Geometric Calculus – Engineering Mathematics for the 21st Century

Eckhard MS HITZER*

Mem. Fac. Eng. Fukui Univ., Vol. 50, No. 1 (March 2002)

This paper treats important questions at the interface of mathematics and the engineering sciences. It starts off with a quick quotation tour through 2300 years of mathematical history. At the beginning of the 21st century, technology has developed beyond every expectation. But do we also learn and practice an adequately modern form of mathematics? The paper argues that this role is very likely to be played by universal *geometric calculus*. The fundamental *geometric product* of vectors is introduced. This gives a quick-and-easy description of rotations as well as the ultimate geometric interpretation of the famous quaternions of Sir W.R. Hamilton. Then follows a one page review of the historical roots of geometric calculus. In order to exemplify the role of geometric calculus for the engineering sciences three representative examples are looked at in some detail: elasticity, image geometry and pose estimation. Next a current snapshot survey of geometric calculus software is provided. Finally the value of geometric calculus for teaching, research and development is commented.

***Key Words* :** Applied Geometric Calculus , Engineering Mathematics, Design of Mathematics, Teaching of Mathematics for Engineering Students, Geometric Calculus Software

Introduction

1.1 Mathematicians 'life'

1. A point is that which has no part.
2. A line is breadthless length. …

Euclid[1]

…never to accept anything as true if I did not have evident knowledge of its truth; …We have an idea of that which has infinite perfection. …The origin of the idea could only be the real existence of the infinite being that we call God.

Rene Descartes[2]

But on the 16th day of the same month …an under-current of thought was going on in my mind, which gave at last a result, whereof it is not too much to say that I felt at once the importance. …Nor could I resist the impulse - unphilosophical as it may have been - to cut with a knife on a stone of Brougham Bridge, as we passed it, the fundamental formula with the symbols, i, j, k;

William R. Hamilton[3]

…extension theory, which extends and intellectualizes the sensual intuitions of geometry into general, logical concepts, and, with regard to abstract generality, is not simply one among other branches of mathematics, such as algebra, combination theory, but rather far surpasses them, in that all fundamental elements are unified under this branch, which thus as it were forms the keystone of the entire structure of mathematics.

Hermann Grassmann[4]

…for geometry, you know, is the gate of science, and the gate is so low and small that one can only enter it as a little child.

William K. Clifford[5]

The symbolical method, however, seems to go more deeply into the nature of things. It …will probably be increasingly used in the future as it becomes better understood and its own special mathematics gets developed.

Paul A.M. Dirac [6]

The geometric operations in question can in an efficient way be expressed in the language of Clifford algebra.

Marcel Riesz[7]

This was Grassmann's great goal, and he would surely be pleased to know that it has finally been achieved, although the path has not been straightforward.

David Hestenes[8]

1.2 Design of Mathematics

Over a span of more than 2300 years, Euclid, Descartes, Hamilton, Grassmann, Clifford, Dirac, Riesz, Hestenes and others all contributed significantly to the development of modern mathematics. Today we enjoy more than ever the fruits of their creative work. Nobody can think of science and technology, research and development, without acknowledging the great reliance on mathematics from beginning to end.

Many forms of mathematics have been developed over thousands of years: geometry, algebra, calculus, matrices, vectors, determinants, etc. All of which find rich applications in the engineering sciences as well. But it takes many years in school and university to train students until they reach the level of mathematics needed for today's advanced requirements.

Yet very important questions seem to largely go unnoticed: Is the present way we learn, exercise, apply and research mathematics really the most efficient and satisfying way there is? In an age, where we can double the speed of computers every 3 years, is there no room for improvement for the teaching and application of one of our most fundamental tools mathematics? How should mathematics be designed, so that students, researchers and engineers alike will benefit most from it?

I do think that at the beginning of the 21^{st} century, we have strong reasons to believe, that all of mathematics can be formulated in a single unified universal[9] way, with concrete geometrical foundations. Why is geometry so important? Because it is that aspect of mathematics, which we can imagine and visualize. The branch of mathematics, which Grassmann said far surpasses all others [4] is now known under the name universal *geometric calculus*.

Its formulation is at the same time surprisingly simply, clear and straightforward in teaching and applications. In my experience it is also of great appeal for students.

The rest of this paper is divided into five major sections. In the next section we will see how geometric calculus defines a new way to multiply vectors. This immediately gives us a new method to do rotations and teaches us the nature of Hamilton's famous quaternions.

Section three briefly reviews the history of geometric calculus.

Section four takes up three examples of geometric calculus applied to elasticity, image geometry and pose estimation. Many other applications more closely related to other fields of engineering exist as well.

Section five surveys the currently available software implementations for geometric calculus computations.

Section six outlines the general benefits for teaching, research and development.

This paper is an updated and expanded version of a talk given in Nov. 2001 at the Pukyong National University in Korea [29].

1. New Vector Product Makes Rotations Easy

At the (algebraic) foundations of Geometric Calculus [9] lies a new definition of vector multiplication, the geometric product. It was introduced by Grassmann [4] and Clifford [14] as a combination of inner product and outer product. The outer product was invented by Grassmann before that. The outer product $\vec{a} \wedge \vec{b}$ of two vectors is the (oriented) parallelogram area spanned by two vectors $\vec{a}$ and $\vec{b}$, illustrated in Fig. 1.

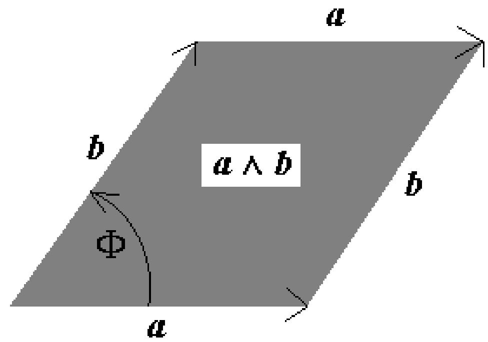

Fig. 1 Oriented parallelogram area $\vec{a} \wedge \vec{b}$

The oriented unit area is denoted by **i**. ***But a warning is in order:* i *is NOT to be confused with the imaginary unit of the complex numbers introduced by Gauss!*** In two dimensions the area unit **i** is of similar importance as the unit length 1 is for one dimension.

The "new" geometric product then simply reads

$$\vec{a}\vec{b} = \vec{a} \cdot \vec{b} + \vec{a} \wedge \vec{b}\,. \qquad (2.1)$$

Yes here we add scalar numbers (inner product) and areas (outer product), but nobody has a problem to put balls and discs in one box, without confusing them. The usual multiplication of real numbers is associative, i.e.

$$(2*3)*4 = 6*4 = 2*(3*4) = 2*12 = 24\,. \qquad (2.2)$$

It simply doesn't matter where you put the brackets, the result is the same. The same is true for the geometric product of vectors.

Let us now again take two vectors, but of unit length: $\hat{a}$, $\hat{b}$. Multiplying their geometric product $\hat{a}\hat{b}$ once more with $\hat{a}$ we get $\hat{b}$ again:

$$\hat{a}(\hat{a}\hat{b}) = (\hat{a}\hat{a})\hat{b} = 1\hat{b} = \hat{b} \qquad (2.3)$$

What we have just done is to rotate the vector $\hat{a}$ into the vector $\hat{b}$ by multiplying it with $\hat{a}\hat{b}$. This is a rotation by the angle Φ as seen in Fig. 1. This is indeed a very general description of rotations in the plane of the rotation. It can be applied to any vector in order to rotate it in the plane of $\hat{a}$ and $\hat{b}$ by the angle Φ. The product $\hat{a}\hat{b}$ deserves thus a separate name

$$R_{ab} \equiv \hat{a}\hat{b} = \cos\Phi + \mathbf{i}\sin\Phi \,. \qquad (2.4)$$

Remember that the inner product of two unit vectors is just cosΦ and the area of the parallelogram they span is base*height = 1*sinΦ = sinΦ. **i** and Φ describe the rotation as good as $\hat{a}$ and $\hat{b}$. A Φ=90° rotation with cos90°=0 and sin90°=1 is therefore given by

$$R(90^o) \equiv \mathbf{i} \,. \qquad (2.5)$$

Rotating twice by 90° gives 180°, turning each vector into the opposite direction. We therefore have:

$$R(90^o)R(90^o) = \mathbf{ii} = \mathbf{i}^2 = -1 \,. \qquad (2.6)$$

Independent of this, the Irish mathematician Sir William R. Hamilton was thinking in 1843 about how to describe rotations in three dimensions in the most simple way. While making a walk he suddenly found the answer

$$\mathbf{i}^2 = -1, \ \mathbf{j}^2 = -1, \ \ \mathbf{k}^2 = -1, \qquad (2.7a)$$

$$\mathbf{ijk} = 1 \,. \qquad (2.7b)$$

Hamilton was so happy that he carved (2.7) immediately into a stone bridge. He called the four entities {1, **i**, **j**, **k**} *quaternions* (=fourfold). [3,10]

For describing a rotation with a quaternion q, we just need to choose the angle of rotation ϑ and the axis [unit vector (u_1, u_2, u_3) in the direction of the axis]:

$$q \equiv 1\cos\frac{\vartheta}{2} + (u_1\mathbf{i} + u_2\mathbf{j} + u_3\mathbf{k})\sin\frac{\vartheta}{2}, \qquad (2.8a)$$

$$\tilde{q} \equiv 1\cos\frac{\vartheta}{2} - (u_1\mathbf{i} + u_2\mathbf{j} + u_3\mathbf{k})\sin\frac{\vartheta}{2} \,. \qquad (2.8b)$$

The rotation of any vector $\vec{x}$ is then given as [11]

$$\vec{x}' = \tilde{q}\vec{x}q \,, \qquad (2.9)$$

which obviously is much more direct, simpler and computationally more efficient than the usual 3 by 3 matrix notation. [In (2.3) the rotation operation was one sided, here it is two sided, because the part of $\vec{x}$ not in the rotation plane must not change.] Instead of *nine* matrix elements, we need only *four* parameters ϑ, u_1, u_2, u_3 in (2.9).

Sir Hamilton knew that his new description of rotations was revolutionary, but what he did not know and even many of today's scientists do not yet know is the geometric meaning of { **i**, **j**, **k** }. But given that **i** represents in two dimensions the (oriented) unit area element, it is natural to take { **i**, **j**, **k** } to represent the three mutually perpendicular (oriented) unit area elements of a cube, as in Fig. 2.

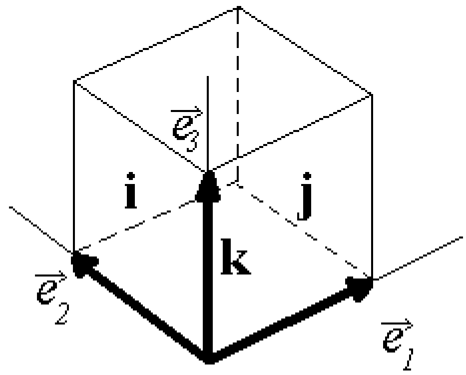

Fig. 2 Oriented unit area elements **i**, **j**, **k** of a cube.

This interpretation is indeed consistent and valid in the framework of Geometric Calculus.[12] In three dimensions, adding plane area elements, is quite similar to adding vectors. The result is a new area element. The sum

$$\mathbf{u} \equiv (u_1\mathbf{i} + u_2\mathbf{j} + u_3\mathbf{k}) \qquad (2.10)$$

in (2.8) is therefore just a new (oriented unit) area element perpendicular to the axis

$$\vec{u} \equiv (u_1\vec{e}_1 + u_2\vec{e}_2 + u_3\vec{e}_3) \,. \qquad (2.11)$$

Just like as in (2.4) each quaternion q (2.8) can therefore be written as a product of two unit vectors in the plane

$$R_q \equiv q = \cos\frac{\vartheta}{2} + \mathbf{u}\sin\frac{\vartheta}{2} = \hat{a}_{\mathbf{u}}\hat{b}_{\mathbf{u}} \,. \qquad (2.12)$$

2. Creation of Geometric Calculus

2300 years ago the ancient Greek scholar Euclid described (synthetic) geometry in his famous *13 books of the elements*. 50 years later (syncopated) algebra entered the stage through the work of Diophantes. Euclid's work [1] was first printed in 1482. But it took yet another 150 years until the French Jesuit

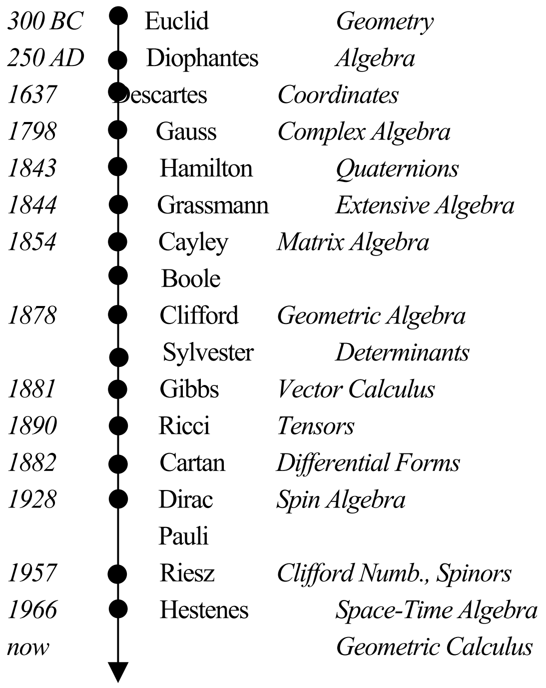

Fig. 3 History of Geometric Calculus [13]

monk Rene Descartes [2] invented analytic geometry. Every student knows him through his introduction of rectangular Cartesian coordinates. After the French revolution, Gauss and Wessel introduced the algebra of complex numbers.

The following 19th century proved very fruitful for the development of modern mathematics. The Irish mathematician Sir William R. Hamilton discovered the quaternions [3,10] in 1843, providing a most elegant way to describe rotations. One year later published the German mathematician Herrmann Grassmann his now famous work on extensive algebra.[4] Yet at first only few mathematicians like Hamilton, later Clifford [14] and Klein and a growing number of others took notice. 10 years later showed G. Boole how algebra can be used to study locigal operations. In the same year, Cayley continued the coordinate approach of Descartes by introducing matrix algebra. Something which Grassmann had no need of in the first place.

Then came the year 1878, when Clifford [14] extensively applied the *geometric product*, which appeared in Grassmann's previous work as *central product*. After Clifford's early death (supposedly because he overworked himself repeatedly), the algebra based on the geometric product became to be known as Clifford algebra, yet following his original intent, it should better be named *geometric algebra*. Again in the same year, Sylvester continued to develop matrix algebra in the form of introducing determinants. In 1881 Gibbs' vector calculus followed, which Ricci enhanced in 1890 to tensor calculus.

In the first half of the 20th century, the names of Cartan (differential forms, 1908) and of Dirac and Pauli (Spin Algebra, 1928) deserve to be mentioned. In the second half of the 20th century (1957), Marcel Riesz [7] gave some lectures on Clifford Numbers and Spinors. Early in his career (1966), a young American David Hestenes came across Riesz lecture notes and created the socalled Space-Time Algebra [15], integrating classical and quantum physics. This marked the beginning of renewed interest in geometric algebra, combined with calculus. Sobczyk and Hestenes published in the early 1980ies a modern classic[9]: Clifford Algebra to Geometric Calculus – A Unified Language for Mathematics and Physics. By the beginning of the 21st century it has become a truly universal geometric calculus, incorporating more or less all areas of mathematics, and starting to be extensively applied in science and technology. [16,17]

The proponents of geometric calculus have no doubt, that this new language for mathematics will make its way into undergraduate syllabi and even school education. Mathematics will thus become easier to understand, teach, learn and apply. As for the applications, the next section will show how geometric calculus is successfully used in engineering.

3. Geometric Calculus for Engineers

4.1 Overview

In order to get an overview of how geometric calculus supports engineering applications, let me first list some relevant topics from a recent conference[18] on applied geometric algebras in computer science and engineering:

- Computer vision, graphics and reconstruction
- Robotics
- Signal and image processing
- Structural dynamics
- Control theory
- Quantum computing
- Bioengineering and molecular design
- Space dynamics
- Elasticity and solid mechanics
- Electromagnetism and wave propagation
- Geometric and Grassmann algebras
- Quaternions and screw theory
- Automated theorem proving
- Symbolic Algebra
- Numerical Algorithms

One should note that the organizers cautioned: "Topics covered will include (but are not limited to):" and that geometric algebra itself is only the algebraic fraction of the full-blown geometric *calculus* [9]. Limitations of space prohibit any complete listing here.

4.2 Three Examples of Engineering Applications

Trying to choose what to present from the recent engineering applications of geometric calculus is a very tough choice, because there are many good applications.

I have chosen three dealing with elasticity, image geometry and pose estimation.

4.2.1 Example 1: Elastically Coupled Rigid Bodies[19]

Modelling elastically coupled rigid bodies is an important problem in multibody dynamics. A flexural joint has two rigid bodies coupled by a more elastic body. Such a system is shown in Fig. 4.

Fig. 4 Elastically coupled rigid bodies. Source: [19].

It is convenient to avoid specifying an origin, i.e. use a new homogeneous formulation.[8,19] Rotations *R* and translations *T* are fully integrated as twistors in screw theory. That is, any relative displacement *D* of two bodies can be written as

$$D \equiv TR\,, \;\; \vec{x}' = D\vec{x}\widetilde{D}\,. \tag{4.1}$$

R is the rotation of section 2 and

$$T \equiv \exp(\tfrac{1}{2}\vec{n}\vec{e}) = 1 + \tfrac{1}{2}\vec{n}\vec{e} \tag{4.2}$$

$\vec{n}$ is the translation vector and $\vec{e}$ represents an infinitely far away point in (conformal) geometric algebra [8,19]. Motion, momentum and kinetic energy are than given as

$$\frac{\partial}{\partial t}\vec{x} = V\vec{x}\,, \tag{4.3}$$

$$P = \underline{M}V\,, \tag{4.4}$$

$$E = \frac{1}{2}V \cdot P\,. \tag{4.5}$$

V is defined by

$$\frac{\partial}{\partial t}D = \frac{1}{2}VD\,. \tag{4.6}$$

Finally the potential energy of the elasticity problem can be written as a sum of basically three kinds of terms,

$$\vartheta^2\mathbf{u}\cdot\underline{K}_{\mathbf{O}}\mathbf{u} + \vec{n}\cdot\underline{K}_{\mathbf{C}}\vartheta\mathbf{u} + \vartheta\mathbf{u}\cdot\overline{K}_{\mathbf{C}}\vec{n} + \vec{n}\cdot\underline{K}_{\mathbf{t}}\vec{n}$$

depending on $\vartheta\mathbf{u}$ and $\vec{n}$. The first term depends only on $\vartheta\mathbf{u}$, the next two on $\vartheta\mathbf{u}$ and $\vec{n}$, and the fourth only on $\vec{n}$. The three kinds of terms are therefore the potential energies of pure rotation, coupled rotation and translation, and pure translation. The *K* are the corresponding stiffnesses.

The method described here is invariant, unambiguous, has a clear geometric interpretation and is very efficient in symbolic computation. Two researchers have applied for a *patent* on the use of the method described here in software for modeling and simulation.

4.2.2 Example 2: Image Geometry[20]

Image processing commonly considers "Euclidean differential invariants" of the image space (picture plane × intensity). But this makes not much sense, because one cannot rotate the image surface to see its "other side", but invariants are supposed not to change under such unrealistic transformations. It also makes no sense to "mix" the physical dimensions of the picture plane with the intensity dimension by transforming one into the other.

But these inconsistencies can be helped by first introducing a logarithmic (log) intensity domain and second making new definitions for the basic formulas of measuring angles and distances in the image space.

The log intensity means to divide by a fiducial intensity I_0 and take the logarithm

$$z(\vec{r}) = \log(I(\vec{r})/I_0)\,. \tag{4.7}$$

A definition very well adapted to the human eye functions.

The definition of measurements is not changed, when considering only the picture plane. But if we look at a plane in the image space perpendicular to the picture plane, a *rotation* becomes a *shear* as shown in Fig. 5. In geometric algebra one continues to use a description of rotation as given by (2.9) and (2.10), but the square of **u** will be zero instead of -1.

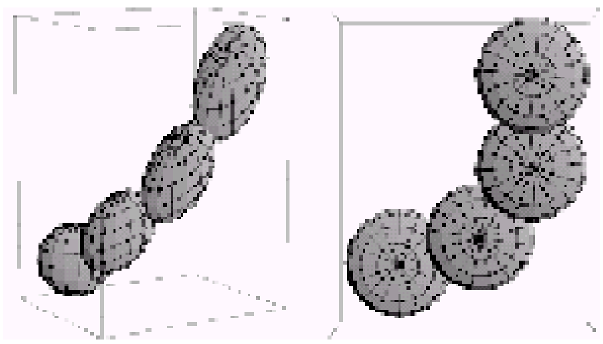

Fig. 5 An object is moved through image space by a parabolic rotation (left). Perpendicular view on the picture plane (right). Source: [20].

Analyzing the image surface curvature in the image space gives very natural descriptions of ruts and ridges. A ridge point, e.g. is "an extremum of (principle) curvature along the direction of the other (principle) curvature." I should be clear that when a curve has one curvature, a surface (e.g. a saddle) must have two (principal) curvatures. Fig. 6 shows a variety of common image transformations easily implemented with our new definition of image space.

Fig. 6 Common image transformations. Source: [20].
Top: original, two gradients;
middle: γ transformation, intensity scaling, edge burning;
bottom: inversion, dodging and burning, flashing.

Another promising new approach is the *structure multivector* which includes information about local amplitude, local phase, and local geometry of both intrinsically 1D and 2D signals, isotropous even in 2D. It gives the proper generalization to the *analytic signal* (amplitude+phase) of 1D.[21]

4.2.3 Example 3: Monocular Pose Estimation[22]

(Conformal) geometric algebra [8,19] can be successfully used to formalize algebraic embedding of monocular pose estimation of kinematic chains. This is helpful for e.g. tracking robot arms or human body movements. As shown in Fig. 7 one relates positions of a 3D object to a reference camera coordinate system. The resulting (constraint) equations are compact and clear, and easy to linearize and iterate.

In a first step the purely kinematic problem of finding the rotation R and the translation T of the observed model in Fig. 7 is solved by way of exploiting obvious point-on-line, point-on-plane, and line-on-plane constraints. Points, lines and planes are defined by the outer product in (conformal) geometric algeba as

$$X \equiv \vec{e} \wedge \vec{x}\,, \qquad (4.8a)$$

$$L \equiv \vec{e} \wedge \vec{x} \wedge \vec{y}\,, \qquad (4.8b)$$

$$P \equiv \vec{e} \wedge \vec{x} \wedge \vec{y} \wedge \vec{z}\,. \qquad (4.8c)$$

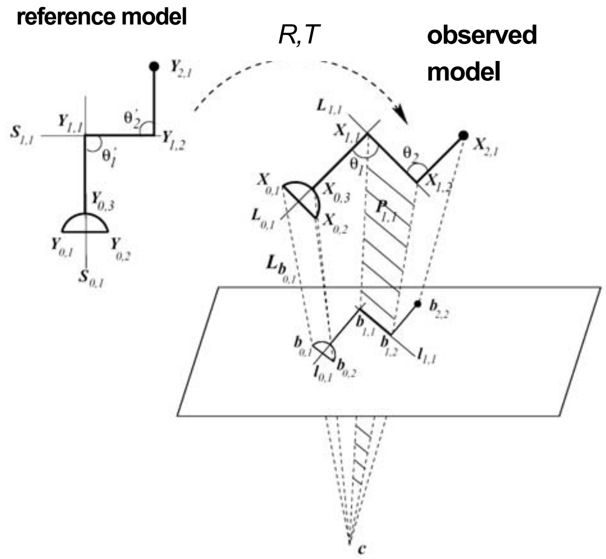

Fig. 7 Solid lines: camera model, object model, extracted lines on image plane. Dashed lines: best pose fit. Source: [22].

The point-on-plane constraint is e.g. simply given by

$$XL - LX = 0\,. \qquad (4.9)$$

The formulation of a *kinematic* constraint is now straightforward by using equation (4.1) for the displacement D (composed of rotation and translation)

$$(DX\widetilde{D})L - L(DX\widetilde{D}) = 0\,. \qquad (4.10)$$

One now has only to find the best displacement D which satisfies the constraint (4.10).

Pose estimation of kinematic chains is also evident. One simply refines the scheme to include internal displacements. In Fig. 7 this can e.g. be internal rotations changing θ_1 and θ_2.

A real application can be seen in Fig. 8. The pose of a doll and the angles of the arms are estimated, by labeling one point on each kinematic chain segment. Already few iterations of the linarized problem give a good estimation of the pose and the kinematic chain parameters.

This concludes our short tour through the world of applications of geometric calculus. The literature, the internet, pending patents [19,23], etc. contain a lot more.

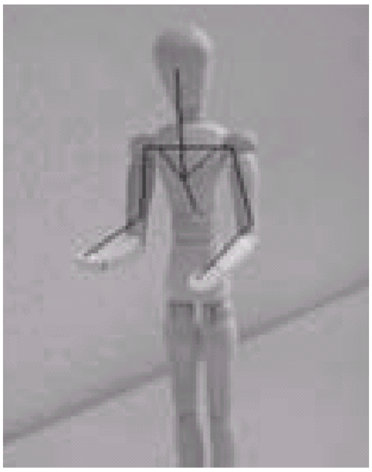

Fig. 8 The pose of the doll and the angles of the arms are estimated. Source: [22].

4. Geometric Calculus Software

The following contains a current snapshot survey of geometric calculus implementing software. Two broad categories are free, standalone software and software packages written for the use together with large commercial mathematical software programs.

The software is listed with its often acronymic name, name explanation, homepage, names of the chief inventors, and a short comment on its particular nature. The homepages are of great importance for downloads, manuals, tutorials, examples of applications, latest version updates, source codes, secondary literature, etc. The interested reader is therefore referred to the relevant homepage.

5.1 Standalone Software

5.1.1 CLICAL

The name CLICAL stands for *Complex Number, Vector Space and Clifford Algebra Calculator for MS-DOS Personal Computers*. The homepage is:

http://www.teli.stadia.fi/~lounesto/CLICAL.htm

It was invented by P. Lounesto at the Helsinki University of Technology in Finland. CLICAL evaluates elementary functions with arguments in complex numbers, and their generalizations: quaternions, octonions and multivectors in Clifford algebras.

5.1.2 CLU, CLUDraw, CLUCalc

CLU stands for *Clifford algebra Library and Utilities.* CLUDraw is a visualization library based on the CLU library. CLUCalc is a visual Clifford algebra calculator. The common homepage is:

http://www.perwass.de/cbup/clu.html

All three programs were developed by C. Perwass, currently at the Univesity of Kiel, Germany. CLU is a C++ Library that implements geometric (or Clifford) algebra. It has been compiled and tested under Windows 98/ME, Linux SUSE 7.0, and Solaris.

CLUDraw allows to visualize points, lines, planes, circles, spheres, rotors, motors and translators, as represented by multivectors. It has been compiled and tested under Windows 98/ME/2000/XP, Linux SUSE 7.0, and Solaris.

With CLUCalc you can type your calculations using an intuitive script language. The results of the CLUCalc calculations and the visualization of multivectors, as in Fig. 9, is then done immediately without the need for an external compiler. CLUCalc has been tested under Windows 98/ME/2000/XP.

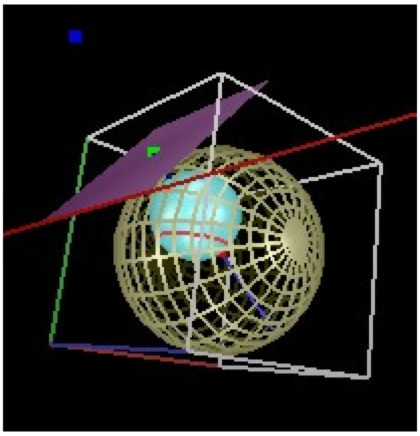

Fig. 9 CLUCalc illustration. Source: http://www.perwass.de/cbup/clucalcdownload.html

5.1.3 Gaigen

Gaigen is a program which can *generate implementations of geometric algebras*.

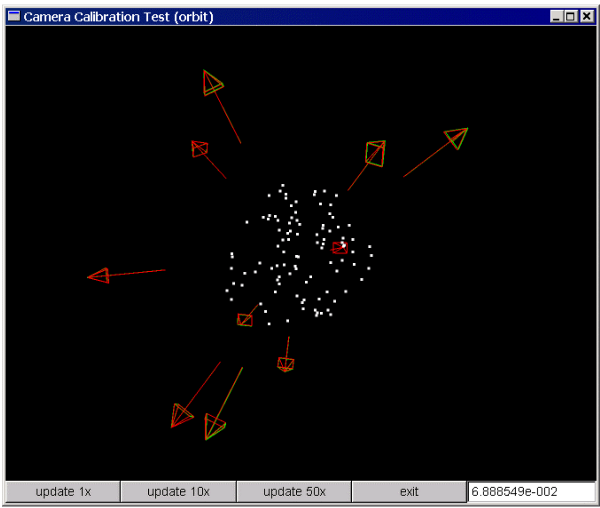

Fig. 10 Motion capture camera calibration with Gaigen. 10 cameras (arrows) and 100 markers (points). Source: http://carol.wins.uva.nl/~fontijne/gaigen/apps_mccc.html

Its homepage is:

http://carol.wins.uva.nl/~fontijne/gaigen/

Gaigen was written by Daniel Fontijne at the University of Amsterdam (Holland) in cooperation with Tim Bouma and Leo Dorst. Gaigen generates C++, C and assembly source code which implements a geometric algebra requested by the user. It is downloadable in the form of standalone executables for Win32, Sun/Solaris and Linux. E.g. the motion capture camera calibration computation of Fig. 10 can be done in a couple of seconds for 10 cameras looking at position 100 markers.

5.1.4 C++ Template Classes for Geometric Algebra

Publically available *C++ template classes to implement geometric algebras* or Clifford algebras. The homepage is:

http://www.nklein.com/products/geoma/

These Template Classes were developed by Patrick Fleckenstein of 'nklein software' in Rochester, New York, US. The available template classes are: *GeometricAlgebra, GeomMultTable,* and *GeomGradTable.*

5.1.5 Online Geometric Calculator

The online geometric calculator shown in Fig. 11 is just an ordinary *desk type calculator that uses the Clifford numbers* over a three dimensional Euclidean space. The homepage is:

http://www.elf.org/calculator/

It was programmed by R. E. Critchlow Jr of Santa Fe, New Mexico, US. In addition to real number computations, this calculator also computes on vectors in two and three space, on the bivectors over those vectors, on the trivector over three space, on complex numbers, and on quaternions.

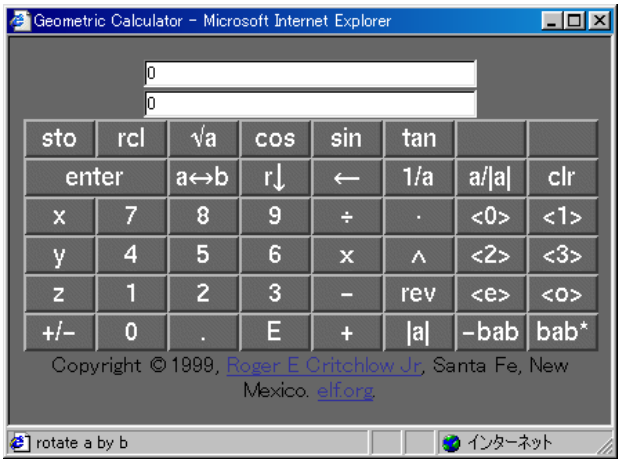

Fig. 11 Critchlow's geometric calculator screenshot. Source: http://www.elf.org/calculator/

5.1.6 Vector Field Design

A computer program that allows a user to *design, modify and visualize a 2D vector field* in real time [27]. An example of such a vector field is shown in Fig. 12. The homepage is:

http://sinai.mech.fukui-u.ac.jp/gcj/software/toyvfield.html

It was programmed by S. Bhinderwala (Arizona State Unversity, US) with credits to G. Scheuermann, H. Hagen, and H. Krueger (University of Kaiserslautern, Germany), Alyn Rockwood, and D. Hestenes (Arizona State Unversity, US). The Windows(TM) based program for the PC allows movement and redrawing of vector glyphs and integration curves in real time, even with a moderate number of critical points.

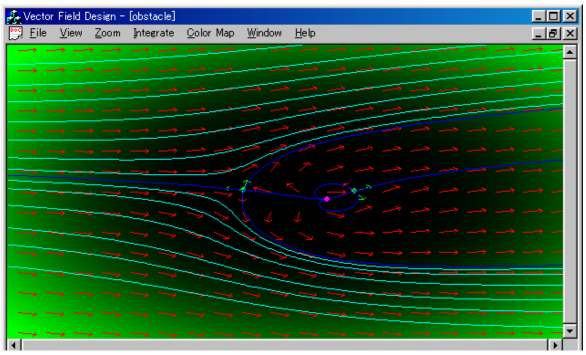

Fig. 12 Vector Field Design 1.0 screenshot.

5.1.7 Clifford Algebra with REDUCE

There have been publications about *Clifford and Grassmann algebra computations* with REDUCE:

http://sinai.mech.fukui-u.ac.jp/gcj/software/gc_soft.html

The official homepage of REDUCE is:

http://www.zib.de/Optimization/Software/Reduce/index.ht ml

Spearheaded 25 years ago by A.C. Hearn (now Santa Monica, California, US), nowadays several groups in different countries take part in the REDUCE development. It is directed towards big formal computations in applied mathematics, physics and engineering with an even broader set of applications. This is the only (half) commercial software of section 5.1: A basic personal PC version is already available for a moderate 100 USD.

5.2 Free Packages for Commercial Software

There are three major rather costly, fully commercial mathematical software packages:

- MAPLE of Waterloo Maple Inc. in Waterloo, Canada
  http://www.maplesoft.com/
- MATLAB of The MathWorks Inc. in Natick, MA, US
  http://www.mathworks.com/
- Mathematica of Wolfram Research Inc. in Champaign, IL, US
  http://www.wolfram.com/

Free of cost geometric calculus software add-on packages are nowadays available for all three of them.

5.2.1 With MAPLE

5.2.1.1 CLIFFORD

CLIFFORD is a *Maple V (now: Rel. 5.1) package for Clifford algebra computations*. Its homepage is:

http://math.tntech.edu/rafal/cliff5/index.html

It was written by both Rafal Ablamowicz (Tennessee Technological University, Cookeville, TN, US) and Bertfried Fauser (University of Konstanz, Germany). It contains:

- **CLIFFORD** for computations in Clifford algebras
- **Bigebra** for computations with Hopf gebras and bi-gebras
- **Cli5plus** extends CLIFFORD to other bases.
- **GTP** extends CLIFFORD to graded tensor products of Clifford algebras.
- **Octonion** for computations with octonions.

5.2.1.2 Geometric Algebra Package

This package enables the user to perform calculations in a geometric/Clifford algebra of arbitrary dimensions and signature. Its homepage is:

http://www.mrao.cam.ac.uk/~clifford/software/GA/

This MAPLE add-on package was written by Mark Ashdown, Astrophysics Group, Cavendish Laboratory, University of Cambridge, UK. The package is available for Unix, Linux etc. and for DOS/Windows. Apart from the geometric algebra functions, there are two functions in this package which perform the geometric *calculus* operations of multivector derivative and multivector differential.

5.2.1.3 LUCY

LUCY: A (*Lancaster University Clifford Yard*) Clifford algebra approach to spinor calculus. Online information is available via:

http://www.birkhauser.com/book/ISBN/0-8176-3907-1/wang/wang.html

It was created by J. Schray, R. W. Tucker and C. Wang of Lancaster University, UK. LUCY exploits the general theory of Clifford algebras to effect calculations involving real or complex spinor algebras and spinor calculus on manifolds in any dimensions.

5.2.1.4 Glyph

Glyph is a Maple V (release 5) package for performing symbolic computations in a Clifford algebra. Its homepage is:

http://bargains.k-online.com/~joer/glyph/glyph.htm

It was written by Joe Riel, an electrical engineer in San Diego, California. The Glyph package currently features: *loadable spaces*, *a solver for systems of equations*, *evaluation Clifford polynomials*, and *conversions to and from matrix equivalents*. In the near future differentiation and integration will be added along with routines for rotating and reflecting multivectors.

5.2.2 With MATLAB: GABLE

GABLE is a MATLAB *geometric algebra learning environment* [26]. Its homepage is:

http://carol.wins.uva.nl/~leo/clifford/gable.html

It was jointly developed by L. Dorst, Tim Bouma (both at the University of Amsterdam, Holland), and S. Mann (University of Waterloo, Canada). It graphically demonstrates in three dimensions the products of geometric algebra and a number of geometric operations. The example of a geometric algebra interpolation of a roation with GABLE is shown in Fig. 13.

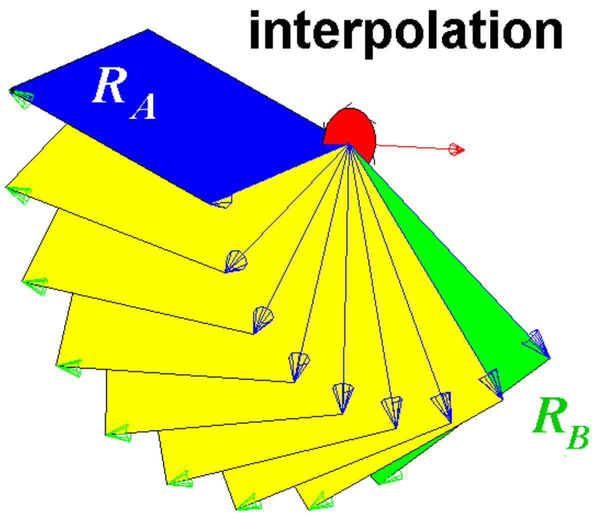

Fig. 13 Rotation interpolation with GABLE. Output of DEMOinterpolation.

5.2.3 With Mathematica

5.2.3.1 Clifford

Clifford is a Mathematica package for calculations with Clifford algebra. Its homepage is:

http://iris.ifisicacu.unam.mx/software.html

It was developed by The Structure of Matter Group of J. L. Aragon, Institute of Physics, Universidad Nacional Autónoma de México.

5.2.3.2 GrassmannAlgebra

The GrassmannAlgebra software is a Mathematica package, for doing a range of manipulations on numeric or symbolic Grassmann expressions in spaces of any dimension and metric. Its homepage is:

http://www.ses.swin.edu.au/homes/browne/grassmannalgebra/book/index.htm

It is currently programmed by J. Browne, Swinburne University of Technology, Australia. The first alpha testing release is scheduled around mid 2002. The author is in the process of writing a new book entitled: "Grassmann Algebra, Exploring applications of extended vector algebra with Mathematica." It has the aim to provide a readable account in modern notation of Grassmann's major algebraic contributions to mathematics and science. The package is intended to be used to extend the examples in the text, experiment with hypothesis, and for independent exploration of the algebra.

5.3 The Benchmark Race

The geometric calculus software sector currently undergoes rapid development and expansion. Soon packages like Gaigen will draw equal with conventional linear algebra software computing benchmarks – but with the advantage for geometric calculus implementations to be both more efficient and functional. [25]

1. Teaching, Research & Development

6.1 Teaching of Engineering Sciences

Already the teaching of engineering sciences will benefit greatly from making use of the general geometric language of geometric calculus. (Linear) algebra and calculus can be taught in a new unified, easy to understand way. Next all of physics is by now formulated in terms of geometric calculus. [9,12,15,24] The same applies to basic crystal structures, molecular interactions, signal theory, etc. Wherever an engineer employs mechanics, electromagnetism, thermodynamics, solid state matter theory, quantum theory, etc. it can be done in one and the same language of geometric calculus catering for diverse needs. The students will not have to learn new mathematics, whenever they encounter a different part of engineering science.

6.2 Research and Development

Research and development do already benefit a great deal from employing geometric calculus. Even the quaternions [3,10] of Hamilton by themselves are already of great advantage for aerospace engineering and virtual reality [11]. Modeling and simulation can now make use of powerful, new methods. Conference participation numbers show that computer vision and graphics people are particularly interested.[18] It also leads to the development of new and very fast computer algorithms both for symbolic and numeric calculations.[16,17] Higher dimensional image geometry may for the first time ever get a solid theoretical footing, enabling systematic study and exploration, not just guessing around.

2. Conclusion

This work started from the historic roots of geometry, a field , which gradually expanded over many centuries to finally provide us with a mathematically universal form of geometric calculus. At the beginning of the 21st century, we find ourselves therefore at a historic crossroad of the traditionally fragmented patchwork of commonly practiced mathematics, and of geometric calculus with its universal unifying structures.

The fundamental geometric product and immediate consequences for the elegant description of rotations were introduced. Next the wide field of geometric calculus applications for engineers was outlined. Three concrete engineering problems were looked at in some detail. The following major section contained an up to date snapshot survey of various software implementations of geometric calculus. This is a vibrant field undergoing rapid development. Finally the future implications for teaching, research and development were discussed.

I finally conclude therefore that geometric calculus is a more than promising candidate to become the single major teaching, research and development tool for engineers of the 21st century.

3. Acknowledgements

I thank God my creator for the joy of doing research in the wonders of his works: "He (Christ) is before all things, and in him all things hold together."[28]

I thank my wife for continuously encouraging my research. I thank the whole Department of Mechanical Engineering at Fukui University for generously providing a suitable research environment for me. I finally thank my friend K. Shinoda, Kyoto for his prayerful personal support. I also thank H. Ishi.

References

[1] Euclid, *The Thirteen Books of the Elements*, translated by Sir Thomas L. Heath, Dover Publications, N.Y. 1956.

[2] Rene Descartes, *Optics, Meteorology, and Geometry - 'Discourse on the Method of Rightly Conducting the Reason and Seeking Truth in the Sciences.',* 1637.

Rene Descartes, *Meditations on the First Philosophy: In Which the Existence of God and the Distinction Between Mind and Body are Demonstrated.*, 1641.
Excerpts: http://www.utm.edu/research/iep/
The Internet Encyclopaedia of Philosophy.

[3] Sir William R. Hamilton, *Letter to [his son] Rev. Archibald H. Hamilton.* August 5, 1865.
http://www.maths.tcd.ie/pub/HistMath/People/Hamilton/Letters/BroomeBridge.html

[4] H. Grassmann, (F. Engel, editor) *Die Ausdehnungslehre von 1844 und die Geometrische Analyse*, vol. 1, part 1, Teubner, Leipzig, 1894.

[5] As quoted in: S. Gull, A. Lasenby, C. Doran, *Imaginary Numbers are not Real. - the Geometric Algebra of Spacetime,* Found. Phys. **23** (9), 1175 (1993).

[6] Paul A. M. Dirac, *The principles of quantum mechanics,* Oxford Science Publ., 1967.

[7] Marcel Riesz, *Clifford Numbers and Spinors,* E.F. Blinder, P. Lounesto (eds.), Kluver Acad. Publ. Dordrecht, 1993.

[8] David Hestenes, *Old Wine in New Bottles: A new algebraic framework for computational geometry.* In E. Baryo-Corochano & G. Sobczyk (eds.), *Geometric Algebra with Applications in Science and Engineering.* Birkhaeuser, Boston, 2001.

[9] D. Hestenes, G. Sobczyk, *Clifford Algebra to Geometric Calculus*, Kluwer Academic Publishers, Dordrecht, 1984.

[10] Sir William R. *Hamilton On a new Species of Imaginary Quantities connected with a theory of Quaternions,* Proceedings of the Royal Irish Academy, Nov. 13, 1843, 2, 424-434. See also: http://www.maths.tcd.ie/pub/HistMath/People/Hamilton/Quatern1/

[11] Jack B. Kuipers, *Quaternions and Rotation Sequences – A primer with Applications to Orbits, Aerospace, and Virtual Reality*, Princeton Univ. Press, Princeton, 1998.

[12] D. Hestenes, *New Foundations for Classical Mechanics*, Kluwer Academic Publishers, Dordrecht, 1999.

[13] D. Hestenes, *Family tree for geometric calculus.* http://modelingnts.la.asu.edu/html/evolution.html part of: http://modelingnts.la.asu.edu/GC_R&D.html

[14] W. K. Clifford. *Applications of Grassmann's extensive algebra.* Am. J. Math., 1:350, 1878.

[15] D. Hestenes, *Space Time Algebra*, Gordon and Breach, New York, 1966.

[16] C. Doran, L. Dorst and J. Lasenby eds. *Applied Geometrical Alegbras in computer Science and Engineering, AGACSE 2001*, Birkhauser, 2001.

[17] R. Ablamowicz, P. Lounesto, J.M. Parra, *Clifford Algebras with Numeric and Symbolic Computations.* Birkhaeuser, Boston, 1996.

[18] *Applied Geometrical Alegbras in computer Science and Engineering, AGACSE 2001,* Conference homepage, http://www.mrao.cam.ac.uk/agacse2001/

[19] D. Hestenes, *Homogeneous Formulation of Classical Mechanics,* to appear in [16].

[20] J.J. Koenderink, *A Generic Framework for Image Geometry,* to appear in [16].

[21] M. Felsberg, G. Sommer, *The Structure Multivector,* to appear in [16].

[22] B. Rosenhahn, O. Granert, G. Sommer, *Monocular Pose Estimation of Kinematic Chains,* to appear in [16].

[23] W. Neddermeyer, M. Schnell, W. Winkler, A. Lilienthal, *Stabilization of 3D pose estimation,* to appear in [16].

[24] D. Hestenes, *Universal Geometric Calculus*, http://modelingnts.la.asu.edu/html/UGC.html

[25] D. Fontijne, *Gaigen, A Geometric Algebra Implementation Generator*, presentation, University of Amsterdam, 19 Feb. 2002. http://carol.wins.uva.nl/~fontijne/gaigen/files/20020219_talk.pdf

[26] L. Dorst, S. Mann, *GABLE – Geometric Algebra Learning Environment (for MATLAB),* free copy and tutorial: http://carol.wins.uva.nl/~leo/clifford/gable.html or http://www.cgl.uwaterloo.ca/~smann/GABLE/

[27] A. Rockwood, S. Binderwala, *A Toy Vector Field Based on Geometric Algebra,* to appear in [16].

[28] Colossians 1:17, The Bible, New International Version, International Bible Society, Colorado, 1984.

[29] E. Hitzer, *Geometric Calculus for Engineers*, Pukyong National University – Fukui University International Symposium 2001 for Promotion of Research Cooperation, Busan, Korea, 2001.