Week 4

Geometric objects include:

- **Points**
- **Scalars**
- **Vectors**

Points

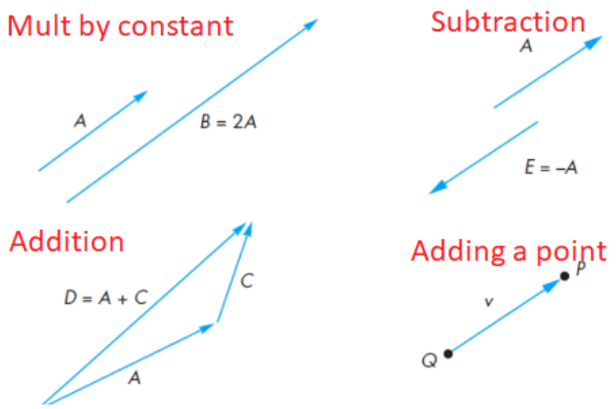
- A fundamental geometric object
- A location in space
- Sizeless and shapeless in mathematics
- Not sufficient alone to describe geometric objects

Scalars

- Real/complex numbers
- Used to identify quantities
- Obey a set of rules
  - Addition, Multiplication
  - Commutativity and Associativity
  - Multiplicative and additive inverses

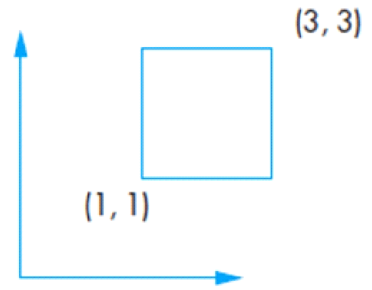
Vectors

- Allows us to work with directions
- Are any quantity with both direction and magnitude (e.g. force, velocity)
- Do not have a fixed location in space
- Have operations:

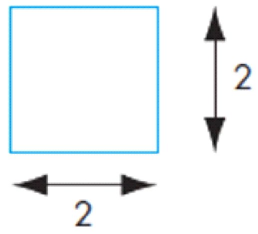


Coordinate Free Geometry

A square with and without coordinates:



Object in a Coordinate System



Object without a Coordinate System

Geometric shapes are independent of coordinate frames, but if we specify them within one, it makes it easy for us to refer to its different parts

Spaces

Scalar field

- A pair of scalars can be combined to form another
- Two operations: addition and multiplication
- Obey the closure, associativity, commutativity, and inverse properties

Linear vector space

- Contains vectors and scalars
- Vector-scalar and vector-vector interactions
- No way of measuring a scalar quantity

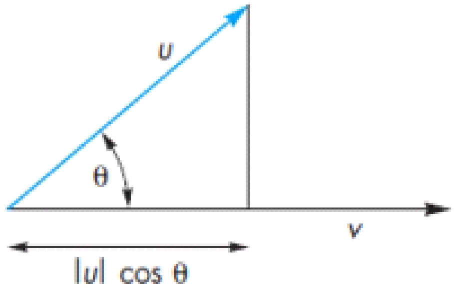
Euclidean Space

- Extension of vector space that adds size/distance

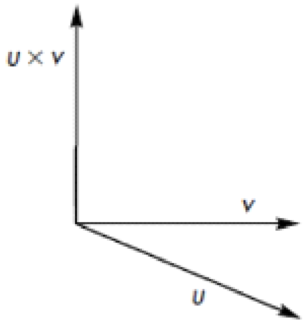
Affine Space

- Extension of vector space that includes points
- Vector-point addition is possible
- Point-point subtraction is possible and produces a vector
- We cannot produce new points
- In this abstract space, objects that can be defined independently of any particular representation

Dot/Inner Product

	The dot product of two vectors ($u \cdot v$) is a scalar $\cos\theta = \dfrac{u \cdot v}{\|u\|\|v\|}$ Square of vector's magnitude = Dot product of itself $\|u\|^2 = u \cdot u.$ If two vectors are orthogonal, their dot product is 0 Orthogonal projection of vector u on v $\|u\| \cos\theta = u \cdot v / \|v\|$

Cross/Outer Product

	The cross product of two vectors ($u \times v$) will give a third vector that is defined by the right hand rule The vector produced is orthogonal to the plane of the original vectors $$\lvert \sin\theta \rvert = \frac{\lvert u \times v \rvert}{\lvert u \rvert \lvert v \rvert}$$ Normal: $n = u \times v.$

Linear Independence

A set of vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ is *linearly independent* if

$$\alpha_1\mathbf{v}_1+\alpha_2\mathbf{v}_2+.. \alpha_n\mathbf{v}_n = 0 \text{ iff } \alpha_1=\alpha_2=\ldots= 0$$

- They are only linearly independent if they have this trivial solution
- If a set of vectors is linearly independent, we cannot represent one vector in terms of the other vectors
- Conversely, if the set is linearly *dependent*, we *can* write one in terms of the others

Example 1

$v1=[1,2]^T$, $v2=[-5,3]^T$

$v1=[2,-1,1]^T$, $v2=[3,-4,2]^T$, $v3=[5,-10,8]^T$

Example 2

$$\mathbf{v}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \mathbf{v}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \mathbf{v}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

For the below to be true, we need each alpha value to be 0

$$\alpha_1 \mathbf{v}_1 + \alpha_2 \mathbf{v}_2 + \alpha_3 \mathbf{v}_3 = \mathbf{0}$$

$$\alpha_1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + \alpha_2 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + \alpha_3 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\Leftrightarrow \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\Leftrightarrow \alpha_1 = \alpha_2 = \alpha_3 = 0$$

Dimension

The **dimension** of a vector space is the fixed *maximum number of linearly independent vectors*

In an n-dimensional space, any set of n linearly independent vectors form a **basis** for the space

Given a basis $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$, any vector $\mathbf{w}$ can be written as

$$\mathbf{w} = \alpha_1\mathbf{v}_1 + \alpha_2\mathbf{v}_2 + \ldots + \alpha_n\mathbf{v}_n$$

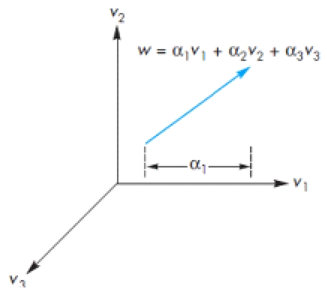
where the coefficients $\{\alpha_i\}$ are unique and are called representations of $\mathbf{w}$

Example

In the example before, the dimension of the space was **3**.

a vector $\mathbf{w} = \begin{pmatrix} 10.5 \\ 21.3 \\ 0.9 \end{pmatrix}$

can be written as $\mathbf{w} = 10.5\ \mathbf{v}_1 + 21.3\ \mathbf{v}_2 + 0.9\ \mathbf{v}_3$ and the coefficients $\alpha_1 = 10.5, \alpha_2 = 21.3,$ and $\alpha_3 = 0.9$ are unique



These are the only alpha values that can represent this vector

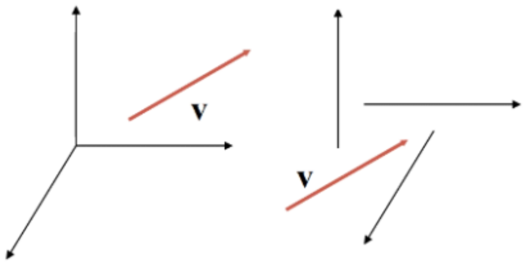
Coordinate Systems

We need a **coordinate system** to act as a <u>frame of reference</u> to relate points and objects to our physical world.

The same point can be represented in its

- World coordinates
- Camera coordinates

Reference Point

	Coordinate systems are *defined by* **three orthogonal basis vectors** They do not need a point, but they can have one Both of these coordinate systems are correct, because vectors represent directions and they have no fixed location

Terminology and Example

- Consider a basis $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$
- A vector $\mathbf{w}$ is written $\mathbf{w} = \alpha_1\mathbf{v}_1 + \alpha_2\mathbf{v}_2 + \ldots + \alpha_n\mathbf{v}_n$
- The list of scalars $\{\alpha_1, \alpha_2, \ldots \alpha_n\}$ is the *representation* of $\mathbf{w}$ with respect to the given basis

We can write the representation as a row or column array of scalars

$$\boldsymbol{\alpha} = [\alpha_1 \quad \alpha_2 \quad \cdots \quad \alpha_n]^{\mathrm{T}} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix}$$

(We can also use square brackets rather than parentheses so we can treat vectors as matrices having just one column.)

For example , let $\mathbf{v} = 2\mathbf{v}_1 + 3\mathbf{v}_2 - 4\mathbf{v}_3$. If $\mathbf{v}_1 = [1 \quad 0 \quad 0]^{\mathrm{T}}$, $\mathbf{v}_2 = [0 \quad 1 \quad 0]^{\mathrm{T}}$, and $\mathbf{v}_3 = [0 \quad 0 \quad 1]^{\mathrm{T}}$, then $\boldsymbol{\alpha} = [2 \quad 3 \quad -4]^{\mathrm{T}}$

- Superscript T means that the values are simply a vertical vector transposed to be horizontal
- Alpha = Representation of vector
- The representation **is with respect to a particular basis**
- The basis is defined with v1, v2 and v3

Changing representation of vectors:

- {v1, v2, v3} and {u1, u2, u3} are two bases that each represent coordinate systems
- Each basis vector in the second set can be represented in terms of the first basis

$$u_1 = \gamma_{11}v_1 + \gamma_{12}v_2 + \gamma_{13}v_3,$$
$$u_2 = \gamma_{21}v_1 + \gamma_{22}v_2 + \gamma_{23}v_3,$$
$$u_3 = \gamma_{31}v_1 + \gamma_{32}v_2 + \gamma_{33}v_3.$$

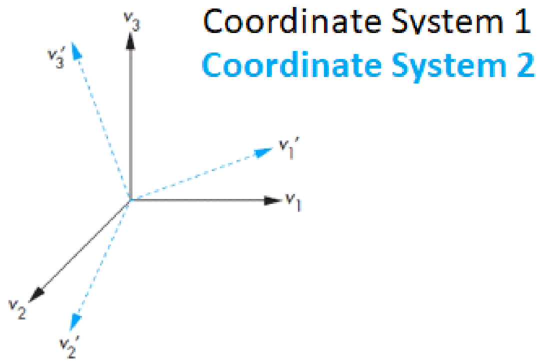
The 3×3 matrix

$$M = \begin{bmatrix} \gamma_{11} & \gamma_{12} & \gamma_{13} \\ \gamma_{21} & \gamma_{22} & \gamma_{23} \\ \gamma_{31} & \gamma_{32} & \gamma_{33} \end{bmatrix}$$

Relating the bases

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = M \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

$$\mathbf{u} = \mathbf{M}\mathbf{v}.$$



Change of Coordinate System 2

<table>
<tr>
<td>Consider the <u>same</u> vector $\mathbf{w}$ with respect to two different coordinate systems having basis vectors $\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$ and $\{\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3\}$. Suppose that
$$\mathbf{w} = \alpha_1\mathbf{v}_1 + \alpha_2\mathbf{v}_2 + \alpha_3\mathbf{v}_3$$
$$\mathbf{w} = \beta_1\mathbf{u}_1 + \beta_2\mathbf{u}_2 + \beta_3\mathbf{u}_3$$
Then the representations are:
$$\mathbf{a} = [\alpha_1 \quad \alpha_2 \quad \alpha_3]^{\mathrm{T}}$$
$$\mathbf{b} = [\beta_1 \quad \beta_2 \quad \beta_3]^{\mathrm{T}}$$
Equivalently,
$$\mathbf{w} = [\alpha]^{\mathrm{T}}\mathbf{v} \text{ and } \mathbf{w} = [\beta]^{\mathrm{T}}\mathbf{u}$$
7</td>
<td>We are trying to change the definition of W from coordinate system to another (from v to u)

w is represented by a and b</td>
</tr>
</table>

Thus, $$w = \mathbf{b}^T \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \mathbf{b}^T \mathbf{M} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \mathbf{a}^T \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$ and $$\mathbf{a} = \mathbf{M}^T \mathbf{b}.$$ Also $\mathbf{b} = \mathbf{T}\mathbf{a}$ where, $\mathrm{T} = (\mathbf{M}^T)^{-1}$	In matrix multiplication, the columns of the first matrix must match the rows of the second matrix

Example

Suppose we have a vector w whose representation in some basis is:

$$\mathbf{a} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

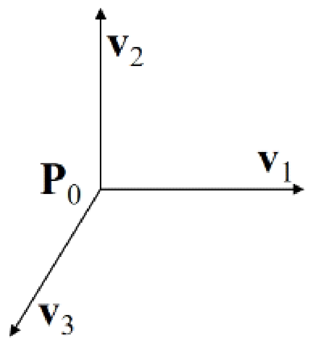
We want to **represent this vector in a new basis system**

$$u_1 = v_1,$$

$$u_2 = v_1 + v_2,$$

$$u_3 = v_1 + v_2 + v_3.$$

Coordinate FRAMES are *defined by* **three orthogonal basis vectors** AND an origin point (P0)



Coordinates Frame are Used for Representation

A coordinate frame is defined by: $\boldsymbol{P0, v1, v2, v3}$

Every vector (v) can be written as:

$$\mathbf{v} = \alpha_1\mathbf{v}_1 + \alpha_2\mathbf{v}_2 + \alpha_3\mathbf{v}_3$$

Every point can be written as:

$$\mathbf{P} = \mathbf{P}_0 + \beta_1\mathbf{v}_1 + \beta_2\mathbf{v}_2 + \beta_3\mathbf{v}_3$$

(We start from the origin then the vectors move the point to the correct spot)

Homogenous Coordinates

Sunday, 15 March 2020 4:08 PM

Non-Homogenous Coordinate Representations

Monday, 9 March 2020 6:20 PM

The point P and the vector (v) are <u>not homogenous</u> coordinate representations here as they have a **different number of terms**

$$\mathbf{P} = \mathbf{P}_0 + \beta_1\mathbf{v}_1 + \beta_2\mathbf{v}_2 + \beta_3\mathbf{v}_3$$

$$\mathbf{v} = \alpha_1\mathbf{v}_1 + \alpha_2\mathbf{v}_2 + \alpha_3\mathbf{v}_3$$

This representation is not good either because it is easy to confuse points with vectors.

$$\mathbf{P} = [\beta_1, \beta_2, \beta_3]^{\mathrm{T}}, \mathbf{v} = [\alpha_1, \alpha_2, \alpha_3]^{\mathrm{T}}$$

Homogenous Coordinate Representations

Since $0 \cdot \mathbf{P} = \mathbf{0}$ and $1 \cdot \mathbf{P} = \mathbf{P}$ then we can write

$$\mathbf{v} = \alpha_1\mathbf{v}_1 + \alpha_2\mathbf{v}_2 + \alpha_3\mathbf{v}_3 = \alpha_1\mathbf{v}_1 + \alpha_2\mathbf{v}_2 + \alpha_3\mathbf{v}_3 + 0 \cdot \mathbf{P}_0$$
$$\mathbf{P} = \beta_1\mathbf{v}_1 + \beta_2\mathbf{v}_2 + \beta_3\mathbf{v}_3 + \mathbf{P}_0 = \beta_1\mathbf{v}_1 + \beta_2\mathbf{v}_2 + \beta_3\mathbf{v}_3 + 1 \cdot \mathbf{P}_0$$

We can obtain a **four dimensional *homogenous coordinate representation***

This the most desirable representation. It involves matrix multiplication

$$\mathbf{v} = [\alpha_1 \quad \alpha_2 \quad \alpha_3 \quad 0]^{\mathrm{T}}\boldsymbol{v} = [\alpha_1 \quad \alpha_2 \quad \alpha_3 \quad 0]\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ P_0 \end{bmatrix}$$

$$\mathbf{P} = [\beta_1 \quad \beta_2 \quad \beta_3 \quad 1]^{\mathrm{T}}\boldsymbol{v}$$

Homogenous Coordinates for 3D points

The homogeneous coordinate form for a three dimensional point $[x \quad y \quad z]^{\mathrm{T}}$ is given as

$$\mathbf{p} = [x \quad y \quad z \quad 1]^{\mathrm{T}} = [wx \quad wy \quad wz \quad w]^{\mathrm{T}} = [x' \quad y' \quad z' \quad w]^{\mathrm{T}}$$

Homogenous coordinates replace points in three dimensions by lines through the origin in four dimensions

We return to a 3D point (w is not 0) by dividing the dash values by w

$$x \leftarrow x'/w$$
$$y \leftarrow y'/w$$
$$z \leftarrow z'/w$$

If w=0, the representation is identical to a vector's representation
For w=1, the point representation is:

$$[x \quad y \quad z \quad 1]^{\mathrm{T}}$$

Use in Graphics

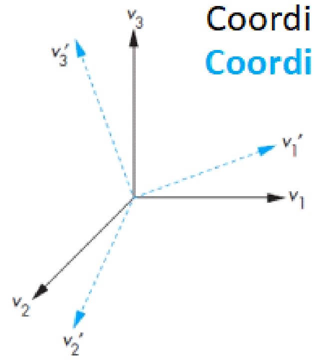
Homogeneous coordinates are key to all computer graphics systems

- All standard transformations (rotation, translation, scaling) can be implemented with matrix multiplications using 4 x 4 matrices (since both points and vectors have 4 components)
- Hardware pipeline works with 4 dimensional representations
- For orthographic viewing, we can maintain **w=0 for vectors** and **w=1 for points**
- For perspective, we need a perspective division (**w is another value**)

Representing a 2nd Basis in terms of the 1st

How can we relate **a** with **b**? (two coordinate bases)
a and **b** are each represented by a set of three vectors

- {v1, v2, v3} and {u1, u2, u3} are two bases that each represent coordinate systems
- Each basis vector in the second set can be represented in terms of the first basis



Coordinate System 1
Coordinate System 2

$$u_1 = \gamma_{11}v_1 + \gamma_{12}v_2 + \gamma_{13}v_3,$$
$$u_2 = \gamma_{21}v_1 + \gamma_{22}v_2 + \gamma_{23}v_3,$$
$$u_3 = \gamma_{31}v_1 + \gamma_{32}v_2 + \gamma_{33}v_3.$$

Matrix Multiplication for Each

$\mathbf{u}_1 = \gamma_{11}\mathbf{v}_1 + \gamma_{12}\mathbf{v}_2 + \gamma_{13}\mathbf{v}_3$ can be written as: $$\mathbf{u}_1 = [\mathbf{v}_1 \quad \mathbf{v}_2 \quad \mathbf{v}_3]\begin{bmatrix}\gamma_{11}\\ \gamma_{12}\\ \gamma_{13}\end{bmatrix} = \mathbf{V}\begin{bmatrix}\gamma_{11}\\ \gamma_{12}\\ \gamma_{13}\end{bmatrix}$$	In matrix multiplication, the row from the first is multiplied by the column of the second. u1 = Vector matrix (V) * weight factor column matrix
$\mathbf{u}_2 = \gamma_{21}\mathbf{v}_1 + \gamma_{22}\mathbf{v}_2 + \gamma_{23}\mathbf{v}_3$ and $\mathbf{u}_3 = \gamma_{31}\mathbf{v}_1 + \gamma_{32}\mathbf{v}_2 + \gamma_{33}\mathbf{v}_3$ can be written as: $$\mathbf{u}_2 = \mathbf{V}\begin{bmatrix}\gamma_{21}\\ \gamma_{22}\\ \gamma_{23}\end{bmatrix}$$ $$\mathbf{u}_3 = \mathbf{V}\begin{bmatrix}\gamma_{31}\\ \gamma_{32}\\ \gamma_{33}\end{bmatrix}$$	We can do this for the other vectors in the other basis (u).

Condensed Representation

We can put the weight factors into a 3x3 matrix:

$$\mathbf{M} = \begin{bmatrix} \gamma_{11} & \gamma_{12} & \gamma_{13} \\ \gamma_{21} & \gamma_{22} & \gamma_{23} \\ \gamma_{31} & \gamma_{32} & \gamma_{33} \end{bmatrix}$$

This gives us: Each (u) vector is coordinate system X weight matrix (TRANSPOSED)

$$[\mathbf{u}_1 \quad \mathbf{u}_2 \quad \mathbf{u}_3] = \mathbf{V}\,\mathbf{M}^{\mathrm{T}}$$

Thus:

$$\mathbf{U} = \mathbf{V}\,\mathbf{M}^{\mathrm{T}}$$

Vector (w) Represented in 2 Coordinate Systems

$$\mathbf{w} = \alpha_1\mathbf{v}_1 + \alpha_2\mathbf{v}_2 + \alpha_3\mathbf{v}_3 \quad \mathbf{w} = \begin{bmatrix}\mathbf{v}_1 & \mathbf{v}_2 & \mathbf{v}_3\end{bmatrix}\begin{bmatrix}\alpha_1\\ \alpha_2\\ \alpha_3\end{bmatrix} = \mathbf{V}\,\mathbf{a}$$

$$\mathbf{w} = \beta_1\mathbf{u}_1 + \beta_2\mathbf{u}_2 + \beta_3\mathbf{u}_3 \quad \mathbf{w} = \begin{bmatrix}\mathbf{u}_1 & \mathbf{u}_2 & \mathbf{u}_3\end{bmatrix}\begin{bmatrix}\beta_1\\ \beta_2\\ \beta_3\end{bmatrix} = \mathbf{U}\,\mathbf{b}$$

$w = Va$, and $w = Ub$, and so $Va = Ub$

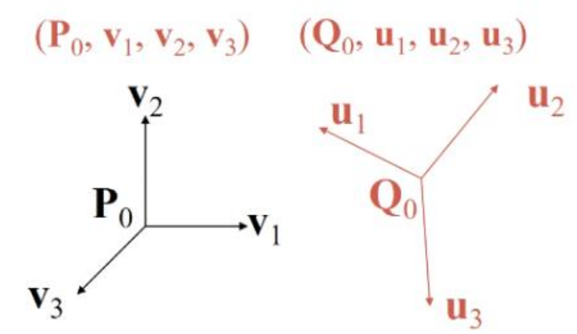
$U = VM^T$, so $Va = VM^Tb \quad \rightarrow \quad a = M^Tb$

Thus a and b are related by M^T

Change of Coordinate Frames

We can apply a similar process in homogeneous coordinates to the representations of both points and vectors.

Consider two coordinate frames:



Each has a point and three vectors

Any point or vector *can be represented in either coordinate frame*

We can represent $(\mathbf{Q}_0, \mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3)$ in terms of $(\mathbf{P}_0, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$

Represent a Coordinate Frame in Terms of Another

We can extend what we did with the change of basis vectors

$\mathbf{u}_1 = \gamma_{11}\mathbf{v}_1 + \gamma_{12}\mathbf{v}_2 + \gamma_{13}\mathbf{v}_3$ $\mathbf{u}_2 = \gamma_{21}\mathbf{v}_1 + \gamma_{22}\mathbf{v}_2 + \gamma_{23}\mathbf{v}_3$ $\mathbf{u}_3 = \gamma_{31}\mathbf{v}_1 + \gamma_{32}\mathbf{v}_2 + \gamma_{33}\mathbf{v}_3$ $\mathbf{Q}_0 = \gamma_{41}\mathbf{v}_1 + \gamma_{42}\mathbf{v}_2 + \gamma_{43}\mathbf{v}_3 + \mathbf{P}_0$	For Q0, we use the other frame's point (P0) In matrix below, the 1 comes from P0 There are zeroes because there are no additions for the other 3.

We replace the 3x3 matrix by a 4x4 matrix as follows:

$$\mathbf{M} = \begin{bmatrix} \gamma_{11} & \gamma_{12} & \gamma_{13} & 0 \\ \gamma_{21} & \gamma_{22} & \gamma_{23} & 0 \\ \gamma_{31} & \gamma_{32} & \gamma_{33} & 0 \\ \gamma_{41} & \gamma_{42} & \gamma_{43} & 1 \end{bmatrix}$$

Working with Representations

Within the two coordinate frames any point or vector has a representation of the same form:

$\mathbf{a} = [\alpha_1 \quad \alpha_2 \quad \alpha_3 \quad \alpha_4]$ in the first frame

$\mathbf{b} = [\beta_1 \quad \beta_2 \quad \beta_3 \quad \beta_4]$ in the second frame

where $\alpha_4 = \beta_4 = 1$ for points and $\alpha_4 = \beta_4 = 0$ for vectors and

$$\mathbf{a} = \mathbf{M}^{\mathrm{T}}\mathbf{b}$$

The matrix $\mathbf{M}^{\mathrm{T}}$ is 4×4 and specifies an affine transformation in homogeneous coordinates

Common Transformations

Sunday, 15 March 2020 5:04 PM

Rigid Transformation preserves **everything** (angle/shape, length, area)

- Equivalent to a change in coordinate frames
- Has 6 degrees of freedom (3 rotations, 3 translations, along each of the three axes)

The 4x4 matrix has the form:

$\begin{bmatrix} R & \mathbf{t} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix}$	• R is a 3x3 rotation matrix • $t \in \mathbb{R}^3$ is a translation vector • 0 (Transposed (Col to Row)) is a vector of zeroes

Similarity Transformation preserves **angle** and ***ratios of lengths/areas***

- Has 7 degrees of freedom (additional degree = scaling equally in all 3 dimensions)

The matrix has the form:

$\begin{bmatrix} sR & \mathbf{t} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix}$ or $\begin{bmatrix} R & \mathbf{t} \\ \mathbf{0}^{\mathrm{T}} & s' \end{bmatrix}$	Scaling factors = $s, s' \neq 1$(otherwise it becomes a rigid trans.) <u>Large values of s</u> and **small values of s'** *enlarge the object*

Affine Transformation preserves **parallelism** and ***ratios of lengths*** (not angles)

- Has 12 degrees of freedom
  - Because 4 elements in last row of matrix are fixed
  - 3 rotations + 3 translations + 3 scaling + 3 shear

The 4x4 matrix has the form:

$\begin{bmatrix} A & \mathbf{t} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix}$	• A is a 3x3 nonsingular matrix • $t \in \mathbb{R}^3$ is a translation vector • 0 (Transposed) is a vector of zeroes

Perspective Transformation

Perspective Transformation preserves **cross ratios** (*ratios of ratios of lengths*)

Happens when you image something with a camera

The matrix can be any non-singular 4x4 matrix

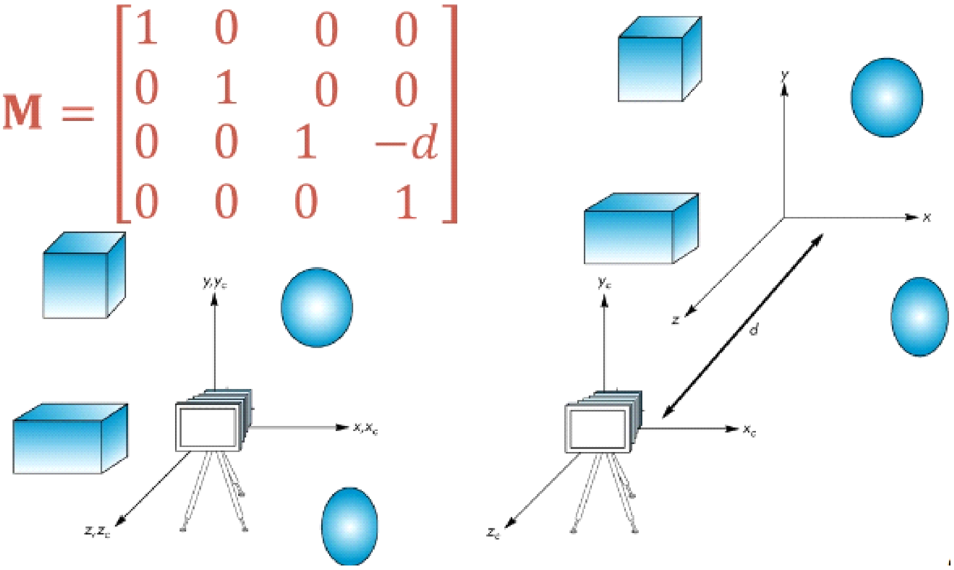
The World and Camera Coordinate Frames

- When we work with representations, we work with n-tuples or arrays of scalars
- *Changes in coordinate frame are then defined by 4 x 4 matrices*
- In OpenGL, the <u>base frame that we start with is the world frame</u>
- Eventually we represent entities in the camera frame by changing the world representation using the model-view matrix
- Initially these frames are the same (i.e. M = I). The transformation is an identity matrix

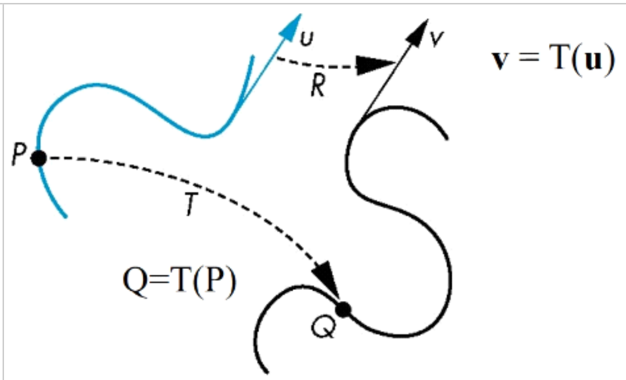
Moving the Camera

If objects are on both sides of $z = 0$, we must move the camera coordinate frame

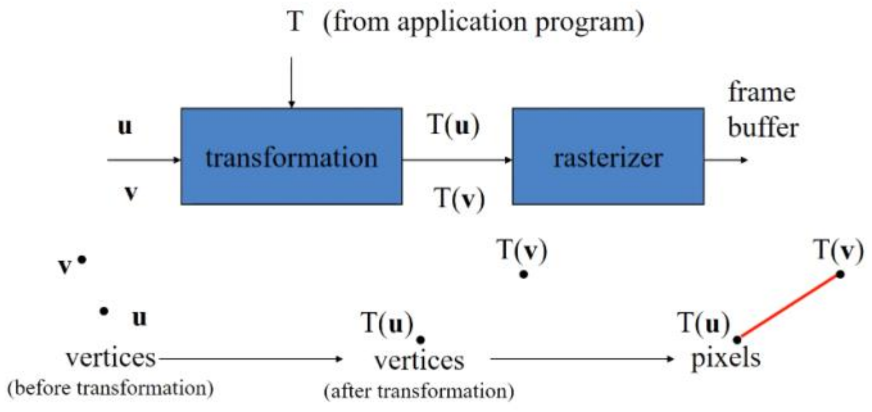
$$\mathbf{M} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



General Transformations

A **transformation** maps: • Points to other points AND/OR • Vectors to other vectors 1 to 1 mapping	

Pipeline Implementation

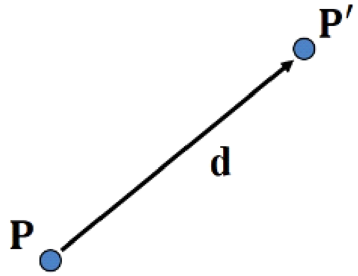
	Transformations shift vertices

Notation

We will be working with both coordinate-free representations of transformations and representations within a particular frame

Symbols	Meaning
$\mathbf{P}, \mathbf{Q}, \mathbf{R}$	Points in an affine space
$\mathbf{p}, \mathbf{q}, \mathbf{r}$	Representations of points Array of 4 scalars in homogeneous coordinates
$\mathbf{u}, \mathbf{v}, \mathbf{w}$	Representations of vectors in an affine space Array of 4 scalars in homogeneous coordinates
α, β, γ	Scalars

Translation is *moving/translating/displacing* a point to a new location



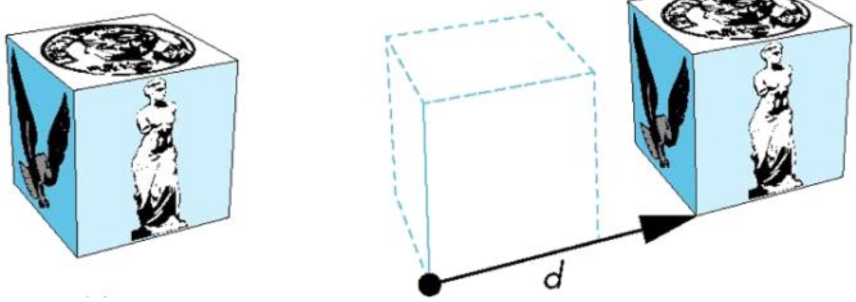
Displacement is determined by a vector d (3 degrees of freedom, d can move P in x,y or z)

$$\mathbf{P}' = \mathbf{P} + \mathbf{d}$$

Options

When we translate many points of a rigid object, there is usually only one way it can occur.

- Each point is displaced by the same vector.
- An animation might involve step-by-step, piecewise transformation



Translation Using Representation

Using the homogeneous coordinate representation in some frame:

-
  - $\mathbf{p} = [x \quad y \quad z \quad 1]^{\mathrm{T}}$
  - $\mathbf{p}' = [x' \quad y' \quad z' \quad 1]^{\mathrm{T}}$
  - $\mathbf{d} = [d_x \quad d_y \quad d_z \quad 0]^{\mathrm{T}}$
- This expression is in four dimensions and expresses point = vector + point

Hence $\mathbf{p}' = \mathbf{p} + \mathbf{d}$ or ▪ $x' = x + d_x$ ▪ $y' = y + d_y$ ▪ $z' = z + d_z$	Point = Point + Vector dx, dy, dz = x,y,z components of vector

Translation Matrix

We can also express translation using a 4 x 4 matrix T in homogeneous coordinates

$\mathbf{p}' = \mathbf{Tp}$ where

$$\mathbf{T} = \mathbf{T}(d_x, d_y, d_z) = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(x, y, z)

Rotation part = Identity matrix

Displacement vector values

Fixed to zeroes

Fixed to one

This form is better for implementation because all affine transformations can be expressed this way and multiple transformations can be concatenated together

No matter what we do, we use a 4x4 matrix

Matrix Representation