Image Registration: Geometric Transformation

Albert Chung

Department of Computer Science
University of Exeter

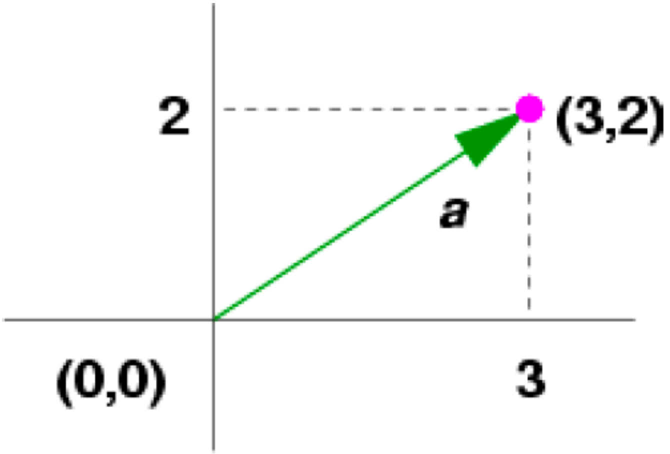
Spring Term 2023

Main Reference

- R. Szeliski, “Computer Vision: Algorithms and Applications”:
  - Chapter 3.6: (“Geometric transformations”)

Coordinates and points

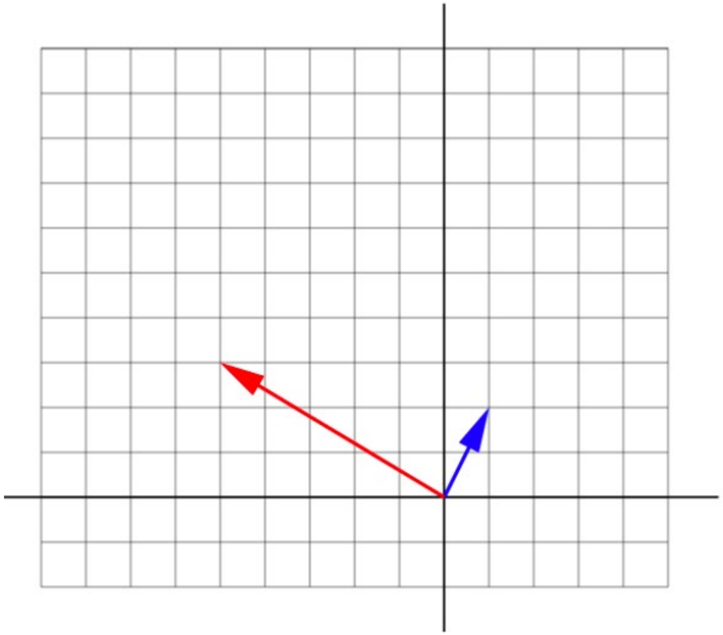
- Coordinate points:
  - $a = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$



Matrix transformations

Matrices: operators for transforming vectors

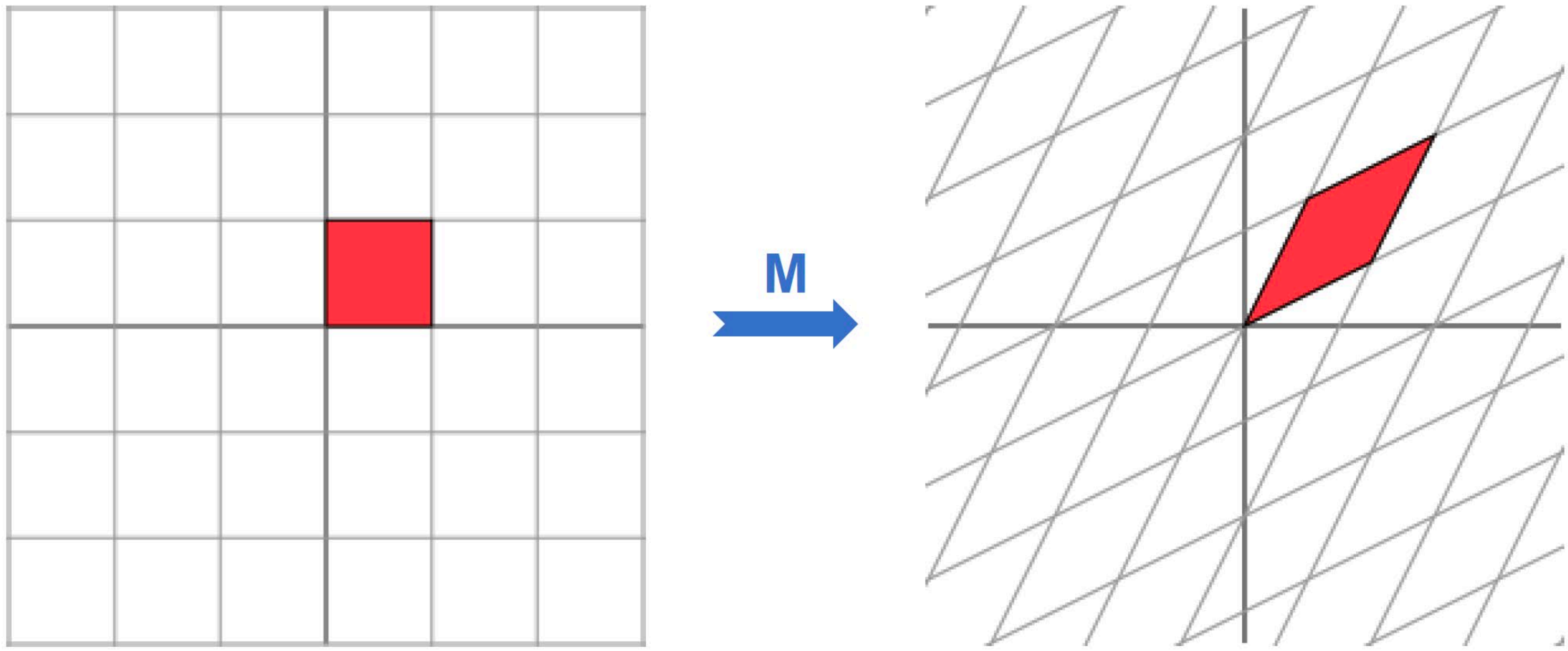
- $\begin{bmatrix} -5 \\ 3 \end{bmatrix} = \begin{bmatrix} 1 & -3 \\ -5 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix}$



- $\mathbf{x}' = M\mathbf{x}$
- $\mathbf{x}$ = local coordinates, $\mathbf{x}'$ = screen coordinates
- Tranformation done by a matrix product are called linear transformations.
- Examples: rotation, scaling, shear, mirroring.

Generic linear transformations

- General transformation form: $\mathbf{x}' = M\mathbf{x}$



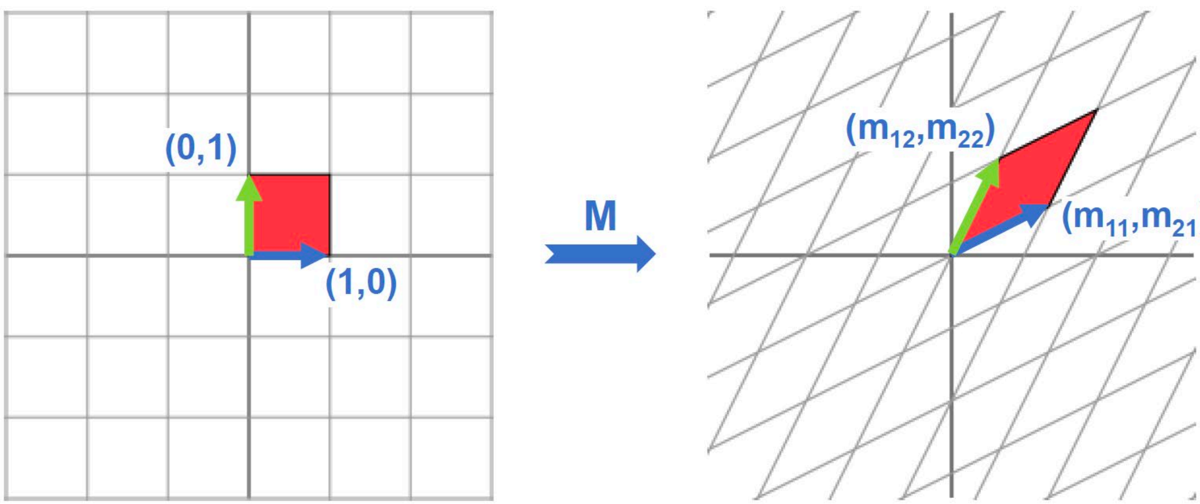
Generic linear transformations

- The columns of M describe the effect on the basis vectors:

$$\mathbf{x}' = M\mathbf{x}, \quad M = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix}$$

$$\text{If } \mathbf{x} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \text{ then: } \mathbf{x}' = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} m_{11} \\ m_{21} \end{bmatrix}$$

$$\text{If } \mathbf{x} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \text{ then: } \mathbf{x}' = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} m_{12} \\ m_{22} \end{bmatrix}$$



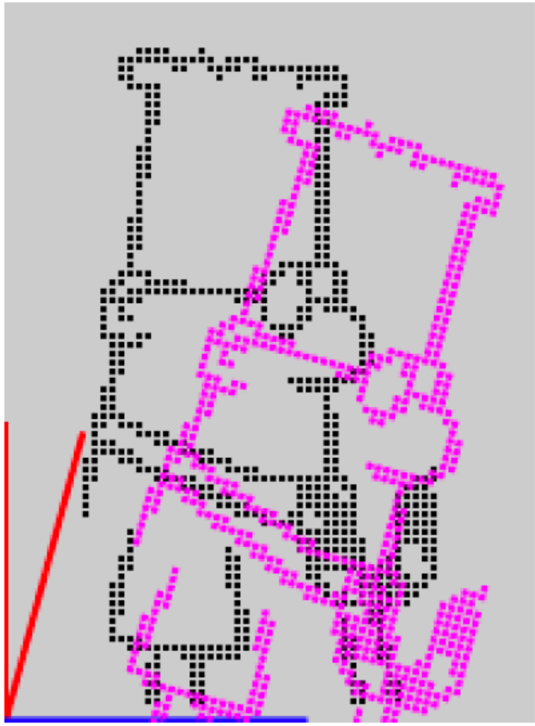
Rotation

- Rotation by θ around the origin of the coordinate system.

$$\mathbf{x}' = R\mathbf{x}$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$= \begin{bmatrix} x\cos\theta - y\sin\theta \\ x\sin\theta + y\cos\theta \end{bmatrix}$$

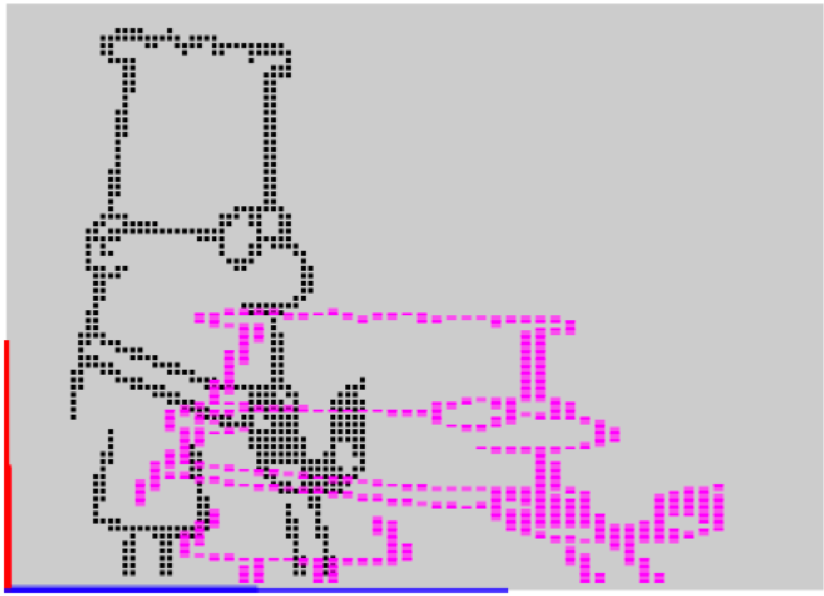


- Rotation matrices leave the length of the vector unchanged

$$|\mathbf{x}'|^2$$
$$= (x\cos\theta - y\sin\theta)^2 + (x\sin\theta + y\cos\theta)^2$$
$$= x^2 + y^2 = \mathbf{x}^2$$

Scaling

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \lambda_1 x \\ \lambda_2 y \end{bmatrix} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

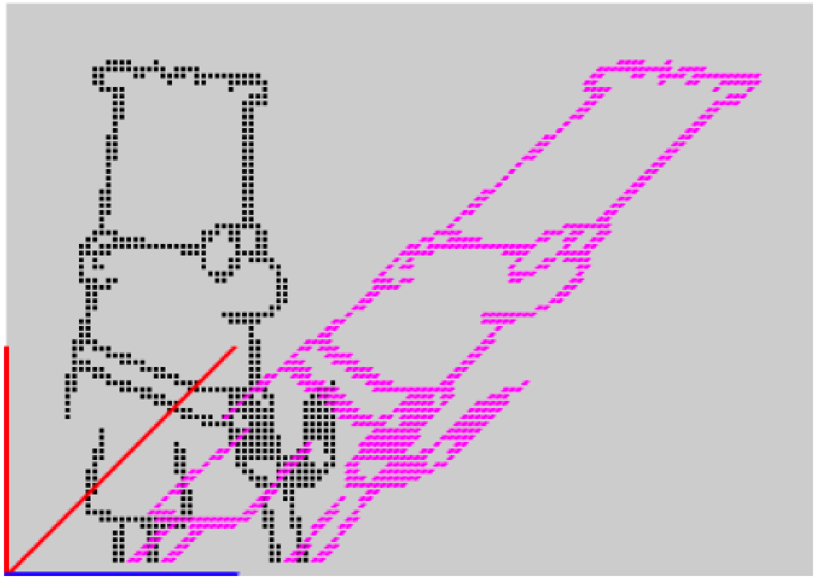


- Diagonal matrices scale the vector
- Equal entries on diagonal give *isotropic* scaling
- Unequal entries on diagonal give *anisotropic* scaling

Shear

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x + \lambda y \\ y \end{bmatrix} = \begin{bmatrix} 1 & \lambda \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

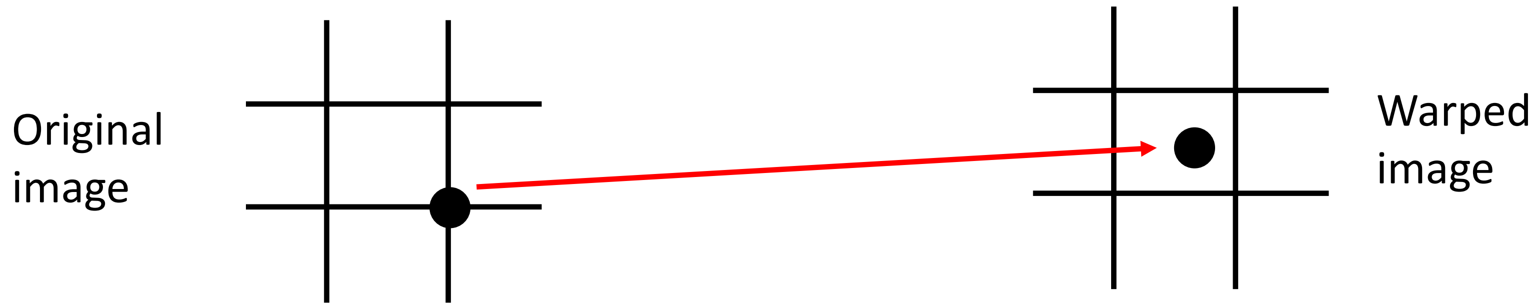
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x \\ \mu x + y \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \mu & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



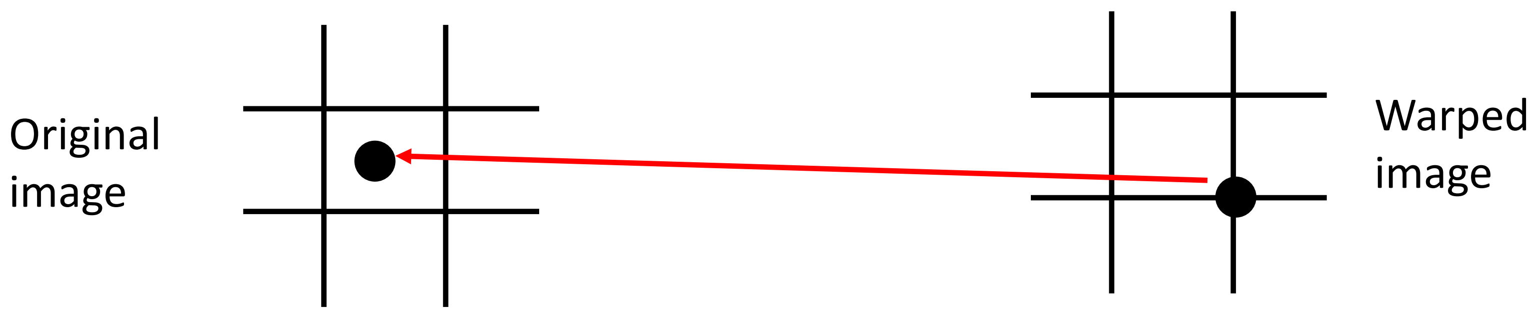
- One coordinate is left unchanged
- Dilbert was sheared with $\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$

Aliasing and inverse warping

- Aliasing occurs when the coordinate values in the warped image are rounded to the nearest integer. It creates intensity discontinuity, and cracks/holes.



- Inverse warping: Each pixel in the warped image is computed from the original image by using bilinear interpolation.



Example of Combining Linear Transformations

- To rotate $\boldsymbol{x}$ with R and then shear the result with S:

$$\mathbf{y} = R\mathbf{x} \qquad \text{Rotate}$$
$$\mathbf{z} = S\mathbf{y} \qquad \text{Shear}$$

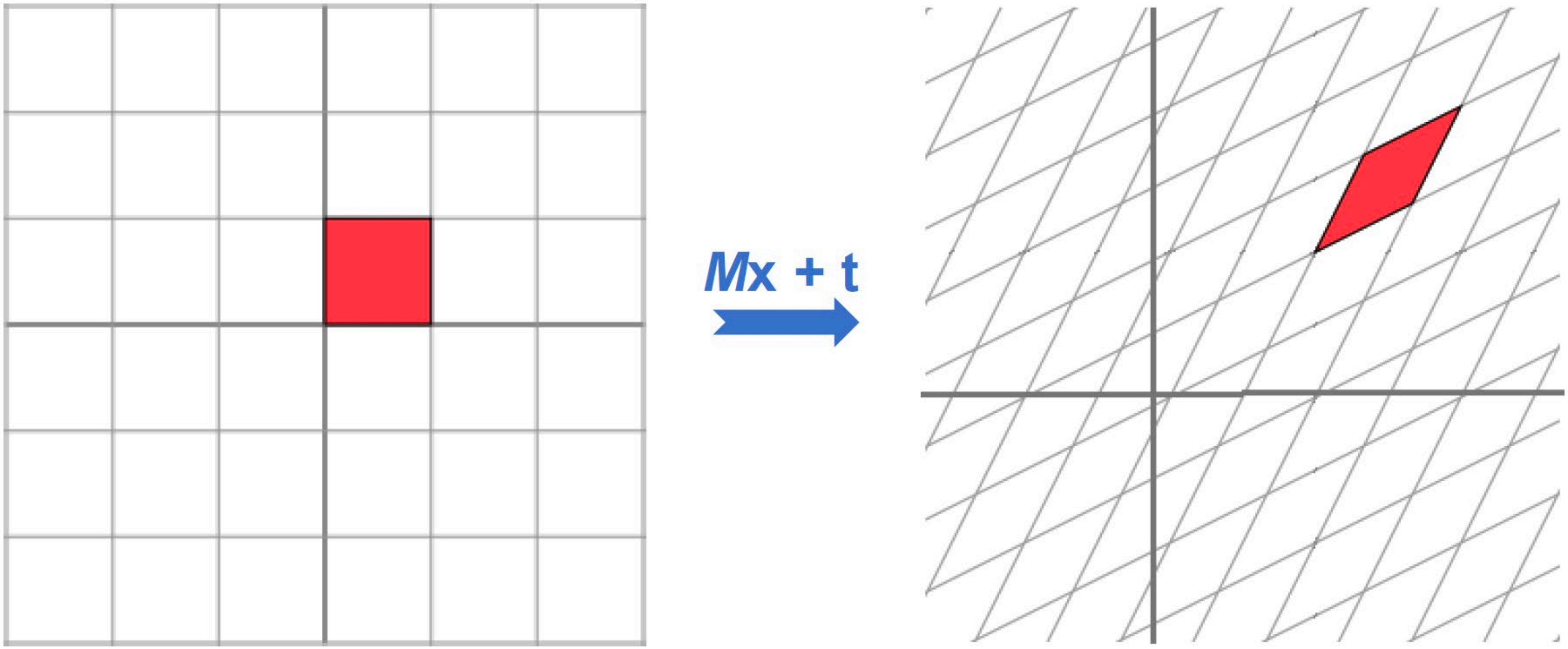
- Combine the operations into one:

$$\mathbf{z} = A\mathbf{x}$$
$$\mathbf{z} = S\mathbf{y} = S(R\mathbf{x}) = SR\mathbf{x}$$

- The matrix product $A = SR$ is defined so that this composition `works'.
- $A = SR$, the (i, j)th element of the multiplication of SR is the dot product of the ith row of S and the jth column of R.

Affine transformations

- General transformation form: $\mathbf{x}' = M\mathbf{x} + \mathbf{t}$



Affine transformations

- General transformation: $\mathbf{x}' = M\mathbf{x} + \mathbf{t}$

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

- Map straight lines into straight lines
- Map parallel lines into parallel lines
- Comprise all combinations of scaling, rotations, shears and translations

Combining affine transformations

- Let us combine 2 affine transformations $(M_1, \mathbf{t}_1)$ and $(M_2, \mathbf{t}_2)$:

$$\begin{cases} \mathbf{x}' = M_1\mathbf{x} + \mathbf{t}_1 \\ \mathbf{x}'' = M_2\mathbf{x}' + \mathbf{t}_2 \end{cases}$$

Then

$$\mathbf{x}'' = M_2(M_1\mathbf{x} + \mathbf{t}_1) + \mathbf{t}_2 = M_2 M_1\mathbf{x} + (M_2\mathbf{t}_1 + \mathbf{t}_2)$$

- affine transformation $(M_{tot}, \mathbf{t}_{tot})$ where:

$$M_{tot} = M_2 M_1, \ \mathbf{t}_{tot} = M_2\mathbf{t}_1 + \mathbf{t}_2$$

Combining affine transformations

- Let us now combine 3 affine transformations:

$$(M_1, \mathbf{t}_1), \ (M_2, \mathbf{t}_2) \text{ and } (M_3, \mathbf{t}_3)$$

- affine transformation $(M_{tot}, \mathbf{t}_{tot})$ where:

$$M_{tot} = M_3 M_2 M_1, \ \mathbf{t}_{tot} = M_3(M_2\mathbf{t}_1 + \mathbf{t}_2) + \mathbf{t}_3$$

Combining affine transformations

- Let us now combine 3 affine transformations:

$$(M_1, \mathbf{t}_1), \ (M_2, \mathbf{t}_2) \text{ and } (M_3, \mathbf{t}_3)$$

- affine transformation $(M_{tot}, \mathbf{t}_{tot})$ where:

$$M_{tot} = M_3 M_2 M_1, \ \mathbf{t}_{tot} = M_3(M_2\mathbf{t}_1 + \mathbf{t}_2) + \mathbf{t}_3$$

- too much complicated!

Homogeneous coordinates

- **Solution**: represent (x, y) as $(x, y, 1)$. Then transformation

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

becomes

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} m_{11} & m_{12} & t_x \\ m_{21} & m_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}}_{\hat{M}} \underbrace{\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}}_{\hat{\mathbf{x}}}$$

Homogeneous coordinates

- **Solution**: represent (x, y) as $(x, y, 1)$. Then transformation

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

becomes

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} m_{11} & m_{12} & t_x \\ m_{21} & m_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}}_{\hat{M}} \underbrace{\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}}_{\hat{\mathbf{x}}}$$

- Uniform representation of all affine transformations as a single matrix multiplication ($\widehat{M}$, $\hat{\mathbf{x}}$)
- Easy to combine transformations
  - e.g. combination of 3 affine transformations is simply: $\widehat{M}_1 \widehat{M}_2 \widehat{M}_3$
- Natural extension to include perspective transformations
- Matrix multiplications are fast

Homogeneous coordinates

Rotation

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

Scaling

$$\begin{bmatrix} \lambda_x & 0 & 0 \\ 0 & \lambda_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Shear

$$\begin{bmatrix} 1 & \lambda & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & 0 \\ \mu & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Translation

$$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

Example of combining affine transformations

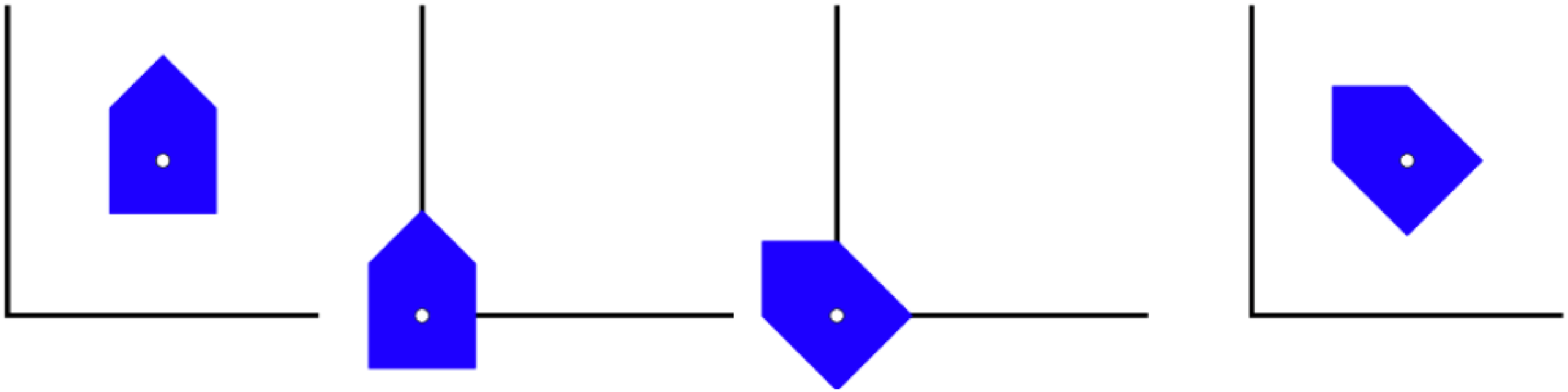
- Rotation followed by translation:

$$\mathbf{x}' = R\mathbf{x} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \qquad \mathbf{x}'' = T\mathbf{x}' = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix}$$

$$\mathbf{x}'' = TR\,\mathbf{x} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x}$$

$$\mathbf{x}'' = \begin{bmatrix} \cos\theta & -\sin\theta & t_x \\ \sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x}$$

Rotation about a point which is not origin

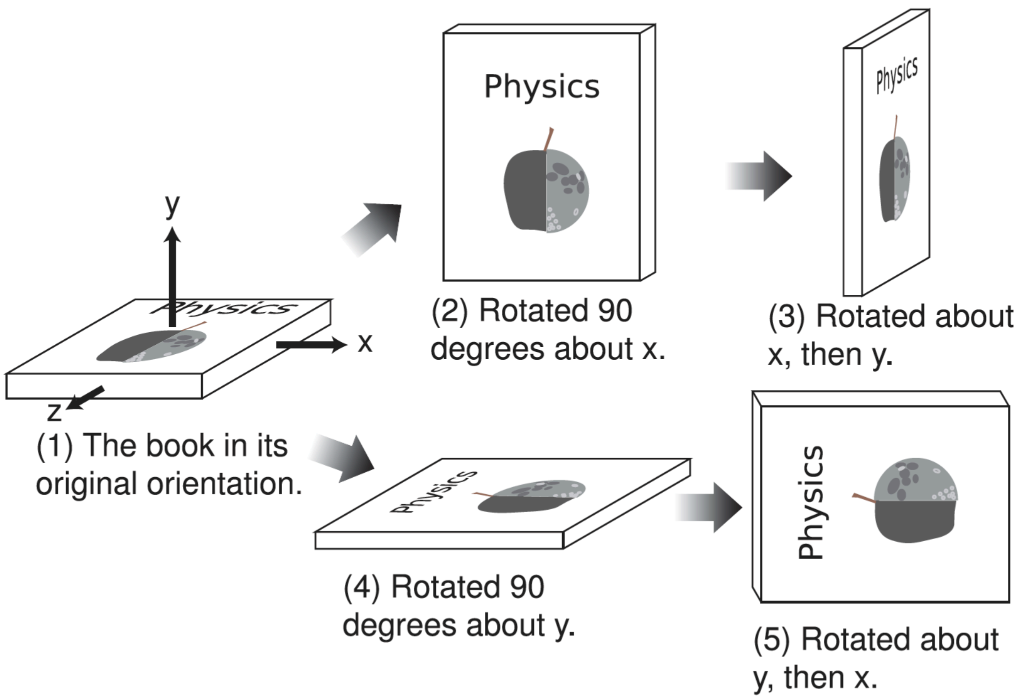


- Translate object back to the origin T — $x' = Tx$
- Rotate around the origin, R — $x'' = RTx$
- Translate back to original location, T^{-1} — $x''' = T^{-1}RTx$

$$x' = T^{-1}RTx$$

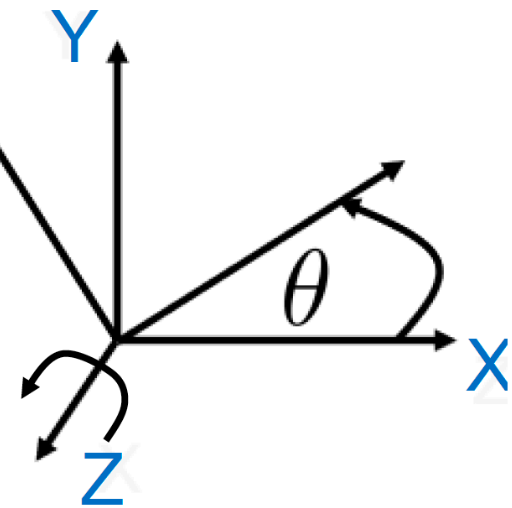
3D coordinates

- **scaling, translation**: straight-forward extension of the 2D case
- **rotation**: things become more complicated:
  - more complex expressions for general 3D rotations
  - unlike 2D, rotations in 3D **do not commute with each other**:



3D Rotations

- Rotation by θ around the **z-axis** of the coordinate system.



$$\mathbf{x}' = R\mathbf{x}$$

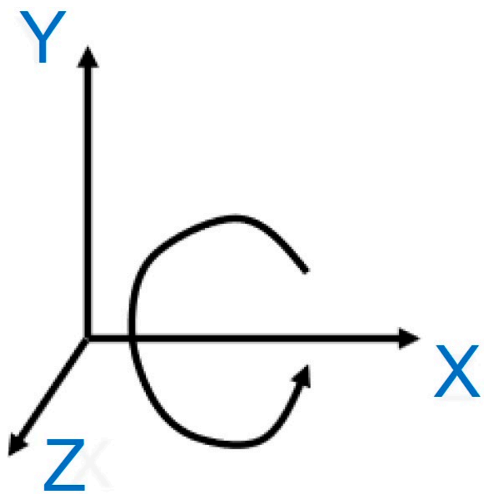
$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

3D Rotations

- Rotation by θ around the **x-axis** of the coordinate system.

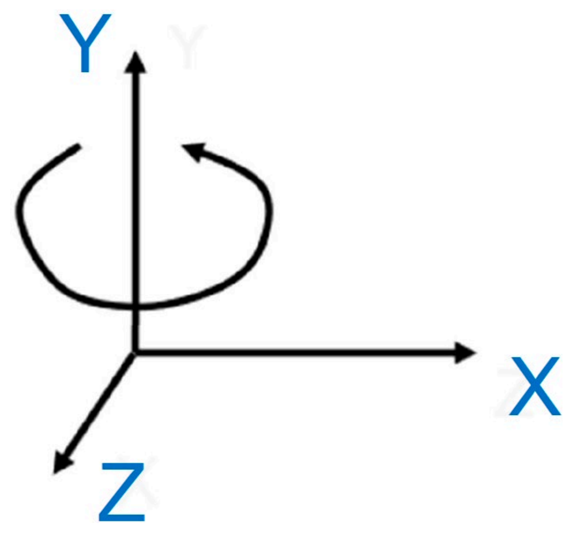
$$\mathbf{x}' = R\mathbf{x}$$

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$



3D Rotations

- Rotation by θ around the **y-axis** of the coordinate system.



$$\mathbf{x}' = R\mathbf{x}$$

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Rotation inverse

What is the inverse of a rotation?

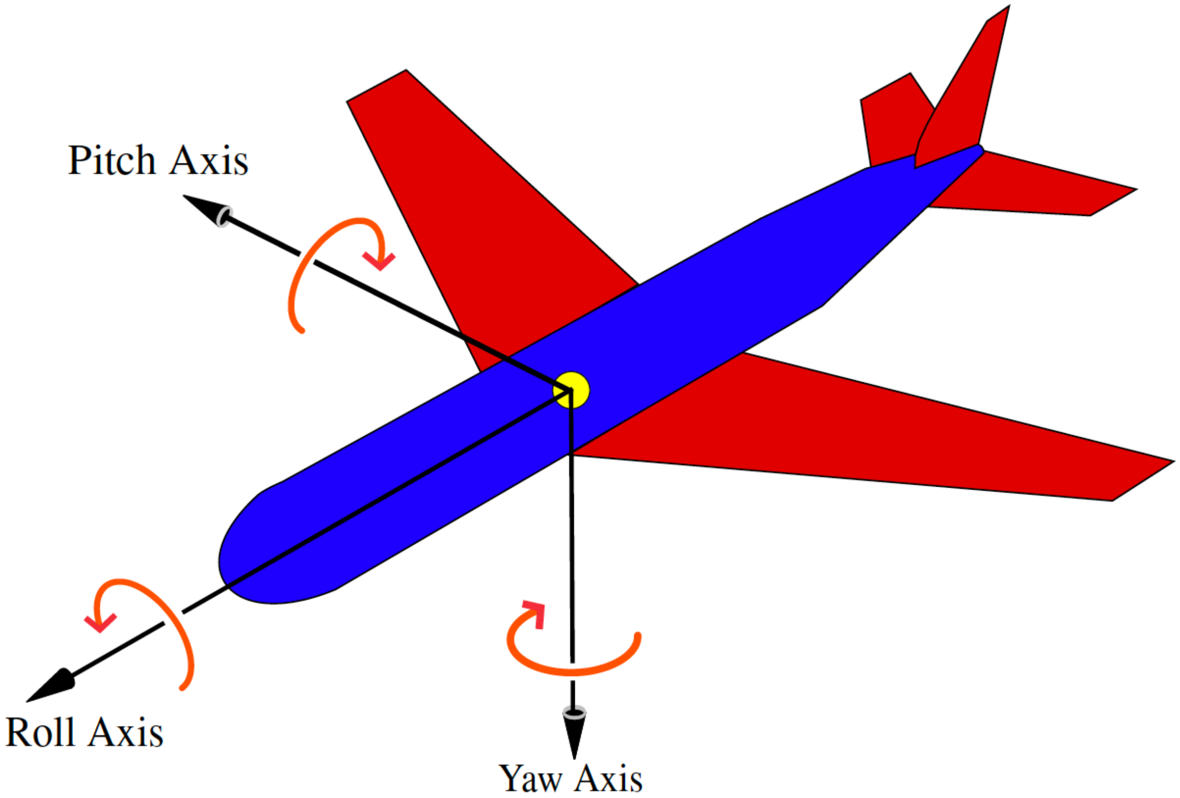
- mathematically, the inverse of a rotation matrix R is given by R^{-1}
- geometrically, the inverse of a rotation by θ is a rotation around the same axis by an angle of $-\theta$
- in practice, because rotation matrices are **orthogonal matrices** the inverse is given by the transpose $R^{-1} = R^T$
- **Note**: The transpose is given by flipping the matrix around the diagonal.

Rotation around an arbitrary axis

- We have seen how to rotate around the three main axes (**x**, **y** and **z**) of the coordinate system. How can we rotate around an arbitrary axis (say, a vector v)?

Rotation around an arbitrary axis

- **Euler angles**: any 3D rotation can be expressed as a sequence of 3 rotations around the main axes:

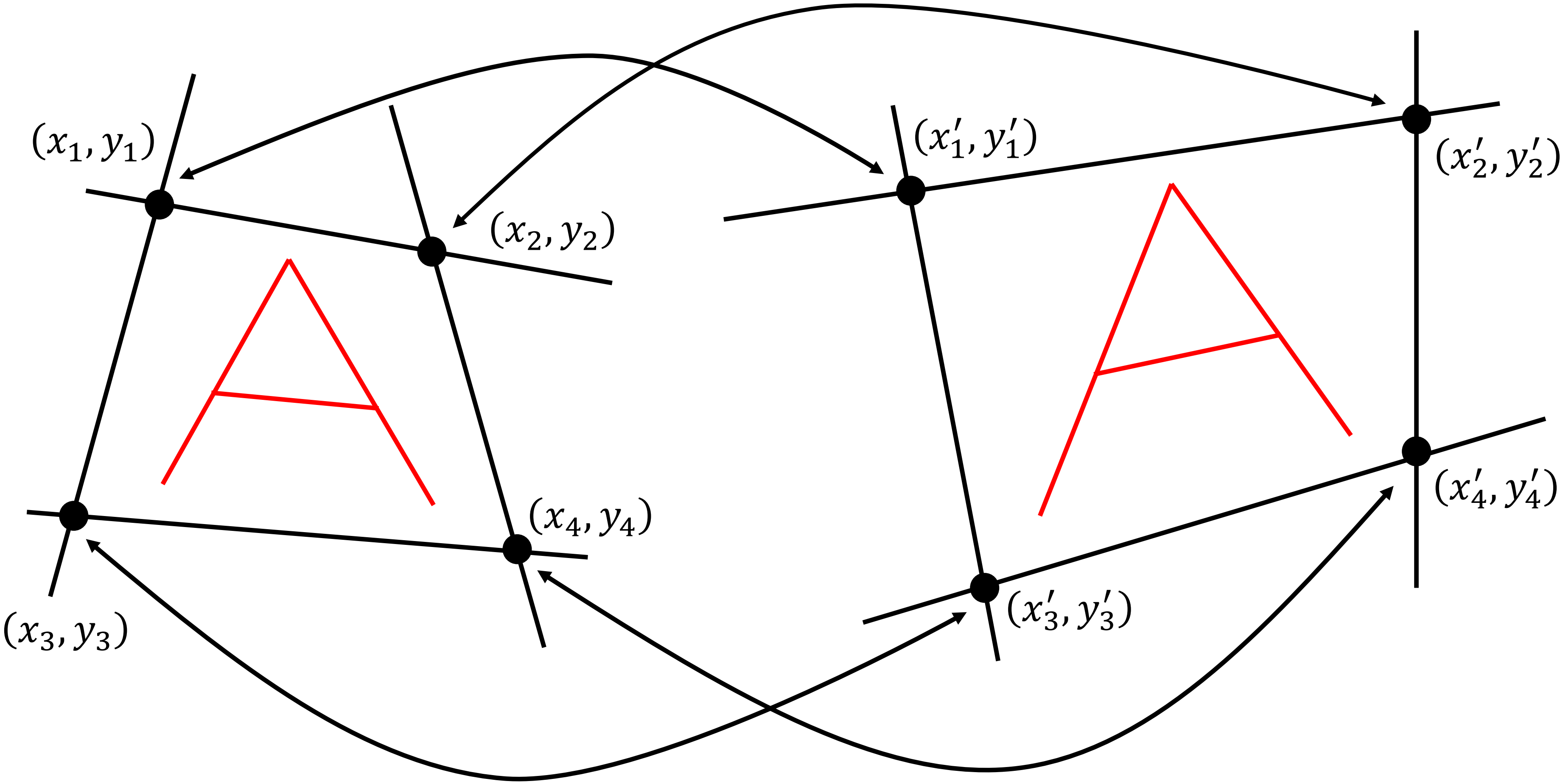


Rotation around an arbitrary axis

Can be obtained through the following combination of rotations:

1. Perform a first rotation to align v to one of the coordinate system's axes (x, y or z).
2. perform rotation around this axis.
3. perform the inverse of step (1).

Distortion Correction



$$x = a_1 x' + a_2 y' + a_3 x' y' + a_4$$
$$y = a_5 x' + a_6 y' + a_7 x' y' + a_8$$

$$x' = c_1 x + c_2 y + c_3 xy + c_4$$
$$y' = c_5 x + c_6 y + c_7 xy + c_8$$

The coefficients of the bilinear equations can be found by using four pairs of correspondence points.

Summary

- Linear transformations: rotation, scaling, shear, mirror
- Affine transformations: linear transformation + translation
- Combination of affine transformations
- Homogeneous coordinates
- Transformation in 3D coordinates