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

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

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.