Geometry and Kinematics with Uncertain Data

Christian Perwass, Christian Gebken, and Gerald Sommer

Institut für Informatik, CAU Kiel
Christian-Albrechts-Platz 4, 24118 Kiel, Germany
{chp,chg,gs}@ks.informatik.uni-kiel.de

Abstract. In Computer Vision applications, one usually has to work with uncertain data. It is therefore important to be able to deal with uncertain geometry and uncertain transformations in a uniform way. The Geometric Algebra of conformal space offers a unifying framework to treat not only geometric entities like points, lines, planes, circles and spheres, but also *transformations* like reflection, inversion, rotation and translation. In this text we show how the uncertainty of all elements of the Geometric Algebra of conformal space can be appropriately described by covariance matrices. In particular, it will be shown that it is advantageous to represent uncertain transformations in Geometric Algebra as compared to matrices. Other important results are a novel pose estimation approach, a uniform framework for geometric entity fitting and triangulation, the testing of uncertain tangentiality relations and the treatment of catadioptric cameras with parabolic mirrors within this framework. This extends previous work by Förstner and Heuel from points, lines and planes to non-linear geometric entities and transformations, while keeping the linearity of the estimation method. We give a theoretical description of our approach and show exemplary applications.

1 Introduction

In Computer Vision one has to deal almost invariably with uncertain data. Appropriate methods to deal with this uncertainty do therefore play an important role. In this text we show how geometric entities and transformations can be described together with their uncertainty in a single, unifying mathematical framework, namely the Geometric Algebra of conformal space.

A particular advantage of the presented approach stems from the linear representation of geometric entities and transformations and from the fact that algebra operations are simply bilinear functions. This allows us to easily construct geometric constraints with the symbolic power of the algebra and then to equivalently express these constraints as multi-linear functions, such that the whole body of linear algebra can be applied. Solutions to many problems, like the estimation of the best line, plane, circle or sphere fit through a set of points, or the best rotation between two point sets (in a least-squares sense), reduces to the estimation of the null space of a matrix. Applying the so called Gauss-Helmert model, it is then also possible to evaluate the uncertainty of the estimated entity.

This text builds on previous works by Förstner et al. [1] and Heuel [2] where uncertain points, lines and planes were treated in a unified manner. The linear estimation of rotation operators in Geometric Algebra was previously discussed in [3], albeit without taking account of uncertainty. In [4] the description of uncertain circles and 2D-conics in Geometric Algebra was first discussed. The stratification of Euclidean, projective and affine spaces in Geometric Algebra, has been previously presented in [5]. In [6] the estimation of uncertain general operators was introduced.

In this text we present a number of new results and show how this method can be used in important applications of Computer Vision. We start out with a short introduction to Geometric Algebra. We then show how uncertain geometry and transformations can be represented in the algebra and discuss the error introduced when embedding Euclidean vectors in conformal space. Then we present the novel result that the uncertainty of transformations can be represented by linear subspaces, i.e. through a covariance matrix. Note that this is, for example, not possible for rotation matrices, since the sub-space of orthogonal matrices is not linear.

Next a number of applications of this methodology are presented. Firstly, estimation of geometric entities is discussed, where it is, for example, shown that triangulation of points and lines can be done in much the same way as the fitting of lines, planes, circles and spheres to a set of points. Next we present two variants of pose estimation, one of which estimates the pose of a known object given a set of projection rays. The corresponding constraint equation is quadratic in the components of the transformation operator, while not making any approximations of the operator. Later on we show how the estimation of projection rays from corresponding image points can be done via a matrix multiplication, for a projective and a catadioptric camera with parabolic mirror. The latter is, to the best of our knowledge, a new result, which makes pose estimation with catadioptric cameras mathematically as complex as pose estimation with projective cameras. Furthermore, we also show how uncertain geometric relations can be tested. This includes next to the test for the intersection of two lines, also tests for tangentiality of planes to circles and spheres.

2 Geometric Algebra

For a detailed introduction to Geometric Algebra see e.g. [7, 8]. Here we can only give a short overview. Geometric Algebra is an associative, graded algebra, whereby the algebra product is called *geometric product*. The Geometric Algebra over a n-dimensional vector space $\mathbb{R}^{p,q}$, with $n = p + q$ has dimension 2^n and is denoted by $\mathbb{G}(\mathbb{R}^{p,q})$ or simply $\mathbb{G}_{p,q}$. Here p denotes the number of basis elements of the vector space that square to $+1$ and q the number of basis elements that square to -1. If only one index is given, it denotes the number of positively squaring basis elements. Elements of different grade of the algebra can be constructed through the *outer product* of linearly independent vectors. For example, if $\{\boldsymbol{a}_i\} \in \mathbb{R}^n$ are a set of k linearly independent vectors, then $\boldsymbol{A}_{\langle k \rangle} := \boldsymbol{a}_1 \wedge \ldots \wedge \boldsymbol{a}_k$ is an element of $\mathbb{G}_n$ of grade k, which is called a *blade*, where $\wedge$ denotes the outer product. A general element of the algebra, called *multivector*, can always be expressed as a linear combination of blades of possibly different grades. Geometric entities are represented in the algebra through blades, while operators are typically represented by linear combinations of blades of different grades.

Geometric Algebra of Conformal Space To combine projective geometry and kinematics we need to consider the Geometric Algebra of the (projective) *conformal space* of 3D-Euclidean space (cf. [7]). The embedding function $\mathcal{K}$ is defined as $\mathcal{K} : \boldsymbol{x} \in \mathbb{R}^3 \mapsto \boldsymbol{x}+\frac{1}{2}\,\boldsymbol{x}^2\,\mathbf{e}_\infty+\mathbf{e}_o \in \mathbb{R}^{4,1}$. The basis of $\mathbb{R}^{4,1}$ can be written as $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3, \mathbf{e}_\infty, \mathbf{e}_o\}$, where $\mathbf{e}_i^2 = +1$, $\mathbf{e}_\infty^2 = \mathbf{e}_o^2 = 0$ and $\mathbf{e}_\infty \cdot \mathbf{e}_o = -1$. The various geometric entities that can be represented by blades in $\mathbb{G}_{4,1}$ are shown in table 1. In this table $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{U}, \boldsymbol{V} \in \mathbb{R}^{4,1}$ are embeddings of points $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^3$, respectively, and the $\mathbf{e}_{ij} \equiv \mathbf{e}_i \wedge \mathbf{e}_j$ etc. denote the algebra basis elements of an entity.

In particular, note that the elements *homogeneous point*, *line* and *plane* represent those elements that can also be expressed in the Geometric Algebra over projective space. For the homogeneous point, the element $\mathbf{e}_{o\infty}$ takes on the role of the homogeneous dimension.

Apart from representing geometric entities by blades, it is also possible to define operators in Geometric Algebra. The class of operators we are particularly interested in are *versors*. A versor $\boldsymbol{V} \in \mathbb{G}_n$ is a multivector that satisfies the following two conditions: $\boldsymbol{V}\widetilde{\boldsymbol{V}} = 1$ and for any blade $\boldsymbol{A}_{\langle k \rangle} \in \mathbb{G}_n$, $\boldsymbol{V}\,\boldsymbol{A}_{\langle k \rangle}\,\widetilde{\boldsymbol{V}}$ is also of grade k, i.e. a versor is *grade preserving*. The expression $\widetilde{\boldsymbol{V}}$ denotes the *reverse* of $\boldsymbol{V}$. The reverse operation changes the sign of the constituent blade elements depending on their grade, which has an effect similar to conjugation in quaternions. The most interesting versors for our purposes in conformal space are rotation operators (rotors), translation operators (translators) and scaling operators (dilators).

All of them share the property that they can be applied to all geometric entities in the same way. That is, it does not matter whether a blade $\boldsymbol{A}_{\langle k \rangle}$ represents a point, line, plane, circle or sphere. If $\boldsymbol{R}$ represents a rotation operations, then the rotated entity is always given by $\boldsymbol{R}\,\boldsymbol{A}_{\langle k \rangle}\,\widetilde{\boldsymbol{R}}$.

Entity	**Grade**	**No.**	**Basis Elements**
Point $\boldsymbol{X}$	1	5	$\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3, \mathbf{e}_\infty, \mathbf{e}_o$
Homogen. Point $\boldsymbol{X} \wedge \mathbf{e}_\infty$	2	4	$\mathbf{e}_{1\infty}, \mathbf{e}_{2\infty}, \mathbf{e}_{3\infty}, \mathbf{e}_{o\infty}$
Point Pair $\boldsymbol{X} \wedge \boldsymbol{Y}$	2	10	$\mathbf{e}_{23}, \mathbf{e}_{31}, \mathbf{e}_{12}, \mathbf{e}_{1o}, \mathbf{e}_{2o}, \mathbf{e}_{3o}, \mathbf{e}_{1\infty}, \mathbf{e}_{2\infty}, \mathbf{e}_{3\infty}, \mathbf{e}_{o\infty}$
Line $\boldsymbol{X} \wedge \boldsymbol{Y} \wedge \mathbf{e}_\infty$	3	6	$\mathbf{e}_{23\infty}, \mathbf{e}_{31\infty}, \mathbf{e}_{12\infty}, \mathbf{e}_{1o\infty}, \mathbf{e}_{2o\infty}, \mathbf{e}_{3o\infty}$
Circle $\boldsymbol{X} \wedge \boldsymbol{Y} \wedge \boldsymbol{Z}$	3	10	$\mathbf{e}_{23\infty}, \mathbf{e}_{31\infty}, \mathbf{e}_{12\infty}, \mathbf{e}_{23o}, \mathbf{e}_{31o}, \mathbf{e}_{12o},$ $\mathbf{e}_{1o\infty}, \mathbf{e}_{2o\infty}, \mathbf{e}_{3o\infty}, \mathbf{e}_{123}$
Plane $\boldsymbol{X} \wedge \boldsymbol{Y} \wedge \boldsymbol{Z} \wedge \mathbf{e}_\infty$	4	4	$\mathbf{e}_{123\infty}, \mathbf{e}_{23o\infty}, \mathbf{e}_{31o\infty}, \mathbf{e}_{12o\infty}$
Sphere $\boldsymbol{X} \wedge \boldsymbol{Y} \wedge \boldsymbol{Z} \wedge \boldsymbol{U}$	4	5	$\mathbf{e}_{123\infty}, \mathbf{e}_{123o}, \mathbf{e}_{23o\infty}, \mathbf{e}_{31o\infty}, \mathbf{e}_{12o\infty}$
Reflection	1	4	$\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3, \mathbf{e}_\infty$
Inversion	1	5	$\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3, \mathbf{e}_\infty, \mathbf{e}_o$
Rotor $\boldsymbol{R}$	0,2	4	$1, \mathbf{e}_{23}, \mathbf{e}_{31}, \mathbf{e}_{12}$
Translator $\boldsymbol{T}$	0,2	4	$1, \mathbf{e}_{1\infty}, \mathbf{e}_{2\infty}, \mathbf{e}_{3\infty}$
Dilator $\boldsymbol{D}$	0,2	2	$1, \mathbf{e}_{o\infty}$
Motor $\boldsymbol{RT}$	0,2,4	8	$1, \mathbf{e}_{23}, \mathbf{e}_{31}, \mathbf{e}_{12}, \mathbf{e}_{1\infty}, \mathbf{e}_{2\infty}, \mathbf{e}_{3\infty}, \mathbf{e}_{123\infty}$
Gen. Rotor $\boldsymbol{TR}\widetilde{\boldsymbol{T}}$	0,2	7	$1, \mathbf{e}_{23}, \mathbf{e}_{31}, \mathbf{e}_{12}, \mathbf{e}_{1\infty}, \mathbf{e}_{2\infty}, \mathbf{e}_{3\infty}$

Table 1. Entities and their algebra basis. Note that the operators are mostly multivectors of mixed grade.

Representation as Component Vectors Let $\{\boldsymbol{E}_i\}$ denote the 2^n-dimensional algebra basis of $\mathbb{G}_n$. Then a multivector $\boldsymbol{A} \in \mathbb{G}_n$ can be written as $\boldsymbol{A} = \mathsf{a}^i\, \boldsymbol{E}_i$, where a^i denotes the i^{th} component of a vector $\mathsf{a} \in \mathbb{R}^{2^n}$ and a sum over the repeated index i is implied. We will use this Einstein summation convention also in the following, i.e. $\mathsf{a}^i\, \boldsymbol{E}_i \equiv \sum_i \mathsf{a}^i\, \boldsymbol{E}_i$. If $\boldsymbol{B} = \mathsf{b}^i\, \boldsymbol{E}_i$ and $\boldsymbol{C} = \mathsf{c}^i\, \boldsymbol{E}_i$, then the components of $\boldsymbol{C}$ in the algebra equation $\boldsymbol{C} = \boldsymbol{A} \circ \boldsymbol{B}$ can be evaluated via $\mathsf{c}^k = \mathsf{a}^i\, \mathsf{b}^j\, \mathsf{G}^k{}_{ij}$, where a summation over i and j is again implied. Such a summation of tensor indices is also called *contraction*. Here $\circ$ is a placeholder for an algebra product and $\mathsf{G}^k{}_{ij} \in \mathbb{R}^{2^n \times 2^n \times 2^n}$ is a tensor encoding this product.

The set of tensor symbols representing the various algebra operations, that we use in the following, is shown in table 2. This table also gives the symbolic abbreviations for the Jacobi matrices of the tensor contractions.

For example, the geometric product of multivectors $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{G}_n$ can be written in terms of their component vectors $\mathsf{a}, \mathsf{b} \in \mathbb{R}^{2^n}$ as $\mathsf{a}^i\, \mathsf{b}^j\, \mathsf{G}^k{}_{ij} = \mathsf{G}(\mathsf{a})\, \mathsf{b} = \bar{\mathsf{G}}(\mathsf{b})\, \mathsf{a}$.

We can reduce the complexity of the tensor equations considerably by only using those components of multivectors that are actually needed. In the following we therefore refer to the minimum number of components as given in table 1, when talking about the component vector of a multivector.

Operation	Geometric Product	Outer Product	Inner Product	Reverse	Dual
Tensor Symbol	$\mathsf{G}^k{}_{ij}$	$\mathsf{O}^k{}_{ij}$	$\mathsf{N}^k{}_{ij}$	$\mathsf{R}^j{}_i$	$\mathsf{D}^j{}_i$
Jacobi Matrices	$\mathsf{G}(\mathsf{a}) := \mathsf{a}^i\, \mathsf{G}^k{}_{ij}$ $\bar{\mathsf{G}}(\mathsf{b}) := \mathsf{b}^j\, \mathsf{G}^k{}_{ij}$	$\mathsf{O}(\mathsf{a}) := \mathsf{a}^i\, \mathsf{O}^k{}_{ij}$ $\bar{\mathsf{O}}(\mathsf{b}) := \mathsf{b}^j\, \mathsf{O}^k{}_{ij}$	$\mathsf{N}(\mathsf{a}) := \mathsf{a}^i\, \mathsf{N}^k{}_{ij}$ $\bar{\mathsf{N}}(\mathsf{b}) := \mathsf{b}^j\, \mathsf{N}^k{}_{ij}$	$\mathsf{R} := \mathsf{R}^j{}_i$	$\mathsf{D} := \mathsf{D}^j{}_i$

Table 2. Tensor symbols for algebra operations and corresponding Jacobi matrices. Note that for tensors with two indices (i.e. matrices) we define the first index to denote the matrix row and the second index the matrix column.

3 Geometric Algebra with Uncertain Entities

In order to describe the uncertainty of multivectors, we need to expressed them as component vectors and algebra operations as tensor contractions.

Operations between Multivectors We now give a short description of *error propagation* for operations between uncertain multivectors. This is based on the assumption that the uncertainty of a multivector can be modeled by a Gaussian distribution. Hence, the probability density function of a random multivector variable is fully described by a mean multivector and a covariance matrix. Using error propagation we can then evaluate the mean and covariance of a function of random multivector variables. In particular, this allows us to evaluate the mean and covariance of algebra products between multivector valued random variables. For a detailed introduction to error propagation see [9, 10].

We will denote a random variable by an underbar, its expectation or mean value by the symbol itself, the expectation operator by $\mathcal{E}$ and the covariance matrix of a random vector variable $\underline{\mathsf{a}}$ by $\Sigma_{\mathsf{a},\mathsf{a}}$. The cross-covariance matrix between two random variables $\underline{\mathsf{a}}$ and $\underline{\mathsf{b}}$, say, will be written as $\Sigma_{\mathsf{a},\mathsf{b}}$.

Let $\underline{\boldsymbol{A}}, \underline{\boldsymbol{B}} \in \mathbb{G}_n$ be two general random multivector variables and $\underline{\mathsf{a}}, \underline{\mathsf{b}} \in \mathbb{R}^{2^n}$ their component vectors. Furthermore, let $\underline{\boldsymbol{C}} \in \mathbb{G}_n$ be given by $\underline{\boldsymbol{C}} = \underline{\boldsymbol{A}}\,\underline{\boldsymbol{B}}$. It then follows that $\underline{\mathsf{c}} = \mathsf{G}(\underline{\mathsf{a}})\,\underline{\mathsf{b}}$. Since we assume the random vector variables to have Gaussian probability density distributions, we would like to know the expectation value and covariance matrix of $\underline{\boldsymbol{C}}$, given the expectation values and covariance matrices of $\underline{\boldsymbol{A}}$ and $\underline{\boldsymbol{B}}$. Error propagation yields,

$$\mathsf{c} = \mathsf{G}(\mathsf{a})\,\mathsf{b} \quad \text{and} \quad \begin{aligned} \Sigma_{\mathsf{c},\mathsf{c}} = \;& \bar{\mathsf{G}}(\mathsf{b})\,\Sigma_{\mathsf{a},\mathsf{a}}\,\bar{\mathsf{G}}(\mathsf{b})^{\mathsf{T}} + \mathsf{G}(\mathsf{a})\,\Sigma_{\mathsf{b},\mathsf{b}}\,\mathsf{G}(\mathsf{a})^{\mathsf{T}} \\ &+ \bar{\mathsf{G}}(\mathsf{b})\,\Sigma_{\mathsf{a},\mathsf{b}}\,\mathsf{G}(\mathsf{a})^{\mathsf{T}} + \mathsf{G}(\mathsf{a})\,\Sigma_{\mathsf{b},\mathsf{a}}\,\bar{\mathsf{G}}(\mathsf{b})^{\mathsf{T}}. \end{aligned} \tag{1}$$

Note that this equation is only an approximation. In the case of the geometric product, the exact expression for evaluating the mean of a product of two random variables is $\mathsf{c}^k = \mathsf{a}^i\,\mathsf{b}^j\;\mathsf{G}^k{}_{ij} + \Sigma^{ij}_{\mathsf{a},\mathsf{b}}\;\mathsf{G}^k{}_{ij}$. Furthermore, the exact expression for the covariance matrix $\Sigma_{\mathsf{c},\mathsf{c}}$ is the one given in equation (1) minus the term $(\Sigma^{rs}_{\mathsf{a},\mathsf{b}}\;\mathsf{G}^i{}_{rs})\,(\Sigma^{rs}_{\mathsf{a},\mathsf{b}}\;\mathsf{G}^j{}_{rs})$. That is, if $\underline{\mathsf{a}}$ and $\underline{\mathsf{b}}$ are statistically independent, then equation (1) is the exact expression for the error propagation in all algebra products.

The meaning of the term $\Sigma^{ij}_{\mathsf{a},\mathsf{b}}\;\mathsf{G}^k{}_{ij}$ can be understood when writing the cross-covariance matrix in terms of a singular value decomposition (SVD). Let $\{\mathsf{u}_n\}$ and $\{\mathsf{v}_n\}$ denote the set of left and right singular column vectors of $\Sigma_{\mathsf{a},\mathsf{b}}$, and let the $\{\sigma_n\}$ denote the corresponding set of singular values. Then $\Sigma_{\mathsf{a},\mathsf{b}} = \sum_n\;\sigma_n\,\mathsf{u}_n\,\mathsf{v}_n^{\mathsf{T}}$, and thus $\Sigma^{ij}_{\mathsf{a},\mathsf{b}}\;\mathsf{G}^k{}_{ij} = \sum_n\;\sigma_n\;\mathsf{u}^i_n\,\mathsf{v}^j_n\;\mathsf{G}^k{}_{ij}$. That is, the correction term $\Sigma^{ij}_{\mathsf{a},\mathsf{b}}\;\mathsf{G}^k{}_{ij}$ is a linear combination of the geometric products of corresponding left and right singular vectors of $\Sigma^{ij}_{\mathsf{a},\mathsf{b}}$. The order of magnitude of this correction is the sum of the singular values. Similarly, the order of magnitude of the correction to the covariance matrix is the square of the sum of the singular values.

Conformal Space We want to work with uncertain geometric entities and operators in conformal space. However, the initial data we will be given, has almost invariably been measured in Euclidean space. We therefore have to embed the Euclidean data and its uncertainty in conformal space.

Let $\underline{\boldsymbol{a}} \in \mathbb{R}^3$ be a Euclidean random vector variable with covariance matrix $\Sigma_{\boldsymbol{a},\boldsymbol{a}}$, and $\underline{\boldsymbol{A}} \in \mathbb{R}^{4,1}$ be defined by $\underline{\boldsymbol{A}} := \mathcal{K}(\underline{\boldsymbol{a}})$. It may then be shown that $\boldsymbol{A} = \mathcal{E}[\mathcal{K}(\underline{\boldsymbol{a}})] = \boldsymbol{a} + \frac{1}{2}\,\boldsymbol{a}^2\,\mathbf{e}_\infty + \mathbf{e}_o + \frac{1}{2}\,\mathrm{tr}(\Sigma_{\boldsymbol{a},\boldsymbol{a}})\,e_\infty$. Note that by definition of the geometric product $\boldsymbol{a}^2 \equiv \|\boldsymbol{a}\|^2$. Typically the trace of $\Sigma_{\boldsymbol{a},\boldsymbol{a}}$ is negligible compared to $\|\boldsymbol{a}\|^2$, which leaves us with $\boldsymbol{A} = \mathcal{K}(\boldsymbol{a})$. If we denote the Jacobi matrix of $\mathcal{K}$ evaluated at $\boldsymbol{a}$ by $\mathsf{J}_{\mathcal{K}}(\boldsymbol{a})$, then the error propagation equation for the covariance matrix can be written as $\Sigma_{\boldsymbol{A},\boldsymbol{A}} = \mathsf{J}_{\mathcal{K}}(\boldsymbol{a})\;\Sigma_{\boldsymbol{a},\boldsymbol{a}}\;\mathsf{J}_{\mathcal{K}}^{\mathsf{T}}(\boldsymbol{a})$. Denoting by $\mathrm{I} \in \mathbb{R}^{3\times 3}$ the identity matrix and by $\mathsf{a} \in \mathbb{R}^3$ the column component vector of $\boldsymbol{a}$, the Jacobi matrix $\mathsf{J}_{\mathcal{K}}(\boldsymbol{a}) \in \mathbb{R}^{5\times 3}$ is given by $\mathsf{J}_{\mathcal{K}}(\boldsymbol{a}) = [\;\mathrm{I}\;\mathsf{a}\;0\;]^{\mathsf{T}}$.

From the definition of the conformal embedding function $\mathcal{K}$ it follows that $\mathcal{K}$ maps the Euclidean space onto a paraboloid in $\mathbb{R}^{4,1}$, the so called *horosphere* [11]. However, this implies that when we move a vector $\boldsymbol{A} = \mathcal{K}(\boldsymbol{a})$ within the subspace spanned by its covariance matrix $\Sigma_{\boldsymbol{A},\boldsymbol{A}}$, it will no longer exactly represent a point. In fact, the subspace spanned by $\Sigma_{\boldsymbol{A},\boldsymbol{A}}$ is tangential to the horosphere at $\boldsymbol{A}$. For small covariances of $\boldsymbol{A}$ this is still a good approximation. Furthermore, if we only need an affine point ($\boldsymbol{A} \wedge \mathbf{e}_\infty$), then the quadratic component of $\boldsymbol{A}$ is removed and the corresponding covariance matrix gives an exact description of the uncertainty.

Depending on the application, it may or may not be necessary to express entities of the Geometric Algebra of conformal space in Euclidean terms. The only geometric entities that may be projected back directly into Euclidean space are points. However, if the goal is to test geometric relations, then a projection back into Euclidean space is not necessary.

Given a point in conformal space as $\boldsymbol{A} = \alpha^1\,\mathbf{e}_1 + \alpha^2\,\mathbf{e}_2 + \alpha^3\,\mathbf{e}_3 + \alpha^\infty\,\mathbf{e}_\infty + \alpha^o\,\mathbf{e}_o$, the projection operation $\mathcal{K}^{-1}$ back into Euclidean space is given by $\mathcal{K}^{-1}(\boldsymbol{A}) = \boldsymbol{a}/\alpha^o$, where $\boldsymbol{a} := \alpha^1\,\mathbf{e}_1 + \alpha^2\,\mathbf{e}_2 + \alpha^3\,\mathbf{e}_3$. That is, $\mathbf{e}_o$ takes on the function of the homogeneous dimension. If we again denote the component vector of $\boldsymbol{a}$ by a, then the corresponding Jacobi matrix $\mathsf{J}_{\mathcal{K}^{-1}}(\boldsymbol{A}) \in \mathbb{R}^{3\times5}$ is given by $\mathsf{J}_{\mathcal{K}^{-1}}(\boldsymbol{A}) = \frac{1}{\alpha^o}\,[\,\mathsf{I}\;\mathsf{0}\;-\mathsf{a}/\alpha^o\,]$.

Blades and Operators In this section we will show that covariance matrices can be used to describe the uncertainty of blades and operators in Geometric Algebra. The fundamental problem is, that while covariance matrices describe the uncertainty of an entity through a linear subspace, the subspace spanned by entities of the same type may not be linear.

For example, Heuel [2] describes the evaluation of general homographies, by writing the homography matrix H as a vector h and solving for it, given appropriate constraints. It is then also possible to evaluate a covariance matrix $\Sigma_{\mathsf{h},\mathsf{h}}$ for h. While this is fine for general homographies, Heuel also notes that it is problematic for constrained transformations like rotations, since the necessary constraints on h are non-linear. The basic problem here is that the subspace of vectors h that represent rotation matrices, is not linear. Hence, a covariance matrix for h is not well suited to describe the uncertainty of the corresponding rotation matrix.

The question therefore is, whether the representation of geometric entities and operators in Geometric Algebra allows for an uncertainty description via covariance matrices. For example, consider a line $\boldsymbol{L}$, which may be represented in conformal space as $\boldsymbol{L} = \boldsymbol{X} \wedge \boldsymbol{Y} \wedge \mathbf{e}_\infty$ (cf. table 1). The six components of $\boldsymbol{L}$ are the well known Plücker coordinates, which have to satisfy the Plücker condition in order to describe a line. In Geometric Algebra the Plücker condition is equivalent demanding that $\boldsymbol{L}$ is a blade, i.e. it can be factorized into the outer product of three vectors.

If we want to describe the uncertainty of a line $\boldsymbol{L}$ with a covariance matrix, the sum of the component vector of $\boldsymbol{L}$ with any component vector in the linear subspace spanned by the covariance matrix, has to satisfy the Plücker condition. Here we only want to motivate that such a linear subspace can exist. For that purpose suppose that the covariance matrix of $\boldsymbol{X}$ has rank 1 with eigenvector $\boldsymbol{D} \in \mathbb{R}^{4,1}$ and $\boldsymbol{Y}$ is a point without uncertainty. If a scaled version of $\boldsymbol{D}$ is added to $\boldsymbol{X}$, then the $\boldsymbol{L}$ changes according to the following equation.

$$(\boldsymbol{X} + \alpha\,\boldsymbol{D}) \wedge \boldsymbol{Y} \wedge \mathbf{e}_\infty = \boldsymbol{X} \wedge \boldsymbol{Y} \wedge \mathbf{e}_\infty + \alpha\,(\boldsymbol{D} \wedge \boldsymbol{Y} \wedge \mathbf{e}_\infty), \tag{2}$$

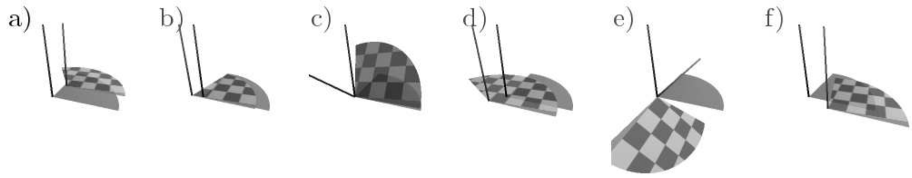

Fig. 1. Effect of adding each of the six eigenvectors of the covariance matrix of a rotor onto the rotor's component vectors. In each of the images, the darker rotor is the initial one.

where $\alpha \in \mathbb{R}$. Thus any scaled version of $\boldsymbol{D} \wedge \boldsymbol{Y} \wedge \mathbf{e}_\infty$ can be added to $\boldsymbol{L}$, such that their sum still satisfies the Plücker condition. Furthermore, $\boldsymbol{D} \wedge \boldsymbol{Y} \wedge \mathbf{e}_\infty$ is the eigenvector of the covariance matrix of $\boldsymbol{L}$.

Since rigid transformation operators also consist of blades, they inherit the same property. For example, a rotor representing a rotation about an arbitrary axis, can be generated by the geometric product of the dual of two planes, that intersect in a line. (If the planes are parallel they result in a translation operator.) The rotation axis is then this intersection line and the rotation angle is twice the angle between the planes. Using error propagation we can in this way construct an uncertain rotor. It turns out that the corresponding covariance matrix can be at most of rank six. The effect on the rotation operation when transforming such an uncertain rotor separately along the six eigenvectors of its covariance matrix is shown in figure 1.

Expressing uncertain transformation operations, like rotation and translation, through elements of the Geometric Algebra of conformal space, therefore offers an advantageous description compared to matrices, since the space of rotation matrices is not linear. In synthetic experiments presented in [6], it was shown that this results in a robust estimation of operators.

Furthermore, note that the sub-algebra of rotors for rotations about the origin, is isomorphic to the quaternion algebra and the sub-algebra of motors is isomorphic to the dual quaternions [12, 13]. Compared to quaternions and dual quaternions, the Geometric Algebra of conformal space allows us not only to describe the operators themselves, but also to apply them to any geometric entity that can be expressed in the algebra. In contrast, when using quaternions only points can be represented by pure quaternions (i.e. no scalar part), and in the dual quaternions only lines can be represented.

4 Applications

In this section we give a number of examples of how uncertain Geometric Algebra can be applied to various problem settings in Computer Vision. The type of problems can be roughly separated into three different categories: construction, estimation and the testing of geometric relations of uncertain entities. For example, given the uncertain optical center of a camera and an uncertain image point, we can construct the uncertain projection ray. On the other hand, given a number of such uncertain projection rays, which should all meet in one point, we can estimate that point and its uncertainty. Alternatively, we could also test the hypothesis that two projection rays meet.

Geometric Entity Estimation A fundamental problem that often occurs is the evaluation of a geometric entity based on the measurement of a number of geometric entities of a different type. For example, suppose we want to find the line $\boldsymbol{L}$ that best fits a given set of points $\{\boldsymbol{X}_n\}$. Additionally, we also want to obtain a covariance matrix for the estimated line. This can be achieved using the Gauss-Helmert (GH) model as described in [6, 2, 9, 10]. The GH-model allows us to evaluate a parameter vector with associated covariance matrix, given a set of data vectors with covariance matrices, a constraint function between data and parameter vectors and possibly a constraint function on the parameters alone. The resultant parameter vector is the solution to a system of linear equations that depends on the Jacobi matrices of the constraint functions, the data and the covariance matrices.

In terms of the GH-model, the parameters are the components l of the line $\boldsymbol{L}$ that is to be estimated, and the data vectors $\{\mathsf{x}_n\}$ are the component vectors of the points $\{\boldsymbol{X}_n\}$. The constraint function $\boldsymbol{Q}(\boldsymbol{X}_n,\, \boldsymbol{L})$ between data and parameters has to be zero if a point lies on the line. The constraint function on the parameters alone $\boldsymbol{H}(\boldsymbol{L})$ has to be zero if $\boldsymbol{L}$ does indeed represent a normalized line, i.e. l satisfies the Plücker condition and $\mathsf{l}^{\mathsf{T}}\,\mathsf{l} = 1$.

In this case $\boldsymbol{Q}(\boldsymbol{X}_n,\, \boldsymbol{L}) = \boldsymbol{X}_n \wedge \boldsymbol{L}$, or $\mathsf{q}^k(\mathsf{x}_n,\, \mathsf{l}) = \mathsf{x}^i_n\, \mathsf{l}^j\, \mathsf{O}^k{}_{ij}$ and $\boldsymbol{H}(\boldsymbol{L}) = \boldsymbol{L}\,\widetilde{\boldsymbol{L}} - 1$, or $\mathsf{h}^k(\mathsf{l}) = \mathsf{l}^{i_1}\, \mathsf{l}^j\, \mathsf{R}^{i_2}{}_j\, \mathsf{G}^k{}_{i_1 i_2} - \delta^k{}_1$, where $\delta^k{}_j$ is the Kronecker delta, and index 1 is assumed to be the index of the scalar component of the corresponding multivector. The Jacobi matrices of q are $\mathsf{Q}^k_{nj} = \mathsf{x}^i_n\, \mathsf{O}^k{}_{ij}$ and $\bar{\mathsf{Q}}^k{}_i = \mathsf{l}^j\, \mathsf{O}^k{}_{ij}$ and the Jacobi matrix of h is $\mathsf{H}^k{}_j = \mathsf{l}^{i_1}\, (\mathsf{R}^{i_2}{}_{i_1}\, \mathsf{G}^k{}_{j i_2} + \mathsf{R}^{i_2}{}_j\, \mathsf{G}^k{}_{i_1 i_2})$. With these definitions of the constraint functions and their Jacobi matrices, we can now apply the GH-model, to evaluate the best uncertain line that fits the given uncertain points.

Table 3 lists the constraint functions $\boldsymbol{Q}$ between geometric entities, that result in a zero vector if one geometric entity is completely contained within the other. For example, the constraint between two lines is only zero if the multivectors describe the same line up to scale. The constraint function $\boldsymbol{H}$ stays the same for all parameter types. Note in particular that instead of fitting a line to a set of points, we can also fit a point to a set of lines. This can, for example, be used for triangulation, where the best intersection of a set of projection rays has to be evaluated. Similarly, the best intersection line of a set of projective planes can found. In table 3, the symbols $\underline{\times}$ and $\overline{\times}$ denote the commutator and anti-commutator product, respectively, which are defined as $\boldsymbol{A}\underline{\times}\boldsymbol{B} = \frac{1}{2}(\boldsymbol{A}\,\boldsymbol{B} - \boldsymbol{B}\,\boldsymbol{A})$ and $\boldsymbol{A}\overline{\times}\boldsymbol{B} = \frac{1}{2}(\boldsymbol{A}\,\boldsymbol{B} + \boldsymbol{B}\,\boldsymbol{A})$.

$\downarrow$ Data, Parameter $\rightarrow$	Point $\boldsymbol{X}$	Line $\boldsymbol{L}$	Plane $\boldsymbol{P}$	Circle $\boldsymbol{C}$	Sphere $\boldsymbol{S}$
Points $\{\boldsymbol{Y}_n\}$	$\boldsymbol{X} \wedge \boldsymbol{Y}_n$	$\boldsymbol{L} \wedge \boldsymbol{Y}_n$	$\boldsymbol{P} \wedge \boldsymbol{Y}_n$	$\boldsymbol{C} \wedge \boldsymbol{Y}_n$	$\boldsymbol{S} \wedge \boldsymbol{Y}_n$
Lines $\{\boldsymbol{K}_n\}$	$\boldsymbol{X} \wedge \boldsymbol{K}_n$	$\boldsymbol{L}\underline{\times}\boldsymbol{K}_n$	$\boldsymbol{P}\overline{\times}\boldsymbol{K}_n$		
Planes $\{\boldsymbol{O}_n\}$	$\boldsymbol{X} \wedge \boldsymbol{O}_n$	$\boldsymbol{L}\overline{\times}\boldsymbol{O}_n$	$\boldsymbol{P}\underline{\times}\boldsymbol{O}_n$		
Circles $\{\boldsymbol{B}_n\}$	$\boldsymbol{X} \wedge \boldsymbol{B}_n$			$\boldsymbol{C}\underline{\times}\boldsymbol{B}_n$	$\boldsymbol{S}\overline{\times}\boldsymbol{B}_n$
Spheres $\{\boldsymbol{R}_n\}$	$\boldsymbol{X} \wedge \boldsymbol{R}_n$			$\boldsymbol{C}\overline{\times}\boldsymbol{R}_n$	$\boldsymbol{S}\underline{\times}\boldsymbol{R}_n$

Table 3. Constraints between data and parameters that are zero if the corresponding geometric entities are contained in one another.

Pose Estimation An important problem in Computer Vision is the estimation of the relative pose of two objects. The simplest instance of this problem is to find the unknown rigid body transformation $\boldsymbol{M}$ that maps a set of points $\{\boldsymbol{X}_n\}$ into the set $\{\boldsymbol{Y}_n\}$, i.e. $\boldsymbol{Y}_n = \boldsymbol{M}\,\boldsymbol{X}_n\,\widetilde{\boldsymbol{M}}$. Since $\boldsymbol{M}\,\widetilde{\boldsymbol{M}} = 1$, the constraint equation is $\boldsymbol{Q}(\boldsymbol{Y}_n, \boldsymbol{M}) = \boldsymbol{M}\,\boldsymbol{X}_n - \boldsymbol{Y}_m\,\boldsymbol{M}$ and in this way gives a linear constraint on $\boldsymbol{M}$. In terms of the parameter vectors this constraint can be written as $\mathsf{Q}(\mathsf{y}_n, \mathsf{m}) = \mathsf{y}_n^j\,\mathsf{m}^r\;\mathsf{Q}^k{}_{jr}$, with $\mathsf{Q}^k{}_{rj} := (\mathsf{x}_n^i\;\mathsf{G}^k{}_{ri} - \mathsf{G}^k{}_{jr})$ and thus an initial solution for m is given by the common right null space of $\mathsf{Q}(\mathsf{y}_n) = \mathsf{y}_n^j\,\mathsf{Q}^k{}_{jr}$ for all n (cf. [3]). When using the GH-model to estimate $\boldsymbol{M}$ and its covariance matrix, then the constraint on $\boldsymbol{M}$ alone is again $\boldsymbol{M}\,\widetilde{\boldsymbol{M}} - 1 = 0$. Experimental results of this method can be found in [6].

A more complicated, but also more interesting case of pose estimation is to fit a given set of model points onto a corresponding set of projection rays. This occurs, when we want to estimate the camera or object pose from a single view of a known object. Let $\boldsymbol{L}_n$ denote the projection ray of the transformed model point $\boldsymbol{M}\,\boldsymbol{X}_n\,\widetilde{\boldsymbol{M}}$, where $\boldsymbol{M}$ denotes the unknown motor. Then the constraint equation is $\boldsymbol{Q}(\boldsymbol{L}_n, \boldsymbol{M}) = \boldsymbol{L}_n \wedge (\boldsymbol{M}\,\boldsymbol{X}_n\,\widetilde{\boldsymbol{M}})$. This equation cannot be made linear in $\boldsymbol{M}$, since $\mathsf{Q}_n(\mathsf{l}_n, \mathsf{m}) = \mathsf{l}_n^{k_1}\;\mathsf{m}^{p_1}\;\mathsf{m}^{q_2}\;\mathsf{Q}^r_{n\;k_1\,p_1\,q_2}$, with

$$\mathsf{Q}^r_{n\;k_1\,p_1\,q_2} = \mathsf{x}_n^{p_2}\;\mathsf{G}^{q_1}{}_{p_1 p_2}\,\mathsf{G}^{k_2}{}_{q_1 q_2}\,\mathsf{O}^r{}_{k_1 k_2}. \tag{3}$$

Thus we also cannot immediately obtain an initial estimate for m from a null space of Q. Nonetheless, we have a constraint equation for the evaluation of a motor, that is only quadratic in the components of the motor, without having made any approximations, like a small angle approximation.

We developed a robust method to evaluate an initial estimate for m using a geometric construction [14]. Alternatively, an initial estimate for m may be given through a tracking assumption. Once an initial estimate for m is known, $\mathsf{Q}_n(\mathsf{l}_n, \mathsf{m})$ may again be used in the GH-model approach. The constraint on $\boldsymbol{M}$ is $\boldsymbol{M}\,\widetilde{\boldsymbol{M}} - 1 = 0$, as before.

We tested this approach on synthetic data in the following way. First random model points were generated and transformed by a "true" rigid transformation. Then a covariance matrix was associated with each transformed model point and error vectors were added to the transformed model points according to their respective covariance matrices. Note that the error vectors were parallel to the image plane. These points were then projected onto a virtual camera. We then estimated the rigid transformation that best mapped the initial model points onto the noisy projection rays using the above described method. The results are shown in table 4. Here μ_r denotes the mean length of the error vectors added to the model points, and μ denotes the mean Euclidean distance between the projection rays and the model points transformed with the true, the initial estimate and the Gauss-Helmert (GH) estimate of the transformation, respectively. The σ columns give the corresponding standard deviations. The values shown are the mean of 800 runs with varying "true" transformations. It can be seen that the Gauss-Helmert approach always leads to good results, which are better than the estimate with the "true" and "initial" transformation. Note that since random vectors were added to the model points, the initially "true" transformation, need not anymore be the best solution.

μ_r	True		Initial		GH	
	μ	σ	μ	σ	μ	σ
0.200	0.227	0.037	0.233	0.045	0.215	0.040
0.283	0.320	0.051	0.330	0.066	0.304	0.055
0.416	0.470	0.074	0.476	0.095	0.441	0.081

Table 4. Results of pose estimation for a synthetic experiment.

↓ Entity →	Line $\boldsymbol{L}$	Circle $\boldsymbol{C}$	Sphere $\boldsymbol{S}$
Line $\boldsymbol{K}$	$\boldsymbol{K}^* \cdot \boldsymbol{L}$	$(\boldsymbol{K}^* \cdot \boldsymbol{C})^2$	$(\boldsymbol{K}^* \cdot \boldsymbol{S})^2$
Circle $\boldsymbol{B}$		$(\boldsymbol{B}^* \cdot \boldsymbol{C})^2$	$(\boldsymbol{B}^* \cdot \boldsymbol{S})^2$
Sphere $\boldsymbol{R}$			$(\boldsymbol{R}^* \cdot \boldsymbol{S})^2$

Table 5. Constraints between geometric entities that yield zero if they intersect in a single point.

Testing Uncertain Geometric Relations Given uncertain geometric entities, a question like "*does point $\boldsymbol{X}$ lie on line $\boldsymbol{L}$*" is not very useful, since the probability that this occurs for ideal points and lines is infinitesimal. We therefore follow the method described by Heuel and Förstner in [2, 1], who apply statistical hypothesis testing as described in [9].

The basic idea is that the hypothesis H_0 "*$\boldsymbol{X}$ lies on $\boldsymbol{L}$*" is tested against the hypothesis H_1 "*$\boldsymbol{X}$ does* not *lie on $\boldsymbol{L}$*". In order to perform the hypothesis test, we need to fix the probability α that we reject H_0 even though it is true. Furthermore, we assume that a vector valued distance measure q with associated covariance matrix $\Sigma_{\mathsf{q},\mathsf{q}}$ is given, which is zero if $\boldsymbol{X}$ is incident with line $\boldsymbol{L}$. Then hypothesis H_0 can be rejected if $\mathsf{q}^{\mathsf{T}}\ \Sigma_{\mathsf{q},\mathsf{q}}^{-1}\ \mathsf{q} > \chi^2_{1-\alpha;n}$, where $\chi^2_{1-\alpha;n}$ is the $(1-\alpha)$-quantile of the χ^2_n distribution for n degrees of freedom. Note that if $\Sigma_{\mathsf{q},\mathsf{q}}$ is not of full rank, its pseudo-inverse can also be used in the above equation.

The distance measure $\boldsymbol{Q}$ for the containment of geometric entities is just given by the constraint equations of table 3. The covariance matrix $\Sigma_{\mathsf{q},\mathsf{q}}$ can then be evaluated with equation (1) using the appropriate Jacobi matrices.

Furthermore, the distance measure $\boldsymbol{Q}$ for the intersection in a single point (not containment as in table 3) is given in table 5. Note that the relation between lines and circles and two circles is also zero, if the entities are co-planar. Also, note that if a plane and a sphere intersect in a single point, the plane is tangential to the sphere. That is, we can also test tangentiality in this way.

In terms of the component vectors we have, for example, for two lines $\mathsf{q}^k(\mathsf{k},\,\mathsf{l}) = \mathsf{k}^i\,\mathsf{l}^{j_2}\ \ \mathsf{D}^{j_1}{}_i\,\mathsf{N}^k{}_{j_1 j_2}$, with Jacobi matrices $\mathsf{Q}^k{}_{j_2}(\mathsf{k}) = \mathsf{k}^i\ \ \mathsf{D}^{j_1}{}_i\,\mathsf{N}^k{}_{j_1 j_2}$ and $\bar{\mathsf{Q}}^k{}_i(\mathsf{l}) = \mathsf{l}^{j_2}\ \mathsf{D}^{j_1}{}_i\,\mathsf{N}^k{}_{j_1 j_2}$, which can be used in equation (1) to evaluate $\Sigma_{\mathsf{q},\mathsf{q}}$. For line and circle we have

$$\mathsf{q}^s(\mathsf{k},\,\mathsf{c}) = \mathsf{w}^{r_1}(\mathsf{k},\,\mathsf{c})\,\mathsf{w}^{r_2}(\mathsf{k},\,\mathsf{c})\ \mathsf{G}^s{}_{r_1 r_2}, \qquad \mathsf{w}^k(\mathsf{k},\,\mathsf{c}) = \mathsf{k}^i\,\mathsf{c}^{j_2}\ \mathsf{D}^{j_1}{}_i\,\mathsf{N}^k{}_{j_1 j_2}. \tag{4}$$

When evaluating the covariance matrix for $\mathsf{q}(\mathsf{k},\,\mathsf{c})$ one also has to include the cross-correlation part of equation (1) with cross-correlation matrix $\Sigma_{\mathsf{w},\mathsf{w}}$ in the calculation.

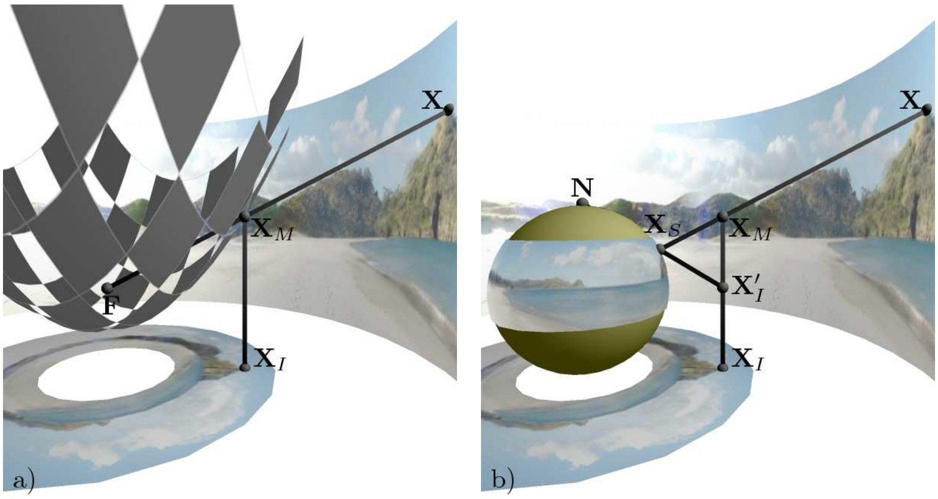

Fig. 2. a) Projection on a parabolic mirror and b) its mathematical representation as stereographic projection.

Projective Camera A central aspect of Computer Vision is the projection of points and lines onto the image plane of a projective camera and also the reconstruction of points and lines in 3D-space from their projections.

The projection of a point $\boldsymbol{X}$ onto the image plane $\boldsymbol{P}_A$ of a camera with optical center $\boldsymbol{A}$ can be evaluated as the intersection of the projective ray $\boldsymbol{A} \wedge \boldsymbol{X} \wedge \mathbf{e}_\infty$ with $\boldsymbol{P}_A$. The projected point $\boldsymbol{X}_A$ is then given by $\boldsymbol{X}_A = (\boldsymbol{A} \wedge \boldsymbol{X} \wedge \mathbf{e}_\infty) \cdot \boldsymbol{P}_A^*$. Note that this description of a camera is intimately related to the corresponding camera matrix as is shown in [15]. Using this formula we can immediately evaluate the projection of an uncertain point, whereby also an uncertainty of the camera basis can be accounted for. Note that the resultant projected point is an affine point as described in section 2.

Conversely, if we are given an uncertain image point $\boldsymbol{X}_A$ (as a standard point), and we would like to estimate the corresponding uncertain projection ray $\boldsymbol{L}$, we can use the relation $\boldsymbol{L} = \boldsymbol{A} \wedge \boldsymbol{X}_A \wedge \mathbf{e}_\infty$. If we assume that $\boldsymbol{A}$ is a certain point, then this becomes, in terms of the component vectors, $\mathsf{l} = \mathsf{K}\,\mathsf{x}_A$ and $\Sigma_{\mathsf{l},\mathsf{l}} = \mathsf{K}\;\Sigma_{\mathsf{x}_A,\mathsf{x}_A}\;\mathsf{K}^\mathsf{T}$, with

$$\mathsf{K}^k{}_{i_2} = \mathsf{a}^{i_1}\,\mathsf{e}_\infty^{j_2}\;\mathsf{O}^{j_1}{}_{i_1 i_2}\,\mathsf{O}^k{}_{j_1 j_2}, \tag{5}$$

Note that $\mathsf{K} \in \mathbb{R}^{6\times 5}$, since x_A contains the five components of a standard point and l the six Plücker coordinates of the projective ray. An uncertain projection ray evaluated in this way may, for example, be used in the pose estimation approach described above.

Catadioptric Camera We now show how the projection ray related to an image point in a catadioptric camera with a parabolic mirror can be constructed using Geometric Algebra. Figure 2a shows the basic setup of a catadioptric imaging system with a parabolic mirror. A light ray emanating from point $\boldsymbol{X}$ in the world that would pass through the focal point $\boldsymbol{F}$ of a parabolic mirror (shown with a half-transparent checkered texture), is reflected down at point $\boldsymbol{X}_M$ with direction parallel to the axis of the parabolic mirror. If below the mirror a projective camera is placed focused to infinity, then an image as shown in the figure is generated. Schematically we can replace the projective camera with an orthogonal one, and then obtain image point $\boldsymbol{X}_I$ from world point $\boldsymbol{X}$.

In [16], Geyer and Daniilidis show that this type of image generation can mathematically be modeled as shown in figure 2b. The world point $\boldsymbol{X}$ is projected onto a unit sphere, centered on the focal point of the parabolic mirror, thus generating $\boldsymbol{X}_S$. A stereographic projection of $\boldsymbol{X}_S$ then results in $\boldsymbol{X}'_I$, which lies on the plane bisecting the sphere perpendicular to the parabolic mirror's axis. Projecting $\boldsymbol{X}'_I$ parallel to the parabolic mirror's axis, then generates the same image point $\boldsymbol{X}_I$ as before.

We found that the stereographic projection of the latter method can be replaced by an inversion in the sphere centered on $\boldsymbol{N}$ with radius $\sqrt{2}$. This allows us to perform the following geometric construction using the Geometric Algebra of conformal space. Suppose we are given an image point $\boldsymbol{X}_I$ with an associated covariance matrix and we would like to evaluate the corresponding uncertain projection ray passing through the focal point of the parabolic mirror $\boldsymbol{F}$ and $\boldsymbol{X}_S$. First of all, we can move $\boldsymbol{X}_I$ to $\boldsymbol{X}'_I$ without the need for error propagation. If $\boldsymbol{S}$ represents the inversion sphere centered on $\boldsymbol{N}$ with radius $\sqrt{2}$, then $\boldsymbol{X}_S = \boldsymbol{S}^* \, \boldsymbol{X}'_I \, \boldsymbol{S}^*$. The projection ray $\boldsymbol{L}$ is then given by $\boldsymbol{L} = \boldsymbol{F} \wedge \mathbf{e}_\infty \wedge \boldsymbol{X}_S = \boldsymbol{F} \wedge \mathbf{e}_\infty \wedge (\boldsymbol{S}^* \, \boldsymbol{X}'_I \, \boldsymbol{S}^*)$. Again we can apply standard error propagation to obtain the covariance matrix of $\boldsymbol{L}$.

If we assume that $\boldsymbol{F}$ and $\boldsymbol{S}$ are ideal, that is they are not regarded as uncertain entities, then $\boldsymbol{L}$ and its covariance matrix can be evaluated from $\boldsymbol{X}'_I$ via matrix multiplications using the corresponding component vectors. Let e_∞, f, s, l and x_I denote the component vectors of $\mathbf{e}_\infty$, $\boldsymbol{F}$, $\boldsymbol{S}^*$, $\boldsymbol{L}$ and $\boldsymbol{X}'_I$, respectively. Then $\mathsf{l} = \mathsf{K}\,\mathsf{x}_I$ and $\Sigma_{\mathsf{l},\mathsf{l}} = \mathsf{K}\,\Sigma_{\mathsf{x}_I,\mathsf{x}_I}\,\mathsf{K}^\mathsf{T}$, where

$$\mathsf{K}^r{}_{k_2} = \mathsf{f}^{i_1}\, \mathsf{e}_\infty^{i_2}\, \mathsf{s}^{k_1}\, \mathsf{s}^{l_2} \;\; \mathsf{G}^{l_1}{}_{k_1 k_2}\, \mathsf{O}^{j_1}{}_{i_1 i_2}\, \mathsf{G}^{j_2}{}_{l_1 l_2}\, \mathsf{O}^r{}_{j_1 j_2}. \tag{6}$$

Note that $\mathsf{K} \in \mathbb{R}^{6\times 5}$, since l contains the six Plücker coordinates of the projective ray and x_I the five components of a standard point in conformal space. Again, an uncertain projection ray evaluated in this way may be used in the pose estimation approach described above.

5 Conclusions

We have presented a unifying framework for the description of uncertain geometry and kinematics. It was shown that the Geometric Algebra of conformal space can be applied to many important applications of Computer Vision and can deal with the invariably occurring uncertainties of geometric entities and transformations, in an appropriate way.

A result of particular importance is that covariance matrices can appropriately represent the uncertainty of algebra entities that represent transformations. This is, for example, not possible for rotation matrices, since orthogonal matrices do not span a linear subspace.

Furthermore, a novel pose estimation approach was introduced, which is quadratic in the components of the transformation, without having made any approximations.

A uniform framework for geometric entity fitting and triangulation and the testing of uncertain geometric relations was presented. Finally, the treatment of catadioptric cameras with parabolic mirrors within this framework was discussed. The main result here was that the construction of projection rays from image points, which is needed for pose estimation, can be achieved by a simple matrix multiplication for projective and catadioptric cameras.

We believe these results show that a combination of an algebraic description of geometric problems, with a linear algebra approach to their numerical solution, offers a valuable framework for the treatment of many Computer Vision applications.

References

1. Förstner, W., Brunn, A., Heuel, S.: Statistically testing uncertain geometric relations. In Sommer, G., Krüger, N., Perwass, C., eds.: Mustererkennung 2000. Informatik Aktuell, Springer, Berlin (2000) 17–26
2. Heuel, S.: Uncertain Projective Geometry. Volume 3008 of LNCS. Springer (2004)
3. Perwass, C., Sommer, G.: Numerical evaluation of versors with Clifford algebra. In Dorst, L., Doran, C., Lasenby, J., eds.: Applications of Geometric Algebra in Computer Science and Engineering, Birkhäuser (2002) 341–349
4. Perwass, C., Förstner, W.: Uncertain geometry with circles, spheres and conics. In Klette, R., Kozera, R., Noakes, L., Weickert, J., eds.: Geometric Properties from Incomplete Data. Volume 31 of Computational Imaging and Vision. Springer-Verlag (2006) 23–41
5. Rosenhahn, B., Sommer, G.: Pose estimation in conformal geometric algebra, part I: The stratification of mathematical spaces. Journal of Mathematical Imaging and Vision **22** (2005) 27–48
6. Perwass, C., Gebken, C., Sommer, G.: Estimation of geometric entities and operators from uncertain data. In: 27. Symposium für Mustererkennung, DAGM 2005, Wien, 29.8.-2.9.005. Number 3663 in LNCS, Springer-Verlag, Berlin, Heidelberg (2005)
7. Perwass, C., Hildenbrand, D.: Aspects of geometric algebra in Euclidean, projective and conformal space. Technical Report Number 0310, CAU Kiel, Institut für Informatik (2003)
8. Hestenes, D., Sobczyk, G.: Clifford Algebra to Geometric Calculus: A Unified Language for Mathematics and Physics. Reidel, Dordrecht (1984)
9. Koch, K.R.: Parameter Estimation and Hypothesis Testing in Linear Models. Springer (1997)
10. Mikhail, E., Ackermann, F.: Observations and Least Squares. University Press of America, Lanham, MD20706, USA (1976)
11. Li, H., Hestenes, D., Rockwood, A.: Generalized Homogeneous Coordinates for Computational Geometry. In: Geometric Computing with Clifford Algebras. Springer, Berlin, Heidelberg (2001) 27–59
12. Clifford, W.K.: Preliminary sketch of bi-quaternions. In: Proceedings of the London Mathematical Society. Volume 4. (1873) 381–395
13. Daniilidis, K.: Using the Algebra of Dual Quaternions for Motion Alignment. In: Geometric Computing with Clifford Algebras. Springer, Berlin, Heidelberg (2001) 489–500
14. Gebken, C., Perwass, C., Buchholz, S., Sommer, G.: A robust geometrical solution to pose estimation using geometric algebra. In: submitted to ECCV 2006. (2006)
15. Perwass, C.: Applications of Geometric Algebra in Computer Vision. PhD thesis, Cambridge University (2000)
16. Geyer, C., Daniilidis, K.: Catadioptric projective geometry. International Journal of Computer Vision (2001) 223–243