Chapter 1
Introduction

Geometric algebra is currently not a widespread mathematical tool in the fields of computer vision, robot vision, and robotics within the engineering sciences, where standard vector analysis, matrix algebra, and, at times, quaternions are mainly used. The prevalent reason for this state of affairs is probably the fact that geometric algebra is typically not taught at universities, let alone at high-school level, even though it appears to be *the* mathematical language for geometry. This unfortunate situation seems to be due to two main aspects. Firstly, geometric algebra combines many mathematical tools that were developed separately over the past 200-odd years, such as the standard vector analysis, Grassmann's algebra, Hamilton's quaternions, complex numbers, and Pauli matrices. To a certain extent, teaching geometric algebra therefore means teaching all of these concepts at once. Secondly, most applications in two- and three-dimensional space, which are the most common spaces in engineering applications, can be dealt with using standard vector analysis and matrix algebra, without the need for additional tools. The goal of this text is thus to demonstrate that geometric algebra, which combines geometric transformations with the construction and intersection of geometric entities in a single framework, can be used advantageously in the analysis and solution of engineering applications.

Matrix algebra, or linear algebra in general, probably represents the most versatile mathematical tool in the engineering sciences. In fact, any geometric algebra can be represented in matrix form, or, more to the point, geometric algebra is a particular subalgebra of general tensor algebra (see e.g. [148]). However, this constraint can be an advantage, as for example in the case of quaternions, which form a subalgebra of geometric algebra. While rotations about an axis through the origin in 3D space can be represented by 3×3 matrices, it is a popular method to use quaternions instead, because their components are easier to interpret (direction of rotation axis and rotation angle) and, with only four parameters, they are a nearly minimal parameterization. Given the four components of a quaternion, the corresponding (scaling) rotation is uniquely determined, whereas the three Euler angles from

C. Perwass, *Geometric Algebra with Applications in Engineering.*
Geometry and Computing.

which a rotation matrix can be constructed do not suffice by themselves. It is also important to define in which order the three rotations about the three basis axes are executed, in order to obtain the correct rotation matrix. It is also not particularly intuitive what type of rotation three Euler angles represent. This is the reason why, in computer graphics software libraries such as OpenGL [23, 22], rotations are always given in terms of a rotation axis and a rotation angle. Internally, this is then transformed into the corresponding rotation matrices.

Apart from the obvious interpretative advantages of quaternions, there are clear numerical advantages, at the cost that only rotations can be described, whereas matrices can represent any linear function. For example, two rotations are combined by multiplying two quaternions or two rotation matrices. The product of two quaternions can be represented by the product of a 4×4 matrix with a 4×1 vector, while in the case of a matrix representation two 3×3 matrices have to be multiplied. That is, the former operation consists of 16 multiplications and 12 additions, while the latter needs 27 multiplications and 18 additions. Furthermore, when one is solving for a rotation matrix, two additional constraints need to be imposed on the nine matrix components: matrix orthogonality and scale. In the case of quaternions, the orthogonality constraint is implicit in the algebraic structure and thus does not have to be imposed explicitly. The only remaining constraint is therefore the quaternion scale.

The quaternion example brings one of the main advantages of geometric algebra to the fore: by reducing the representable transformations from all (multi)linear functions to a particular subset, for example rotation, a more optimal parameterization can be achieved and certain constraints on the transformations are embedded in the algebraic structure. In other words, the group structure of a certain set of matrices is made explicit in the algebra. On the downside, this implies that only a subset of linear transformations is directly available. Clearly, any type of function, including linear transformations, can still be defined on algebraic entities, but not all functions profit in the same way from the algebraic structure.

The above discussion gives an indication of the type of problems where geometric algebra tends to be particularly beneficial: situations where only a particular subset of transformations and/or geometric entities are present. The embedding of appropriate constraints in the algebraic structure can then lead to descriptive representations and optimized numerical constraints.

One area where the embedding of constraints in the algebraic structure can be very valuable is the field of artificial neural networks, or classification algorithms in general. Any classification algorithm has to make some assumptions about the structure of the feature space or, rather, the form of the separation boundaries between areas in the feature space that belong to different classes. Choosing the best basis functions (kernels) for such a separation can improve the classification results considerably. geometric algebra offers, through its algebraic structure, methods to advantageously implement such basis functions, in particular for geometric constraints. This has been shown, for example, by Buchholz and Sommer [27, 29, 28] and by Bayro-Corrochano and Buchholz [17]. Banarer, Perwass, and Sommer have shown, furthermore, that hyperspheres as represented in the geometric algebra of conformal space are effective basis functions for classifiers [134, 14, 15]. In this text, however, these aspects will not be detailed further.

1.1 History

Before discussing further aspects of geometric algebra, it is helpful to look at its roots. Geometric algebra is basically just another name for Clifford algebra, a name that was introduced by David Hestenes to emphasize the geometric interpretation of the algebra. He first published his ideas in a refined version of his doctoral thesis in 1966 [87]. A number of books extending this initial "proof of concept" (as he called it) followed in the mid 1980s and early 1990s [91, 88, 92]. Additional information on his current projects can be found on the website [90].

Clifford algebra itself is about 100 years older. It was developed by William K. Clifford (1845–1879) in 1878 [37, 35]. A collection of all his papers can be found in [36]. Clifford's main idea was that the *Ausdehnungslehre* of Hermann G. Grassmann (1809–1877) and the quaternion algebra of William R. Hamilton (1805–1865) could be combined in a single geometric algebra, which was later to be known as *Clifford algebra*. Unfortunately, Clifford died very young and had no opportunity to develop his algebra further. In around 1881 the physicist Josiah W. Gibbs (1839–1903) developed a method that made it easier to deal with vector analysis, which advanced work on Maxwell's electrodynamics considerably. This was probably one of the reasons why Gibbs's vector analysis became the standard mathematical tool for physicists and engineers, instead of the more general but also more elaborate Clifford algebra.

Within the mathematics community, the Clifford algebra of a quadratic module was a well-established theory by the end of the 1970s. See, for example, the work of O'Meara [129] on the theory over fields or the work of Baeza [13] on the theory over rings. The group structure of Clifford algebra was detailed by, for example, Porteous [148] and Gilbert and Murray [79] in the 1990s. The latter also discussed the relation between Clifford algebra and Dirac operators, which is one of the main application areas of Clifford algebra in physics.

The Dirac equation of quantum mechanics has a natural representation in Clifford algebra, since the Pauli matrices that appear in it form an algebra that is isomorphic to a particular Clifford algebra. Some other applications of Clifford algebra in physics are to Maxwell's equations of electrodynamics, which have a very concise representation, and to a flat-space theory of gravity

[46, 106]. An introduction to geometric algebra for physicists was published by Doran and Lasenby in 2003 [45].

The applications of Clifford algebra or geometric algebra in engineering were initially based on the representation of geometric entities in projective space, as introduced by Hestenes and Ziegler in [92]. These were, for example, applications to projective invariants (see e.g. [107, 108]) and multiple-view geometry (see e.g. [18, 145, 144]). Initial work in the field of robotics used the representation of rotation operators in geometric algebra (see e.g. [19, 161]). The first collections of papers discussing applications of geometric algebra in engineering appeared in the mid 1990s [16, 48, 164, 165]. The first design of a geometric algebra coprocessor and its implementation in a field-programmable gate array (FPGA) was developed by Perwass, Gebken, and Sommer in 2003 [137].

In 2001 Hongbo Li, David Hestenes, and Alyn Rockwood published three articles in [164] introducing the *conformal model* [116], spherical conformal geometry [117], and a universal model for conformal geometries [118]. These articles laid the foundation for a whole new class of applications that could be treated with geometric algebra. Incidentally, Pierre Anglés had already developed this representation of the conformal model independently in the 1980s [6, 7, 8] but, apparently, it was not registered by the engineering community. His work on conformal space can also be found in [9].

The conformal model is based on work by Friedrich L. Wachter (1792–1817), a student of J. Carl F. Gauss (1777–1855), who showed that a certain surface in hyperbolic geometry was metrically equivalent to Euclidean space. Forming a geometric algebra over a homogeneous embedding of this space extends the representation of points, lines, planes, and rotations about axes through the origin to point pairs, circles, spheres, and conformal transformations, which include all Euclidean transformations. Note that the representation of Euclidean transformations in the conformal model is closely related to biquaternions, which had been investigated by Clifford himself in 1873 [34], a couple of years before he developed his algebra. The extended set of basic geometric entities in the conformal model and the additionally available transformations made geometric algebra applicable to a larger set of application areas, for example pose estimation [155, 152], a new type of artificial neural network [15], and the description of space groups [89].

Although the foundations of the conformal model were laid by Li, Hestenes, and Rockwood in 2001, its various facets, properties, and extensions are still a matter of ongoing research (see e.g. [49, 105, 140, 160]). One of the aims of this text is to present the conformal model in the context of other geometric algebra models and to give a detailed derivation of the model itself, as well as in-depth discussions of its geometric entities and transformation operators. Even though the conformal model is particularly powerful, it is presented as one special case of a general method for representing geometry and transformations with geometric algebra. One result of this more general outlook is the geometric algebra of conic space, which is published here in all of its details for the first time. In this geometric algebra, the algebraic entities represent all types of projective conic sections.

1.2 Geometry

One of the novel features of the discussion of geometric algebra in this text is the explicit separation of the algebra from the representation of geometry through algebraic entities. In the author's opinion this is an advantageous view, since it clarifies the relation between the various geometric models and indicates how geometric algebra may be developed for new geometric models. The basic idea that blades represent linear subspaces through their null space has already been noted by Hestenes. However, the consequent explicit application of this notion to the discussion of different geometric models and algebraic operations was first developed in [140] and is extended in this text. Note that the notion of representing geometry through null spaces is directly related to *affine varieties* in algebraic geometry [38].

One conclusion that can be drawn from this view of geometric algebra is that there exist only three fundamental operations in the algebra, all based on the same algebraic product: "addition", "subtraction", and reflection of linear subspaces. All geometric entities, such as points, lines, planes, spheres, circles, and conics, are represented through linear subspaces, and all transformations, such as rotation, inversion, translation, and dilation, are combinations of reflections of linear subspaces. Nonlinear geometric entities such as spheres and nonlinear transformations such as inversions stem from a particular embedding of Euclidean space in a higher-dimensional embedding space, such that linear subspaces in the embedding space represent nonlinear subspaces in the Euclidean space, and combinations of reflections in the embedding space represent nonlinear transformations in the Euclidean space. This aspect is detailed further in Chap. 4, where a number of geometries are discussed in detail.

The fact that all geometric entities and all transformation operations are constructed through fundamental algebraic operations leads to two key properties of geometric algebra:

1. Geometric entities and transformation operators are constructed in exactly the same way, independent of the dimension of the space they are constructed in.
2. The intersection operation and the transformation operators are the same for all geometric entities in all dimensions.

For example, it is shown in Sect. 4.2 that in the geometric algebra of the projective space of Euclidean 3D space $\mathbb{R}^3$, a vector $\boldsymbol{A}$ represents a point. The *outer product* (see Sect. 3.2.2) of two vectors $\boldsymbol{A}$ and $\boldsymbol{B}$ in this space, denoted by $\boldsymbol{A} \wedge \boldsymbol{B}$, then represents the line through $\boldsymbol{A}$ and $\boldsymbol{B}$. If $\boldsymbol{A}$ and $\boldsymbol{B}$

are vectors in the projective space of $\mathbb{R}^{10}$, say, they still represent points and $\boldsymbol{A} \wedge \boldsymbol{B}$ still represents the line through these points.

The intersection operation in geometric algebra is called the *meet* and is denoted by $\vee$. If $\boldsymbol{A}$ and $\boldsymbol{B}$ represent any two geometric entities in any dimension, then their intersection is always determined with the meet operation as $\boldsymbol{A} \vee \boldsymbol{B}$. Note that even if the geometric entities represented by $\boldsymbol{A}$ and $\boldsymbol{B}$ have no intersection, the meet operation results in an algebraic entity. Typically this entity then represents an imaginary geometric object or one that lies at infinity. This property of the intersection operation has the major advantage that no additional checks have to be performed before the intersection of two entities is computed.

The application of transformations is similarly dimension-independent. For example, if $\boldsymbol{R}$ is an algebraic entity that represents a rotation and $\boldsymbol{A}$ represents any geometric entity, then $\boldsymbol{R}\,\boldsymbol{A}\,\boldsymbol{R}^{-1}$ represents the rotated geometric entity, independent of its type and dimension. Note that juxtaposition of two entities denotes the algebra product, the *geometric product*.

To a certain extent, it can therefore be said that geometric algebra allows a coordinate-free representation of geometry. Hestenes emphasized this property by developing the algebra with as little reference to a basis as possible in [87, 91]. However, since in those texts the geometric algebra over real-valued vector spaces is discussed, a basis can always be found, but it is not essential for deriving the properties of the algebra. The concept of a basis-independent geometric algebra was extended even further by Frank Sommen in 1997. He constructed a Clifford algebra completely without the notion of a vector space [162, 163]. In this approach abstract basic entities, called *vector-variables*, are considered, which are not entities of a particular vector space. Instead, these vector-variables are characterized purely through their algebraic properties. An example of an application of this *radial algebra* is the construction of Clifford algebra on super-space [21].

The applications considered in this text, however, are all related to particular vector spaces, which allows a much simpler construction of geometric algebra. The manipulation of analytic expressions in geometric algebra is still mostly independent of the dimensionality of the underlying vector space or any particular coordinates.

1.3 Outlook

From a mathematical point of view, geometric algebra is an elegant and analytically powerful formalism to describe geometry and geometric transformations. However, the particular aim of the engineering sciences is the development of solutions to problems in practical applications. The tools that are used to achieve that aim are only of interest insofar as the execution speed and the accuracy of solutions are concerned. Nevertheless, this does not preclude research into promising mathematical tools, which could lead to a substantial gain in application speed and accuracy or even new application areas. At the end of the day, though, a mathematical tool will be judged by its applicability to practical problems. In this context, the question has to be asked:

What are the advantages of geometric algebra?

Or, more to the point, *when should geometric algebra be used?* There appears to be no simple, generally applicable answer to this question. Earlier in this introduction, some pointers were given in this context. Basically, geometric algebra has three main properties:

1. Linear subspaces can be represented in arbitrary dimensions.
2. Subspaces can be added, subtracted, and intersected.
3. Reflections of subspaces in each other can be performed.

Probably the most important effects that these fundamental properties have are the following:

- With the help of a non-linear embedding of Euclidean space, it is possible to represent non-linear subspaces and transformations. This allows a (multi)linear representation of circles and spheres, and non-linear transformations such as inversions.
- The combination of the basic reflection operations results in more complex transformations, such as rotation, translation, and dilation. The corresponding transformation operators are nearly minimal parameterizations of the transformation. Additional constraints, such as the orthogonality of a rotation matrix, are encoded in the algebraic structure.
- Owing to the dimension-independent representation of geometric entities, only a single intersection operation is needed to determine the intersections between arbitrary combinations of entities. This can be a powerful analytical tool for geometrical constructions.
- The dimension-independent representation of geometric entities and reflections also has the effect that a transformation operator can be applied to any element in the algebra in any dimension, be it a geometric entity or another transformation.
- Since all transformations are represented as multilinear operations, the uncertainty of Gaussian distributed transformation operators can be effectively represented by covariance matrices. That is, *the uncertainty of geometric entities and transformations can be represented in a unified fashion.*

These properties are used in Chap. 6 to construct and estimate uncertain geometric entities and transformations. In Chap. 7, the availability of a linearized inversion operator leads to a unifying camera model. Chapter 8 exploits the encoding of transformation constraints in the algebraic structure. In Chap. 9, the dimension independence of the reflection operation leads to an immediate extension of Pythagorean-hodograph curves to arbitrary dimensions. And Chap. 10 demonstrates the usefulness of representing linear subspaces in the Hilbert space of random variables.

It is hoped that the subjects discussed in this text will help researchers to identify advantageous applications of geometric algebra in their field.

1.4 Overview of This Text

This section gives an overview of the main contributions of this text in the context of research in geometric algebra and computer vision. The main aim of this text is to give a detailed presentation of and, especially, to develop new tools for the three aspects that are necessary to apply geometric algebra to engineering problems:

1. **Algebra**, the mathematical formalism.
2. **Geometry**, the representation of geometry and transformations.
3. **Numerics**, the implementation of numerical solution methods.

This text gives a thorough description of geometric algebra, presents the geometry of the geometric algebra of Euclidean, projective, conformal, and a novel conic space in great detail, and introduces a novel numerical calculation method for geometric algebra that incorporates the notion of random algebra variables. The methodology presented combines the representative power of the algebra and its effective algebraic manipulations with the demands of real-life applications, where uncertain data is unavoidable.

A number of applications where this methodology is used are presented, which include the description of uncertain geometric entities and transformations, a novel camera model, and monocular pose estimation with uncertain data. In addition, applications of geometric algebra to some special polynomial curves (Pythagorean-hodograph curves), and the geometric algebra over the Hilbert space of random variables are presented.

In addition to the mathematical contributions, the software tool CLUCalc, developed by the author, is introduced. CLUCalc is a stand-alone software program that implements geometric algebra calculations and, more importantly, can visualize the geometric content of algebraic entities automatically. It is therefore an ideal tool to help in the learning and teaching of geometric algebra.

In the remainder of this section, the main aspects of all chapters are detailed.

1.4.1 CLUCalc

The software tool CLUCALC (Chap. 2) [133] was developed by the author with the aim of furthering the understanding, supporting the teaching, and implementing applications of geometric algebra. For this purpose, a whole new programming language, called CLUSCRIPT, was developed, which was honed for the programming of geometric algebra expressions. For example, most operator symbols typically used in geometric algebra are available in CLUSCRIPT. However, probably the most useful feature in the context of geometric algebra is the automatic visualization of the geometric content of multivectors. That is, the user need not know what a multivector represents in order to draw it. Instead, the meaning of multivectors and the effect of algebraic operations can be *discovered* interactively. Since geometric algebra is all about geometry, CLUSCRIPT offers simple access to powerful visualization features, such as transparent objects, lighting effects, texture mapping, animation, and user interaction. It also supports the annotation of drawings using LaTeX text, which can also be mapped onto arbitrary surfaces. Note that virtually all of the figures in this text were created with CLUCALC, and the various applications presented were implemented in CLUSCRIPT.

1.4.2 Algebra

One aspect that has been mentioned before is the clear separation of algebraic entities and their geometric interpretation. The foundation for this view is laid in Chap. 3 by introducing the inner- and outer-product null spaces. This concept is extended to the *geometric* inner- and outer-product null spaces in Chap. 4 on geometries, to give a general methodology of how to represent geometry by geometric algebra. Note that this concept is very similar to *affine varieties* in algebraic geometry [38].

Another important aspect that is treated explicitly in Chap. 3 is that of null vectors and null blades, which is something that is often neglected. Through the definition of an algebra conjugation, a *Euclidean scalar product* is introduced, which, together with a corresponding definition of the magnitude of an algebraic entity, or *multivector*, allows the definition of a Hilbert space over a geometric algebra. While this aspect is not used directly, algebra conjugation is essential in the general definition of subspace addition and subtraction, which eventually leads to factorization algorithms for blades and versors that are also valid for null blades and null versors. Furthermore, a *pseudoinverse* of null blades is introduced. The relevance of these operations is very high when working with the conformal model, since geometric entities are represented by blades of null vectors. In order to determine the meet, i.e. the general intersection operation, between arbitrary blades of null vectors, a factorization algorithm for such blades has to be available.

Independent of the null-blade aspect, a novel set of algorithms that are essential when implementing geometric algebra on a computer are presented. This includes, in particular, the evaluation of the *join* of blades, which is necessary for the calculation of the meet. In addition, the versor factorization algorithm is noteworthy. This factorizes a general transformation into a set of reflections or, in the case of the conformal model, inversions.

1.4.3 Geometries

While the representation of geometric objects and transformations through algebraic entities is straightforward, extracting the geometric information from the algebraic entities is not trivial. For example, in the conformal model, two vectors $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ can represent two spheres and their outer product $\boldsymbol{C} = \boldsymbol{S}_1 \wedge \boldsymbol{S}_2$ the intersection circle of the two spheres (see Sect. 4.3.4). Extracting the circle parameters center, normal, and radius from the algebraic entity $\boldsymbol{C}$ is not straightforward. Nevertheless, a knowledge of how this can be done is essential if the algebra is to be used in applications.

Therefore, the analysis of the geometric interpretation of algebraic entities that represent elements of 3D Euclidean space is discussed in some detail. A somewhat more abstract discussion of the geometric content of algebraic entities in the conformal model of arbitrary dimension can be found in [116]. Confining the discussion in this text to the conformal model of 3D Euclidean space simplifies the formulas considerably.

Another important aspect of Chap. 4 is a discussion of how incidence relations between geometric entities are represented through algebraic operations. This knowledge is pivotal when one is expressing geometric constraints in geometric algebra.

A particularly interesting contribution is the introduction of the conic space, which refers to the geometric algebra over the vector space of reduced symmetric matrices. In the geometric algebra of conic space, the outer product of five vectors represents the conic section that passes through the corresponding five points. The outer product of four points represents a point quadruplet, which is also the result of the meet of two five-blades, i.e. the intersection of two conic sections. The discussion of conic space starts out with an even more general outlook, whereby the conic and conformal spaces are particular subspaces of the general polynomial space.

1.4.4 Numerics

The essential difference in the treatment of numerical calculation with geometric algebra between this text and the standard approach introduced by Hestenes is that algebraic operations in geometric algebra are regarded here as bilinear functions and expressed through tensor contraction. This approach was first introduced by the author and Sommer in [146]. While the standard approach has its merits in the analytical description of derivatives as algebraic entities, the tensorial approach allows the direct application of (multi)linear optimization algorithms. At times, the tensorial approach also leads to solution methods that cannot be easily expressed in algebraic terms.

An example of the latter case is the versor equation (see Sect. 5.2.2). Without delving into all the details, the problem comes down to solving for the multivector $\boldsymbol{V}$, given multivectors $\boldsymbol{A}$ and $\boldsymbol{B}$, the equation

$$\boldsymbol{V}\,\boldsymbol{A} - \boldsymbol{B}\,\boldsymbol{V} = 0\,.$$

The problem is that it is impossible to solve for $\boldsymbol{V}$ through algebraic manipulations, since the various multivectors typically do not commute. However, in the tensorial representation of this equation, it is straightforward to solve for the components of $\boldsymbol{V}$ by evaluating the null space of a matrix.

In the standard approach a function $\boldsymbol{C}(\boldsymbol{V})$, say, would be defined as

$$\boldsymbol{C} \;:\; \boldsymbol{V} \mapsto \boldsymbol{V}\,\boldsymbol{A} - \boldsymbol{B}\,\boldsymbol{V}\,.$$

The solution to $\boldsymbol{V}$ is then the vector $\widehat{\boldsymbol{V}}$, that minimizes $\Delta(\widehat{\boldsymbol{V}}) := \boldsymbol{C}(\widehat{\boldsymbol{V}}) \cdot \widetilde{\boldsymbol{C}}(\widehat{\boldsymbol{V}})$. To evaluate $\widehat{\boldsymbol{V}}$, the derivative of $\Delta(\widehat{\boldsymbol{V}})$ with respect to $\widehat{\boldsymbol{V}}$ has to be calculated, which can be done with the multivector derivative introduced by Hestenes [87, 91] (see Sect. 3.6). Then a standard gradient descent method or something more effective can be used to find $\widehat{\boldsymbol{V}}$. For an example of this approach, see [112]. It is shown in Sect. 5.2.2, however, that the tensorial approach also minimizes $\Delta(\widehat{\boldsymbol{V}})$; this method is much easier to apply.

Another advantage of the tensorial approach is that Gaussian distributed random multivector variables can be treated directly. This use of the tensorial approach was developed by the author in collaboration with W. Förstner and presented at a Dagstuhl workshop in 2004 [136]. The present text extends this and additional publications [76, 138, 139] to give a complete and thorough discussion of the subject area. The two main aspects with respect to random multivector variables, whose foundations are laid in Chap. 5, are the construction and estimation of uncertain geometric entities and uncertain *transformations* from uncertain data. For example, an uncertain circle can be constructed from the outer product of three uncertain points, and an uncertain rotation operator may be constructed through the geometric product of two uncertain reflection planes.

The representation of uncertain transformation operators is certainly a major advantage of geometric algebra over a matrix representation. This is demonstrated in Sect. 5.6.1, where it is shown that the variation of a random transformation multivector in the conformal model, such as a rotation operator, lies (almost) in a linear subspace. A covariance matrix is therefore well suited to representing the uncertainty of a Gaussian distributed random transformation multivector variable. It appears that the representation of uncertain transformations with matrices is more problematic. Heuel investigated an approach whereby transformation matrices are written as column vectors and their uncertainty as an associated covariance matrix [93]. He notes that this method is numerically not very stable.

A particularly convincing example is the representation of rotations about the origin in 3D Euclidean space. Corresponding transformation operators in the conformal model lie in a linear subspace, which also forms a subalgebra, and a subgroup of the Clifford group. Rotation matrices, which are elements of the orthogonal group, do not form a subalgebra at the same time. That is, the sum of two rotation matrices does not, in general, result in a rotation matrix. A covariance matrix on a rotation matrix can therefore only represent a tangential uncertainty.

Instead of constructing geometric entities and transformation operators from uncertain data, they can also be *estimated* from a set of uncertain data. For this purpose, a linear least-squares estimation method, the Gauss–Helmert model, is presented (see Sect. 5.9), which accounts for uncertainties in measured data. Although this is a well-known method, a detailed derivation is given here, to hone the application of this method to typical geometric-algebra problems.

In computer vision, the measured data consists typically of the locations of image points. Owing to the digitalization in CCD chips and/or the point spread function (PSF) of the imaging system, there exists an unavoidable uncertainty in the position measurement. Although this uncertainty is usually small, such small variations can lead to large deviations in a complex geometric construction. Knowing the final uncertainty of the elements that constitute the data used in an estimation may be essential to determining the reliability of the outcome.

One such example is provided by a catadioptric camera, i.e. a camera with a 360 degree view that uses a standard projective camera which looks at a parabolic mirror. While it can be assumed that the uncertainties in the position of all pixels in the camera are equal, the uncertainty in the corresponding projection rays reflected at the parabolic mirror varies considerably depending on where they hit the mirror. Owing to the linearization of the inversion operation in the conformal model, which can be used to model reflection in a parabolic mirror (see Sect. 7.2), simple error propagation can be used here to evaluate the final uncertainty of the projection rays. These uncertain rays may then form the data that is used to solve a pose estimation problem (see Chap. 8).

1.4.5 Uncertain Geometric Entities and Operators

In Chap. 6, the tools developed in Chap. 5 are applied to give examples of the construction and estimation of uncertain geometric entities and transformation operators. Here the construction of uncertain lines, circles, and conic sections from uncertain points is visualized, to show that the standard-deviation envelopes of such geometric entities are not simple surfaces. The use of a covariance matrix should always be favored over simple approximations such as a tube representing the uncertainty of a line.

Furthermore, the effect of an uncertain reflection and rotation on an ideal geometric entity is shown. This has direct practical relevance, for example, in the evaluation of the uncertainty induced in a light ray which is reflected off an uncertain plane.

With respect to the estimation of geometric entities and transformation operators, standard problems that occur in the application of geometric algebra to computer vision problems are presented, and it is shown how the methods introduced in Chap. 5 can be used to solve them. In addition, the metrics used implicitly in these problems are investigated. This includes the first derivation of the point-to-circle metric and the versor equation metric.

The quality of the estimation methods presented is demonstrated in two experiments: the estimation of a circle and of a rotation operator. Both experiments demonstrate that the Gauss–Helmert method gives better results than a simple null-space estimation. The estimation of the rotation operator is also compared with a standard method, where it turns out that the Gauss–Helmert method gives better and more stable results.

Another interesting aspect is hypothesis testing, where questions such as "does a point lie on a line?" are answered in a statistical setting. Here, the first visualization of what this means for the question "does a point lie on a circle?" in the conformal model is given (see Fig. 6.10).

1.4.6 The Inversion Camera Model

The inversion camera model is a novel camera model that was first published by Perwass and Sommer in [147]. In Chap. 7 an extended discussion of this camera model is given, which is applied in Chap. 8 to monocular pose estimation. The inversion camera model combines the pinhole camera model, a lens distortion model, and the catadioptric-camera model for the case of a parabolic mirror. All these configurations can be obtained by varying the position of a focal point and an inversion sphere. Since inversion can be represented through a linear operator in the conformal model, the geometric algebra of conformal space offers an ideal framework for this camera model. The unification of three camera models that are usually treated separately can lead to generalized constraint equations, as is the case for monocular pose estimation. Note that the camera model is represented by a transformation operator in the geometric algebra of conformal space, which implies that it can be easily associated with a covariance matrix that represents its uncertainty.

1.4.7 Monocular Pose Estimation

In Chap. 8 an application is presented that uses all aspects of the previously presented methodology: a geometric setup is translated into geometric-algebra expressions, algebraic operations are used to express a geometric constraint, and a solution is found using the tensorial approach, which incorporates uncertain data. While the pose estimation problem itself is well known, a number of novel aspects are introduced in this chapter:

- a simple and robust method to find an initial pose,
- the incorporation of the inversion camera model into the pose constraint,
- a pose constraint equation that is quadratic in the components of the pose operator without making any approximations, and
- the covariance matrix for the pose operator and the inversion camera model operator.

The pose estimation method presented is thoroughly tested against ground truth pose data generated with a robotic arm for a number of different imaging systems.

1.4.8 Versor Functions

In Chap. 9 some instances of versor functions are discussed, that is, functions of the type $\mathcal{F} : t \mapsto \boldsymbol{A}(t)\, \boldsymbol{N}\, \widetilde{\boldsymbol{A}}(t)$. One geometric interpretation of this type of function is that a *preimage* $\boldsymbol{A}(t)$ scales and rotates a vector $\boldsymbol{N}$. It is shown that cycloidal curves generated by coupled motors, Fourier series of complex-valued functions, and Pythagorean-hodograph (PH) curves are all related to this form.

In the context of PH curves, a new representation based on the reflection of vectors is introduced, and it is shown that this is equivalent to the standard quaternion representation in the case of cubic and quintic PH curves. This novel representation has the advantage that it can be immediately extended to arbitrary dimensions, and it gives a geometrically more intuitive representation of the degrees of freedom. This also leads to the identification of parameter subsets that generate PH curves of constant length but of different shape. The work on PH curves resulted from a collaboration of the author with R. Farouki [135].

1.4.9 Random Variable Space

Chap. 10 gives an example of a geometric algebra over a general Hilbert space, instead of a real-valued vector space. The Hilbert space chosen here is that of random variables. This demonstrates how the geometric concepts of geometric algebra can be applied to function spaces. Some fundamental properties of random variables are presented in this context. This leads to a straightforward derivation of the Cauchy–Schwarz inequality and an extension of the correlation coefficient to an arbitrary number of random variables. The geometric insight into geometric algebra operations gained for the Euclidean space can be applied directly to random variables, which gives properties such as the expectation value, the variance, and the correlation a direct geometric meaning.

1.5 Overview of Geometric Algebra

In this section, a mathematical overview of some of the most important aspects of geometric algebra is presented. The purpose is to give those readers who are not already familiar with geometric algebra a gentle introduction to the main concepts before delving into the detailed mathematical analysis. To keep it simple, only the geometric algebra of Euclidean 3D space is used in this introduction and formal mathematical proofs are avoided. A detailed introduction to geometric algebra is given in Chap. 3.

One problem in introducing geometric algebra is that depending on the reader's background, different introductions are best suited. For pure mathematicians the texts on Clifford algebra by Porteous [148], Gilbert and Murray [79], Lounesto [119], Riesz [149], and Ablamowicz et al. [3], to name just a few, are probably most instructive. In this context, the text by Lounesto presenting counterexamples of theorems in Clifford algebra should be of interest [120].

The field of Clifford analysis is only touched upon when basic multivector differentiation is introduced in Sect. 3.6. This is sufficient to deal with the simple functions of multivectors that are encountered in this text. Thorough treatments of Clifford analysis can be found in [24, 41, 42].

This text is geared towards the use of geometric algebra in engineering applications and thus stresses more those aspects that are related to the representation of geometry and Euclidean transformations. It may thus not treat all aspects that the reader is interested in. As always, the best way is to read a number of different introductions to geometric algebra. After the books by Hestenes [87, 88, 91], there are a number of papers and books geared towards various areas, in particular physics and engineering. For physics-related introductions see, for example [45, 83, 110], and for engineering-related introductions [47, 48, 111, 140, 50, 172].

1.5.1 Basics of the Algebra

The real-valued 3D Euclidean vector space is denoted by $\mathbb{R}^3$, with an orthonormal basis $\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3 \in \mathbb{R}^3$. That is,

$$\boldsymbol{e}_i * \boldsymbol{e}_j = \delta_{ij} \;, \qquad \delta_{ij} := \begin{cases} 1 : \; i = j \;, \\ 0 : \; i \neq j \;, \end{cases}$$

where $*$ denotes the standard scalar product. The geometric algebra of $\mathbb{R}^3$ is denoted by $\mathbb{G}(\mathbb{R}^3)$, or simply $\mathbb{G}_3$. Its algebra product is called the *geometric product*, and is denoted by the juxtaposition of two elements. This is just as in matrix algebra, where the matrix product of two matrices is represented by juxtaposition of two matrix symbols. The effect of the geometric product on the basis vectors of $\mathbb{R}^3$ is

$$\boldsymbol{e}_i \, \boldsymbol{e}_j = \begin{cases} \boldsymbol{e}_i * \boldsymbol{e}_j : \; i = j \;, \\ \qquad \boldsymbol{e}_{ij} : \; i \neq j \;. \end{cases} \tag{1.1}$$

One of the most important points to note for readers who are new to geometric algebra is the case when $i \neq j$, where $\boldsymbol{e}_{ij} \equiv \boldsymbol{e}_i \, \boldsymbol{e}_j$ represents a *new algebraic element*, sometimes also denoted by $\boldsymbol{e}_{ij}$ for brevity. Similarly, if $i, j, k \in \{1, 2, 3\}$ are three different indices, $\boldsymbol{e}_{ijk} := \boldsymbol{e}_i \, \boldsymbol{e}_j \, \boldsymbol{e}_k$ is yet another new algebraic entity. That this process of creating new entities cannot be continued indefinitely is ensured by defining the geometric product to be associative and to satisfy $\boldsymbol{e}_i \, \boldsymbol{e}_i = 1$. The latter is also called the *defining equation* of the algebra. For example,

$$(\boldsymbol{e}_i \, \boldsymbol{e}_j \, \boldsymbol{e}_k) \, \boldsymbol{e}_k = (\boldsymbol{e}_i \, \boldsymbol{e}_j) \, (\boldsymbol{e}_k \, \boldsymbol{e}_k) = (\boldsymbol{e}_i \, \boldsymbol{e}_j) \, 1 = \boldsymbol{e}_i \, \boldsymbol{e}_j \;.$$

In Chap. 3 it is shown that (1.1) together with the standard axioms of an associative algebra suffices to show that

$$\boldsymbol{e}_i \, \boldsymbol{e}_j = -\boldsymbol{e}_j \, \boldsymbol{e}_i \qquad \text{if} \quad i \neq j \;. \tag{1.2}$$

This implies, for example, that

$$(\boldsymbol{e}_i \, \boldsymbol{e}_j \, \boldsymbol{e}_k) \, \boldsymbol{e}_j = (\boldsymbol{e}_i \, \boldsymbol{e}_j) \, (\boldsymbol{e}_k \, \boldsymbol{e}_j) = -(\boldsymbol{e}_i \, \boldsymbol{e}_j) \, (\boldsymbol{e}_j \, \boldsymbol{e}_k) = -\boldsymbol{e}_i \, (\boldsymbol{e}_j \, \boldsymbol{e}_j) \, \boldsymbol{e}_k = -\boldsymbol{e}_i \, \boldsymbol{e}_k \;.$$

From these basic rules, the basis of $\mathbb{G}_3$ can be found to be

$$\overline{\mathbb{G}}_3 := \left\{ \, 1, \, \boldsymbol{e}_1, \, \boldsymbol{e}_2, \, \boldsymbol{e}_3, \, \boldsymbol{e}_{12}, \, \boldsymbol{e}_{13}, \, \boldsymbol{e}_{23}, \, \boldsymbol{e}_{123} \, \right\} . \tag{1.3}$$

In general, the dimension of the geometric algebra of an n-dimensional vector space is 2^n.

1.5.2 General Vectors

The operations introduced in the previous subsection are valid only for a set of orthonormal basis vectors. For general vectors of $\mathbb{R}^3$, the algebraic operations have somewhat more complex properties, which can be related to the properties of the basis vectors by noting that any vector $\boldsymbol{a} \in \mathbb{R}^3$ can be written as

$$\boldsymbol{a} := a^1\, \boldsymbol{e}_1 + a^2\, \boldsymbol{e}_2 + a^3\, \boldsymbol{e}_3 \;.$$

Note that the scalar components of the vector $\boldsymbol{a}$, a^1, a^2, a^3, are indexed by a superscript index. This notation will be particularly helpful when algebra operations are expressed as tensor contractions. In particular, the *Einstein summation convention* can be used, which states that a subscript index repeated as a superscript index within a product implies a summation over the range of the index. That is,

$$\boldsymbol{a} := a^i\, \boldsymbol{e}_i \equiv \sum_{i=1}^{3} a^i\, \boldsymbol{e}_i \;.$$

It is instructive to consider the geometric product of two vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^3$ with $\boldsymbol{a} := a^i\, \boldsymbol{e}_i$ and $\boldsymbol{b} := b^i\, \boldsymbol{e}_i$. Using the multiplication rules of the geometric product between orthonormal basis vectors as introduced in the previous subsection, it is straightforward to find

$$\begin{aligned} \boldsymbol{a}\,\boldsymbol{b} = \;& (a^1\, b^1 + a^2\, b^2 + a^3\, b^3) \\ &+ (a^2\, b^3 - a^3\, b^2)\, \boldsymbol{e}_{23} \\ &+ (a^3\, b^1 - a^1\, b^3)\, \boldsymbol{e}_{31} \\ &+ (a^1\, b^2 - a^2\, b^1)\, \boldsymbol{e}_{12} \;. \end{aligned} \tag{1.4}$$

Recall that $\boldsymbol{e}_{23}$, $\boldsymbol{e}_{31}$ and $\boldsymbol{e}_{12}$ are new basis elements of the geometric algebra $\mathbb{G}_3$. Because these basis elements contain two basis vectors from the vector space basis of $\mathbb{R}^3$, they are said to be of *grade* 2. Hence, the expression

$$(a^2\, b^3 - a^3\, b^2)\, \boldsymbol{e}_{23} + (a^3\, b^1 - a^1\, b^3)\, \boldsymbol{e}_{31} + (a^1\, b^2 - a^2\, b^1)\, \boldsymbol{e}_{12}$$

is said to be a vector of grade 2, because it is a linear combination of the basis elements of grade 2.

The sum of scalar products in (1.4) is clearly the standard scalar product of the vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, i.e. $\boldsymbol{a} * \boldsymbol{b}$. The grade 2 part can also be evaluated separately using the *outer product*. The outer product is denoted by $\wedge$ and defined as

$$\boldsymbol{e}_i \wedge \boldsymbol{e}_j = \begin{cases} 0 : & i = j \\ \boldsymbol{e}_i\, \boldsymbol{e}_j : & i \neq j \end{cases} \tag{1.5}$$

Since $\boldsymbol{e}_i \wedge \boldsymbol{e}_j = \boldsymbol{e}_i \, \boldsymbol{e}_j$ if $i \neq j$, it is clear that $\boldsymbol{e}_i \wedge \boldsymbol{e}_j = -\boldsymbol{e}_j \wedge \boldsymbol{e}_i$. Also, note that the expression $\boldsymbol{e}_i \wedge \boldsymbol{e}_i = 0$ is the defining equation for the Grassmann, or exterior, algebra (see Sect. 3.8.4). Therefore, by using the outer product, the Grassmann algebra is recovered in geometric algebra.

Using this definition of the outer product, it is not too difficult to show that

$$\boldsymbol{a} \wedge \boldsymbol{b} = (a^2\, b^3 - a^3\, b^2)\, \boldsymbol{e}_{23} + (a^3\, b^1 - a^1\, b^3)\, \boldsymbol{e}_{31} + (a^1\, b^2 - a^2\, b^1)\, \boldsymbol{e}_{12} \;,$$

which is directly related to the standard vector cross product of $\mathbb{R}^3$, since

$$\boldsymbol{a} \times \boldsymbol{b} = (a^2\, b^3 - a^3\, b^2)\, \boldsymbol{e}_1 + (a^3\, b^1 - a^1\, b^3)\, \boldsymbol{e}_2 + (a^1\, b^2 - a^2\, b^1)\, \boldsymbol{e}_3 \;.$$

To show how one expression can be transformed into the other, the concept of the *dual* has to be introduced. To give an example, the dual of $\boldsymbol{e}_1$ is $\boldsymbol{e}_2\, \boldsymbol{e}_3$, that is, the geometric product of the remaining two basis vectors. This can be interpreted in geometric terms by saying that the dual of the subspace parallel to $\boldsymbol{e}_1$ is the subspace perpendicular to $\boldsymbol{e}_1$, which is spanned by $\boldsymbol{e}_2$ and $\boldsymbol{e}_3$.

A particularly powerful feature of geometric algebra is that this dual operation can be expressed through the geometric product. For this purpose, the *pseudoscalar* $\boldsymbol{I}$ of $\mathbb{G}_3$ is defined as the basis element of highest grade, i.e. $\boldsymbol{I} := \boldsymbol{e}_1\, \boldsymbol{e}_2\, \boldsymbol{e}_3$. Using again the rules of the geometric product, it follows that the inverse pseudoscalar is given by $\boldsymbol{I}^{-1} = \boldsymbol{e}_3\, \boldsymbol{e}_2\, \boldsymbol{e}_1$, such that $\boldsymbol{I}\, \boldsymbol{I}^{-1} = \boldsymbol{I}^{-1}\, \boldsymbol{I} = 1$. The dual of some multivector $\boldsymbol{A} \in \mathbb{G}_3$ is denoted by $\boldsymbol{A}^*$ and can be evaluated via

$$\boldsymbol{A}^* = \boldsymbol{A}\, \boldsymbol{I}^{-1} \;.$$

It now follows that

$$(\boldsymbol{a} \wedge \boldsymbol{b})^* = (\boldsymbol{a} \wedge \boldsymbol{b})\, \boldsymbol{I}^{-1} = \boldsymbol{a} \times \boldsymbol{b} \;.$$

What does this show? First of all, it shows that the important vector cross product can be recovered in geometric algebra. Even more importantly, it implies that the vector cross product is only a special case, for a three-dimensional vector space, of a much more general operation: the outer product. Whereas the expression $\boldsymbol{a} \wedge \boldsymbol{b}$ is a valid operation in any dimension greater than or equal to two, the vector cross product is defined only in three dimensions.

From the geometric interpretation of the dual, the geometric relation between the vector cross product and the outer product can be deduced. The vector cross product of $\boldsymbol{a}$ and $\boldsymbol{b}$ represents a vector perpendicular to those two vectors. Hence, the outer product $\boldsymbol{a} \wedge \boldsymbol{b}$ represents the plane spanned by $\boldsymbol{a}$ and $\boldsymbol{b}$. It may actually be shown that with an appropriate definition of the magnitude of multivectors, the magnitude $\|\boldsymbol{a} \wedge \boldsymbol{b}\|$ is the area of the parallelogram spanned by $\boldsymbol{a}$ and $\boldsymbol{b}$, as illustrated in Fig. 1.1 (see Sect. 4.1.1).

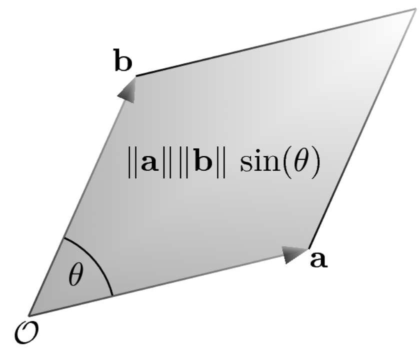

Fig. 1.1 The magnitude of a blade $\boldsymbol{a}\wedge\boldsymbol{b}$ is the area of the parallelogram spanned by $\boldsymbol{a}$ and $\boldsymbol{b}$

One important equation that follows from the above analysis is the relation between the geometric, the scalar and the outer product:

$$\boldsymbol{a}\,\boldsymbol{b} = \boldsymbol{a} * \boldsymbol{b} + \boldsymbol{a} \wedge \boldsymbol{b}\ . \tag{1.6}$$

That is, the geometric product of two vectors results in the sum of a scalar and a grade 2 vector (also called a *bivector*). Note that this is true only for vectors of grade 1. While it may appear strange at first that this is a sum of different types of elements (scalars and bivectors), it is simply a linear combination of elements of the algebraic basis. Adding elements of different types is quite natural, for example, when working with complex numbers, where real and imaginary numbers are added.

It is interesting to note that, in fact, certain subalgebras of geometric algebra are isomorphic to complex numbers. This can be seen quite easily, by first evaluating the square of the bivector $\boldsymbol{e}_1\,\boldsymbol{e}_2 \in \mathbb{G}_3$:

$$(\boldsymbol{e}_{12})^2 = \boldsymbol{e}_{12}\,\boldsymbol{e}_{12} = -(\boldsymbol{e}_1\,\boldsymbol{e}_2)\,(\boldsymbol{e}_2\,\boldsymbol{e}_1) = -\boldsymbol{e}_1\,(\boldsymbol{e}_2\,\boldsymbol{e}_2)\,\boldsymbol{e}_1 = -(\boldsymbol{e}_1\,\boldsymbol{e}_1) = -1\ .$$

This shows that bivectors in geometric algebra square to minus 1. In fact, there are many such entities. Using the simple rules of the geometric product for orthonormal basis vectors, it can be easily shown that algebraic entities of the type $\boldsymbol{w} = a + b\,\boldsymbol{e}_{12}$ behave in the same way as complex numbers $w = a + \mathsf{i}\,b$, where $\mathsf{i} := \sqrt{-1}$ denotes the imaginary unit (see Sect. 3.8.2).

1.5.3 Geometry

So far, mainly the algebraic properties of geometric algebra have been discussed. In this subsection, the notion of how geometry is represented through algebraic entities is introduced. The basic idea is to associate an algebraic entity with the null space that it generates with respect to a particular operation. One operation that is particularly useful for this purpose is the outer product; this stems from the fact that

$$\boldsymbol{x} \wedge \boldsymbol{a} = 0 \qquad \Longleftrightarrow \qquad \boldsymbol{x} = u\, \boldsymbol{a}\ ,\ \forall\, u \in \mathbb{R}\ ,$$

where $\boldsymbol{x}, \boldsymbol{a} \in \mathbb{R}^3$. Similarly, it can be shown that for $\boldsymbol{x}, \boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^3$,

$$\boldsymbol{x} \wedge \boldsymbol{a} \wedge \boldsymbol{b} = 0 \qquad \Longleftrightarrow \qquad \boldsymbol{x} = u\, \boldsymbol{a} + v\, \boldsymbol{b}\ ,\ \forall\, u, v \in \mathbb{R}\ .$$

In this way, $\boldsymbol{a}$ can be used to represent the line through the origin in the direction of $\boldsymbol{a}$ and $\boldsymbol{a} \wedge \boldsymbol{b}$ to represent the plane through the origin spanned by $\boldsymbol{a}$ and $\boldsymbol{b}$. Later on in the text, this will be called the *outer-product null space* (see Sect. 3.2.2). This concept is extended in Chap. 4 to allow the representation of arbitrary lines, planes, circles, and spheres.

For example, in the geometric algebra of the projective space of $\mathbb{R}^3$, vectors represent points in Euclidean space and the outer product of two vectors represents the line passing through the points represented by those vectors. Similarly, the outer product of three vectors represents the plane through the three corresponding points (see Sect. 4.2).

In the geometric algebra of conformal space, there exist two special points: the point at infinity, denoted by $\boldsymbol{e}_\infty$, and the origin $\boldsymbol{e}_o$. Vectors in this space again represent points in Euclidean space. However, the outer product of two vectors $\boldsymbol{A}$ and $\boldsymbol{B}$, say, represents the corresponding point pair. The line through the points $\boldsymbol{A}$ and $\boldsymbol{B}$ is represented by $\boldsymbol{A} \wedge \boldsymbol{B} \wedge \boldsymbol{e}_\infty$, that is, the entity that passes through the points $\boldsymbol{A}$ and $\boldsymbol{B}$ and infinity. Furthermore, the outer product of three vectors represents the circle through the three corresponding points, and similarly for spheres (see Sect. 4.3).

Another important product that has not been mentioned yet is the inner product, denoted by $\cdot$. The inner product of two vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^3$ is equivalent to the scalar product, i.e.

$$\boldsymbol{a} \cdot \boldsymbol{b} = \boldsymbol{a} * \boldsymbol{b}\ .$$

However, the inner product of a grade 1 and a grade 2 vector results in a grade 1 vector and not a scalar. That is, given three vectors $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c} \in \mathbb{R}^3$, then

$$\boldsymbol{x} := \boldsymbol{a} \cdot (\boldsymbol{b} \wedge \boldsymbol{c})$$

is a grade 1 vector. It is shown in Sect. 3.2.7 that this equation can be expanded into

$$\boldsymbol{x} = (\boldsymbol{a} \cdot \boldsymbol{b})\, \boldsymbol{c} - (\boldsymbol{a} \cdot \boldsymbol{c})\, \boldsymbol{b} = (\boldsymbol{a} * \boldsymbol{b})\, \boldsymbol{c} - (\boldsymbol{a} * \boldsymbol{c})\, \boldsymbol{b}\ .$$

Using this expansion, it is easy to verify that

$$\boldsymbol{a} * \boldsymbol{x} = 0 \qquad \Longleftrightarrow \qquad \boldsymbol{a} \perp \boldsymbol{x} \,,$$

i.e. $\boldsymbol{a}$ is perpendicular to $\boldsymbol{x}$. The geometric meaning of this property is that the inner product removes a linear subspace. That is, while the outer product combines linear subspaces, the inner product subtracts them. Using these operations, a general intersection operation, the *meet*, can be defined (see Sect. 3.2.12). Using the meet, the intersection between any pair of geometric entities can be evaluated.

1.5.4 Transformations

It was mentioned earlier that the fundamental transformation available in geometric algebra is reflection. All other transformations, such as rotation, inversion, and translation, are represented through combinations of this basic transformation. In Sect. 3.3 it is shown that given vectors $\boldsymbol{a}, \boldsymbol{n} \in \mathbb{R}^3$, the vector

$$\boldsymbol{b} := \boldsymbol{n}\,\boldsymbol{a}\,\boldsymbol{n}^{-1}$$

is the reflection of $\boldsymbol{a}$ in $\boldsymbol{n}$, as indicated in Fig. 1.2.

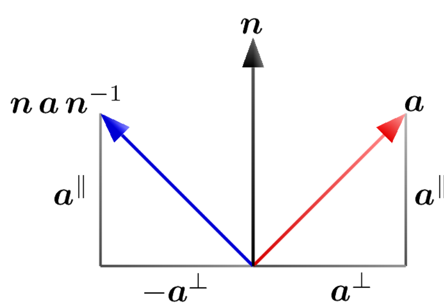

Fig. 1.2 Reflection of $\boldsymbol{a}$ in $\boldsymbol{n}$

The conformal space of the Euclidean space $\mathbb{R}^1$ is a good example of how the reflection operation can represent a non-linear transformation. The first step in constructing the conformal space is to embed the Euclidean space $\mathbb{R}^1$ as the unit circle $\mathbb{S}^1 \subset \mathbb{R}^2$ in $\mathbb{R}^2$. Owing to the particular embedding of $\mathbb{R}^1$, a reflection in $\mathbb{R}^2$ has the effect of an inversion in $\mathbb{R}^1$, as indicated in Fig. 1.3 (see Sect. 4.3.9). Here $\boldsymbol{e}_1$ and $\boldsymbol{e}_+$ denote orthonormal basis vectors of $\mathbb{R}^2$, and $\mathcal{S}$ denotes the embedding operator from $\mathbb{R}^1$ into $\mathbb{R}^2$. That is,

$$\boldsymbol{x}^{-1} = \mathcal{S}^{-1}\Big(\, \boldsymbol{e}_1\, \mathcal{S}(\boldsymbol{x})\, \boldsymbol{e}_1 \,\Big) \,.$$

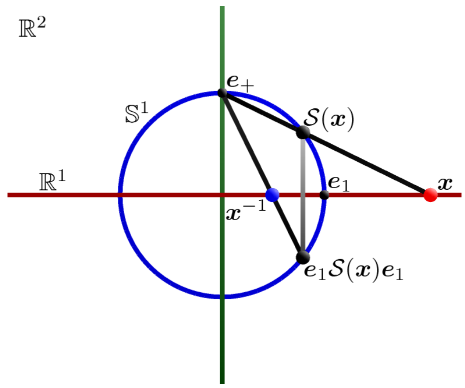

Fig. 1.3 Reflection of $\mathcal{S}(\boldsymbol{x})$ in $\boldsymbol{e}_1$ represents inversion of $\boldsymbol{x}$

Returning again to Euclidean space, given unit vectors $\hat{\boldsymbol{n}}, \hat{\boldsymbol{m}} \in \mathbb{S}^2 \subset \mathbb{R}^3$, where $\mathbb{S}^2$ denotes the unit sphere in $\mathbb{R}^3$, the consecutive reflection of $\boldsymbol{a}$ in $\hat{\boldsymbol{n}}$ and $\hat{\boldsymbol{m}}$ can be evaluated via

$$\boldsymbol{b} = \hat{\boldsymbol{m}}\,\hat{\boldsymbol{n}}\;\boldsymbol{a}\;\hat{\boldsymbol{n}}\,\hat{\boldsymbol{m}}\;.$$

The effect of this double reflection is a rotation of $\boldsymbol{a}$ in the plane spanned by $\hat{\boldsymbol{n}}$ and $\hat{\boldsymbol{m}}$ by twice the angle between $\hat{\boldsymbol{n}}$ and $\hat{\boldsymbol{m}}$, as shown in Fig. 1.4 (see Sect. 4.1.5).

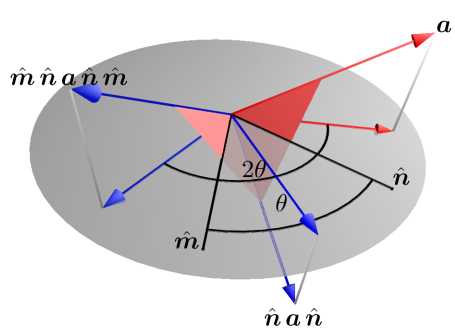

Fig. 1.4 Rotation of $\boldsymbol{a}$ by consecutive reflection in $\hat{\boldsymbol{n}}$ and $\hat{\boldsymbol{m}}$

The product $\hat{\boldsymbol{m}}\,\hat{\boldsymbol{n}}$ may therefore be interpreted as a rotation operator, which is typically called a *rotor*. Defining $\boldsymbol{R} := \hat{\boldsymbol{m}}\,\hat{\boldsymbol{n}}$, the rotation of $\boldsymbol{a}$ can be written as $\boldsymbol{b} = \boldsymbol{R}\,\boldsymbol{a}\,\widetilde{\boldsymbol{R}}$ where $\widetilde{\boldsymbol{R}}$ denotes the *reverse* of $\boldsymbol{R}$, which is defined as $\widetilde{\boldsymbol{R}} := \hat{\boldsymbol{n}}\,\hat{\boldsymbol{m}}$. From the relation between the geometric product and the scalar and outer products as given in (1.6), it follows that

$$\boldsymbol{R} = \hat{\boldsymbol{m}}\,\hat{\boldsymbol{n}} = \hat{\boldsymbol{m}} * \hat{\boldsymbol{n}} + \hat{\boldsymbol{m}} \wedge \hat{\boldsymbol{n}} = \cos\theta + \sin\theta\,\frac{\hat{\boldsymbol{m}} \wedge \hat{\boldsymbol{n}}}{\|\hat{\boldsymbol{m}} \wedge \hat{\boldsymbol{n}}\|}\;, \tag{1.7}$$

because $\|\hat{\boldsymbol{m}} \wedge \hat{\boldsymbol{n}}\| = \sin\theta$, if $\theta = \angle(\hat{\boldsymbol{m}}, \hat{\boldsymbol{n}})$ (see Sect. 4.1.1). It was shown earlier that a bivector $\boldsymbol{e}_1\,\boldsymbol{e}_2$, for example, squares to -1. The same is true for any unit bivector such as $\hat{\boldsymbol{U}} := (\hat{\boldsymbol{m}} \wedge \hat{\boldsymbol{n}})/\|\hat{\boldsymbol{m}} \wedge \hat{\boldsymbol{n}}\|$, i.e. $\hat{\boldsymbol{U}}^2 = -1$. Identifying $\hat{\boldsymbol{U}} \cong \mathsf{i}$, where i denotes the imaginary unit $\mathsf{i} = \sqrt{-1}$, it is clear that (1.7) can be written as

$$\cos\theta + \sin\theta\,\hat{\boldsymbol{U}} \cong \cos\theta + \sin\theta\,\mathsf{i} = \exp(\theta\,\mathsf{i})\;.$$

Extending the definition of the exponential function to geometric algebra, it may thus be shown that the rotor given in (1.7) can be written as

$$\boldsymbol{R} = \exp(\theta\,\hat{\boldsymbol{U}})\;.$$

Just as a rotation operator in Euclidean space is a combination of reflections, the available transformations in conformal space are combinations of inversions, which generates the group of conformal transformations. This includes, for example, dilation and translation. Hence, rotors in the conformal embedding space can represent dilations and translations in the corresponding Euclidean space.

1.5.5 Outermorphism

The *outermorphism* property of transformation operators, not to be confused with an *automorphism*, plays a particularly important role, and is one of the reasons why geometric algebra is such a powerful language for the description of geometry. The exact mathematical definition of the outermorphism of transformation operators is given in Sect. 3.3. The basic idea is as follows: if $\boldsymbol{R}$ denotes a rotor and $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^3$ are two vectors, then

$$\boldsymbol{R}\,(\boldsymbol{a} \wedge \boldsymbol{b})\,\widetilde{\boldsymbol{R}} = (\boldsymbol{R}\,\boldsymbol{a}\,\widetilde{\boldsymbol{R}}) \;\wedge\; (\boldsymbol{R}\,\boldsymbol{b}\,\widetilde{\boldsymbol{R}})\;.$$

Since geometric entities are represented by the outer product of a number of vectors, called a *blade*, the above equation implies that the rotation of a blade is equivalent to the outer product of the rotation of the constituent vectors. That is, a rotor rotates any geometric entity, unlike rotation matrices, which differ for different geometric entities. Of course, all transformation operators that are constructed from the geometric product of vectors satisfy the outermorphism property.

http://www.springer.com/978-3-540-89067-6

Geometric Algebra with Applications in Engineering
Perwass, C.
2009, XIV, 386 p., Hardcover
ISBN: 978-3-540-89067-6