Imaging Geometry

Computer Vision

Part-1 CV

Image & IP fundamentals

- Basic
- pixels.
- Enhancement.
- Spatial filtering.

Smoothening → avg.

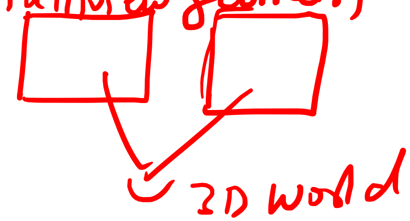
Sharpening → derivative

- Edge detection. → boundaries.
- Edge linking → Hough transform.
- Segmentation → threshold.

Part 2

(Rel^n b/w 3D world points & images)

- Various geometric transf → 2D & 3D
- Camera calibration.
- Multiview geometry

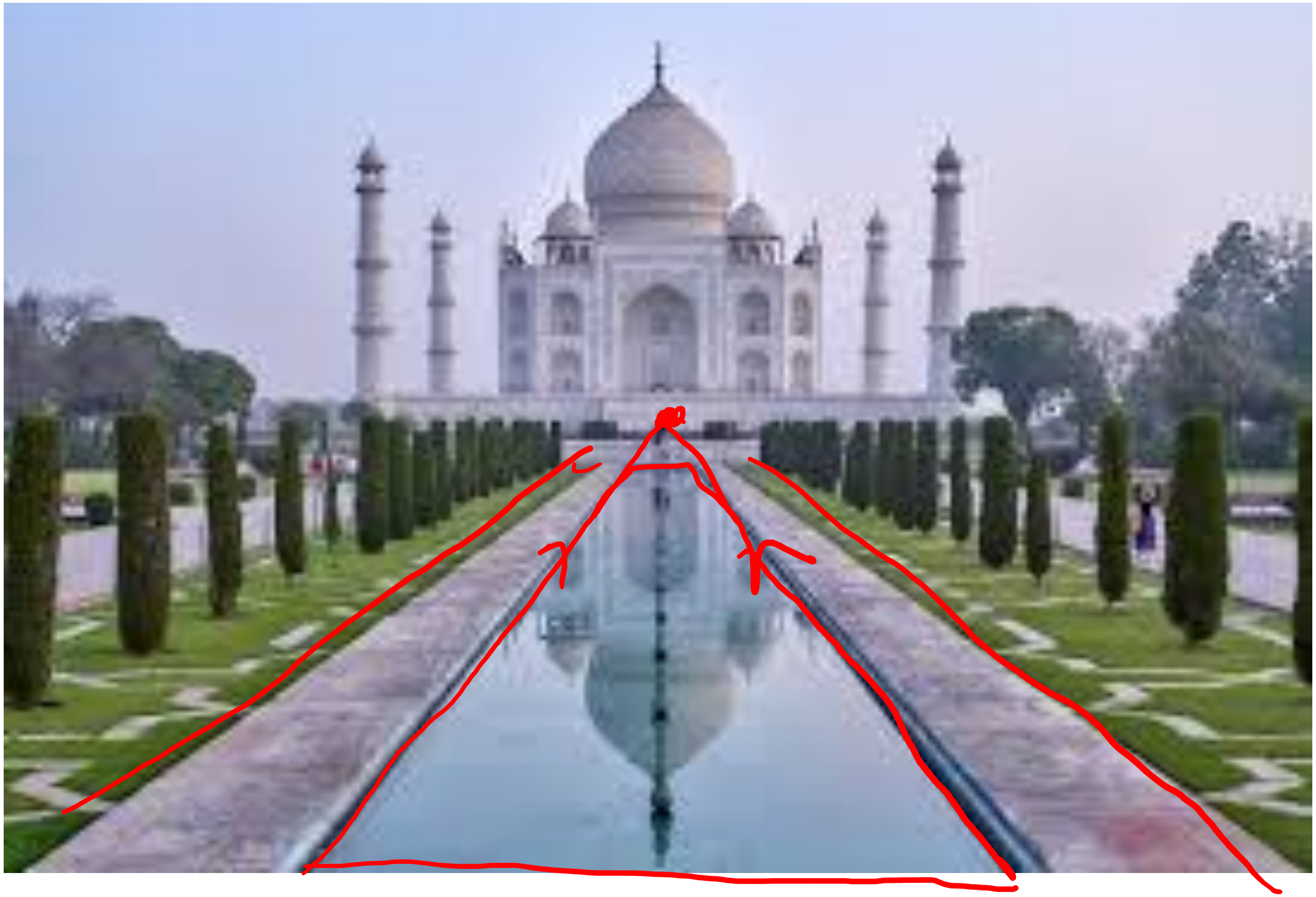


- optical flow
- Stereo.

Modelling from 3D to 2D world

Perspective Matters!!

Taj Mahal




Paris town hall
"Anamorphosis"

Cricket

optical flow.

Cameras, Multiple Views, and Motion

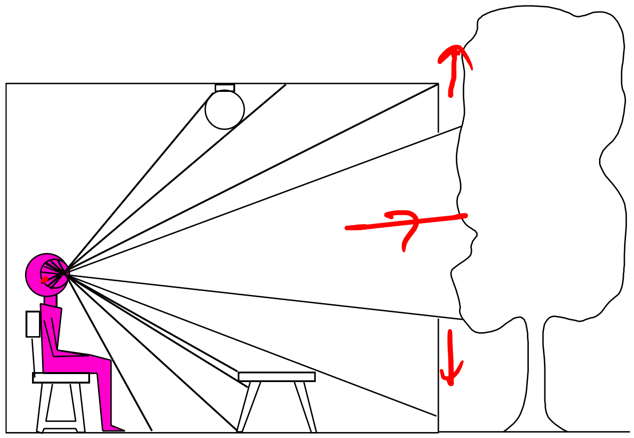
- Imaging Geometry
  - Image transforms like scaling, rotation etc
- Perspective Transformation
- Projective Transformation
- Camera Model
- Camera Calibration

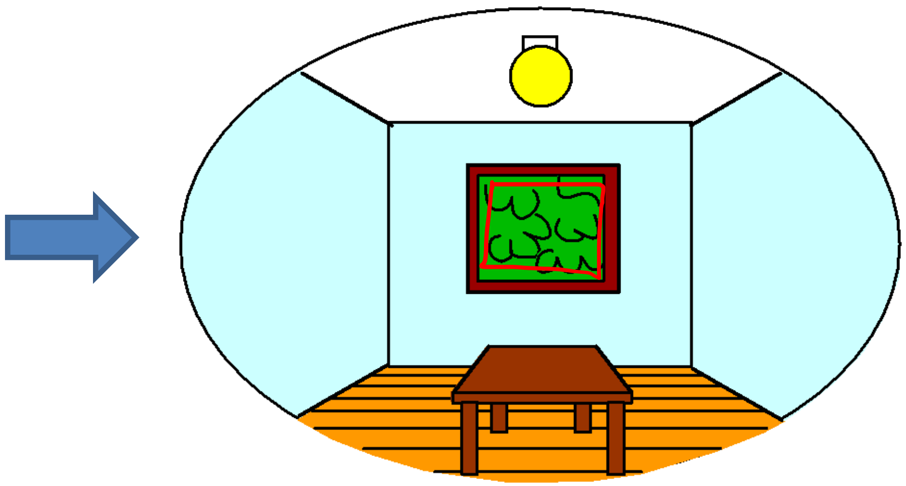
Dimensionality Reduction Machine (3D to 2D)

Camera.

3D world

2D image





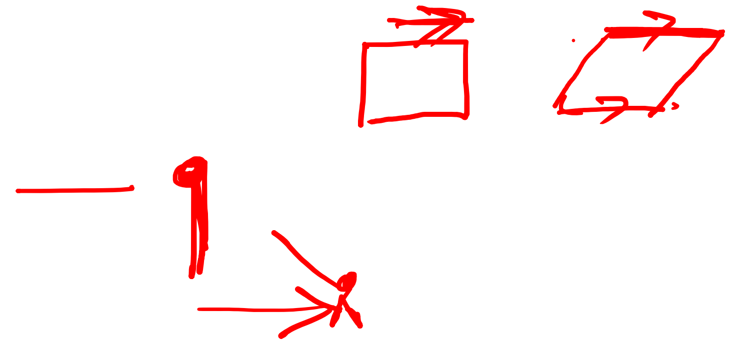
Point of observation

How to recover knowledge about 3D from 2D…??

Common transformations

(a)

Original



Transformed

Translation

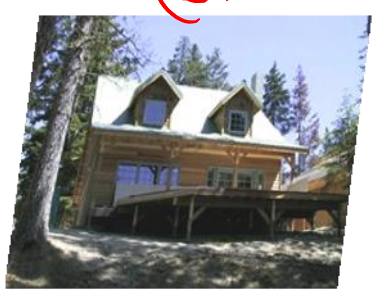
(b)

Rotation

(d)

Scaling

(e)



Affine

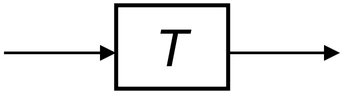
(f)

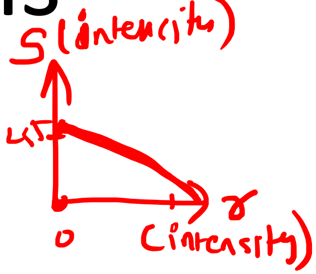
Perspective

Slide credit (next few slides):
A. Efros and/or S. Seitz

Parametric (global) transformations

original





$p' = T(p)$

p = (x,y)   **p'** = (x',y')

Transformation T is a coordinate-changing machine:

$$p' = T(p)$$

$$p' = T(p)$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} T \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

What does it mean that *T* is global?

– *T* is the same for any (every) point p in the image

– *T* can be described by just a few numbers (parameters)

For linear transformations, we can represent T as a matrix

$$p' = \mathbf{T}p$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \mathbf{T} \begin{bmatrix} x \\ y \end{bmatrix}$$

Common transformations

Original

Transformed

Translation

Rotation

Scaling

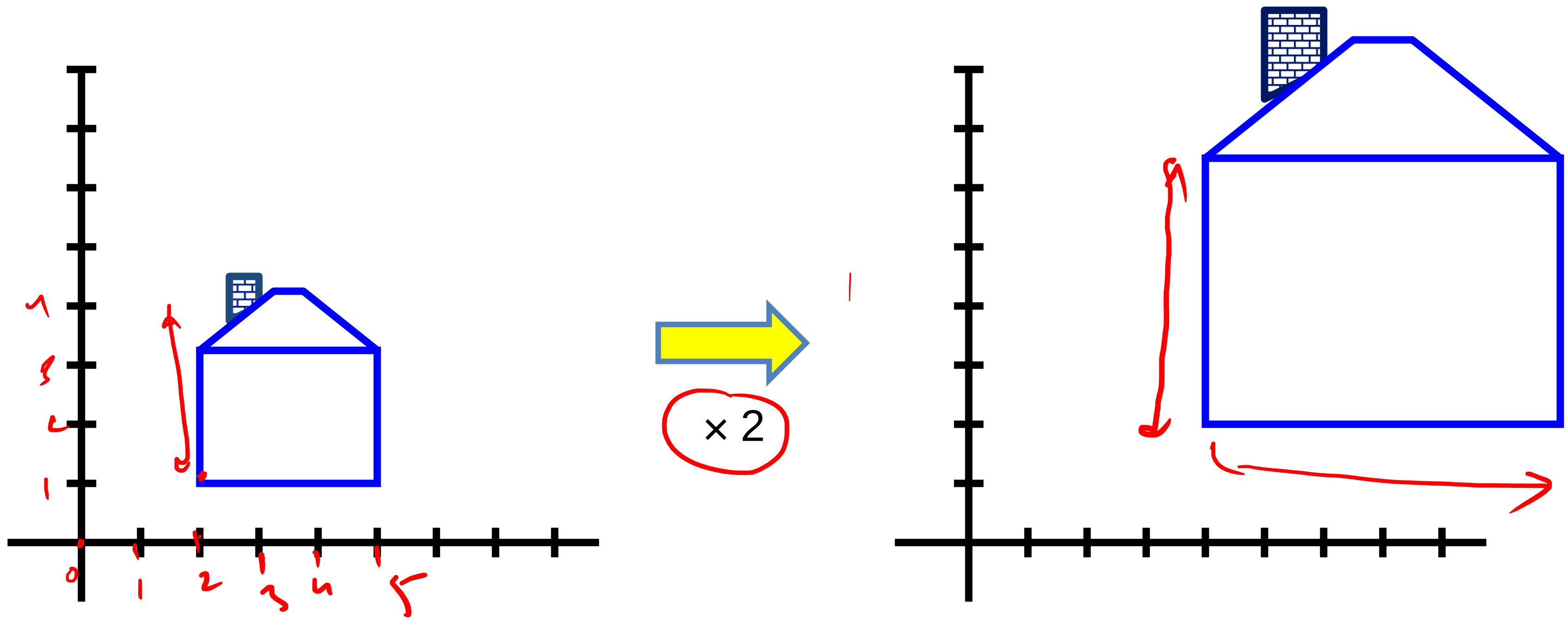
Affine

Perspective

Slide credit (next few slides):
A. Efros and/or S. Seitz

① Scaling

- *Scaling* a coordinate means multiplying each of its components by a scalar
- *Uniform scaling* means this scalar is the same for all components:

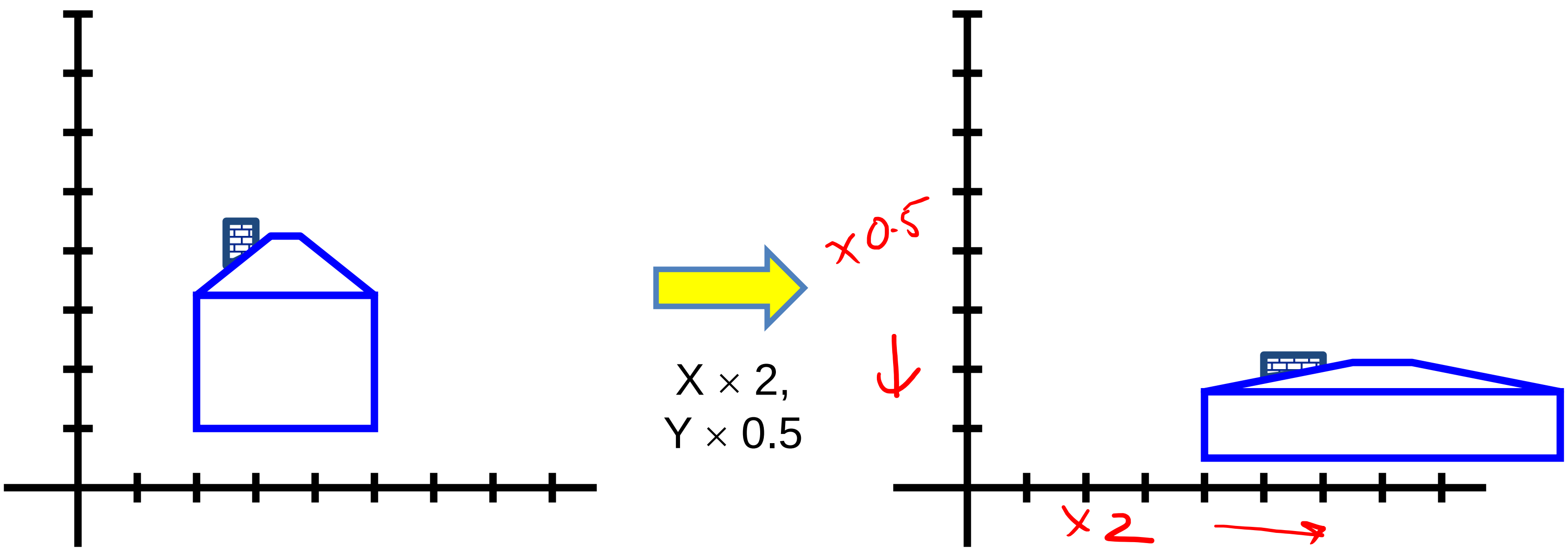


Scaling

- *Non-uniform scaling*: different scalars per component:

$$p' = T(p)$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} T \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



Scaling

$x' = a x$
$y' = b y$

- Scaling operation: $x' = ax$

  $y' = by$

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

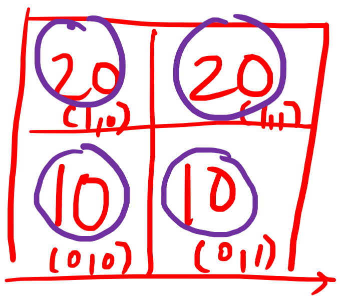
Scaling Transformation matrix.

- Or, in matrix form:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}}_{\textit{scaling matrix } S} \begin{bmatrix} x \\ y \end{bmatrix}$$

$x' = ax$
$y' = by$

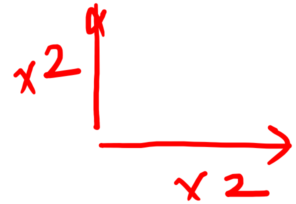
o s



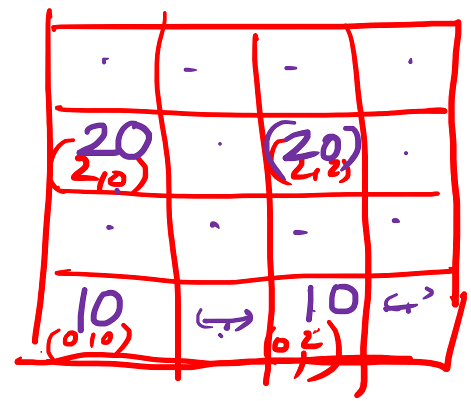
2 × 2

Scale this imge

2 x 2

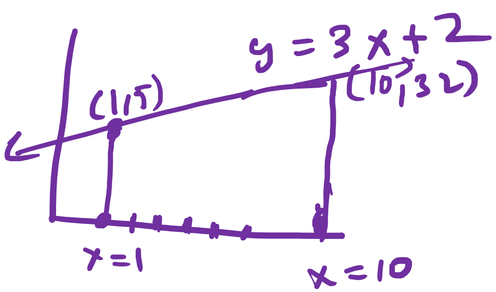


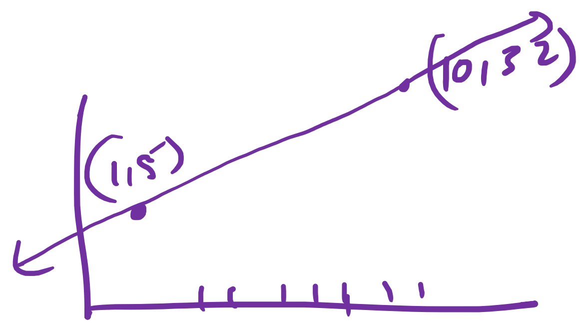
Scaled imge



We fill these empty values → Image Interpolation.

$$\begin{bmatrix} \ \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} \ \\ \ \end{bmatrix}$$

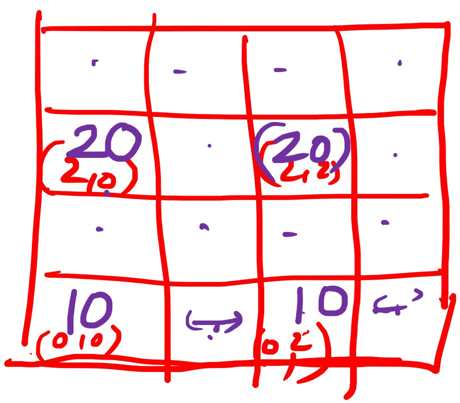




s imge

2

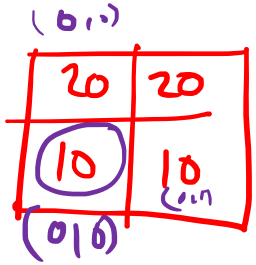
For Scaled imge



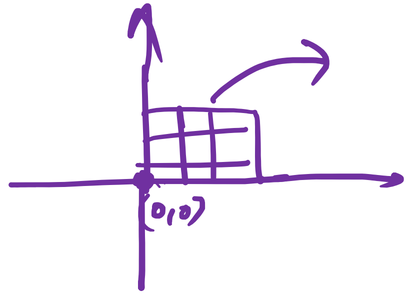
ese

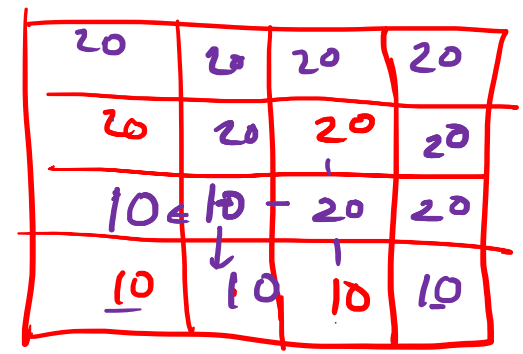
→ Image Interpolation .

(10, 3/2)



2x2





① Nearest Neighbour Image Interpl^n

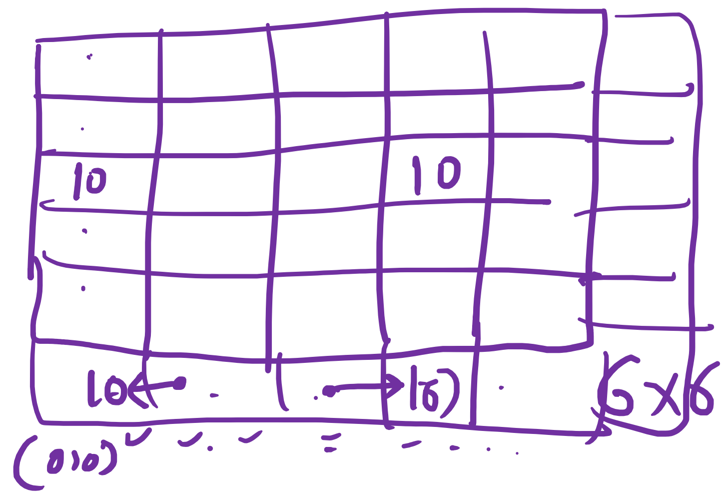
$(L, R) \Rightarrow L$

$(L, R, T, B) \Rightarrow T$

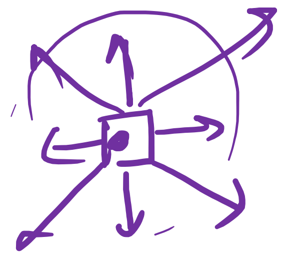
$(L, R, T) = T$

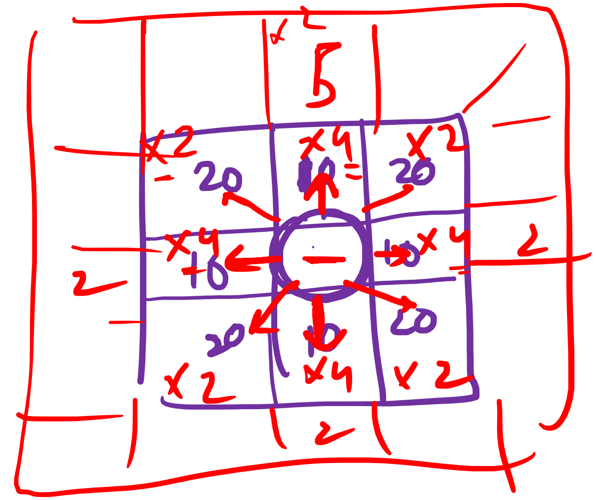
$(L\ R, B) = B$

$(T, B) = B$



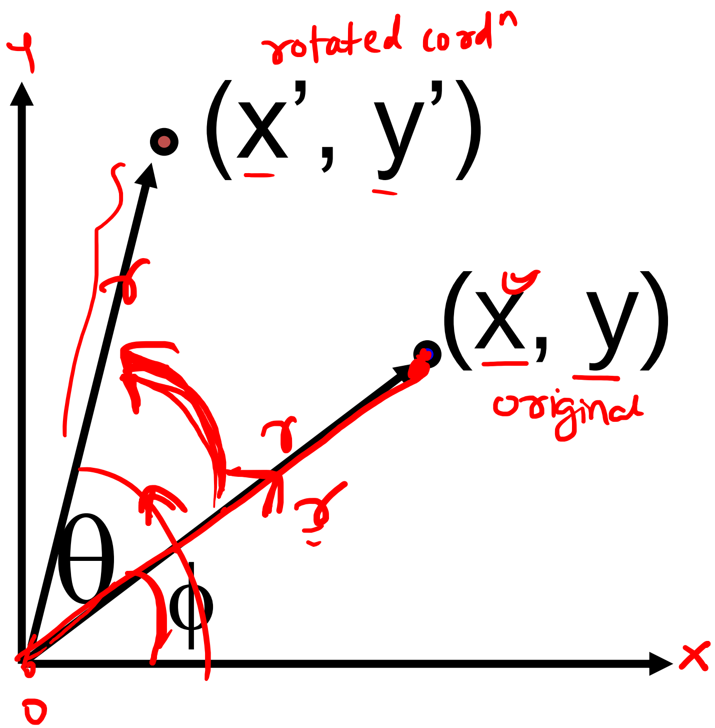
② Bilinear Interpolⁿ





③ Bicubic Interpolⁿ.

2-D Rotation



Polar coordinates…

$x = r \cos(\phi)$
$y = r \sin(\phi)$ original

$x' = r \cos(\phi + \theta)$
$y' = r \sin(\phi + \theta)$

Trig Identity…

$x' = \overbrace{r \cos(\phi)}^{x} \cos(\theta) - \overbrace{r \sin(\phi)}^{y} \sin(\theta)$
$y' = r \sin(\phi) \cos(\theta) + \underbrace{r \cos(\phi)}_{x} \sin(\theta)$

Substitute…

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

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

2-D Rotation

This is easy to capture in matrix form: θ

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}}_{\mathbf{R}} \begin{bmatrix} x \\ y \end{bmatrix}$$

$(-\theta)$

$$\begin{bmatrix} \cos(-\theta) & -\sin(-\theta) \\ \sin(-\theta) & \cos(-\theta) \end{bmatrix}$$

$$\underbrace{\begin{bmatrix} \cos\theta & +\sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}}_{R^{-1}}$$

Even though sin(θ) and cos(θ) are nonlinear functions of θ,

- ***x' is a linear combination of x and y***
- ***y' is a linear combination of x and y***

$R^{-1} = (R)^T$

What is the inverse transformation?

- Rotation by –θ
- For rotation matrices $\mathbf{R}^{-1} = \mathbf{R}^T$

Basic 2D transformations

$x' = \alpha x + y$
$y' = x + \beta y$

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

Scale

③
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & \alpha_x \\ \alpha_y & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Shear

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

②
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\Theta & -\sin\Theta \\ \sin\Theta & \cos\Theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Rotate

④
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Translate

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Affine

Affine is any combination of translation, scale, rotation, and shear

$$\left(\begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \end{bmatrix} \right)$$

Affine Transformations

apply on 2D image signals.

Affine transformations are combinations of

- Linear transformations, and
- Translations

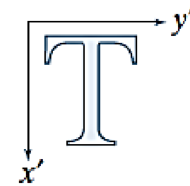
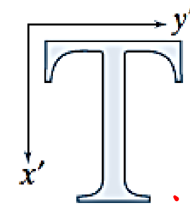
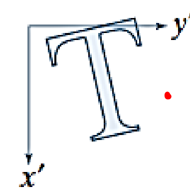
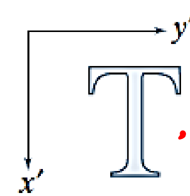
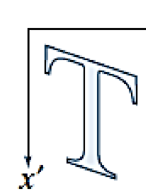
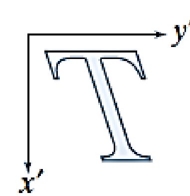
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

or

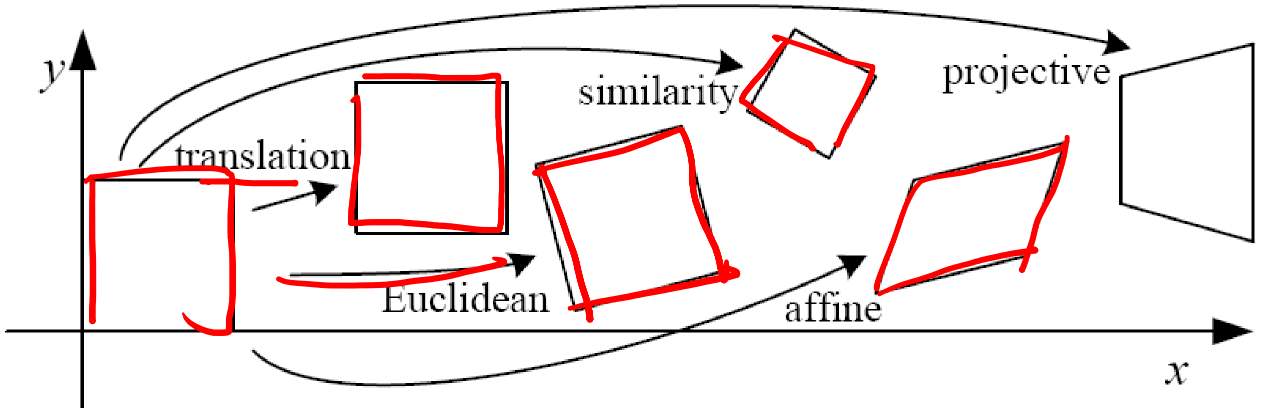
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Properties of affine transformations:

- Lines map to lines
- Parallel lines remain parallel
- Ratios are preserved
- Closed under composition

Transformation Name	Affine Matrix, A	Coordinate Equations	Example
Identity	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x$ $y' = y$	
Scaling/Reflection (For reflection, set one scaling factor to –1 and the other to 0)	$\begin{bmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = c_x x$ $y' = c_y y$	
Rotation (about the origin)	$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x\cos\theta - y\sin\theta$ $y' = x\sin\theta + y\cos\theta$	
Translation	$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x + t_x$ $y' = y + t_y$	
Shear (vertical)	$\begin{bmatrix} 1 & s_v & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x + s_v y$ $y' = y$	 v
Shear (horizontal)	$\begin{bmatrix} 1 & 0 & 0 \\ s_h & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x$ $y' = s_h x + y$	 h

2D image transformations (reference table)



Transformation	Matrix	# DoF	Preserves	Icon
translation	$\begin{bmatrix} I \mid t \end{bmatrix}_{2\times 3}$	2	orientation	
rigid (Euclidean)	$\begin{bmatrix} R \mid t \end{bmatrix}_{2\times 3}$	3	lengths	
similarity	$\begin{bmatrix} sR \mid t \end{bmatrix}_{2\times 3}$	4	angles	
affine	$\begin{bmatrix} A \end{bmatrix}_{2\times 3}$	6	parallelism	
projective	$\begin{bmatrix} \tilde{H} \end{bmatrix}_{3\times 3}$	8	straight lines	

(rot + transt)

'Homography'

→ view point manipulation.

Table 2.1 Hierarchy of 2D coordinate transformations. Each transformation also preserves the properties listed in the rows below it, i.e., similarity preserves not only angles but also parallelism and straight lines. The 2×3 matrices are extended with a third $[\mathbf{0}^T\ 1]$ row to form a full 3×3 matrix for homogeneous coordinate transformations.

Szeliski 2.1

Projective Transformations

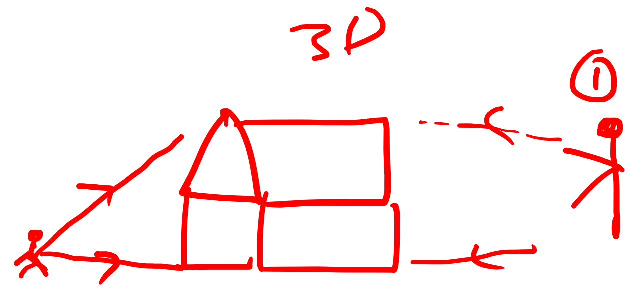
$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix}$$

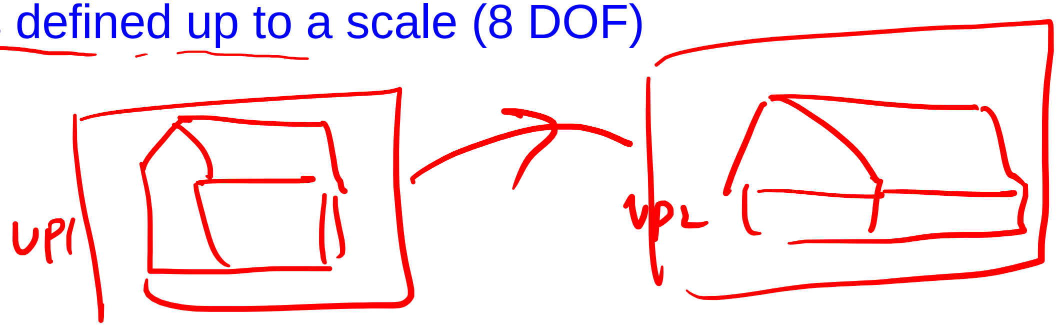
Projective transformations are combos of

- Affine transformations, and
- Projective warps

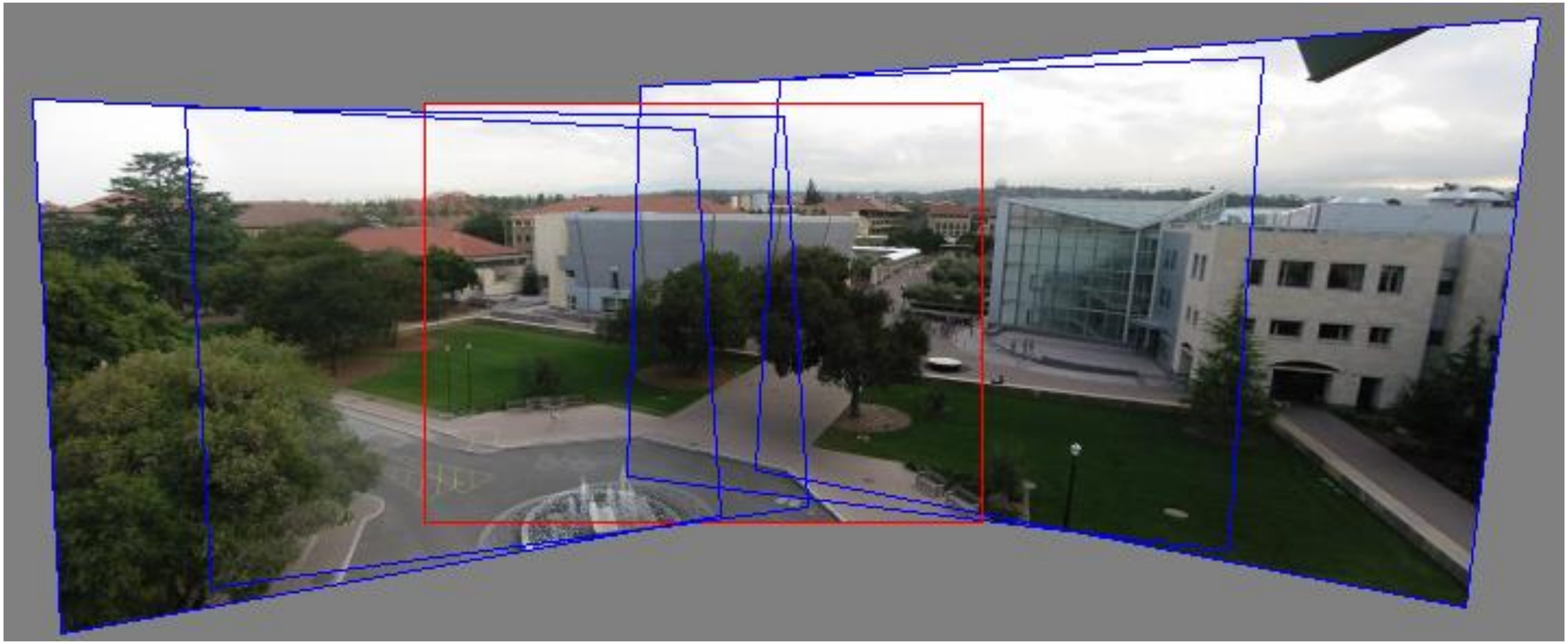
Properties of projective transformations:

- Lines map to lines
- Parallel lines do not necessarily remain parallel
- Ratios are not preserved
- Models change of basis
- Projective matrix is defined up to a scale (8 DOF)





we use projective transforms to create a 360 panorama



- In order to figure this out, we need to learn what a **camera** is

The Geometry of Image Formation

Szeliski 2.1, parts of 2.2

"2D" "3D"

Mapping between image and world coordinates

- (1) Pinhole camera model
- (2) Projective geometry
  - Vanishing points and lines
- (3) Projection matrix

2 ways of Image form

Orthographic projection

Perspective projection

Slides from James Hays, Derek Hoiem, Alexei Efros, Steve Seitz, and David Forsyth

Image Formation: Orthographic Projection

- Means of representing 3-dimensional objects in 2-Dimensions.
- It is a form of parallel projection, in which all the projection lines are orthogonal to the projection plane

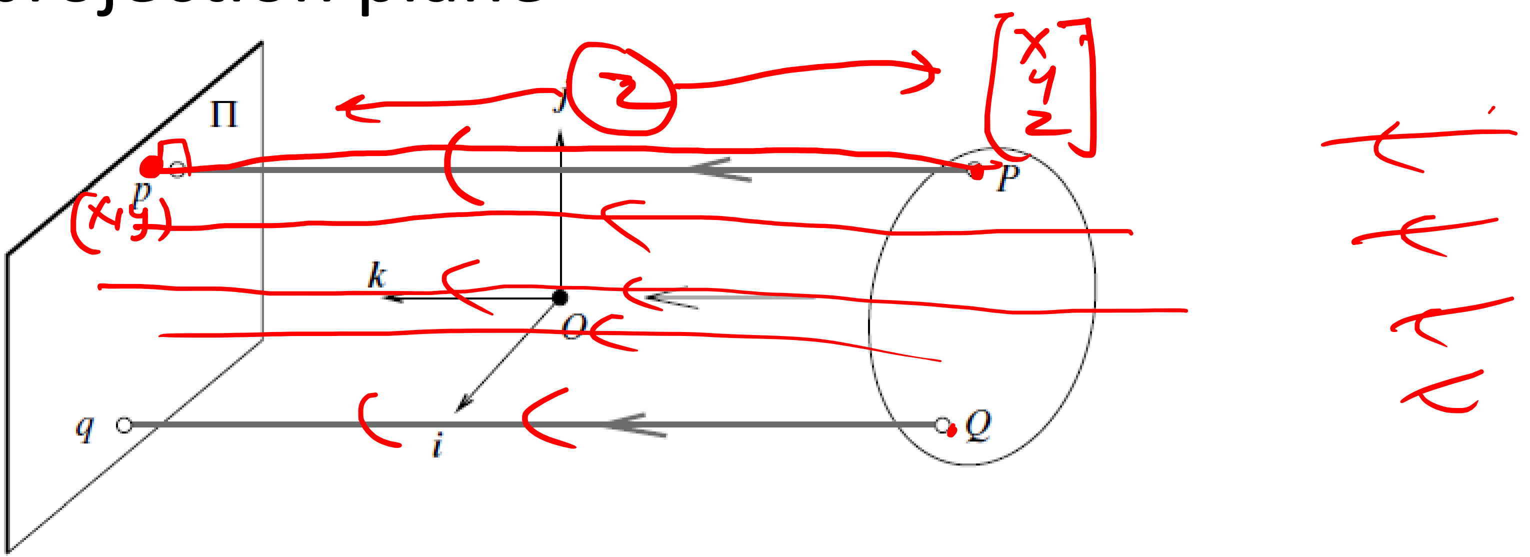
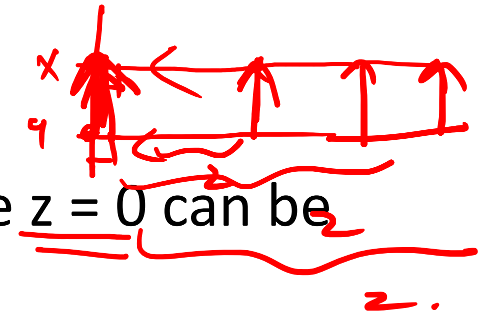


FIGURE 1.6: Orthographic projection. Unlike other geometric models of the image formation process, orthographic projection does not involve a reversal of image features.

Orthographic Projections

does'nt give me any info about z



- A simple orthographic projection onto the plane z = 0 can be defined by the following matrix:

$$P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

- For each point $v = (v_x, v_y, v_z)$, the transformed point Pv would be

$$Pv = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \overbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}}^{P} \overbrace{\begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}}^{3D} \quad x' = v_x,\ y' = v_y,\ z' = 0$$

- Often, it is more useful to use homogeneous coordinates. The transformation in homogeneous coordinates

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

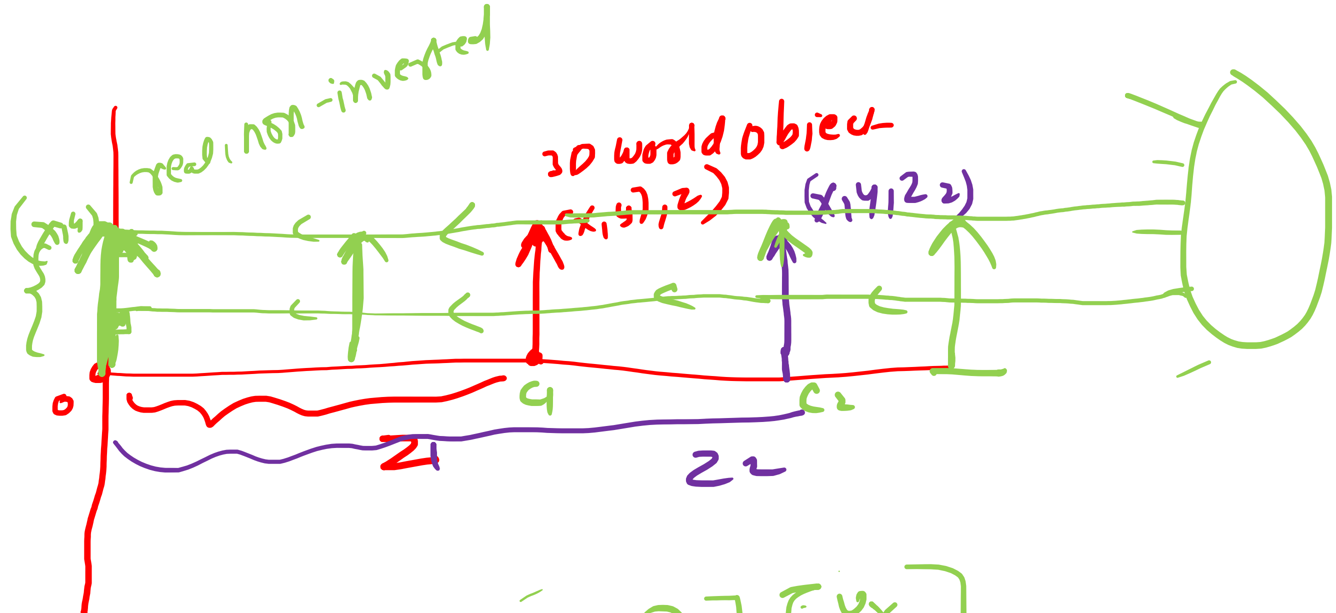
2D vector $= (x, y) \xrightarrow{3D} \underbrace{(fx, fy, f)}_{H.C}$

$(fx/f, fy/f) \leftarrow$

- For each homogeneous vector $v = (v_x, v_y, v_z, 1)$, the transformed vector Pv would be

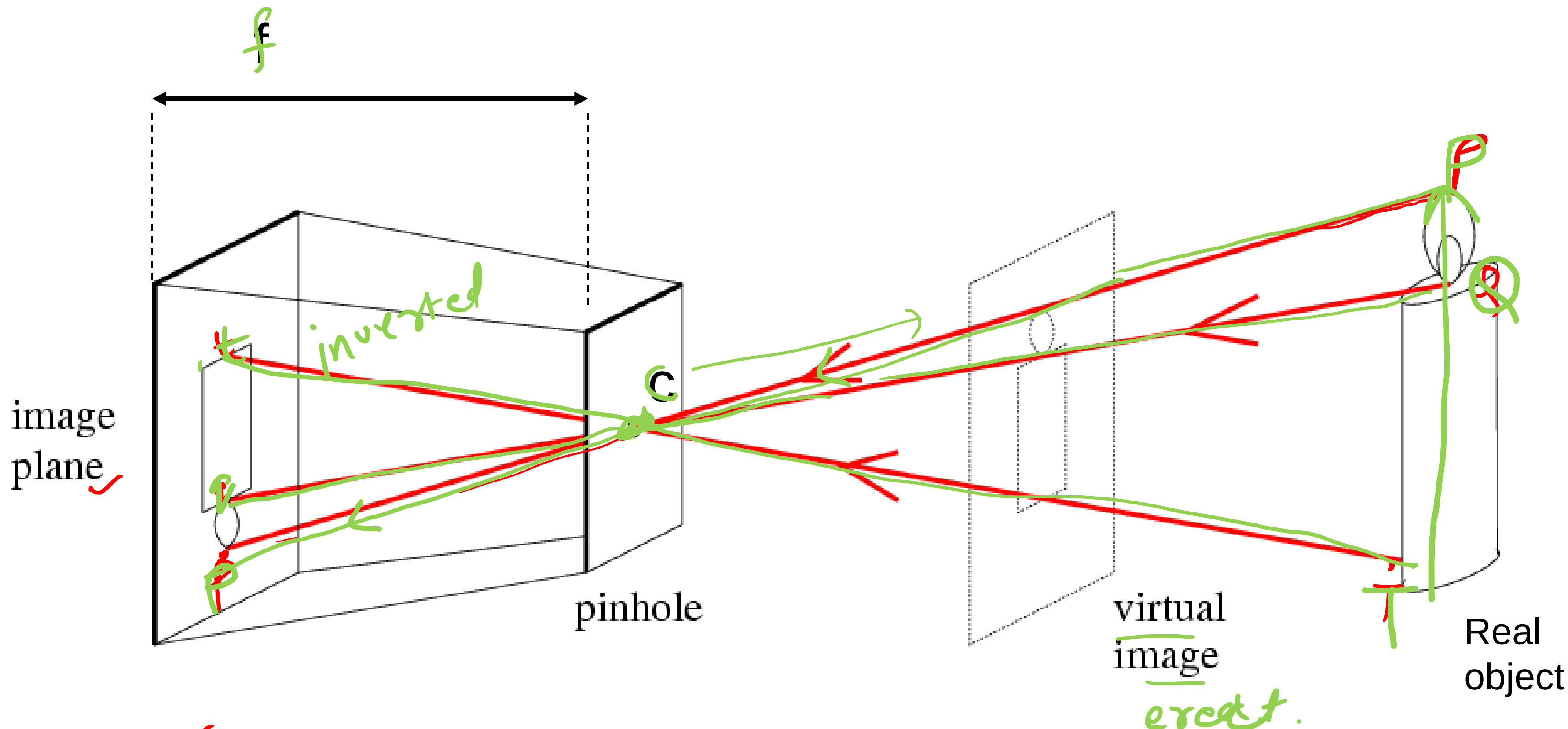
$$Pv = \overbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}^{4\times4} \overbrace{\begin{bmatrix} v_x \\ v_y \\ v_z \\ 1 \end{bmatrix}}^{3D \to 4D} = \overbrace{\begin{bmatrix} v_x \\ v_y \\ 0 \\ 1 \end{bmatrix}}^{2D \text{ image points}}$$

https://en.wikipedia.org/wiki/Orthographic_projection



$$P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}$$

Pinhole camera model



f = Focal length

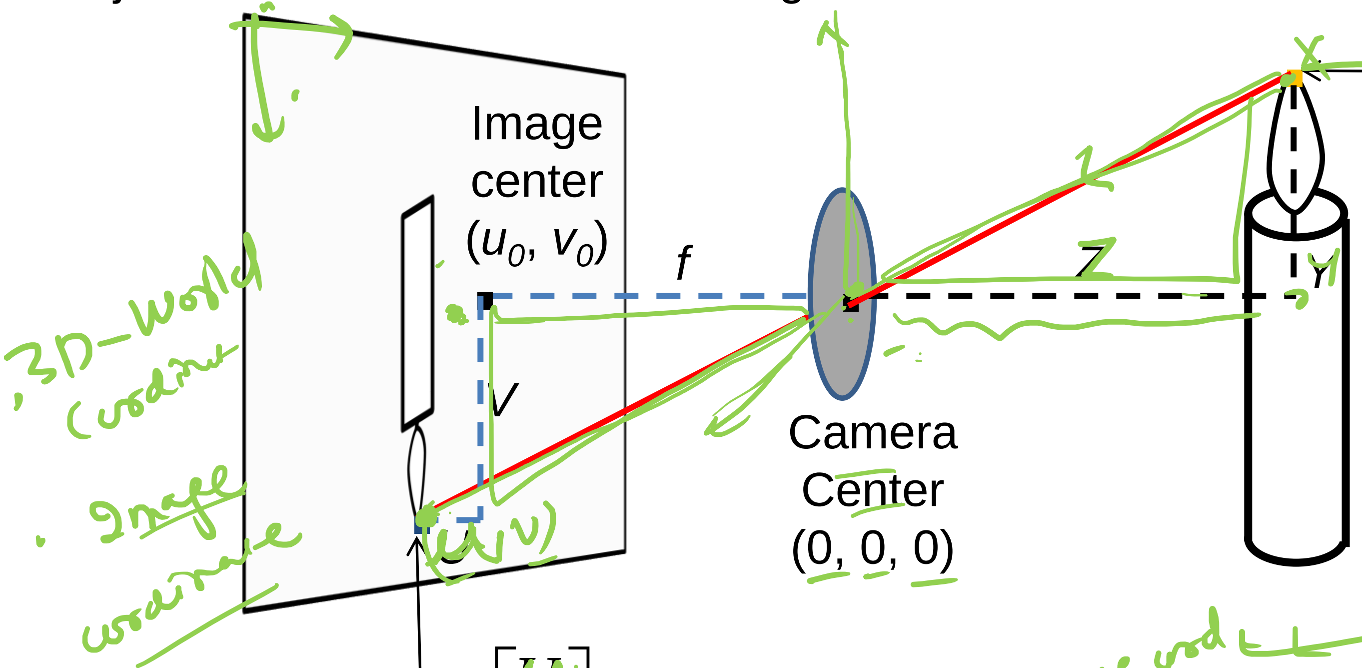
c = Optical center of the camera

Figure from Forsyth

Perspective Projection

3D-world uppercase

Projection: world coordinates→image coordinates



$$\mathbf{P} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

3D world coordinates

$$\mathbf{p} = \begin{bmatrix} U \\ V \end{bmatrix}$$

p = distance from image center

image plane coordinate

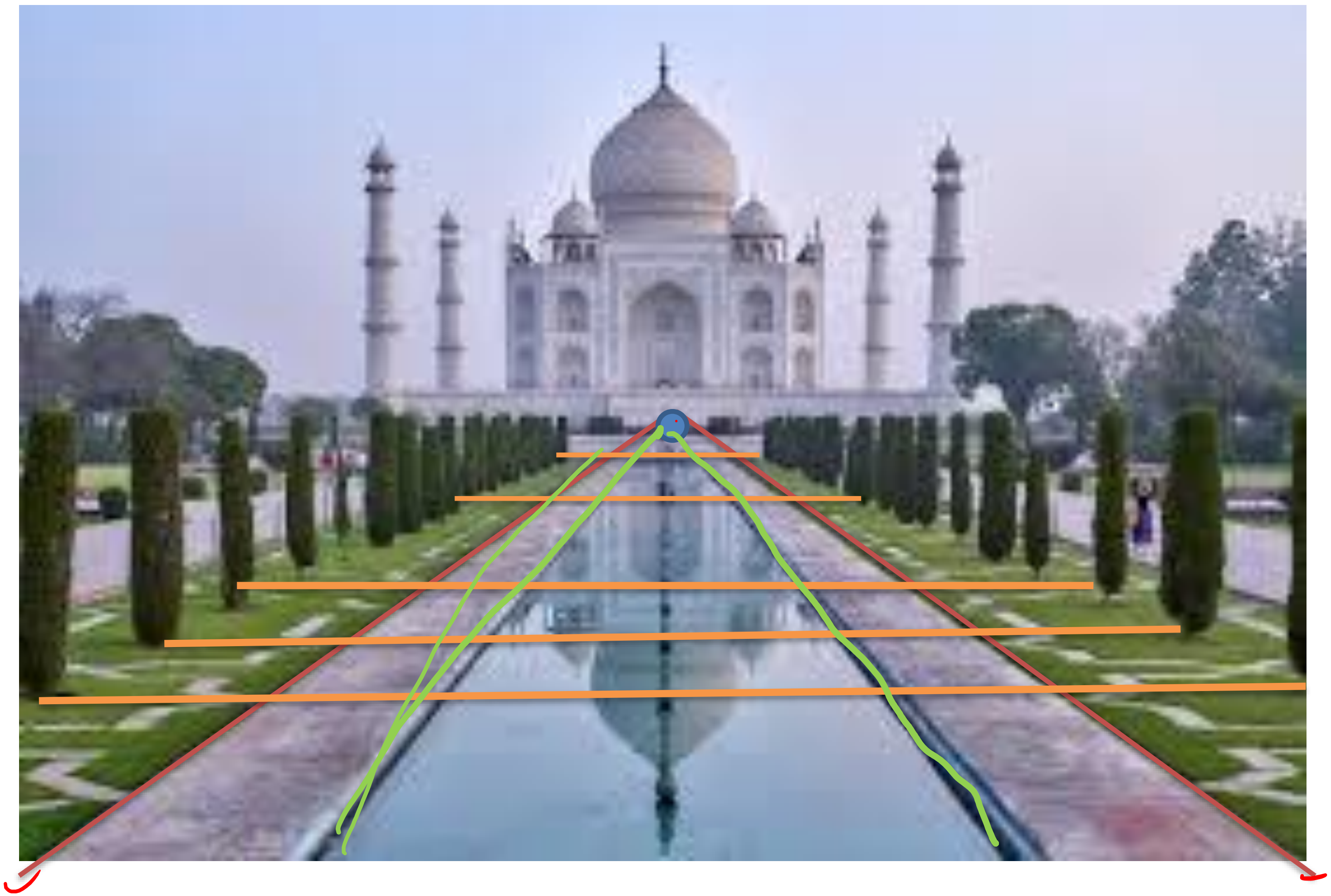
image cord

$$U = -X * \frac{f}{Z} \qquad V = -Y * \frac{f}{Z}$$

What is the effect if f and Z are equal?

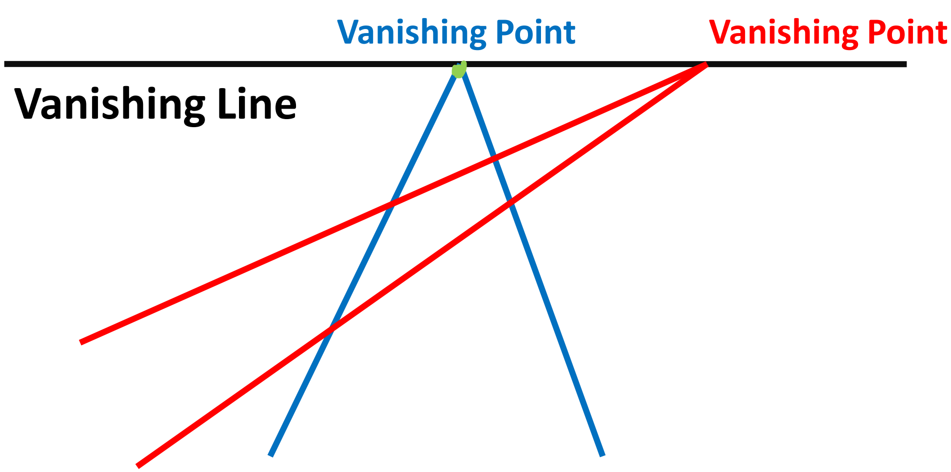
Taj Mahal

Taj Mahal



Perspective Transforms

- Parallel lines in the world intersect in the projected image at a “vanishing point”.
- Parallel lines on the same plane in the world converge to vanishing points on a “vanishing line”.



Perspective Projection:

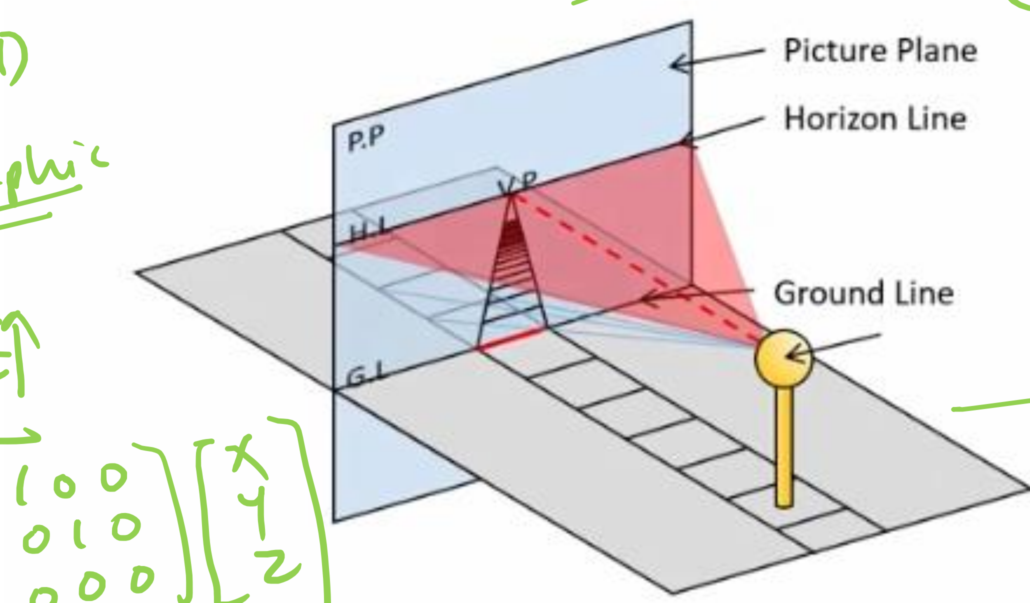
- https://www.youtube.com/watch?v=17kqhGRDHc8

3D Transfm — 2D Transf

↳ Rotn
↳ Scaling
↳ Sheer
↳ Trans
↳ Affine

3D → 2D

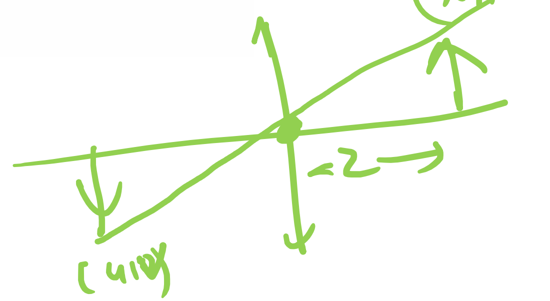
orthographic



$$\begin{bmatrix} P_x \\ P_y \\ P_{z=0} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

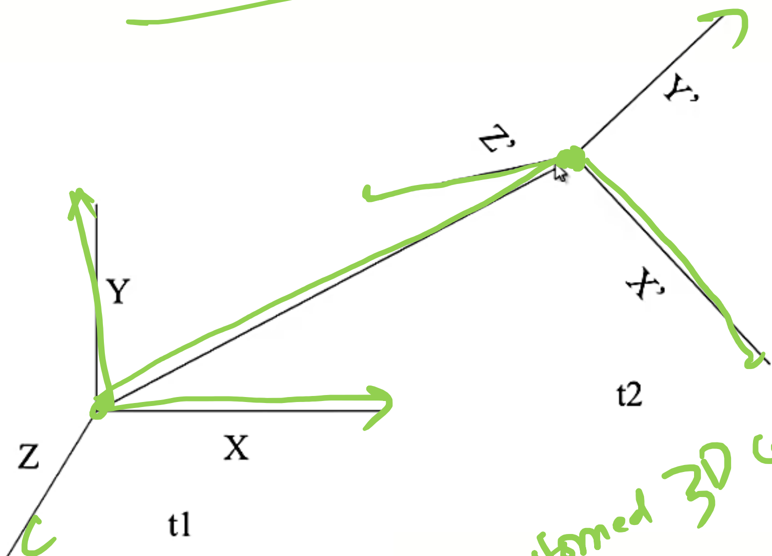
Perspective Pr

3D → 2D



3D Rigid Body Transform

2D- Euclid
(Rot + transl)



Transformed 3D coordinates

ori coord 3D

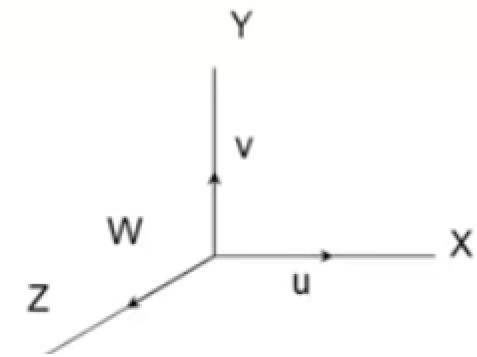
$$\begin{bmatrix}X'\\Y'\\Z'\end{bmatrix} = R\begin{bmatrix}X\\Y\\Z\end{bmatrix} + T = \begin{bmatrix}r_{11} & r_{12} & r_{13}\\ r_{21} & r_{22} & r_{23}\\ r_{31} & r_{32} & r_{33}\end{bmatrix}\begin{bmatrix}X\\Y\\Z\end{bmatrix} + \begin{bmatrix}T_X\\T_Y\\T_Z\end{bmatrix}$$

Rotation matrix (9 unknowns)

Translation (3 unknowns)

Rotation

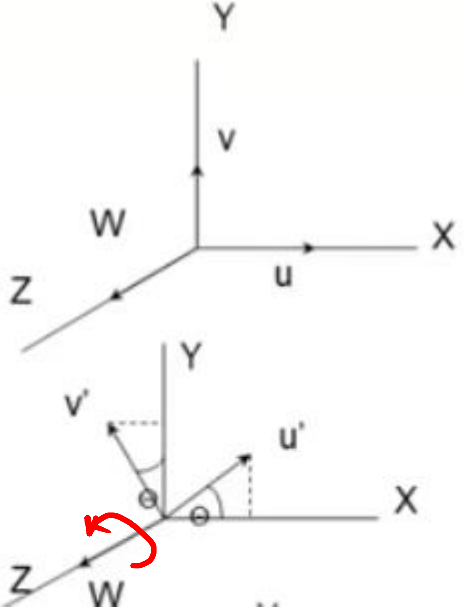
$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



Rotation

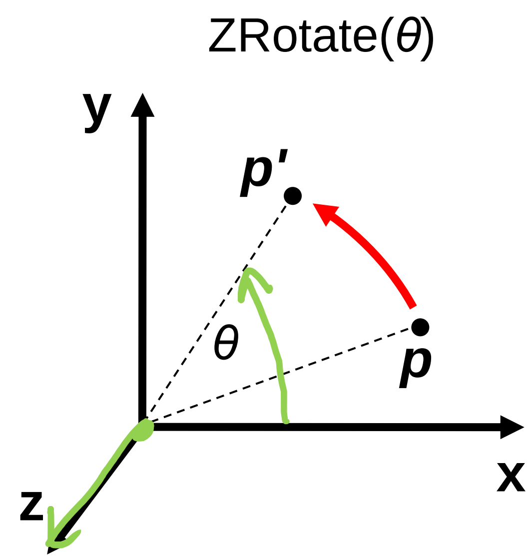
$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R = \begin{bmatrix} \cos\Theta & -\sin\Theta & 0 \\ \sin\Theta & \cos\Theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



Rotation

- About z axis



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

Rotation

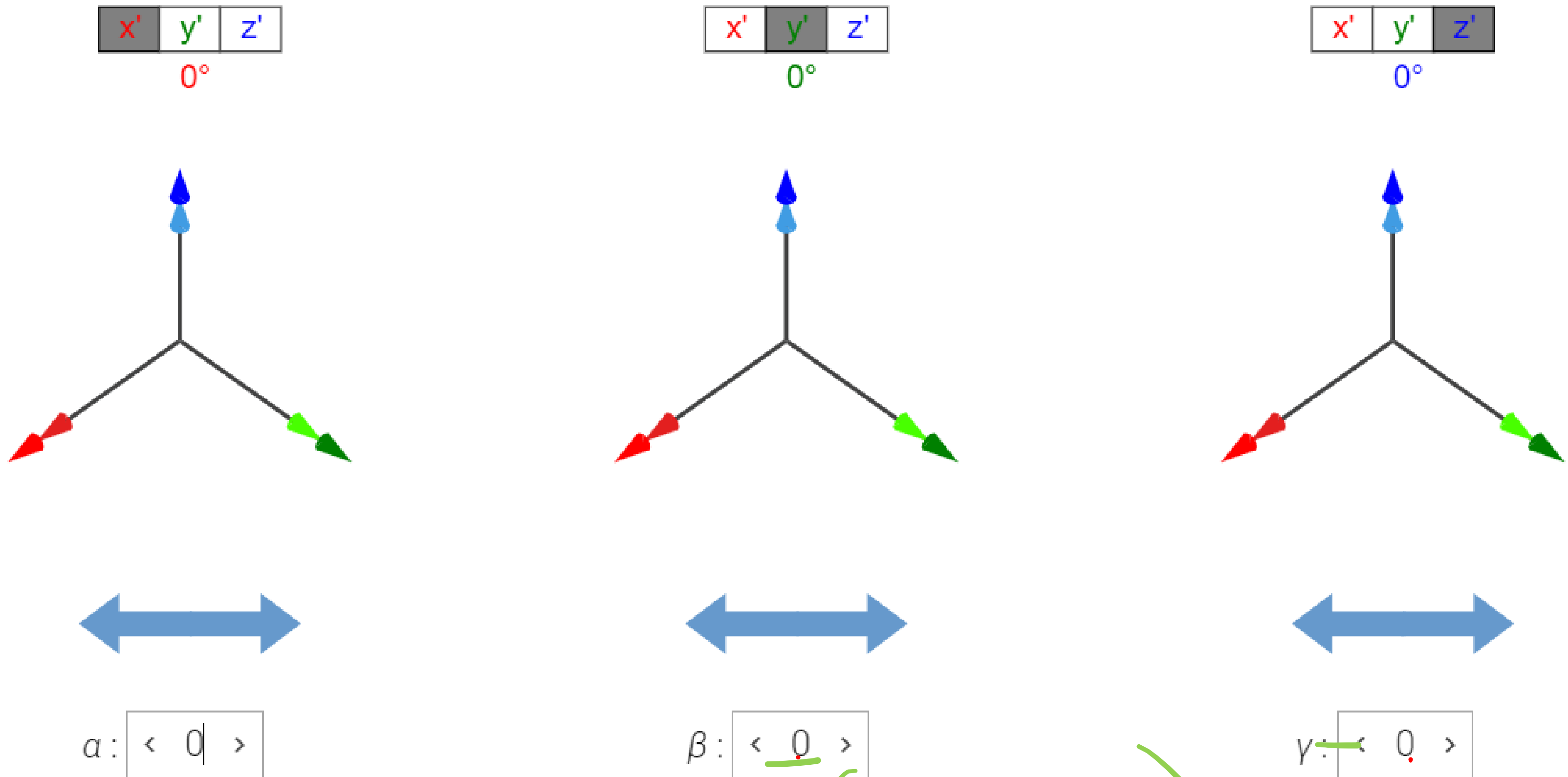
- About x axis:

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

- About y axis:

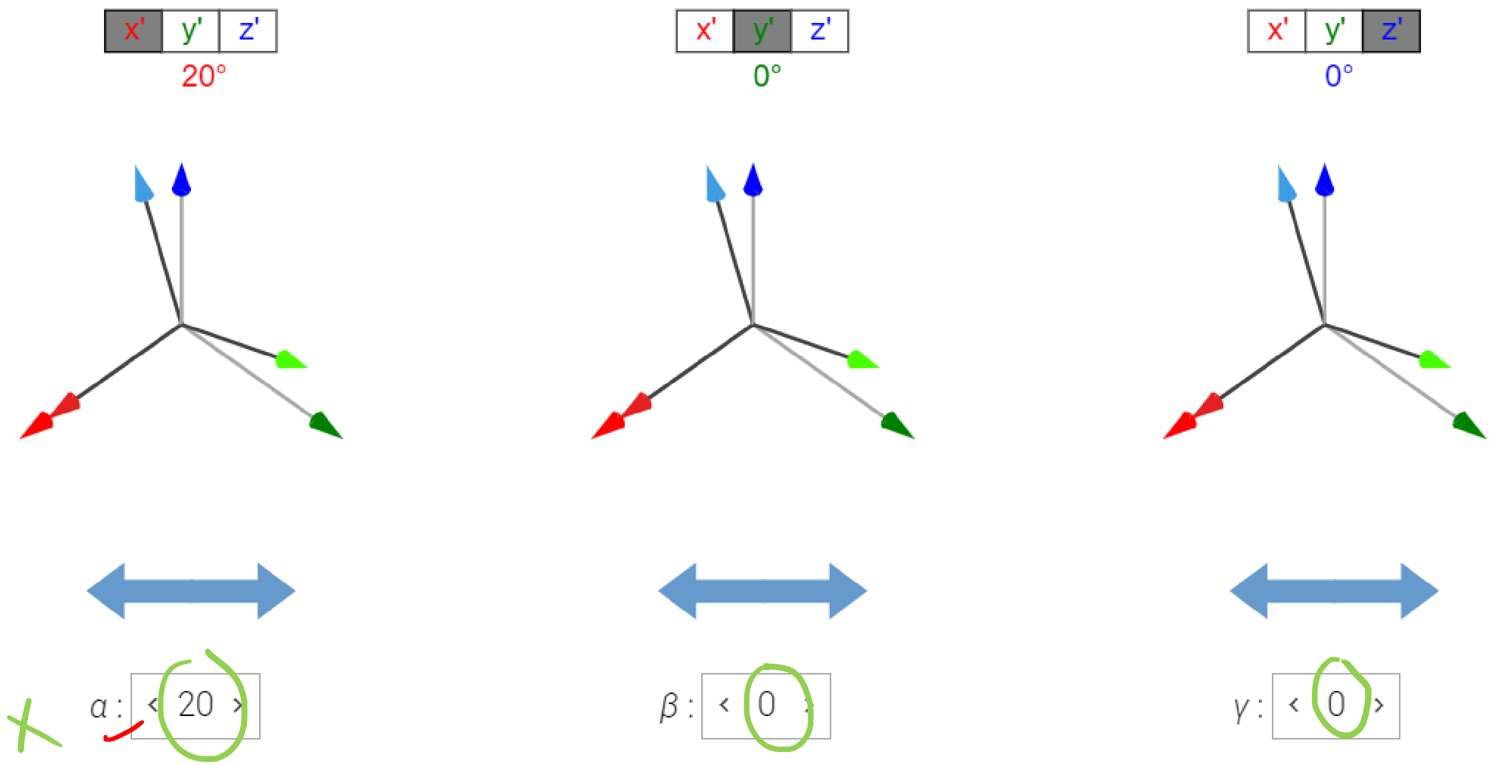
$$\begin{pmatrix} x' \\ \boxed{y'} \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 1 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

Euler Angles



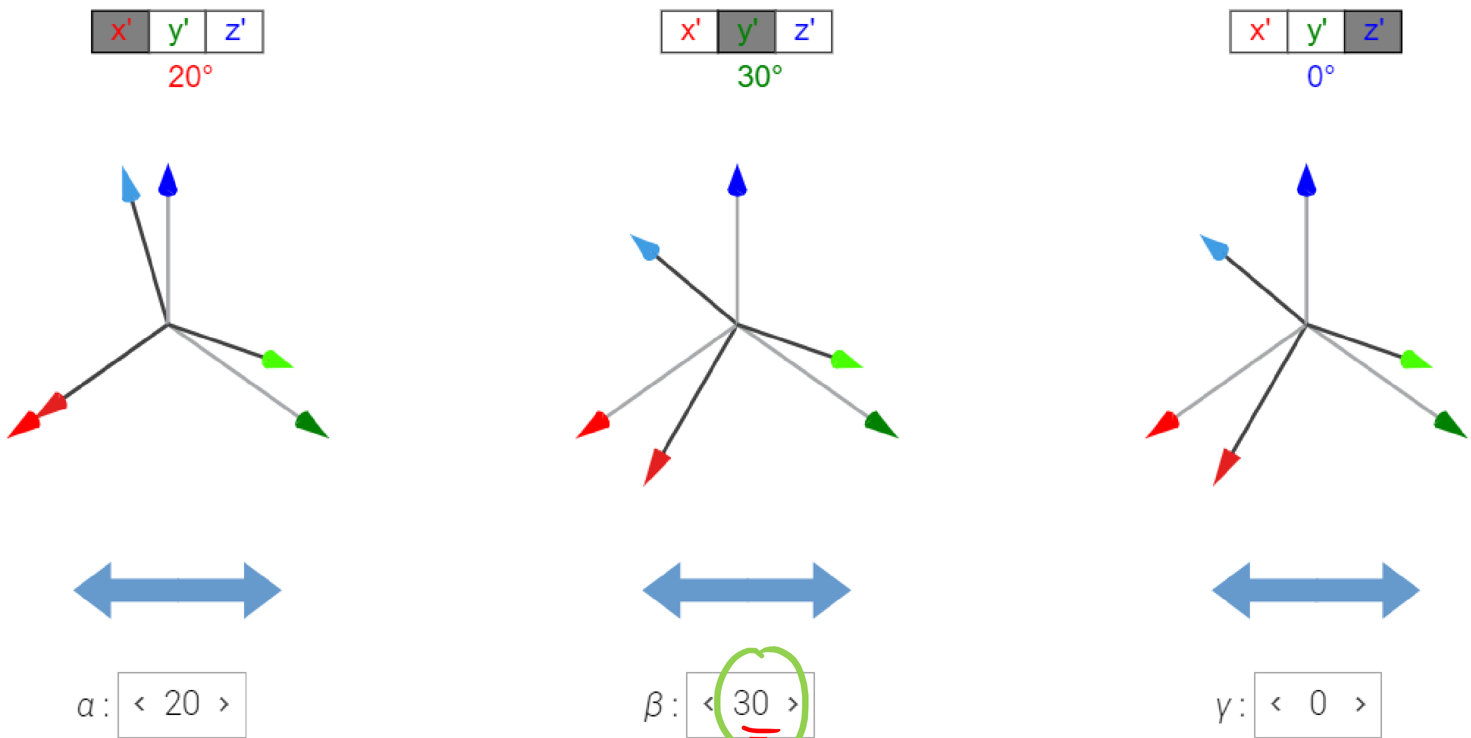
$$\mathbf{R} = \mathbf{R}_x(0°)\ \mathbf{R}_y(0°)\ \mathbf{R}_z(0°) = \begin{bmatrix} 1.000 & 0.000 & 0.000 \\ 0.000 & 1.000 & 0.000 \\ 0.000 & 0.000 & 1.000 \end{bmatrix}$$

Euler Angles



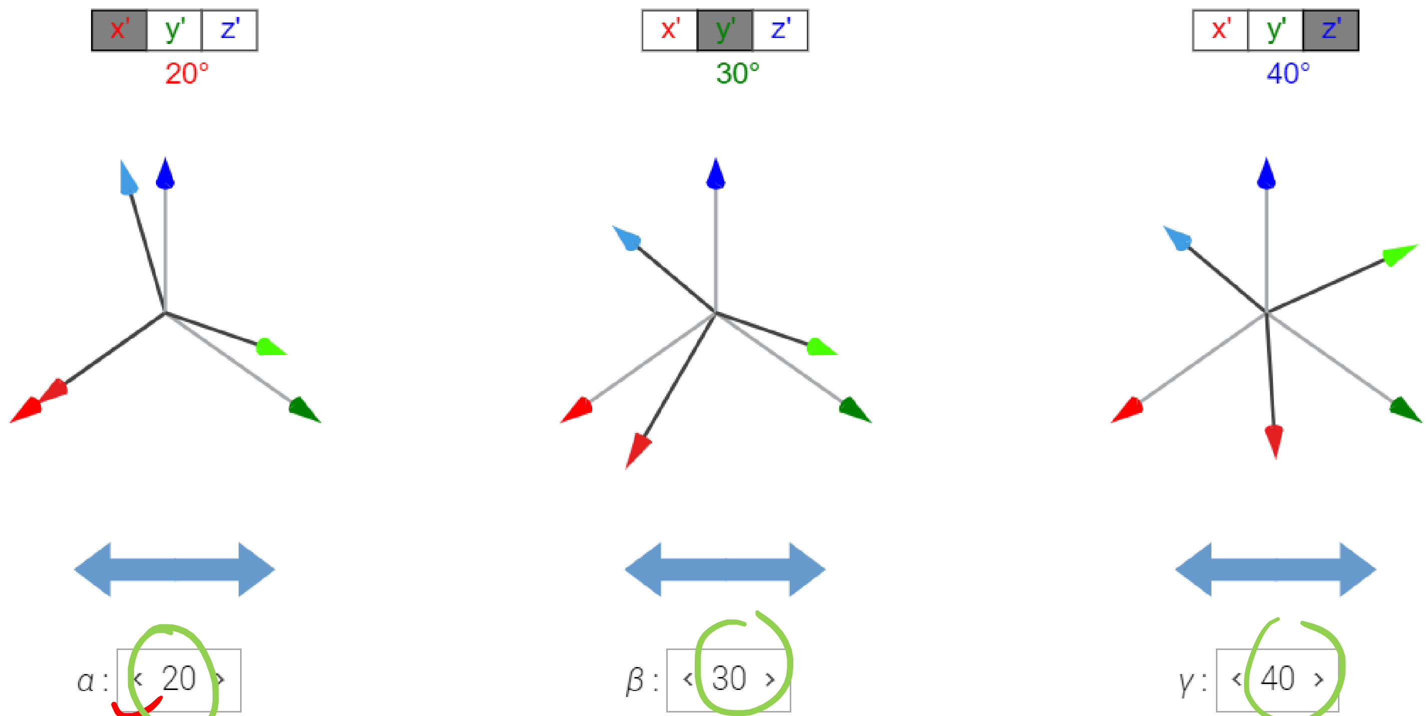
$$\mathbf{R} = \mathbf{R}_x(20°)\ \mathbf{R}_y(0°)\ \mathbf{R}_z(0°) = \begin{bmatrix} 1.000 & 0.000 & 0.000 \\ 0.000 & 0.940 & -0.342 \\ 0.000 & 0.342 & 0.940 \end{bmatrix}$$

Euler Angles



$$\mathbf{R} = \mathbf{R}_x(20°)\,\mathbf{R}_y(30°)\,\mathbf{R}_z(0°) = \begin{bmatrix} 0.866 & 0.000 & 0.500 \\ 0.171 & 0.940 & -0.296 \\ -0.470 & 0.342 & 0.814 \end{bmatrix}$$

Euler Angles



$$\mathbf{R} = \mathbf{R}_x(20°)\,\mathbf{R}_y(30°)\,\mathbf{R}_z(40°) = \begin{bmatrix} 0.663 & -0.557 & 0.500 \\ 0.735 & 0.610 & -0.296 \\ -0.140 & 0.564 & 0.814 \end{bmatrix}$$

Euler Angles

$$R = R_Z^{\alpha} R_Y^{\beta} R_X^{\gamma} = \underset{\alpha}{\overset{x}{\begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}}} \underset{\beta}{\overset{y}{\begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}}} \underset{\gamma}{\overset{z}{\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{bmatrix}}}$$

$$R = R_Z^{\alpha} R_Y^{\beta} R_X^{\gamma} = \begin{bmatrix} \cos\alpha\cos\beta & \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma & \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma \\ \sin\alpha\cos\beta & \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma & \sin\alpha\sin\beta\cos\gamma - \cos\alpha\sin\gamma \\ -\sin\beta & \cos\beta\sin\gamma & \cos\beta\cos\gamma \end{bmatrix}$$

$\Downarrow$ if angles are small $\cos\Theta \approx 1$ $\sin\Theta \approx \Theta$

$$R = \begin{bmatrix} 1 & -\alpha & \beta \\ \alpha & 1 & -\gamma \\ -\beta & \gamma & 1 \end{bmatrix}$$

Important Definitions

- **Frame of reference:** a measurements are made with respect to a particular coordinate system called the frame of reference.
- **World Frame:** a fixed coordinate system for representing objects (points, lines, surfaces, etc.) in the world.
- **Camera Frame:** coordinate system that uses the camera center as its origin (and the optic axis as the Z-axis)
- **Image or retinal plane:** plane on which the image is formed, note that the image plane is measured in camera frame coordinates (mm)
- **Image Frame:** coordinate system that measures pixel locations in the image plane.
- ***Intrinsic Parameters:*** Camera parameters that are internal and fixed to a particular camera/digitization setup
- ***Extrinsic Parameters:*** Camera parameters that are external to the camera and may change with respect to the world frame.