Insper

Computer Vision

Class 10: Position Models as Geometric Transformations

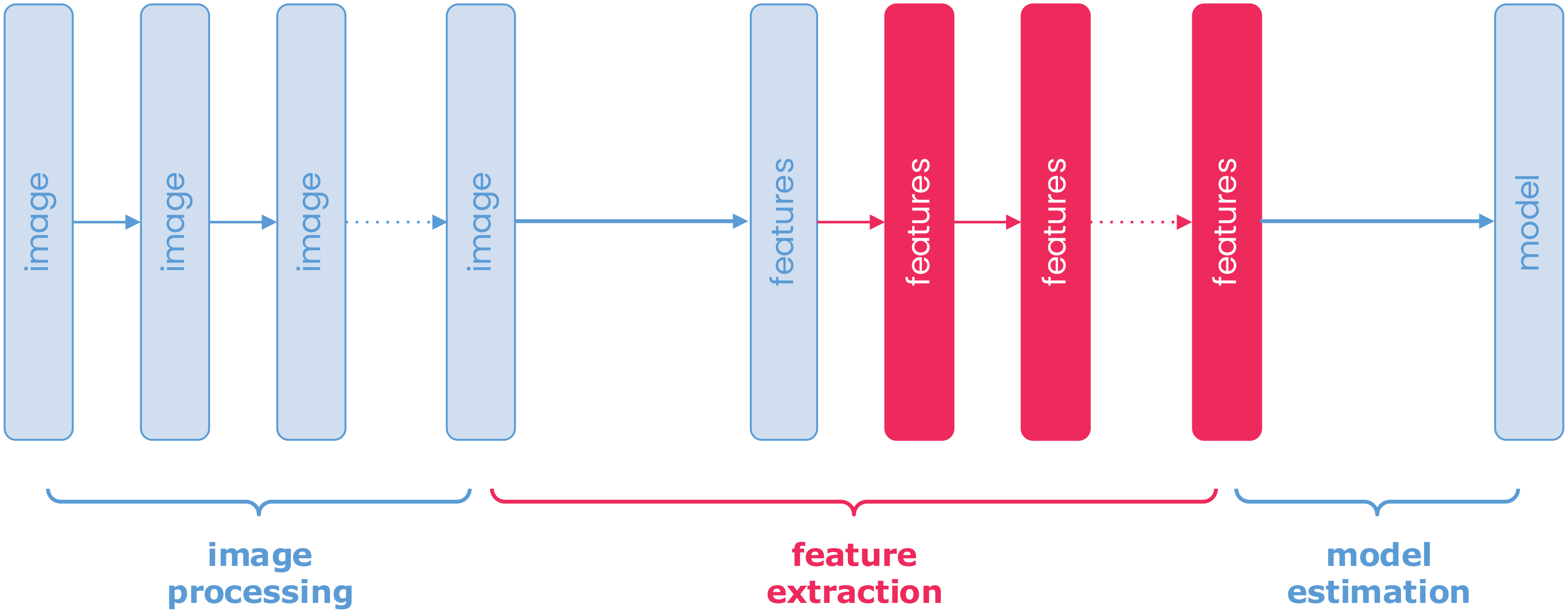
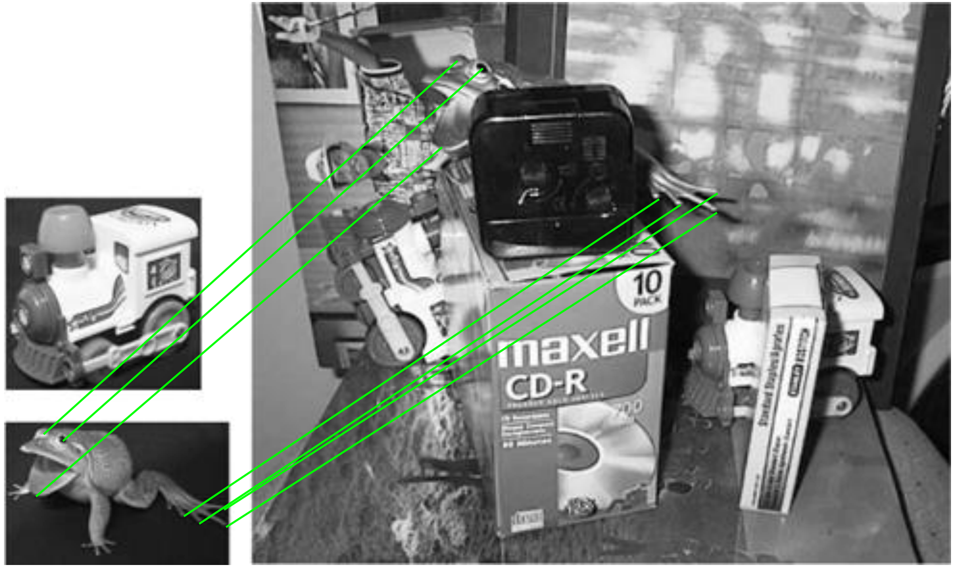


image
image
image
image
features
features
features
features
model
image processing
feature extraction
model estimation

The keypoint matching framework

infer the position of the object from matches of keypoints *(for example, corners)*



The object detection problem

given an image of an object and a scene, detect the position of the object in the scene

The keypoint matching framework

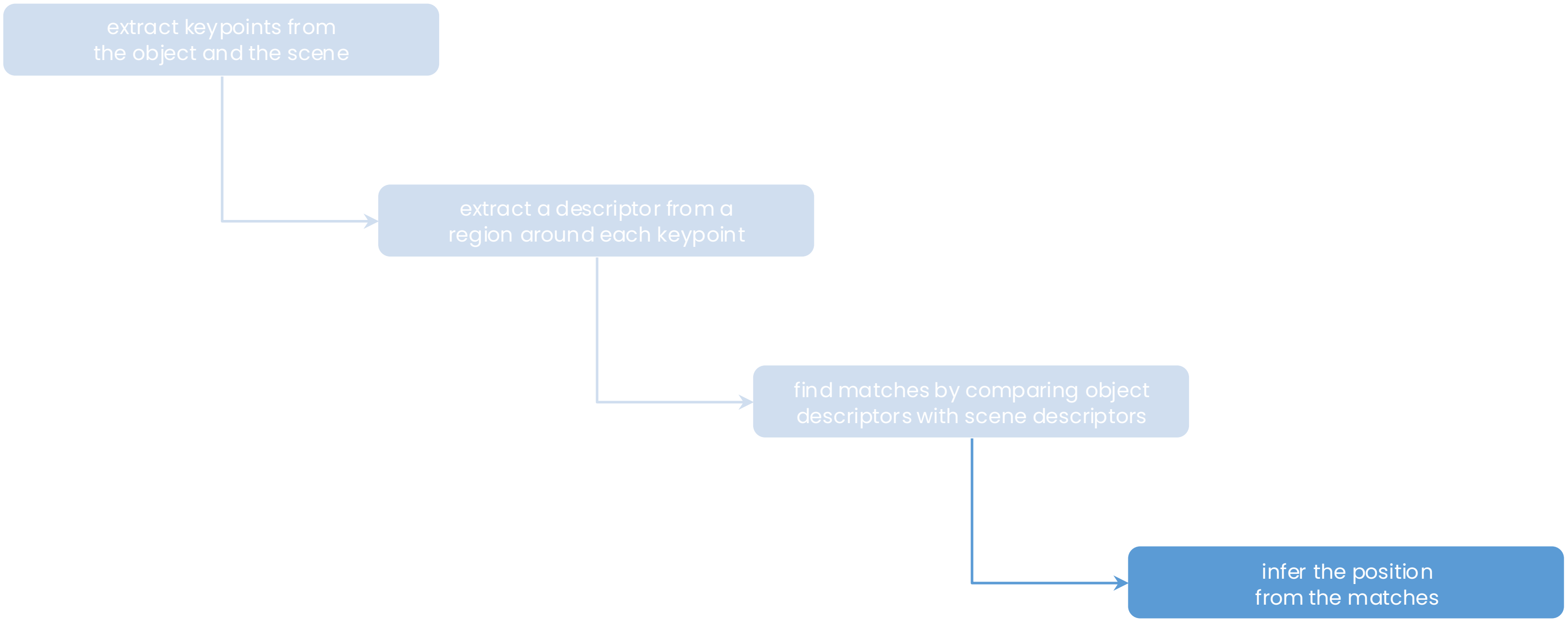
extract keypoints from
the object and the scene

extract a descriptor from a
region around each keypoint

find matches by comparing object
descriptors with scene descriptors

infer the position
from the matches

The keypoint matching framework



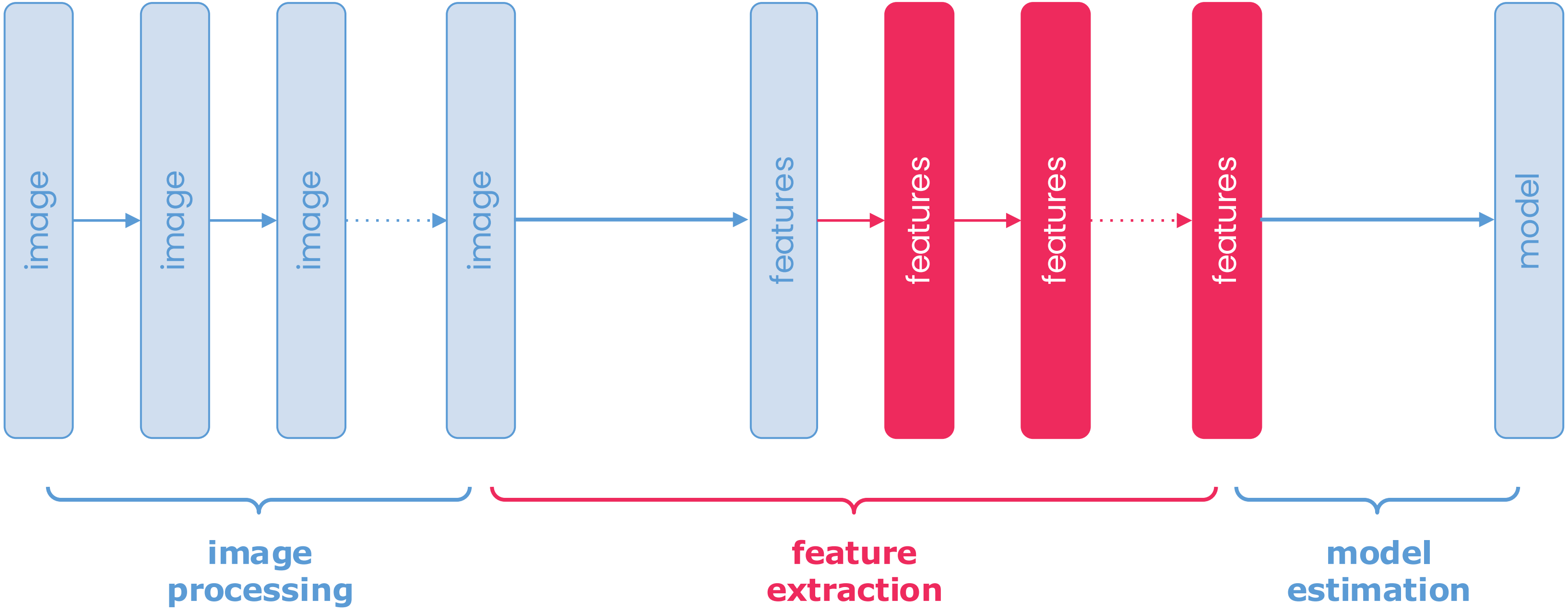


image
image
image
image
features
features
features
features
model
image processing
feature extraction
model estimation

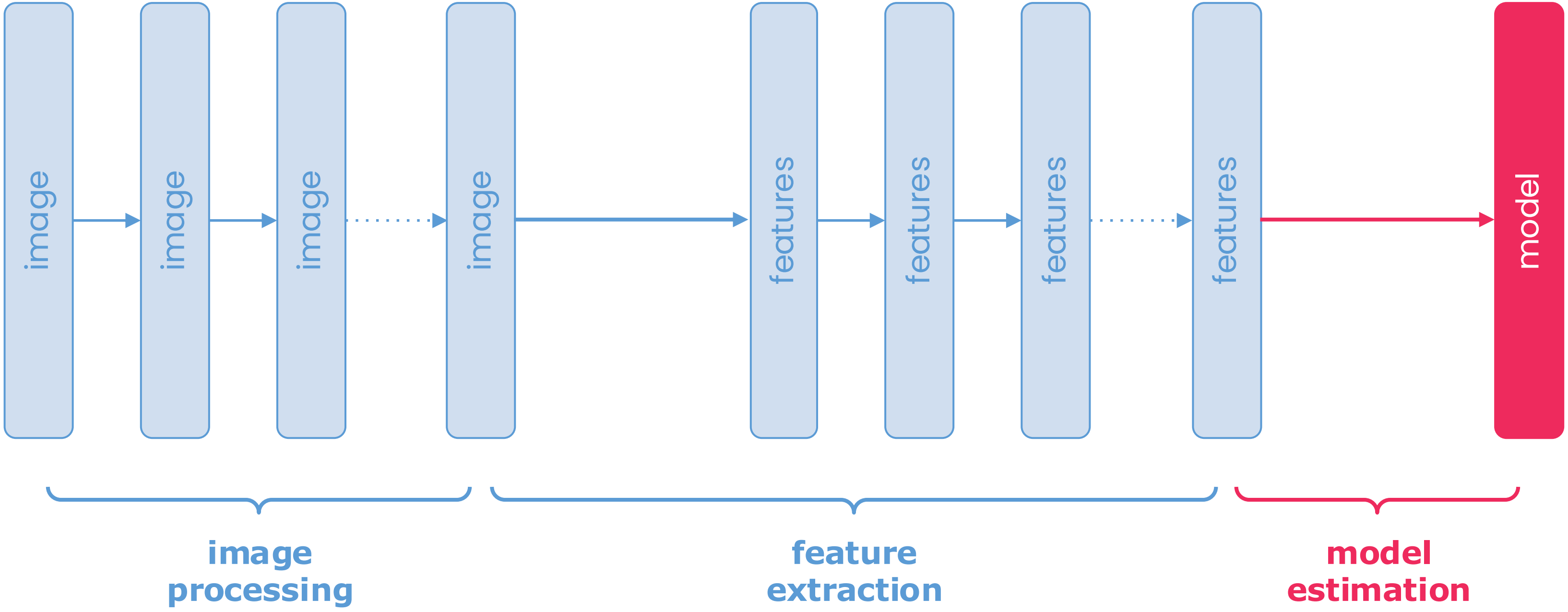


image
image
image
image
features
features
features
features
model
image processing
feature extraction
model estimation

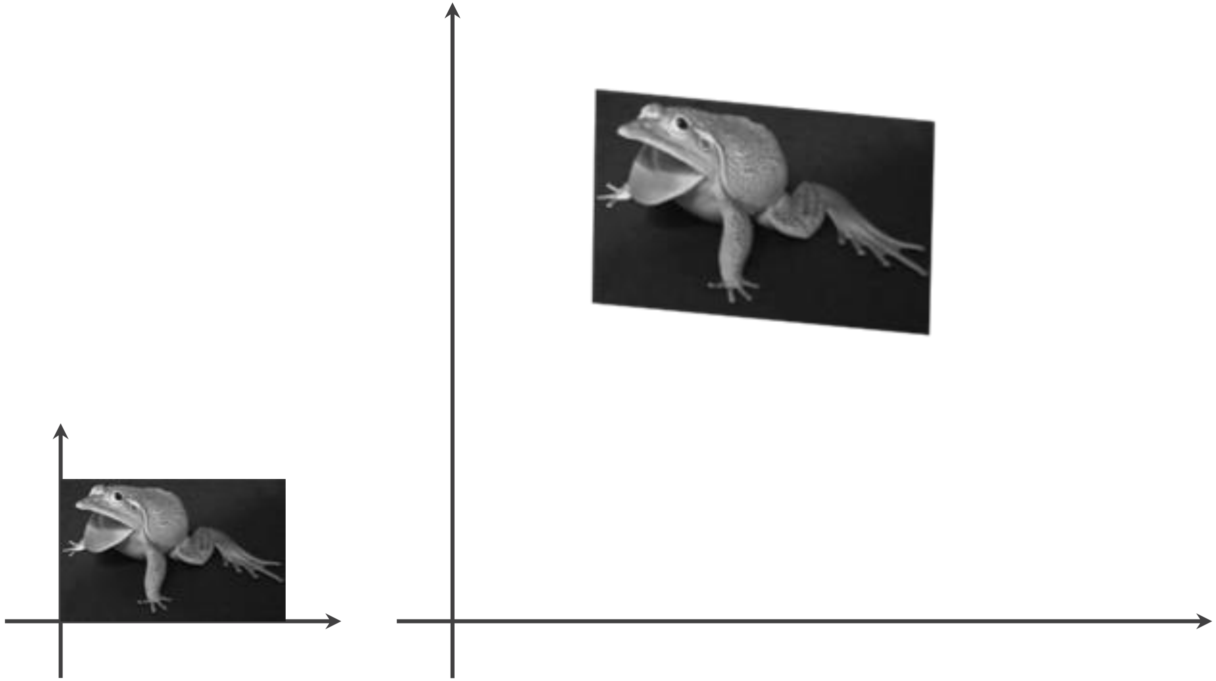
To infer the position, we first need to model the concept of position.

The object detection problem

given an image of an object and a scene, detect the position of the object in the scene

The object position model

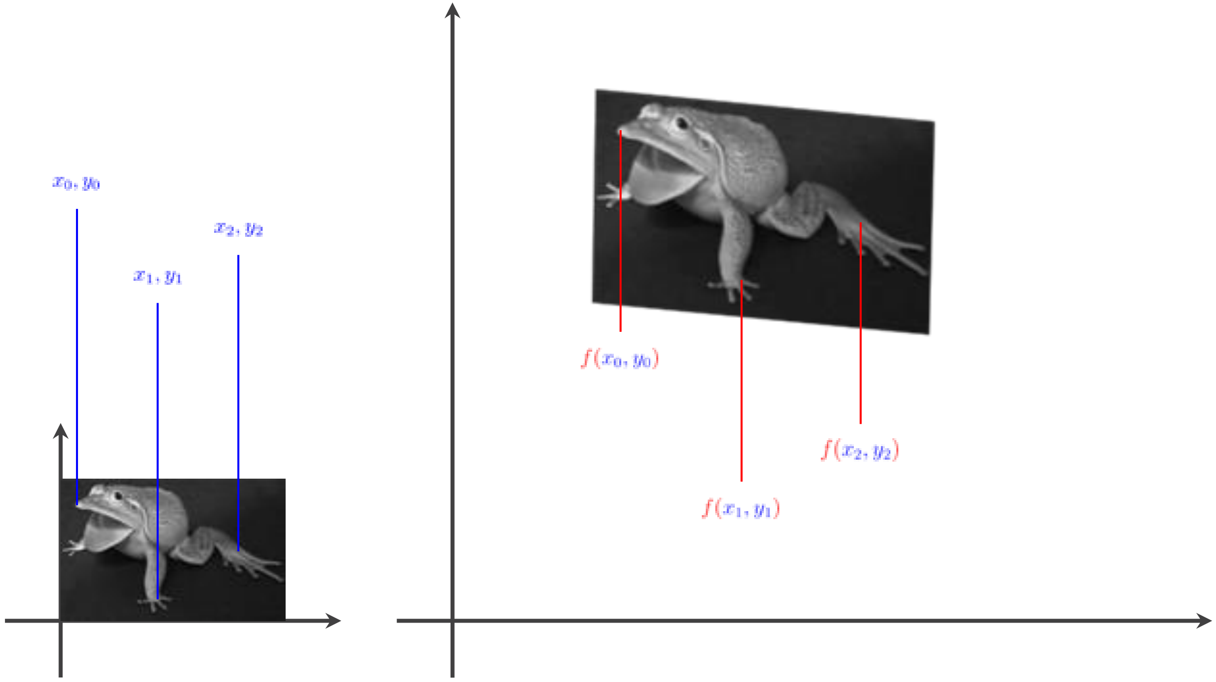
a geometric transformation that maps object coordinates to scene coordinates



The object position model

a geometric transformation that maps object coordinates to scene coordinates

$f : \mathbb{R}^2 \rightarrow \mathbb{R}^2$



A geometric transformation can…

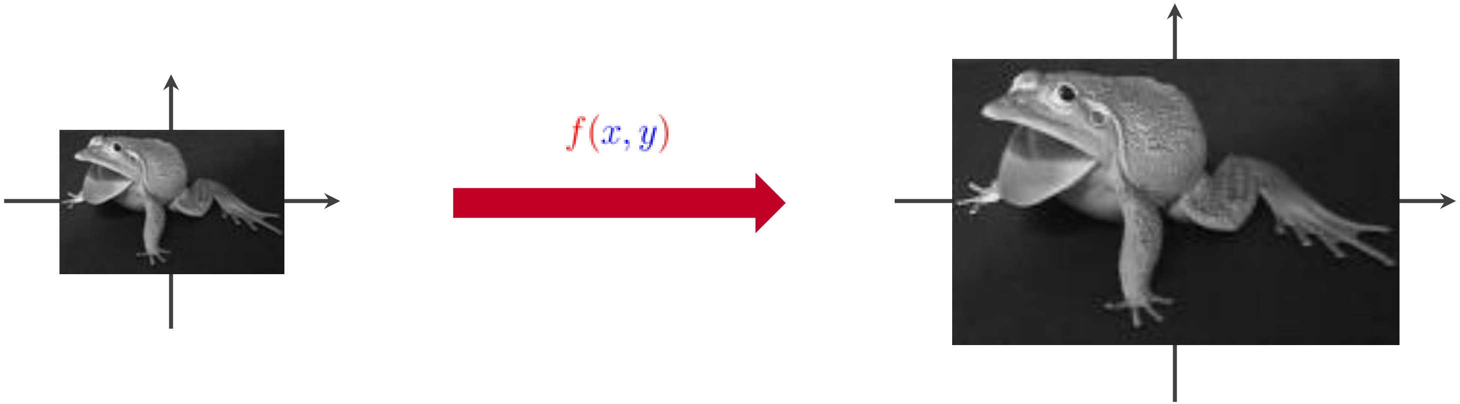
- …scale.
- …rotate.
- …translate.
- …warp.

For simplicity, we will analyse each one of the four changes separately.

(and will conclude that this separation is not relevant)

Scale transformation

scale the object by a given factor

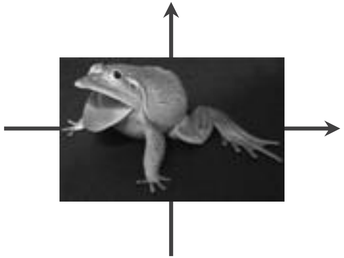


For simplicity, we will assume the object center is the origin.

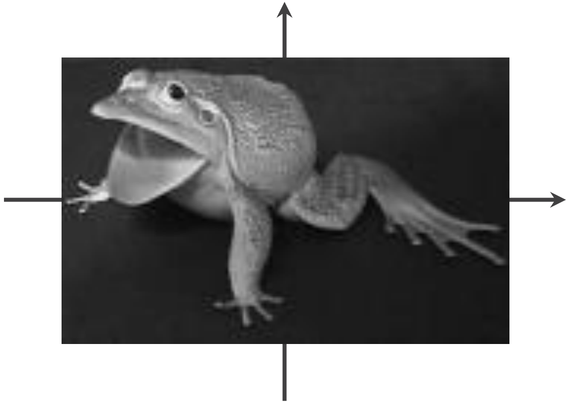
(and will conclude that this assumption is not relevant)

Scale transformation

scale the object by a given factor

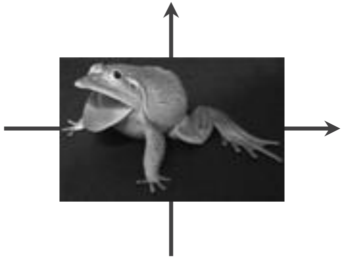


$2x + 2y$

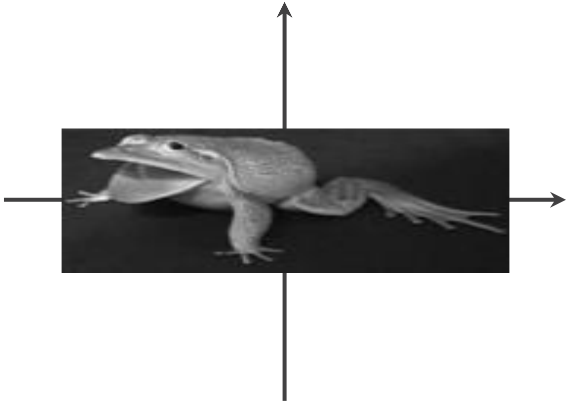


Scale transformation

scale the object by given factors

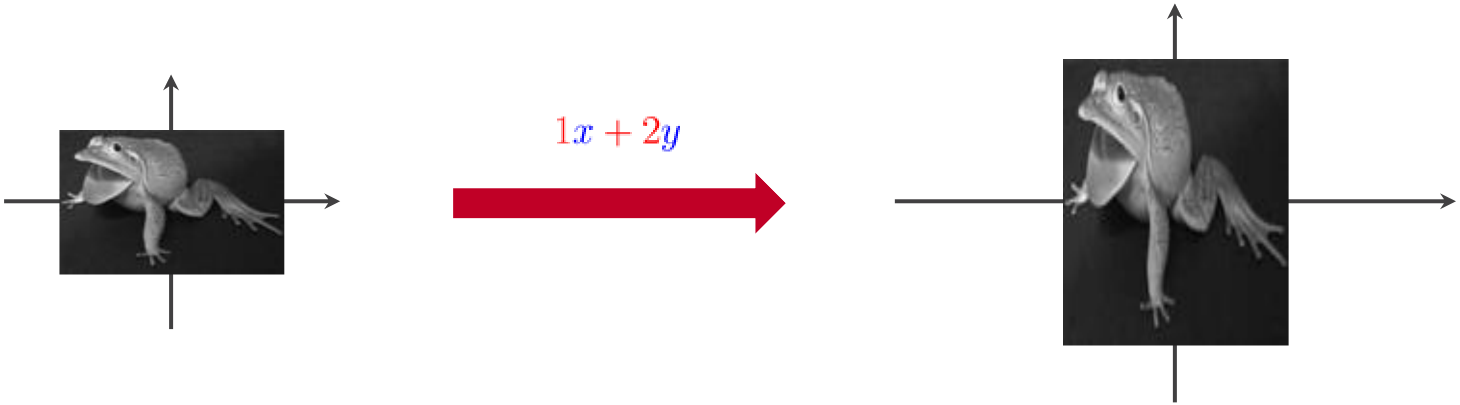


$2x + 1y$



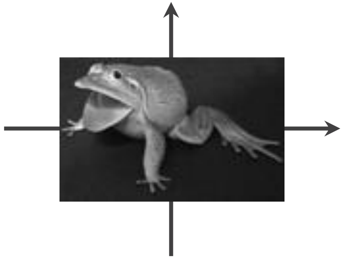
Scale transformation

scale the object by given factors

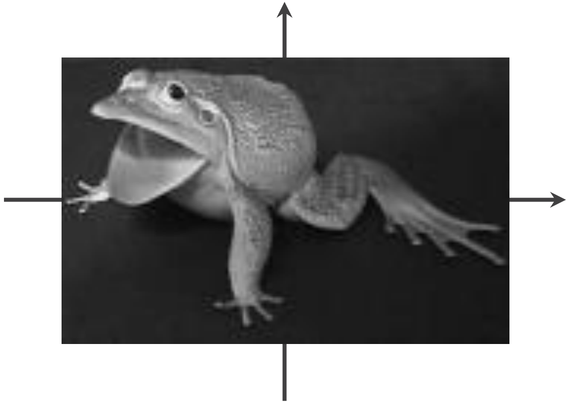


Scale transformation

scale the object by given factors

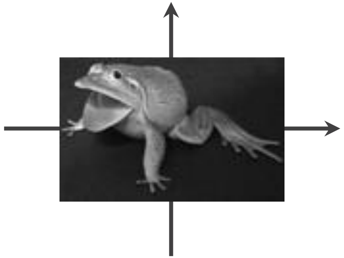


$$s_x x + s_y y$$

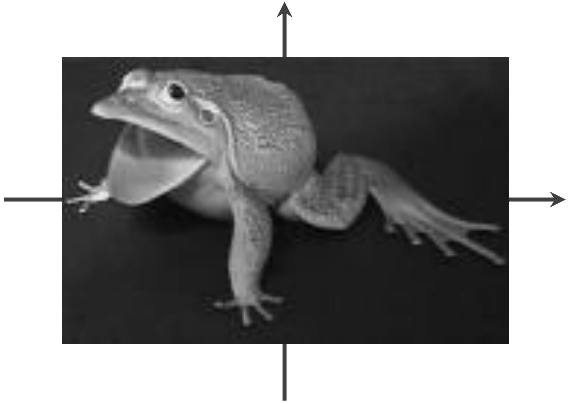


Scale transformation

scale the object by given factors



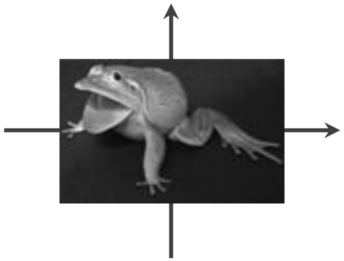
$$\begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



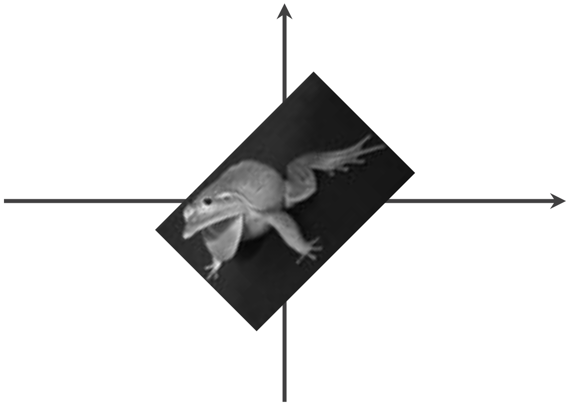
Maybe the other effects are also matrix multiplications?

Rotation transformation

rotate the object by a given angle

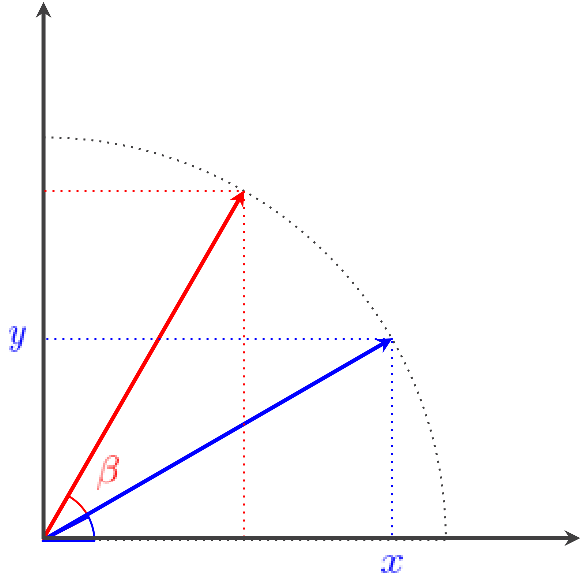


$$\begin{bmatrix} ? & ? \\ ? & ? \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



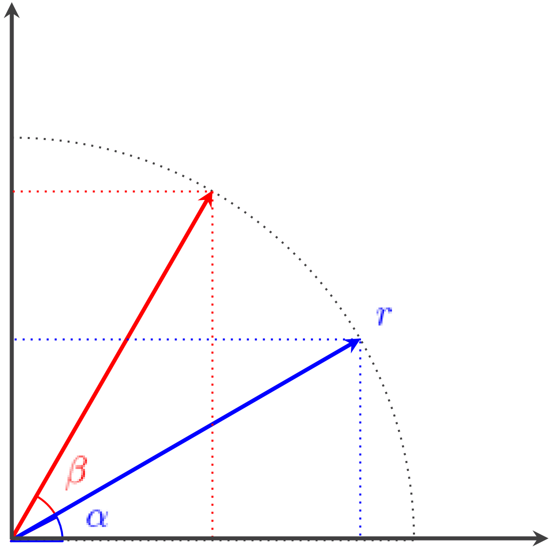
Rotation transformation

rotate the object by a given angle



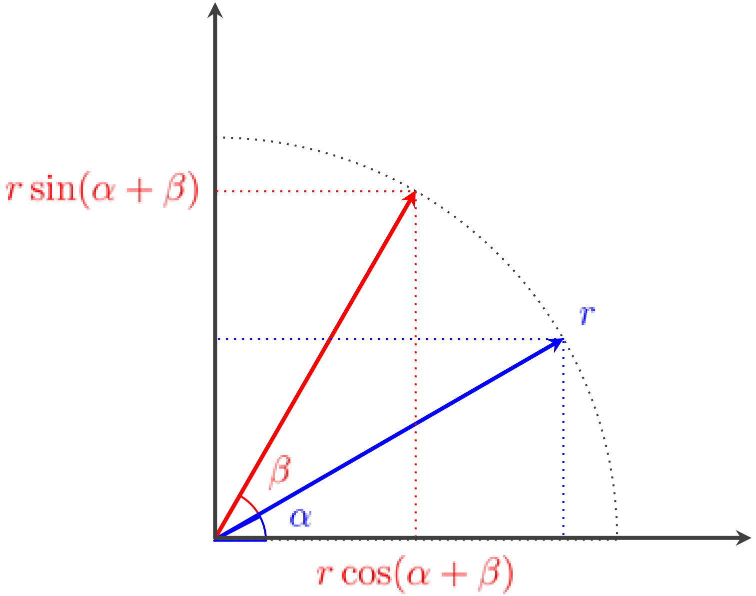
Rotation transformation

rotate the object by a given angle

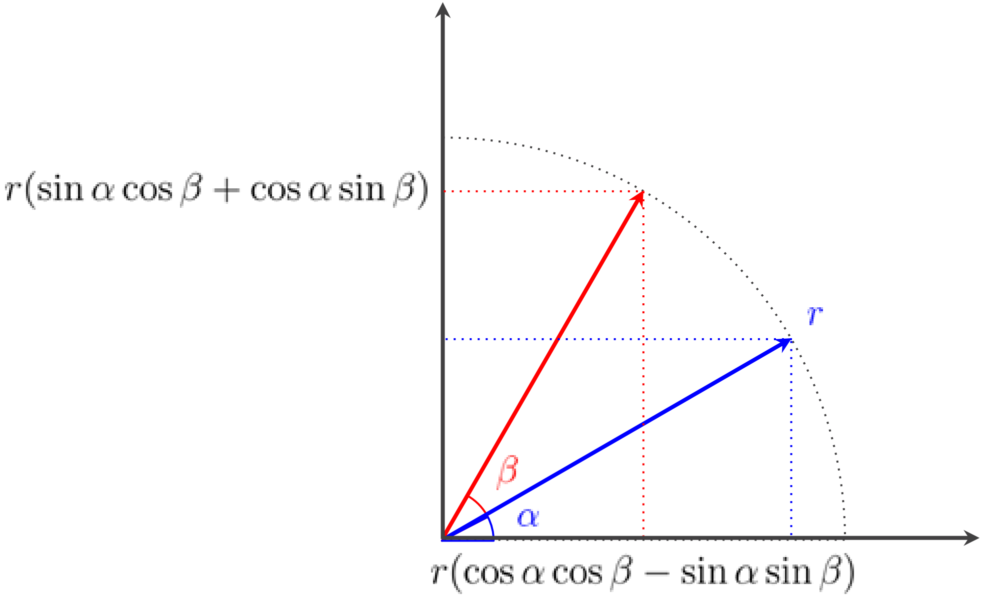


Rotation transformation

rotate the object by a given angle

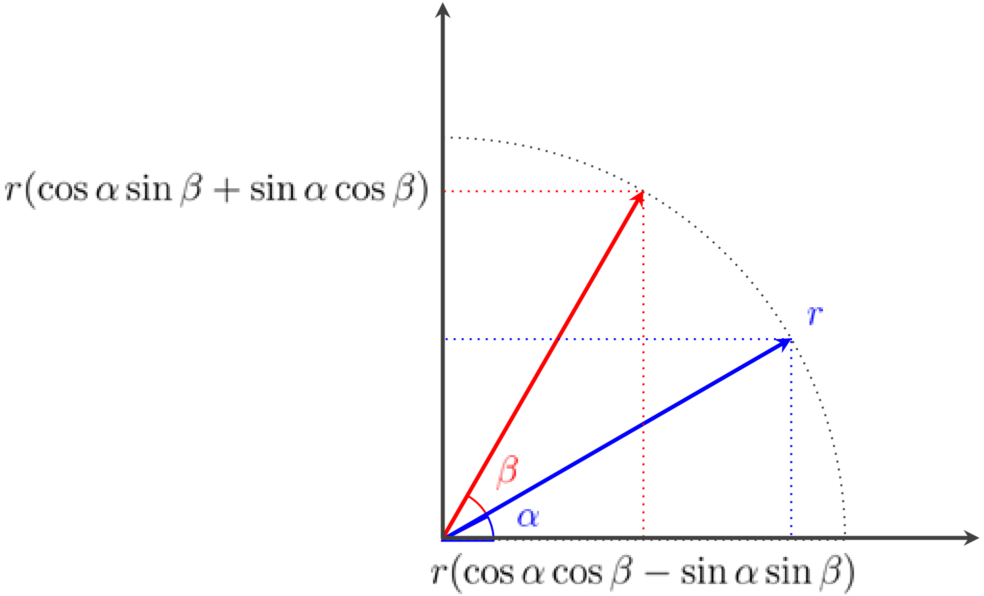


Rotation transformation
rotate the object by a given angle



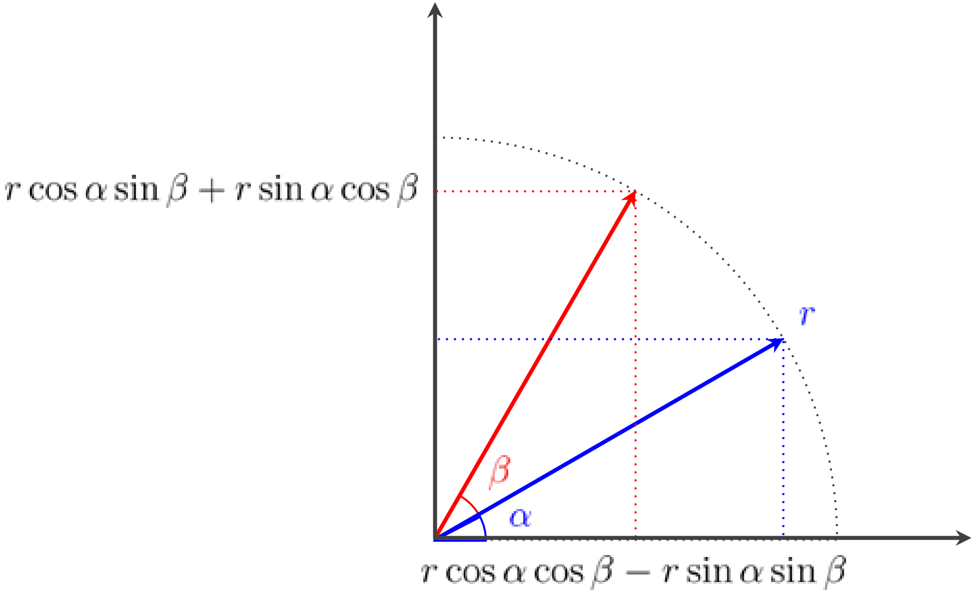
Rotation transformation

rotate the object by a given angle



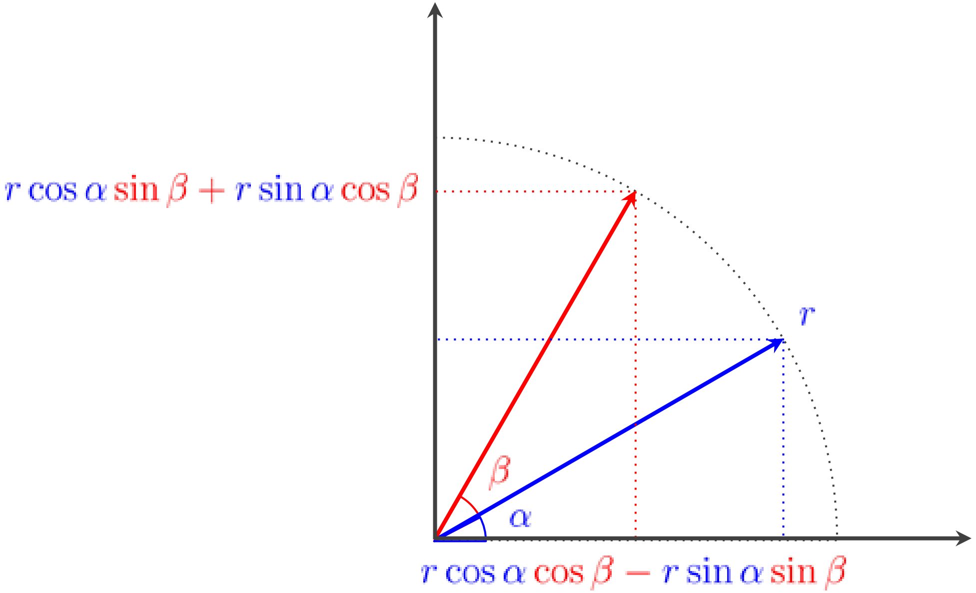
Rotation transformation

rotate the object by a given angle



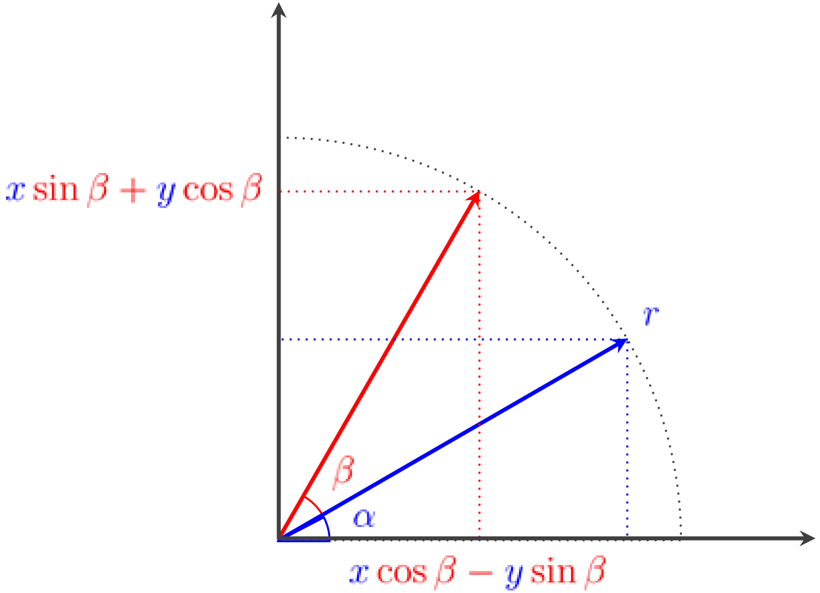
Rotation transformation

rotate the object by a given angle



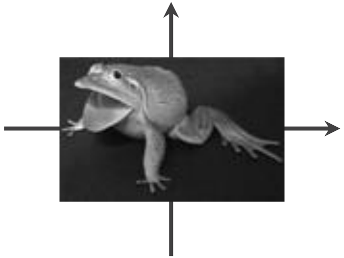
Rotation transformation

rotate the object by a given angle

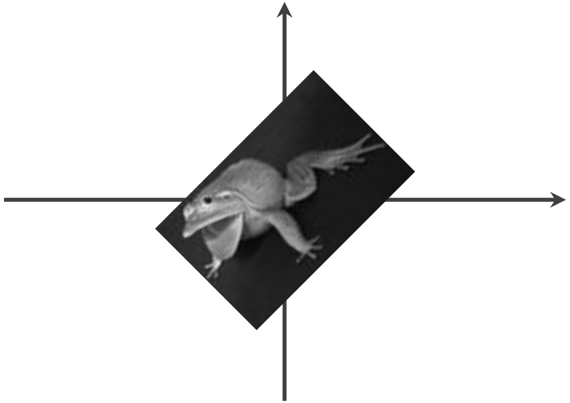


Rotation transformation

rotate the object by a given angle

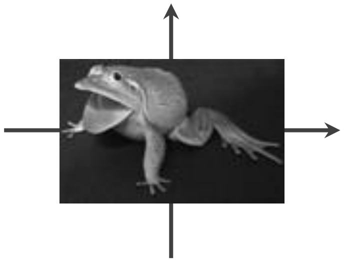


$$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

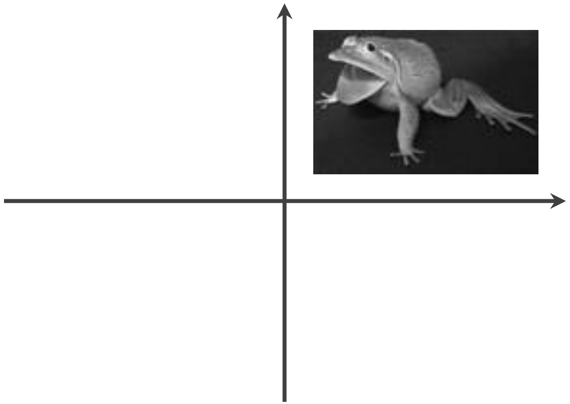


Translation transformation

translate the object by given shifts

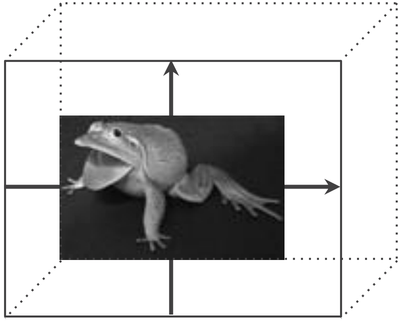


$$\begin{bmatrix} ? & ? \\ ? & ? \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

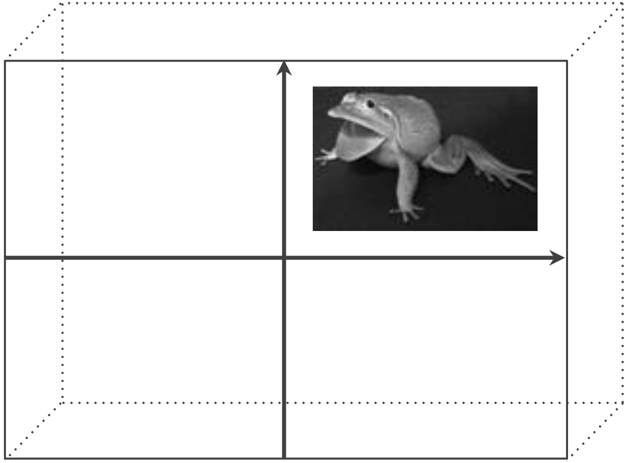


Homogeneous coordinates

assume the image coordinates are in the tridimensional space, but always in the $z = 1$ plane

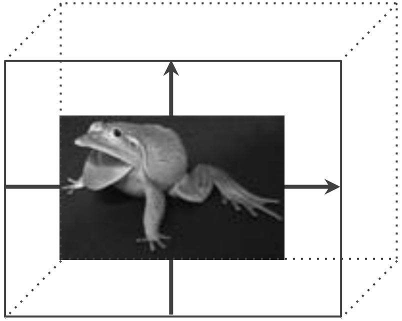


$$\begin{bmatrix} ? & ? & ? \\ ? & ? & ? \\ ? & ? & ? \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

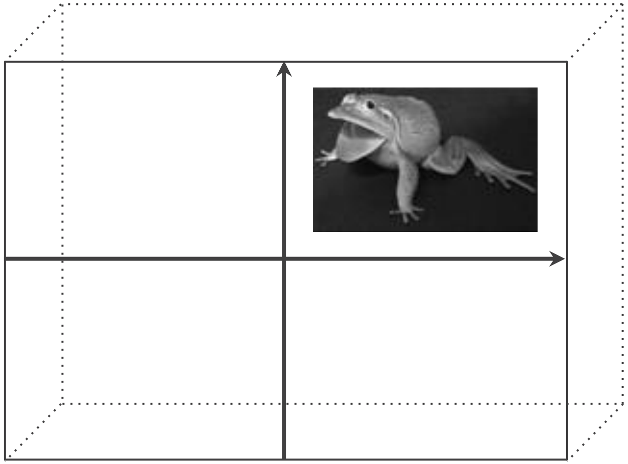


Homogeneous transformation

if the third line is $[0, 0, 1]$, the coordinates stay in the $z = 1$ plane

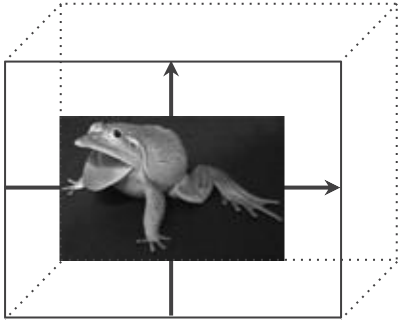


$$\begin{bmatrix} ? & ? & ? \\ ? & ? & ? \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

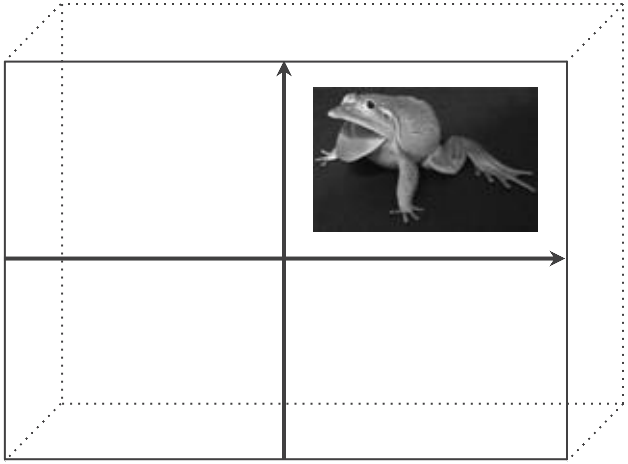


Translation transformation

translate the object by given shifts

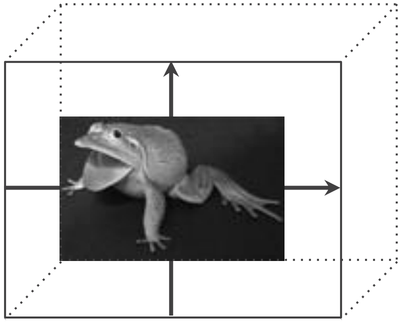


$$\begin{bmatrix} 1 & 0 & d_x \\ 0 & 1 & d_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

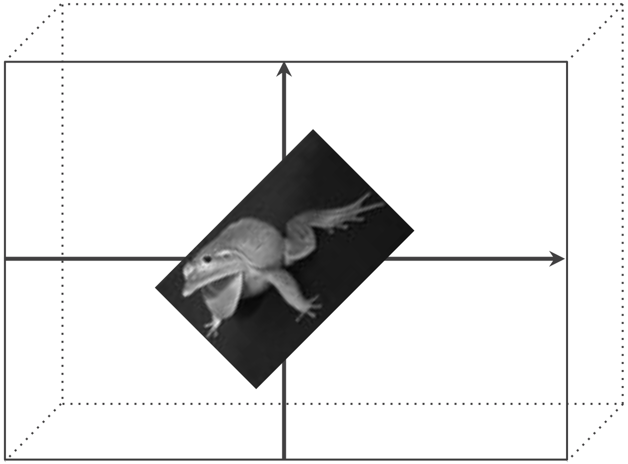


Rotation transformation

rotate the object by a given angle

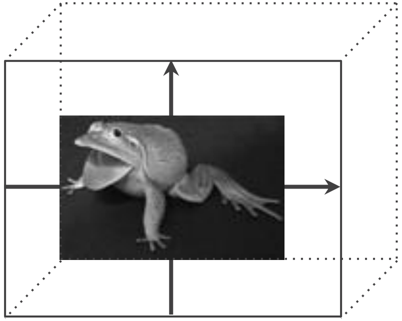


$$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

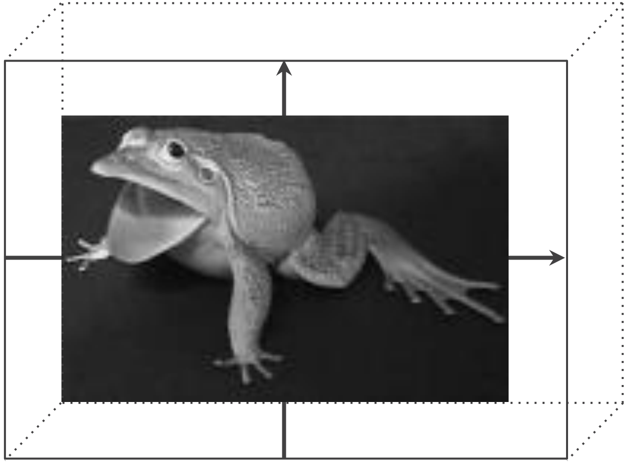


Scale transformation

scale the object by given factors

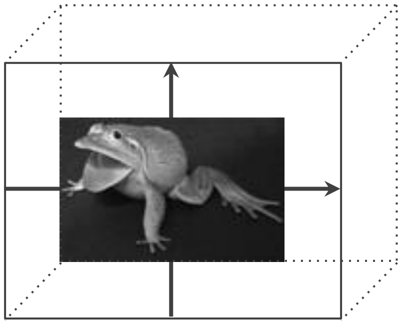


$$\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

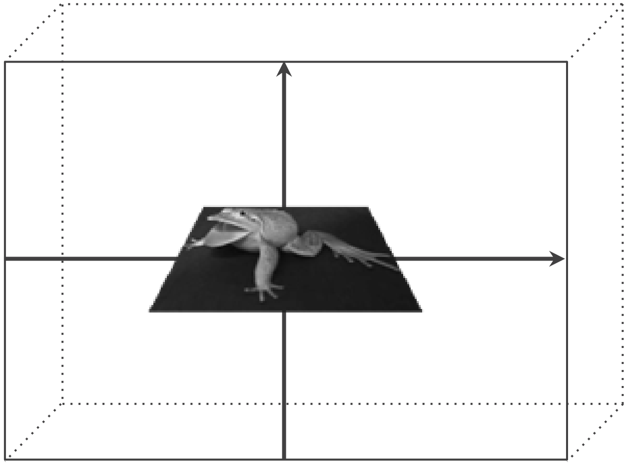


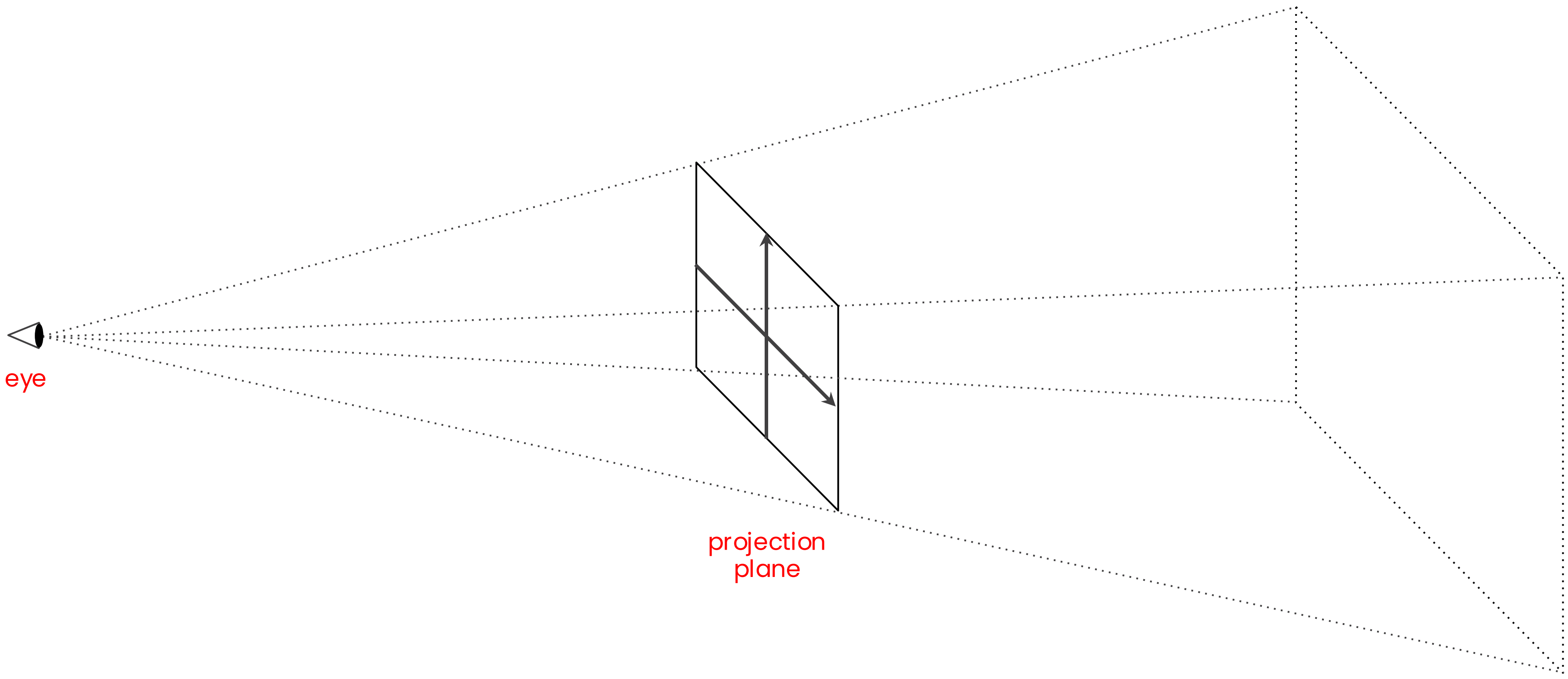
Warp transformation

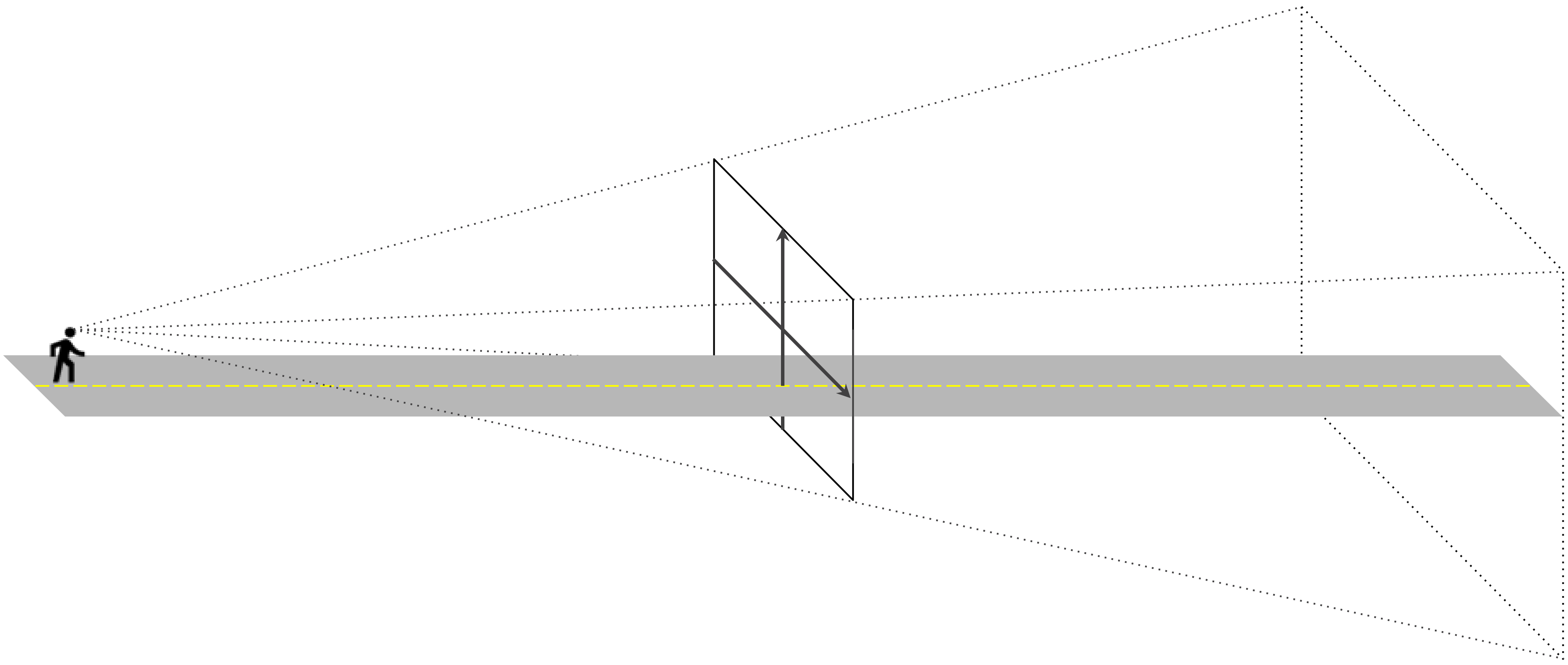
change the perspective of the object

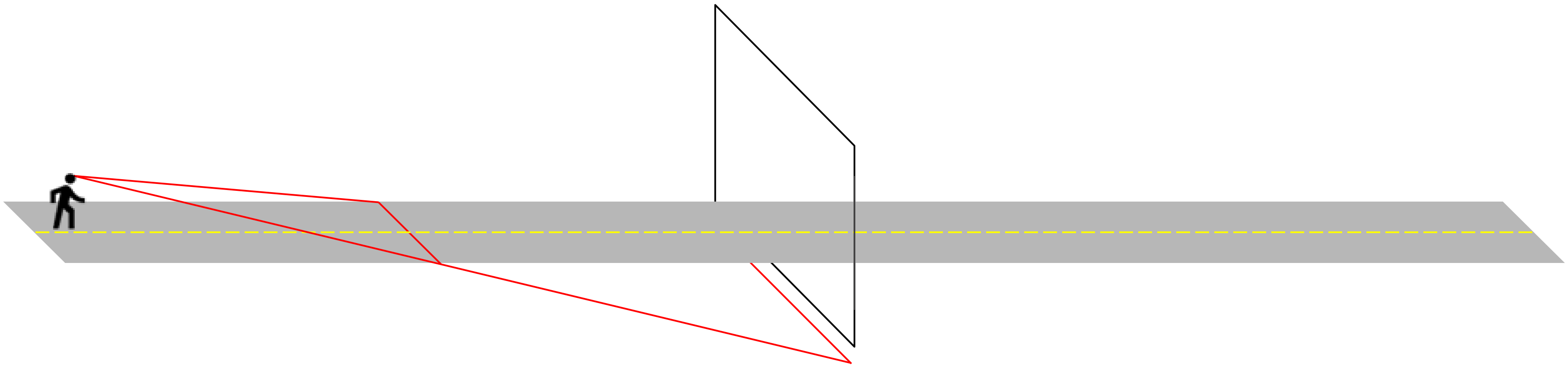


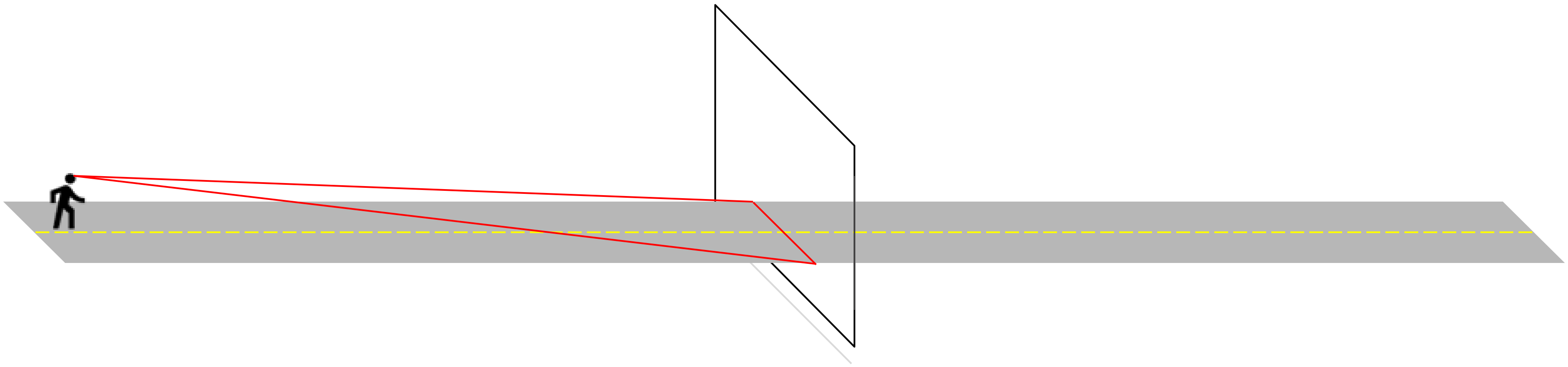
$$\begin{bmatrix} ? & ? & ? \\ ? & ? & ? \\ ? & ? & ? \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

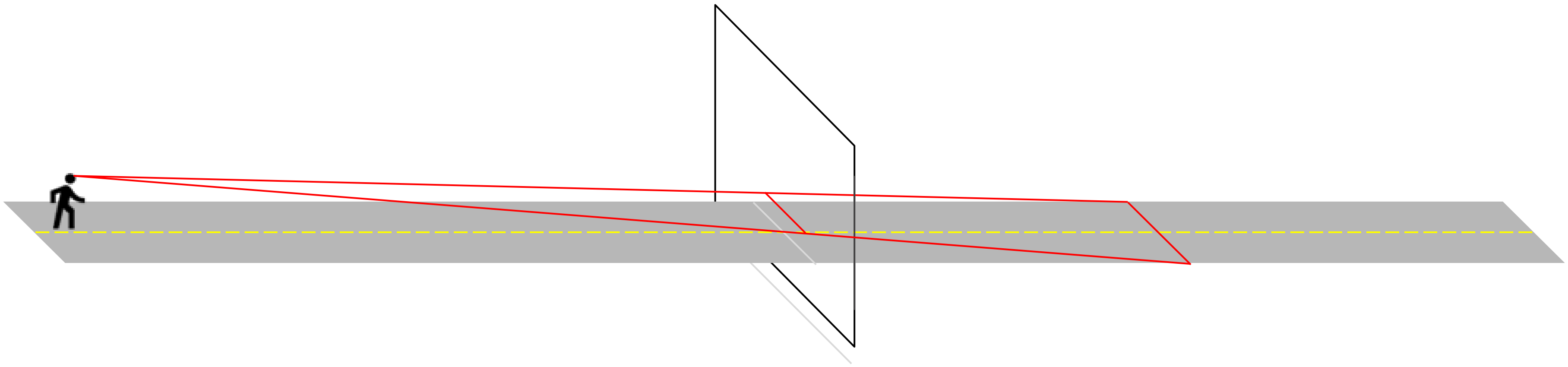


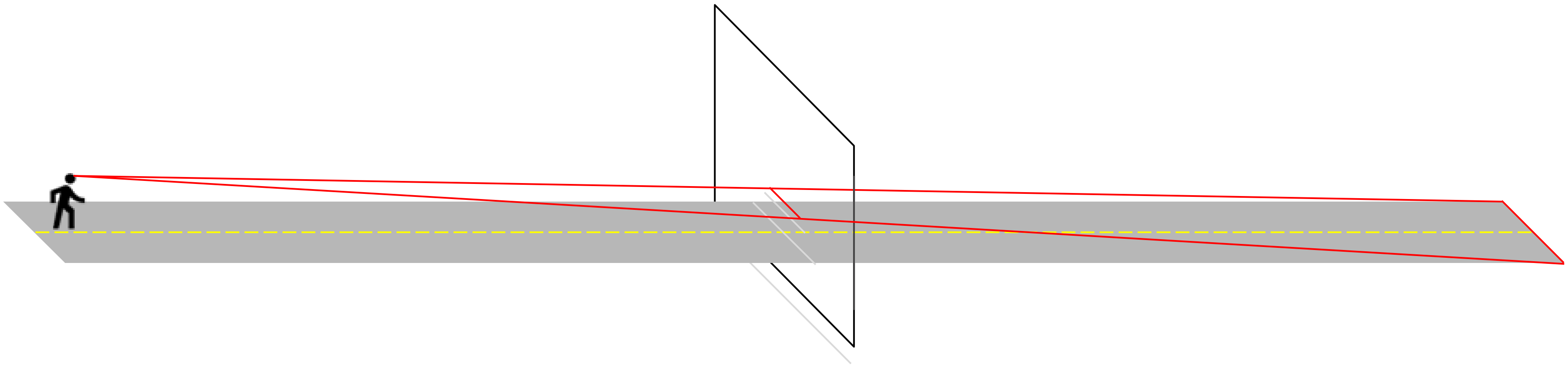

eye
projection
plane

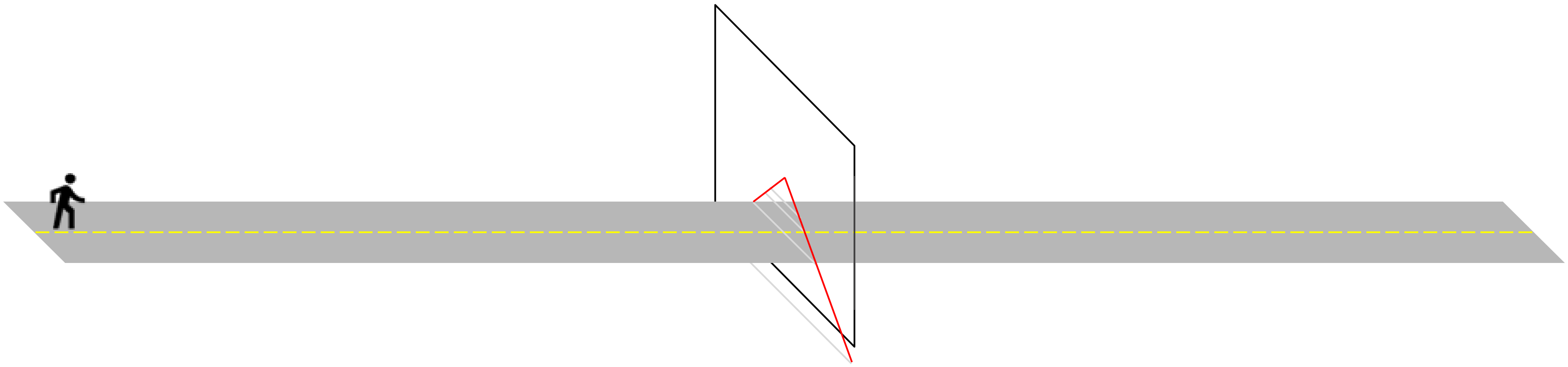






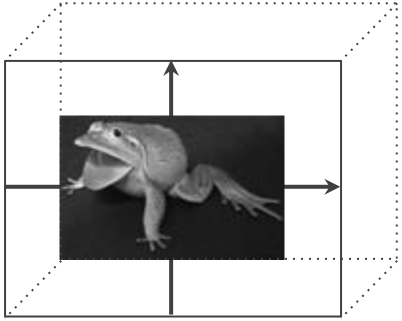




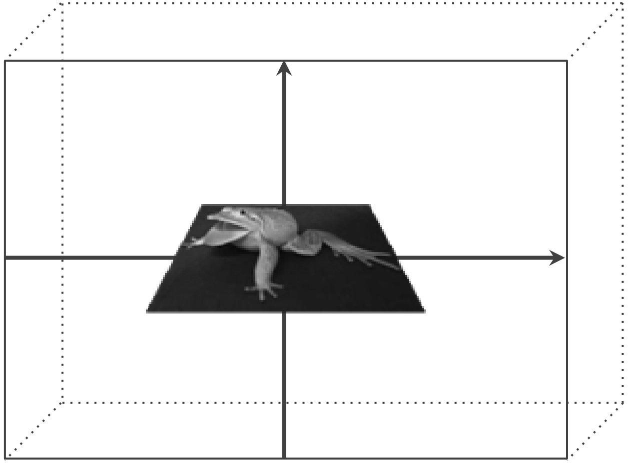


Homography matrix

if the third line is not $[0, 0, 1]$, the coordinates do not necessarily stay in the $z = 1$ plane

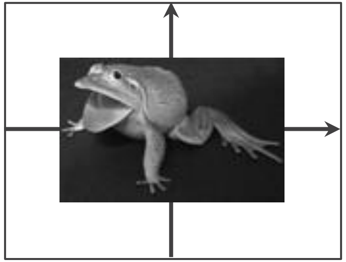


$$\begin{bmatrix} h_{0,0} & h_{0,1} & h_{0,2} \\ h_{1,0} & h_{1,1} & h_{1,2} \\ h_{2,0} & h_{2,1} & h_{2,2} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$



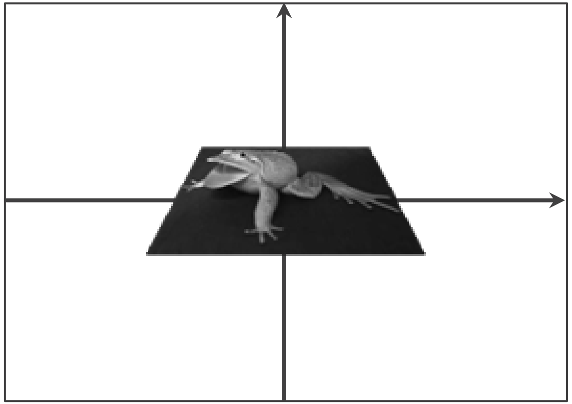
Homography matrix

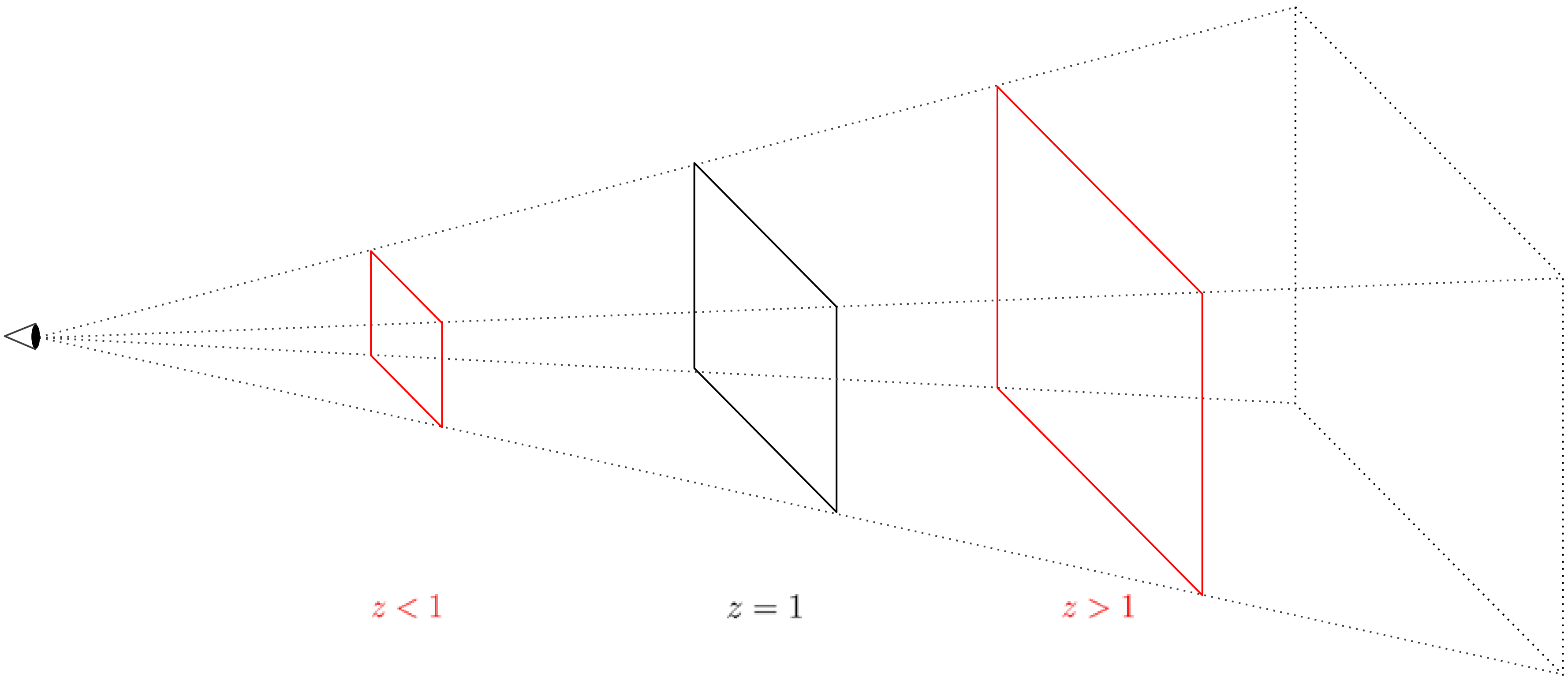
if the third line is not $[0, 0, 1]$, the coordinates do not necessarily stay in the $z = 1$ plane,
but a division by z is enough to bring them back



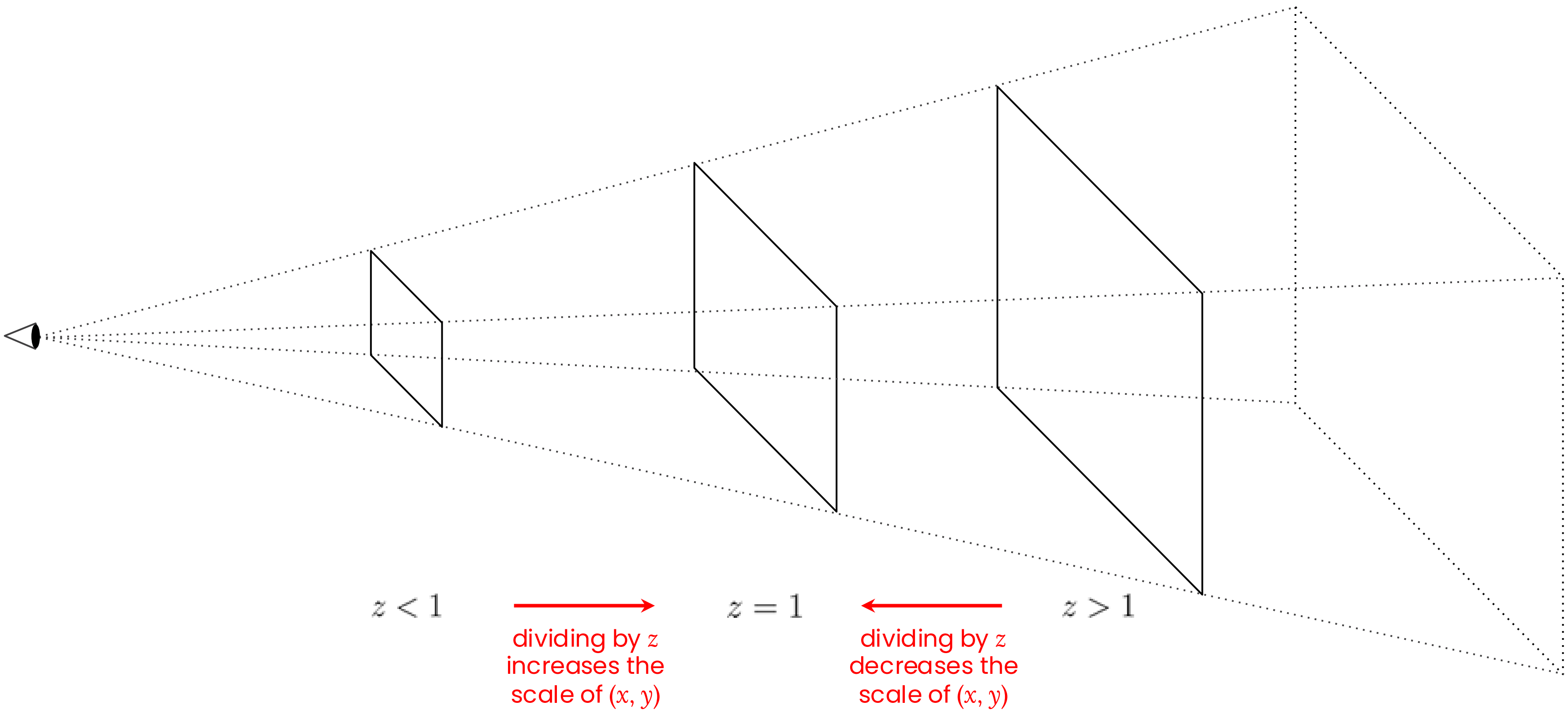
$$\begin{bmatrix} h_{0,0} & h_{0,1} & h_{0,2} \\ h_{1,0} & h_{1,1} & h_{1,2} \\ h_{2,0} & h_{2,1} & h_{2,2} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$z = h_{2,0}x + h_{2,1}y + h_{2,2}$$





$z < 1$
$z = 1$
$z > 1$


$z < 1$
dividing by z increases the scale of (x, y)
$z = 1$
dividing by z decreases the scale of (x, y)
$z > 1$

A matrix multiplication can...

- ...scale.
- ...rotate.
- ...translate.
- ...warp.

A matrix multiplication can...

- ...scale.
- ...rotate.
- ...translate.
- ...warp.

$$A_1 \cdot A_2 \cdots A_k \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

A matrix multiplication can...

- ...scale.
- ...rotate.
- ...translate.
- ...warp.
- ...perform any combination of scales, rotations, translations, and warps, and in any order.

$$\overbrace{A_1 \cdot A_2 \cdots A_k}^{\text{can merge into a single matrix}} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Hence, the analysis separation of the four changes is not relevant...

...and neither is the assumption of the center being in the origin.

handout

Toolkit

- **Language:** Python
- **Library:** OpenCV
- **Platform:** Google Colab

Instructions

1. Organize in groups of 2 or 3 members. No more, no less.

1. Make a copy of the notebook, read it, and do the activities.

1. Clean the notebook, save as `ipynb`, and submit via form.

Next class:

- what is (not) deep learning.

Credits

This material was based on the work of other professors, listed below.

- Fabio Miranda (fabiomiranda@insper.edu.br)
- Raul Ikeda (RaulIGS@insper.edu.br)
- Fabio Ayres (FabioJA@insper.edu.br)
- Igor Montagner (IgorSM1@insper.edu.br)
- Andrew Kurauchi (AndrewTNK@insper.edu.br)
- Luciano Silva (LucianoS4@insper.edu.br)
- Tiago Sanches (tiagoss4@insper.edu.br)

Well, except for the errors. Any errors you might find are probably my fault.

Images

https://fonts.google.com/icons
https://www.amazon.com/Algorithms-Models-Network-Data-Analysis/dp/1107125774
https://www.amazon.com.br/Toddynho-Achocolatado-200ml/dp/B07XJ6RMVD
https://www.amazon.com.br/%C3%81gua-Coco-200Ml-Kero-Sabor/dp/B0768XTBKX

Lowe, D. G. *Distinctive image features from scale-invariant keypoints.* International Journal of Computer Vision, 60. (2004)