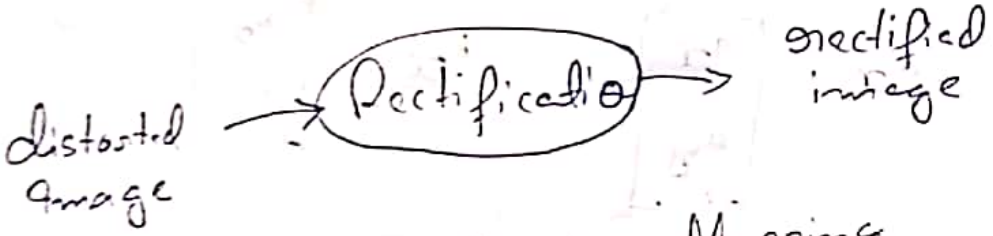
Geometric Transformations

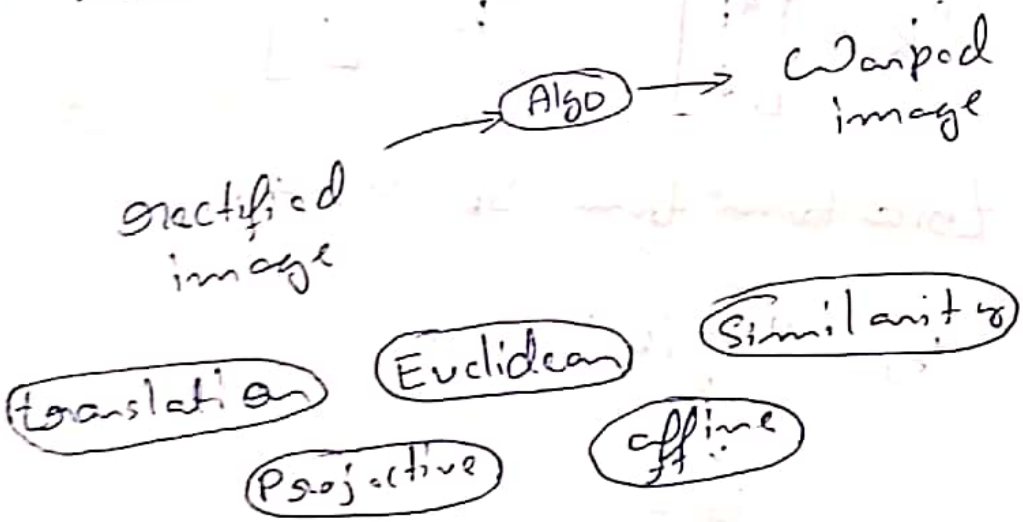
Why Geometric transformation

① Rectification

Process of correcting an image distortion by transforming the image.



② Rendering or Texture Mapping



③ Registration

{ Alignment of two images }.

* Geometric Transformation between Images

⇒ Every image has an own coordinate system.

- Image $a(i,j)$ or $a(x,y)$ with c.s S_a.
- Image $b(i,j)$ or $b(x,y)$ with c.s S_b
- Coordinate:

$$ {}^{a}x = \begin{bmatrix} {}^{a}x \\ {}^{a}y \end{bmatrix} \quad \ldots \quad {}^{b}x = \begin{bmatrix} {}^{b}x \\ {}^{b}y \end{bmatrix} $$

$$ {}^{a}K = \begin{bmatrix} {}^{a}i \\ {}^{a}j \end{bmatrix} \qquad {}^{b}K = \begin{bmatrix} {}^{b}i \\ {}^{b}j \end{bmatrix} $$

Goal: transform from S_b to S_a

$$ {}^{a}x = {}^{a}T_{b}\,({}^{b}x) $$

(To: a, From: b)

$$ \boxed{ {}^{b}T_{a}\, {}^{a}T_{c} = {}^{b}T_{c} } $$

* Affine Transformation

$$ \begin{bmatrix} {}^{a}x_c \\ {}^{b}x_c \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} {}^{b}x \\ {}^{b}y \end{bmatrix} + \begin{bmatrix} h_{13} \\ h_{23} \end{bmatrix} $$

* Resampling

⇒ The transformation leads to non-integer coordinates

⇒ To assign intensity value from the input to the output image, we need to interpolate.

⇒ Performing a discretization and quantization (again) is called resampling.

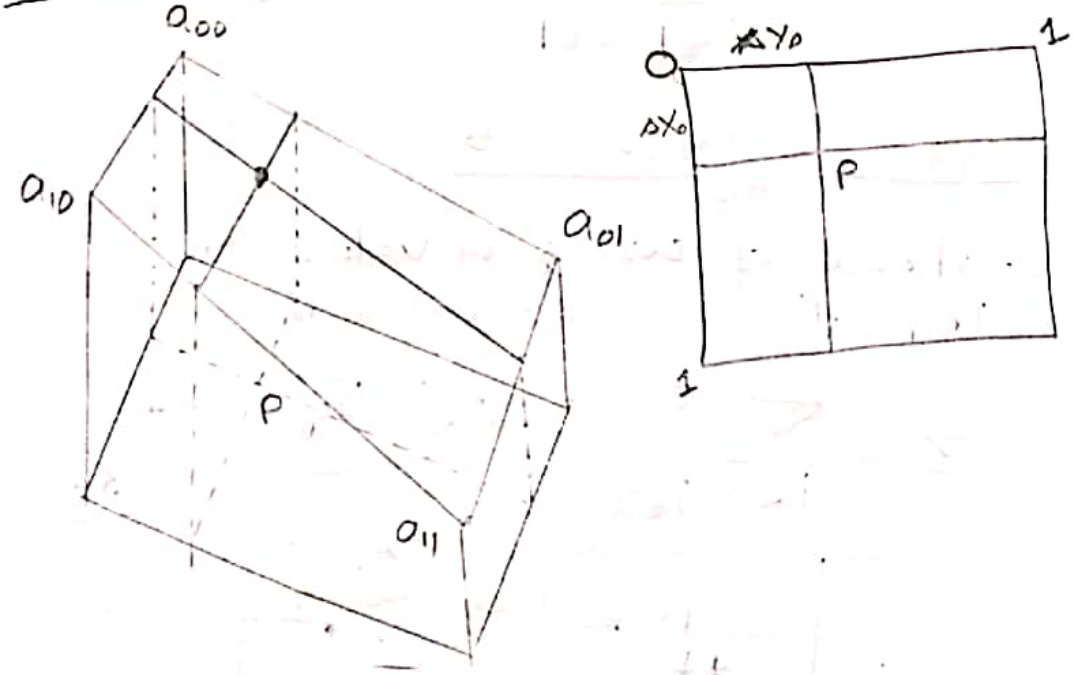
* Nearest Neighbor Interpolation

⇒ Choose the same value as the closest pixel.

⇒ Results in rounding two values

$$b(^bx, ^by) = a([^ax], [^ay])$$

* Bilinear Interpolation



⇒ Linear interpolation in x-direction, followed by the Y-direction

$$b(^b x, ^b y) = a_{00}(1-\Delta x)(1-\Delta y) + a_{01}(1-\Delta x)\Delta y + a_{10}\Delta x(1-\Delta y) + a_{11}\Delta x \Delta y$$

$$= a_{00} + \Delta x (a_{10} - a_{00}) + \Delta y (a_{01} - a_{00}) + \Delta x \Delta y (a_{00} - a_{01} - a_{10} + a_{11})$$

C_{00} C_{10} C_{11} C_{01}

$$Z = \sum_{i \le 1} \sum_{j \le 1} C_{ij} \Delta x^i \Delta y^j$$

★ Bicubic Interpolation.

⇒ Instead of using 4 values for interpolation, we are using 16 values.

$$Z = \sum_{i \le 3} \sum_{j \le 3} C_{ij} \Delta x^i \Delta y^j$$

	Speed	Quality
NN	++	−
Bilinear	+	0
Bicubic	−	++

Geometric Transformation

⇒ Inverse method allows for the direct application of bilin/cubic interpolation

⇒ Always use Inverse Warping

* Image Half-Sizing

→ Aliasing artifacts
→ lost details

⇒ Apply a Gaussian filter before Subsampling

How much Smoothing is Needed

→ Depends on the Kernel
→ Depends on the width of the Kernel
→ Depends on the scale of the transformation.

Consider the Scale for Resampling

- $m < 1$: Image becomes smaller
  Recommendation $\sigma \approx 0.5$
- $m = 1$: Scale same
  Use bilinear interpolation
  (bicubic for high quality results)
- $m > 1$: Image becomes large
  Use bicubic interpolation

★ Image Pyramid

- A list of images
- Each image has half of the size of its predecessor.