John Vince

Geometric Algebra for Computer Graphics

Springer

John Vince

Geometric Algebra for Computer Graphics

Springer

John Vince, MTech, PhD, DSc
Emeritus Professor in Digital Media
Bournemouth University
UK

ISBN 978-1-84628-996-5 e-ISBN 978-1-84628-997-2
DOI 10.1007/978-1-84628-997-2

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

Library of Congress Control Number: 2008921355

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

Springer Science+Business Media
springer.com

Geometric Algebra for Computer Graphics

This book is affectionately dedicated to my family: Annie, Samantha, Anthony, Genny, Peter, Megan, Mia and Monty.

Preface

In December 2006 I posted my manuscript *Vector Analysis for Computer Graphics* to Springer and looked forward to a short rest before embarking upon another book. But whilst surfing the Internet, and probably before my manuscript had reached its destination, I discovered a strange topic called *geometric algebra*. Advocates of geometric algebra (GA) were claiming that a revolution was coming and that the cross product was dead. I couldn't believe my eyes. I had just written a book about vectors extolling the power and benefits of the cross product, and now moves were afoot to have it banished! I continued to investigate GA and was amazed that a Google search revealed over 2 million entries. I started to read up the subject and discovered that GA was a Clifford algebra which had a natural affinity with geometry. It appeared that Prof. David Hestenes [14] had invented *geometric calculus* and successfully applied it to classical and quantum mechanics, electrodynamics, projective and conformal geometry. Chris Doran, Anthony and Joan Lasenby at Cambridge University had continued this research and were a driving force behind its understanding, dissemination and application to computer graphics. It seems that if I had been attending SIGGRAPH regularly, I would have been aware of these developments, but alas that was not the case, and I had a lot of catching up to do.

As I started reading various technical papers, especially by Hestenes, Doran and the Lasenbys, I realized the importance of the subject and the need to understand it. Slowly I was drawn into a world of complex numbers, antisymmetric operators, non-commutative products, conformal space, null vectors and the promise of elegance in CGI algorithms. I would be able to divide, rotate and reflect vectors with an ease never before known.

As I was finding it so difficult to understand GA, probably other people would also be finding it difficult, and then I realized the title of my next book: *Geometric Algebra for Computer Graphics*. But how could I write about a subject of which I knew nothing? This was a real challenge and became the driving force that has kept me working day and night for the past year. I took every opportunity to read about the subject: in bed, on planes, trains and boats; whilst waiting at the dentist and even waiting whilst my car was being serviced!

Before embarking on my summer vacation this year (2007) I bought a copy of Doran & Lasenby's excellent book *Geometric Algebra for Physicists* and took it, and my embryonic manuscript, with me to the south of France. My wife and I stayed at the Hotel Horizon in Cabris, overlooking Grasse and Cannes on the Côtes d'Azur. Previous guests have included authors, philosophers and musicians such as Leonard Bernstein, Jean-Paul Sartre, Simone de Beauvoir, Gregory Peck and Antoine de St. Exupèry whose names have been carved into table tops in the bar. Now that I have spent a few days at Hotel Horizon studying bivectors, trivectors and multi-vector products, I am looking forward to seeing my name cut into a table top when I return next year!

This book is a linear narrative of how I came to understand geometric algebra. For example, when I started writing the manuscript, *conformal geometry* were no more than two words, about which, I knew I would eventually have to master and write a chapter. The conformal model has been the most challenging topic I have ever had to describe. To say that I understand conformal geometry would be an overstatement. I understand the action of the algebra but I do not have a complete picture in my mind of 5D Minkowski space which is the backdrop for the conformal model. I admire the authors who have written so confidently about the conformal model, not only for their mathematical skills but their visual skills to visualize what is happening at a geometric level.

When I first started to read about GA I was aware of the complex features of the algebra, in that certain elements had imaginary qualities. Initially, I thought that this would be a major stumbling block, but having now completed the manuscript, the imaginary side of GA is a red herring. If one accepts that some algebraic elements square to -1, that is all there is to it. Consequently, do not be put off by this aspect of the algebra.

Another, stumbling block that retarded my progress in the early days was the representation at a programming level of bivectors, trivectors, quadvectors, etc. I recall spending many days walking my dog Monty trying to resolve this problem. Monty, a Westie, whose knowledge of Clifford algebra was only slightly less than my own, made no contribution whatsoever, but this daily mental and physical exercise eventually made the penny drop and I realized that bivectors, trivectors, quadvectors, etc., were just names recording a numerical value within the algebra. Why had I found it so difficult? Why had this not been explicitly described by other authors? If only someone had told me, I could have avoided this unnecessary mental anguish. But, in retrospect, the mental pain of learning about GA single-handed, has provided me with some degree of confidence when talking about the subject. In fact, in September 2007, I organized a one-day Workshop on GA in London where Dr. Hugh Vincent, Dr. Chris Doran, Dr. Joan Lasenby and me gave presentations to an audience from the computer animation and computer games sectors. It was extremely successful.

I have structured this book such that the first six chapters provide the reader with some essential background material covering complex algebra, vector algebra, quaternion algebra and geometric conventions. These can be skipped if you are already familiar with theses topics. Chapter 7 goes into the history of geometric algebra, but I was already prepared for this as I had read Michael Crowe's fantastic book *A History of Vector Analysis*. In fact, this book is so good I have read it at least four times! Chapter 8 describes the geometric product, which was introduced by Clifford and is central to GA. Chapter 9 explores how GA handles reflections and rotations. Chapter 10 shows how GA is used to solve various problems in 2D and 3D geometry. Chapter 11 describes the conformal model. Chapter 12 is a short review of some typical applications of GA and Chapter 13 identifies important programming tools for GA. Finally, chapter 14 draws the book to a conclusion.

I am not a mathematician, just a humble consumer of mathematics, and whenever I read a book about mathematics I need to see examples, which is why I have included so many in this book. It is so tempting to write:

"It is obvious that Eq. (12.56) is the required rotor",

for very often it is not obvious that this equation is a rotor, or even how it is used in practice. Therefore, whenever I have introduced an equation, I have shown its derivation and its application.

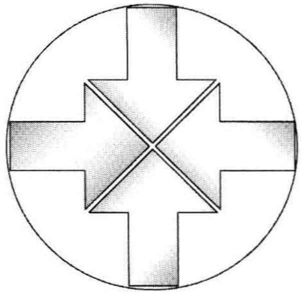
I would like to thank Dr. Hugh Vincent for reading through an early manuscript and offering some constructive feedback. I would also like to thank Dr. Chris Doran for taking the time to read the manuscript and advising me on numerous inconsistencies, and Dr. Joan Lasenby for her responsive, supportive emails when I had lost my way in untangling conformal null vectors. Once again I would like to acknowledge Chris Doran and Anthony Lasenby's excellent book *Geometric Algebra for Physicists.* I could not have written this book without their book. I also must not forget to thank Helen Desmond and Beverley Ford, General Manager of Springer, UK, for their continuous support, memorable lunches and transforming my manuscript into such a beautiful book.

Although I have done my best to ensure that the book is error free, if there are any inconsistencies, I apologize, as they are entirely my fault.

Finally, I must remind the reader that this book is intended only as a gentle introduction to GA. Hopefully, it will provide a bridge that will ease the understanding of technical papers and books about GA, where the subject is covered at a more formal and rigorous level.

Ringwood, UK John Vince

Contents



1 Introduction

1.1 Aims and objectives of this book

The aim of this book is to provide the reader with a gentle introduction to the embryonic subject of geometric algebra (GA). The GA books that currently exist are either directed at physicists or assume that their readers possess a formal understanding of mathematics. To my knowledge, this is the first book that introduces GA without overwhelming the reader with the formalism of linear algebra that supports the subject. The real objective of the book is to make the reader familiar with the concepts of GA. Hopefully on completing the book readers will be able to read more advanced books and technical papers.

1.2 Mathematics for CGI software

Anyone who has written software for computer animation or computer games will know the wide range of mathematical tools needed to implement the algorithms for resolving 2D and 3D geometric problems. Perhaps one of the most important mathematical tools is the matrix transform, where it is difficult to imagine how one could get by without using

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{1.1}$$

to rotate a point about the origin. Although matrices exploit the ability to represent a transform as an array of numbers, the origin of these numbers is linear algebra. Matrix notation simply introduces a degree of elegance that permits the solution to a problem to be addressed at a higher symbolic level, without becoming bogged down in the longhand notation of algebra. Even writing down an array of numbers eventually becomes tedious, and a further substitution can be made by giving names to the transforms such as

$$S=\begin{bmatrix} S_x & 0 & 0 & 0\\ 0 & S_y & 0 & 0\\ 0 & 0 & S_z & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \qquad T=\begin{bmatrix} 1 & 0 & 0 & T_x\\ 0 & 1 & 0 & T_y\\ 0 & 0 & 1 & T_z\\ 0 & 0 & 0 & 1\end{bmatrix} \tag{1.2}$$

which permits us to write their product as $P = ST$. At this stage we have basically created another algebra with its own axioms and embellishments such as $[\![P]\!]^{-1}$ and $[\![P]\!]^T$.

When the algebra of matrices is combined with the algebra of vectors, quaternions, analytic geometry, barycentric coordinates, etc., one realizes the wide range of algebraic notation employed in CGI. Fortunately, this notation is relatively easy to understand and master and has even been incorporated at a hardware level in graphics cards. GA reveals that matrices, determinants, complex numbers, quaternions and vectors are all closely related, which must have an impact upon the design of CGI algorithms.

As you will discover, the notation of GA is rather elegant, even though the underlying algebra is fussy. But we mortals should not be concerned with any inherent fussiness, just in the same way that matrix multiplication or inversion does not prevent us from using matrices. Any complexity associated with GA is readily hidden inside software so that programmers can develop solutions using high-level controls and commands.

1.3 The book's structure

This book is designed to be read in a linear fashion. Chapters 2 to 5 review elementary, complex, vector and quaternion algebra. These chapters are very concise and have been included to provide a unified reference source when some of their features are discussed in later chapters. Those readers already familiar with these topics should consider starting at chapter 6 where geometric conventions such as clockwise and anticlockwise traditions, left and right-handed axial systems are reviewed.

Chapter 7 introduces the reader to the reasons why GA has surfaced in the 21st century rather than the 19th century when it was discovered. Researching this material was very enlightening and brought home the existence of politics in mathematical and scientific progress. GA could have easily become established at the end of the 19th century, but influential mathematicians and scientists of the day decided between them the direction vector analysis would take in future years. Fortunately, enlightened people such as Clifford and Hestenes know a good idea when they see one, and their personal tenacity and dedication have ensured that Grassmann's original ideas have prevailed.

Chapter 8 covers the geometric product, which combine the inner and outer products into a single non-commutative new vector product. Discovering this for the first time is something I will always remember, as its simplicity and structure make one wonder why it took so long to come to become part of everyday vector analysis. Initially, I was cautious of the outer product portion of the geometric product as it possesses imaginary qualities, and I thought that this would be a dominant feature of GA. However, eventually you will discover that elements that square to -1 are so natural you will wonder what all the fuss is about.

Chapter 9 applies GA to calculating reflections and rotations, which is where the power and elegance of the notation emerges. Chapter 10 applies GA to a variety of simple geometric problems encountered in computer graphics. It is far from exhaustive, but illustrates alternative approaches to geometric problem solving.

Chapter 11 addresses the conformal model developed by David Hestenes *et al.* Its use of 5D Minkowski space is a recent development and has natural applications to quantum physics and electrodynamics, but is also being applied to computer graphics. This chapter introduces the reader to the basic concepts, and there is a wide range of technical literature awaiting those readers possessing the appropriate mathematical skills.

Chapter 12 reviews how GA is being compared to existing approaches to algorithm design and the programming implications of GA. Chapter 13 identifies programming tools for GA, and chapter 14 summarizes the book's aims and objectives.