3D Transforms; Part I - Principles

Vector Spaces

- Vectors, Coordinates, and Bases
- Linear Transformations

Affine Spaces

- Points and Frames
- Homogeneous Coordinates
- Affine Transformations

Geometric Data Types

At this stage, you're probably comfortable thinking of 3D points, and 3D vectors as being represented by 3 numbers. For instance:

$$\begin{bmatrix} 0.125 \\ \pi/4 \\ \sqrt[3]{7} \end{bmatrix}$$

This representation is, however, horribly underspecified. Implied in this notation is an agreed upon ***coordinate system***. This coordinate system has an agreed upon ***set of directions***, and an agreed upon ***origin***. If we change coordinate systems, our representation (set of numbers) changes. If we move points relative to our coordinate system, our representation also changes.

In order to understand and affect these changes we will make our representation more explicit.

We start by defining some notation:

- Points will be denoted as $\dot{p}$
- Vectors will be denoted as $\vec{v}$
- Coordinates are denoted as $\mathbf{c}$
  (a lower-case boldface variable).
  These are those numbers we are so fond of.
- Coordinate Systems are denoted as $\vec{s}^t$

We will deal with two different types of coordinate systems:

A coordinate ***basis*** defines vectors.
A coordinate ***frame*** defines points.

Thus, a completely specified point looks like:

$$\dot{x} = \vec{a}^t \mathbf{u} \qquad \vec{v} = \vec{z}^t \mathbf{s}$$

(If this is unclear, we'll get back to it in a couple of slides)

Vector Spaces

Vectors are actually more simple than points. So we will start our discussion with them. Vectors are entities that live in a *vector space*. A vector space is any set of elements that satisfies the following rules.

1. The operations of addition and scalar multiplication must be defined, and the set must be closed under them:

$$\text{If } \vec{p}, \vec{q} \in V \text{ and } \alpha \in \mathcal{R}, \text{ then } \vec{p} + \vec{q} \in V \text{ and } \alpha\vec{p} \in V$$

2. Associativity of addition must hold:

$$\text{For } \vec{p}, \vec{q}, \text{and } \vec{r} \in V, \quad (\vec{p} + \vec{q}) + \vec{r} = \vec{p} + (\vec{q} + \vec{r})$$

3. There exists a zero vector in *V*, denoted as $\vec{0}$, such that:

$$\text{For } \vec{p} \in V, \quad \vec{0} + \vec{p} = \vec{p} + \vec{0} = \vec{p}$$

4. For every element in *V* there exists an additive inverse:

$$\vec{p} \in V \rightarrow -\vec{p} \in V, \quad \text{such that } (-\vec{p}) + \vec{p} = \vec{p} + (-\vec{p}) = \vec{0}$$

5. Scalar multiplication distributes over addition:

$$\text{If } \vec{p}, \vec{q} \in V \text{ and } \alpha, \beta \in \mathcal{R}, \text{ then } (\alpha + \beta)\vec{p} = \alpha\vec{p} + \beta\vec{p} \text{ and } \alpha(\vec{p} + \vec{q}) = \alpha\vec{p} + \alpha\vec{q}$$

Basis Vectors

A vector basis is a subset of vectors from *V* that can be used to generate any other element in *V*, using just additions and scalar multiplications.

A basis set, $\vec{v}_1, \vec{v}_2, \ldots, \vec{v}_n$, is *linearly dependent* if:

$$\exists \alpha_1, \alpha_2, \ldots, \alpha_n \quad \text{such that } \sum_{i=0}^{n} \alpha_i \vec{v}_i = \vec{0}$$

Otherwise, the basis set is *linearly independent*.
A linearly independent basis set with *i* elements is said to *span* an *i-dimensional* vector space.

Now we'll explain things with pictures.

(Click below)

A basis set can be used to name or address a vector. This is the done by assigning the vector coordinates as follows:

$$\vec{x} = \sum_{i=0}^{3} c_i \vec{v}_i = \begin{bmatrix} \vec{v}_1 & \vec{v}_2 & \vec{v}_3 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \vec{v}^t \mathbf{c}$$

Linear Transformations

A linear transformation, $\mathcal{L}$, is just a mapping from *V* to *V* which satifies the following properties:

$$\mathcal{L}(\vec{u} + \vec{v}) = \mathcal{L}(\vec{u}) + \mathcal{L}(\vec{v}) \text{ and } \mathcal{L}(\alpha\vec{v}) = \alpha\,\mathcal{L}(\vec{v})$$

Linearity implies:

$$\vec{x} \Rightarrow \mathcal{L}(\vec{x}) = \mathcal{L}\left(\sum_i c_i \vec{v}_i\right) = \sum_i c_i\, \mathcal{L}(\vec{v}_i)$$

Expressing $\vec{x}$ with a basis and coordinate vector gives:

$$\begin{bmatrix}\vec{v}_1 & \vec{v}_2 & \vec{v}_3\end{bmatrix}\begin{bmatrix}c_1\\c_2\\c_3\end{bmatrix} \Rightarrow \begin{bmatrix}\mathcal{L}(\vec{v}_1) & \mathcal{L}(\vec{v}_2) & \mathcal{L}(\vec{v}_3)\end{bmatrix}\begin{bmatrix}c_1\\c_2\\c_3\end{bmatrix}$$

Matrices are a common class of linear operators. Furthermore, when a matrix operator is applied to *any* vector the result, $\mathcal{L}(\vec{v}_i)$, is an element of *V*. Thus,

$$\begin{bmatrix}\vec{v}_1 & \vec{v}_2 & \vec{v}_3\end{bmatrix}\begin{bmatrix}c_1\\c_2\\c_3\end{bmatrix} \Rightarrow \begin{bmatrix}\vec{v}_1 & \vec{v}_2 & \vec{v}_3\end{bmatrix}\begin{bmatrix}m_{11} & m_{12} & m_{13}\\ m_{21} & m_{22} & m_{23}\\ m_{31} & m_{32} & m_{33}\end{bmatrix}\begin{bmatrix}c_1\\c_2\\c_3\end{bmatrix}$$

Use of Matrix Operators

We will use matrices to perform *three distinct classes* of operations. In fact, these operations are all duals of one another, but experience suggests that it is best to first understand these operations separately.

We can use a matrix to transform one vector to another:

$$\vec{v}^t \boldsymbol{c} \Rightarrow \vec{v}^t \mathrm{M}\boldsymbol{c}$$

We can use a matrix to change basis vectors:

$$\vec{v}^t \Rightarrow \vec{v}^t \mathrm{M}$$

And, we can use a matrix to change coordinates:

$$\boldsymbol{c} \Rightarrow \mathrm{M}\boldsymbol{c}$$

Once more, some illustrations may help in understanding these distinctions:

(Click on the images below)

How to Read a Matrix Expression

Often we desire to apply sequences of operations to vectors. For instance, we might want to translate a point to the origin, rotate it about some vector, and then translate it back. In order to specify and interpret such sequences, you should become proficient at reading matrix expressions.

The expression

$$\vec{v}^t \boldsymbol{c} \Rightarrow \vec{v}^t \mathbf{M}\boldsymbol{c}$$

can be read in one of two ways depending on the associativity of the multiplication.

Associating the left part of the expression is interpreted as changing the basis while keeping the coordinates fixed

$$\vec{v}^t \boldsymbol{c} \Rightarrow (\vec{v}^t \mathbf{M})\boldsymbol{c} = \vec{l}^t \boldsymbol{c}$$

Associating the right part of the expression is interpreted as changing coordinates while keeping the basis fixed

$$\vec{v}^t \boldsymbol{c} \Rightarrow \vec{v}^t (\mathbf{M}\boldsymbol{c}) = \vec{v}^t \boldsymbol{d}$$

The Basis is Important!

If you are given coordinates and told to transform them using a matrix, you have not been given enough information to determine the final mapping.

Consider the matrix:

$$\mathrm{M} = \begin{bmatrix}2 & 0 & 0\\0 & 1 & 0\\0 & 0 & 1\end{bmatrix}$$

If we apply this matrix to coordinates there *must* be some implied basis, because coordinates alone are not geometric entities (a basis is required to convert coordinates into a vector). Assume this implied basis is $\vec{w}^t$. Thus, our coordinates describe the vector $\vec{v} = \vec{w}^t \boldsymbol{c}$. The resulting transform, $\vec{w}^t \boldsymbol{c} \Rightarrow \vec{w}^t \mathbf{M}\boldsymbol{c}$, will stretch this vector by a factor of 2 in the direction of the first element of the basis set. Of course that direction depends entirely on $\vec{w}^t$.

These illustrations show the significance of the basis when transforming vectors.

(Click on the images below)

Points

Points and vectors are different concepts. A point is a fixed place in space. A vector can be thought of as the motion between points. As mentioned previously, we will distinguish between points and vectors in our notation.

Points are denoted as $\dot{p}$ and vectors as $\vec{v}$.

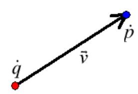
Furthermore, we will consider vectors to live in the linear space ***R*³** and points to to live in the Affine space ***A*³**. Let's consider this distinction.

Conceptually, the operations of addition and multiplication by a scalar are well defined for vectors. The addition of 2 vectors expresses the concatenation of 2 motions. Multiplying a vector by some factor scales the motion.

However, these operations don't make sense for points. What should it mean to add to points together? For example, what is Cambridge plus Boston? What does it mean to multiply a point by an arbitrary scalar? What is 7 times Brookline?

Making Sense of Points

There are some operations that do make sense for points. For instance, if you want to compute a vector that describes the motion from one point to another.



$$\dot{p} - \dot{q} = \vec{v}$$

We'd also like to compute one point that is some vector away from a given point.

$$\dot{q} + \vec{v} = \dot{p}$$

One of the goals of our definitions is to make the subtle distictions between points and vectors apparent. The key distiction between vectors and points are that points are *absolute* whereas vectors are *relative*. We can capture this notion in our definition of a basis set for points. A vector space is completely defined by a set of basis vectors, however, the space that points live in requires the specification of an absolute origin.

$$\dot{p} = \dot{o} + \sum_i c_i \vec{v}_i = \begin{bmatrix}\vec{v}_1 & \vec{v}_2 & \vec{v}_3 & \dot{o}\end{bmatrix}\begin{bmatrix}c_1\\c_2\\c_3\\1\end{bmatrix}$$

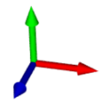
Frames

In our basis definition, we will accommodate this difference between the spaces that points live in and the spaces that vectors live in. We will call the spaces that points live in ***Affine*** spaces, and explain why shortly. We will also call affine-basis-sets ***frames***.

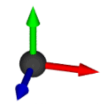
$$\vec{f}^{t} = \begin{bmatrix}\vec{v}_1 & \vec{v}_2 & \vec{v}_3 & \dot{o}\end{bmatrix}$$

In order to use this new basis, we will need to adjust our coordinates. Noting that the origin component of our basis is a point, and remembering from our previous discussion that it makes no sense to multiply points by arbitrary scalar values, we arrive at the following convention for giving points (and vectors) coordinates:

$$\dot{p} = \begin{bmatrix}\vec{v}_1 & \vec{v}_2 & \vec{v}_3 & \dot{o}\end{bmatrix}\begin{bmatrix}c_1\\c_2\\c_3\\1\end{bmatrix} \qquad \vec{x} = \begin{bmatrix}\vec{v}_1 & \vec{v}_2 & \vec{v}_3 & \dot{o}\end{bmatrix}\begin{bmatrix}c_1\\c_2\\c_3\\0\end{bmatrix}$$



A vector basis



An affine basis

Graphically, we will distinguish between vector bases and affine bases (frames) using the convention shown on the left.

Homogeneous Coordinates

Notice, how we have snuck up on the idea of ***Homogeneous Coordinates***, based on simple logical arguments. Keep the following in mind: coordinates are not geometric, they are just scales for basis elements. Thus, you should not be bothered by the fact that our coordinates suddenly have 4 numbers. We could have had more (no one said we have to have a linearly independent basis set).

Note how this approach to coordinates is completely consistent with our intuitions. Subtracting two points yields a vector. Adding a vector to a point produces a point. If you multiply a vector by a scalar you still get a vector. And, in most cases, when you scale points you'll get some nonsense 4th coordinate element that reminds you that the thing you're left with is no longer a point.

$$\begin{bmatrix}a_1\\a_2\\a_3\\1\end{bmatrix} - \begin{bmatrix}b_1\\b_2\\b_3\\1\end{bmatrix} = \begin{bmatrix}a_1-b_1\\a_2-b_2\\a_3-b_3\\0\end{bmatrix}$$

$$\begin{bmatrix}a_1\\a_2\\a_3\\1\end{bmatrix} + \begin{bmatrix}v_1\\v_2\\v_3\\0\end{bmatrix} = \begin{bmatrix}a_1+v_1\\a_2+v_2\\a_3+v_3\\1\end{bmatrix}$$

Isn't it strange how seemingly bizarre things make sense sometimes?

Affine Combinations

There are even certain situations where it *does* make sense to scale and add points.

If you add scaled points together carefully, you can end up with a valid point. Suppose you have two points, one scaled by α_1 and the other scaled by α_2. If we restrict the sum of these alphas, $\alpha_1 + \alpha_2 = 1$, we can assure that the result will have *1* as its 4th coordinate value.

$$\alpha_1 \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ 1 \end{bmatrix} + \alpha_2 \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_1 a_1 + \alpha_2 b_1 \\ \alpha_1 a_2 + \alpha_2 b_2 \\ \alpha_1 a_3 + \alpha_2 b_3 \\ \alpha_1 + \alpha_2 \end{bmatrix} = \begin{bmatrix} \alpha_1 a_1 + \alpha_2 b_1 \\ \alpha_1 a_2 + \alpha_2 b_2 \\ \alpha_1 a_3 + \alpha_2 b_3 \\ 1 \end{bmatrix}$$

This combination defines all points that share the line connecting our two initial points. This idea can be simply extended to 3, 4, or any number of points. This type of constrained-scaled addition is called *affine combination* (hence, the name of our space). In fact, one could define an entire space in terms of the affine combinations of elements by using the α_i*'s* as coordinates, but that is a topic for another day.

Affine Transformations

As with vectors, we can operate on points using matrices. However, we will need to use 4 by 4 matrices since our basis set has four components. However, we will initially limit ourselves to transforms that preserve the integrity of our points and vectors. Literally, those transforms that produce a point or vector when given one of the same.

$$\dot{p} = \begin{bmatrix} \vec{v}_1 & \vec{v}_2 & \vec{v}_3 & \dot{o} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ 1 \end{bmatrix} \Rightarrow \dot{p}' = \begin{bmatrix} \vec{v}_1 & \vec{v}_2 & \vec{v}_3 & \dot{o} \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ 1 \end{bmatrix}$$

This subset of 4 by 4 matrices has the property that a point will be obtained from any input point, and a vector will be obtained from an input vector, independent of the point or vector's coordinates. This subset of matrices is called, you guessed it, the *affine* subset.

Our rules for interpreting left and right association that we developed when transforming vectors still apply here. We can transform affine frames and we can transform the coordinates of points. The next time we meet we will discuss, and give names to, various sub-subsets of these transformations. But doing so is merely mechanics, the magic is all here.

Multiple Transforms

We will often want to specify complicated transformations by stringing together sequences of simple manipulations. For instance, if you want to translate points and then rotate them about the origin. Suppose that the translation is accomplished by the matrix operator **T**, and the rotation is achieved using the matrix, **R**.

Given what we know now it is a simple matter to construct this series of operations.

$$\dot{p} = \vec{w}^t \boldsymbol{c} \Rightarrow \dot{p}' = \vec{w}^t \mathbf{RT}\boldsymbol{c} = \vec{w}^t(\mathbf{R}(\mathbf{T}\boldsymbol{c})) = \vec{w}^t(\mathbf{R}(\mathbf{T}\boldsymbol{c})) = \vec{w}^t(\mathbf{R}\boldsymbol{c}') = \vec{w}^t \mathbf{R}\boldsymbol{c}''$$

Each step in the process can be considered as a transformation of coordinates.

Alternatively, we could have considered the same sequence of operations as follows:

$$\dot{p} = \vec{w}^t \boldsymbol{c} \Rightarrow \dot{p}' = \vec{w}^t \mathrm{RT}\boldsymbol{c} = ((\vec{w}^t \mathrm{R})\mathrm{T})\boldsymbol{c} = (\vec{x}^t \mathrm{T})\boldsymbol{c} = \vec{y}^t \boldsymbol{c}$$

Where each step is considered as a change of basis frames.

These are alternate interpretations of the same transformations. They mean entirely different things, however they result in the same set of transformed points. The first sequence is considered as a transformation about a *global* frame. The second sequence is considered as a change in *local* frames. Frequently, we will mix together these ideas in a single transformation.

The Same Point in Different Frames

Given our framework, some rather difficult problems become easy to solve. For instance, suppose you have 2 frames, and you know the coordinates of a particular point relative to one of them. How would you go about computing the coordinate of your point relative to the other frame?

$$\dot{p} = \vec{w}^t \boldsymbol{c} = \vec{z}_t\ ?$$

Suppose that my two frames are related by the transform **S** as shown below.

$$\vec{z}^t = \vec{w}^t \mathbf{S} \quad \text{and} \quad \vec{w}^t = \vec{z}^t \mathbf{S}^{-1}$$

Thus, the coordinate for the point in the second frame is simply:

$$\dot{p} = \vec{w}^t \boldsymbol{c} = \vec{z}^t \mathbf{S}^{-1} \boldsymbol{c} = \vec{z}^t(\mathbf{S}^{-1}\boldsymbol{c}) = \vec{z}^t \boldsymbol{d}$$

Even harder problems become simple. Suppose that you want to rotate the points describing some object (say a child) about some arbitrary axis in space (say a merry-go-round). This is easy so long as we have the transform relating our two frames.

$$\vec{w}^t \mathrm{M} = \vec{a}^t \quad \text{and} \quad \vec{w}^t = \vec{a}^t \mathrm{M}^{-1} \qquad \text{Thus,} \qquad \vec{w}^t = \vec{a}^t \mathrm{M}^{-1} \Rightarrow \vec{a}^t \mathrm{RM}^{-1} = \vec{w}^t \mathrm{MRM}^{-1}$$

Attribution

Today's lecture comes straight from the mind of a *real* wizard.

(soon to be available in book form)

The keys to his approach are:

- A consistent notation
- A pragmatic approach to representation and transformation
- The introduction of complicated mathematical concepts by appealing to common sense rather than magic.

Next Time