16. Analysis and Computation of the Intrinsic Camera Parameters*

Eduardo Bayro-Corrochano and Bodo Rosenhahn

Institute of Computer Science and Applied Mathematics,
Christian-Albrechts-University of Kiel

16.1 Introduction

The computation of the intrinsic camera parameters is one of the most important issues in computer vision. The traditional way to compute the intrinsic parameters is using a known calibration object. One of the most important methods is based on the absolute conic and it requires as input only information about the point correspondences [163, 107]. As extension a recent approach utilizes the absolute quadric [235]. Other important groups of self-calibration methods either reduce the complexity if the camera motion is known in advance, for example as translation [66], or as rotation about known angles [5, 67], or by using active strategies and e.g. the vanishing point [56].

In this chapter we re-establish the idea of the absolute conic in the context of Pascal's theorem and we get equations different to the Kruppa equations [163, 107]. Although the equations are different, they rely on the same principle of invariance of the mapped absolute conic. The consequence is that we can generate equations so that we require only a couple of images whereas the Kruppa equation method requires at least three views [163]. However, as a prior knowledge the method requires the translational motion direction of the camera and the rotation about at least one fixed axis through a known

* This work has been supported by DFG Grant So-320-2-1.

angle in addition to the point correspondences. The paper will show that although the algorithm requires the extrinsic camera parameters in advance it has the following clear advantages: It is derived from geometric observations, it does not stick in local minima in the computation of the intrinsic parameters and it does not require any initialization at all. We hope that this proposed method derived from geometric thoughts gives a new point of view to the problem of camera calibration.

The chapter is organized as follows. Section two explains the conics and the theorem of Pascal. Section three reformulates the well known Kruppa equations for computer vision in terms of algebra of incidence. Section four presents a new method for computing the intrinsic camera parameters based on Pascal's theorem. Section five is devoted to the experimental analysis and section six to the conclusion part.

16.2 Conics and the Theorem of Pascal

The role of the conics and quadrics is well known in the projective geometry [213] because of their invariant properties with respect to projective transformations. This knowledge lead to the solution of crucial problems in computer vision [177]. The derivation of the Kruppa equations relies on the conic concept. These equations have been used in the last decade to compute the intrinsic camera parameters. In this chapter we will exploit further the conics concept and use Pascal's theorem to establish an equation system with clear geometric transparency. Next, we will explain the role of conics and that of Pascal's theorem in relation with a fundamental projective invariant. This section is mostly based on the interpretation of the linear algebra together with projective geometry in the Clifford algebra framework realized by Hestenes and Ziegler [118].

When we want to use projective geometry in computer vision, we utilize homogeneous coordinate representations. Doing that, we embed the 3–D Euclidean visual space in the 3–D projective space $\mathbb{P}^3$ or $\mathbb{R}^4$ and the 2–D Euclidean space of the image plane in the 2–D projective space $\mathbb{P}^2$ or $\mathbb{R}^3$. In the geometric algebra framework we select for $\mathbb{P}^2$ the 3–D Euclidean geometric algebra $\mathbb{C}_{3,0,0}$ and for $\mathbb{P}^3$ the 4–D geometric algebra $\mathbb{C}_{1,3,0}$. The reader should see chapter 14 for more details about the connection of geometric algebra and projective geometry. Any geometric object of $\mathbb{P}^3$ will be linearly projective mapped to $\mathbb{P}^2$ via a projective transformation, for example the projective mapping of a quadric at infinity in the projective space $\mathbb{P}^3$ results in a conic in the projective plane $\mathbb{P}^2$.

Let us first consider a pencil of lines lying on the plane. Doing that, we will follow the ideas of Hestenes and Ziegler [118]. Any pencil of lines is well defined by a bivector addition of two of its lines: $\boldsymbol{l} = \boldsymbol{l}_a + s\boldsymbol{l}_b$ with $s \in \mathbb{R} \cup \{-\infty, +\infty\}$. If two pencils of lines, $\boldsymbol{l}$ and $\boldsymbol{l}' = \boldsymbol{l}'_a + s'\boldsymbol{l}'_b$, can be related one–to–one so that $\boldsymbol{l} = \boldsymbol{l}'$ for $s = s'$, we can say that they are in projective correspondence. Using this idea, the set of intersecting points of lines in correspondence build a conic. Since the intersecting points $\boldsymbol{x}$ of the line pencils $\boldsymbol{l}$ and $\boldsymbol{l}'$ fulfill for $s = s'$ the following constraints

$$\begin{aligned} \boldsymbol{x}\wedge\boldsymbol{l} &= \boldsymbol{x}\wedge\boldsymbol{l}_a + s\boldsymbol{x}\wedge\boldsymbol{l}_b = 0 \\ \boldsymbol{x}\wedge\boldsymbol{l}' &= \boldsymbol{x}\wedge\boldsymbol{l}'_a + s\boldsymbol{x}\wedge\boldsymbol{l}'_b = 0, \end{aligned} \tag{16.1}$$

the elimination of the scalar s yields a second order geometric product equation in $\boldsymbol{x}$

$$(\boldsymbol{x}\wedge\boldsymbol{l}_a)(\boldsymbol{x}\wedge\boldsymbol{l}'_b) - (\boldsymbol{x}\wedge\boldsymbol{l}_b)(\boldsymbol{x}\wedge\boldsymbol{l}'_a) = 0. \tag{16.2}$$

We can also get the parameterized conic equation simply by computing the intersecting point $\boldsymbol{x}$, taking the meet of the line pencils as follows

$$\boldsymbol{x} = (\boldsymbol{l}_a + s\boldsymbol{l}_b) \vee (\boldsymbol{l}'_a + s\boldsymbol{l}'_b) = \boldsymbol{l}_a \vee \boldsymbol{l}'_a + s(\boldsymbol{l}_a \vee \boldsymbol{l}'_b + \boldsymbol{l}_b \vee \boldsymbol{l}'_a) + s^2(\boldsymbol{l}_b \vee \boldsymbol{l}'_b). \tag{16.3}$$

Let us for now define the involved lines in terms of wedge of points $\boldsymbol{l}_a = \boldsymbol{a}\wedge\boldsymbol{b}$, $\boldsymbol{l}_b = \boldsymbol{a}\wedge\boldsymbol{b}'$, $\boldsymbol{l}'_a = \boldsymbol{a}'\wedge\boldsymbol{b}$ and $\boldsymbol{l}'_b = \boldsymbol{a}'\wedge\boldsymbol{b}'$ such that $\boldsymbol{l}_a \vee \boldsymbol{l}'_a = \boldsymbol{b}$, $\boldsymbol{l}_a \vee \boldsymbol{l}'_b = \boldsymbol{d}$, $\boldsymbol{l}_b \vee \boldsymbol{l}'_a = \boldsymbol{d}'$ and $\boldsymbol{l}_b \vee \boldsymbol{l}'_b = \boldsymbol{b}'$, see Figure 16.1.a. By substituting $\boldsymbol{b}'' = \boldsymbol{l}_a \vee \boldsymbol{l}'_b + \boldsymbol{l}_b \vee \boldsymbol{l}'_a = \boldsymbol{d} + \boldsymbol{d}'$ in the last equation, we get

$$\boldsymbol{x} = \boldsymbol{b} + s\boldsymbol{b}'' + s^2\boldsymbol{b}', \tag{16.4}$$

which represents a nondegenerated conic for $\boldsymbol{b}\wedge\boldsymbol{b}''\wedge\boldsymbol{b}' = \boldsymbol{b}\wedge(\boldsymbol{d} + \boldsymbol{d}')\wedge\boldsymbol{b}' \neq 0$. Now, using this equation let us compute the generating line pencils. Define $\boldsymbol{l}_1 = \boldsymbol{b}''\wedge\boldsymbol{b}'$, $\boldsymbol{l}_2 = \boldsymbol{b}'\wedge\boldsymbol{b}$ and $\boldsymbol{l}_3 = \boldsymbol{b}\wedge\boldsymbol{b}''$. Then using the equation (16.4), its two projective pencils are

$$\begin{aligned} \boldsymbol{b}\wedge\boldsymbol{x} &= s\boldsymbol{b}\wedge\boldsymbol{b}'' + s^2\boldsymbol{b}\wedge\boldsymbol{b}' = s(\boldsymbol{l}_3 - s\boldsymbol{l}_2) \\ \boldsymbol{b}'\wedge\boldsymbol{x} &= \boldsymbol{b}'\wedge\boldsymbol{b} + s\boldsymbol{b}'\wedge\boldsymbol{b}'' = \boldsymbol{l}_2 - s\boldsymbol{l}_1. \end{aligned} \tag{16.5}$$

Considering the points $\boldsymbol{a}$, $\boldsymbol{a}'$, $\boldsymbol{b}$ and $\boldsymbol{b}'$ and some other point $\boldsymbol{c}'$ lying on the conic depicted in Figure 16.1.a, and the equation (16.2) for $s = \rho s'$ slightly different to s', we get the bracket expression

$$\begin{aligned} &[\boldsymbol{c}'\boldsymbol{a}\boldsymbol{b}][\boldsymbol{c}'\boldsymbol{a}'\boldsymbol{b}'] - \rho[\boldsymbol{c}'\boldsymbol{a}\boldsymbol{b}'][\boldsymbol{c}'\boldsymbol{a}'\boldsymbol{b}] = 0 \\ &\qquad \Leftrightarrow \rho = \frac{[\boldsymbol{c}'\boldsymbol{a}\boldsymbol{b}][\boldsymbol{c}'\boldsymbol{a}'\boldsymbol{b}']}{[\boldsymbol{c}'\boldsymbol{a}\boldsymbol{b}'][\boldsymbol{c}'\boldsymbol{a}'\boldsymbol{b}]} \end{aligned} \tag{16.6}$$

for some $\rho \neq 0$. This equation is well known and represents a projective invariant which has been used quite a lot in real applications of computer vision [177]. For a thorough study of the role of this invariant using brackets of points, lines, bilinearities and the trifocal tensor see Bayro and Lasenby [145, 18]. Now evaluating ρ in terms of some other point $\boldsymbol{c}$ we get a conic equation fully represented in terms of brackets

$$\begin{aligned} &[\boldsymbol{c}\boldsymbol{a}\boldsymbol{b}][\boldsymbol{c}\boldsymbol{a}'\boldsymbol{b}'] - \frac{[\boldsymbol{c}'\boldsymbol{a}\boldsymbol{b}][\boldsymbol{c}'\boldsymbol{a}'\boldsymbol{b}']}{[\boldsymbol{c}'\boldsymbol{a}\boldsymbol{b}'][\boldsymbol{c}'\boldsymbol{a}'\boldsymbol{b}]}[\boldsymbol{c}\boldsymbol{a}\boldsymbol{b}'][\boldsymbol{c}\boldsymbol{a}'\boldsymbol{b}] = 0 \\ &\Leftrightarrow [\boldsymbol{c}\boldsymbol{a}\boldsymbol{b}][\boldsymbol{c}\boldsymbol{a}'\boldsymbol{b}'][\boldsymbol{a}\boldsymbol{b}'\boldsymbol{c}'][\boldsymbol{a}'\boldsymbol{b}\boldsymbol{c}'] - [\boldsymbol{c}\boldsymbol{a}\boldsymbol{b}'][\boldsymbol{c}\boldsymbol{a}'\boldsymbol{b}][\boldsymbol{a}\boldsymbol{b}\boldsymbol{c}'][\boldsymbol{a}'\boldsymbol{b}'\boldsymbol{c}'] = 0. \end{aligned} \tag{16.7}$$

Again we get a well known concept, which says that a conic is uniquely determined by the five points in general position $\boldsymbol{a}$, $\boldsymbol{a}'$, $\boldsymbol{b}$, $\boldsymbol{b}'$ and $\boldsymbol{c}$. Now, considering Figure 16.1.b, we assume six points on the conic and we can identify three collinear intersecting points $\boldsymbol{\alpha}_1$, $\boldsymbol{\alpha}_2$ and $\boldsymbol{\alpha}_3$. Using the collinearity constraint and the lines which belong to pencils in projective correspondence we can write down a very useful equation

$$\begin{aligned} \boldsymbol{\alpha}_1 \wedge \boldsymbol{\alpha}_2 \wedge \boldsymbol{\alpha}_3 &= 0 \\ \Leftrightarrow \Big((\boldsymbol{a}' \wedge \boldsymbol{b}) \vee (\boldsymbol{c}' \wedge \boldsymbol{c}) \Big) \wedge \Big((\boldsymbol{a}' \wedge \boldsymbol{a}) \vee (\boldsymbol{b}' \wedge \boldsymbol{c}) \Big) \wedge \Big((\boldsymbol{c}' \wedge \boldsymbol{a}) \vee (\boldsymbol{b}' \wedge \boldsymbol{b}) \Big) &= 0. \end{aligned} \tag{16.8}$$

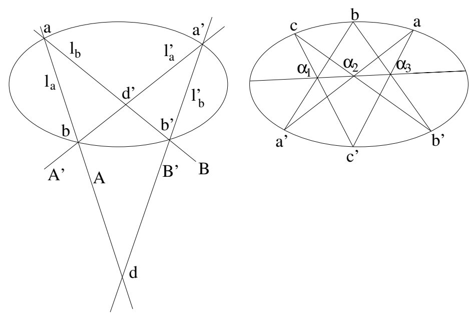

Fig. 16.1. a) Two projective pencils generate a conic b) Pascal's theorem

This expression is a geometric formulation of Pascal's theorem. This theorem proves that the three intersecting points of the lines which connect opposite vertices of a hexagon circumscribed by a conic are collinear ones. The equation (16.8) will be used in later section for computing the intrinsic camera parameters.

16.3 Computing the Kruppa Equations in the Geometric Algebra

In this section we will formulate in two ways the Kruppa equations in the geometric algebra framework. First, we derive the Kruppa equations in its polynomial form using the bracket conic equation (16.7). Secondly, we formulate them in terms of pure brackets. The goal of the section is to compare the bracket representation with the standard one.

16.3.1 The Scenario

Next, we will briefly summarize the scenario for observing a conic at infinity (the absolute conic) in the image planes of multiple views with the aim of self-calibration of the camera. We are applying the standard pinhole camera

model. As described in chapter 14 a pinhole camera can be described by four homogeneous vectors in $\mathbb{P}^3$: One vector gives the optical centre and the other three define the image plane. Let $\{A_\mu\}$ be a reference coordinate system, which consists of four vectors and defines the frame $\mathcal{F}_0$. Let X be a point in a frame $\{Z_\mu\} = \mathcal{F}_1$. The image X_A of the point X on the image plane A of $\{A_\mu\} = \mathcal{F}_0$ can be described by several transformations.
In the first step the frame $\mathcal{F}_1$ can be related to $\mathcal{F}_0$ by a transformation $M_{\mathcal{F}_0}^{\mathcal{F}_1}$. This transformation represents a 3-D rotation R and a 3-D translation $\boldsymbol{t}$ in the 3-D projective space $\mathbb{P}^3$ and depends on six camera parameters. So the frames $\mathcal{F}_0$ and $\mathcal{F}_1$ are first related by a 4×4 matrix

$$M_{\mathcal{F}_0}^{\mathcal{F}_1} = \begin{pmatrix} R & \boldsymbol{t} \\ 0_3^T & 1 \end{pmatrix}. \tag{16.9}$$

The matrix $M_{\mathcal{F}_0}^{\mathcal{F}_1}$ is the matrix of the extrinsic camera parameters.
In the next step changes between the camera planes have to be considered. So the focal length, rotations and translations in the image planes have to be adapted. This affine transformation will be described by the matrix K and has the well known form

$$K = \begin{pmatrix} \alpha_u & \gamma & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{16.10}$$

The parameters u_0, v_0 describe a translation along the image plane and α_u, α_v, γ describe scale changes along the image axes and a rotation in the image plane. So the whole projective transformation can be described by

$$P = KP_0M_{\mathcal{F}_0}^{\mathcal{F}_1}, \tag{16.11}$$

where $P_0 = [I|0]$ is a 3×4 matrix and I is the 3×3 identity matrix. P_0 describes the projection matrix from the 3-D camera frame $\mathcal{F}_1$ to the normalized camera plane, given in homogeneous coordinates.
The task is to find out the intrinsic camera parameters, which can be found in the matrix K (see equation 16.10) of the affine transformation from the normalized camera coordinate plane to the image coordinate plane. As depicted in Figure 16.2, the images of the points defining the absolute conic are observed from different positions and orientations, and the point correspondences between the images are evaluated. Generally, the relation between points of cameras at different locations depends on both, the extrinsic and the intrinsic parameters. But in case of formulating the Kruppa equations, it will happen that these only depend on intrinsic parameters. An often used notation of equation (16.11), which we want to adopt here for the camera at the i-th frame $\mathcal{F}_i$ with respect to frame $\mathcal{F}_0$, is

$$P_i = K[R|\boldsymbol{t}], \tag{16.12}$$

where $[R|\boldsymbol{t}]$ is a 3×4 matrix constituted by the rotation matrix R and the translation vector $\boldsymbol{t}$, resulting from the fusion of P_0 and $M_{\mathcal{F}_0}^{\mathcal{F}_i}$. For the sake of simplicity, we will set for the first camera $\mathcal{F}_1 \equiv \mathcal{F}_0$, thus, its projective transformation becomes $P_1 = K[I|0]$, where I is the 3×3 identity matrix.

16.3.2 Standard Kruppa Equations

This approach uses the equation (16.7) for the conic in terms of brackets considering five points $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{a}'$, $\boldsymbol{b}'$, $\boldsymbol{c}'$ which lie on the conic in the image plane:

$$[\boldsymbol{cab}][\boldsymbol{ca'b'}][\boldsymbol{ab'c'}][\boldsymbol{a'bc'}] - [\boldsymbol{cab'}][\boldsymbol{ca'b}][\boldsymbol{abc'}][\boldsymbol{a'b'c'}] = 0$$
$$[\boldsymbol{abc}][\boldsymbol{a'b'c}] - \frac{[\boldsymbol{a'b'c'}][\boldsymbol{abc'}]}{[\boldsymbol{ab'c'}][\boldsymbol{a'bc'}]}[\boldsymbol{ab'c}][\boldsymbol{a'bc}] = 0. \qquad (16.13)$$

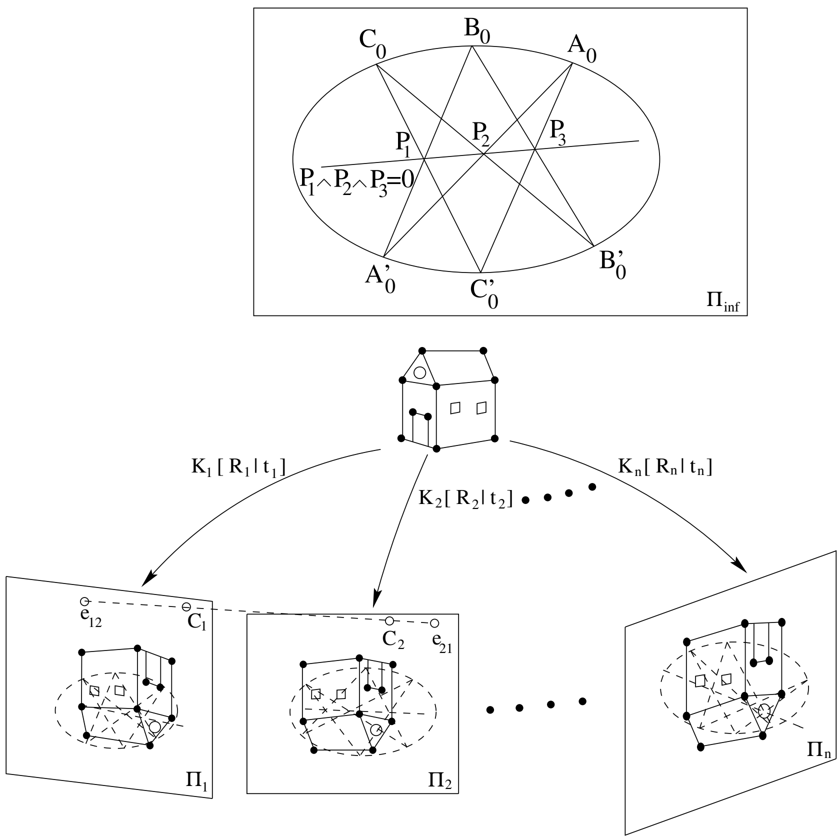

Fig. 16.2. The conics at infinity, the real 3–D visual space and n uncalibrated cameras

These five points are images of points on the absolute conic. A conic at infinity Ω_{inf} in $\mathbb{P}^3$ can be defined employing any imaginary five points lying on the conic, e.g.

$$\boldsymbol{A}_0 = \begin{pmatrix} 1 \\ i \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{B}_0 = \begin{pmatrix} i \\ 1 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{A}'_0 = \begin{pmatrix} i \\ 0 \\ 1 \\ 0 \end{pmatrix}, \boldsymbol{B}'_0 = \begin{pmatrix} 1 \\ 0 \\ i \\ 0 \end{pmatrix}, \boldsymbol{C}'_0 = \begin{pmatrix} 0 \\ i \\ 1 \\ 0 \end{pmatrix}, \quad (16.14)$$

where $i^2 = -1$. Note that we use upper case letters to represent points of the projective space $\mathbb{P}^3$ in $\mathbb{C}_{1,3,0}$. Because these points at infinity fulfill the property $\boldsymbol{A}_0^T \boldsymbol{A}_0 = \boldsymbol{B}_0^T \boldsymbol{B}_0 = \boldsymbol{A}_0^{'T} \boldsymbol{A}'_0 = \boldsymbol{B}_0^{'T} \boldsymbol{B}'_0 = \boldsymbol{C}_0^{'T} \boldsymbol{C}'_0 = 0$ they lie on the absolute conic. In geometric algebra a conic can be described by the points lying on the conic. Furthermore, the image of the absolute conic can be described by the image of the points lying on the absolute conic. In the next step, let us first define the point $\boldsymbol{A}$ as a 3×1-vector which consists of the first three elements of $\boldsymbol{A}_0$. Doing similary with the other points we get the points

$$\boldsymbol{A} = \begin{pmatrix} 1 \\ i \\ 0 \end{pmatrix}, \boldsymbol{B} = \begin{pmatrix} i \\ 1 \\ 0 \end{pmatrix}, \boldsymbol{A}' = \begin{pmatrix} i \\ 0 \\ 1 \end{pmatrix}, \boldsymbol{B}' = \begin{pmatrix} 1 \\ 0 \\ i \end{pmatrix}, \boldsymbol{C}' = \begin{pmatrix} 0 \\ i \\ 1 \end{pmatrix}. \quad (16.15)$$

Since the projection of the points $\boldsymbol{A}_0, \ldots, \boldsymbol{C}'_0$ are translation invariant, their projections $\boldsymbol{x} = P\boldsymbol{X}$ on any image plane are independent of $\boldsymbol{t}$ and thus given by

$$\begin{aligned} &\boldsymbol{a} = K[R|\boldsymbol{t}]\boldsymbol{A}_0 = KR\boldsymbol{A}\ , \ \boldsymbol{b} = K[R|\boldsymbol{t}]\boldsymbol{B}_0 = KR\boldsymbol{B} \\ &\boldsymbol{a}' = K[R|\boldsymbol{t}]\boldsymbol{A}'_0 = KR\boldsymbol{A}'\ , \ \boldsymbol{b}' = K[R|\boldsymbol{t}]\boldsymbol{B}'_0 = KR\boldsymbol{B}' \\ &\boldsymbol{c}' = K[R|\boldsymbol{t}]\boldsymbol{C}'_0 = KR\boldsymbol{C}'\ . \end{aligned} \quad (16.16)$$

In addition the rotated points $R^T\boldsymbol{A}$, $R^T\boldsymbol{B}$, $R^T\boldsymbol{A}'$, $R^T\boldsymbol{B}'$ and $R^T\boldsymbol{C}'$ lie also at the conic, because they fulfill the property

$$\begin{aligned} &(R^T\boldsymbol{A})^T(R^T\boldsymbol{A}) = (R^T\boldsymbol{B})^T(R^T\boldsymbol{B}) = (R^T\boldsymbol{A}')^T(R^T\boldsymbol{A}') = \\ &\qquad (R^T\boldsymbol{B}')^T(R^T\boldsymbol{B}') = (R^T\boldsymbol{C}')^T(R^T\boldsymbol{C}') = 0. \end{aligned} \quad (16.17)$$

Using these rotated points, the rotation R of the camera transformation is canceled and the points on the image of the absolute conic will be described by

$$\boldsymbol{a} = K\boldsymbol{A}, \quad \boldsymbol{b} = K\boldsymbol{B}, \quad \boldsymbol{a}' = K\boldsymbol{A}', \quad \boldsymbol{b}' = K\boldsymbol{B}', \quad \boldsymbol{c}' = K\boldsymbol{C}'. \quad (16.18)$$

To use the points $\boldsymbol{a}, \ldots, \boldsymbol{c}'$ in the bracket notation of conics it is usefull to translate the matrix multiplication $\boldsymbol{x} = K\boldsymbol{X}$ in terms of geometric algebra. Suppose an orthonormal basis $B_1 = \{\mathbf{e}_1, \ldots, \mathbf{e}_3\}$ and $\boldsymbol{X}$ as a linear combination of B_1 i.e. $\boldsymbol{X} = \sum_{i=1}^{3} x_i \mathbf{e}_i$. The matrix K describes a linear transformation. As can be seen in chapter 1.3 this linear transformation can be expressed by

$$\underline{K}\mathbf{e}_i = \underline{K}(\mathbf{e}_i) = \sum_{j=1}^{3} \mathbf{e}_j k_{ji} \tag{16.19}$$

with k_{ji} the elements of the matrix K. So the matrix multiplication $K\boldsymbol{X}$ can be substituted by $\underline{K}\boldsymbol{X}$ in terms of geometric algebra. Therefore, the point $\boldsymbol{c}$ lies on the image of the absolute conic iff

$$[(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B})\boldsymbol{c}][(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{B}')\boldsymbol{c}] - \frac{[(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{B}')(\underline{K}\boldsymbol{C}')][(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B})(\underline{K}\boldsymbol{C}')]}{[(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B}')(\underline{K}\boldsymbol{C}')][(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{B})(\underline{K}\boldsymbol{C}')]} \cdot \\ \cdot [(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B}')\boldsymbol{c}][(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{B})\boldsymbol{c}] = 0. \tag{16.20}$$

We can further extract of the brackets the determinant of the intrinsic parameters in the multiplicative ratio of the previous equation. This is explained in chapter 1.3. Now the invariant reduces to a constant

$$\begin{aligned} Inv &= \frac{([\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{B}')(\underline{K}\boldsymbol{C}')][(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B})(\underline{K}\boldsymbol{C}')]}{[(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B}')(\underline{K}\boldsymbol{C}'))][(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{B})(K\boldsymbol{C}')]} \\ &= \frac{det(\underline{K})[\boldsymbol{A}'\boldsymbol{B}'\boldsymbol{C}']det(\underline{K})[\boldsymbol{A}\boldsymbol{B}\boldsymbol{C}']}{det(\underline{K})[\boldsymbol{A}\boldsymbol{B}'\boldsymbol{C}']det(\underline{K})[\boldsymbol{A}'\boldsymbol{B}\boldsymbol{C}']} \\ &= \frac{[\boldsymbol{A}'\boldsymbol{B}'\boldsymbol{C}'][\boldsymbol{A}\boldsymbol{B}\boldsymbol{C}']}{[\boldsymbol{A}\boldsymbol{B}'\boldsymbol{C}'][\boldsymbol{A}'\boldsymbol{B}\boldsymbol{C}']}. \end{aligned} \tag{16.21}$$

Substituting the values from equation (16.15) for $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{A}'$, $\boldsymbol{B}'$, $\boldsymbol{C}'$ in this equation, we get the value of $Inv = 2$. This value will be used for further computations later on. The equation (16.21) is as expected invariant to the affine transformation $\underline{K}$. Thus, the bracket equation (16.6) of the projective invariant resulting in the image of the absolute conic can be written as

$$[(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B})\boldsymbol{c}][(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{B}')\boldsymbol{c}] - Inv[(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B}')\boldsymbol{c}][(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{B})\boldsymbol{c}] = 0. \tag{16.22}$$

Let be $Q = K^{-T}K^{-1}$ the matrix of the image of the absolute conic, then $\boldsymbol{c}^T Q \boldsymbol{c} = 0$ in matrix notation means that $\boldsymbol{c}$ is a point on the image of the absolute conic. According to the duality principle of points and lines the dual image of the absolute conic, i.e. its matrix $Q^* \sim Q^{-1} = KK^T$ is related to a line $\boldsymbol{l}_c$, tangential to the image of the absolute conic. Because this can be expressed as

$$0 = \boldsymbol{c}^T Q \boldsymbol{c} = \boldsymbol{c}^T Q^T \boldsymbol{c} = (\boldsymbol{c}^T Q^T) Q^{-1} (Q\boldsymbol{c}) = \boldsymbol{l}_c^T Q^* \boldsymbol{l}_c, \tag{16.23}$$

we have $Q\boldsymbol{c} = \boldsymbol{l}_c$ or $\boldsymbol{c} = KK^T\boldsymbol{l}_c$. To use $KK^T\boldsymbol{l}_c$ in the bracket description of conics, it is usefull to translate the matrix multiplications in terms of geometric algebra. The line $\boldsymbol{l}_c$ is tangential to the image of the absolute conic, so it has the form $\sum_{i=1}^{3} l_{c_i} \mathbf{e}_i$. The product $K^T\boldsymbol{l}_c$ can be described using the adjoint $\overline{K}$ of $\underline{K}$ by the expression $\overline{K}\boldsymbol{l}_c$, see chapter 1.3. The expression $\boldsymbol{c} = KK^T\boldsymbol{l}_c$ can thus be formulated as $\boldsymbol{c} = \underline{K}\overline{K}\boldsymbol{l}_c$. We can substitute this line tangent in equation (16.22):

$$
\begin{aligned}
&[(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B})\boldsymbol{c}][(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{B}')\boldsymbol{c}] - Inv[(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B}')\boldsymbol{c}][(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{B})\boldsymbol{c}] = 0\\
\Leftrightarrow\ &[(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B})(\underline{K}\overline{K}\boldsymbol{l}_c)][(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{B}')(\underline{K}\overline{K}\boldsymbol{l}_c)] -\\
&\quad -Inv[(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B}')(\underline{K}\overline{K}\boldsymbol{l}_c)][(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{B})(\underline{K}\overline{K}\boldsymbol{l}_c)] = 0\\
\Leftrightarrow\ &det(\underline{K})[\boldsymbol{AB}(\overline{K}\boldsymbol{l}_c)]det(\underline{K})[\boldsymbol{A}'\boldsymbol{B}'(\overline{K}\boldsymbol{l}_c)] -\\
&\quad -Inv\,det(\underline{K})[\boldsymbol{AB}'(\overline{K}\boldsymbol{l}_c)]det(\underline{K})[\boldsymbol{A}'\boldsymbol{B}(\overline{K}\boldsymbol{l}_c)] = 0\\
\Leftrightarrow\ &[\boldsymbol{AB}(\overline{K}\boldsymbol{l}_c)][\boldsymbol{A}'\boldsymbol{B}'(\overline{K}\boldsymbol{l}_c)] - Inv[\boldsymbol{AB}'(\overline{K}\boldsymbol{l}_c)][\boldsymbol{A}'\boldsymbol{B}(\overline{K}\boldsymbol{l}_c)] = 0. \qquad (16.24)
\end{aligned}
$$

To further proceed on the classical way of deriving Kruppa's equations [169, 163, 162], it will be possible to formulate two polynomial constraint equations on the dual of the image of the absolute conic in the frame of epipolar geometry. Let be $\boldsymbol{p} = p_1\mathbf{e}_1 + p_2\mathbf{e}_2 + p_3\mathbf{e}_3$ the epipole of an image and let be $\boldsymbol{q} = \mathbf{e}_1 + \tau\mathbf{e}_2$ a point at infinity. The aim will be to force the line

$$
\begin{aligned}
\boldsymbol{l}_c &= (\boldsymbol{p}\wedge\boldsymbol{q})\boldsymbol{I}^{-1}\\
&= \left(\left(\sum_{i=1}^{3} p_i\mathbf{e}_i\right)\wedge(\mathbf{e}_1+\tau\mathbf{e}_2)\right)(\mathbf{e}_1\mathbf{e}_2\mathbf{e}_3)^{-1}\\
&= (-p_3\tau)\mathbf{e}_1 + (p_3)\mathbf{e}_2 + (p_1\tau - p_2)\mathbf{e}_3, \qquad (16.25)
\end{aligned}
$$

to be tangential to the dual of the image of the absolute conic by means of the unknown τ. Then we can substitute the term $\boldsymbol{l}_c$ in equation (16.24). With

$$
\begin{aligned}
\overline{K}\boldsymbol{l}_c = (-k_{11}p_3\tau)\mathbf{e}_1 + (-k_{12}p_3\tau + k_{22}p_3)\mathbf{e}_2 +\\
(-k_{13}p_3\tau + k_{23}p_3 + p_1\tau - p_2)\mathbf{e}_3 \qquad (16.26)
\end{aligned}
$$

and the value for $Inv = 2$ the equation (16.24) simplifies to a second order polynomial with respect to τ as follows

$$
\begin{aligned}
&[\boldsymbol{AB}(\overline{K}\boldsymbol{l}_c)][\boldsymbol{A}'\boldsymbol{B}'(\overline{K}\boldsymbol{l}_c)] - Inv[\boldsymbol{AB}'(\overline{K}\boldsymbol{l}_c)][\boldsymbol{A}'\boldsymbol{B}(\overline{K}\boldsymbol{l}_c)] =\\
&4p_1\tau p_2 - 2p_1^2\tau^2 - 2k_{22}^2p_3^2 - 4k_{23}p_3p_1\tau + 4k_{23}p_3p_2 - 2k_{13}^2p_3^2\tau^2 -\\
&2k_{12}^2p_3^2\tau^2 - 2k_{23}^2p_3^2 - 2p_2^2 - 2k_{11}^2p_3^2\tau^2 + 4k_{12}p_3^2\tau k_{22} - 4k_{13}p_3\tau p_2 +\\
&4k_{13}p_3^2\tau k_{23} + 4k_{13}p_3\tau^2 p_1. \qquad (16.27)
\end{aligned}
$$

Expressing the polynomial in the form $P(\tau) = k_0 + k_1\tau + k_2\tau^2$, we get the following coefficients

$$
\begin{aligned}
k_0 &= -2k_{22}^2p_3^2 + 4k_{23}p_3p_2 - 2k_{23}^2p_3^2 - 2p_2^2\\
k_1 &= 4p_1p_2 - 4k_{23}p_3p_1 + 4k_{12}p_3^2k_{22} - 4k_{13}p_3p_2 + 4k_{13}p_3^2k_{23}\\
k_2 &= -2p_1^2 - 2k_{13}^2p_3^2 - 2k_{12}^2p_3^2 - 2k_{11}^2p_3^2 + 4k_{13}p_3p_1. \qquad (16.28)
\end{aligned}
$$

Because $\boldsymbol{l}_c$ can be also considered as an epipolar line tangent to the conic in the first camera, according the homography of a point lying at the line at infinity of the second camera, we can use the operator $\underline{F}$ for the describtion of the fundamental matrix F in terms of geometric algebra, and can compute $\boldsymbol{l}_c = \underline{F}(\mathbf{e}_1 + \tau\mathbf{e}_2)$. Using the new expression of $\boldsymbol{l}_c$ we can gain similarly as above new equations for the coefficients of the polynomial $P(\tau)$, now called k_i'. Taking now these equations for the two cameras, we finally can write down the well known Kruppa equations

$$\begin{aligned} k_2k_1' - k_2'k_1 &= 0 \\ k_0k_1' - k_0'k_1 &= 0 \\ k_0k_2' - k_0'k_2 &= 0. \end{aligned} \tag{16.29}$$

We get up to a scalar factor the same Kruppa equations as presented by Luong and Faugeras [162]. The scalar factor is present in all of these equations, thus it can be canceled straightforwardly. The algebraic manipulation of this formulas was checked entirely using a Maple program.

16.3.3 Kruppa's Equations Using Brackets

In this section we will formulate the Kruppa coefficients k_0, k_1, k_2 of the polynomial $P(\tau)$ in terms of brackets. This kind of representation will obviously elucidate the involved geometry. First let us consider again the bracket $[\boldsymbol{AB}(\overline{K}\boldsymbol{l}_c)]$ of equation (16.24). Each bracket can be split in two brackets, one independent of τ and another depending of it

$$[\boldsymbol{AB}(\overline{K}\boldsymbol{l}_c)] = [\boldsymbol{AB}(\overline{K}(p_3\mathbf{e}_2 - p_2\mathbf{e}_3))] + [\boldsymbol{AB}(\overline{K}(-p_3\mathbf{e}_1 + p_1\mathbf{e}_3))]\tau. \tag{16.30}$$

In short, $[\boldsymbol{AB}(\overline{K}\boldsymbol{l}_c)] = a_1 + \tau b_1$. Now using this bracket representation the equation (16.24) can be written as

$$\begin{aligned} & [\boldsymbol{AB}(\overline{K}\boldsymbol{l}_c)][\boldsymbol{A'B'}(\overline{K}\boldsymbol{l}_c)] - Inv[\boldsymbol{AB'}(\overline{K}\boldsymbol{l}_c)][\boldsymbol{A'B}(\overline{K}\boldsymbol{l}_c)] = 0 \\ \Leftrightarrow\ & (a_1 + \tau b_1)(a_2 + \tau b_2) - Inv(a_3 + \tau b_3)(a_4 + \tau b_4) = 0 \\ \Leftrightarrow\ & a_1a_2 + \tau b_1a_2 + a_1\tau b_2 + \tau^2 b_1b_2 - \\ & -Inv(a_3a_4 + a_3a_4\tau + b_3a_4\tau + b_3b_4\tau^2) = 0 \\ \Leftrightarrow\ & \underbrace{a_1a_2 - Inv(a_3a_4)}_{k_0} + \tau(\underbrace{a_1b_2 + b_1a_2 - Inv(a_3b_4 + a_4b_3)}_{k_1}) + \\ & +\tau^2(\underbrace{b_1b_2 - Inv(b_3b_4)}_{k_2}) = 0. \end{aligned} \tag{16.31}$$

Now let us take a partial vector part of $\overline{K}\boldsymbol{l}_c$ and call it

$$\overline{K}\boldsymbol{l}_{c1} := -k_{11}p_3\mathbf{e}_1 - k_{12}p_3\mathbf{e}_2 + (-k_{13}p_3 + p_1)\mathbf{e}_3$$

and the "rest"-part as

$$\overline{K}\boldsymbol{l}_{c2} := (k_{22}p_3)\mathbf{e}_2 + (k_{23}p_3 - p_2)\mathbf{e}_3.$$

Using both parts we can write the coefficients of the polynomial in a bracket form as follows:

$$k_0 = [\boldsymbol{AB}(\overline{K}\boldsymbol{l}_{c2})][\boldsymbol{A}'\boldsymbol{B}'(\overline{K}\boldsymbol{l}_{c2})] - Inv[\boldsymbol{AB}'(\overline{K}\boldsymbol{l}_{c2})][\boldsymbol{A}'\boldsymbol{B}(\overline{K}\boldsymbol{l}_{c2})] \qquad (16.32)$$

$$\begin{aligned} k_1 = {} & [\boldsymbol{AB}(\overline{K}\boldsymbol{l}_{c1})][\boldsymbol{A}'\boldsymbol{B}'(\overline{K}\boldsymbol{l}_{c2})] + [\boldsymbol{AB}(\overline{K}\boldsymbol{l}_{c2})][\boldsymbol{A}'\boldsymbol{B}'(\overline{K}\boldsymbol{l}_{c1})] \\ & -Inv[\boldsymbol{AB}'(\overline{K}\boldsymbol{l}_{c2})][\boldsymbol{A}'\boldsymbol{B}(\overline{K}\boldsymbol{l}_{c1})] - Inv[\boldsymbol{AB}'(\overline{K}\boldsymbol{l}_{c1})][\boldsymbol{A}'\boldsymbol{B}(\overline{K}\boldsymbol{l}_{c2})] \end{aligned} \qquad (16.33)$$

$$k_2 = [\boldsymbol{AB}(\overline{K}\boldsymbol{l}_{c1})][\boldsymbol{A}'\boldsymbol{B}'(\overline{K}\boldsymbol{l}_{c1})] - Inv[\boldsymbol{AB}'(\overline{K}\boldsymbol{l}_{c1})][\boldsymbol{A}'\boldsymbol{B}(\overline{K}\boldsymbol{l}_{c1})]. \qquad (16.34)$$

Since $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{A}', \boldsymbol{B}'$ and Inv are known given an epipole $\boldsymbol{p} = p_1\mathbf{e}_1 + p_2\mathbf{e}_2 + p_3\mathbf{e}_3$, we can finally compute the coefficients k_0, k_1, k_2 straightforwardly. The striking aspect of these equations is twofold. They are expressed in terms of brackets and they depend of the invariant real magnitude Inv. This can certainly help us to explore the involved geometry of the Kruppa equations using brackets.

Let us first analyze the k's. Since the elements of k_1 consists of the elements of k_0 and k_2, it should be sufficient to explore the involved geometry of k_0 and k_2 if these are expressed as follows:

$$\begin{aligned} k_0 = {} & a_1a_2 - Inv(a_3a_4) \\ = {} & [\boldsymbol{AB}(\overline{K}\boldsymbol{l}_{c2})][\boldsymbol{A}'\boldsymbol{B}'(\overline{K}\boldsymbol{l}_{c2})] - Inv[\boldsymbol{AB}'(\overline{K}\boldsymbol{l}_{c2})][\boldsymbol{A}'\boldsymbol{B}(\overline{K}\boldsymbol{l}_{c2})] \\ = {} & ((\mathbf{e}_1 + i\mathbf{e}_2) \wedge (i\mathbf{e}_1 + \mathbf{e}_2) \wedge (k_{22}p_3\mathbf{e}_2 + (k_{23}p_3 - p_2)\mathbf{e}_3)\boldsymbol{I}^{-1}) \\ & ((i\mathbf{e}_1 + \mathbf{e}_3) \wedge (\mathbf{e}_1 + i\mathbf{e}_3) \wedge (k_{22}p_3\mathbf{e}_2 + (k_{23}p_3 - p_2)\mathbf{e}_3)\boldsymbol{I}^{-1}) - \\ & Inv((\mathbf{e}_1 + i\mathbf{e}_2) \wedge (\mathbf{e}_1 + i\mathbf{e}_3) \wedge (k_{22}p_3\mathbf{e}_2 + (k_{23}p_3 - p_2)\mathbf{e}_3)\boldsymbol{I}^{-1}) \\ & ((i\mathbf{e}_1 + \mathbf{e}_3) \wedge (i\mathbf{e}_1 + \mathbf{e}_2) \wedge (k_{22}p_3\mathbf{e}_2 + (k_{23}p_3 - p_2)\mathbf{e}_3)\boldsymbol{I}^{-1}) \end{aligned} \qquad (16.35)$$

$$\begin{aligned} k_2 = {} & b_1b_2 - Inv(b_3b_4) \\ = {} & [\boldsymbol{AB}(\overline{K}\boldsymbol{l}_{c1})][\boldsymbol{A}'\boldsymbol{B}'(\overline{K}\boldsymbol{l}_{c1})] - Inv[\boldsymbol{AB}'(\overline{K}\boldsymbol{l}_{c1})][\boldsymbol{A}'\boldsymbol{B}(\overline{K}\boldsymbol{l}_{c1})] \\ & ((\mathbf{e}_1 + i\mathbf{e}_2) \wedge (i\mathbf{e}_1 + \mathbf{e}_2) \wedge \\ & \quad (-k_{11}p_3\mathbf{e}_1 - k_{12}p_3\mathbf{e}_2 + (-k_{13}p_3 + p_1)\mathbf{e}_3)\boldsymbol{I}^{-1}) \\ & ((i\mathbf{e}_1 + \mathbf{e}_3) \wedge (\mathbf{e}_1 + i\mathbf{e}_3) \wedge \\ & \quad (-k_{11}p_3\mathbf{e}_1 - k_{12}p_3\mathbf{e}_2 + (-k_{13}p_3 + p_1)\mathbf{e}_3)\boldsymbol{I}^{-1}) \\ & -Inv((\mathbf{e}_1 + i\mathbf{e}_2) \wedge (\mathbf{e}_1 + i\mathbf{e}_3) \wedge \\ & \quad (-k_{11}p_3\mathbf{e}_1 - k_{12}p_3\mathbf{e}_2 + (-k_{13}p_3 + p_1)\mathbf{e}_3)\boldsymbol{I}^{-1}) \\ & ((i\mathbf{e}_1 + \mathbf{e}_3) \wedge (i\mathbf{e}_1 + \mathbf{e}_2) \wedge \\ & \quad (-k_{11}p_3\mathbf{e}_1 - k_{12}p_3\mathbf{e}_2 + (-k_{13}p_3 + p_1)\mathbf{e}_3)\boldsymbol{I}^{-1}). \end{aligned} \qquad (16.36)$$

Let us analyze some effects of camera motions in these two equations. If the camera moves on a straight path parallel to the object, the epipole lies at infinity. Because $p_3 = 0$ in this case, the intrinsic parameters become zero resulting a trivial polynomial, i.e. we can not get the coefficients of the intrinsic camera parameters. On the other hand, for example trying the values $-k_{13}p_3 + p_1 = 0$ or $k_{23}p_3 - p_2 = 0$, the rest of the brackets will have the rank two and their determinant value is also zero. Since the epipole can be normalized with $p_3 = 1$, the equations are equivalent to $k_{13} = p_1$ and $k_{23} = p_2$. This means there is a superposition of the value of the epipole with a parameter of the intrinsic camera parameters. These simple examples show that analyzing the brackets for certain kinds of camera motions can avoid certain camera motions which generate trivial Kruppa equations. It is also interesting to see that for $k_0 = 0$ and $k_2 = 0$ we have also conic equations. So in order to avoid trivial equations we have to consider always $k_0 \neq 0$ and $k_2 \neq 0$. In other words, the splitted parts $\overline{K}\boldsymbol{l}_{c1}$ and $\overline{K}\boldsymbol{l}_{c2}$ of $\overline{K}\boldsymbol{l}_c$ should not lie on the image of the absolute conic.

Now let us consider the invariant real magnitude Inv of the bracket equation (16.24).

$$\begin{aligned} & [\boldsymbol{AB}(\overline{K}\boldsymbol{l}_c)][\boldsymbol{A'B'}(\overline{K}\boldsymbol{l}_c)] - Inv[\boldsymbol{AB'}(\overline{K}\boldsymbol{l}_c)][\boldsymbol{A'B}(\overline{K}\boldsymbol{l}_c)] = 0 \\ \Leftrightarrow\ & Inv = \frac{[\boldsymbol{AB}(\overline{K}\boldsymbol{l}_c)][\boldsymbol{A'B'}(\overline{K}\boldsymbol{l}_c)]}{[\boldsymbol{AB'}(\overline{K}\boldsymbol{l}_c)][\boldsymbol{A'B}(\overline{K}\boldsymbol{l}_c)]}. \end{aligned} \tag{16.37}$$

That the invariant value Inv like in the equation (16.6) plays a role in the Kruppa equations is a fact that has been overseen so far. This can be simply explained as the fact that when we formulate the Kruppa equations using the condition $\boldsymbol{c}^T Q \boldsymbol{c} = 0$, we are actually implicitly employing the invariant given by equation (16.37).

16.4 Camera Calibration Using Pascal's Theorem

This section presents a new technique in the geometric algebra framework for computing the intrinsic camera parameters. The previous section used the equation of (16.7) to compute the Kruppa coefficients which in turn can be used to get the intrinsic camera parameters. Along this lines we will proceed here.

In section two it is shown that the equation (16.7) can be reformulated to express the constraint of equation (16.8) known as Pascal's theorem. Since Pascal's theorem fulfills a property of any conic, it should be also possible using this equation to compute the intrinsic camera parameters. Let us consider the three intersecting points which are collinear and fulfill

$$\underbrace{((\boldsymbol{a}'\wedge\boldsymbol{b})\vee(\boldsymbol{c}'\wedge\boldsymbol{c}))}_{\boldsymbol{\alpha}_1}\wedge\underbrace{((\boldsymbol{a}'\wedge\boldsymbol{a})\vee(\boldsymbol{b}'\wedge\boldsymbol{c}))}_{\boldsymbol{\alpha}_2}\wedge\underbrace{((\boldsymbol{c}'\wedge\boldsymbol{a})\vee(\boldsymbol{b}'\wedge\boldsymbol{b}))}_{\boldsymbol{\alpha}_3} = 0. \tag{16.38}$$

Similar to chapter 1.3.2, in Figure 16.3 at the first camera the projected rotated points of the conic at infinity are

$$\boldsymbol{a} = \underline{K}\boldsymbol{A}, \quad \boldsymbol{b} = \underline{K}\boldsymbol{B}, \quad a' = \underline{K}\boldsymbol{A}', \quad \boldsymbol{b}' = \underline{K}\boldsymbol{B}', \quad \boldsymbol{c}' = \underline{K}\boldsymbol{C}'. \tag{16.39}$$

The point $\boldsymbol{c} = \underline{K}\overline{K}\boldsymbol{l}_c$ depends of the intrinsic parameters and of the line $\boldsymbol{l}_c$ tangent to the conic which is computed in terms of the epipole $\boldsymbol{p} = p_1\mathbf{e}_1 + p_2\mathbf{e}_2 + p_3\mathbf{e}_3$ and a point $\boldsymbol{q} = \mathbf{e}_1 + \tau\mathbf{e}_2$ lying at the line at infinity of the first camera, i.e. $\boldsymbol{l}_c = (\boldsymbol{p} \wedge \boldsymbol{q})\boldsymbol{I}^{-1}$.

Now using this expression for $\boldsymbol{l}_c$ we can simplify equation (16.38) and get the bracket equations of the $\boldsymbol{\alpha}$'s

$$\begin{aligned}
&\big([\boldsymbol{a}'\boldsymbol{b}\boldsymbol{c}']\boldsymbol{c} - [\boldsymbol{a}'\boldsymbol{b}\boldsymbol{c}]\boldsymbol{c}'\big)\wedge\big([\boldsymbol{a}'\boldsymbol{a}\boldsymbol{b}']\boldsymbol{c} - [\boldsymbol{a}'\boldsymbol{a}\boldsymbol{c}]\boldsymbol{b}'\big)\wedge\big([\boldsymbol{c}'\boldsymbol{a}\boldsymbol{b}']\boldsymbol{b} - [\boldsymbol{c}'\boldsymbol{a}\boldsymbol{b}]\boldsymbol{b}'\big) = 0 \\
\Leftrightarrow\ &\big([(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{B})(\underline{K}\boldsymbol{C}')](\underline{K}\overline{K}\boldsymbol{l}_c) - [(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{B})(\underline{K}\overline{K}\boldsymbol{l}_c)](\underline{K}\boldsymbol{C}')\big)\wedge \\
&\big([(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B}')](\underline{K}\overline{K}\boldsymbol{l}_c) - [(\underline{K}\boldsymbol{A}')(\underline{K}\boldsymbol{A})(\underline{K}\overline{K}\boldsymbol{l}_c)](\underline{K}\boldsymbol{B}')\big)\wedge \\
&\big([(\underline{K}\boldsymbol{C}')(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B}')](\underline{K}\boldsymbol{B}) - [(\underline{K}\boldsymbol{C}')(\underline{K}\boldsymbol{A})(\underline{K}\boldsymbol{B})](\underline{K}\boldsymbol{B}')\big) = 0 \\
\Leftrightarrow\ &\Big(det(\underline{K})\underline{K}\big([\boldsymbol{A}'\boldsymbol{B}\boldsymbol{C}'](\overline{K}\boldsymbol{l}_c) - [\boldsymbol{A}'\boldsymbol{B}(\overline{K}\boldsymbol{l}_c)]\boldsymbol{C}'\big)\Big)\wedge \\
&\Big(det(\underline{K})\underline{K}\big([\boldsymbol{A}'\boldsymbol{A}\boldsymbol{B}'](\overline{K}\boldsymbol{l}_c) - [\boldsymbol{A}'\boldsymbol{A}(\overline{K}\boldsymbol{l}_c)]\boldsymbol{B}'\big)\Big)\wedge \\
&\Big(det(\underline{K})\underline{K}\big([\boldsymbol{C}'\boldsymbol{A}\boldsymbol{B}']\boldsymbol{B} - [\boldsymbol{C}'\boldsymbol{A}\boldsymbol{B}]\boldsymbol{B}'\big)\Big) = 0 \\
\Leftrightarrow\ &det(\underline{K})^4\Big(\big([\boldsymbol{A}'\boldsymbol{B}\boldsymbol{C}'](\overline{K}\boldsymbol{l}_c) - [\boldsymbol{A}'\boldsymbol{B}(\overline{K}\boldsymbol{l}_c)]\boldsymbol{C}'\big)\wedge \\
&\big([\boldsymbol{A}'\boldsymbol{A}\boldsymbol{B}']\overline{K}\boldsymbol{l}_c - [\boldsymbol{A}'\boldsymbol{A}(\overline{K}\boldsymbol{l}_c)]\boldsymbol{B}'\big)\wedge \\
&\big([\boldsymbol{C}'\boldsymbol{A}\boldsymbol{B}']\boldsymbol{B} - [\boldsymbol{C}'\boldsymbol{A}\boldsymbol{B}]\boldsymbol{B}'\big)\Big) = 0 \\
\Leftrightarrow\ &\underbrace{\big([\boldsymbol{A}'\boldsymbol{B}\boldsymbol{C}'](\overline{K}\boldsymbol{l}_c) - [\boldsymbol{A}'\boldsymbol{B}(\overline{K}\boldsymbol{l}_c)]\boldsymbol{C}'\big)}_{\boldsymbol{\alpha}_1}\wedge \\
&\underbrace{\big([\boldsymbol{A}'\boldsymbol{A}\boldsymbol{B}'](\overline{K}\boldsymbol{l}_c) - [\boldsymbol{A}'\boldsymbol{A}(\overline{K}\boldsymbol{l}_c)]\boldsymbol{B}'\big)}_{\boldsymbol{\alpha}_2}\wedge \\
&\underbrace{\big([\boldsymbol{C}'\boldsymbol{A}\boldsymbol{B}']\boldsymbol{B} - [\boldsymbol{C}'\boldsymbol{A}\boldsymbol{B}]\boldsymbol{B}'\big)}_{\boldsymbol{\alpha}_3} = 0.
\end{aligned} \tag{16.40}$$

Note that the scalar $det(\underline{K})^4$ is cancelled out simplifying the expression for the $\boldsymbol{\alpha}$'s. The computation of the intrinsic parameters will be done first considering that the intrinsic parameters remain stationary under camera motions and second when these parameters change.

16.4.1 Computing Stationary Intrinsic Parameters

Let us assume that the basis $\mathcal{F}_0$ is attached to the optical center of the first camera and consider a second camera which has a motion of $[R_1|\boldsymbol{t}_1]$ with respect to the first one. Accordingly the involved projective transformations are given in matrix notation by

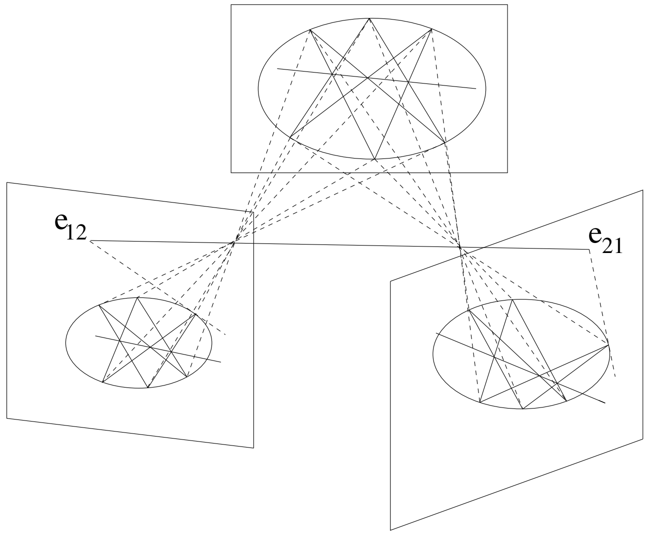

Fig. 16.3. Pascal's theorem at the conic images

$$P_1 = K[I|0] \tag{16.41}$$

$$P_2 = P_1 \begin{pmatrix} R_1 & \boldsymbol{t}_1 \\ 0_3^T & 1 \end{pmatrix}^{-1} = P_1 \left(M_{\mathcal{F}_\mathcal{C}}^{\mathcal{F}_0} \right)^{-1} \tag{16.42}$$

and their optical centres by $\boldsymbol{C}_1 = (0,0,0,1)^T$ and $\boldsymbol{C}_2 = M_{\mathcal{F}_\mathcal{C}}^{\mathcal{F}_0} \boldsymbol{C}_1$. In geometric algebra we use the notations $\underline{P_1}$, $\underline{P_2}$, $\boldsymbol{C}_1 = \mathbf{e}_4$ and $\boldsymbol{C}_2 = \underline{M_{\mathcal{F}_\mathcal{C}}^{\mathcal{F}_0}} \boldsymbol{C}_1$. Thus, we can compute their epipoles as $\boldsymbol{e}_{21} = \underline{P_2}\boldsymbol{C}_1$, $\boldsymbol{e}_{12} = \underline{P_1}\boldsymbol{C}_2$.

Next, we will show by means of an example that the coordinates of the points $\boldsymbol{\alpha}_1$, $\boldsymbol{\alpha}_2$, $\boldsymbol{\alpha}_3$ are entirely independent of the intrinsic parameters. This condition is necessary for solving the problem. Let us choose a camera motion given by

$$[R_1|\boldsymbol{t}_1] = \left(\begin{array}{ccc|c} 0 & -1 & 0 & 2 \\ 1 & 0 & 0 & -1 \\ 0 & 0 & 1 & 3 \end{array}\right). \tag{16.43}$$

For this motion the epipoles are

$$\begin{aligned} \boldsymbol{e}_{12} &= (2k_{11} - k_{12} + 3k_{13})\mathbf{e}_1 + (-k_{22} + 3k_{23})\mathbf{e}_2 + 3\mathbf{e}_3 \quad \text{and} \\ \boldsymbol{e}_{21} &= (k_{11} + 2k_{12} - 3k_{13})\mathbf{e}_1 + (2k_{22} - 3k_{23})\mathbf{e}_2 - 3\mathbf{e}_3. \end{aligned} \tag{16.44}$$

By using the rotated conic points given by the equation (16.15) and replacing $\boldsymbol{e}_{12}$ in the equation (16.40), we can make explicit the $\boldsymbol{\alpha}$'s

$$
\begin{aligned}
\boldsymbol{\alpha}_1 &= ((-3+3i)k_{11}\tau)\mathbf{e}_1 + \\
&\quad (3k_{11}\tau - ik_{12}\tau + ik_{22} + 2ik_{11}\tau - 3k_{12}\tau + 3k_{22})\mathbf{e}_2 + \\
&\quad (ik_{11}\tau + 3k_{12}\tau - 3k_{22} + ik_{12}\tau - ik_{22})\mathbf{e}_3 \\
\boldsymbol{\alpha}_2 &= (-3ik_{11}\tau - 2k_{12}\tau + 2k_{22} - 2k_{11}\tau)\mathbf{e}_1 + \\
&\quad (-6i(k_{12}\tau - k_{22}))\mathbf{e}_2 + (-3k_{11}\tau - 4ik_{12}\tau + 4ik_{22} + 2ik_{11}\tau)\mathbf{e}_3 \\
\boldsymbol{\alpha}_3 &= (1-i)\mathbf{e}_1 + (1-i)\mathbf{e}_2 + 2\mathbf{e}_3. \qquad (16.45)
\end{aligned}
$$

Note that $\boldsymbol{\alpha}_3$ is fully independent of $\underline{K}$. According to Pascal's theorem these three points lie on the same line, therefore, by replacing these points in the equation (16.38) we get the following second order polynomial in τ

$$
\begin{aligned}
&(-40ik_{12}^2 - 52ik_{11}^2 + 16ik_{11}k_{12})\tau^2 + \\
&(-16ik_{11}k_{22} + 80ik_{12}k_{22})\tau - 40ik_{22}^2 = 0. \qquad (16.46)
\end{aligned}
$$

Solving this polynomial and choosing one of the solutions which is nothing else than the solution for one of the two lines tangent to the conic we get

$$
\tau := \frac{16ik_{11}k_{22} - 80ik_{12}k_{22} + 24\sqrt{14}k_{11}k_{22}}{2(-40ik_{12}^2 - 52ik_{11}^2 + 16ik_{11}k_{12})}. \qquad (16.47)
$$

Now considering the homogeneous representation of these intersection points

$$
\boldsymbol{\alpha}_i = \alpha_{i1}\mathbf{e}_1 + \alpha_{i2}\mathbf{e}_2 + \alpha_{i3}\mathbf{e}_3 \sim \frac{\alpha_{i1}}{\alpha_{i3}}\mathbf{e}_1 + \frac{\alpha_{i2}}{\alpha_{i3}}\mathbf{e}_2 + \mathbf{e}_3, \qquad (16.48)
$$

we can finally express their homogeneous coordinates as follows

$$
\alpha_{11} = \frac{-(2k_{11} - 10k_{12} + 3ik_{11}\sqrt{14} + 8ik_{12} + 2k_{12}\sqrt{14} - 10ik_{11} + 2\sqrt{14}k_{11})}{2ik_{11} - 10ik_{12} - 3\sqrt{14}k_{11} - 4k_{12} - 4ik_{12}\sqrt{14} - 16k_{11} + 2ik_{11}\sqrt{14}} \qquad (16.49)
$$

$$
\alpha_{12} = \frac{2i(-2ik_{12} - 3k_{12}\sqrt{14} + 13ik_{11})}{2ik_{11} - 10ik_{12} - 3\sqrt{14}k_{11} - 4k_{12} - 4ik_{12}\sqrt{14} - 16k_{11} + 2ik_{11}\sqrt{14}} \qquad (16.50)
$$

$$
\alpha_{21} = \frac{(1-i)(2ik_{11} - 10ik_{12} - 3\sqrt{14}k_{11})}{5k_{11} - 4k_{12} + ik_{11}\sqrt{14} + 2ik_{12} + 3k_{12}\sqrt{14} - 13ik_{11} + ik_{12}\sqrt{14}} \qquad (16.51)
$$

$$
\alpha_{22} = \frac{11ik_{11} + 8ik_{12} + 3\sqrt{14}k_{11} - 6k_{12} - ik_{12}\sqrt{14} - 3k_{11} + 2ik_{11}\sqrt{14} - 3k_{12}\sqrt{14}}{5k_{11} - 4k_{12} + ik_{11}\sqrt{14} + 2ik_{12} + 3k_{12}\sqrt{14} - 13ik_{11} + ik_{12}\sqrt{14}}. \qquad (16.52)
$$

In the case of exactly orthogonal image axis, we can set in previous equation $k_{12} = 0$ and get

$$
\alpha_{11} = \frac{2i - 3\sqrt{14} + 10 + 2i\sqrt{14}}{2 + 3i\sqrt{14} + 16i + 2\sqrt{14}} \qquad (16.53)
$$

$$
\alpha_{12} = 26\frac{i}{2 + 3i\sqrt{14} + 16i + 2\sqrt{14}} \qquad (16.54)
$$

$$
\alpha_{21} = \frac{(1+i)(-2i + 3\sqrt{14})}{-5i + \sqrt{14} - 13} \qquad (16.55)
$$

$$
\alpha_{22} = -\frac{-11 + 3i\sqrt{14} - 3i - 2\sqrt{14}}{-5i + \sqrt{14} - 13}. \qquad (16.56)
$$

The coordinates of the intersection points are indeed independent of the intrinsic parameters.

After this illustration by an example we will get now the coordinates for any general camera motion. For that it is necessary to separate in the projections the intrinsic parameters from the extrinsic ones. Let us define

$$\boldsymbol{s} = s_1\mathbf{e}_1 + s_2\mathbf{e}_2 + s_3\mathbf{e}_3 = \underline{[I|0]}\ \underline{M_{\mathcal{F}_\mathcal{C}}^{\mathcal{F}_0}}\boldsymbol{C}_1. \tag{16.57}$$

Thus, the epipole is

$$\boldsymbol{e}_{12} = \underline{K}\ \underline{[I|0]}\ \underline{M_{\mathcal{F}_\mathcal{C}}^{\mathcal{F}_0}}\boldsymbol{C}_1 = \underline{K}\boldsymbol{s}. \tag{16.58}$$

Note that in this expression the intrinsic parameters are separate from the extrinsic ones. Similar as above for the general camera motion with the corresponding epipole value the coordinates for the intersecting points read

$$\alpha_{11} = -\frac{(-s_3s_1s_2+is_3\sqrt{s_3^2(s_1^2+s_2^2+s_3^2)}-is_3^3-is_3s_1^2+s_1\sqrt{s_3^2(s_1^2+s_2^2+s_3^2)}-is_2s_3^2)}{(-is_3s_1s_2-s_3\sqrt{s_3^2(s_1^2+s_2^2+s_3^2)}-s_3^3-s_3s_1^2+is_1\sqrt{s_3^2(s_1^2+s_2^2+s_3^2)}+s_2s_3^2)} \tag{16.59}$$

$$\alpha_{21} = \frac{-2s_3(s_3^2+s_1^2)}{-is_3s_2s_1-s_3\sqrt{s_3^2(s_1^2+s_2^2+s_3^2)}-s_3^3-s_3s_1^2+is_1\sqrt{s_3^2(s_1^2+s_2^2+s_3^2)}+s_2s_3^2} \tag{16.60}$$

$$\alpha_{12} = \frac{(-1-i)(is_1s_2+\sqrt{s_3^2(s_1^2+s_2^2+s_3^2)})s_3}{-is_3s_1s_2-s_3\sqrt{s_3^2(s_1^2+s_2^2+s_3^2)}+s_3s_1^2+s_3^3+s_1\sqrt{s_3^2(s_1^2+s_2^2+s_3^2)}-is_2s_3^2} \tag{16.61}$$

$$\alpha_{22} = \frac{i(is_3s_1s_2+s_3\sqrt{s_3^2(s_1^2+s_2^2+s_3^2)}+is_1\sqrt{s_3^2(s_1^2+s_2^2+s_3^2)}+s_2s_3^2+is_3s_1^2+is_3^3)}{-is_3s_1s_2-s_3\sqrt{s_3^2(s_1^2+s_2^2+s_3^2)}+s_3s_1^2+s_3^3+s_1\sqrt{s_3^2(s_1^2+s_2^2+s_3^2)}-is_2s_3^2}. \tag{16.62}$$

Note that the intrinsic parameters are totally cancelled out. The invariance properties can be used to obtain equations which depend on the four unknown intrinsic camera parameters. The algorithm can be summarized in the following steps.

1. Suppose point correspondences between two cameras and motion between the cameras.
2. Calculate the values of the homogeneous $\boldsymbol{\alpha}_i$ by using the known camera motion and the formulas (16.59–16.62).
3. Calculate $\overline{K}\boldsymbol{l}_c$ with the epipole, evaluated from the point correspondences. To fulfill Pascal's theorem solve the equations system to τ similar to (16.47).
4. Replace τ in (16.45) and calculate the homogeneous representation of these intersection points to get quadratic polynomials which depends on the four unknown intrinsic parameters. Note that the intrinsic parameters are not cancelled out because of the insert of the real values from the epipol. Because of the invariant properties of the $\boldsymbol{\alpha}$'s the polynomials must be equal to the evaluated values of the $\boldsymbol{\alpha}$'s in step 2. This leads to four quadratic equations.

Since we are assuming that the intrinsic parameters remain constant, we can consequently gain a second set of four equations depending again of the four intrinsic parameters from the second epipole.

The interesting aspect here is that we require only one camera motion to find a solvable equation system. Other methods gain for each camera motion only a couple of equations, thus they require at least three camera motions to solve the problem [169, 163]. This particular advantage of our approach relies in the investigation of Pascal's theorem and its formulation in geometric algebra.

16.4.2 Computing Non–stationary Intrinsic Parameters

In this case we will consider that due to the camera motion the intrisic parameters may have been changed. The procedure can be formulated along the same previous ideas with the difference that we compute the line $\boldsymbol{l}_c$ using the operator for the fundamental matrix and a point lying at line at infinite of the second camera as $\boldsymbol{l}_c = \overline{F}(\mathbf{e}_1 + \tau' \mathbf{e}_2)$.

Note that the fundamental matrix can be expressed in terms of the motion between cameras and the K of the camera, i.e. $F = K^{-T}[\boldsymbol{t}]_\times R_{12} K^T$ where $[\boldsymbol{t}]_\times$ is the tensor notation of the antisymmetric matrix representing the translation [163]. The term $E = [\boldsymbol{t}]_\times R_{12}$ is called the essential matrix. The decomposition of F can instantaneous be described by $\underline{F} = \overline{K}^{-1} \underline{[\boldsymbol{t}]_\times}\ \underline{R_{12}} \overline{K}$ in terms of geometric algebra.

Now similar as in previous case we will use an example for facilitating the understanding. We will use the same camera motion given in equation (16.43). The fundamental matrix in terms of the intrinsic parameters of the first camera K and of the second one K', with the assumption of perpendicular pixel grids $k_{12} = k'_{12} = 0$, and the camera motion reads in matrix notation

$$
\begin{aligned}
F &= {K^{-1}}^T [\boldsymbol{t}]_\times R K'^{-1} \\
&= \begin{pmatrix} -3\frac{k'_{22}k_{22}}{v_2} & 0 & -\frac{(k'_{11}-3k'_{13})k_{22}k'_{22}}{v_2} \\ 0 & -3\frac{k'_{11}k_{11}}{v_2} & -\frac{k_{11}k'_{11}(2k'_{22}-3k'_{23})}{v_2} \\ \frac{(2k_{11}+3k_{13})k_{22}k'_{22}}{v_2} & -\frac{(k_{22}-3k_{23})k_{11}k'_{11}}{v_2} & 1 \end{pmatrix}
\end{aligned} \tag{16.63}
$$

where $v_2 = -3k'_{22}k_{22}k_{13}k'_{13} + k_{22}k'_{22}k'_{11}k_{13} + k_{22}k'_{23}k'_{11}k_{11} - 2k_{22}k'_{22}k'_{13}k_{11} + 2k_{23}k'_{22}k'_{11}k_{11} - 3k_{23}k'_{23}k'_{11}k_{11}$.

The value of the line $\boldsymbol{l}_c$ is now computed in terms of the operator of the fundamental matrix, i.e. $\boldsymbol{l}_c = \overline{F}(\mathbf{e}_1 + \tau' \mathbf{e}_2)$ Similar as above we compute the $\boldsymbol{\alpha}$'s and according the Pascal's theorem we gain a polynomial similar as equation (16.46). This reads

$$
10{k'}_{11}^2 {\tau'}^2 - 4k'_{22}k'_{11}\tau' + 13{k'}_{22}^2 = 0. \tag{16.64}
$$

We select one of both solutions of this second order polynomial

$$
\tau' = \frac{4k'_{22}k'_{11} + 6ik'_{22}k'_{11}\sqrt{14}}{20({k'}_{11}^2)} \tag{16.65}
$$

and substitute it in the homogeneous coordinates of the $\boldsymbol{\alpha}$'s

$$\alpha_{11} = -\frac{i(-5i-4+i\sqrt{14})}{5i+2+2i\sqrt{14}} \tag{16.66}$$

$$\alpha_{21} = \frac{-2+3i\sqrt{14}}{5i+2+2i\sqrt{14}} \tag{16.67}$$

$$\alpha_{12} = \frac{10-10i}{-4i-2+3i\sqrt{14}-\sqrt{14}} \tag{16.68}$$

$$\alpha_{22} = -\frac{8+6i-\sqrt{14}+3i\sqrt{14}}{-4i-2+3i\sqrt{14}-\sqrt{14}}, \tag{16.69}$$

where $\boldsymbol{\alpha_3} = (1-i)\mathbf{e}_1 + (1-i)\mathbf{e}_2 + 2\mathbf{e}_3$ is again fully independent of the intrinsic parameters.

Finally, we will show the expression when we consider now a general motion

$$[R|\boldsymbol{t}] = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{pmatrix}. \tag{16.70}$$

In matrix algebra the fundamental matrix reads

$$F = K^{-T} E K'^{-1} \tag{16.71}$$

$$= \begin{pmatrix} k_{11} & 0 & k_{13} \\ 0 & k_{22} & k_{23} \\ 0 & 0 & 1 \end{pmatrix}^{-T} \begin{pmatrix} E_{11} & E_{12} & E_{13} \\ E_{21} & E_{22} & E_{23} \\ E_{31} & E_{32} & E_{33} \end{pmatrix} \begin{pmatrix} k'_{11} & 0 & k'_{13} \\ 0 & k'_{22} & k'_{23} \\ 0 & 0 & 1 \end{pmatrix}^{-1} \tag{16.72}$$

and in geometric algebra the operator of the fundamental matrix reads $\underline{F} = \overline{K}^{-1} \underline{E}\, \underline{K}'^{-1}$.

Using this formulation we compute the homogeneous coordinates of the $\boldsymbol{\alpha}$'s

$$\begin{aligned} \alpha_{11} &= i(iE_{11}E_{22}^2 + iE_{11}E_{32}^2 - iE_{12}E_{21}E_{22} - iE_{12}E_{31}E_{32} \\ &-iE_{12}\sqrt{v_3} + E_{21}E_{12}^2 + E_{21}E_{32}^2 - E_{22}E_{11}E_{12} - E_{22}E_{31}E_{32} \\ &-E_{22}\sqrt{v_3} - E_{31}E_{12}^2 - E_{31}E_{22}^2 + E_{32}E_{11}E_{12} + E_{32}E_{21}E_{22} + E_{32}\sqrt{v_3})/ \\ &(iE_{11}E_{22}^2 + iE_{11}E_{32}^2 - iE_{12}E_{21}E_{22} - iE_{12}E_{31}E_{32} - iE_{12}\sqrt{v_3} - \\ &E_{31}E_{12}^2 - E_{31}E_{22}^2 + E_{32}E_{11}E_{12} + E_{32}E_{21}E_{22} + E_{32}\sqrt{v_3} - \\ &E_{21}E_{12}^2 - E_{21}E_{32}^2 + E_{22}E_{11}E_{12} + E_{22}E_{31}E_{32} + E_{22}\sqrt{v_3}) \end{aligned} \tag{16.73}$$

$$
\begin{aligned}
\alpha_{12} \; &= 2(-E_{21}E_{12}^2 - E_{21}E_{32}^2 + E_{22}E_{11}E_{12} + E_{22}E_{31}E_{32} + E_{22}\sqrt{v_3})/ \\
&(iE_{11}E_{22}^2 + iE_{11}E_{32}^2 - iE_{12}E_{21}E_{22} - iE_{12}E_{31}E_{32} \\
&-iE_{12}\sqrt{v_3} - E_{31}E_{12}^2 - E_{31}E_{22}^2 + E_{32}E_{11}E_{12} + E_{32}E_{21}E_{22} + \\
&E_{32}\sqrt{v_3} - E_{21}E_{12}^2 - E_{21}E_{32}^2 + E_{22}E_{11}E_{12} + E_{22}E_{31}E_{32} + E_{22}\sqrt{v_3})
\end{aligned} \tag{16.74}
$$

$$
\begin{aligned}
\alpha_{21} \; &= (1-i)(E_{11}E_{22}^2 + E_{11}E_{32}^2 - E_{12}E_{21}E_{22} - E_{12}E_{31}E_{32} - E_{12}\sqrt{v_3})/ \\
&(-iE_{11}E_{22}^2 - iE_{11}E_{32}^2 + iE_{12}E_{21}E_{22} + iE_{12}E_{31}E_{32} \\
&+iE_{12}\sqrt{v_3} - E_{21}E_{12}^2 - E_{21}E_{32}^2 + E_{22}E_{11}E_{12} + E_{22}E_{31}E_{32} \\
&+E_{22}\sqrt{v_3} - iE_{31}E_{12}^2 - iE_{31}E_{22}^2 + iE_{32}E_{11}E_{12} \\
&+iE_{32}E_{21}E_{22} + iE_{32}\sqrt{v_3})
\end{aligned} \tag{16.75}
$$

$$
\begin{aligned}
\alpha_{22} \; &= -(E_{11}E_{22}^2 + E_{11}E_{32}^2 - E_{12}E_{21}E_{22} - E_{12}E_{31}E_{32} - E_{12}\sqrt{v_3} \\
&-iE_{31}E_{12}^2 - iE_{31}E_{22}^2 + iE_{32}E_{11}E_{12} + iE_{32}E_{21}E_{22} \\
&+iE_{32}\sqrt{v_3} - E_{21}E_{12}^2 - E_{21}E_{32}^2 + E_{22}E_{11}E_{12} + E_{22}E_{31}E_{32} \\
&+E_{22}\sqrt{v_3})/(-iE_{11}E_{22}^2 - iE_{11}E_{32}^2 + iE_{12}E_{21}E_{22} \\
&+iE_{12}E_{31}E_{32} + iE_{12}\sqrt{v_3} - E_{21}E_{12}^2 - E_{21}E_{32}^2 \\
&+E_{22}E_{11}E_{12} + E_{22}E_{31}E_{32} + E_{22}\sqrt{v_3} - iE_{31}E_{12}^2 \\
&-iE_{31}E_{22}^2 + iE_{32}E_{11}E_{12} + iE_{32}E_{21}E_{22} + iE_{32}\sqrt{v_3})
\end{aligned} \tag{16.76}
$$

where

$$
\begin{aligned}
v_3 &= 2E_{11}E_{12}E_{21}E_{22} + 2E_{11}E_{12}E_{31}E_{32} + 2E_{21}E_{22}E_{31}E_{32} \\
&-E_{12}^2E_{31}^2 - E_{12}^2E_{21}^2 - E_{22}^2E_{31}^2 - E_{22}^2E_{11}^2 - E_{32}^2E_{21}^2 - E_{32}^2E_{11}^2 .
\end{aligned} \tag{16.77}
$$

Note that for the general case the α's are fully independent of the intrinsic camera coefficients k_{ij} or k'_{ij}. Together with the equations of the $\boldsymbol{\alpha}$'s obtained using the first epipole the intrinsic parameters can be found solving a quadratic equation system.

16.5 Experimental Analysis

In this section we present tests of the method based on Pascal's theorem using firstly simulated images. We explore the effect of different kinds of camera motion and the effect of increasing noise in the computing of the intrinsic camera parameters. The experiments with real images show that the performance of the method is reliable.

16.5.1 Experiments with Simulated Images

Using a Maple simulation we firstly test the method based on the theorem of Pascal to explore the dependency of the type and the amount of necessary camera motions for solving the problem. The experiments show that at least a rotation about only one axis and a displacement along the three axes are necessary for stabile computations of all intrinsic parameters. Then, we realize a test of our approach by increasing noise.

The camera is rotated about the y–axis with translation along the three camera axes. For the tests we used exact arithmetic of the Maple program instead of floating point arithmetic of the C language. The Table 1 shows the computed intrinsic parameters. The most right column of the table shows the error obtained substituting these parameters in the polynomial (16.64) which gives zero for the case of zero noise. The values in this column show that by increasing noise the computed intrinsic parameters cause a tiny deviation of the ideal value of zero. This indicates that the procedure is relatively stable against noise. We could image that there is a relative flat surface around the global minimum of the polynomial. Note that there are remarkable deviations shown by noise 1.25.

Table 16.1. Intrinsic parameters by rotation about the y–axis and translation along the three axes with increasing noise

Noise(pixels)	k_{11}	k_{13}	k_{22}	k_{23}	Error
0	500	256	500	256	10^{-8}
0.1	505	259	509	261	0.001440
0.5	504	259.5	503.5	258	0.004897
0.75	498	254	503.5	258	0.001668
1	482	242	485	254	0.011517
1.25	473	220	440	238	0.031206
1.5	517	272	518	266	0.015
2	508	262.5	504	258.5	0.006114
2.5	515	268	501.9	257	0.011393
3	510	265	524	276	0.011440

16.5.2 Experiments with Real Images

In this section we present experiments using real images with one general camera motion, see Figure 16.4.

The motion was done about the three coordinate axes. We use a calibration dice and for comparison purposes we compute the intrinsic parameters from the involved projective matrices by splitting the intrinsic parameters from the extrinsic ones. The reference values were: First camera

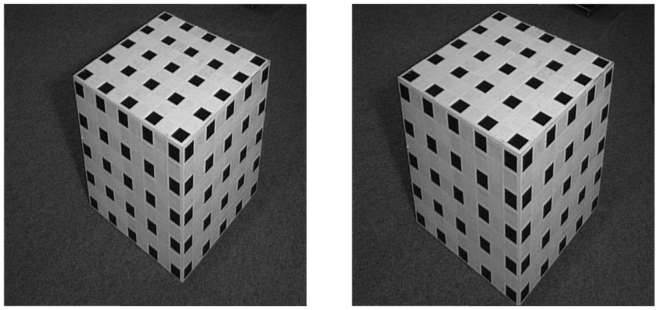

Fig. 16.4. Scenario

$k_{11} = 1200.66$, $k_{22} = 1154.77$, $k_{13} = 424.49$, $k_{23} = 264.389$ and second camera $k_{11} = 1187.82$, $k_{22} = 1141.58$, $k_{13} = 386.797$, $k_{23} = 288.492$ with mean errors of 0.688 and 0.494, respectively.

Thereafter, using the gained extrinsic parameters $[R_1|\boldsymbol{t}_1]$ and $[R_2|\boldsymbol{t}_2]$ we compute the relation $[R|\boldsymbol{t}]$ between cameras which is required for the Pascal's theorem based method. The fundamental matrix is computed using a non-linear method. Using the Pascal's theorem based method with 12 point correspondences unlike 160 point correspondences used by the algorithm with the calibration dice we compute the following intrinsic parameters $k_{11} = 1244$, $k_{22} = 1167$, $k_{13} = 462$ and $k_{23} = 217$. The error is computed using the eight equations gained from the $\boldsymbol{\alpha}$'s of the first and second camera. These values resemble quite well to the reference ones and cause an error of $\sqrt{|eqn_1|^2 + ... + |eqn_8|^2}$: 0.004961 in the error function. The difference with the reference values is attributable to inherent noise in the computation and to the fact that the reference values are not exact, too.

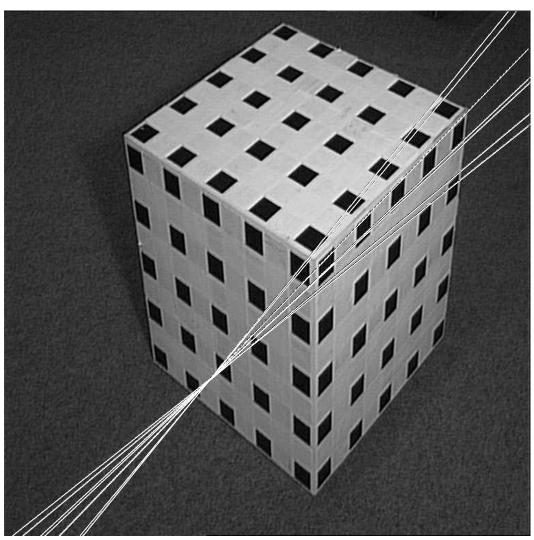

Fig. 16.5. Superimposed epipolar lines using the reference and Pascal's theorem based method

Since this is a system of quadratic equations we resort to an iterative procedure for finding the solution. First we tried the Newton–Raphson method [196] and the continuation method [163]. These methods are not practicable enough due to their complexity. We use instead a variable in size window minima search which through the computation ensure the reduction of the quadratic error. This simply approach work faster and reliable.

In order to visualize how good we gain the epipolar geometry we superimposed the epipolar lines for some points using the reference method and Pascal's theorem based method. In both cases we computed the fundamental matrix in terms of their intrinsic parameters, i.e. $F = K^{-T}[\boldsymbol{t}]_{\times}RK^{-1}$. Figure 16.5 shows this comparison. It is clear that both methods give quite similar epipolar lines and interesting enough it is shown that the intersecting point or epipole coincide almost exactly.

16.6 Conclusions

This paper presents a geometric approach to formulate the Kruppa equations in terms of pure brackets. This can certainly help to explore the geometry of the calibration problem and to find degenerated cases. Furthermore this paper presents an approach to compute the intrinsic camera parameters in the geometric algebra framework using Pascal's theorem. We adopt the projected characteristics of the absolute conic in terms of Pascal's theorem to propose a new camera calibration method based on geometric thoughts. The use of this theorem in the geometric algebra framework allows us the computing of a projective invariant using the conics of only two images. Then, this projective invariant expressed in terms of brackets helps us to set enough equations to solve the calibration problem. Our method requires to know the point correspondences and the values of the camera motion. The method gives a new point of view for the understanding of the problem thanks to the application of Pascal's theorem and it also explains the overseen role of the projective invariant in terms of the brackets. Using synthetic and real images we show that the method performs efficiently without any initialization or getting trapped in local minima.