Learning Objectives

- Students completing this lecture will be able to
  - Explain the roles of scalars, vectors and points in defining geometry
  - Sketch vectors based on the head-to-tail axiom
  - Describe the following terms: affine space, parametric form, linearly (in)dependent, dimension, basis, coordinate system
  - Differentiate between affine and convex sums, dot and cross products
  - Describe the difference between points and vectors in the homogeneous coordinate representation

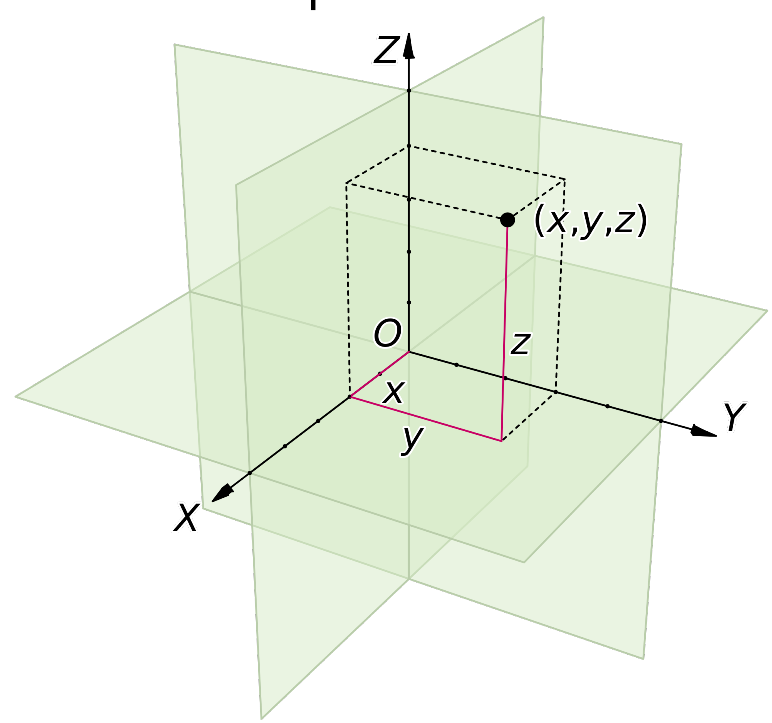
Scalars, Points and Vectors

Basic Elements

- Geometry is the study of the relationships among objects in an n-dimensional space
  - In computer graphics, we are interested in objects that exist in three dimensions
- Want a minimum set of primitives from which we can build more sophisticated objects
- We will need three basic elements
  - Scalars
  - Vectors
  - Points

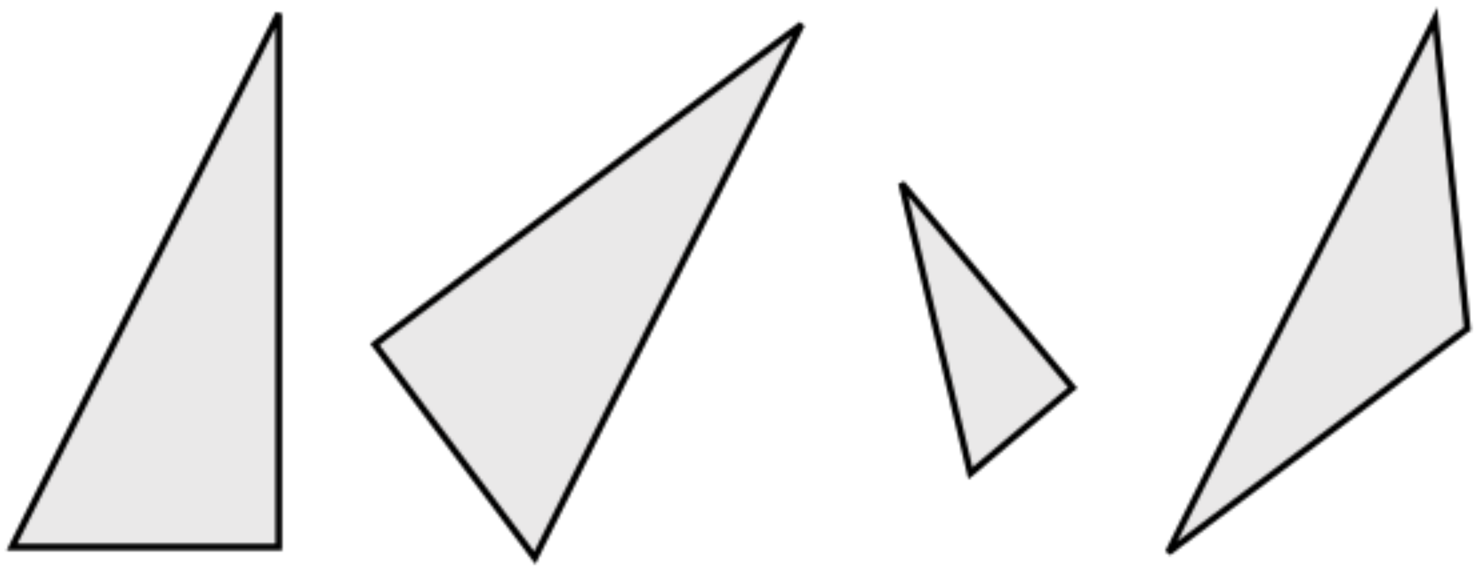
Cartesian Approach

- When we learned simple geometry, most of us started with a Cartesian approach
  - Points were at locations in space $\mathbf{P} = (x, y, z)$
  - We derived results by algebraic manipulations involving these coordinates



Coordinate-Free Geometry

- The Cartesian approach is nonphysical
  - Physically, points exist regardless of the location of an arbitrary coordinate system
  - Most geometric results are independent of the coordinate system

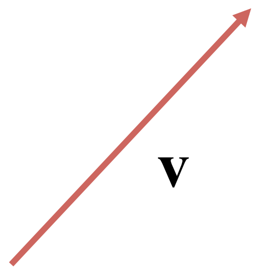


Scalars

- Need three basic elements in geometry
  - Scalars, vectors, points
- Scalars can be defined as members of sets which can be combined by two operations (addition and multiplication) obeying some fundamental axioms (associativity, commutivity, inverses)
- Examples include the real and complex number systems under the ordinary rules
- Scalars alone have no geometric properties

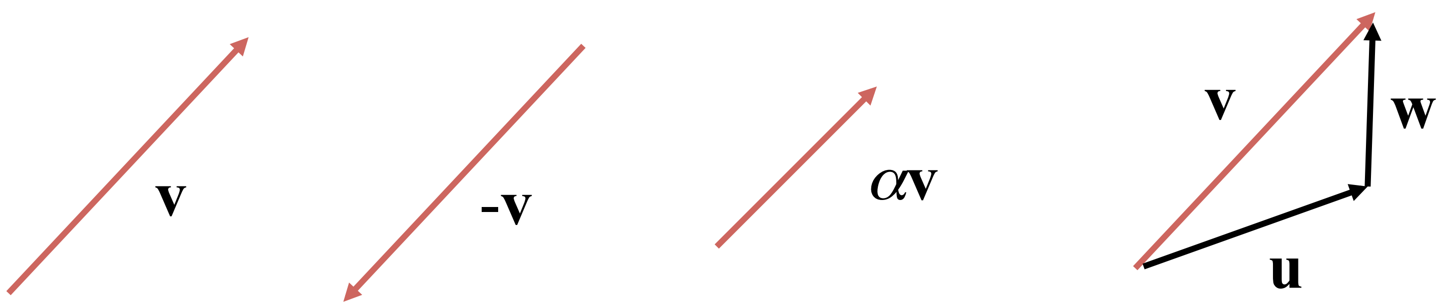
Vectors

- Physical definition: a vector is a quantity with two attributes
  - Direction
  - Magnitude
- Examples include
  - Force, velocity
  - Directed line segments
    - Most important example for graphics
    - Can map to other types



Vector Operations

- Every vector has an inverse
  - Same magnitude but points in opposite direction
- Every vector can be multiplied by a scalar
- There is a zero vector
  - Zero magnitude, undefined orientation
- The sum of any two vectors is a vector
  - Use head-to-tail axiom



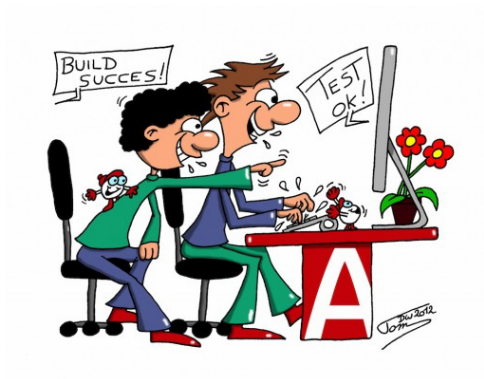
$$\mathbf{v} = \mathbf{u} + \mathbf{w}$$

Linear Vector Spaces

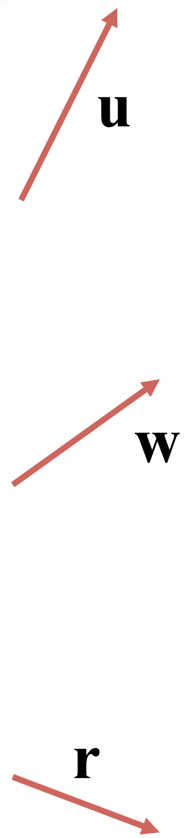
- Mathematical system for manipulating vectors
- Operations
  - Scalar-vector multiplication: $\mathbf{u} = \alpha\mathbf{v}$
  - Vector-vector addition: $\mathbf{w} = \mathbf{u} + \mathbf{v}$
- Expressions such as

  $$\mathbf{v} = \mathbf{u} + 2\mathbf{w} - 3\mathbf{r}$$

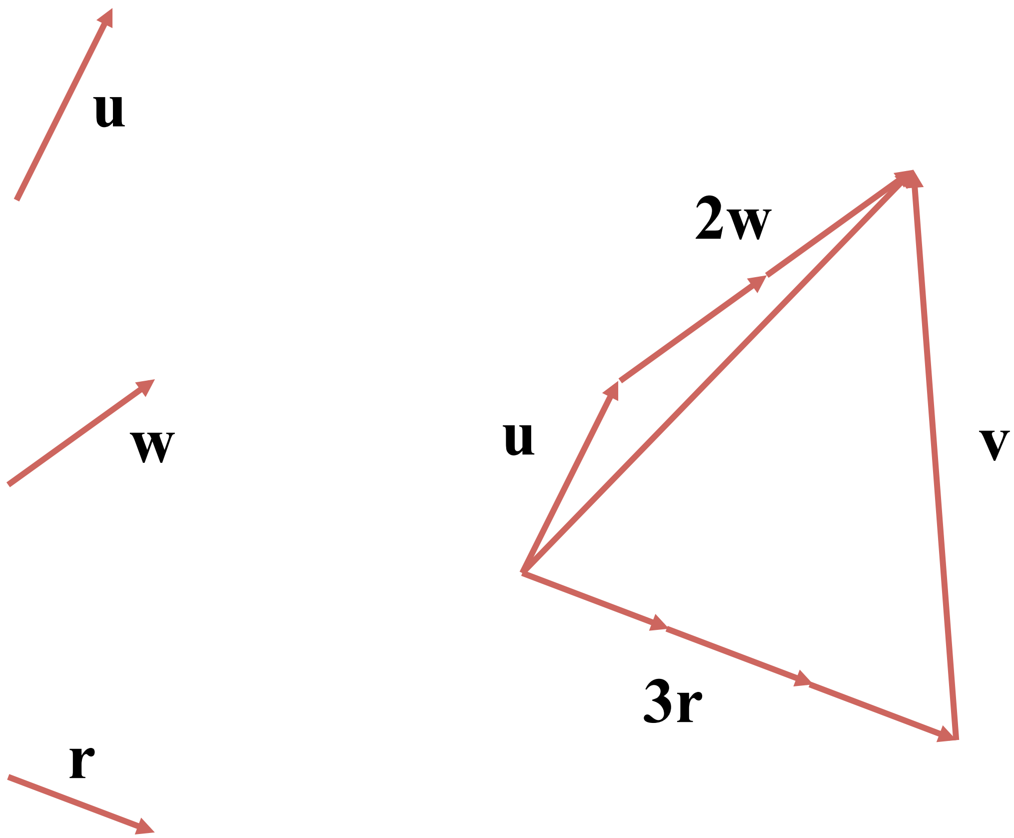
  make sense in a vector space (we can draw it!)



$$\mathbf{v} = \mathbf{u} + 2\mathbf{w} - 3\mathbf{r}$$

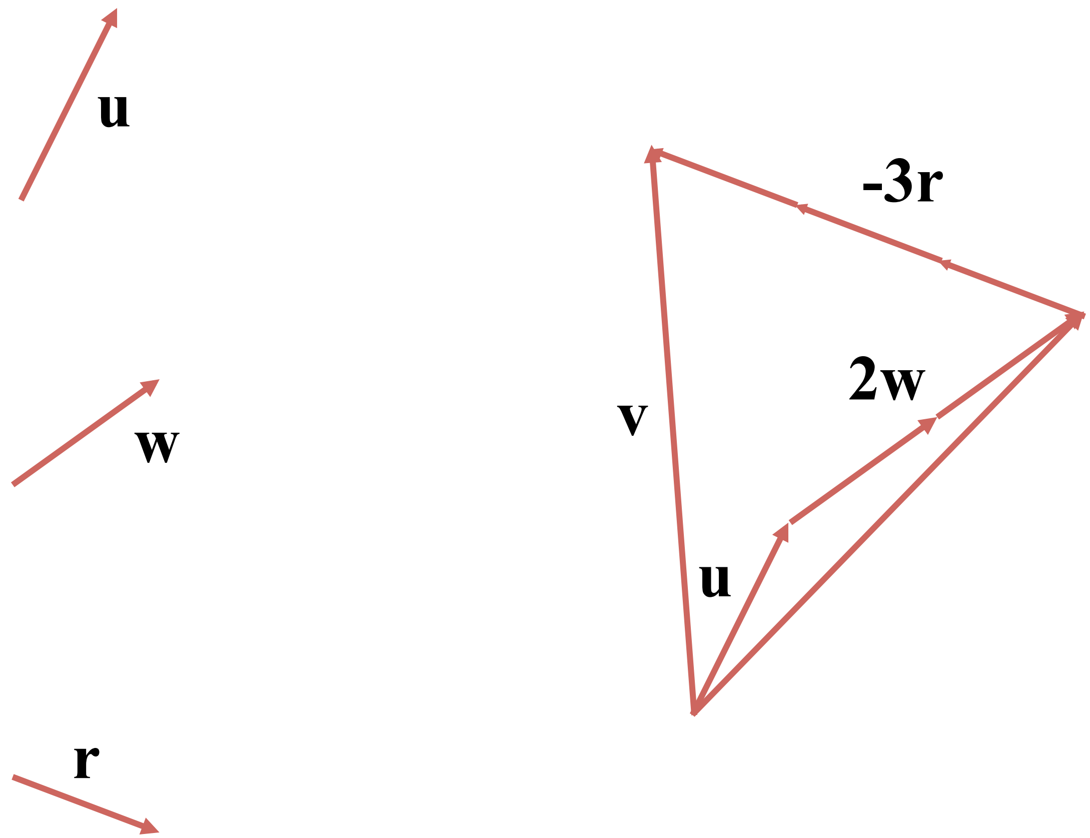


$$\mathbf{v} = \mathbf{u} + 2\mathbf{w} - 3\mathbf{r}$$

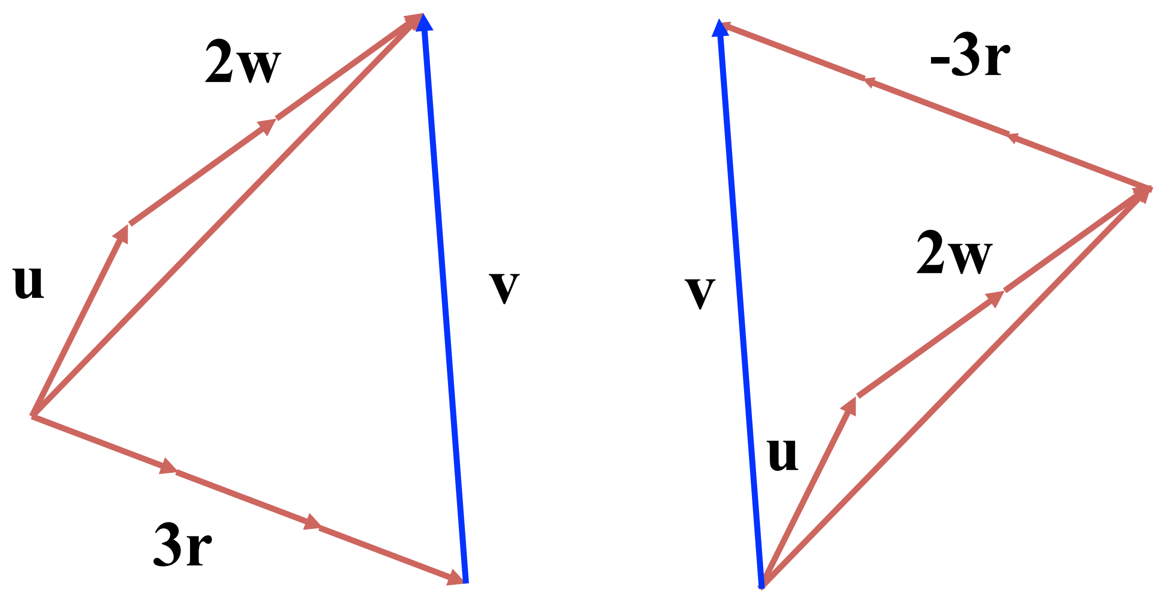


$3\mathbf{r} + \mathbf{v} = \mathbf{u} + 2\mathbf{w}$

$$\mathbf{v} = \mathbf{u} + 2\mathbf{w} + (-3\mathbf{r})$$

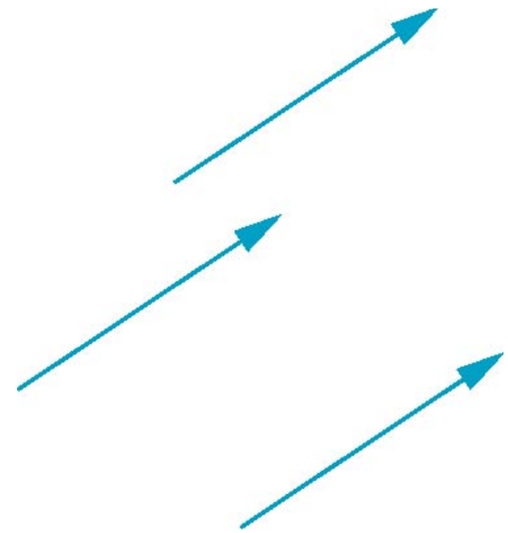


They are the same!



Vectors Lack Position

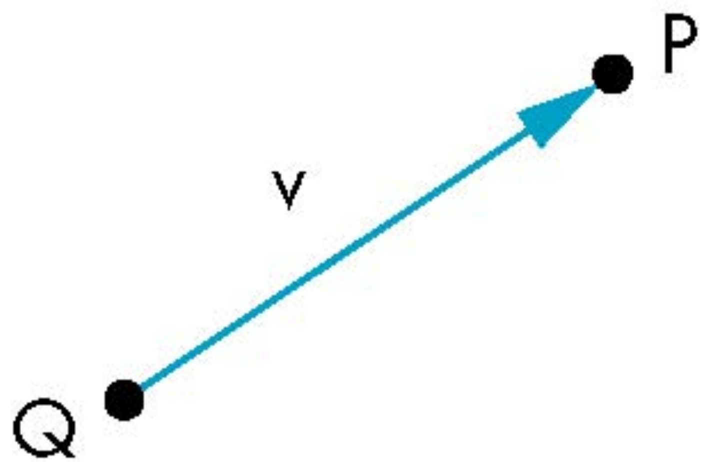
- These vectors are identical
  - Same length and magnitude



- Vectors spaces insufficient for geometry
  - Need points

Points

- Location in space
- Operations allowed between points and vectors
  - Point-point subtraction yields a vector $\mathbf{v} = \mathbf{P} - \mathbf{Q}$
  - Equivalent to point-vector addition $\mathbf{P} = \mathbf{v} + \mathbf{Q}$

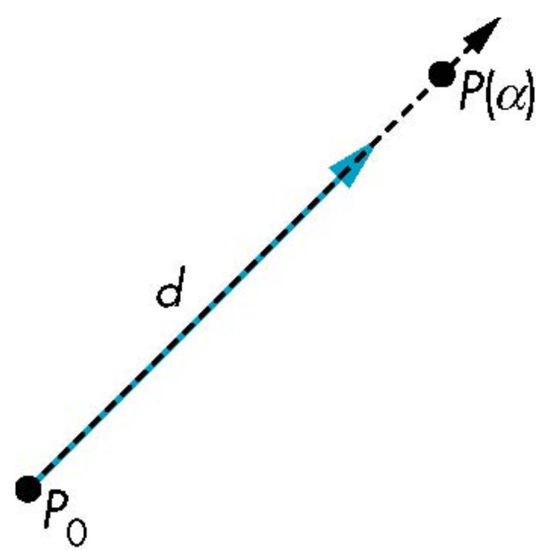


Affine Spaces

- Point + vector space
- Operations
  - Vector-vector addition $\mathbf{u} = \mathbf{v} + \mathbf{w}$
  - Scalar-vector multiplication $\mathbf{u} = \alpha\mathbf{v}$
  - Point-vector addition $\mathbf{P} = \mathbf{v} + \mathbf{Q}$
  - Scalar-scalar operations $\alpha = \beta + \gamma$
  - $\mathbf{P} + 3\mathbf{Q} - \mathbf{v}$ does not make sense! (why?)
- For any point define
  - $1 \bullet \mathbf{P} = \mathbf{P}$
  - $0 \bullet \mathbf{P} = \mathbf{0}$ (zero vector)

Lines

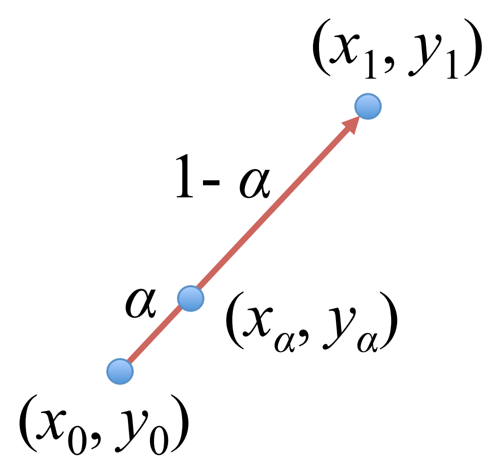
- Consider all points of the form
  - $\mathbf{P}(\alpha) = \mathbf{P}_0 + \alpha\,\mathbf{d}$
  - Set of all points that pass through $\mathbf{P}_0$ in the direction of the vector $\mathbf{d}$



Parametric Form

- This form is known as the parametric form of the line
  - More robust and general than other forms
  - Extends to curves and surfaces
- Two-dimensional forms
  - Explicit: $y = mx + h$
  - Implicit: $ax + by + c = 0$
  - Parametric:

$$x(\alpha) = x_0 + \alpha(x_1 - x_0) = \alpha x_1 + (1 - \alpha)x_0$$
$$y(\alpha) = y_0 + \alpha(y_1 - y_0) = \alpha y_1 + (1 - \alpha)y_0$$

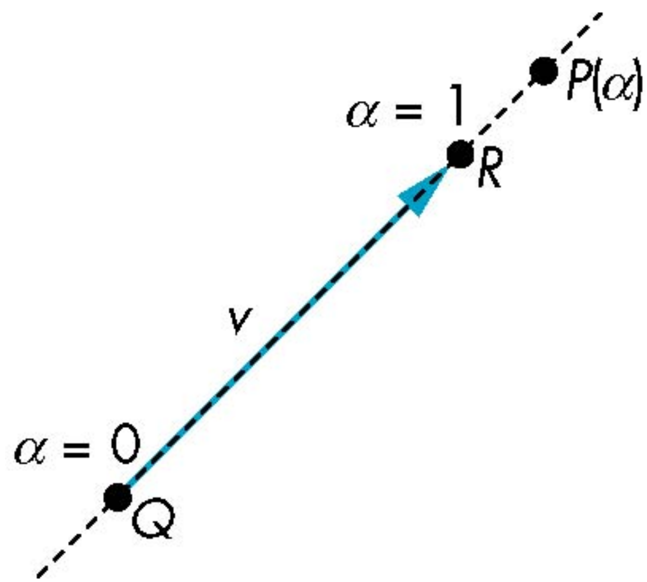


Rays and Line Segments

- If $\alpha \geq 0$, then $\mathbf{P}(\alpha)$ is the ray leaving $\mathbf{P}_0$ in the direction $\mathbf{d}$
- If we use two points to define $\mathbf{v}$, then

$$\begin{aligned}\mathbf{P}(\alpha) &= \mathbf{Q} + \alpha\,(\mathbf{R} - \mathbf{Q}) = \mathbf{Q} + \alpha\mathbf{v} \\ &= \alpha\mathbf{R} + (1 - \alpha)\mathbf{Q}\end{aligned}$$

- For $0 \leq \alpha \leq 1$ we get all the points on the line segment joining $\mathbf{R}$ and $\mathbf{Q}$



Affine Sums

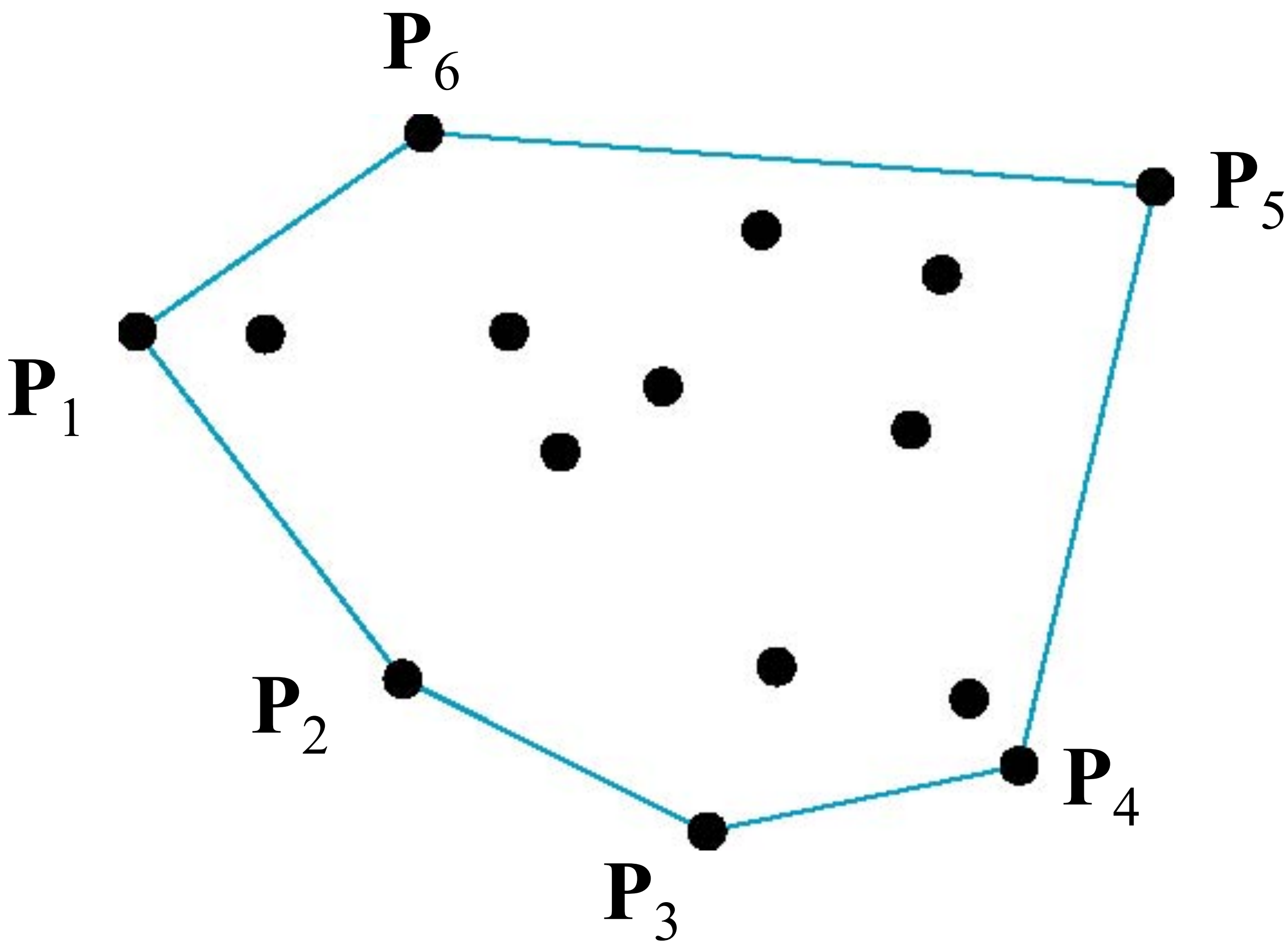
- Consider the “sum”

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

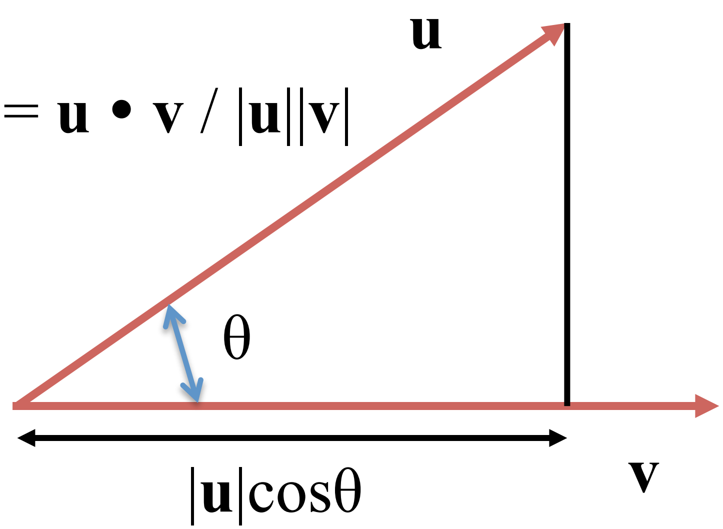
- Can show by induction that this sum makes sense iff $\alpha_1 + \alpha_2 + \ldots + \alpha_n = 1$, in which case we have the affine sum of the points $\mathbf{P}_1$, $\mathbf{P}_2$, $\ldots$, $\mathbf{P}_n$
- If, in addition, $\alpha_i \geq 0$, we have the convex sum (i.e., convex hull) of $\mathbf{P}_1$, $\mathbf{P}_2$, $\ldots$, $\mathbf{P}_n$

Convex Hull

- Smallest convex object containing $\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_n$
- Formed by "shrink wrapping" points



Dot Product

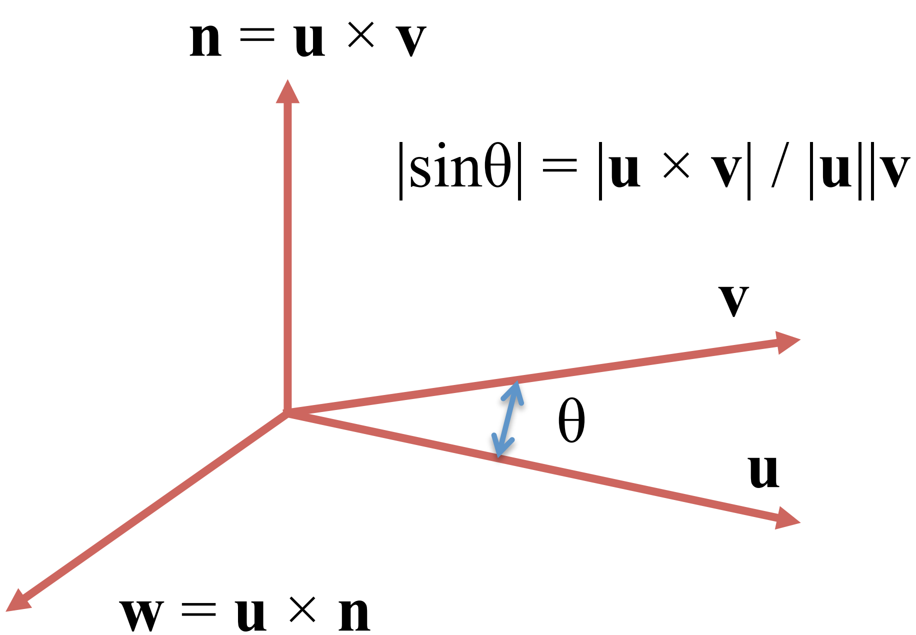


$\cos\theta = \mathbf{u} \bullet \mathbf{v} \,/\, |\mathbf{u}||\mathbf{v}|$

$|\mathbf{u}|^2 = \mathbf{u} \bullet \mathbf{u}$

If $\mathbf{u} \bullet \mathbf{v} = 0$, then $\mathbf{u}$ and $\mathbf{v}$ are orthogonal

Cross Product



$$\mathbf{u} = (u_1, u_2, u_3)$$
$$\mathbf{v} = (v_1, v_2, v_3)$$
$$\mathbf{u} \times \mathbf{v} = (u_2 v_3 - u_3 v_2,\; u_3 v_1 - u_1 v_3,\; u_1 v_2 - u_2 v_1)$$

Mutually orthogonal vectors in 3D $(\mathbf{u}, \mathbf{n}, \mathbf{w})$ right-hand rule

$\mathbf{n} = \mathbf{u} \times \mathbf{v}$

$\mathbf{w} = \mathbf{u} \times \mathbf{n}$

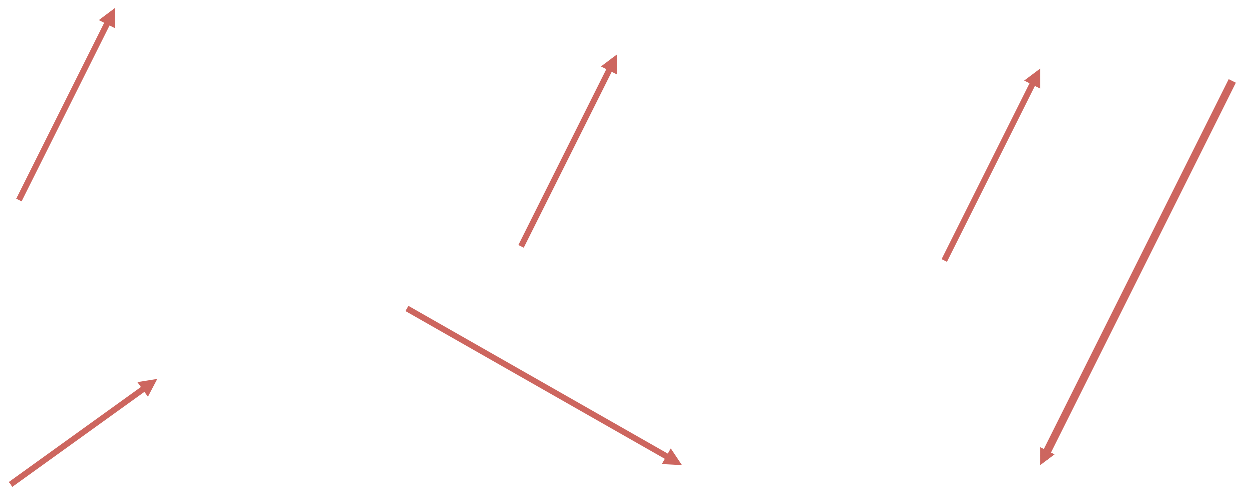
Coordinate Systems and Frames

Linear Independence

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

  $\alpha_1\mathbf{v}_1 + \alpha_2\mathbf{v}_2 + \ldots + \alpha_n\mathbf{v}_n = \mathbf{0}$ iff $\alpha_1 = \alpha_2 = \ldots = 0$
- If a set of vectors is linearly independent, we cannot represent one in terms of the others
- If a set of vectors is linearly dependent, as least one can be written in terms of the others

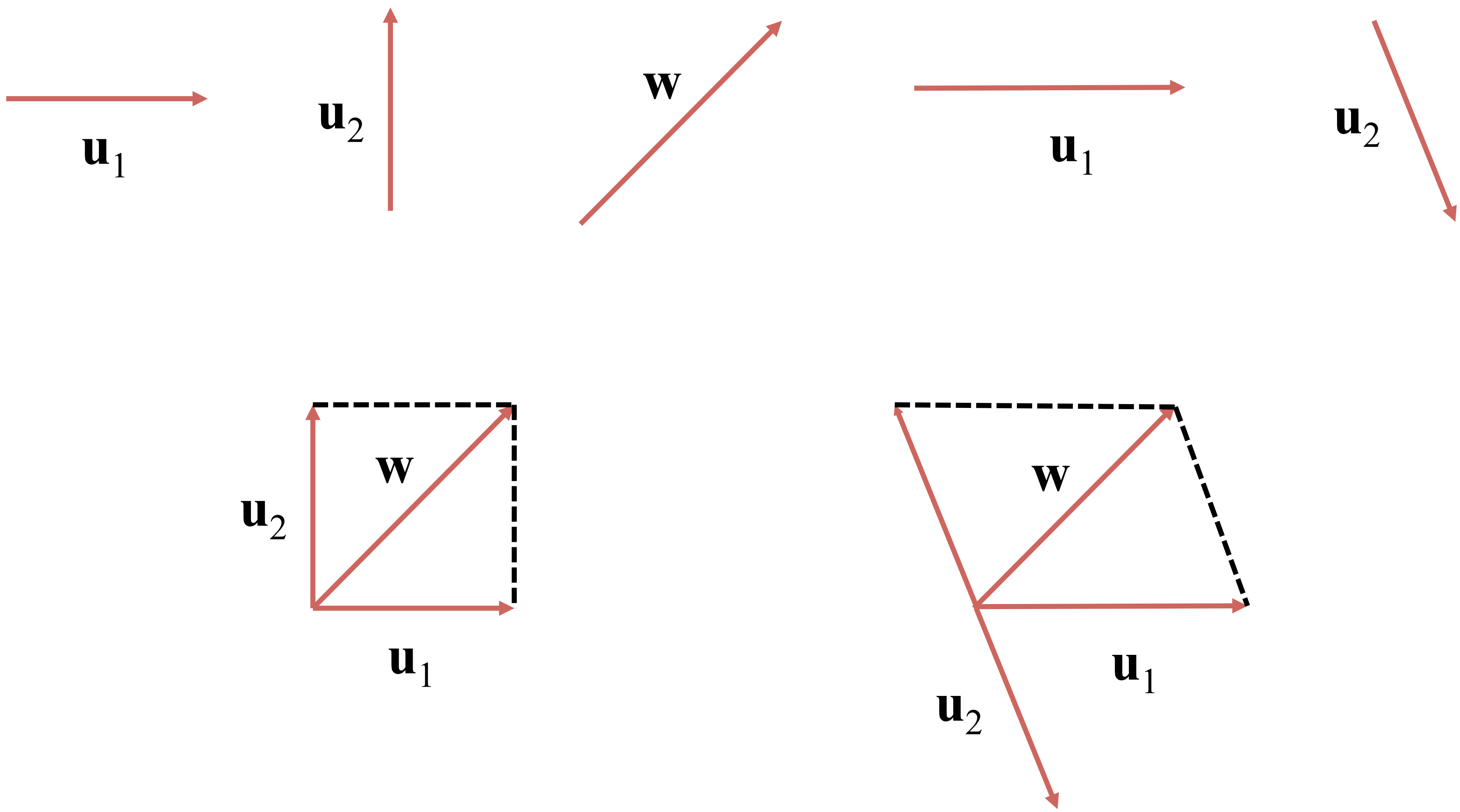
Are they linearly independent?



Dimension

- In a vector space, the maximum number of linearly independent vectors is fixed and is called the dimension of the space
- In an n-dimensional space, any set of n linearly independent vectors form a basis for the space
- Given a basis $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_\mathrm{n}$, any vector $\mathbf{v}$ can be written as $\mathbf{v} = \alpha_1\mathbf{v}_1 + \alpha_2\mathbf{v}_2 + \ldots + \alpha_n\mathbf{v}_\mathrm{n}$, where the $\{\alpha_i\}$ are unique

Linear independence ≠ orthogonal



Representation

- Until now we have been able to work with geometric entities without using any frame of reference, such as a coordinate system
- A frame of reference is needed to relate points and objects to our physical world
  - For example, where is a point? Can't answer without a reference system
  - World coordinates
  - Camera coordinates

Coordinate Systems

- Consider a basis $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_\mathrm{n}$
- A vector is written $\mathbf{v} = \alpha_1\mathbf{v}_1 + \alpha_2\mathbf{v}_2 + \ldots + \alpha_n\mathbf{v}_n$
- The list of scalars $\{\alpha_1, \alpha_2, \ldots, \alpha_n\}$ is the representation of $\mathbf{v}$ with respect to the given basis
- We can write the representation as a row or column array of scalars

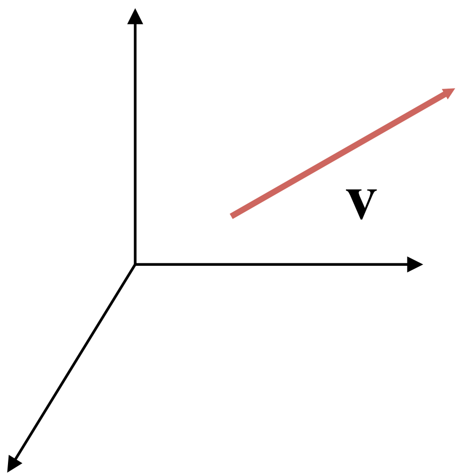
$$\mathbf{a} = [\alpha_1\ \alpha_2\ \ldots\ \alpha_\mathrm{n}]^\mathrm{T} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \cdot \\ \alpha_\mathrm{n} \end{bmatrix}$$

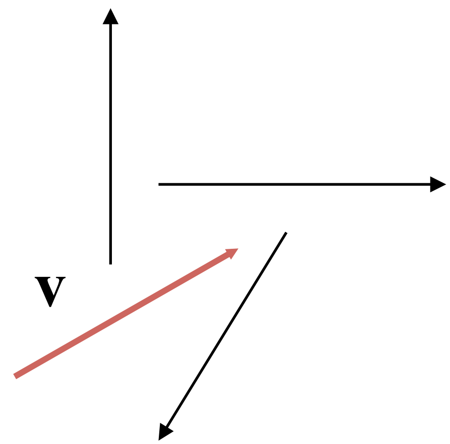
Example

- $\mathbf{v} = 2\mathbf{v}_1 + 3\mathbf{v}_2 - 4\mathbf{v}_3 \Rightarrow \mathbf{a} = [2\ 3\ {-4}]^T$
- Note that this representation is with respect to a particular basis
- For example, in WebGL we start by representing vectors using the object basis (i.e., object frame) but later the system needs a representation in terms of the world basis (i.e., world frame), followed by camera or eye basis (i.e., eye frame)

Coordinate Systems

- Which is correct?

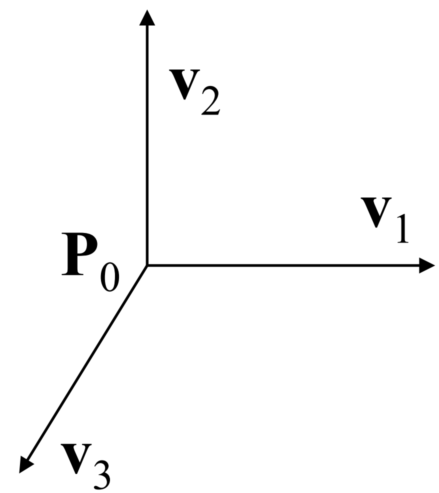




- Both are because vectors have no fixed location

Frames

- A coordinate system is insufficient to represent points
- If we work in an affine space we can add a single point, the origin, to the basis vectors to form a frame



Representation in a Frame

- Frame determined by $(\mathbf{P}_0, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$
- Within this frame, every vector can be written as

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

- Every point can be written as

$$\mathbf{P} = \mathbf{P}_0 + \beta_1\mathbf{v}_1 + \beta_2\mathbf{v}_2 + \ldots + \beta_n\mathbf{v}_n$$

Confusing Points and Vectors

Consider the point and the vector

$$\mathbf{P} = \mathbf{P}_0 + \beta_1\mathbf{v}_1 + \beta_2\mathbf{v}_2 + \ldots + \beta_n\mathbf{v}_n$$

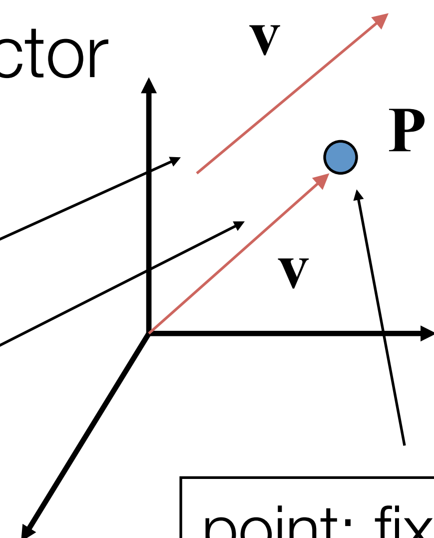
$$\mathbf{v} = \alpha_1\mathbf{v}_1 + \alpha_2\mathbf{v}_2 + \ldots + \alpha_n\mathbf{v}_n$$

They appear to have the similar representations

$$\mathbf{P} = [\beta_1\ \beta_2\ \beta_3] \qquad\qquad \mathbf{v} = [\alpha_1\ \alpha_2\ \alpha_3]$$

which confuses the point with the vector

A vector has no position



Vector can be placed anywhere

point: fixed

A Single Representation

- Recall that we define $0 \bullet \mathbf{P} = \mathbf{0}$ and $1 \bullet \mathbf{P} = \mathbf{P}$ then we can write:

$$\mathbf{v} = \alpha_1 \mathbf{v}_1 + \alpha_2 \mathbf{v}_2 + \alpha_3 \mathbf{v}_3 = [\alpha_1\ \alpha_2\ \alpha_3\ 0]\ [\mathbf{v}_1\ \mathbf{v}_2\ \mathbf{v}_3\ \mathbf{P}_0]^T$$

$$\mathbf{P} = \mathbf{P}_0 + \beta_1 \mathbf{v}_1 + \beta_2 \mathbf{v}_2 + \beta_3 \mathbf{v}_3 = [\beta_1\ \beta_2\ \beta_3\ 1]\ [\mathbf{v}_1\ \mathbf{v}_2\ \mathbf{v}_3\ \mathbf{P}_0]^T$$

- Thus we obtain the four-dimensional homogeneous coordinate representation

$$\mathbf{v} = [\alpha_1\ \alpha_2\ \alpha_3\ 0]^T$$

$$\mathbf{P} = [\beta_1\ \beta_2\ \beta_3\ 1]^T$$

Homogeneous Coordinates

- The homogeneous coordinates form for a three dimensional point $[x\ y\ z]$ is given as

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

- We return to a three dimensional point (for $w \neq 0$) by

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

- If $w = 0$, the representation is that of a vector
- If $w = 1$, the representation is that of a point

Homogeneous Coordinates and Computer Graphics

- Homogeneous coordinates are key to all computer graphics systems
  - All standard transformations (rotation, translation, scaling) can be implemented with matrix multiplications using 4×4 matrices
  - Hardware pipeline works with four-dimensional representations