Lecture 4/5:

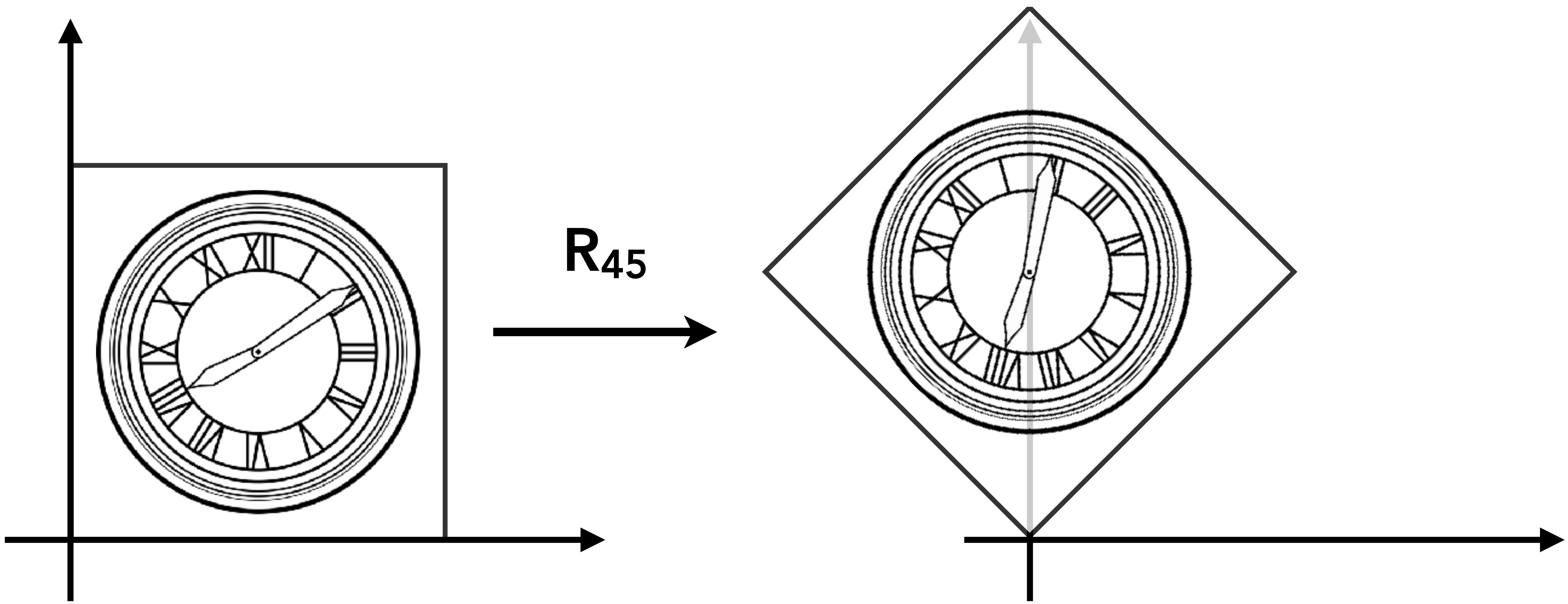
Transforms

Computer Graphics and Imaging

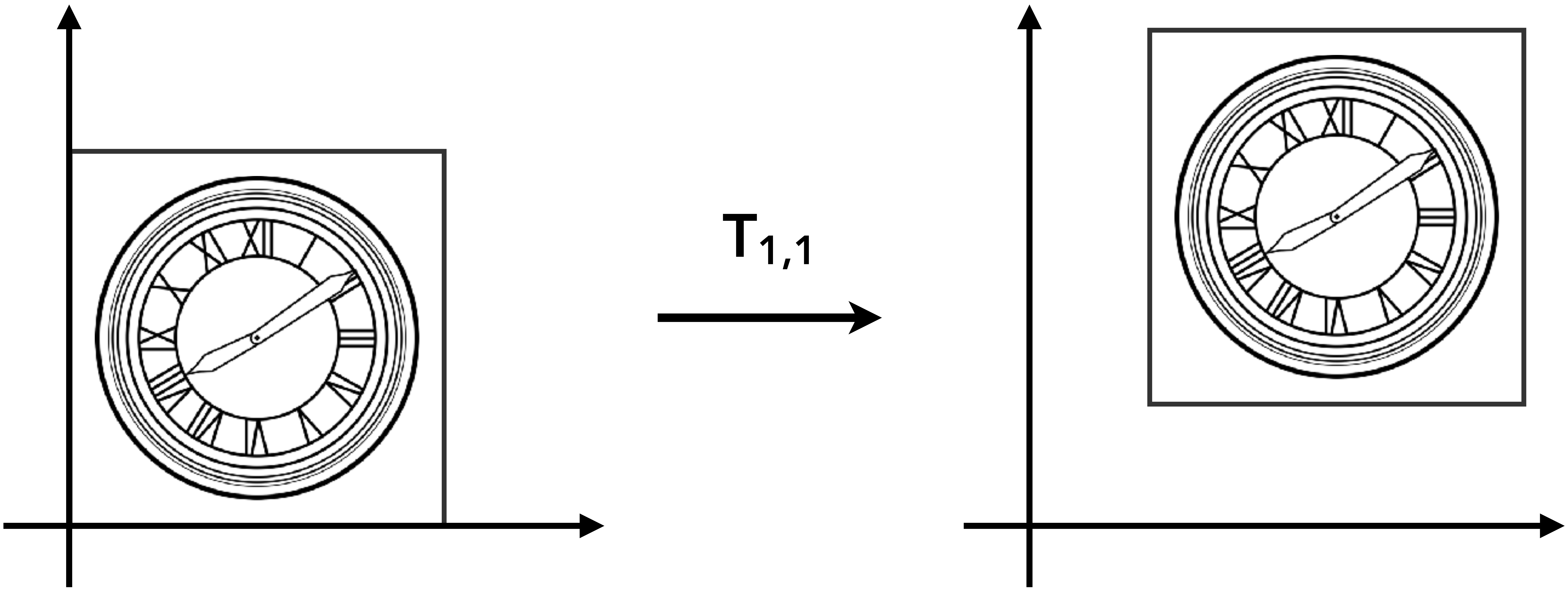
UC Berkeley CS184/284A

Basic Transforms

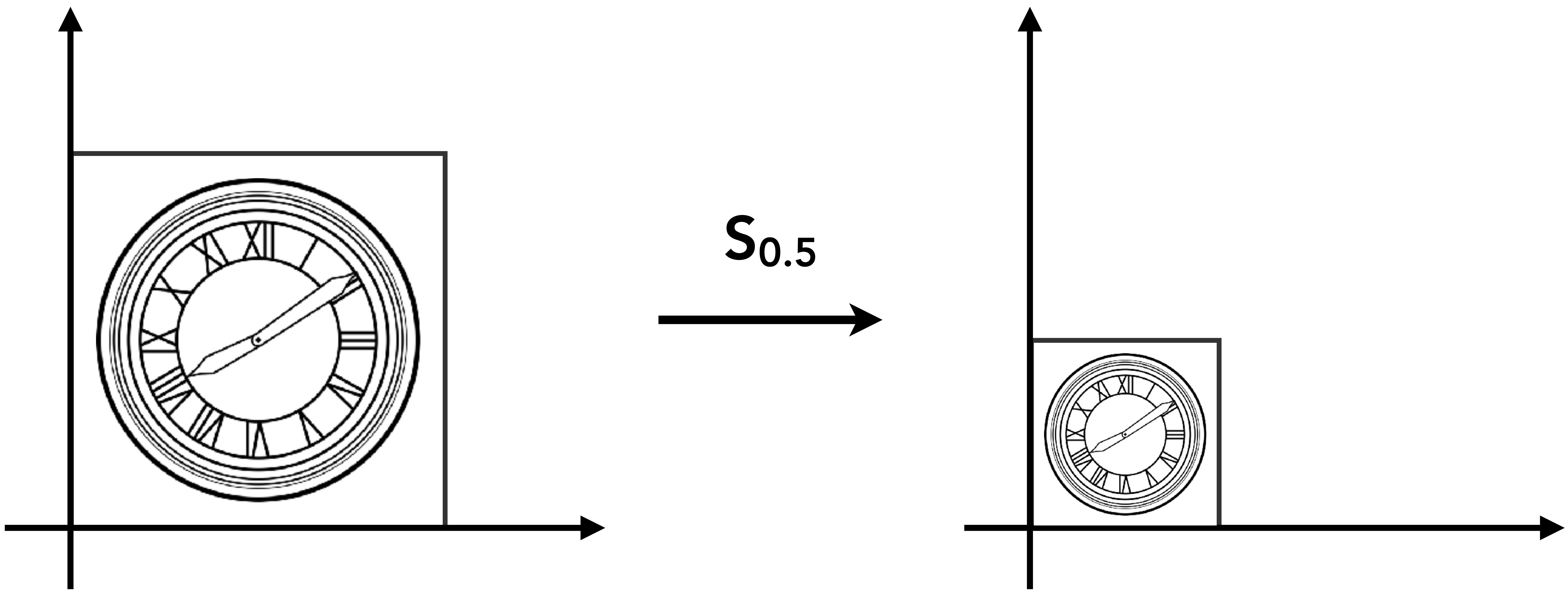
Rotate



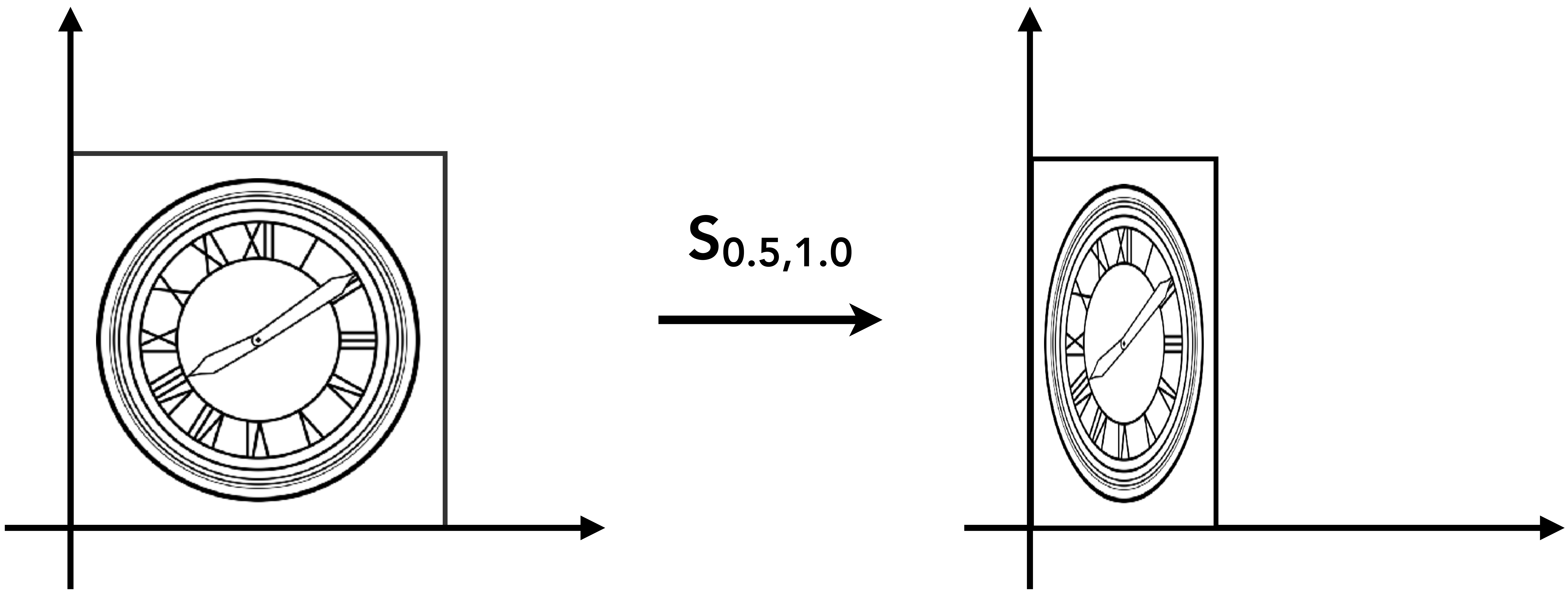
Translate



Scale



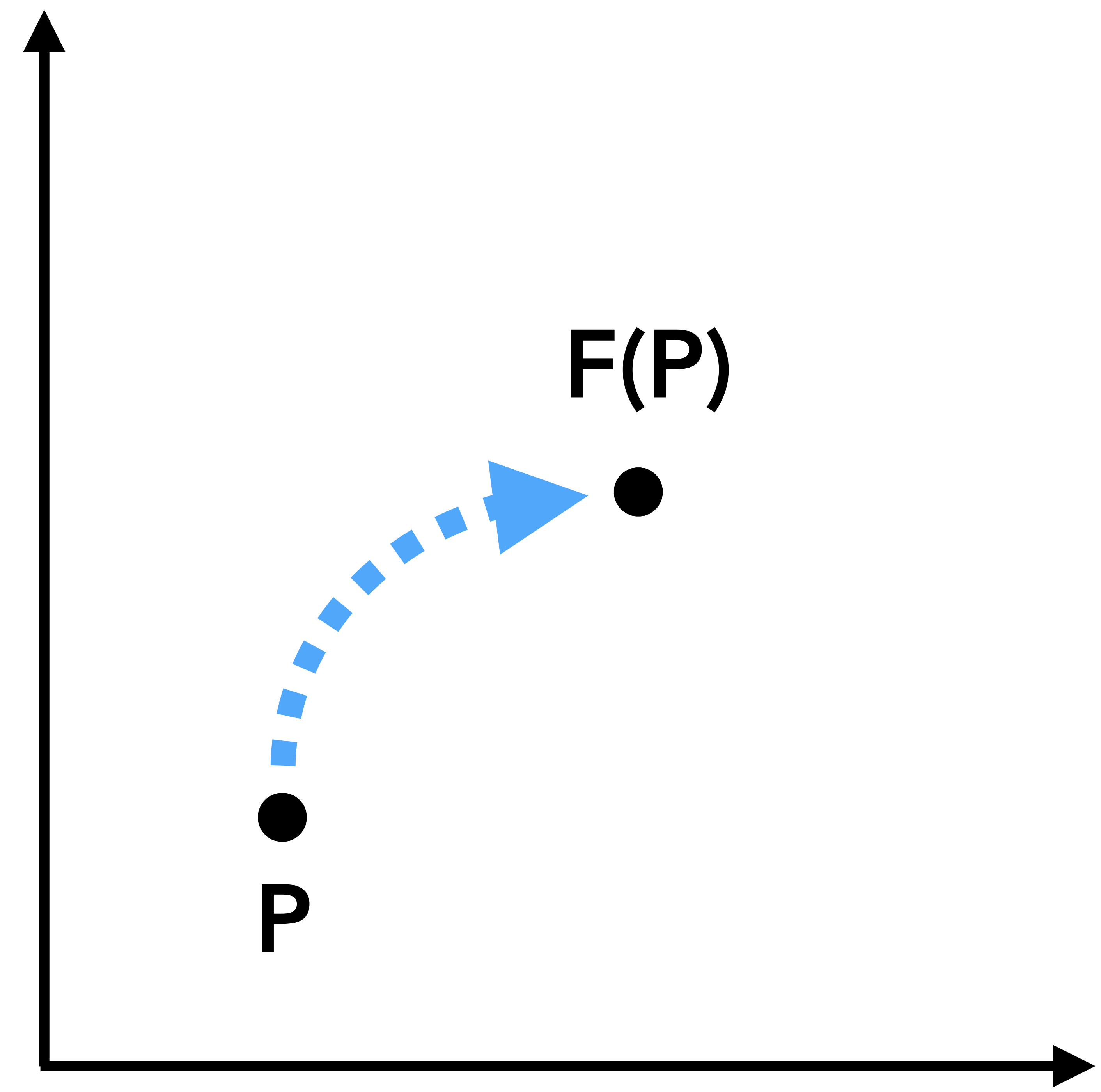
Scale (Non-Uniform)



What Are Transforms?

Just functions acting on points

- $(x', y', z') = F(x, y, z)$
- $P' = F(P)$



Why Study Transforms?

Modeling

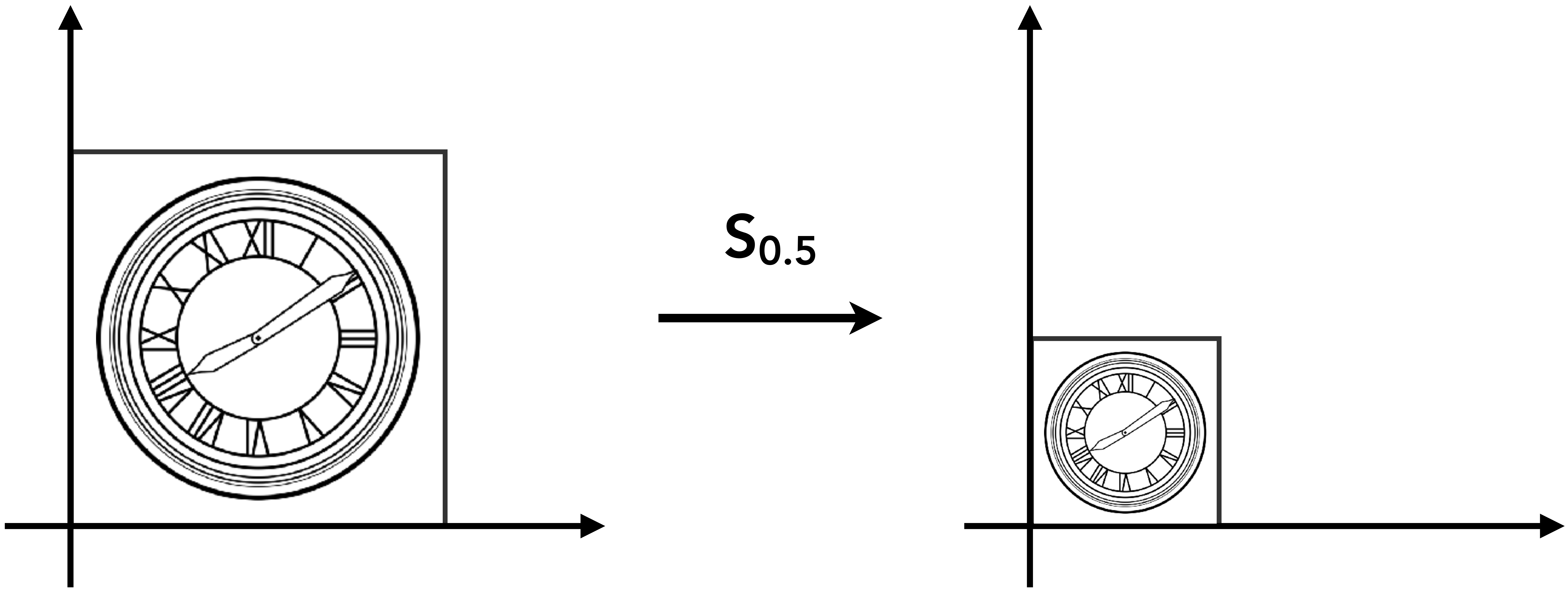
- Define shapes in convenient coordinates
- Enable multiple copies of the same object
- Efficiently represent hierarchical scenes

Viewing

- World coordinates to camera coordinates
- Parallel / perspective projections from 3D to 2D

Linear Transforms = Matrices

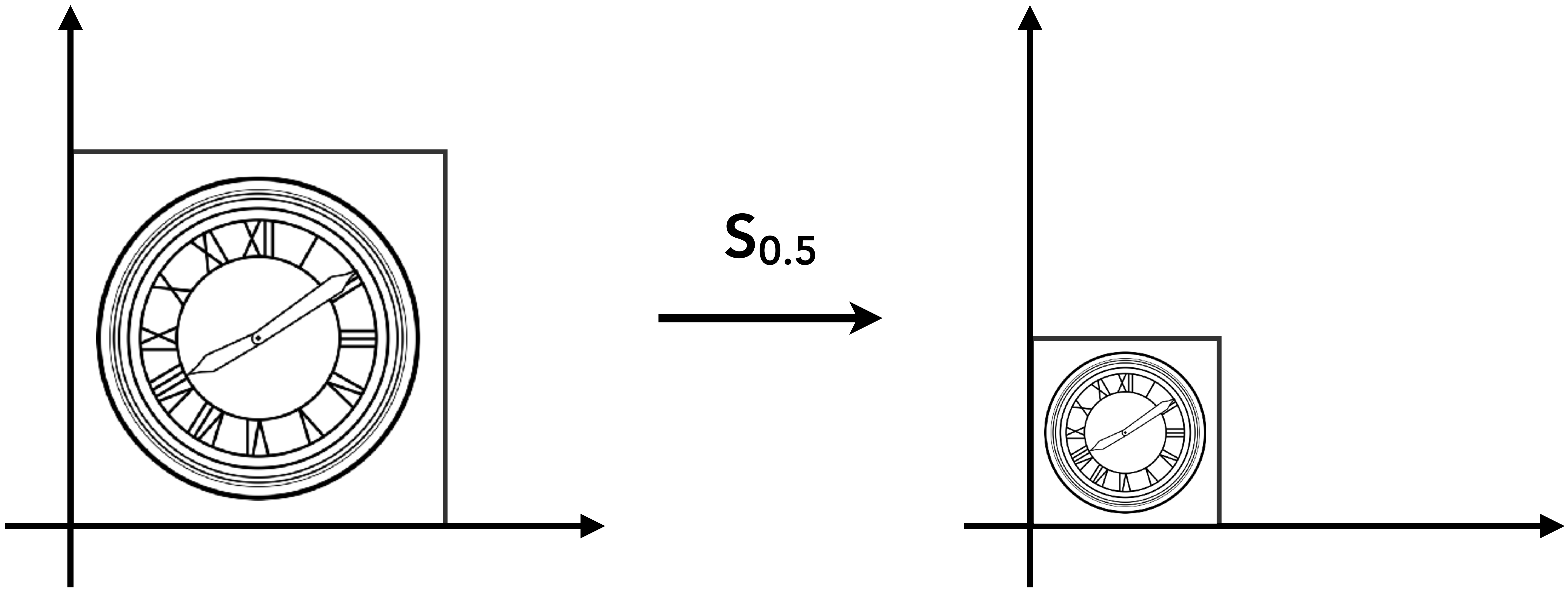
Scale Transform



$$x' = sx$$

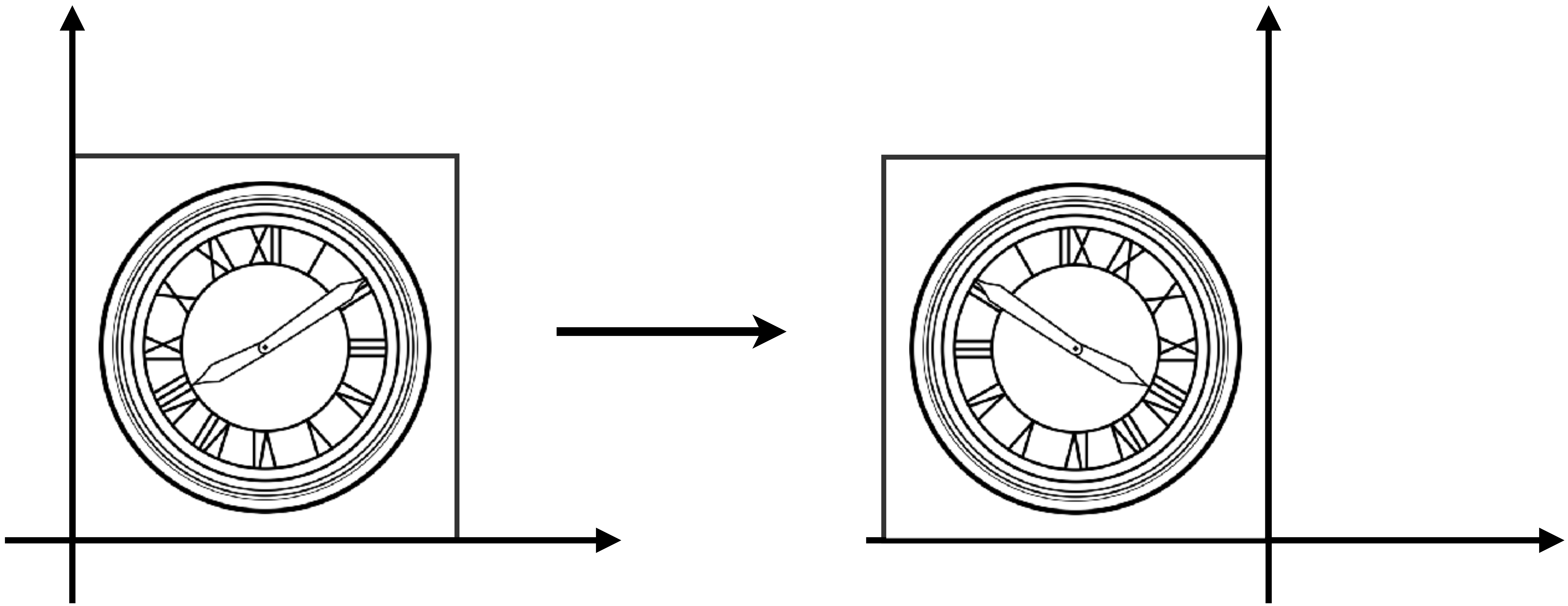
$$y' = sy$$

Scale Matrix



$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

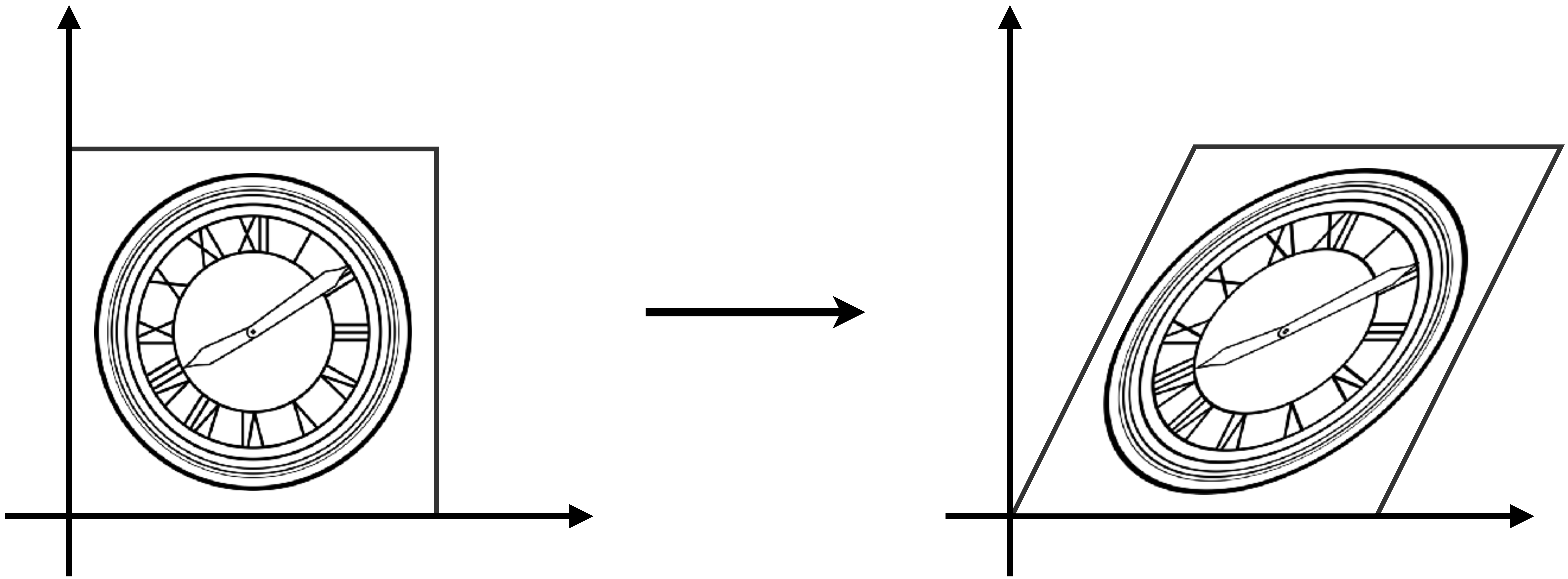
Reflection Matrix



$$x' = ??$$

$$y' = ??$$

Shear Matrix



$$x' = ??$$

$$y' = ??$$

Linear Transforms = Matrices

$$x' = a\,x + b\,y$$
$$y' = c\,x + d\,y$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\mathbf{x}' = \mathbf{M}\,\mathbf{x}$$

2D Coordinate Systems

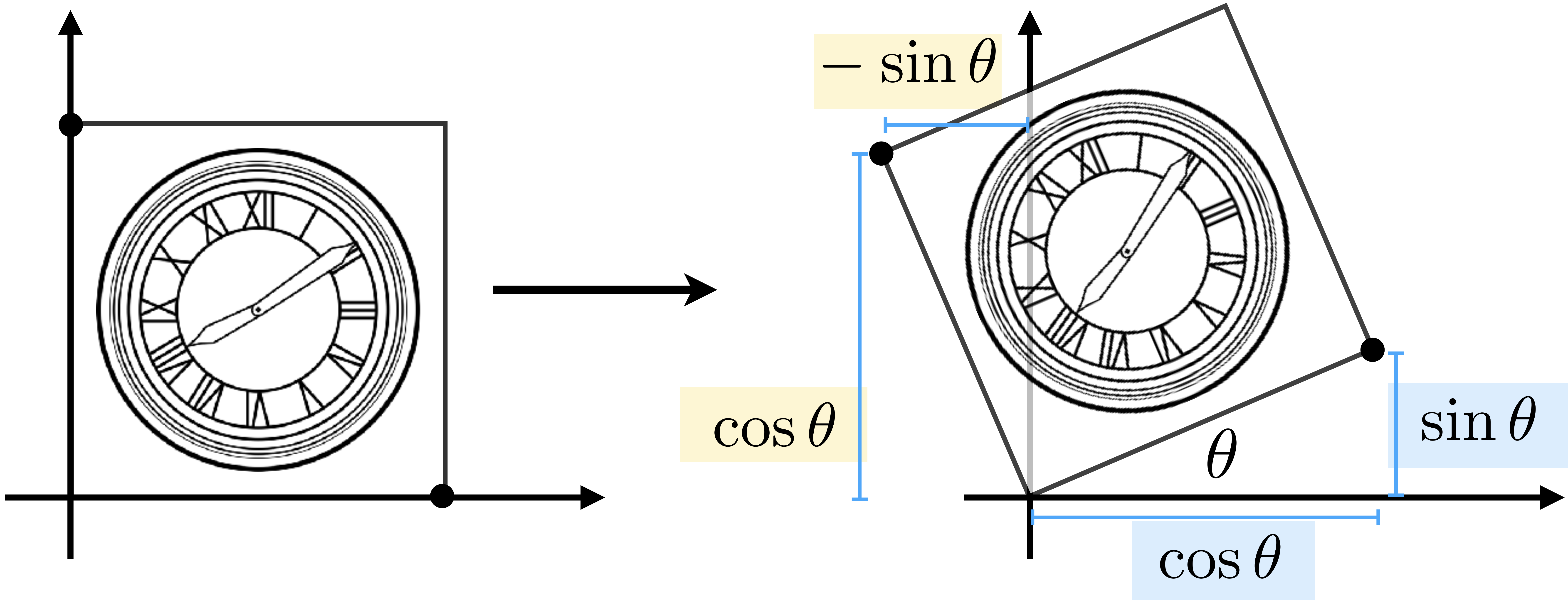
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\begin{bmatrix} a \\ c \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} b \\ d \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Can interpret the columns of the matrix as the x and y axes of the coordinate frame

Rotation Matrix

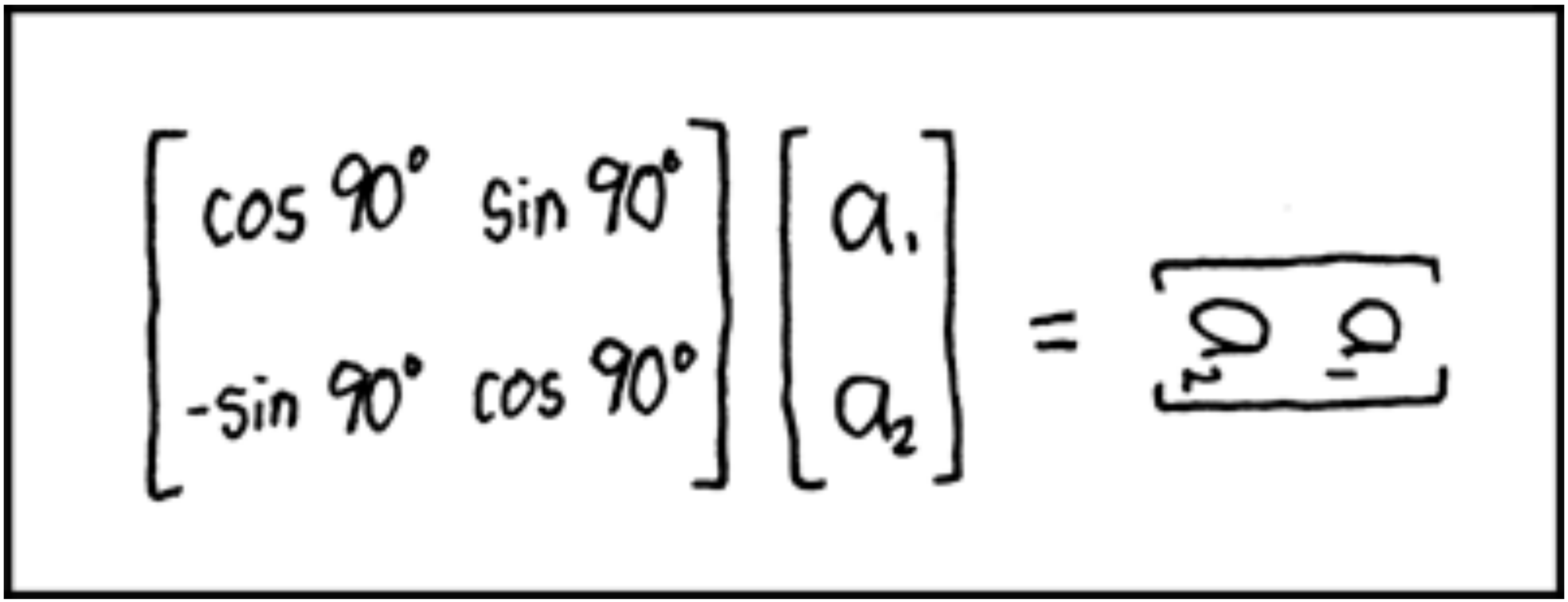


$$\mathbf{R}_\theta = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

2D Rotation Matrix: Another Way

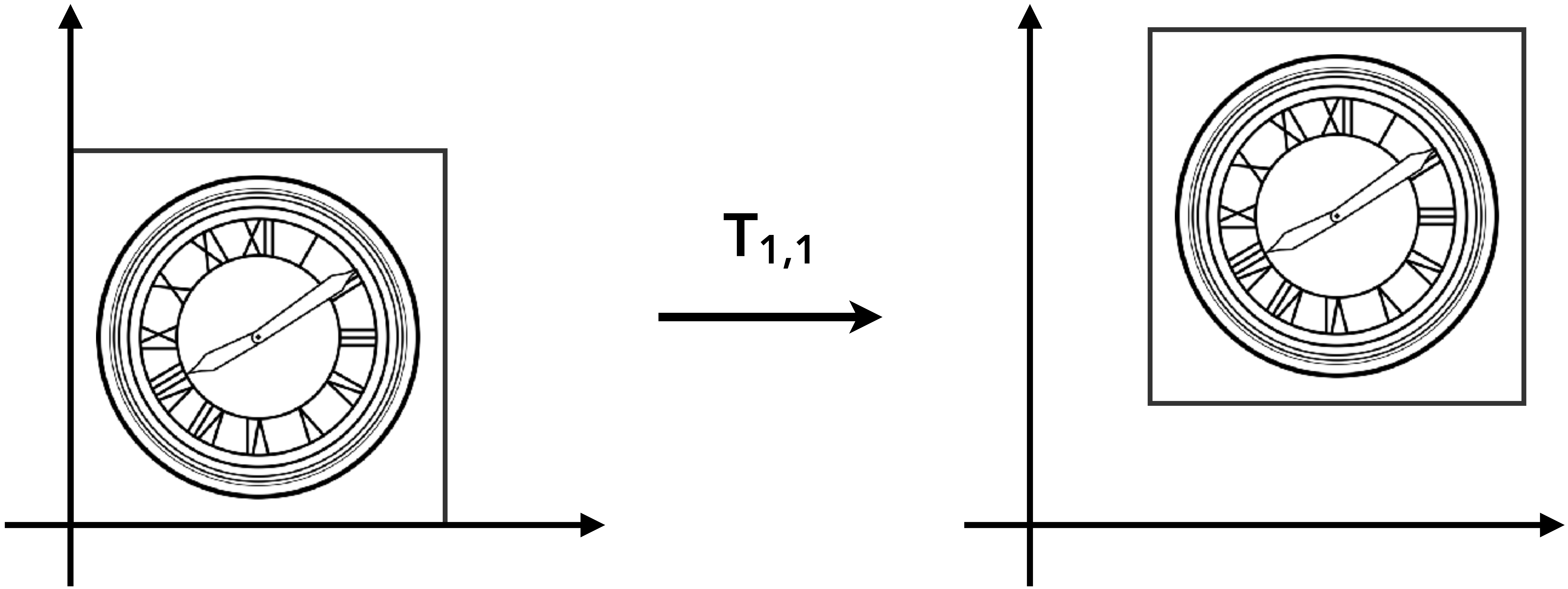
xkcd

A WEBCOMIC OF ROMANCE, SARCASM, MATH, AND LANGUAGE.



http://xkcd.com/184/

Translation??



$$x' = x + t_x$$
$$y' = y + t_y$$

Solution: Homogenous Coordinates

Add a third coordinate (*w-coordinate*)

- 2D point = $(x, y, 1)^T$
- 2D vector = $(x, y, 0)^T$

Matrix representation of translations

$$\begin{pmatrix} x' \\ y' \\ w' \end{pmatrix} = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} x + t_x \\ y + t_y \\ 1 \end{pmatrix}$$

Homogenous Coordinates

Valid operation if w-coordinate of result is 1 or 0

- vector + vector = vector
- point – point = vector
- point + vector = point
- point + point = ??

Affine Transformations

Affine map = linear map + translation

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix}$$

Using homogenous coordinates:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & t_x \\ c & d & t_y \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

2D Transformations

Scale

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

Rotation

$$\mathbf{R}(\alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

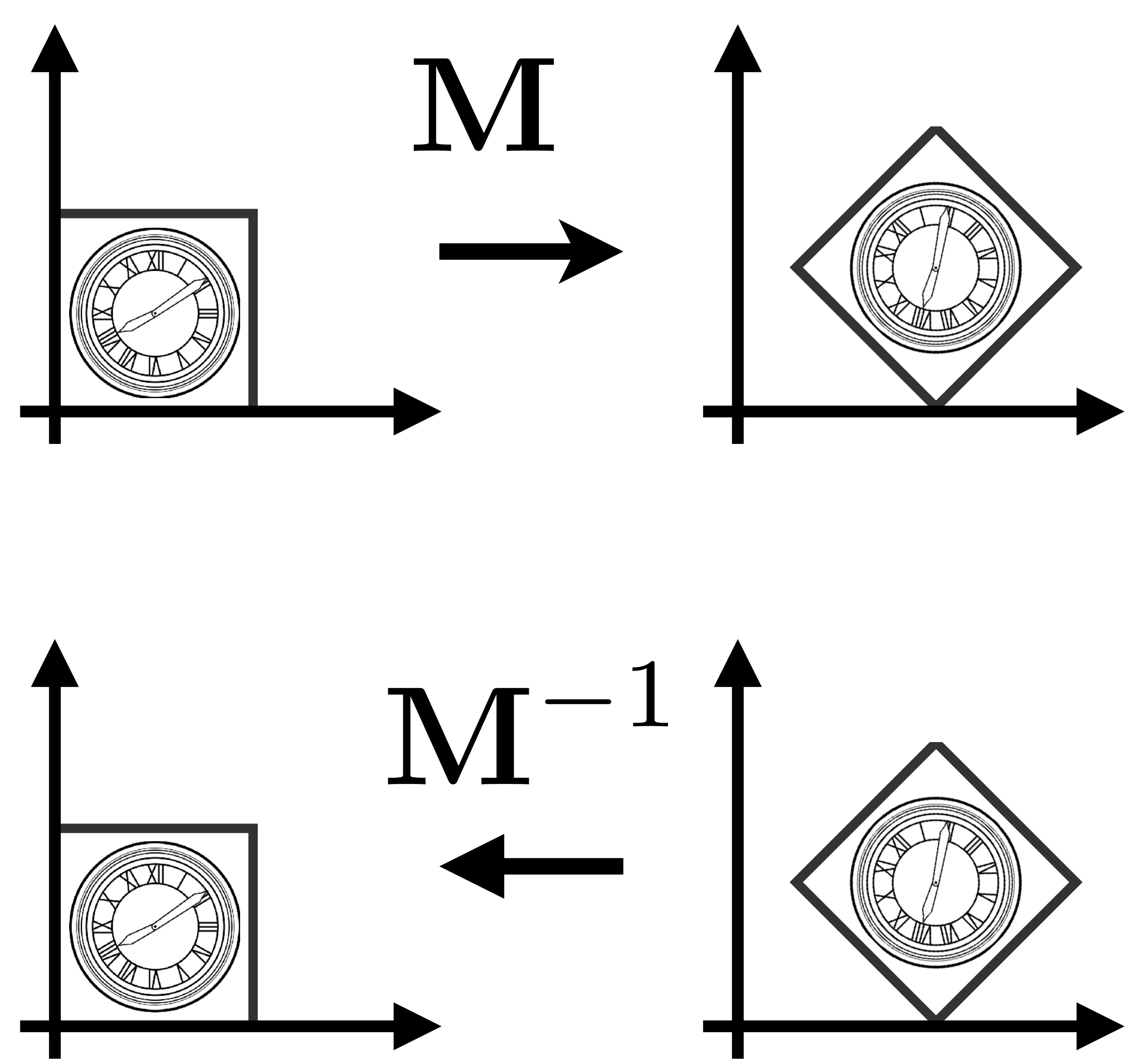
Translation

$$\mathbf{T}(t_x, t_y) = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix}$$

Inverse Transform

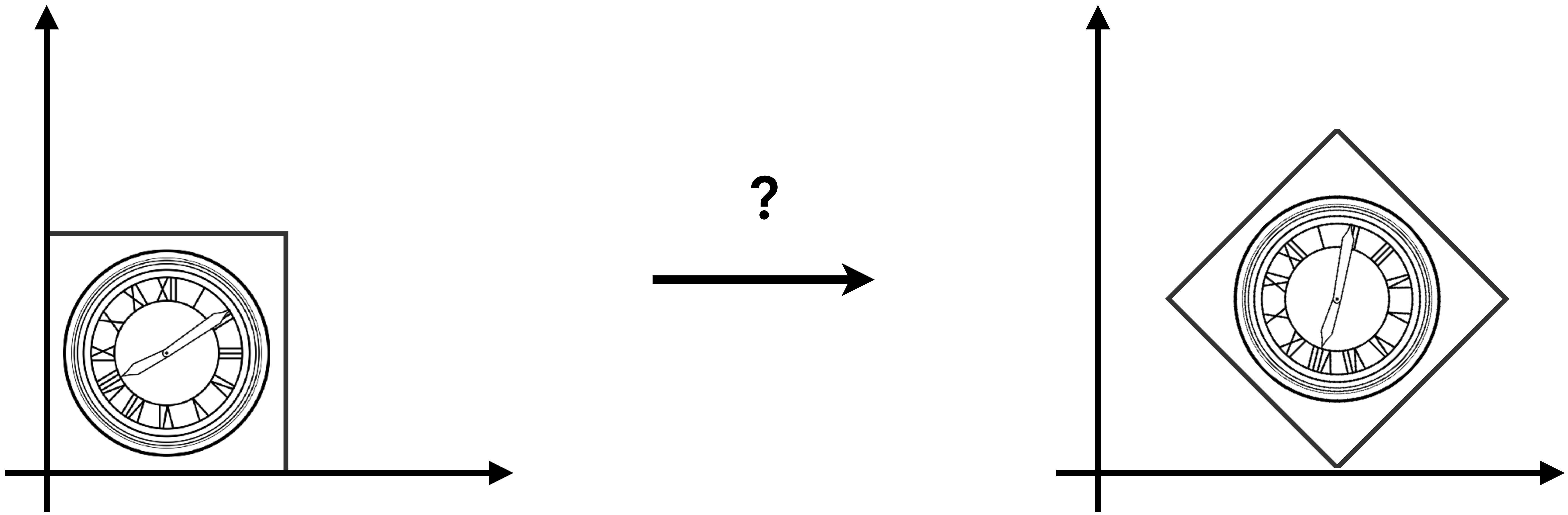
$$\mathbf{M}^{-1}$$

$\mathbf{M}^{-1}$ is the inverse of transform $\mathbf{M}$ in both a matrix and geometric sense

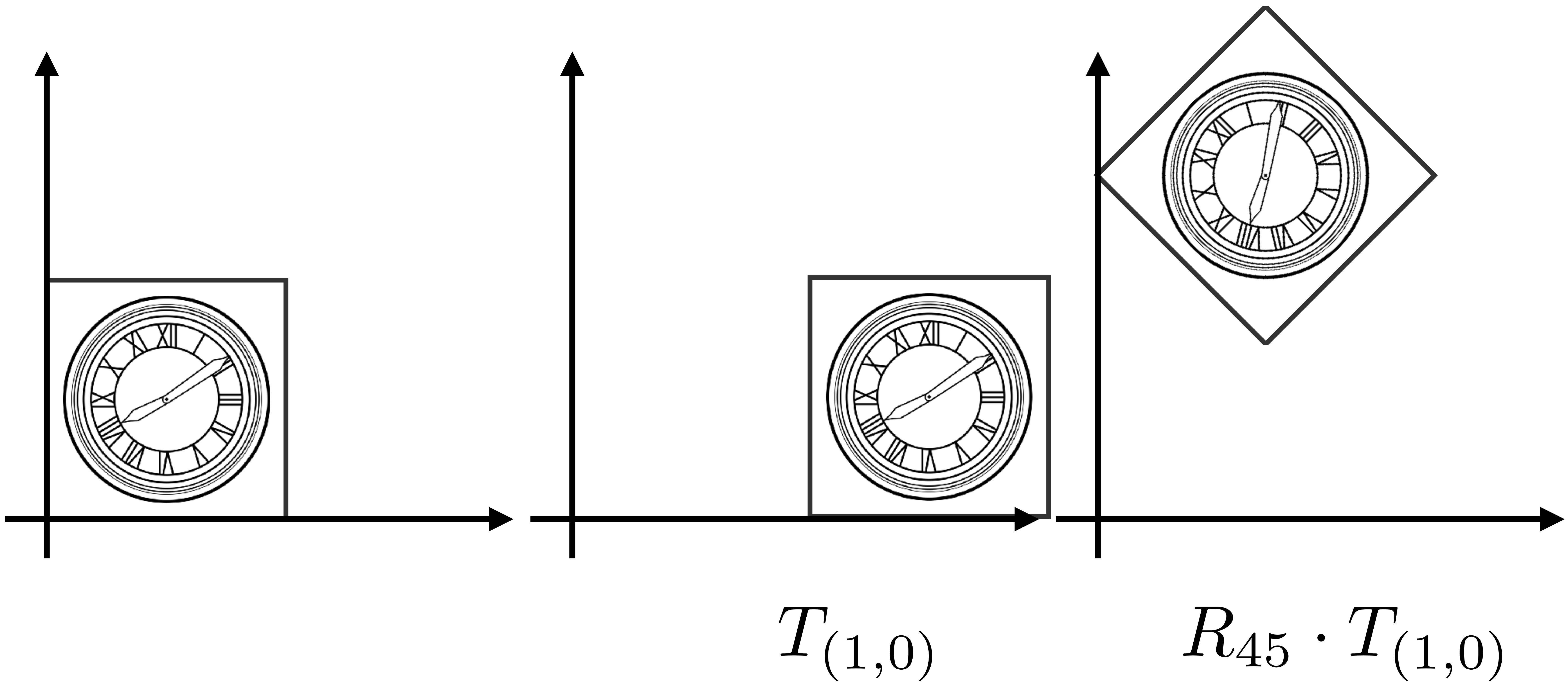


Composing Transforms

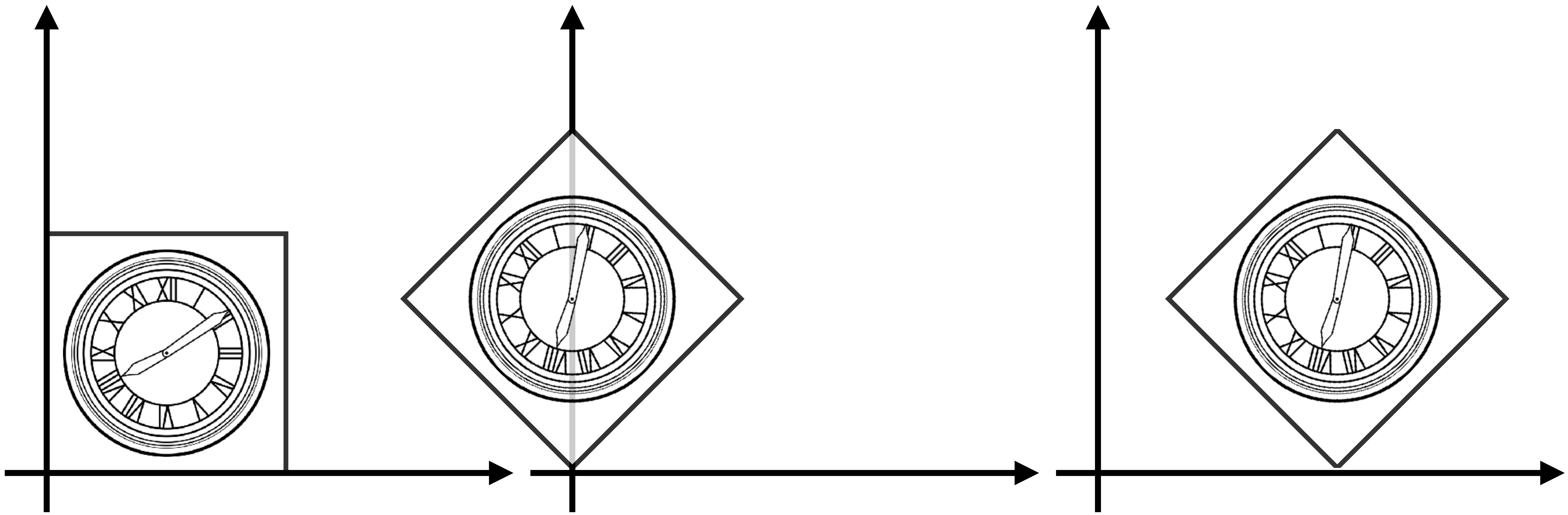
Composite Transform



Translate Then Rotate?

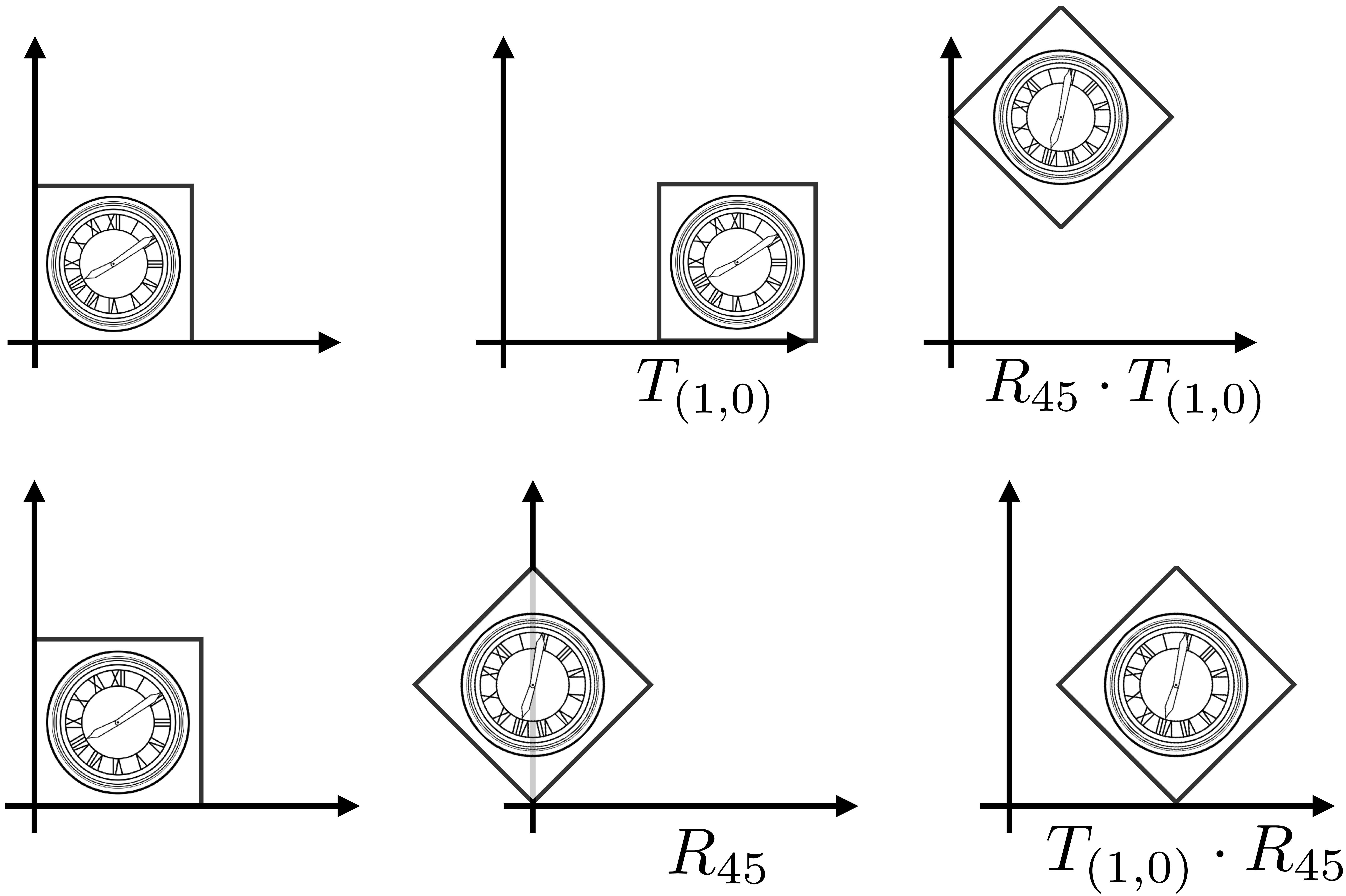


Rotate Then Translate



R_{45} $\quad$ $T_{(1,0)} \cdot R_{45}$

Transform Ordering Matters!



Transform Ordering Matters!

Matrix multiplication is <u>not</u> commutative

$$R_{45} \cdot T_{(1,0)} \neq T_{(1,0)} \cdot R_{45}$$

Recall the matrix math represented by these symbols:

$$\begin{bmatrix} \cos 45^\circ & -\sin 45^\circ & 0 \\ \sin 45^\circ & \cos 45^\circ & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \neq \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos 45^\circ & -\sin 45^\circ & 0 \\ \sin 45^\circ & \cos 45^\circ & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Note that matrices are applied right to left:

$$T_{(1,0)} \cdot R_{45} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos 45^\circ & -\sin 45^\circ & 0 \\ \sin 45^\circ & \cos 45^\circ & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Composing Transforms

Sequence of affine transforms A_1, A_2, A_3, ...

- Compose by matrix multiplication
  - Very important for performance!

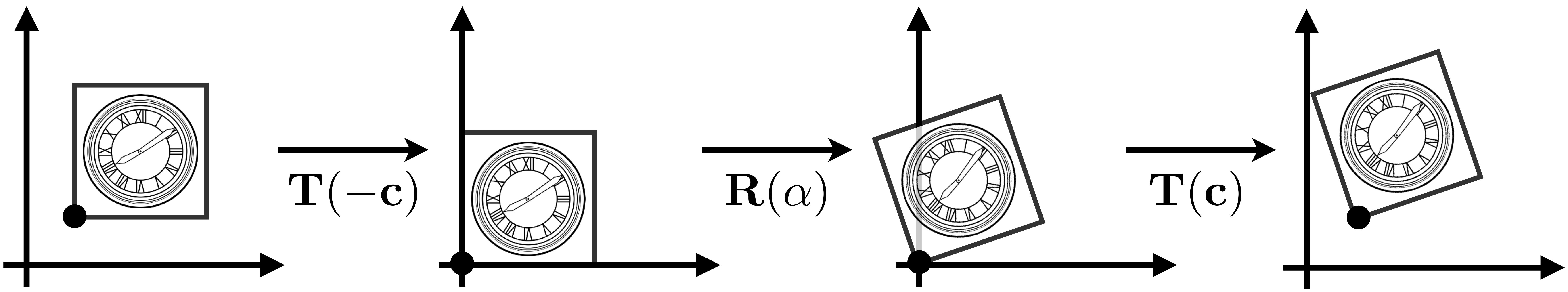
$$A_n(\dots A_2(A_1(\mathbf{x}))) \;=\; \underbrace{\mathbf{A}_n \cdots \mathbf{A}_2 \cdot \mathbf{A}_1}_{} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Pre-multiply *n* matrices to obtain a
single matrix representing combined transform

Decomposing Complex Transforms

How to rotate around a given point c?

1. Translate center to origin
2. Rotate
3. Translate back



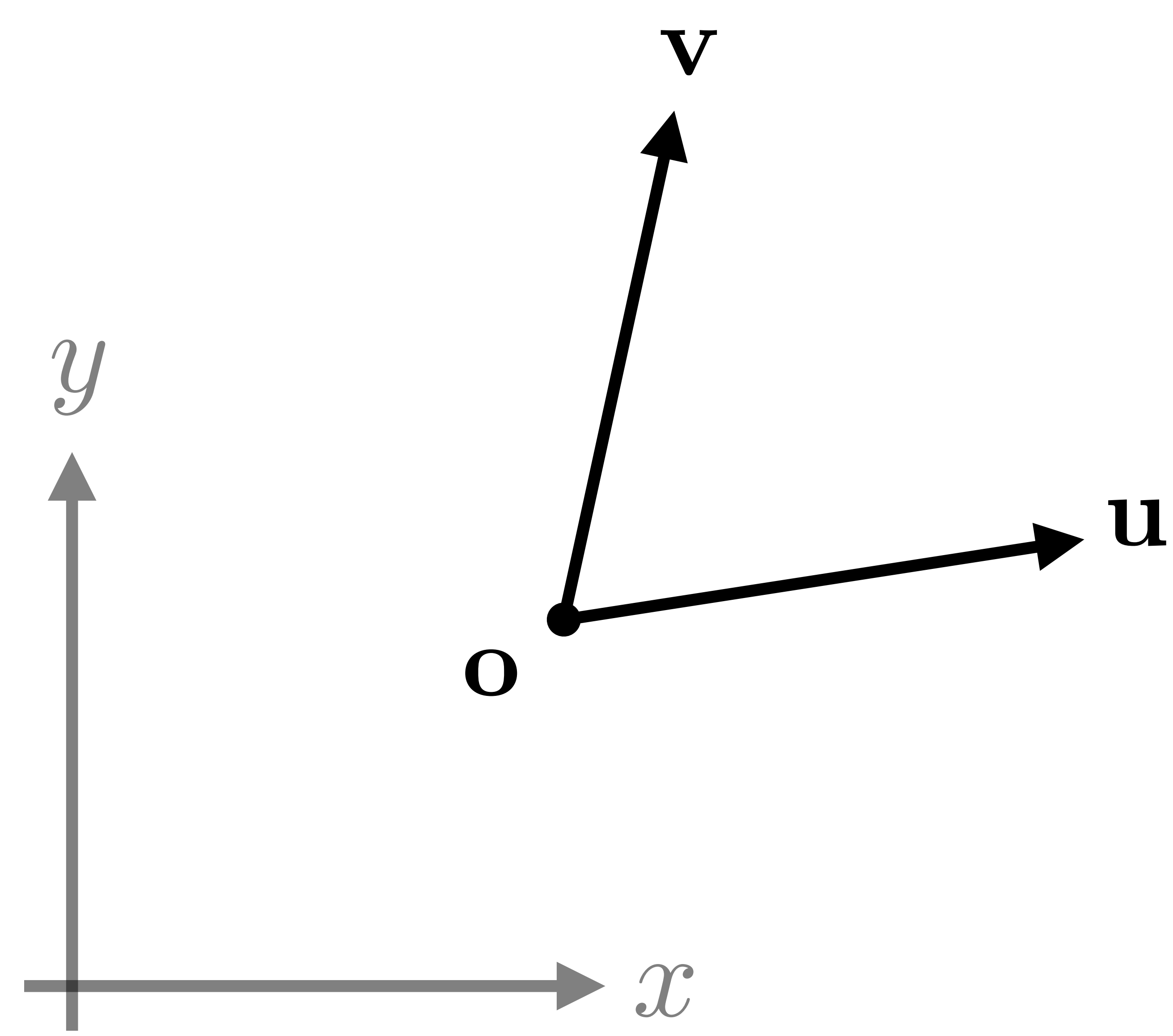
Matrix representation?

$$\mathbf{T}(\mathbf{c}) \cdot \mathbf{R}(\alpha) \cdot \mathbf{T}(-\mathbf{c})$$

Coordinate Systems

Coordinate System Transform

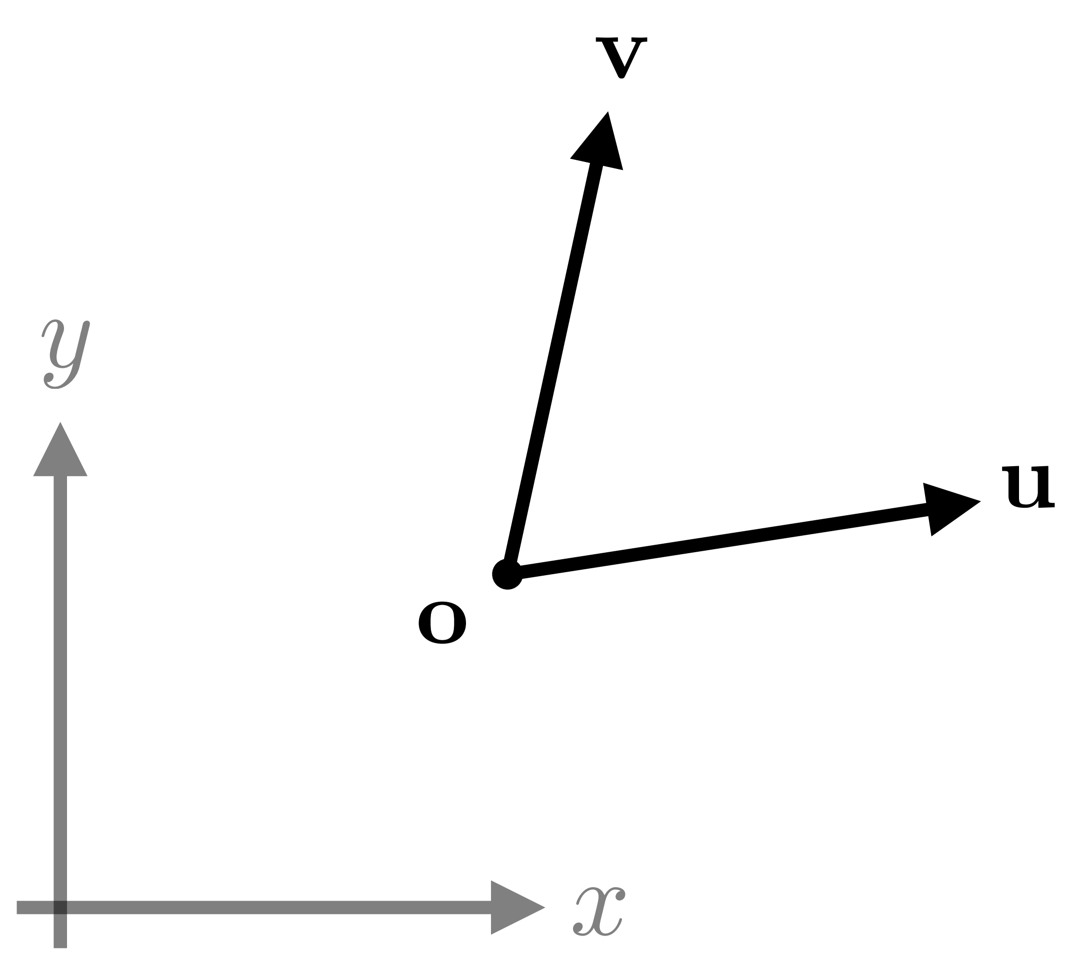
In general, a new coordinate frame is defined by an origin (point) and two unit axes (vectors)



Given coordinates in the (o,u,v) reference frame, what is the transform to coordinates in the (x,y) frame?

Coordinate System Transform Matrix

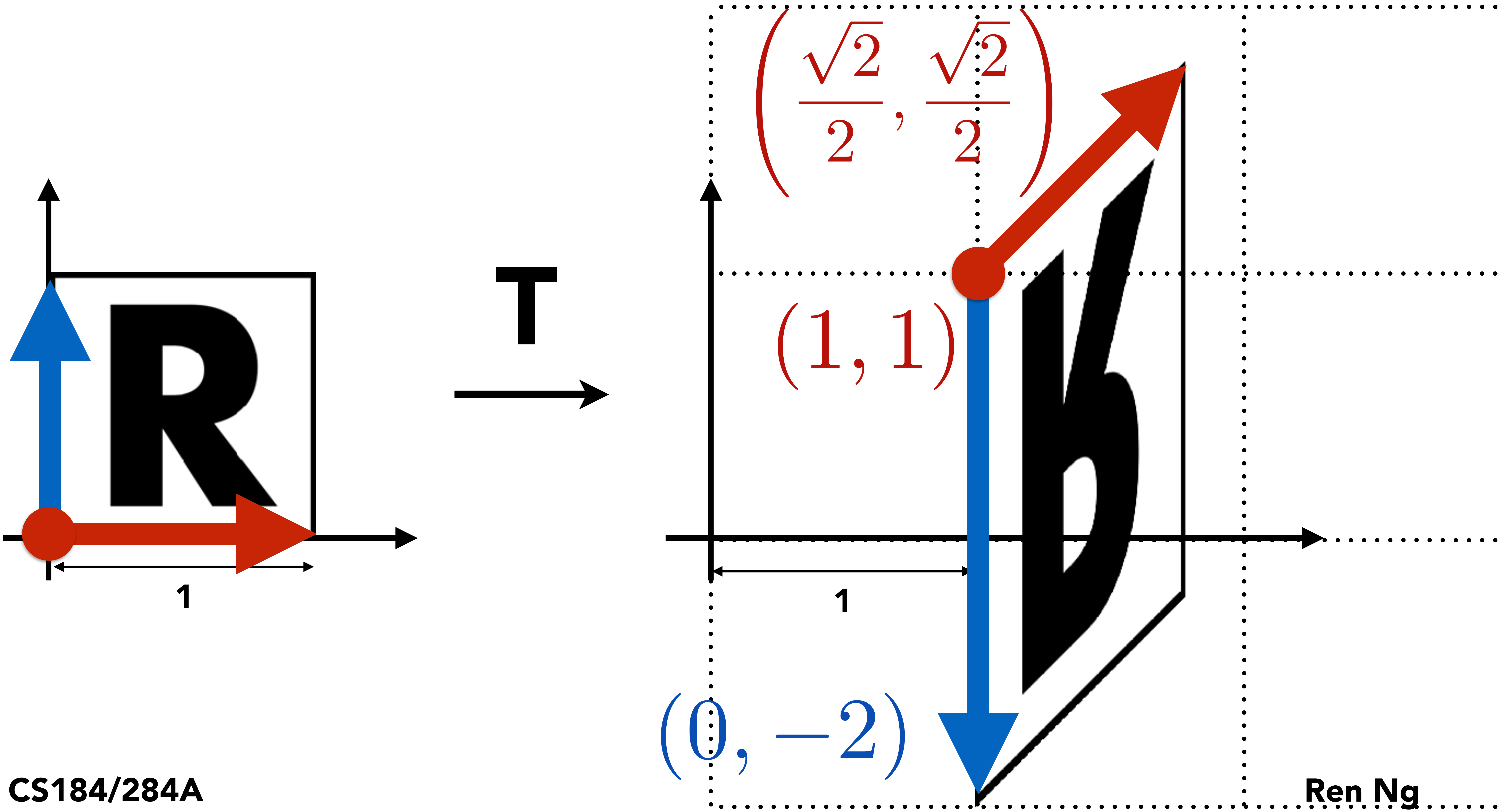
$$F = \begin{bmatrix} \mathbf{u} & \mathbf{v} & \mathbf{o} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} u_x & v_x & o_x \\ u_y & v_y & o_y \\ 0 & 0 & 1 \end{bmatrix}$$



- Columns of matrix are defined by the reference frame's coordinates in the world
- Gives a new way to "read off" columns of matrix

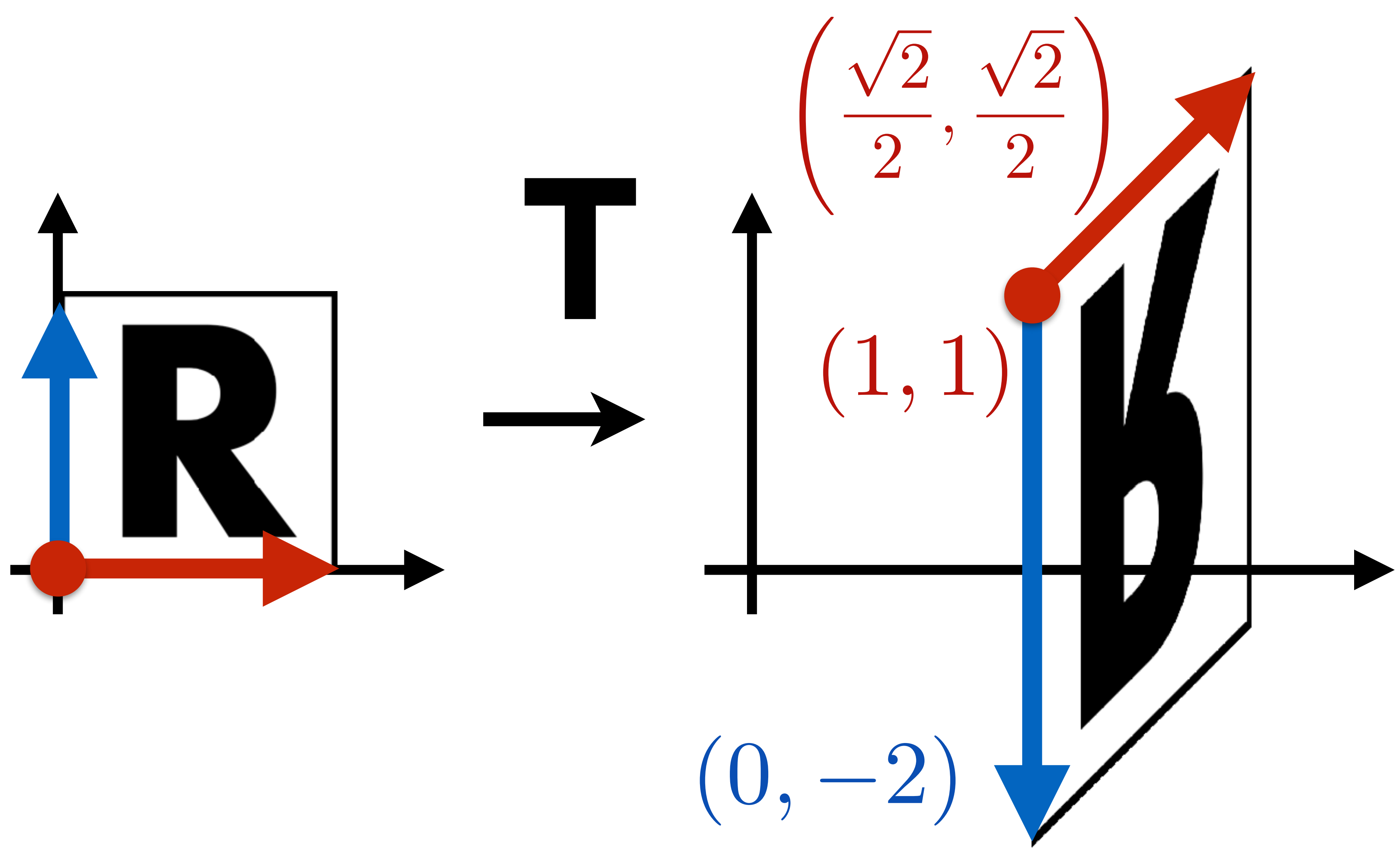
Coordinate System Transform - Example

Write down a matrix T representing this transform:



Coordinate System Transform - Example

Write down a matrix T representing this transform:

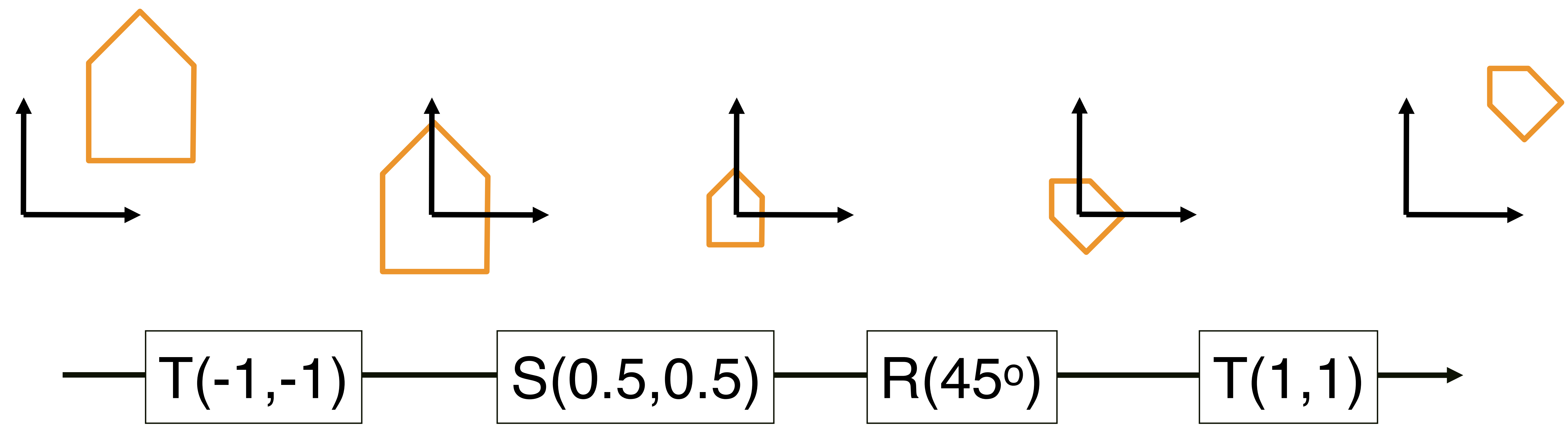


$$\begin{bmatrix} \mathbf{u} & \mathbf{v} & \mathbf{o} \\ 0 & 0 & 1 \end{bmatrix}$$

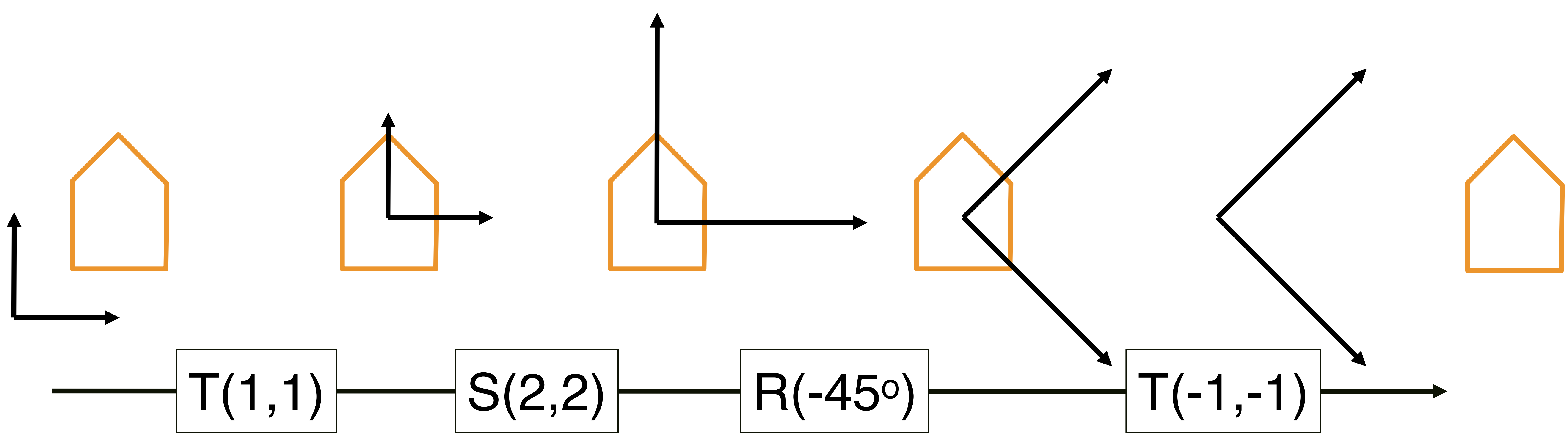
$$\begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & 1 \\ \frac{\sqrt{2}}{2} & -2 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

Two Interpretations of A Transform

Interpretation 1: Transforms object points



Interpretation 2: Transforms coordinate system



3D Transforms

3D Transformations

Use homogeneous coordinates again:

- 3D point = $(x, y, z, 1)^T$
- 3D vector = $(x, y, z, 0)^T$

Use 4×4 matrices for affine transformations

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & c & t_x \\ d & e & f & t_y \\ g & h & i & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

3D Transformations

Scale

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

Translation

$$\mathbf{T}(t_x, t_y, t_z) \ = \ \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Coordinate Change (Frame-to-world)

$$\mathbf{F}(\mathbf{u}, \mathbf{v}, \mathbf{w}, \mathbf{o}) = \begin{bmatrix} \mathbf{u} & \mathbf{v} & \mathbf{w} & \mathbf{o} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

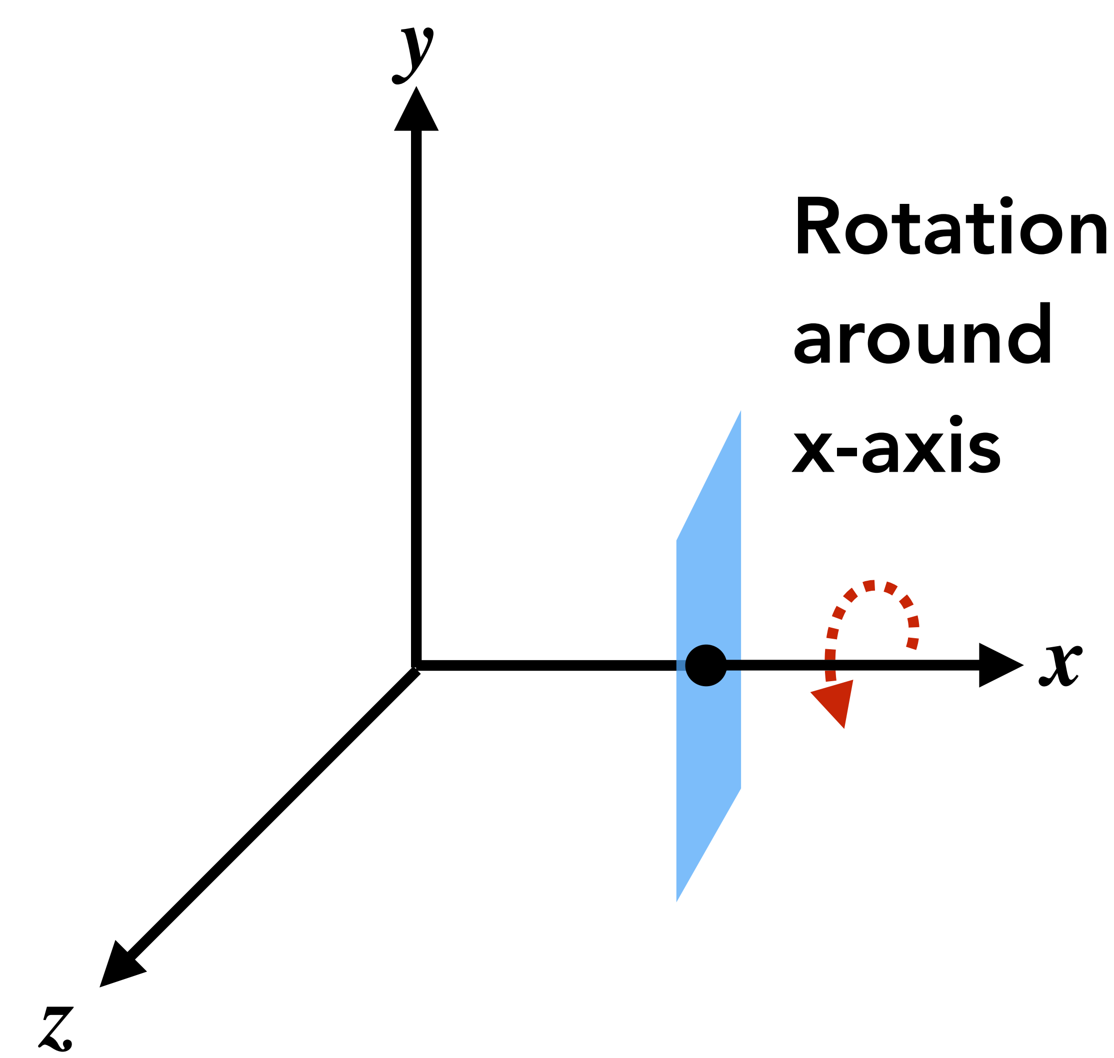
3D Transformations

Rotation around x-, y-, or z-axis

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

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

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

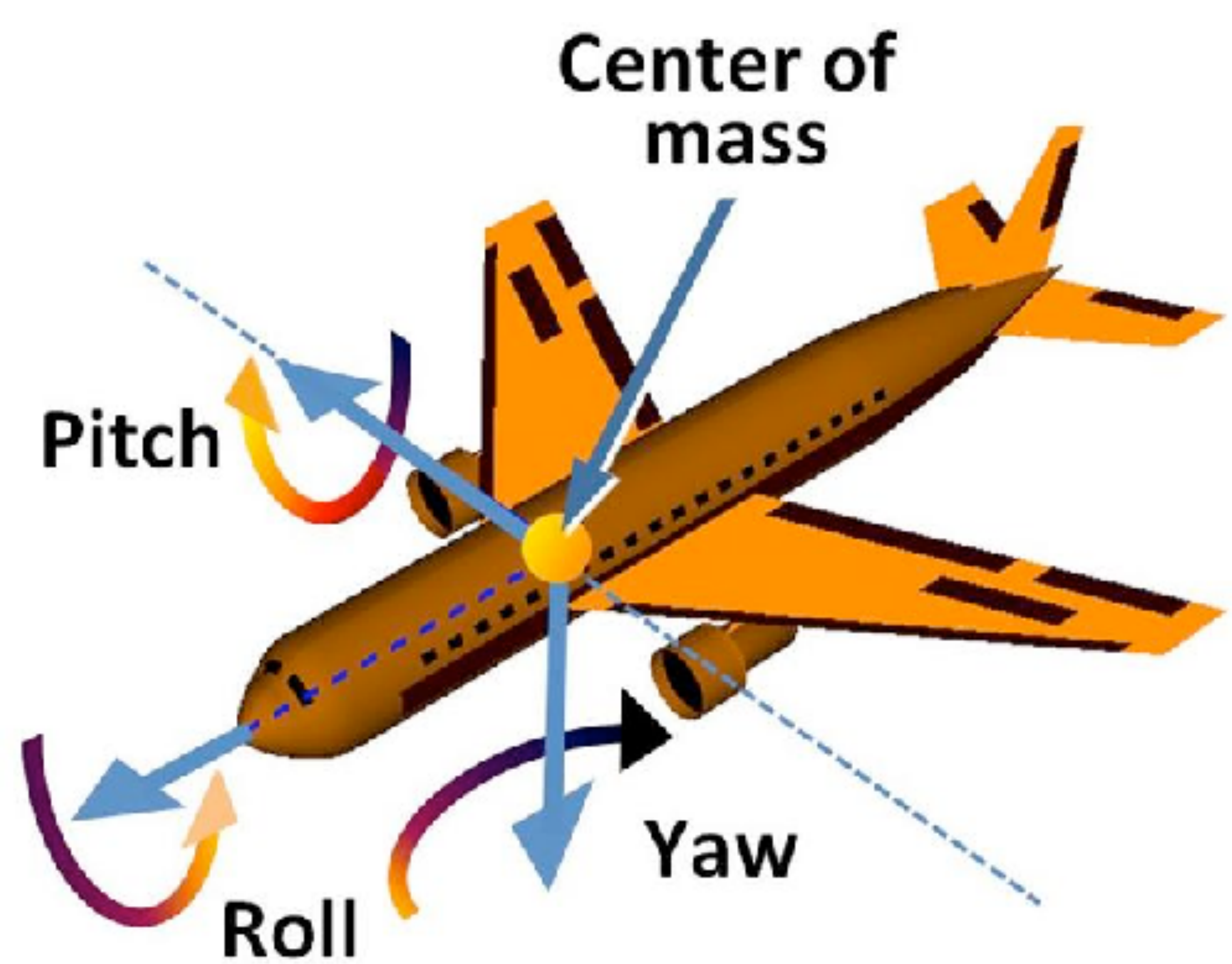


3D Rotations

Compose any 3D rotation from $\mathbf{R}_x, \mathbf{R}_y, \mathbf{R}_z$?

$$\mathbf{R}_{xyz}(\alpha, \beta, \gamma) = \mathbf{R}_x(\alpha)\,\mathbf{R}_y(\beta)\,\mathbf{R}_z(\gamma)$$

- So-called *Euler angles*
- Often used in flight simulators: roll, pitch, yaw



3D Rotations

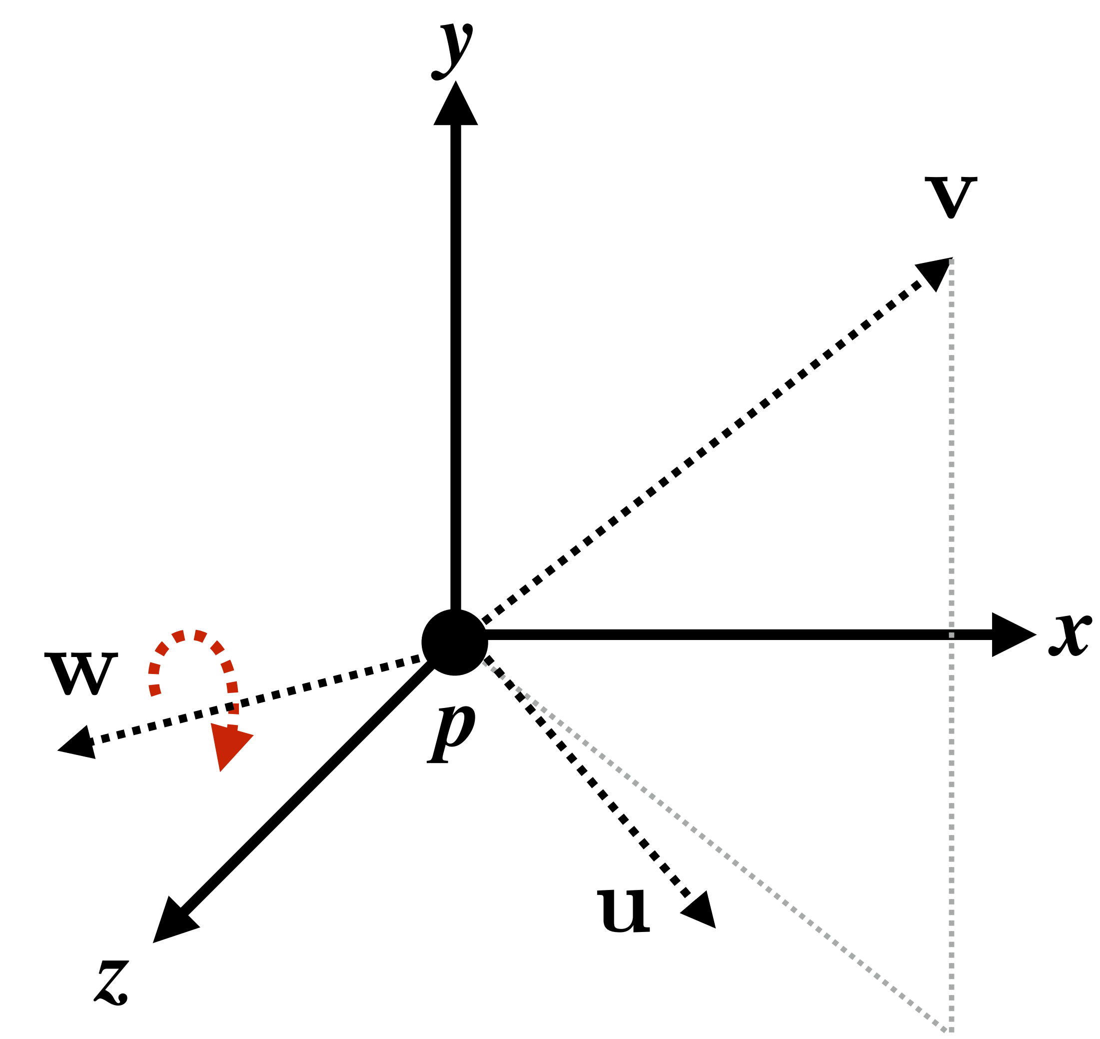
Compose any 3D rotation from $\mathbf{R}_x, \mathbf{R}_y, \mathbf{R}_z$?

$$\mathbf{R}_{xyz}(\alpha, \beta, \gamma) \;=\; \mathbf{R}_x(\alpha)\,\mathbf{R}_y(\beta)\,\mathbf{R}_z(\gamma)$$

- So-called *Euler angles*
- Often used in flight simulators: roll, pitch, yaw

3D Rotation Around Arbitrary Axis

Construct orthonormal frame transformation F with p, u, v, w, where p and w match the rotation axis



Apply the transform ($F\ R_z(\theta)\ F^{-1}$)

Interpretations (both valid):

- Move to Z axis, rotate, then move back
- Cast w-axis rotation in new coordinate frame

Rodrigues' Rotation Formula

Rotation by angle α around axis $\mathbf{n}$

$$\mathbf{R}(\mathbf{n}, \alpha) \;=\; \cos(\alpha)\,\mathbf{I} \;+\; (1-\cos(\alpha))\,\mathbf{n}\mathbf{n}^T \;+\; \sin(\alpha)\underbrace{\begin{pmatrix} 0 & -n_z & n_y \\ n_z & 0 & -n_x \\ -n_y & n_x & 0 \end{pmatrix}}_{\mathbf{N}}$$

How to prove this magic formula?

- Matrix $\mathbf{N}$ computes a cross-product: $\mathbf{N}\,\boldsymbol{x} = \mathbf{n} \times \boldsymbol{x}$
- Assume orthonormal system $\mathbf{e}_1, \mathbf{e}_2, \mathbf{n}$

$$\begin{aligned} \mathbf{R}\mathbf{n} &= \mathbf{n} \\ \mathbf{R}\mathbf{e}_1 &= \cos\alpha\,\mathbf{e}_1 + \sin\alpha\,\mathbf{e}_2 \\ \mathbf{R}\mathbf{e}_2 &= -\sin\alpha\,\mathbf{e}_1 + \cos\alpha\,\mathbf{e}_2 \end{aligned}$$