New Advances in Geometric Algebra

Chris Doran
Astrophysics Group
Cavendish Laboratory

C.Doran@mrao.cam.ac.uk
www.mrao.cam.ac.uk/~clifford

EPSRC

UNIVERSITY OF CAMBRIDGE

Outline

- History
- Introduction to geometric algebra (GA)
- Rotors, bivectors and rotations
- Interpolating rotors
- Computer vision
- Camera auto-calibration
- Conformal Geometry

What is Geometric Algebra?

- Geometric Algebra is a universal Language for physics based on the mathematics of *Clifford Algebra*
- Provides a new product for vectors
- Generalizes complex numbers to arbitrary dimensions
- Treats points, lines, planes, etc. in a single algebra
- Simplifies the treatment of rotations
- Unites Euclidean, affine, projective, spherical, hyperbolic and conformal geometry

History

- Foundations of geometric algebra (GA) were laid in the 19th Century
- Two key figures: Hamilton and Grassmann
- Clifford unified their work into a single algebra
- Underused (associated with quaternions)
- Rediscovered by Pauli and Dirac for quantum theory of spin
- Developed by mathematicians in the 50s and 60s
- Reintroduced to physics in the 70s

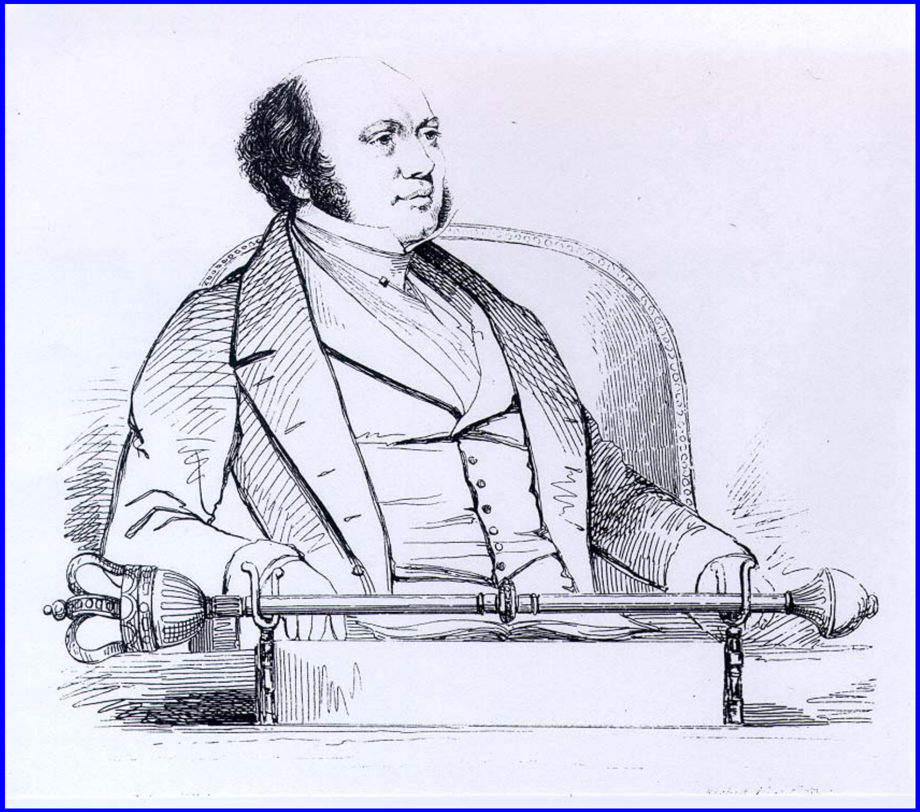
Hamilton

Introduced his quaternion algebra in 1844

$$i^2 = j^2 = k^2 = ijk = -1$$

Generalises complex arithmetic to 3 (4?) dimensions

Very useful for rotations, but confusion over the status of ***vectors***



Grassmann

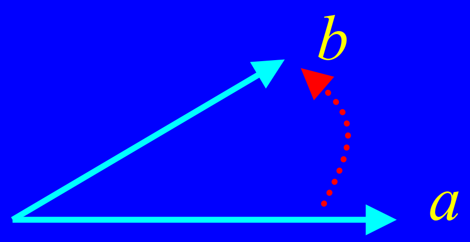
German schoolteacher 1809-1877

Published the *Lineale Ausdehnungslehre* in 1844

Introduced the ***outer product***

$$a \wedge b = -b \wedge a$$

Encodes a plane segment



2D Outer Product

- Antisymmetry implies $a \wedge a = -a \wedge a = 0$
- Introduce basis $\{e_1, e_2\}$

$$a = a_1 e_1 + a_2 e_2 \qquad b = b_1 e_1 + b_2 e_2$$

- Form product

$$\begin{aligned} a \wedge b &= a_1 b_2 e_1 \wedge e_2 + a_2 b_1 e_2 \wedge e_1 \\ &= (a_1 b_2 - b_2 a_1) e_1 \wedge e_2 \end{aligned}$$

- Returns area of the plane + orientation.
- Result is a ***bivector***
- Extends (antisymmetry) to arbitrary vectors

Complex Numbers

- Have a product for vectors in 2D
- Length given by aa^*
- Suggests forming

$$\begin{aligned} ab^* &= (a_1 + a_2 i)(b_1 - b_2 i) \\ &= (a_1 b_1 + a_2 b_2) - (a_1 b_2 - b_2 a_1) i \end{aligned}$$

- Complex multiplication forms the inner and outer products of the underlying vectors!
- Clifford generalised this idea

Clifford

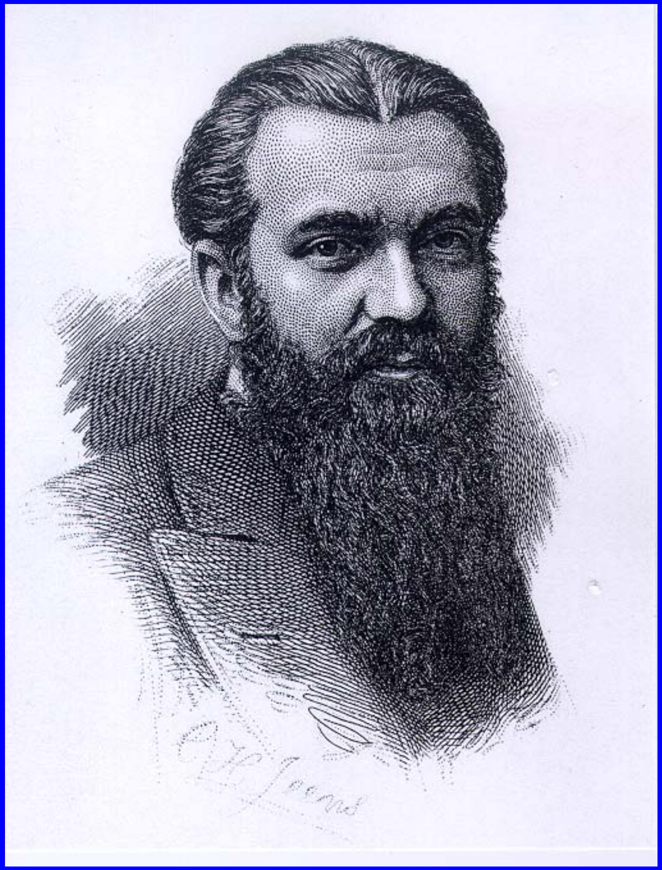
W.K. Clifford 1845-1879

Introduced the ***geometric product***

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

Product of two vectors returns the sum of a scalar and a bivector.

Think of this sum as like the real and imaginary parts of a complex number.



Properties

- Geometric product is associative and distributive

$$a(bc) = (ab)c = abc$$
$$a(b+c) = ab + ac$$

- Square of any vector is a scalar. This implies

$$(a+b)^2 = (a+b)(a+b) = a^2 + b^2 + ab + ba$$

- Can define inner (scalar) and outer (exterior) products

$$a \bullet b = \frac{1}{2}(ab + ba) \qquad a \wedge b = \frac{1}{2}(ab - ba)$$

2D Algebra

- Orthonormal basis is 2D $\{e_1, e_2\}$

$$e_1 \cdot e_1 = e_2 \cdot e_2 = 1 \qquad e_1 \cdot e_2 = 0$$

- Parallel vectors commute

$$e_1 e_1 = e_1 \cdot e_1 + e_1 \wedge e_1 = 1$$

- Orthogonal vectors ***anticommute*** since

$$e_1 e_2 = e_1 \cdot e_2 + e_1 \wedge e_2 = -e_2 \wedge e_1 = -e_2 e_1$$

- Unit bivector has negative square

$$(e_1 \wedge e_2)^2 = (e_1 e_2)(e_1 e_2) = e_1 e_2 (-e_2 e_1)$$

$$= -e_1 e_1 = -1$$

2D Basis

Build into a basis for the algebra

$$1 \qquad e_1, e_2 \qquad e_1 e_2 = I$$

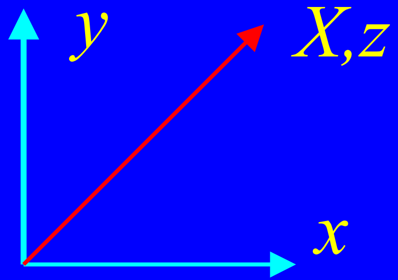
1	2	1
scalar	vectors	bivector

The even grade objects form the complex numbers.

Map between vectors and complex numbers:

$$x + Iy = e_1(xe_1 + ye_2) = e_1 X$$

$$z^* = x - Iy = Xe_1$$



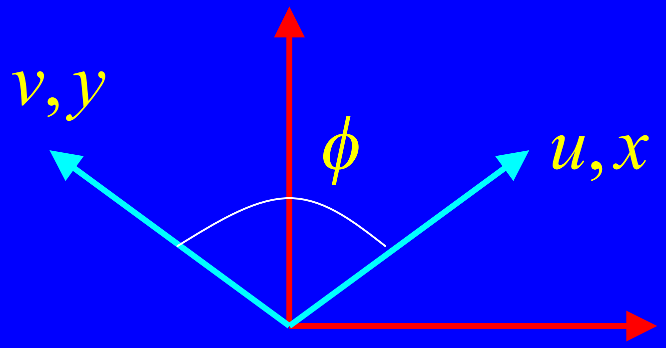
2D Rotations

- In 2D vectors can be rotated using complex phase rotations

$$v = \exp(i\phi)u$$

$$u = e_1 x$$

$$v = e_1 y$$



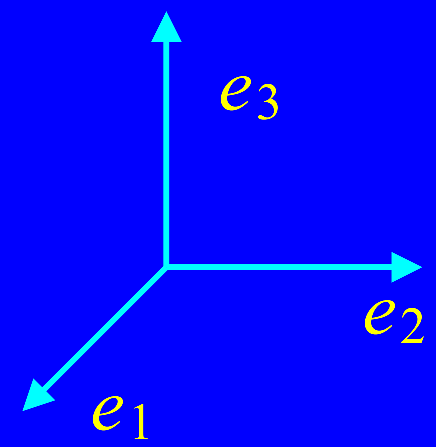
$$y = e_1 v = e_1 \exp(I\phi) e_1 x = e_1(\cos(\phi) + I\sin(\phi)) e_1 x$$

- But $e_1 I e_1 = e_1(e_1 e_2) e_1 = e_2 e_1 = -I$
- Rotation $y = \exp(-I\phi)x = x\exp(I\phi)$

3 Dimensions

- Now introduce a third vector

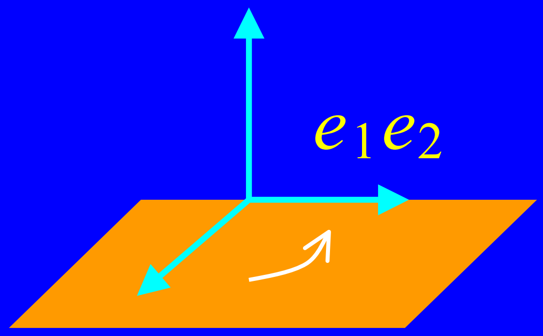
  $\{e_1, e_2, e_3\}$

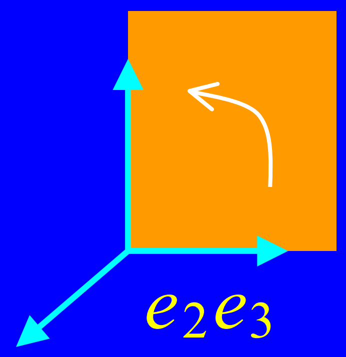


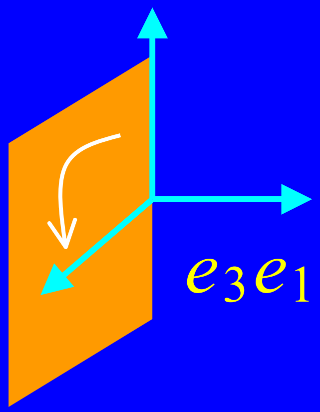
- These all ***anticommute***

  $e_1 e_2 = -e_2 e_1$ *etc.*

- Have 3 bivectors now $\{e_1 e_2, e_2 e_3, e_3 e_1\}$







Bivector Products

- Various new products to form in 3D
- Product of a vector and a bivector

$$e_1(e_1e_2) = e_2 \qquad e_1(e_2e_3) = e_1e_2e_3 = I$$

- Product of two perpendicular bivectors:

$$(e_2e_3)(e_3e_1) = e_2e_3e_3e_1 = e_2e_1 = -e_1e_2$$

- Set

$$i = e_2e_3, \qquad j = -e_3e_1, \qquad k = e_1e_2$$

- Recover quaternion relations

$$i^2 = j^2 = k^2 = ijk = -1$$

3D Pseudoscalar

- 3D Pseudoscalar defined by $I = e_1 e_2 e_3$
- Represents a directed volume element
- Has negative square

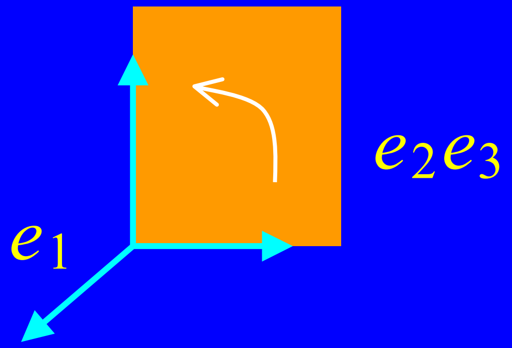
$$I^2 = e_1 e_2 e_3 e_1 e_2 e_3 = e_2 e_3 e_2 e_3 = -1$$

- Commutes with all vectors

$$e_1 I = e_1 e_1 e_2 e_3 = -e_1 e_2 e_1 e_3 = e_1 e_2 e_3 e_1 = I e_1$$

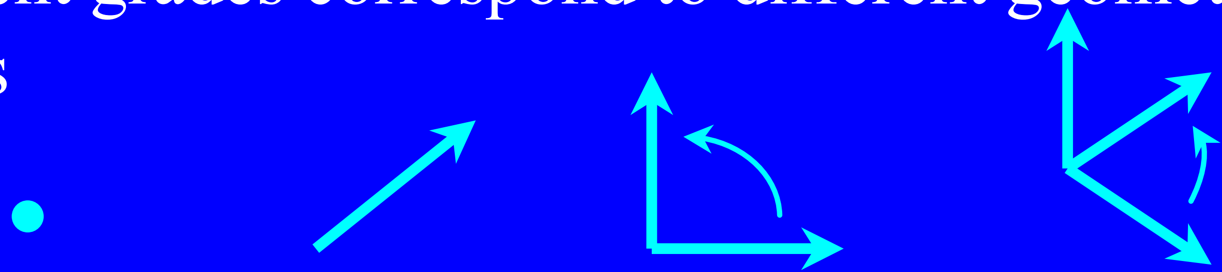
- Interchanges vectors and planes

$$e_1 I = e_2 e_3$$
$$I e_2 e_3 = -e_1$$



3D Basis

Different grades correspond to different geometric objects



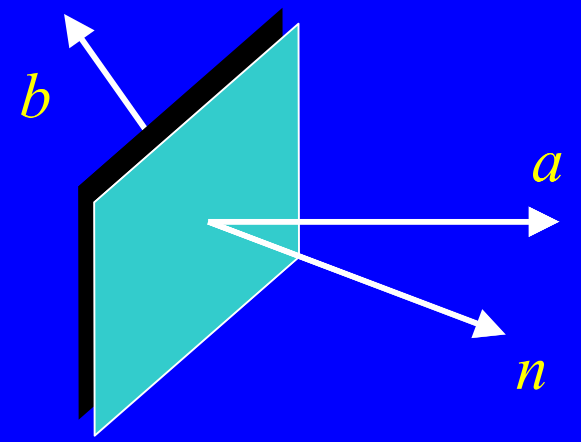
Grade 0	Grade 1	Grade 2	Grade 3
Scalar	Vector	Bivector	Trivector
1	e_1, e_2, e_3	e_1e_2, e_2e_3, e_3e_1	I

Generators satisfy Pauli relations $e_i e_j = \delta_{ij} + \epsilon_{ijk} I e_k$

Recover vector (cross) product $a \times b = -I a \wedge b$

Reflections

- Build rotations from reflections
- Good example of geometric product – arises in ***operations***



$$a_{\parallel} = (a \cdot n)n$$
$$a_{\perp} = a - (a \cdot n)n$$

Image of reflection is

$$b = a_{\perp} - a_{\parallel} = a - 2(a \cdot n)n$$
$$= a - (an + na)n = -nan$$

Rotations

- Two rotations form a reflection

$$a \rightarrow -m(-nan)m = mnanm$$

- Define the rotor $R = mn$
- This is a geometric product! Rotations given by

$$a \rightarrow RaR^{\dagger} \qquad R^{\dagger} = nm$$

- Works in spaces of any dimension or signature
- Works for all grades of multivectors $A \mapsto RAR^{\dagger}$
- More efficient than matrix multiplication

3D Rotations

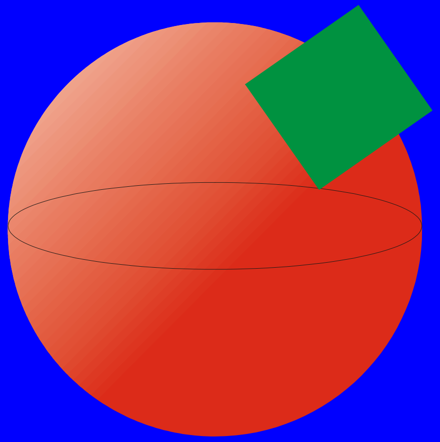
- Rotors even grade (scalar + bivector in 3D)
- Normalised: $RR^{\dagger} = mnnm = 1$
- Reduces d.o.f. from 4 to 3 – enough for a rotation.
- In 3D a rotor is a normalised, even element

$$R = \alpha + B \qquad RR^{\dagger} = \alpha^2 - B^2 = 1$$

- Can also write $R = \exp(-B/2)$
- Rotation in plane B with orientation of B
- In terms of an axis $R = \exp(-\theta In/2)$

Group Manifold

- Rotors are elements of a 4D space, normalised to 1
- They lie on a ***3-sphere***
- This is the ***group manifold***
- ***Tangent space*** is 3D
- Can use Euler angles



$$R = \exp(-e_1 e_2 \phi/2)\exp(-e_2 e_3 \theta/2)\exp(-e_1 e_2 \psi/2)$$

- Rotors R and $-R$ define the same rotation
- Rotation group manifold is more complicated

Lie Groups

- Every rotor can be written as $\exp(-B/2)$
- Rotors form a continuous (*Lie*) group
- Bivectors form a *Lie algebra* under the commutator product
- All finite Lie groups are rotor groups
- All finite Lie algebras are bivector algebras
- (Infinite case not fully clear, yet)
- In conformal case (later) starting point of screw theory (Clifford, 1870s)!

Rotor Interpolation

- How do we interpolate between 2 rotations?
- Form path between rotors

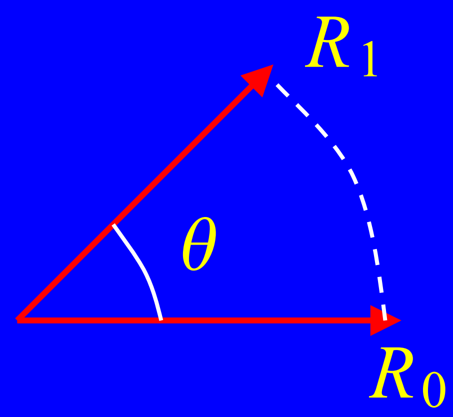
$$R(0) = R_0$$
$$R(1) = R_1$$

$$R(\lambda) = R_0 \exp(\lambda B)$$

- Find B from $\exp(B) = R_0^\dagger R_1$
- This path is invariant. If points transformed, path transforms the same way
- Midpoint simply $R(1/2) = R_0 \exp(-B/2)$
- Works for ***all*** Lie groups

Interpolation 2

- For rotors in 3D can do even better!
- View rotors as unit vectors in 4D
- Path is a circle in a plane
- Use simple trig' to get SLERP



$$R(\lambda) = \frac{1}{\sin(\theta)}\left(\sin((1-\lambda)\theta)R_0 + \sin(\lambda\theta)R_1\right)$$

- For midpoint add the rotors and normalise!

$$R(1/2) = \frac{\sin(\theta/2)}{\sin(\theta)}(R_0 + R_1)$$

Interpolation - Applications

- Rigid body dynamics – interpolating a discrete series of configurations
- Elasticity of a rod or shell
- Framing a curve (either Frenet or parallel transport)
- Calculus – Relax the norm and write

$$RAR^{\dagger} = \psi A \psi^{-1}$$

- ψ belongs to a linear space. Has a natural calculus. Very useful in computer vision.

Rotor Calculus

- Can simplify problems with rotations by relaxing the normalisation and using unnormalised rotors

$$a \to \psi a \psi^\dagger / \rho \qquad \rho = \psi \psi^\dagger$$

- ψ belongs to a *linear* space – has an associated ***calculus***
- Start with obvious result

$$\partial_\psi \langle \psi A \rangle = A$$

- Now write a rotation as

$$RAR^\dagger = \psi A \psi^{-1}$$

Rotor Calculus 2

- Key result (not hard to prove)

$$\partial_\psi \langle \psi^{-1} A \rangle = -\psi^{-1} A \psi^{-1}$$

- Complete the derivation

$$\partial_\psi \langle \psi a \psi^{-1} b \rangle = a\psi^{-1} b - \psi^{-1} b \psi a \psi^{-1}$$

- Pre-multiply by ψ

$$\psi \partial_\psi \langle \psi a \psi^{-1} b \rangle = \psi a \psi^{-1} b - b \psi a \psi^{-1} = 2(RaR^\dagger) \wedge b$$

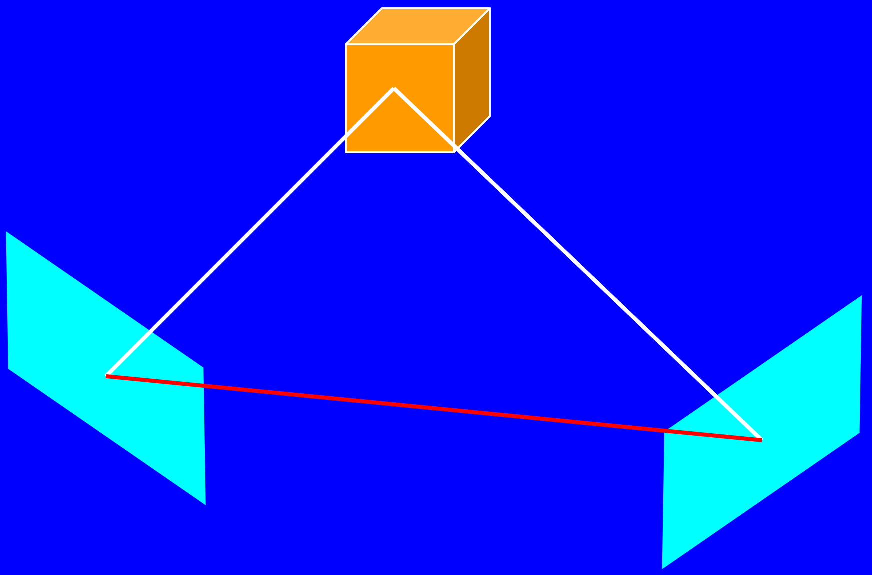
- Note the ***geometric*** product
- Result is a bivector
- Useful in many applications

Computer Vision

- Interested in reconstructing moving scenes from fixed cameras. Various problems:
  - Self calibration - the camera matrix. (Camera parameters can change easily)
  - Localization (Establish camera geometry from point matches).
  - Reconstruction - have to deal with occlusion, etc.
- Standard techniques make heavy use of linear algebra (fundamental matrix, epipolar geometry)
- Rotor treatment provides better alternatives

Camera Auto-calibration

- Different camera views of the same scene
- Given point matches, recover relative translation and rotation between cameras
- Applications
  - Gait analysis
  - Motion analysis
  - Neurogeometry
  - 3D reconstruction



Known Range Data

- Start with simplified problem

$$X = X_i e_i \qquad Y' = Y_i e_i \qquad Y = Y_i f_i$$

$$X - t = Y = RY'R^\dagger$$

- Minimise least square error

$$S = \sum_{k=1}^{n} \left(X^k - t - RY^{k'} R^\dagger \right)^2$$

- Get this from a ***Bayesian*** argument with Gaussian distribution

$$P(X_i) \simeq \exp\left(-\frac{1}{2\sigma^2} (X_i - e_i \cdot \overline{X})^2 \right)$$

Known Range Data 2

- Minimise S wrt t and R

$$\overline{t} = \frac{1}{n}\sum_{k=1}^{n}(X^k - RY^kR^\dagger)$$

$$\psi\partial_\psi S = -4\sum_{k=1}^{n}\left(RY^{k'}R^\dagger\right)\wedge(X^k - t) = 0$$

- Substitute for t write as

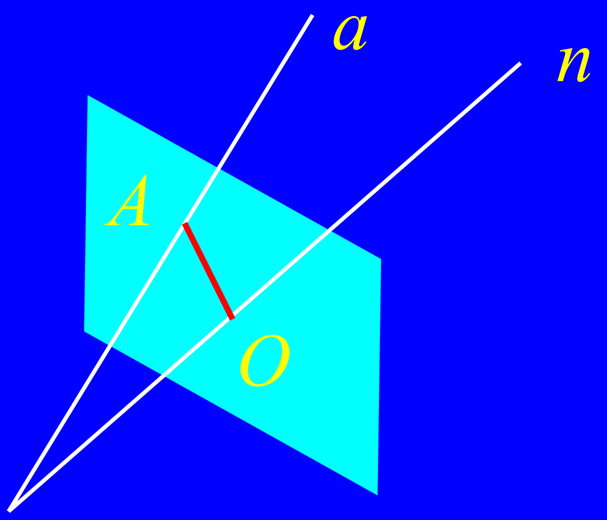
$$e_\alpha \wedge e_\beta F_{\alpha\beta} = 0 \qquad F_{\alpha\beta} = \sum_{k=1}^{n} e_\alpha \cdot Y^{k'}\, e_\beta \cdot X^k$$

- Solve with an SVD of $F_{\alpha\beta}$
- Solution is well known, though derivation extends easily to arbitrary camera numbers

Camera Vision

- Usually only know projective vectors on camera plane

$$n + OA = \lambda a$$



- Choose scale $n^2 = 1$

$$OA = \frac{a - a \cdot n\, n}{a \cdot n} = \frac{a \wedge n}{a \cdot n} n$$

- Represent line with bivector

$$A = \frac{a \wedge n}{a \cdot n} = \frac{a_1}{a_3} e_1 e_3 + \frac{a_2}{a_3} e_2 e_3$$

Homogenous coordinates

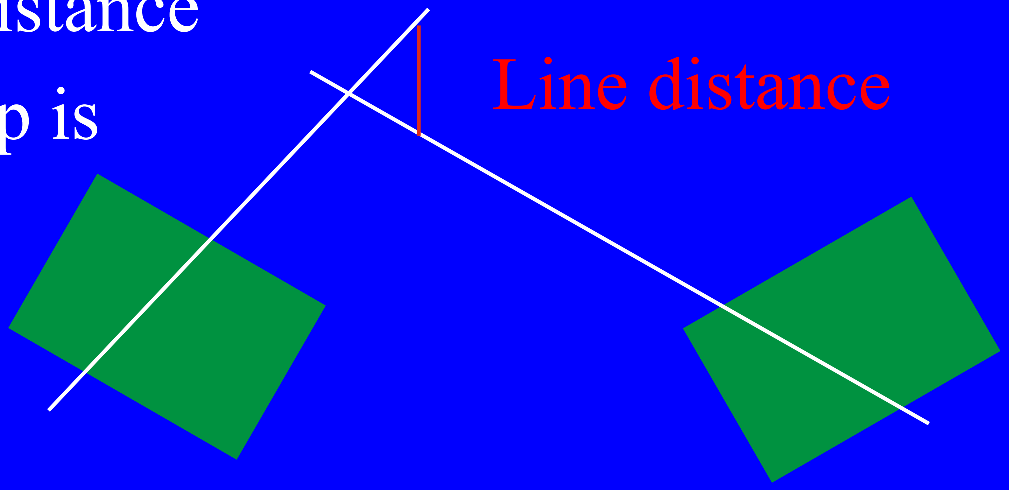
Noise

- Assume camera matrix is known
- 2 main sources of noise:
  – Pixelisation noise (finite resolution)
  – Random noise (camera motion, etc.)
- Assume the noise is all Gaussian, and marginalize over unknown range data
- Get likelihood function

$$S = \sum_{k=1}^{n} \frac{\left(t \wedge x^k \wedge R x^{k'} R^{\dagger} \right)^2}{\left| x^k \wedge R x^{k'} R^{\dagger} \right|^2}$$

Interpretation

- For 2 cameras result has a simple interpretation
- Minimise the line distance
- Problem – the set-up is scale invariant
- Set $t^2 = 1$
- Enforce this as a constraint



Solution

- Include a Lagrange multiplier

$$\sum_{k=1}^{n} (t \cdot n^k)2 - \lambda(t^2 - 1) \qquad n^k = \frac{Ix^k \wedge (Rx^{k'} R^\dagger)}{\left| x^k \wedge (Rx^{k'} R^\dagger) \right|}$$

- Quadratic in t so minimising gives

$$\sum_{k=1}^{n} a \cdot n^k n^k = \lambda t$$

- Define $F(a) = \sum_{k=1}^{n} a \cdot n^k n^k$
- Minimise the lowest eigenvalue of F wrt R
- Numerically efficient and stable

n Cameras

- Previous scheme extends easily to arbitrary camera numbers
- Marginalization integrals more complicated, but can just use sum of individual line distances
- Include Lagrange multipliers for all constraints

$$S = \sum_{k=1}^{n} (t_{12} \cdot n_{12}^{k})^2 + (t_{23} \cdot n_{23}^{k})^2 + (t_{31} \cdot n_{31}^{k})^2$$

$$- 2a \cdot (t_{12} + t_{23} + t_{31}) - \lambda(t_{12}^2 + t_{23}^2 + t_{31}^2 - 1)$$

Consistency Scale

Solution

- Now get 3 symmetric equations

$$\sum_{k=1}^{n} t_{12} \cdot n_{12}^{k} n_{12}^{k} = \lambda t_{12} + a \quad etc$$

- With two constraints

$$t_{12} + t_{23} + t_{31} = 0 \qquad t_{12}^2 + t_{23}^2 + t_{31}^2 = 1$$

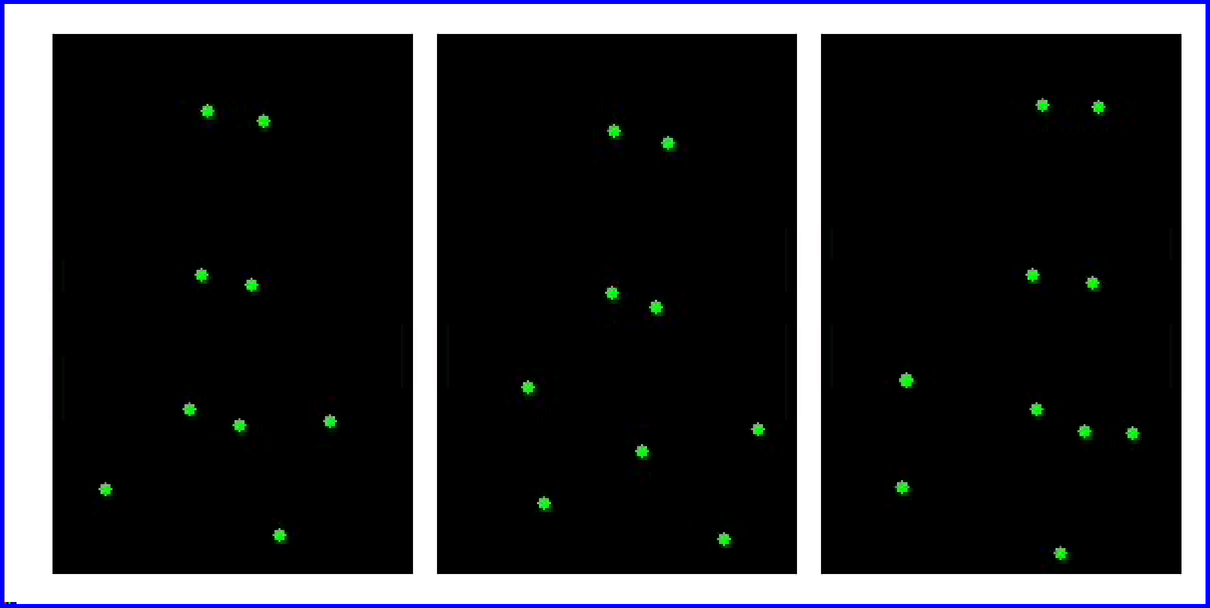
- Re-express as a matrix problem

$$M = \begin{pmatrix} 2F_{12} + (F_{23} + F_{31})/2 & \sqrt{3}\,(F_{23} - F_{31})/2 \\ \sqrt{3}\,(F_{23} - F_{31})/2 & 3(F_{23} + F_{31}) \end{pmatrix}$$

$$Ma = 3\lambda a$$

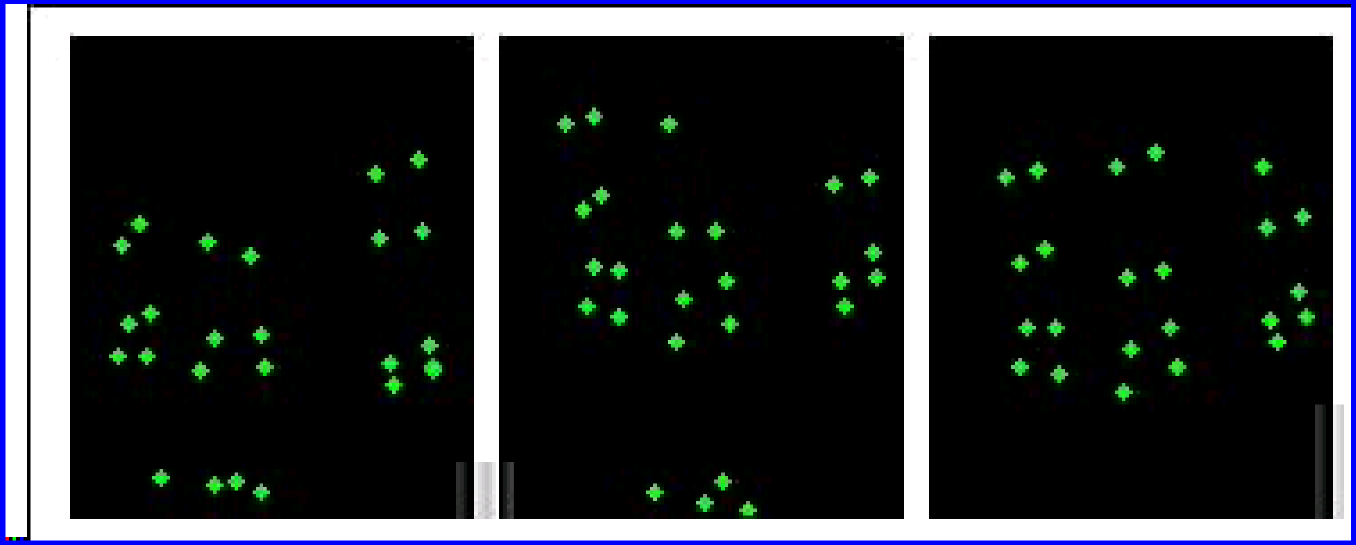
Applications

- Single person walking on model of the Millennium bridge



Applications 2

- 3 people falling in and out of phase

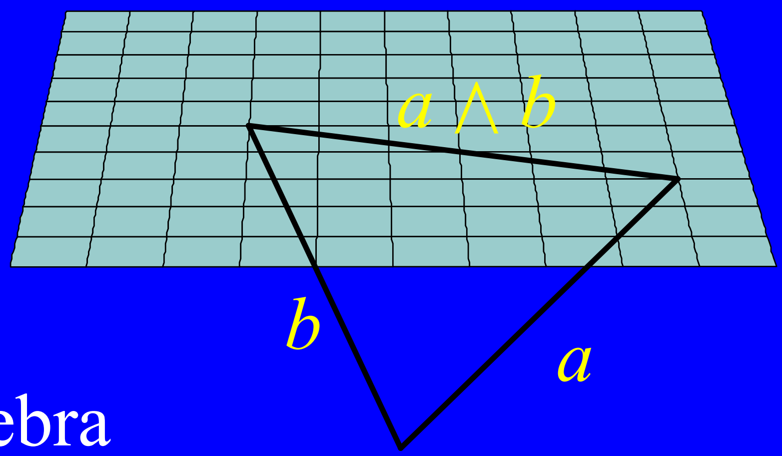


Prospects

- Expand to 7 cameras
- Sports motion analysis studies (running, golf, tennis, hockey …)
- Realistic models to help deal with occlusions, point crossing, *etc*.
- Without an internal model, reconstruction can easily go wrong (example)
- Rigid models make heavy use of rotors for joints
- Include elasticity models for limbs?

Projective Geometry

- Remove the origin by introducing an extra dimension
- Points with vectors
- Lines with bivectors
- Planes with trivectors
- Homogeneous coordinates
- For 3-d space use a 4-d algebra
- Contains 6 bivectors, for lines in space
- Very useful in computer vision and graphics



Projective Line

- Simplest case. Suppose x is a real number and introduce homogeneous coordinates x_1, x_2
- Can translate and invert using

$$x' = \frac{x_1'}{x_{2'}} \qquad \begin{pmatrix} x_1' \\ x_2' \end{pmatrix} = \begin{pmatrix} a & b \\ c & c \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

- This is a linear transformation, but not a rotor group.
- Use the group relation $SL(2,R) \simeq spin(2,1)$

Mixed Signature

- Group relation tells us to consider space $G_{2,1}$
- This is a *mixed signature* space, but no need to modify any of our axioms.
- Basis set $\{e_1, e, \bar{e}\}$ $\quad e^2 = e_1^2 = 1 \quad \bar{e}^2 = -1$
- Everything as before, but now have *null* vectors

$$n = e + \bar{e} \quad \bar{n} = e - \bar{e}$$

$$n^2 = \bar{n}^2 = 0 \quad n \bullet \bar{n} = 2$$

- Also have bivectors with positive square

$$(e\bar{e}e\bar{e})^2 = -ee\bar{e}\bar{e} = +1$$

Map to *R*(2,1)

- Projective points under *SL*(2,*R*) have an invariant inner product

$$\langle a, b \rangle = a_1 b_2 - a_2 b_1$$

- Analogy with *2-spinors* tells us to form the matrix

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} (x_2, -x_1) = \begin{pmatrix} x_1 x_2 & -x_1^2 \\ x_2^2 & -x_1 x_2 \end{pmatrix}$$

- This has vector equivalent

$$x_1 x_2 e_1 + \tfrac{1}{2} x_1^2 n - \tfrac{1}{2} x_2^2 \bar{n}$$

Conformal Geometry

- Projective geometry disposes with the origin as a special point
- But ***infinity*** is not dealt with so easily
- Deal with this by first mapping a line to a circle

$$x \to \left(\frac{2x}{1+x^2}, \frac{1-x^2}{1+x^2} \right)$$

- Introduce a projective dimension, with ***negative*** signature, form

$$\left(\frac{2x}{1+x^2}, \frac{1-x^2}{1+x^2}, 1 \right) \sim \left(x, \frac{1-x^2}{2}, \frac{1+x^2}{2} \right)$$

Null Vectors

- Scale is irrelevant, so have constructed the vector

$$X = 2xe_1 + x^2 n - \bar{n}$$

- Simple to check that

$$X^2 = 0$$

Points as null vectors

- In a 3d space conformal line is intersection of *null cone* with a plane
- Result is a hyperbola
- Picture

Conformal Plane

- Similar trick works for the plane
- Map to complex numbers and go to homogeneous complex coordinates
- Use group relation $SL(2,C) \simeq spin(3,1)$
- This time get null vectors in a 4d space
- Points in *n* dimensions represented as null vectors in (n+1,1) space. This is the *conformal* rep

$$X = 2x + x^2 n - \bar{n}$$

Euclidean Geometry

- Inner product between vectors gives

$$A \cdot B = (a^2 n + 2a - \bar{n}) \cdot (b^2 n + 2b - \bar{n})$$
$$= -2(a-b)^2$$

- Returns the Euclidean distance!
- This means that reflections and rotations in conformal space preserve the Euclidean distance
- Provides a rotor formulation of translations, inversions, *etc*.

Inversions

- Can carry out further Euclidean transforms

$$x \mapsto x^{-1} = x/x^2$$

- For conformal vector, effect is

$$X \mapsto X' = x^{-2}(n + 2x - x^2\bar{n})$$

- Need to swap the null vectors
- Do this with a reflection

$$-ene = \bar{n} \qquad -e\bar{n}e = n$$

- So an inversion reduces to a reflection

$$X \mapsto X' \sim -eXe$$

Translation

- Similar is true for translations. Define the rotor

$$R = \exp(na/2) = 1 + \frac{na}{2}$$

- Find that

$$RXR^\dagger = \left(1 + \frac{na}{2}\right)(x^2 n + 2x - \bar{n})\left(1 - \frac{na}{2}\right)$$
$$= (x+a)^2 n + 2(x+a) - \bar{n}$$

- So set $x' = x + a$
- Have $RXR^\dagger = (x')^2 n + 2x' - \bar{n}$
- Translations performed by rotors!

Lines and Circles

- Lines are encoded as circles through 3 points
- Equation through *A, B, C* is simply

$$X \wedge A \wedge B \wedge C = 0$$

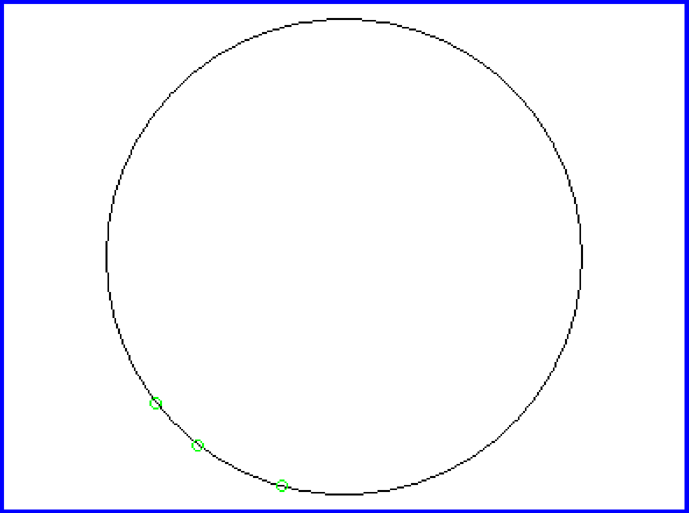
- If one is the point at infinity, the line is straight

•Find centre with

$$c = TnT$$

•A reflection of infinity

•Also have inner product, gives angle between circles



Spheres

- Similarly, 4 points define a sphere

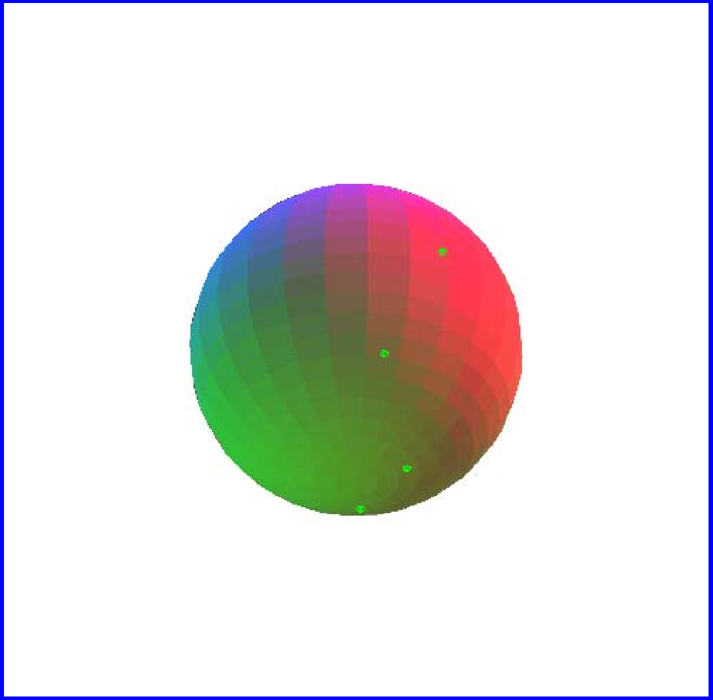
$$X \wedge A \wedge B \wedge C \wedge D = 0$$

- A 4-vector in a 5-d algebra
- Introduce dual

$$a = I A \wedge B \wedge C \wedge D$$

- A positive norm vector

$$\frac{a^2}{(a \cdot n)^2} = \rho^2$$

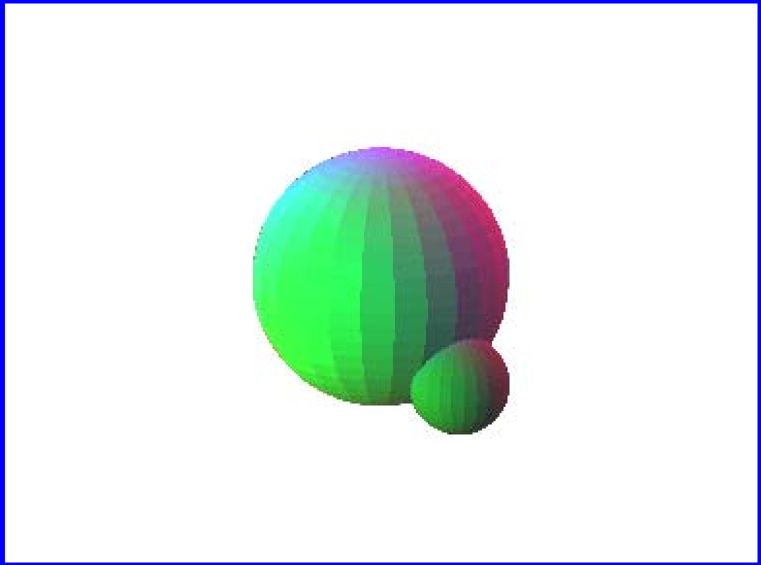


Points are spheres of zero radius

Intersection and Normals

- Projective notions of meet and join extend to circles and spheres
- Form the trivector

  $A \cdot (IB)$
- Extends easily to all objects
- Circles, spheres etc are ***oriented***

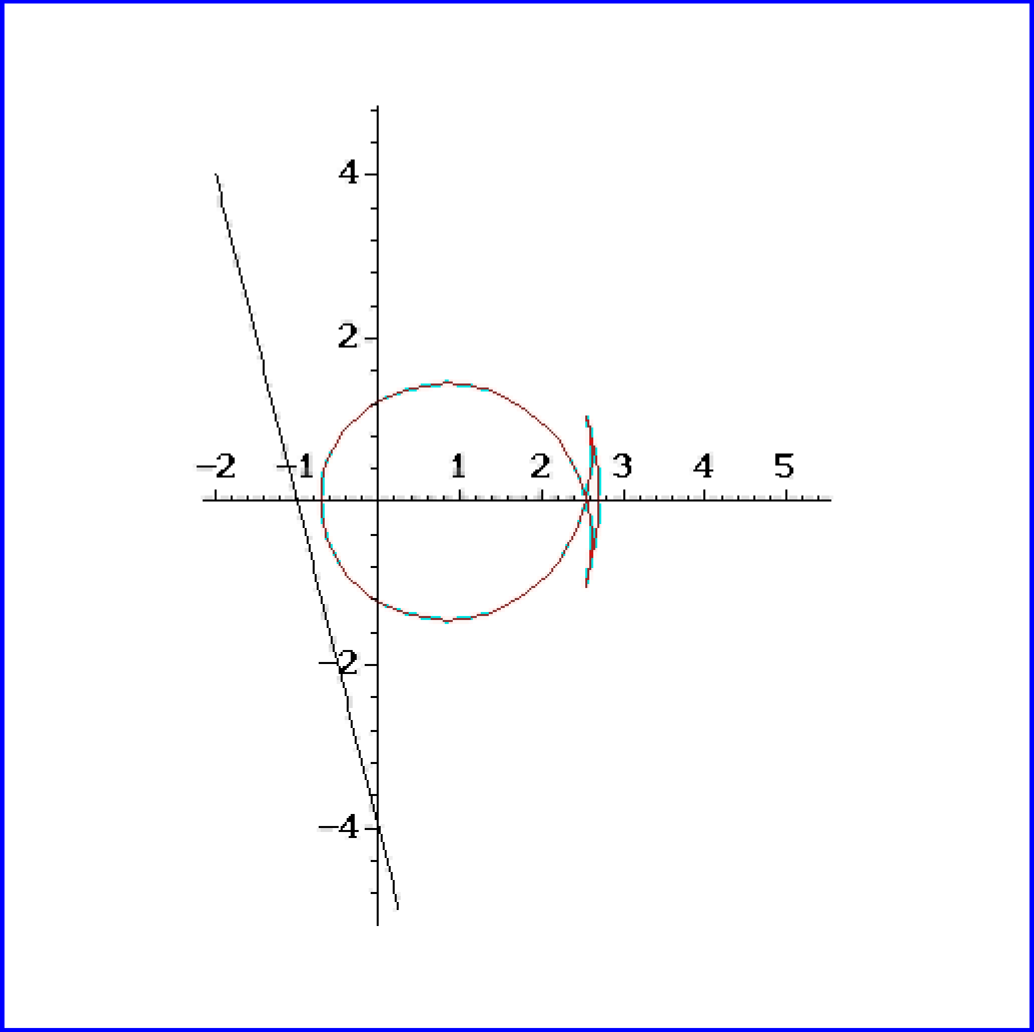


- Multivectors carry round intrinsic definition of the normal to a surface
- Useful for surface propagation

Examples

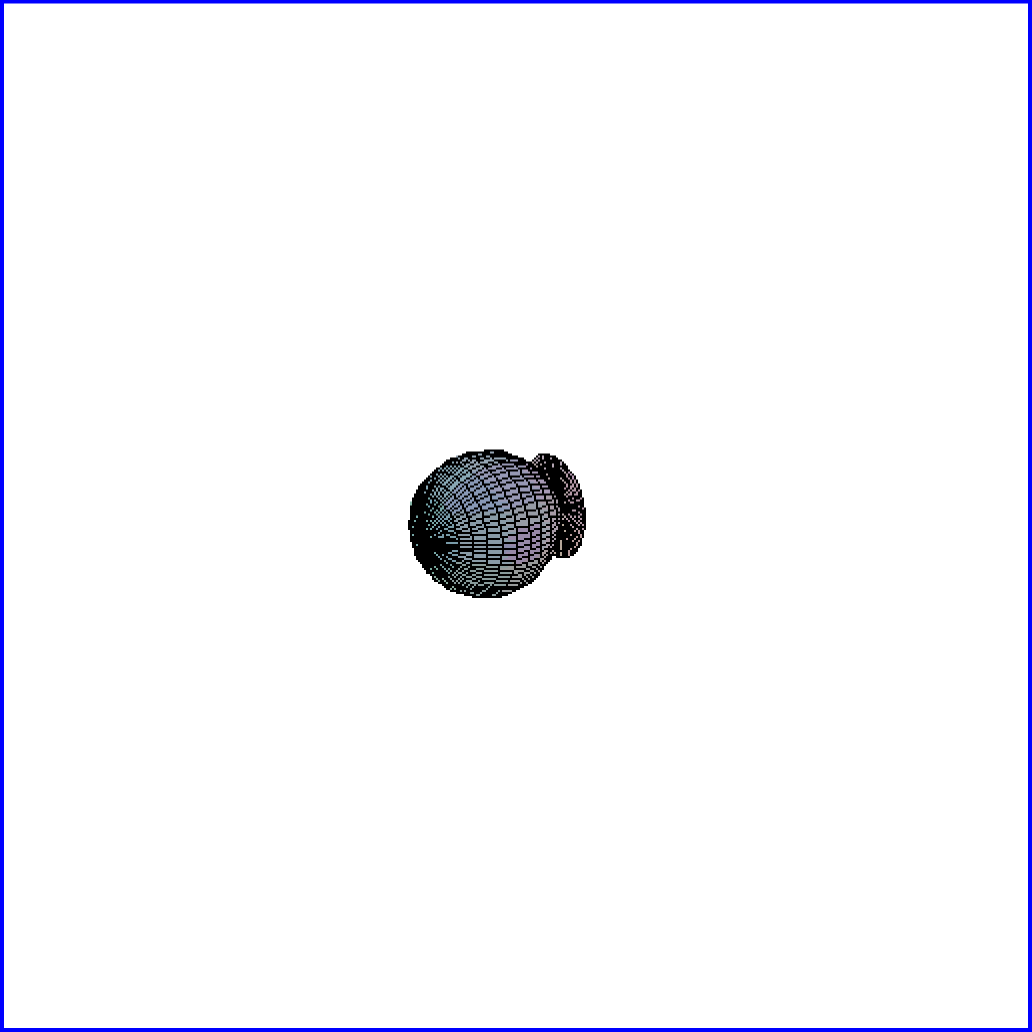
Thanks to Anthony Lasenby

Conformal rep' very useful for spherical propagation and refection



Examples 2

- A 3D Example
- Plenty more of these!
- Currently implemented in Maple
- Conformal algebra will form basis for robust algorithms



Summary

- Conformal geometry unifies Euclidean, spherical and hyperbolic geometry
- Wealth of new techniques for CG industry
- Natural objects for data streaming
- Theorems proved in one geometry have immediate counterparts in others
- Completes 19^{th} century geometry
- Only realised in last 10 years!

Further Details

- All papers on Cambridge GA group website: www.mrao.cam.ac.uk/~clifford
- Applications of GA to computer science and engineering discussed at AGACSE 2001 conference. Next week! www.mrao.cam.ac.uk/agacse2001
- A 1 day course on GA will be presented at SIGGRAPH 2001 in Los Angeles www.siggraph.org/s2001
- IMA Conference in Cambridge, 9th Sept 2002

Applications

- ***Computer graphics***. Build robust algorithms with control over data types
- ***Dynamics***. Algebra of screw theory automatically incorporated
- ***Surface evolution***. Represent evolution and reflections from a surface easily
- ***Robotics***. Combine rotations and translations in a single rotor
- ***Relativity***. Forms the basis of twistor theory.