Computer Graphics CS248
Transformation Matrices

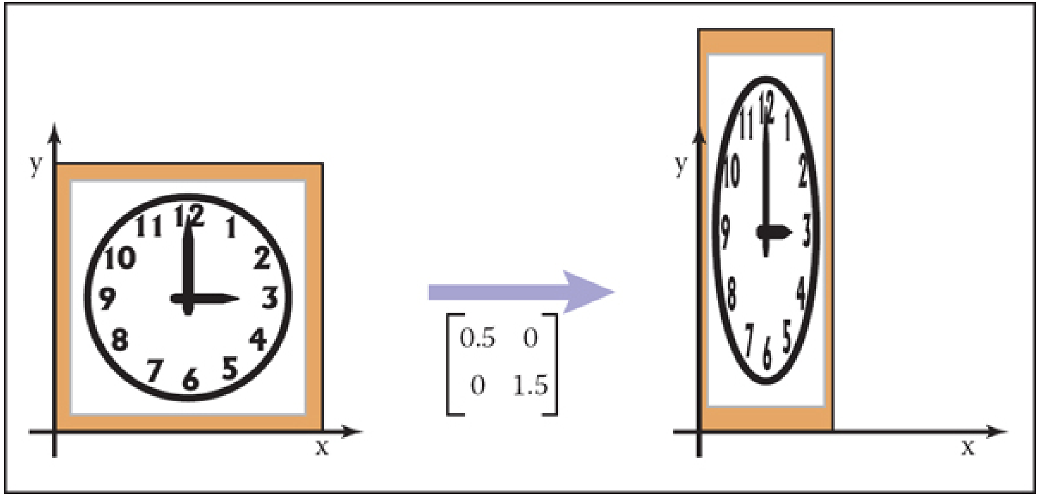
Ivan Viola
NANOVISUALIZATION GROUP

2D Linear Transformations

$$\begin{bmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_{1,1}x + a_{1,2}y \\ a_{2,1}x + a_{2,2}y \end{bmatrix}$$

Scale:

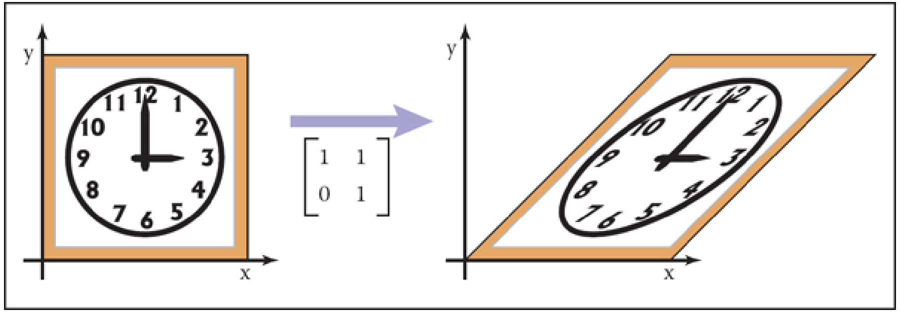
$$\begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} s_x x \\ s_y y \end{bmatrix}$$

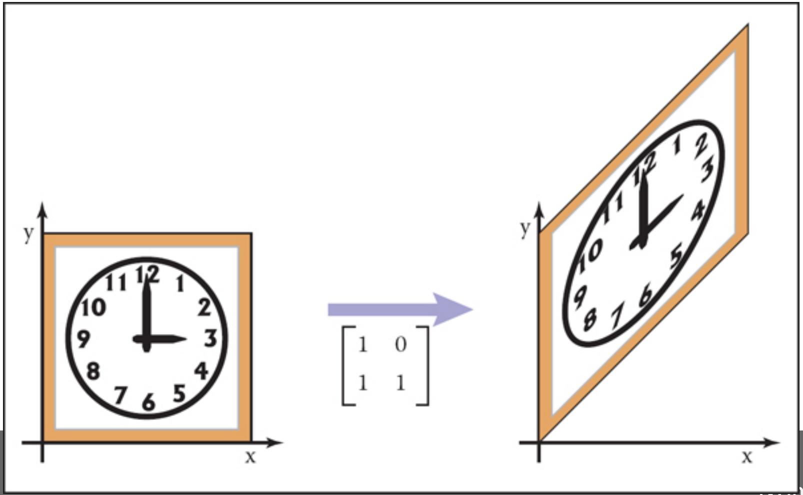


2D Linear Transformations (cont.)

Shearing (horizontal / vertical $s = \tan\varphi$):

$$\begin{bmatrix} 1 & s \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x + sy \\ y \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ s & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x \\ sx + y \end{bmatrix}$$





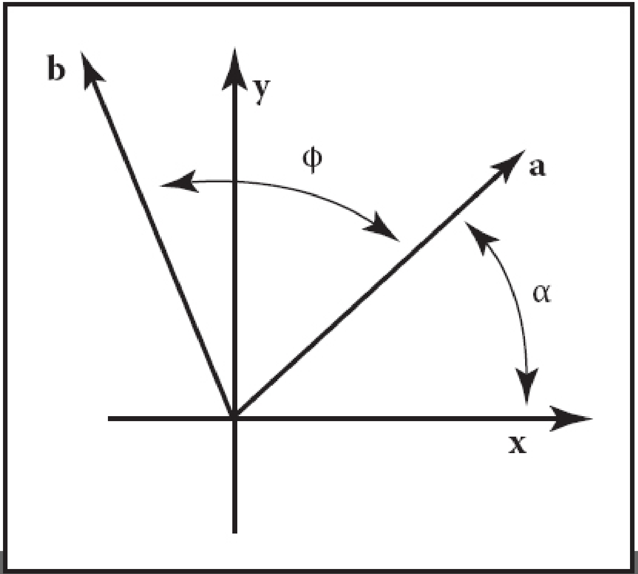
2D Linear Transformations (cont.)

Rotation of vector $\boldsymbol{a}$ into vector $\boldsymbol{b}$:

$$x_a = r\cos\alpha,\ y_a = r\sin\alpha,\ r = \sqrt{x_a^2 + y_a^2}$$

$$x_b = r\cos(\alpha + \varphi)$$
$$= r\cos\alpha\cos\varphi - r\sin\alpha\sin\varphi$$

$$y_b = r\sin(\alpha + \varphi)$$
$$= r\sin\alpha\cos\varphi + r\cos\alpha\sin\varphi$$

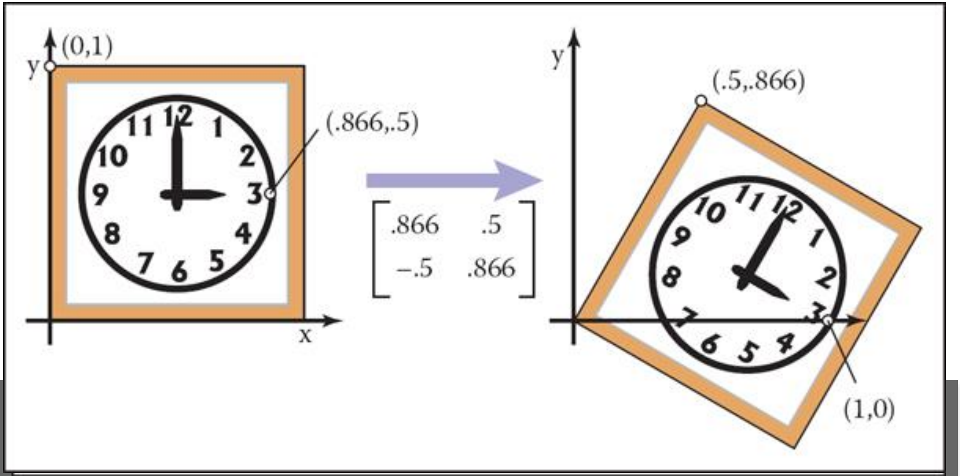


2D Linear Transformations (cont.)

$$x_b = r\cos\alpha\cos\varphi - r\sin\alpha\sin\varphi$$
$$= x_a\cos\varphi - y_a\sin\varphi$$

$$y_b = r\sin\alpha\cos\varphi + r\cos\alpha\sin\varphi$$
$$= y_a\cos\varphi + x_a\sin\varphi$$

$$\begin{bmatrix}\cos\varphi & -\sin\varphi\\ \sin\varphi & \cos\varphi\end{bmatrix}\begin{bmatrix}x\\ y\end{bmatrix}$$

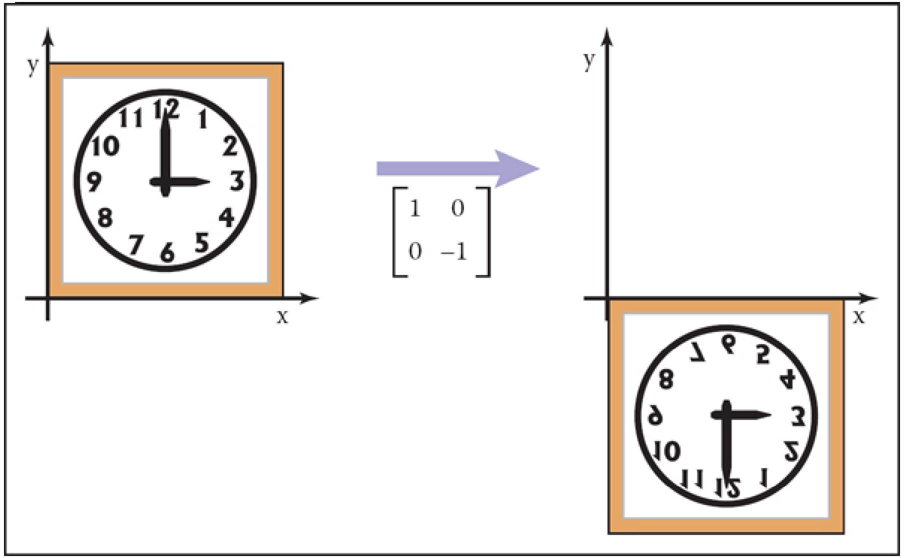


2D Linear Transformations (cont.)

Reflection (across y (or x) coordinate scale):

$$\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

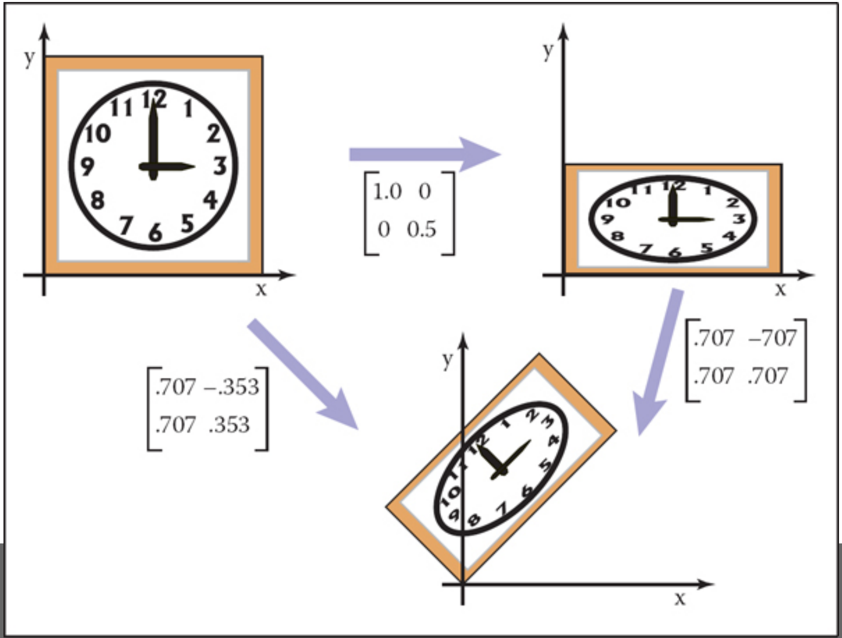


Composition of Transformations

$$v_2 = Sv_1, v_3 = Rv_2$$

$$v_3 = R(Sv_1) = (RS)v_1, M = RS \Rightarrow v_3 = Mv_1$$

Transformations are applied from right to left.



Composition of Transformations (cont.)

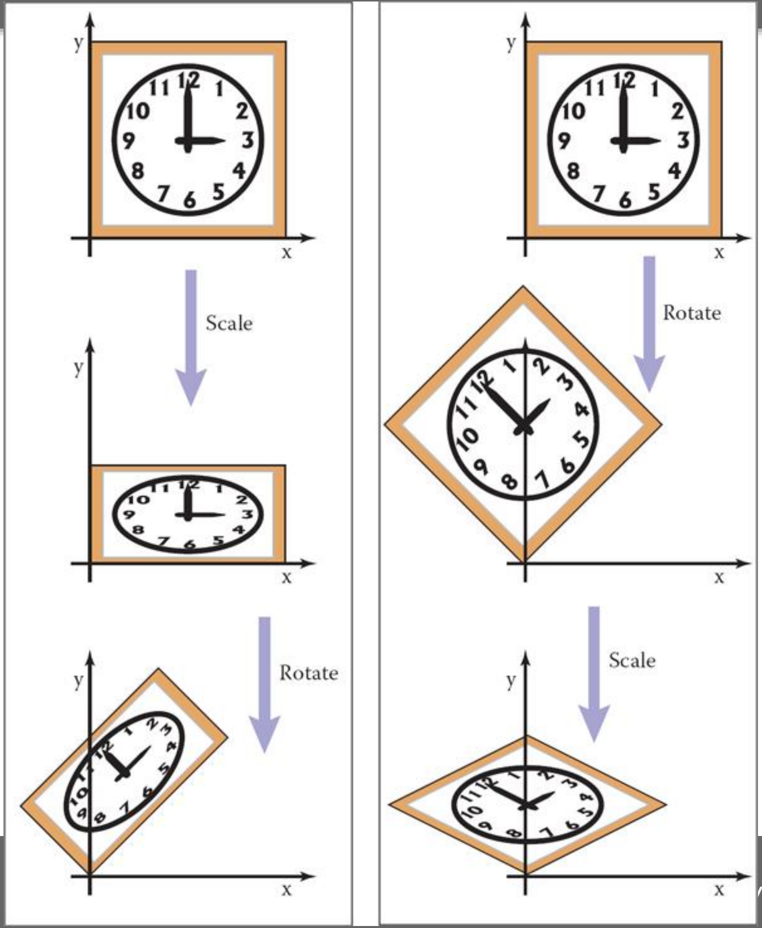
$$\begin{bmatrix} 0.707 & -0.707 \\ 0.707 & 0.707 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0.5 \end{bmatrix}$$

$$\begin{bmatrix} 0.707 & -0.353 \\ 0.707 & 0.353 \end{bmatrix}$$

$$\begin{bmatrix} 0.707 & -0.707 \\ 0.353 & 0.353 \end{bmatrix}$$

rotate$(-\frac{\pi}{4})$ scale$(1,5,1)$ rotate$(\frac{\pi}{4})$

$\boldsymbol{RSR}^{\mathrm{T}}$



Decomposition of Transformations

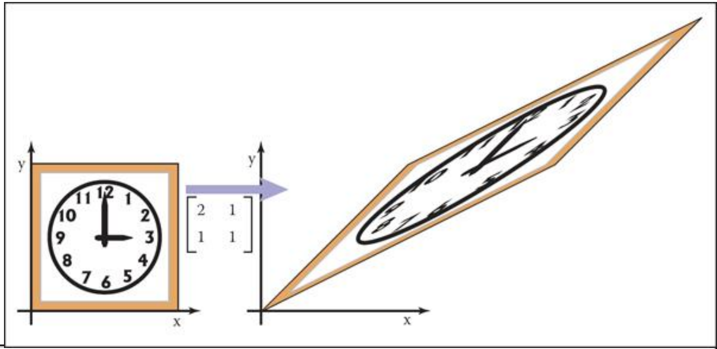
Symmetric matrix $\boldsymbol{A} = \boldsymbol{RSR}^{\mathrm{T}} (= (\boldsymbol{RSR}^{\mathrm{T}})^{\mathrm{T}})$

$\boldsymbol{R}$: orthogonal matrix, $\boldsymbol{S}$: diagonal matrix

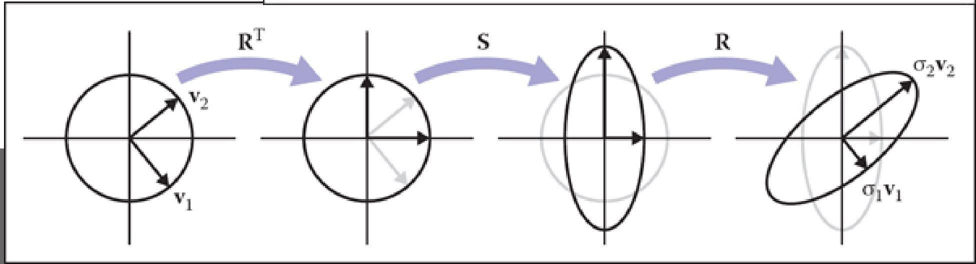
Eigenvalue decomposition, find $\boldsymbol{v}_1, \boldsymbol{v}_2, \lambda_1, \lambda_2$

$$[\boldsymbol{v}_1 \quad \boldsymbol{v}_2] \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} [\boldsymbol{v}_1 \quad \boldsymbol{v}_2]^{\mathrm{T}}$$

$$\begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 0.85 & -0.53 \\ 0.53 & 0.85 \end{bmatrix} \begin{bmatrix} 2.61 & 0 \\ 0 & 0.382 \end{bmatrix} \begin{bmatrix} 0.85 & 0.53 \\ -0.53 & 0.85 \end{bmatrix}$$



Partial undoing possible



Decomposition of Transformations (cont.)

Nonsymmetric case $\boldsymbol{A} = \boldsymbol{USV}^{\mathrm{T}}$

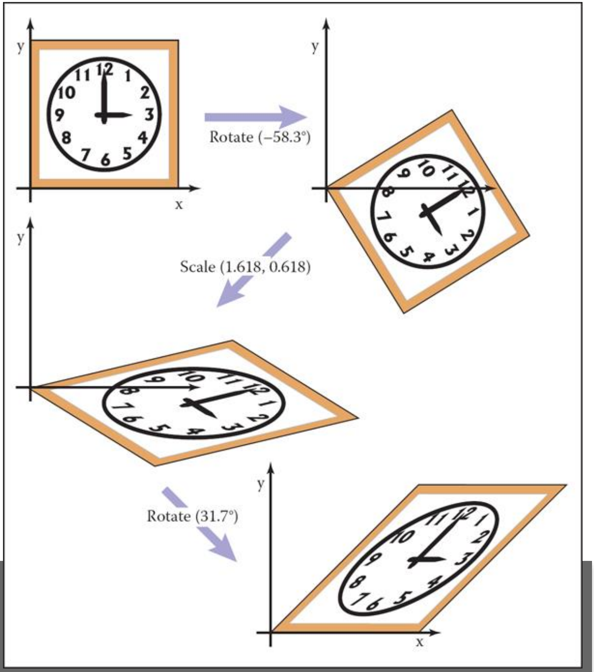
$\boldsymbol{U}, \boldsymbol{V}$: orthogonal matrix, $\boldsymbol{S}$: diagonal matrix

Singular value decomposition

Find $\boldsymbol{u}_1, \boldsymbol{u}_2, s_1, s_2, \boldsymbol{v}_1, \boldsymbol{v}_2$

$$[\boldsymbol{u}_1 \quad \boldsymbol{u}_2]\begin{bmatrix} s_1 & 0 \\ 0 & s_2 \end{bmatrix}[\boldsymbol{v}_1 \quad \boldsymbol{v}_2]^{\mathrm{T}}$$

$$\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.85 & -0.53 \\ 0.53 & 0.85 \end{bmatrix}\begin{bmatrix} 1.62 & 0 \\ 0 & 0.62 \end{bmatrix}\begin{bmatrix} 0.53 & 0.85 \\ -0.85 & 0.53 \end{bmatrix}$$

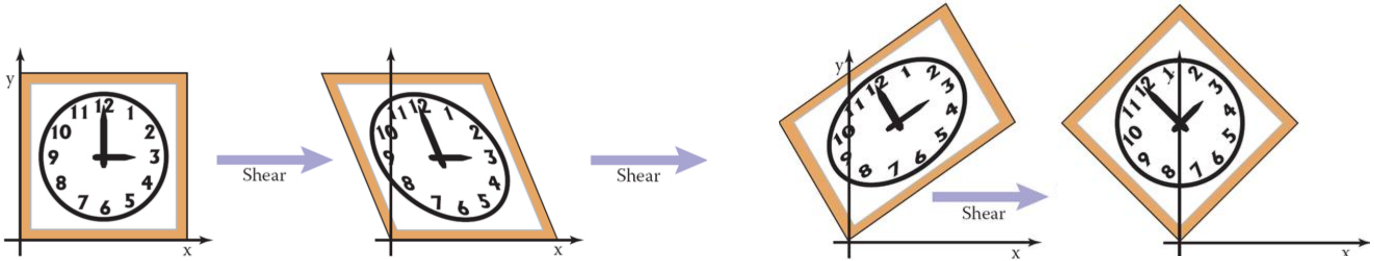


Decomposition of Rotations

2D rotations can be decomposed into three shears.

Beneficial for raster graphics.

$$\begin{bmatrix}\cos\varphi & -\sin\varphi\\ \sin\varphi & \cos\varphi\end{bmatrix} = \begin{bmatrix}1 & \dfrac{\cos\varphi - 1}{\sin\varphi}\\ 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0\\ \sin\varphi & 1\end{bmatrix}\begin{bmatrix}1 & \dfrac{\cos\varphi - 1}{\sin\varphi}\\ 0 & 1\end{bmatrix}$$



$$\begin{bmatrix}1 & s\\ 0 & 1\end{bmatrix}$$

3D Linear Transformations

Scale:

$$\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{bmatrix}$$

Shear:

$$\begin{bmatrix} 1 & d_y & d_z \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Rotate Z:

$$\begin{bmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Rotate Y:

$$\begin{bmatrix} \cos\varphi & 0 & \sin\varphi \\ 0 & 1 & 0 \\ -\sin\varphi & 0 & \cos\varphi \end{bmatrix}$$

Rotate X:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & -\sin\varphi \\ 0 & \sin\varphi & \cos\varphi \end{bmatrix}$$

3D Linear Transformations (cont.)

Arbitrary Rotation:

$$\boldsymbol{R}_{u,v,w} = \begin{bmatrix} x_u & y_u & z_u \\ x_v & y_v & z_v \\ x_w & y_w & z_w \end{bmatrix} \quad \begin{bmatrix} x_u & y_u & z_u \\ x_v & y_v & z_v \\ x_w & y_w & z_w \end{bmatrix} \begin{bmatrix} x_u \\ y_u \\ z_u \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \boldsymbol{x}$$

$$\boldsymbol{u}.\boldsymbol{u} = \boldsymbol{v}.\boldsymbol{v} = \boldsymbol{w}.\boldsymbol{w} = 1 \qquad \boldsymbol{R}_{u,v,w}^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{v}$$

$$\boldsymbol{u}.\boldsymbol{v} = \boldsymbol{v}.\boldsymbol{w} = \boldsymbol{w}.\boldsymbol{u} = 0 \qquad \boldsymbol{R}_{u,v,w}^{\mathrm{T}}\boldsymbol{z} = \boldsymbol{w}$$

3D Linear Transformations (cont.)

Axis-Angle Rotation axis $\boldsymbol{a} = \boldsymbol{w}$:

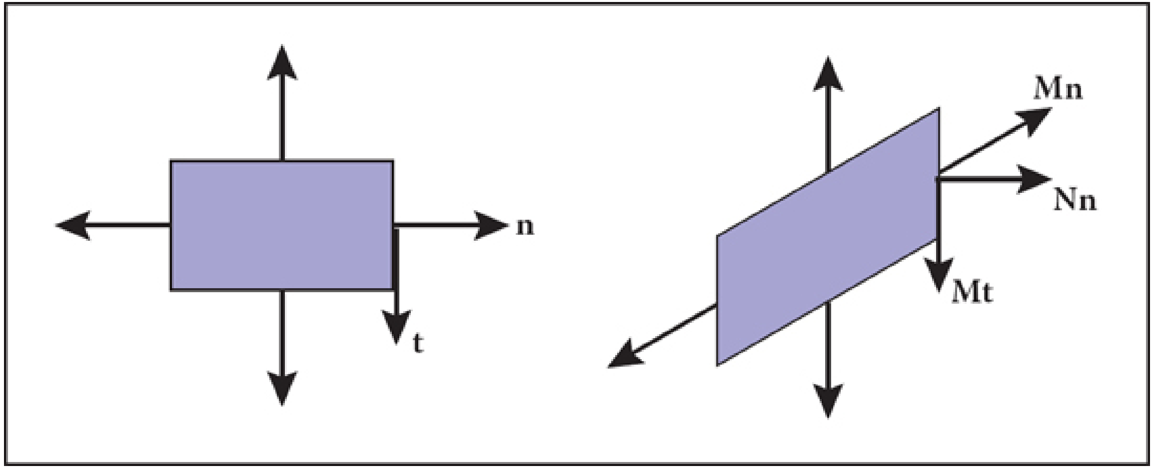
Rotate back to canonical basis, rotate around $\boldsymbol{z}$ and rotate back. If $\boldsymbol{u}$ & $\boldsymbol{v}$ are unknown, construct them.

$$\begin{bmatrix} x_u & x_v & x_w \\ y_u & y_v & y_w \\ z_u & z_v & z_w \end{bmatrix} \begin{bmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_u & y_u & z_u \\ x_v & y_v & z_v \\ x_w & y_w & z_w \end{bmatrix}$$

Transforming Normal Vectors

Tangent vectors remain tangent to the transformed surface.

Normal vectors after a transform of the geometry are not normal to the surface.



Transforming Normal Vectors

$$\boldsymbol{n}_N = \boldsymbol{N}\boldsymbol{n},\ \boldsymbol{t}_M = \boldsymbol{M}\boldsymbol{t}$$

$$\boldsymbol{n}^{\mathrm{T}}\boldsymbol{t} = 0 = \boldsymbol{n}^{\mathrm{T}}\boldsymbol{I}\boldsymbol{t} = \boldsymbol{n}^{\mathrm{T}}(\boldsymbol{M}^{-1}\boldsymbol{M})\boldsymbol{t}$$

$$\boldsymbol{n}_N^{\mathrm{T}}\boldsymbol{t}_M = 0 = (\boldsymbol{n}_N^{\mathrm{T}}\boldsymbol{M}^{-1})\ \boldsymbol{t}_M = (\boldsymbol{N}\boldsymbol{n})^{\mathrm{T}}\boldsymbol{t}_M$$

$$\boldsymbol{N} = (\boldsymbol{M}^{-1})^{\mathrm{T}} = k.\begin{bmatrix} m_{11}^c & m_{12}^c & m_{13}^c \\ m_{21}^c & m_{22}^c & m_{23}^c \\ m_{31}^c & m_{32}^c & m_{33}^c \end{bmatrix}$$

Translation and Affine Transformations

$$x' = x + x_t$$
$$y' = y + y_t$$

$$x' = m_{11}x + m_{12}y$$
$$y' = m_{21}x + m_{22}y$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & x_t \\ m_{21} & m_{22} & y_t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11}x + m_{12}y + x_t \\ m_{21}x + m_{22}y + y_t \\ 1 \end{bmatrix}$$

Handling Points and Vectors

Location: $$\begin{bmatrix} 1 & 0 & x_t \\ 0 & 1 & y_t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x + x_t \\ y + y_t \\ 1 \end{bmatrix}$$

Direction: $$\begin{bmatrix} 1 & 0 & x_t \\ 0 & 1 & y_t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 0 \end{bmatrix} = \begin{bmatrix} x \\ y \\ 0 \end{bmatrix}$$

Homogeneous Coordinates

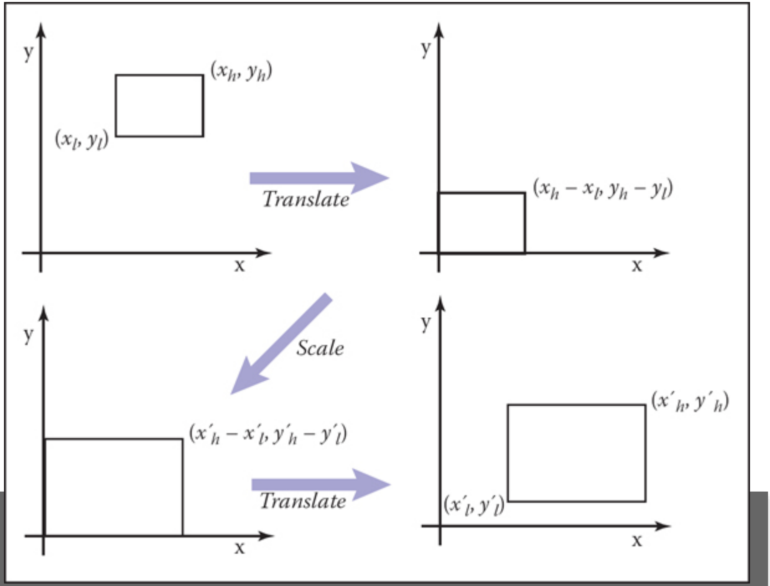
Extend by one dimension ($z = 1$)

$$\begin{bmatrix} 1 & 0 & x_t \\ 0 & 1 & y_t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x + x_t z \\ y + y_t z \\ z \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & x_t \\ 0 & 1 & y_t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x + x_t \\ y + y_t \\ 1 \end{bmatrix}$$

Concatenating Transformations

$$\begin{bmatrix} 1 & 0 & x_t \\ 0 & 1 & y_t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\varphi & -\sin\varphi & x_t \\ \sin\varphi & \cos\varphi & y_t \\ 0 & 0 & 1 \end{bmatrix}$$

Translation in concatenated matrix comes last.



Move the point $[x_l, y_l]$ to origin.

Scale to target rectangle size.

Move origin to point $[x_l', y_l']$.

Inverses of Transformation Matrices

$$\boldsymbol{M}^{-1} = (\boldsymbol{M}_N \boldsymbol{M}_3 \boldsymbol{M}_2 \boldsymbol{M}_1)^{-1} = \boldsymbol{M}_1^{-1} \boldsymbol{M}_2^{-1} \boldsymbol{M}_3^{-1} \boldsymbol{M}_N^{-1}$$

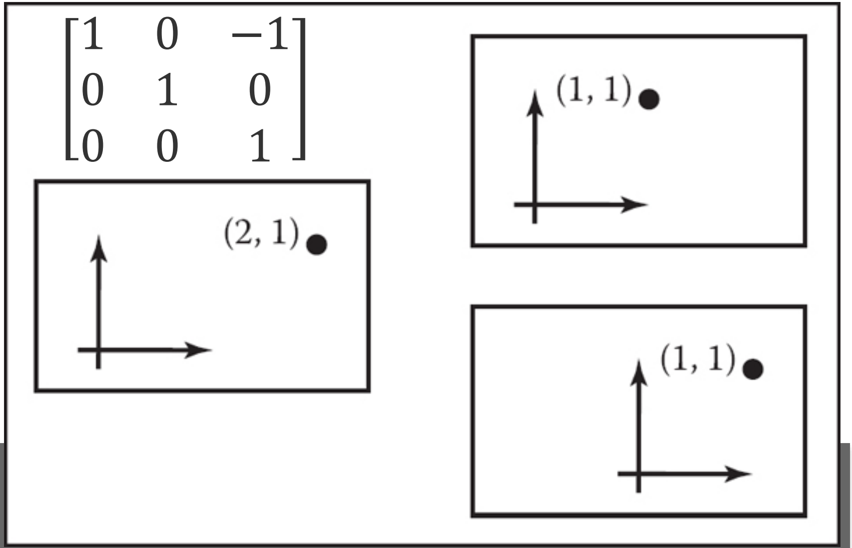
$$\boldsymbol{M} = \boldsymbol{R}_1 \boldsymbol{S}(s_x, s_y, s_z) \boldsymbol{R}_2$$

$$\boldsymbol{M}^{-1} = \boldsymbol{R}_2^{\mathrm{T}} \boldsymbol{S}\left(\frac{1}{s_x}, \frac{1}{s_y}, \frac{1}{s_z}\right) \boldsymbol{R}_1^{\mathrm{T}}$$

Coordinate Transformations

Transforming objects wrt. the coordinate system is equivalent to inverse transformation of the coordinate system.

Ex.: moving car in a city



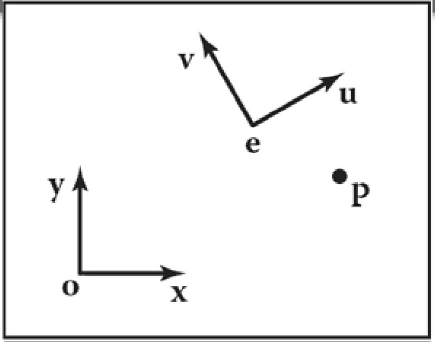
Coordinate Transformations (cont.)

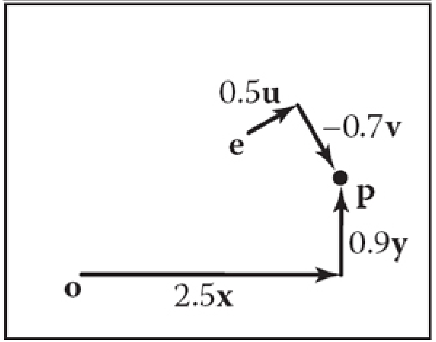
Coordinate frame $\boldsymbol{e} + u\boldsymbol{u} + v\boldsymbol{v}$

$$\boldsymbol{p} = (x_p, y_p) \equiv \boldsymbol{0} + x_p\boldsymbol{x} + y_p\boldsymbol{y}$$

$$\boldsymbol{p} = (u_p, v_p) \equiv \boldsymbol{e} + u_p\boldsymbol{u} + v_p\boldsymbol{v}$$

$$\begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & x_e \\ 0 & 1 & y_e \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_u & x_v & 0 \\ y_u & y_v & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_p \\ v_p \\ 1 \end{bmatrix}$$



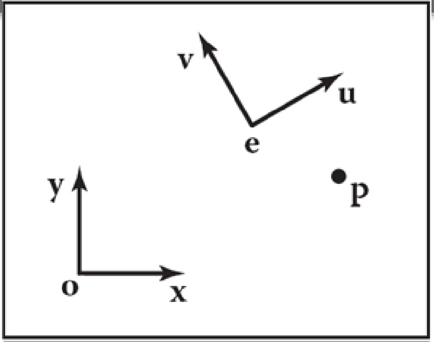


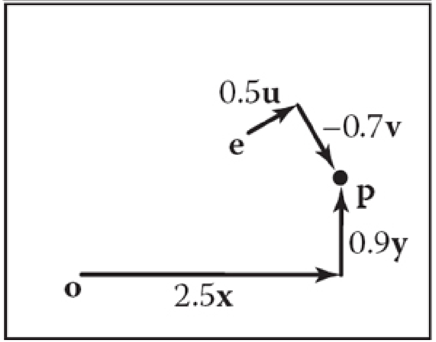
Coordinate Transformations (cont.)

Coordinate frame $\boldsymbol{e} + u\boldsymbol{u} + v\boldsymbol{v}$

$$\boldsymbol{p} = (x_p, y_p) \equiv \boldsymbol{0} + x_p\boldsymbol{x} + y_p\boldsymbol{y}$$

$$\boldsymbol{p} = (u_p, v_p) \equiv \boldsymbol{e} + u_p\boldsymbol{u} + v_p\boldsymbol{v}$$





$$\begin{bmatrix} u_p \\ v_p \\ 1 \end{bmatrix} = \begin{bmatrix} x_u & y_u & 0 \\ x_v & y_v & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -x_e \\ 0 & 1 & -y_e \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix}$$

Coordinate Transformations (cont.)

$$\boldsymbol{p}_{x,y,z} = \begin{bmatrix} \boldsymbol{u} & \boldsymbol{v} & \boldsymbol{w} & \boldsymbol{e} \\ 0 & 0 & 0 & 1 \end{bmatrix} \boldsymbol{p}_{u,v,w} =$$

$$\begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & x_e \\ 0 & 1 & 0 & y_e \\ 0 & 0 & 1 & z_e \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_u & x_v & x_w & 0 \\ y_u & y_v & y_w & 0 \\ z_u & z_v & z_w & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_p \\ v_p \\ w_p \\ 1 \end{bmatrix}$$

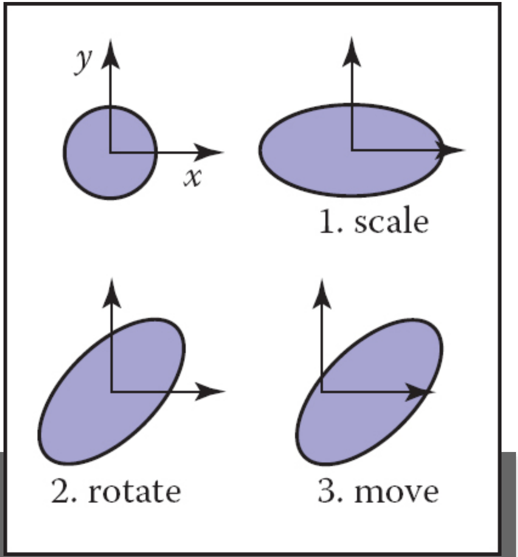
Coordinate Transformations (cont.)

$$\boldsymbol{p}_{u,v,w} = \begin{bmatrix} \boldsymbol{u} & \boldsymbol{v} & \boldsymbol{w} & \boldsymbol{e} \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} \boldsymbol{p}_{x,y,z} =$$

$$\begin{bmatrix} u_p \\ v_p \\ w_p \\ 1 \end{bmatrix} = \begin{bmatrix} x_u & y_u & z_u & 0 \\ x_v & y_v & z_v & 0 \\ x_w & y_w & z_w & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -x_e \\ 0 & 1 & 0 & -y_e \\ 0 & 0 & 1 & -z_e \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix}$$

Instancing

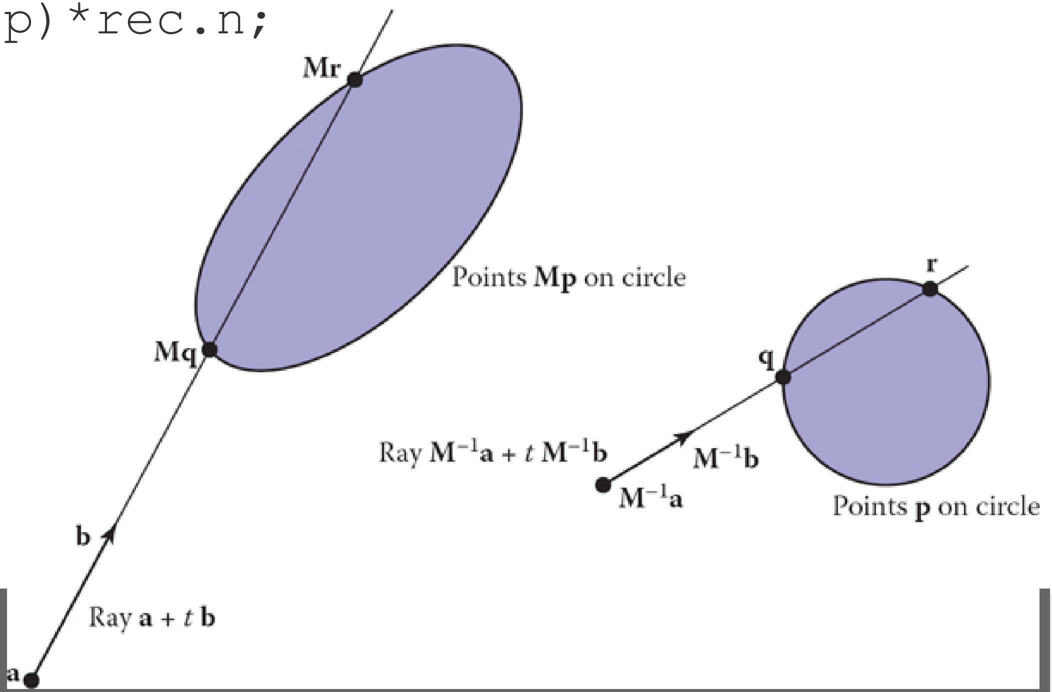
- One instance of untransformed object
- Transform an object, before the raytracing
- Choose the space for intersection
  - Ray on transformed point $\boldsymbol{Mp}$
  - Inverse-transformed ray on untransformed point p



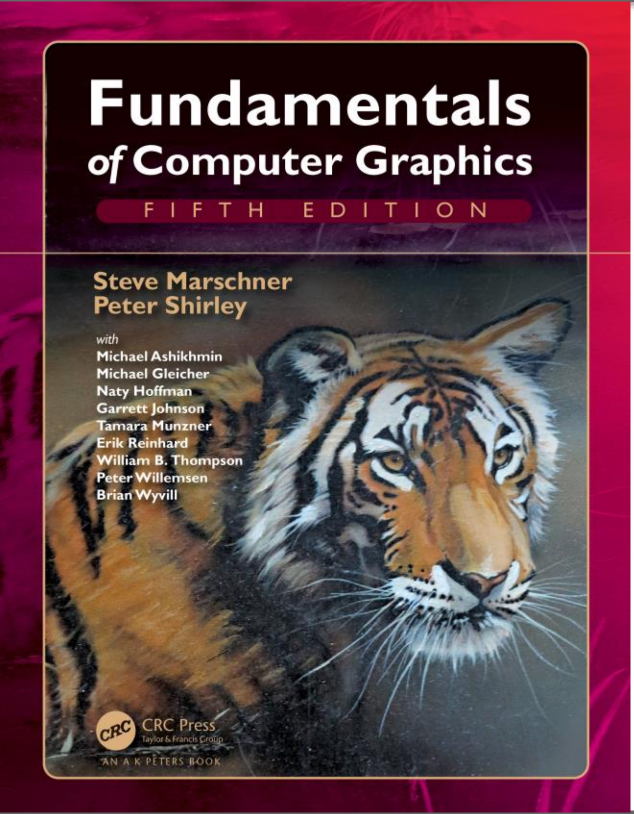
Instancing (cont.)

```
instance::hit(ray a+tb,real t0,real t1,hit-record rec) {
    ray r' = M_inv*a + t*M_inv*b;
    if (base-object→hit(r',t0,t1,rec)) {
        rec.n = (M_inv_transp)*rec.n;
        return true;
    } else return false;
```

- Ray parameter t is the same in either space
- Viewing vector should not be normalized



Credits



Fundamentals of Computer Graphics, 5th Edition

by Peter Shirley, Steve Marschner

Publisher: A K Peters/CRC Press

Release Date: September 30, 2021

ISBN: 9781000426359

https://learning.oreilly.com/library/view/fundamentals-of-computer/9781000426359/