Affine Transformations

- Line preserving.
- **Transformation**: function that maps a point or vector to another point or vector.
- **Linear function**: *f()* is a linear function iff for any scalars α and β and any vertices *p* and *q*:

$$f(\alpha p+\beta q)=\alpha f(p)+\beta f(q)$$

  - Transforms of linear combinations of vertices obtained from linear combinations of transformed vertices.
- For our 4-tuple representations or points and vertices, we write $v=Au$.

Linear Transformations

- A is a way or writing the linear transformation
- If A is nonsingular each linear transformation corresponds to a change of frame.
- Two possible interpretations:
  - Change of frame
  - Transformation of vertices in the same frame.

Graphics Transformations

- Transformations move points describing geometric objects to new positions.
- **Rigid Body Transformations**:
  - **Translation:** displaces points a fixed distance in a particular direction.
  - **Rotation:** circular motion of a configuration about a point or line.
  - Cannot alter shape of an object, only position and orientation.
- **Scaling:** makes objects bigger or smaller, all (uniform) or some (non-uniform) directions.

Translation Matrix

We can also express translation using a 4 x 4 matrix **T** in homogeneous coordinates **p**'=**Tp** where

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

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

Rotation Matrix

$$\mathbf{R} = \mathbf{R}_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotation about x and y axes

- Same argument as for rotation about *z* axis
  - For rotation about *x* axis, *x* is unchanged
  - For rotation about *y* axis, *y* is unchanged

$$\mathbf{R} = \mathbf{R}_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

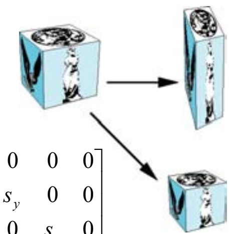
$$\mathbf{R} = \mathbf{R}_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Scaling

Expand or contract along each axis (fixed point of origin)

$x' = s_x x$
$y' = s_y x$
$z' = s_z x$

$\mathbf{p}' = \mathbf{S}\mathbf{p}$

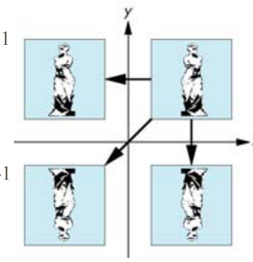


$$\mathbf{S} = \mathbf{S}(s_x, s_y, s_z) = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Reflection

corresponds to negative scale factors

$s_x = -1 \; s_y = 1$ original



$s_x = -1 \; s_y = -1$ $s_x = 1 \; s_y = -1$

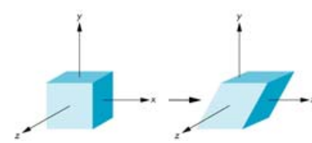
Inverses

- Although we could compute inverse matrices by general formulas, we can use simple geometric observations
  - Translation: $\mathbf{T}^{-1}(d_x, d_y, d_z) = \mathbf{T}(-d_x, -d_y, -d_z)$
  - Rotation: $\mathbf{R}^{-1}(\theta) = \mathbf{R}(-\theta)$
    - Holds for any rotation matrix
    - Note that since $\cos(-\theta) = \cos(\theta)$ and $\sin(-\theta) = -\sin(\theta)$

    $\mathbf{R}^{-1}(\theta) = \mathbf{R}^{T}(\theta)$
  - Scaling: $\mathbf{S}^{-1}(s_x, s_y, s_z) = \mathbf{S}(1/s_x, 1/s_y, 1/s_z)$

Shear

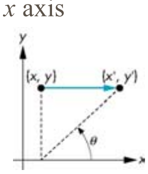
- Helpful to add one more basic transformation
- Equivalent to pulling faces in opposite directions



Shear Matrix

Consider simple shear along *x* axis

$x' = x + y \cot\theta$
$y' = y$
$z' = z$



$$\mathbf{H}(\theta) = \begin{bmatrix} 1 & \cot\theta & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Concatenation

- We can form arbitrary affine transformation matrices by multiplying together rotation, translation, and scaling matrices
- Because the same transformation is applied to many vertices, the cost of forming a matrix $\mathbf{M} = \mathbf{ABCD}$ is not significant compared to the cost of computing $\mathbf{Mp}$ for many vertices $\mathbf{p}$
- The difficult part is how to form a desired transformation from the specifications in the application

Order of Transformations

- Note that matrix on the right is the first applied
- Mathematically, the following are equivalent

  $\mathbf{p}' = \mathbf{ABCp} = \mathbf{A}(\mathbf{B}(\mathbf{Cp}))$

- Note many references use column matrices to present points. In terms of column matrices

  $\mathbf{p}^{T\prime} = \mathbf{p}^T\mathbf{C}^T\mathbf{B}^T\mathbf{A}^T$