Modelling Motion: Tracking, Analysis and Inverse Kinematics

Joan Lasenby, Sahan Gamage, and Maurice Ringer

Department of Engineering, University of Cambridge
Trumpington Street, Cambridge CB2 1PZ, UK
{j1,ssh23,mar39}@eng.cam.ac.uk
http://www-sigproc.eng.cam.ac.uk/vision/

Abstract. In this paper we will use the mathematical framework of geometric algebra (GA) to illustrate the construction of articulated motion models. The advantages of solving the forward kinematics of a given problem in this way are that the equations can be constructed in a coordinate-free fashion using the GA representations of rotation, rotors. One can then use the rotor parameters as our variables in a tracking scheme where we use Kalman filters to track real motion data from moving subjects – this has various advantages over the standard Euler angle approach. The paper then looks at the advantages of this system for solving the inverse kinematics – estimating the model given the positions/observations. It will be shown that the often complicated inversion procedures can be simplified by a combination of incidence geometry and rotor inversion.

Keywords: Rotations, geometric algebra, articulated motion, motion estimation, motion modelling, tracking, Kalman filters, forward and inverse kinematics, conformal geometry.

1 Introduction

The main driving force behind the development of the modelling techniques we will describe in subsequent sections has been the need to provide fast and efficient algorithms for optical motion capture. Optical motion capture is a relatively cheap method of producing 3D reconstructions of a subject’s motion over time, the results of which can be used in a variety of applications; biomechanics, robotics, medicine, animation etc. Using a system with few cameras (3 or 4) we find that in order to reliably match and track the data (consisting of bright markers placed at strategic points on the subject) we must use realistic models of the possible motion. Once the data has been tracked using such models, we are in a position to analyse the motion in terms of the rotors we have recovered.

The mathematical language we will use throughout will be that of geometric algebra (GA). This language is based on the algebras of Clifford and Grassmann and the form we follow here is that formalised by David Hestenes [1]. There are now many texts and useful introductions to GA, [2,3,4,5], so we do no more here than outline why it is so useful for the problems we will discuss.
In a geometric algebra of $n$-dimensions, we have the standard *inner* product which takes two vectors and produces a scalar, plus an outer or *wedge* product that takes two vectors and produces a new quantity we call a *bivector* or oriented area. Similarly, the outer product between three vectors produces a *trivector* or oriented volume etc. Thus the algebra has basic elements which are oriented geometric objects of different orders. The highest order object in a given space is called the *pseudoscalar* with the unit pseudoscalar denoted by $I$, e.g. in 3D $I$ is the unit trivector $e_1 \wedge e_2 \wedge e_3$ for basis vectors $\{e_i\}$. Multivectors are quantities which are made up of linear combinations of these different geometric objects. More fundamental than the inner or wedge products is the *geometric product* which can be defined between any multivectors – the geometric product, unlike the inner or outer products, is invertible. For vectors the inner and outer products are the symmetric and antisymmetric parts of the geometric product;

$$ab = a \cdot b + a \wedge b$$

(1)

In effect the manipulations within geometric algebra are keeping track of the objects of different grades that we are dealing with (much as complex number arithmetic does). For a general multivector $X$, we will use the notation $\langle X \rangle_r$ to denote the $r$th grade part of $X$.

In what follows we shall use the convention that vectors will be represented by non-bold lower case roman letters, while we use non-bold, upper case roman letters for multivectors – exceptions to this are stated in the text. Unless otherwise stated, repeated indices will be summed over.

## 2 Rotations

If, in 3D, we consider a rotation to be made up of two consecututive reflections, one in the plane perpendicular to a unit vector $m$ and the next in the plane perpendicular to a unit vector $n$, it can easily be shown [4] that we can represent this rotation by a quantity $R$ we call a *rotor* which is given by

$$R = nm$$

Thus a rotor in 3D is made up of a scalar plus a bivector and can be written in one of the following forms

$$R = e^{-B/2} = \exp \left( -I \frac{\theta}{2} n \right) = \cos \frac{\theta}{2} - In \sin \frac{\theta}{2},$$

(2)

which represents a rotation of $\theta$ radians about an axis parallel to the unit vector $n$ in a right-handed screw sense. Here the bivector $B$ represents the plane of rotation. Rotors act two-sidedly, i.e. if the rotor $R$ takes the vector $a$ to the vector $b$ then

$$b = RaR.$$