COMP3271 Computer Graphics

Transformation

2019-20

Objectives

Introduce the three fundamental transformations

- Translation
- Scaling
- Rotation

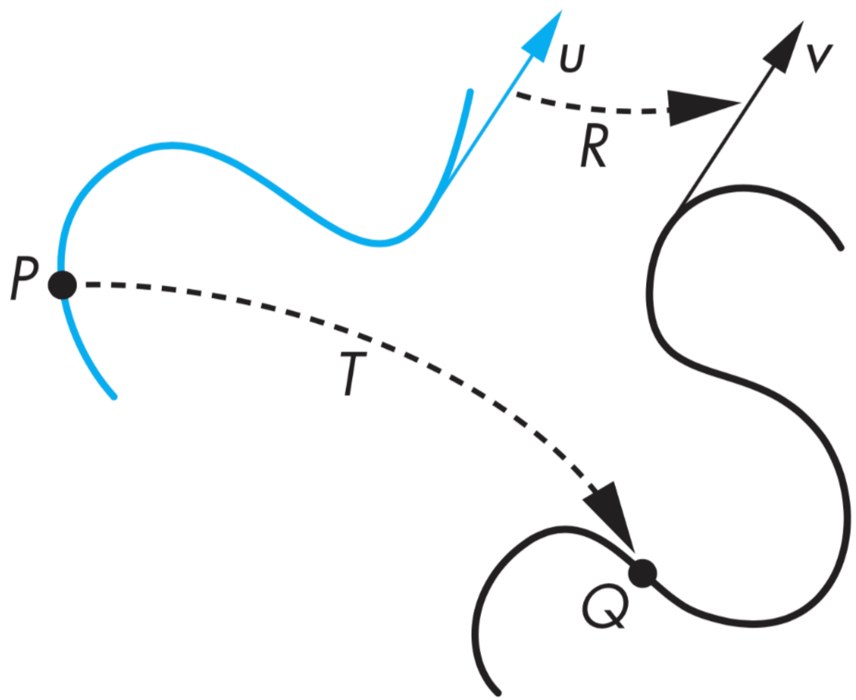
Derive homogeneous coordinate transformation matrices

Build arbitrary transformation from simple transformations

Materials adapted from textbook ppt and previous runs of the course

General Transformations

A transformation maps points to other points and/or vectors to other vectors



$v=R(u)$

$Q=T(P)$

Provides a mechanism to manipulate objects

Why Do We Need Transformations?

Makes modeling more convenient

- for example, often easier to generate models around origin
  - gluSphere() draws a sphere of radius r about the origin
- then move them to final position with transformations

Model viewing process via transformations

- projecting 3-D to 2-D will be done this way

Animation

- transformations as a function of time creates motion

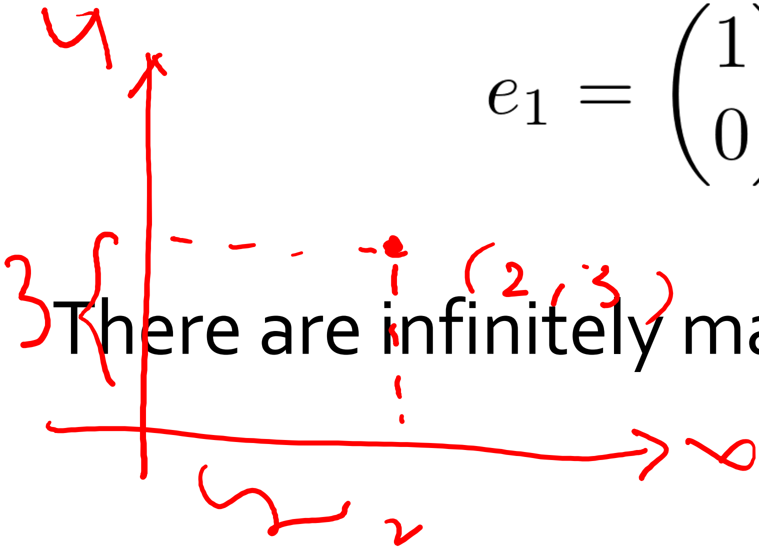
A demo: https://processing.org/examples/tree.html

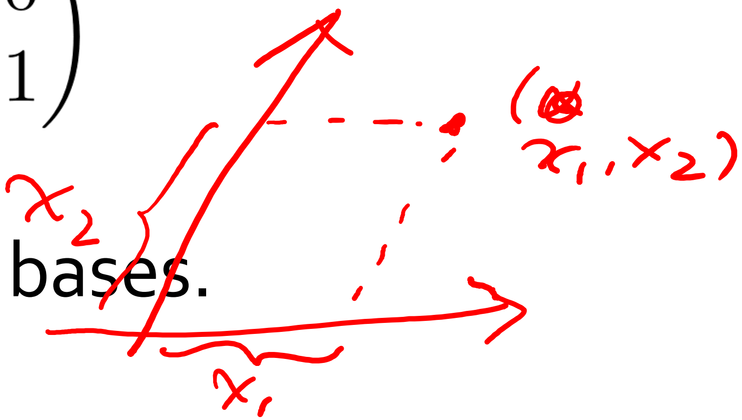
Linear Algebra (very quick review)

A linear combination of two vectors v and w is given by α v + β w, where α and are β scalars.

A **basis** for a space is a linearly independent set of vectors whose linear combinations include all vectors in the space, e.g., standard basis for 2-D plane:

$$e_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad e_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$



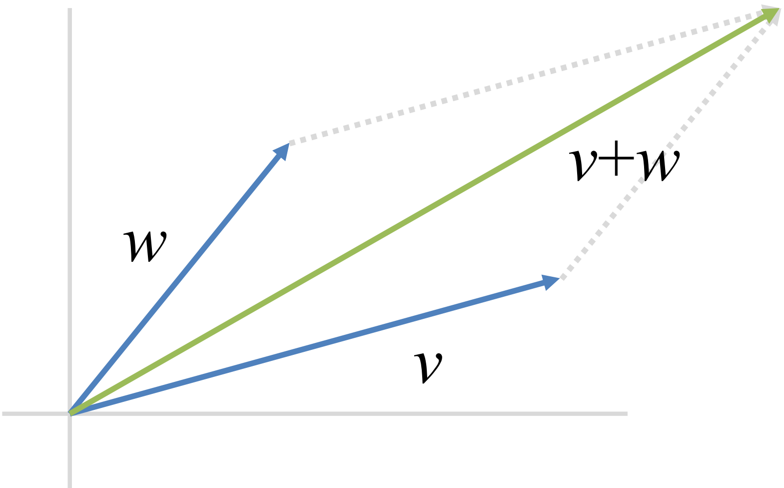


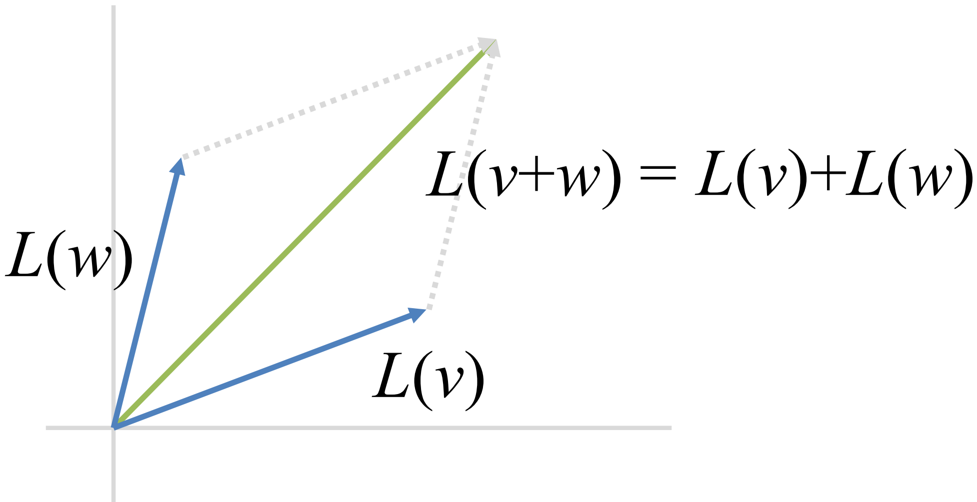
There are infinitely many possible bases.

Linear Transformation

A transformation (or mapping) L is linear when given any two vectors v and $w \in \mathrm{R}^n$,

- $L(v + w) = L(v) + L(w)$
- $L(kv) = k\,L(v)$ for some scalar k





Linear Transformation

Considering the Cartesian coordinates, where a vector $v = (x, y)^T$ is represented as a linear combination of the base vectors $e^1 = (1, 0)^T$ and $e^2 = (0, 1)^T$:

$$v = \begin{pmatrix} x \\ y \end{pmatrix} = x \begin{pmatrix} 1 \\ 0 \end{pmatrix} + y \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Applying a linear transformation to v:

$$L(\begin{pmatrix} x \\ y \end{pmatrix}) = L(x \begin{pmatrix} 1 \\ 0 \end{pmatrix} + y \begin{pmatrix} 0 \\ 1 \end{pmatrix}) = xL(\begin{pmatrix} 1 \\ 0 \end{pmatrix}) + yL(\begin{pmatrix} 0 \\ 1 \end{pmatrix})$$

Transformation of the base vectors

Linear Transformation

Linear transformations can be represented as **matrices**.

$$L\left(\begin{pmatrix} x \\ y \end{pmatrix}\right) = L\left(x\begin{pmatrix} 1 \\ 0 \end{pmatrix} + y\begin{pmatrix} 0 \\ 1 \end{pmatrix}\right) = xL\left(\begin{pmatrix} 1 \\ 0 \end{pmatrix}\right) + yL\left(\begin{pmatrix} 0 \\ 1 \end{pmatrix}\right)$$

$$= \underbrace{\left[L\left(\begin{pmatrix} 1 \\ 0 \end{pmatrix}\right) L\left(\begin{pmatrix} 0 \\ 1 \end{pmatrix}\right)\right]}_{\text{A 2x2 matrix}} \begin{pmatrix} x \\ y \end{pmatrix}$$

Affine Transformation

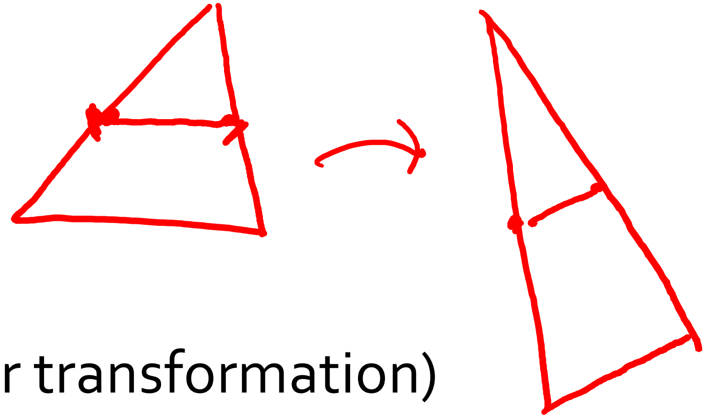
Affine transformation takes a more general form of

$$A(v) = Lv + b$$

where matrix L represents a non-singular linear transformation (i.e., $\det(L) \neq 0$) and b is a vector.

It can be viewed as a linear transformation plus a translation

Affine Transformations



Preserve geometric properties such as:

- Collinearity (lines remain lines under transformation)
- Parallelism
- Ratios of distances (e.g., mid-points remain mid-points)

Characteristic of many physically important transformations

- Rigid body transformations: rotation, translation
- Scaling, shear

Importance in graphics: we need only transform endpoints of line segments and let implementation draw line segment between the transformed endpoints