T_{n\times r}\right)\\ &= \begin{pmatrix} P_1 \\ P_2 \\ P_3 \\ \cdot \\ \cdot \\ \cdot \\ P_m \end{pmatrix}_{3m\times 4} (X_1 X_2 X_3 \ldots X_n)_{4\times n}. \end{aligned} \tag{110}$$

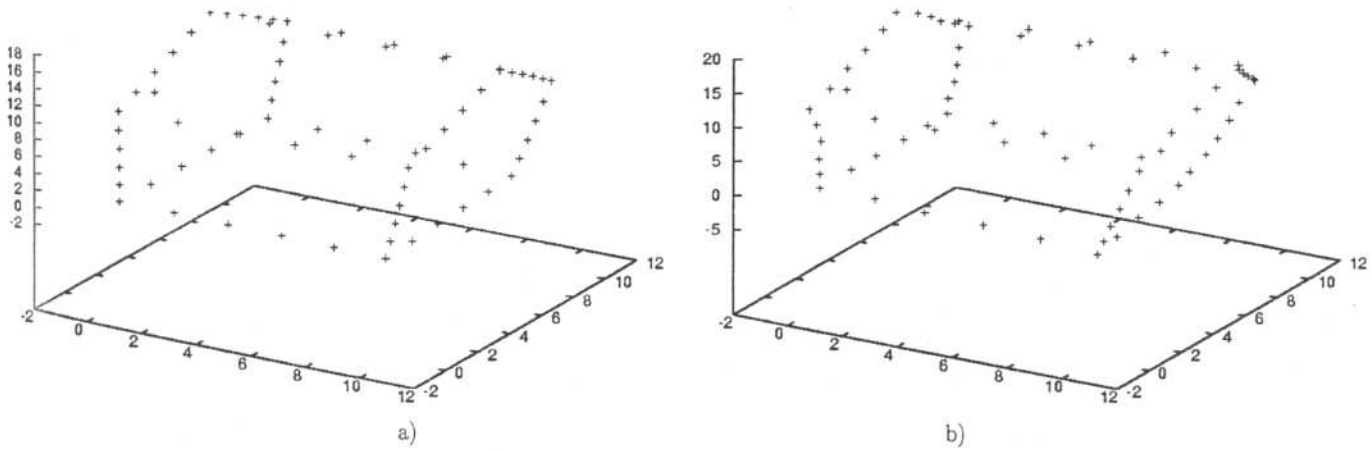


Figure 14. Reconstructed house using (a) noise-free observations and (b) noisy observations.

Using this method to divide $S_{r\times r}$ is not unique. Since the rank of $\mathcal{J}$ is 4, we should use the first four biggest singular values for $S_{r\times r}$. The matrices P_i correspond to the projective mappings or *motion* from the projective space to the individual images, and X_j represents the point structure or *shape*. We can test our approach by using a simulation program written in Maple. Using the method described in Section 10, We first compute the projective depth of the points of a wire house observed with nine cameras, and we then use SVD to obtain the house's shape and motion. The reconstructed house, after the Euclidean readjustment for the presentation, is shown in Fig. 14.

We note that the reconstruction preserves the original form of the model quite well.

In the following section, we will show how to improve the shape of the reconstructed model by using the geometric expressions $\cap$ (meet) and $\wedge$ (join) from the algebra of incidence along with particular tensor-based invariants.

11.3. Completion of the 3D Shape Using Geometric Invariants

Projective structure can be improved in one of two ways: (1) by adding points on the images, expanding the join- image, and then applying the SVD procedure; or (2) after the reconstruction is done, by computing new or occluded 3D space points. Both approaches can use, on the one hand, geometric inference rules based on symmetries, or on the other, concrete knowledge about the object. Using three real views of a similar model house with its rightmost lower corner missing (see Fig. 15(b)), we computed in each image the virtual image point of this 3D point. Then we reconstructed the scene, as shown in Fig. 15(c). We also tried using geometric incidence operations to complete the house, employing space points as depicted in Fig. 15(d). The figures show 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 into a Euclidean one for the presentation of the results. We also used lines to connect the reconstructed points but only so as to make the form of the house visible. Similarly, we used the same procedures to reconstruct the house using nine images. The results are presented in Fig. 16(a–d).

The figure shows that the resulting reconstructed point is virtually the same in both cases, which allows us to conclude that for a limited number of views the

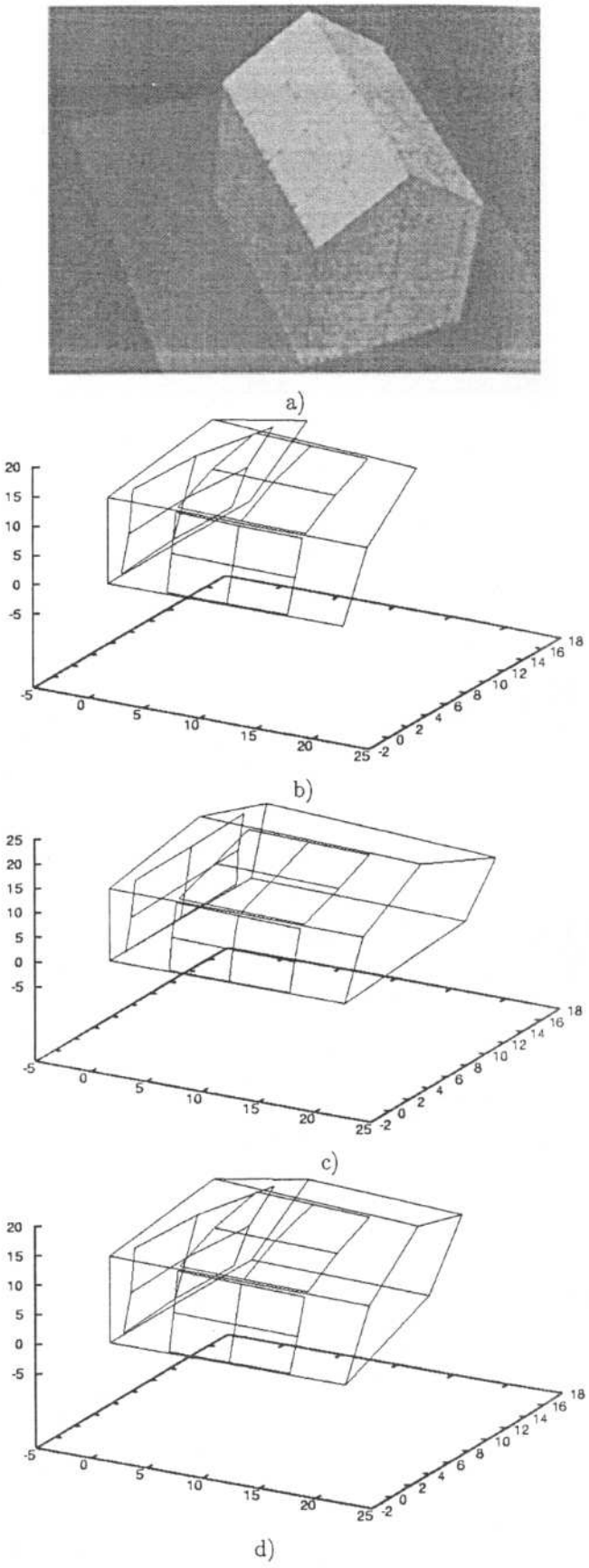


Figure 15. 3D reconstruction using three images: (a) one of the three images; (b) reconstructed incomplete house using three images; (c) extending the join-image; (d) completing in the 3D space.

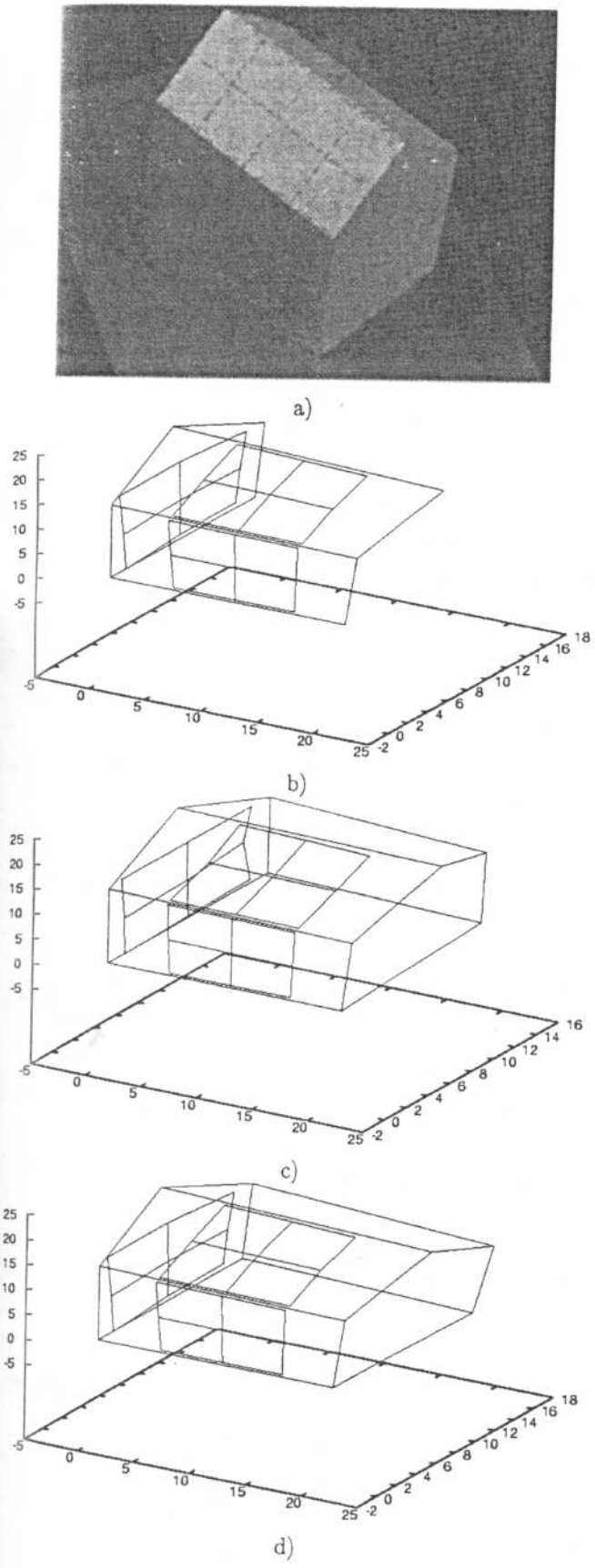


Figure 16. 3D reconstruction using nine images: (a) one of the nine images; (b) reconstructed incomplete house using nine images; (c) extending the join-image; (d) completing in the 3D space.

join- image procedure is preferable, but for the case of several images, an extension of the point structure in the 3D space is preferable.

12. Conclusions

This paper has presented a brief introduction to the techniques of geometric algebra and shown how they can be used in the algebra of incidence and in the formation and computation of projective invariants from n views. Analyzing problems using geometric algebra provides the enormous advantage of working in a system which has very powerful associated linear algebra and calculus frameworks and which can be used for most areas of computer vision.

This work focused on the study and application of projective invariants computed using two and three cameras. We conducted experiments using simulated and real images to compare projective invariants using two or three uncalibrated cameras. As applications we design geometric rules for conducting a task of visual guided grasping and We also presented the computation of the view center of a moving camera. The authors believe however 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 systems involving noisy data.

References

1. M. Barnabei, A. Brini, and G-C. Rota, "On the exterior calculus of invariant theory," *Journal of Algebra*, Vol. 96, pp. 120–160, 1985.
2. E. Bayro-Corrochano and J. Lasenby, "Object modelling and motion analysis using Clifford algebra," in *Proceedings of Europe-China Workshop on Geometric Modeling and Invariants for Computer Vision*, Roger Mohr and Wu Chengke (Eds.), Xi'an, China, April 27–29, 1995, pp. 143–149.
3. E. Bayro-Corrochano, J. Lasenby, and G. Sommer, "Geometric Algebra: A framework for computing point and line correspondences and projective structure using n-uncalibrated cameras," in *Proceedings of the International Conference on Pattern Recognition (ICPR'96)*. Vienna, Aug. 1996, Vol. I, pp. 393–397.
4. E. Bayro-Corrochano and G. Sobczyk, "Applications of Lie algebras and the algebra of incidence," in *Geometric Algebra with Applications in Science and Engineering*, Eduardo Bayro-Corrochano and G. Sobczyk (Eds.), Birkhauser: Boston, ch.13, 2001.
5. S. Carlsson, "The double algebra: An effective tool for computing invariants in computer vision," *Applications of Invariance in Computer Vision*, Lecture Notes in Computer Science, Vol. 825; in *Proceedings of 2nd-joint Europe-US Workshop*, Azores, Oct. 1993, J.L. Mundy, A. Zisserman, and D.A. Forsyth (Eds.), Springer-Verlag: Berlin.

6. S. Carlsson, "Symmetry in perspective," in *Proceedings of the European Conference on Computer Vision*, Freiburg, Germany, 1998, pp. 249–263.
7. W.K. Clifford, "Applications of Grassmann's extensive algebra," *Am. J. Math.*, Vol. 1878, pp. 350–358.
8. Ch.I. Colios and P.E. Trahanias, "Landmark identification based on projective and permutation invariant vectors," in *Proceedings of the International Conference on Pattern Recognition (ICPR'2000)*, Barcelona, Sept. 3–7, Vol. 4, 2000, pp. 128–131.
9. G. Csurka and O. Faugeras, "Computing three dimensional project invariants from a pair of images using the Grassmann-Cayley algebra," *Journal of Image and Vision Computing*, Vol. 16, pp. 3–12, 1998.
10. H. Grassmann, "Der ort der Hamilton'schen quaternionen in der ausdehnungslehre," *Math. Ann.*, Vol. 12, p. 375, 1877.
11. R.I. Hartley, "Lines and points in three views—a unified approach," in *ARPA Image Understanding Workshop*, Monterey, California, 1994.
12. R.I. Hartley, "The quadrifocal tensor," in *ECCV98*. LNCS, Springer-Verlag: Berlin, 1998.
13. R.I. Hartley, "Projective reconstruction and invariants from multiple images," *IEEE Trans. PAMI*, Vol. 16, No.10, pp. 1036–1041, 1994.
14. D. Hestenes and G. Sobczyk, "Clifford algebra to geometric calculus: A unified language for mathematics and physics," D. Reidel: Dordrecht, 1984.
15. D. Hestenes and R. Ziegler, "Projective geometry with Clifford algebra," *Acta Applicandae Mathematicae*, Vol. 23, pp. 25–63, 1991.
16. J. Lasenby, "Geometric algebra: Applications in engineering," in *Clifford (Geometric) Algebras: with Applications in Physics, Mathematics and Engineering*, W. Baylis (Ed.), Birkhauser: Boston, 1996, pp. 423–440.
17. J. Lasenby and E. Bayro-Corrochano, "Computing invariants in computer vision using geometric algebra," Technical Report CUED/F-INENG/TR.244, Cambridge University Engineering Department, Cambridge, UK, 1998.
18. J. Lasenby and E. Bayro-Corrochano, "Analysis and computation of projective invariants from multiple views in the geometric algebra framework," in *Special Issue on Invariants for Pattern Recognition and Classification*, M.A. Rodrigues (Ed.), *Int. Journal of Pattern Recognition and Artificial Intelligence*, Vol. 13, No. 8, 1999, pp. 1105–1121.
19. J. Lasenby, E. Bayro-Corrochano, A.N. Lasenby, and G. Sommer, "A new methodology for computing invariants in computer vision," in *Proceedings of the International Conference on Pattern Recognition (ICPR'96)*, Vienna, Aug. 1996, Vol. I, pp. 334–338.
20. J. Lasenby, E. Bayro-Corrochano, A.N. Lasenby, and G. Sommer, "A new framework for the computation of invariants and multiple-view constraints in computer vision," in *Proceedings of the International Conference on Image Processing (ICIP)*, Lausanne, Sept. 1996, Vol. II, pp. 313–316.
21. J. Lasenby, A.N. Lasenby, Lasenby, C.J.L. Doran, and W.J. Fitzgerald, "New geometric methods for computer vision—an application to structure and motion estimation," *International Journal of Computer Vision*, Vol. 26, No. 3, pp. 191–213, 1998.
22. Q.T. Luong and O.D. Faugeras, "The fundamental matrix: Theory, algorithms, and stability analysis," *Int. J. Comput. Vision*, Vol. 17, No. 1, pp. 43–76, 1995.
23. J. Mundy, "Object recognition based on geometry: Progress over three decades," *Phil. Trans. Roy. Soc. A*, J. Lasenby, A. Zisserman, R. Cipolla, and H.C. Longuet-Higgins (Eds.), Vol. 356, 1998, pp. 1213–1229.
24. J. Mundy and A. Zisserman (Eds.), *Geometric Invariance in Computer Vision*, MIT Press: Cambridge, MA, 1992.
25. C.J. Poelman and T. Kanade, "A paraperspective factorization method for shape and motion recovery," in J-O. Eklundh (Ed.), *European Conference on Computer Vision*, Stockholm, 1994, pp. 97–108.
26. L. Quan, "Invariants of 6 points from 3 uncalibrated images," in *Proc. of the European Conference of Computer Vision*, Vol. II, 1994, pp. 459–470.
27. C.A. Rothwell, D.A. Forsyth, A. Zisserman, and J.L. Mumndy, "Extracting projective structure from single perspective views of 3D point sets," in *Proc. of the 4th Int. Conf. Computer Vision, ICCV'93*, 1993, pp. 573–582.
28. A. Shashua and M. Werman, "Trilinearity of three perspective views and its associated tensor," in *Proceedings of ICCV'95*, MIT, 1995.
29. C. Tomasi and T. Kanade, "Shape and motion from image streams under orthography: A factorization method," *Int. J. Computer Vision*, Vol. 9, No. 2, pp. 137–154, 1992.
30. W. Triggs, "Matching constraints and the joint image," in *IEEE Computer Society Press, Proceedings of the Int. Conference of Computer Vision ICCV'95*, Boston, 1995, pp. 338–343.