Reference frame transformation IN THE BRAIN

Travis DeWolf

CNRG - Waterloo University

February 13, 2009

Visuomotor Transformation Model (VTM)

- From *Computations for geometrically accurate visually guided reaching in 3-D space* by Blohm and Crawford

Visuomotor Transformation Model (VTM)

- From *Computations for geometrically accurate visually guided reaching in 3-D space* by Blohm and Crawford
- Mathematical model of transforming eye-centered motor commands into shoulder-centered motor commands

Visuomotor Transformation Model (VTM)

- From *Computations for geometrically accurate visually guided reaching in 3-D space* by Blohm and Crawford
- Mathematical model of transforming eye-centered motor commands into shoulder-centered motor commands
- Describes the early, feed-forward 3-D visuomotor transformation for reach planning

What is the big deal?

- Math shmath, I'm pretty sure this is easy

What is the big deal?

- Math shmath, I'm pretty sure this is easy
- No

Can has example?

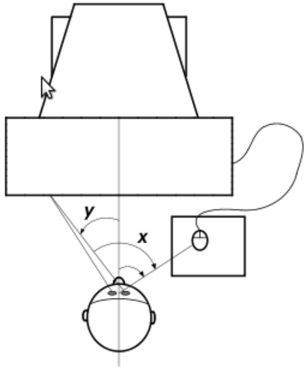


Figure: x is target location, y is the gaze angle

Boom

- Current hand position and target location must be in the same reference frame

Boom

- Current hand position and target location must be in the same reference frame
- Must account for direction of gaze

Boom

- Current hand position and target location must be in the same reference frame
- Must account for direction of gaze
- Listing's and Donder's laws

Boom

- Current hand position and target location must be in the same reference frame
- Must account for direction of gaze
- Listing's and Donder's laws
- Centers of rotation of the eye, head, and shoulder don't align and shift relative to eachother with head rotation

That is so much!

- Lots of things going on here
- VTM is the first attempt at modeling the mathematics behind this reference-frame transformation
- Accomplished with dual quaternions

What the hell is a quaternion?

A quaternion is a magical mathematical 4-tuple that has one real number and three mutually orthogonal imaginary units with real coefficients, and can be used to significantly simplify the calculations involved when rotating an object in 3-D space!

I...see...

It works like this:

- Take your object in 3-D space with rotation θ around axis $\vec{r}$

I...see...

It works like this:

- Take your object in 3-D space with rotation θ around axis $\vec{r}$
- Throw it into a vector $q = (cos(\theta/2), \vec{r}.(sin(\theta/2)))$

I...see...

It works like this:

- Take your object in 3-D space with rotation θ around axis $\vec{r}$
- Throw it into a vector $q = (cos(\theta/2), \vec{r}.(sin(\theta/2)))$
- To rotate it now, multiply it by another quaternion

Quaternion multiplication

So let's say we have two unit quaternions:

- $q_1 = (w_1, x_1, y_1, z_1) = (w_1, v_1)$
- $q_2 = (w_2, x_2, y_2, z_2) = (w_2, v_2)$

Then to multiply q_1 and q_2 we use:

- $q_1 * q_2 = (w_1.w_2 - v_1.v_2, w_1.v_2 + w_2.v_1 + v_1 * v_2)$

where . and $*$ are the standard vector dot and cross product.

Well what's a dual quaternion then?

For a transformation, it's not enough to just be able to represent rotation. We must also be able to represent translation! This is what dual quaternions are for. We write dual quaternion as:

$$Q = (q, \epsilon q_0)$$

Where q and q_0 are quaternions and ϵ is a duality operator where $\epsilon^2 = 0$.

Multiplication?

For two dual quaternions, $Q = (q, \epsilon q_0)$ and $P = (p, \epsilon p_0)$, multiplication works as follows:

$$Q{*}P = q{*}p + \epsilon(q_0{*}p + q{*}p_0)$$

Well goodness, how do we use one?

So, if we want to represent a rotation θ around $\vec{r} = (x, y, z)$ and translation of $\vec{s} = (a, b, c)$ along (x, y, z) we create a dual quaternion

$$Q = (q, \epsilon q_0)$$

where

$$q = (cos(\theta/2), \vec{r}.(sin(\theta/2)))$$
$$q_0 = (0, \vec{s}/2)$$

Well goodness, how do we use one?

Let P_1 be a dual quaternion representing our object's location in 3-D space, to transform it by a dual quaternion D we do the following:

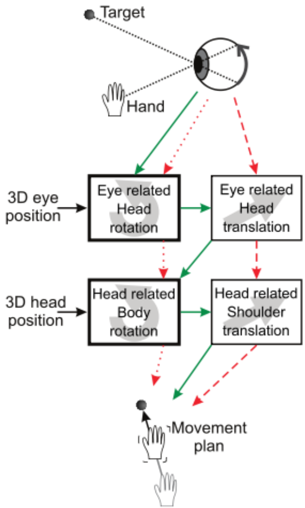
$$P_2 = D * P_1 * D'$$

where $D' = (w_1, -\vec{v_1}, \epsilon(w_2, -\vec{v_2}))$ is the complex conjugate of D and P_2 is the resulting location of the object in 3-D space.

Sweet

Now that we have an absolute understanding of how to use dual quaternions we move on to the Visuomotor Transformation Model (VTM)!

The VTM!


Target
Hand
3D eye position
Eye related Head rotation
Eye related Head translation
3D head position
Head related Body rotation
Head related Shoulder translation
Movement plan

The VTM!

The VTM is comprised of 5 dual quaternions that account for the transformations described on the previous slide.

- Two dual quaternions account for the translations from eye-centered to head-centered, and from head-centered to shoulder-centered using specified average distances (Q_{eyet} and Q_{ht})

The VTM!

The VTM is comprised of 5 dual quaternions that account for the transformations described on the previous slide.

- Two dual quaternions account for the translations from eye-centered to head-centered, and from head-centered to shoulder-centered using specified average distances (Q_{eyet} and Q_{ht})
- Listing and Donders' laws are accounted for with two more dual quaternions (Q_l and Q_d)

The VTM!

The VTM is comprised of 5 dual quaternions that account for the transformations described on the previous slide.

- Two dual quaternions account for the translations from eye-centered to head-centered, and from head-centered to shoulder-centered using specified average distances (Q_{eyet} and Q_{ht})
- Listing and Donders' laws are accounted for with two more dual quaternions (Q_l and Q_d)
- And one dual quaternion to account for occular counterroll in the head (Q_{ocr})

The VTM!

So in the end, when we have the VTM assembled in dual quaternions, we will take our motor command in gaze-centered coordinates P_1 and apply the dual quaternions (in order!) to get our shoulder-centered motor command P_2 like so:

$$\text{Let } Q_{head} = Q_{eyet} * Q_{ocr} * Q_{l}$$
$$\text{Let } Q_{shoulder} = Q_{ht} * Q_{d}$$

Then

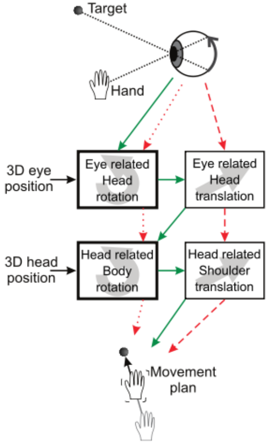
$$P_2 = Q_{shoulder} * Q_{head} * P_1 * Q'_{head} * Q'_{shoulder}$$

Cool. So what do we do with it?

Now we set up some tests!

Experiments!

Remember this?



A bunch of tests later...

From these, Blohm and Crawford came to the conclusion that the body does not simplify the calculations being performed. Instead, an accurate 3-D visuomotor transformation model is used during reference-coordinate transformation.

That is neat! So; biologically plausible?

How biologically plausible is it that these calculations are performed in the brain?

Where in the brain might it happen?

Funny you should ask

Not! Blohm and Crawford state themselves that they don't believe this happens in the brain. However, they do point out that for accurate reference-coordinate transformation to occur, these calculations must be accounted for in some manner.

However, there are a number of places in the brain where things from this model appear to be represented or used. Let's take a look.

- Position of objects in space relative to their location on the retina store in the posterior parietal cortex (PPC)

- Position of objects in space relative to their location on the retina store in the posterior parietal cortex (PPC)
- Hand position signals are represented in gaze-centered coordinates in the PPC as well

- Position of objects in space relative to their location on the retina store in the posterior parietal cortex (PPC)
- Hand position signals are represented in gaze-centered coordinates in the PPC as well
- Wrist movement direction independent of wrist orientation found in the PreMotor cortex (PM)

Gain fields!

Salinas and Abbot presented a more biologically plausible model explaining how coordinate transformation could occur in the brain using gain fields in their 2001 paper *Coordinate transformations in the visual system: How to generate gain fields and what to compute with them*.

Gain fields!

Salinas and Abbot presented a more biologically plausible model explaining how coordinate transformation could occur in the brain using gain fields in their 2001 paper *Coordinate transformations in the visual system: How to generate gain fields and what to compute with them*.

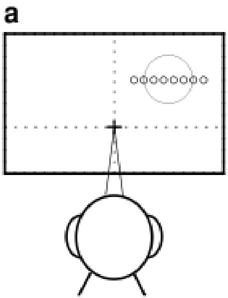
We will briefly look at their model now.

Gain modulation and fields

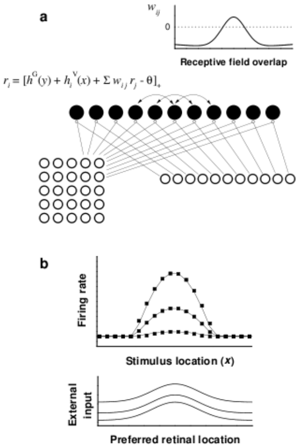
Gain modulation is a change in the response amplitude of a neuron that occurs without a change in the response selection.

'Gain field' was first used to describe the gaze-dependent gain modulation observed in the parietal cortex.

A sweet example



A sweet picture



The basic idea

For a certain target position in retinal coordinates, x, and the gaze angle, y, some parietal neurons are activated and they must drive output neurons such that they encode $x + y$ in different reference coordinates.

So we hook up the gain field neurons to a layer that is trained through supervised learning (ie watching your hand when you move it) such that $x + y$ is output in an appropriate reference frame.

Neat!

So there it is. The VTM nails down the mathematics of reference coordinates transformation and gain fields with supervised neural networks provide a biologically plausible implementation. Teamwork!

Questions?

Any questions?