Image Processing

INT3404 1/ INT3404E 21

Giảng viên: TS. Nguyễn Thị Ngọc Diệp

Email: ngocdiep@vnu.edu.vn

1

Schedule

Tuần	Nội dung	Yêu cầu đối với sinh viên (ngoài việc đọc tài liệu tham khảo)
1	Giới thiệu môn học	Cài đặt môi trường: Python 3, OpenCV 3, Numpy, Jupyter Notebook
2	Ảnh số (Digital image) – Phép toán điểm (Point operations) Làm quen với OpenCV + Python	
3	Điều chỉnh độ tương phản (Contrast adjust)– Ghép ảnh (Combining images)	Làm bài tập 1: điều chỉnh gamma tìm contrast hợp lý
4	Histogram - Histogram equalization	Thực hành ở nhà
5	Phép lọc trong không gian điểm ảnh (linear processing filtering)	Thực hành ở nhà
6	Phép lọc trong không gian điểm ảnh cont. (linear processing filtering) Thực hành: Ứng dụng của histogram; Tìm ảnh mẫu (Template matching)	Bài tập mid-term
7	Trích rút đặc trưng của ảnh Cạnh (Edge) và đường (Line) và texture	Thực hành ở nhà
8	Các phép biến đổi hình thái (Morphological operations)	Làm bài tập 2: tìm barcode
9	Chuyển đổi không gian – Miền tần số – Phép lọc trên miền tần số Thông báo liên quan đồ án môn học	Đăng ký thực hiện đồ án môn học
10	Xử lý ảnh màu (Color digital image)	Làm bài tập 3: Chuyển đổi mô hình màu và thực hiện phân vùng
11	Các phép biến đổi hình học (Geometric transformations)	Thực hành ở nhà
12	Nhiễu – Mô hình nhiễu – Khôi phục ảnh (Noise and restoration)	Thực hành ở nhà
13	Nén ảnh (Compression)	Thực hành ở nhà
14	Hướng dẫn thực hiện đồ án môn học	Trình bày đồ án môn học
15	Hướng dẫn thực hiện đồ án môn học Tổng kết cuối kỳ	Trình bày đồ án môn học

2

Recall week 10: Color image processing

- Color phenomenon
- Color matching/creation: additive vs substractive
- Attributes of colors:
  - Hue
  - Saturation, Chroma, Colorfulness
  - Brightness, Lightness, Value, Intensity
- Color space: RGB, HSV, CYMK, YCbCr
- Image segmentation using color information

3

Frequency: Highpass ~ edge

$$F[k,l] = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f[m,n] e^{-j2\pi\left(\frac{k}{M}m+\frac{l}{N}n\right)}$$

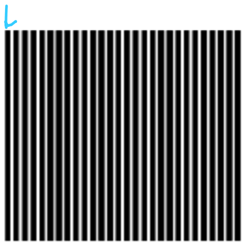
fast change
edge

- ”Frequency”:
  - Frequency of changing
  - Frequency of events

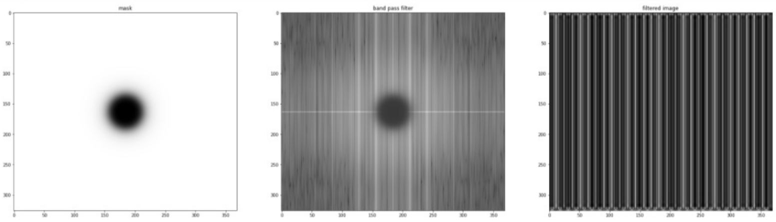
fix duration

4

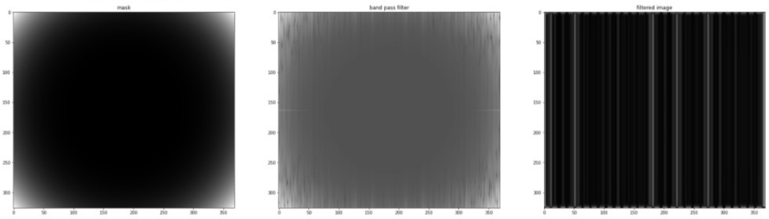
High pass



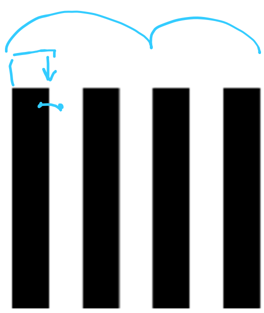
```
1 filter_img('highfreq.png', 30)
```



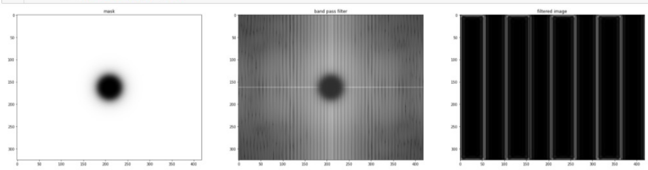

```
1 filter_img('highfreq.png', 400)
```



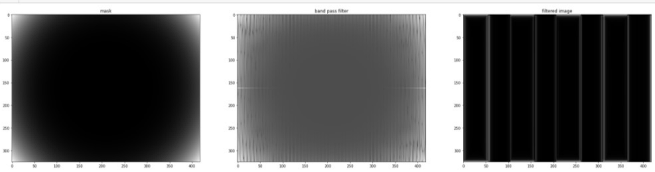
5



```
1 filter_img('lowfreq.png', 30)
```




```
1 filter_img('lowfreq.png', 400)
```



6

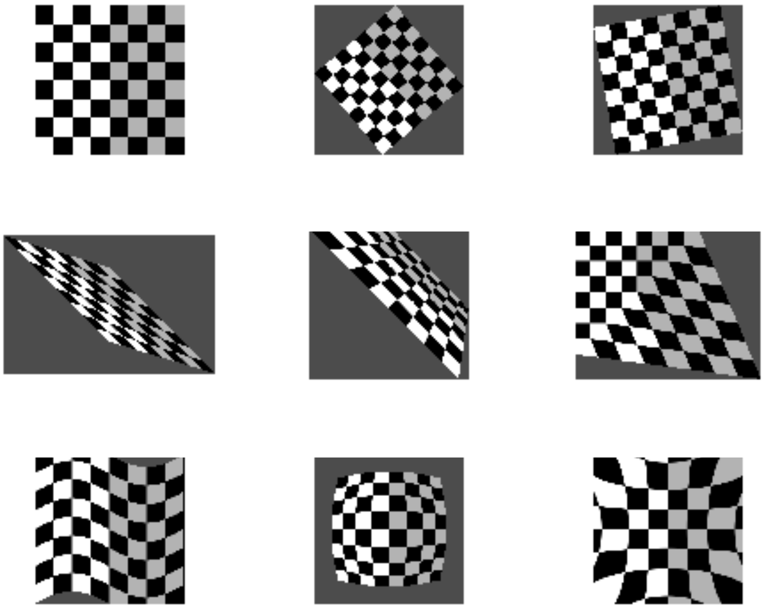
Geometric transformation

7

What is Geometric transformation?

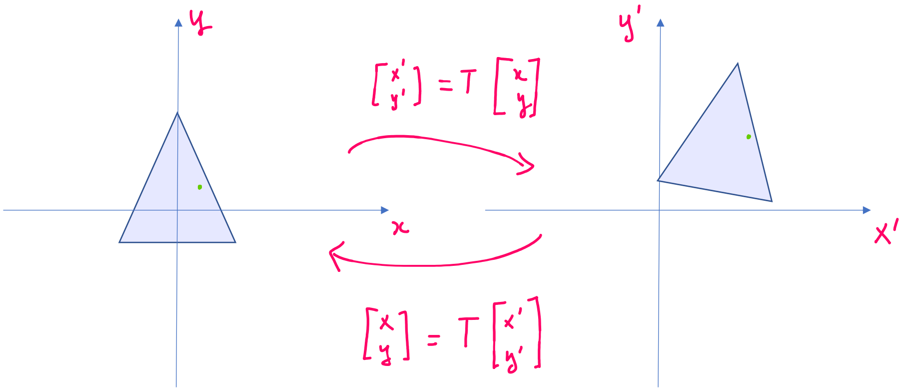
- So far, image processing operations → modify the color values of pixels in a given image
- Geometric transformation → modify the positions of pixels in an image, but keep the color unchanged
- Purpose:
  - To create special effects
  - To register two images taken of the same scene at different times
  - To morph one image to another

8



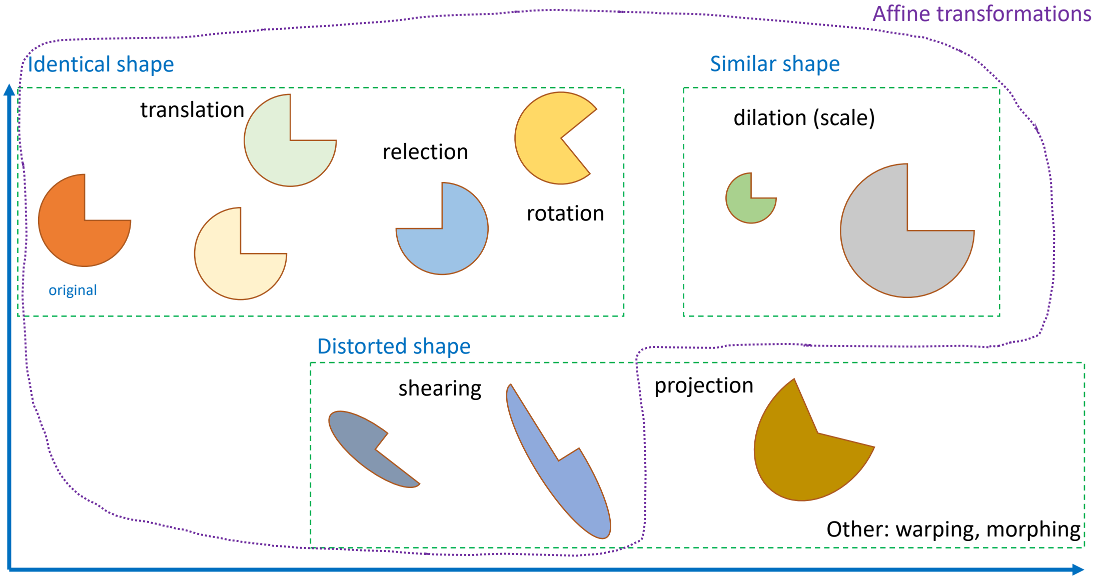
9

Illustration of mapping functions



10

Geometric transformations



11

Two basic operations of geometric transformation

1. Spatial transformation of coordinates

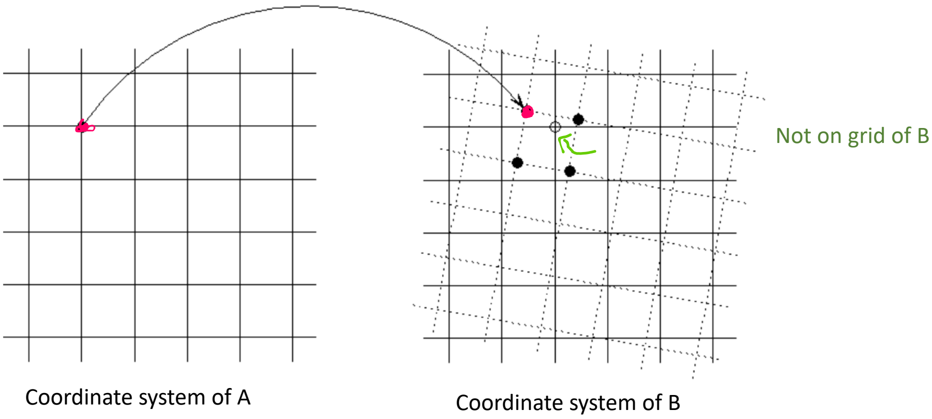
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \mathbf{T} \begin{bmatrix} x \\ y \end{bmatrix}$$

2. Intensity interpolation that assigns intensity values to the spatially transformed pixels

12

Transforming coordinate

- Each point (x, y) is mapped to a point (x', y') in a new coordinate system



13

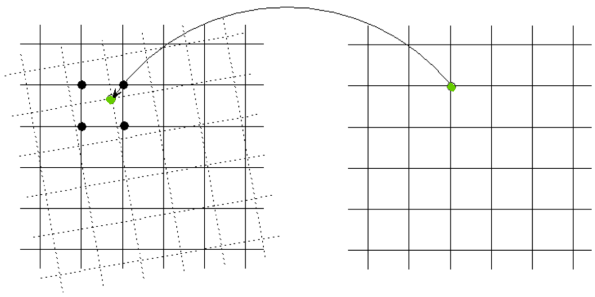
Interpolating values

- To find the values on the grid points
- Finding the closest projected points to a given grid point can be computationally expensive
- ➔ Inverse projection

14

Inverse mapping

- Projecting the grid of B into the coordinate system of A
  - Known image values on a regular grid
  - Simple to find the nearest points for each interpolation calculation

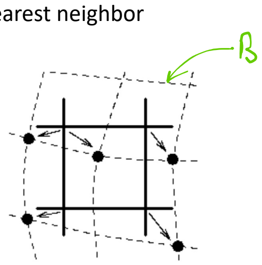


15

Intensity interpolation

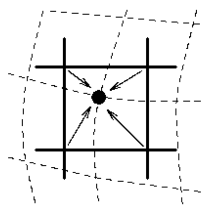
Use the distance-weighted average of nearest values to estimate new value

Nearest neighbor



Assigns to the new location the intensity of its nearest neighbor

Bilinear

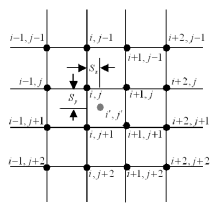


Use four nearest neighbors to estimate the intensity at a given location

$$v(x,y) = ax + by + cxy + d$$

-> solve for a, b, c, d

Bicubic

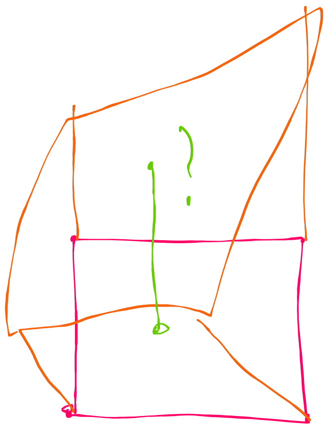


Use sixteen nearest neighbors to estimate the intensity at a given location

$$v(x,y) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} x^i y^j$$

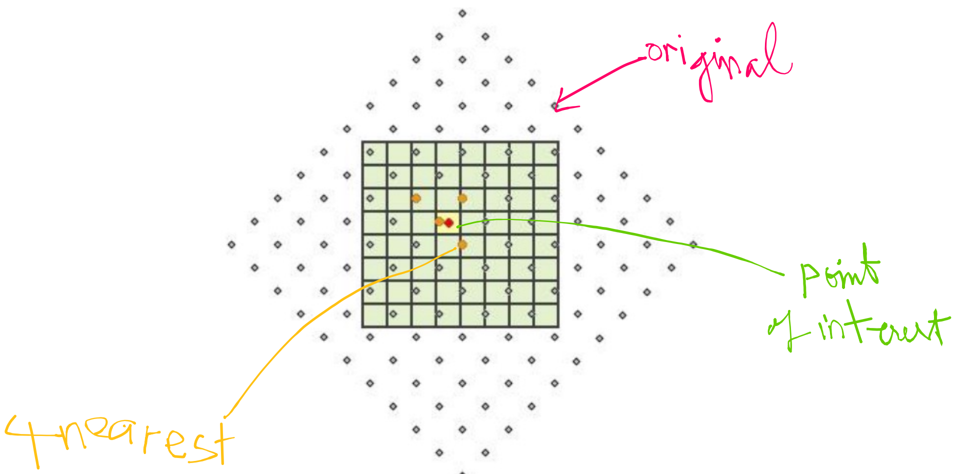
-> solve for a_{ij}

16



- Terrain surface is continuous and smooth
- High correlation between neighboring data points

17



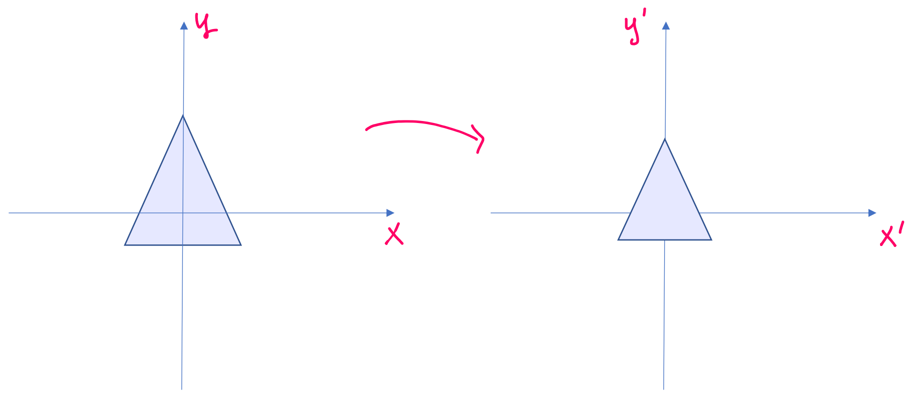
18

Image mapping A to B

- If the projection from B to A is known then we can
  1. Project the coordinates of the B pixels onto the A pixel grid
  2. Find the nearest A grid points for each projected B grid point
  3. Compute the value for the B pixels by interpolation of the A image

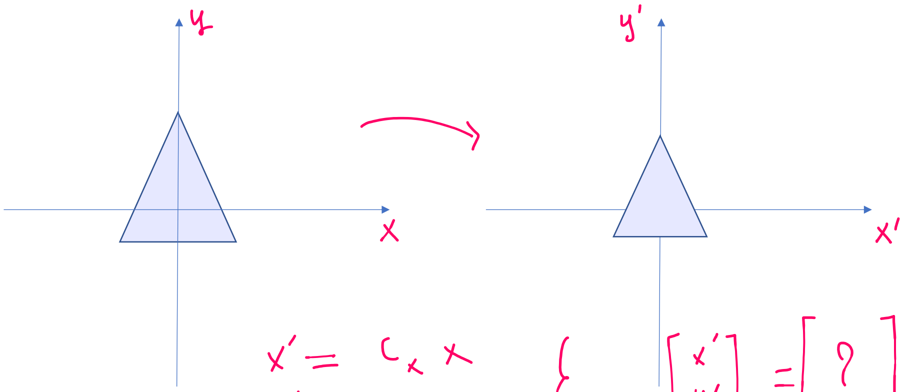
19

Scaling



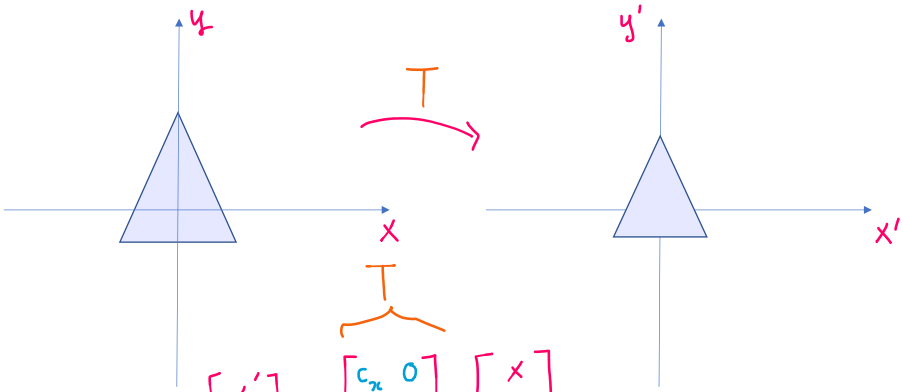
20

Scaling



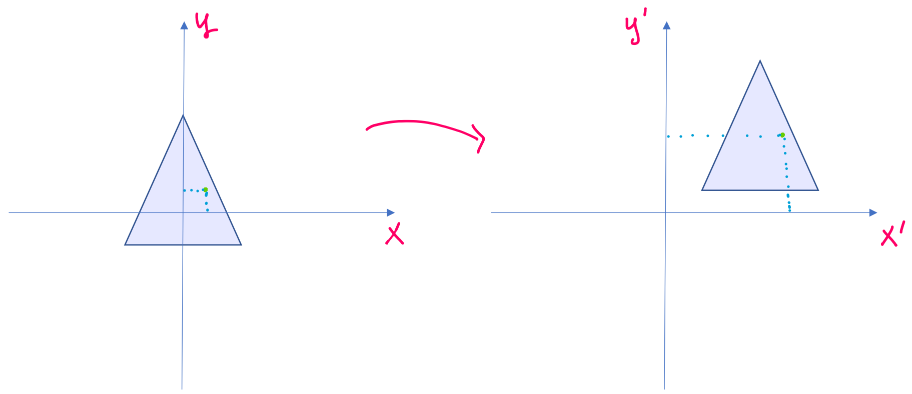
$$\left.\begin{aligned} x' &= c_x\, x \\ y' &= c_y\, y \end{aligned}\right\} \quad \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} ? \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Scaling



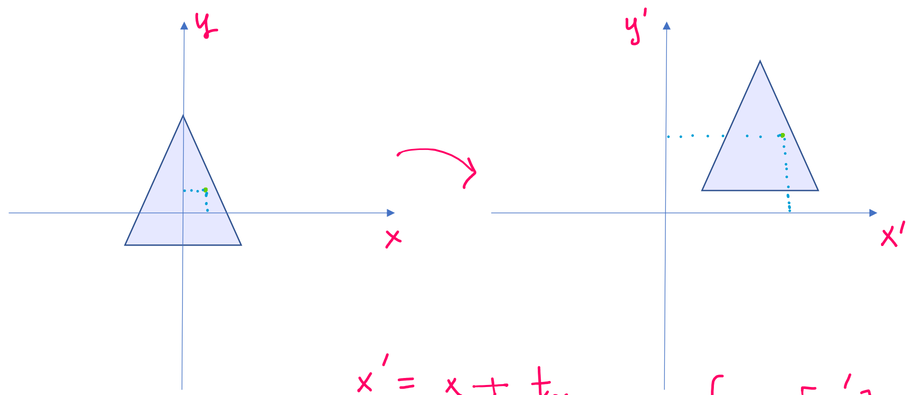
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} c_x & 0 \\ 0 & c_y \end{bmatrix}}_{T} \begin{bmatrix} x \\ y \end{bmatrix}$$

Translation



23

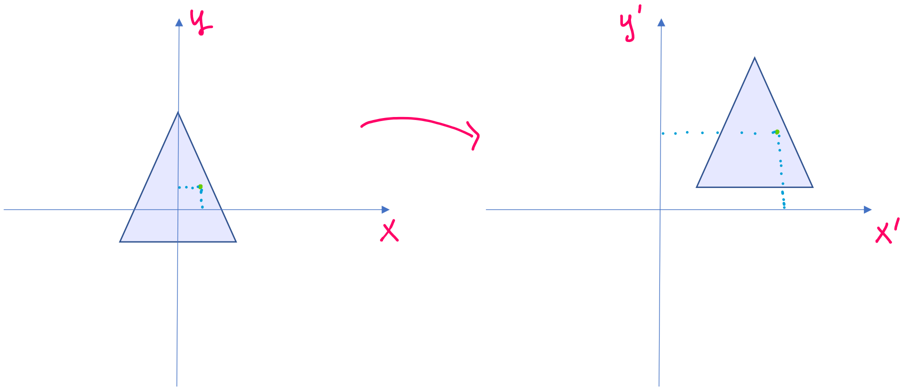
Translation



$$\left\{\begin{aligned} x' &= x + t_x \\ y' &= y + t_y \end{aligned}\right. \qquad \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \ ? \ \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

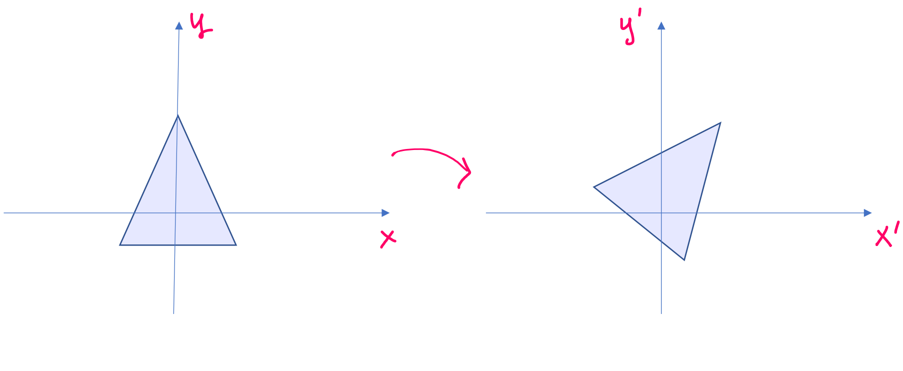
24

Translation

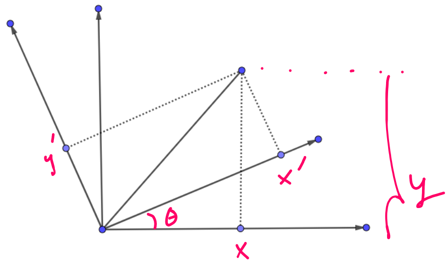


25

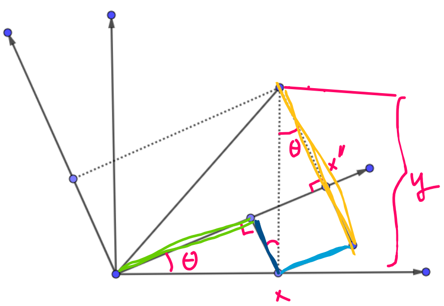
Rotation



26



27



$$x' = x\cos\theta + y\sin\theta$$

$$y' = y\cos\theta - x\sin\theta$$

28

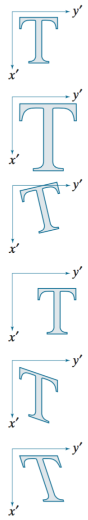
Affine transformation

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

Expressed using a 3x3 matrix

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \mathbf{A}\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

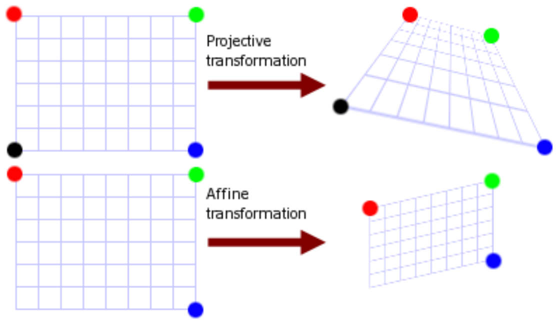
Transformation Name	Affine Matrix, A	Coordinate Equations	Example
Identity	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x$ $y' = y$	
Scaling/Reflection (For reflection, set one scaling factor to −1 and the other to 0)	$\begin{bmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = c_x x$ $y' = c_y y$	
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)	$\begin{bmatrix} 1 & s_v & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x + s_v y$ $y' = y$	
Shear (horizontal)	$\begin{bmatrix} 1 & 0 & 0 \\ s_h & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x$ $y' = s_h x + y$	



Example of interpolation

Linear transformation

- Two types of linear transformation:
  - Affine transformation:
    - Preserve parallelism, length, and angle
  - Projective transformation:
    - Preserve collinearity, and incidence



Source image: https://www.graphicsmill.com/docs/gm5/Transformations.htm

31

General tranformation matrix

Translation vector

Rotation matrix

$$\begin{bmatrix} a1 & a2 & b1 \\ a3 & a4 & b2 \\ c1 & c2 & 1 \end{bmatrix}$$

Projection vector

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} \cong \begin{bmatrix} wx' \\ wy' \\ w \end{bmatrix} = \begin{bmatrix} a1 & a2 & b1 \\ a3 & a4 & b2 \\ c1 & c2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Also referred to as “homography matrix”

More on projective transformation: https://mc.ai/part-ii-projective-transformations-in-2d/

32

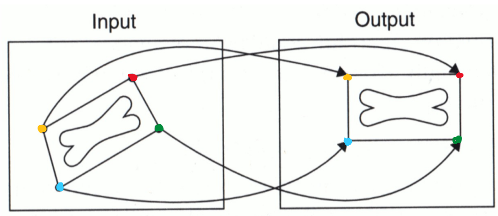
General transformations

- Control points
- Projective transformation
- Image warp and image morphing

33

General transformations

- Till now: rotation, scaling, translation (shift), projective transformation are all linear in x and y
- More general transformations: polynormial transformations
  - Order is higher than 1
- Extract transformation is derived from control points
  - a point in the input image and its corresponding point in the output image



Ref: http://www.iup.uni-bremen.de/~melsheim/dip/dipSS17-L3.pdf

34

Polynomial transformation

- Assume a(x,y) and b(x, y) as polynomials of order N with unknown coefficients

$$\begin{aligned} x' &= a(x,y) = \sum_{i=0}^{N} \sum_{j=0}^{N-i} a_{ij} x^i y^j \\ y' &= b(x,y) = \sum_{i=0}^{N} \sum_{j=0}^{N-i} b_{ij} x^i y^j \end{aligned}$$

Determining the coefficients requires at least as many control points as the polynomials have coefficients.

35

Polynomial transformation with order N=2

$$\begin{aligned} x' &= a(x,y) = a_{00} + a_{10}x + a_{01}y + a_{11}xy + a_{20}x^2 + a_{02}y^2 \\ y' &= b(x,y) = b_{00} + b_{10}x + b_{01}y + b_{11}xy + b_{20}x^2 + b_{02}y^2 \end{aligned}$$

a_{00}, b_{00}: Shift vector
a_{10}, b_{01}: Linear scaling in x, y direction
a_{01}, b_{10}: Shear in x, y direction[1]
a_{11}, b_{11}: y-dependent scale in x, x-dependent scale in y
a_{20}, b_{02}: non-linear (quadratic) scale in x, y

[1] A rotation can be described as a combination of shear and linear scaling first in one, then the other coordinate: Any Rotation by angle $\theta \neq \pm 90°$ can be decomposed in the following way:

$$\begin{bmatrix} a_{10} & a_{01} \\ b_{10} & b_{01} \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} = \begin{bmatrix} 1/\cos\theta & \sin\theta/\cos\theta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -\sin\theta & \cos\theta \end{bmatrix} \quad (4.18)$$

The first (the rightmost one) is a 1D scale and shear in y, the second (the left one) is a 1D scale and shear in x.

36

Example of polynomial geometric warps

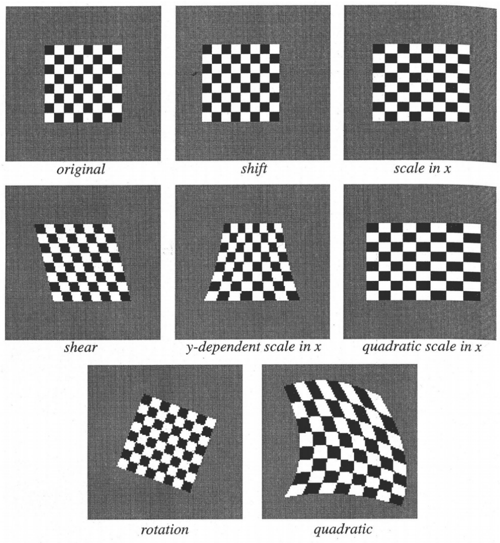


Fig. 4.9: *Some polynomial geometric warps (Fig 7-30 in Schowengerdt, 1997)*

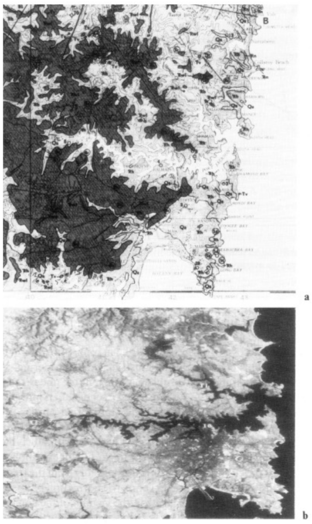
37

Applications of geometric transformation

- Geometric calibration/Image rectification
  - Remove camera-induced distortion, i.e., convert non-rectangular pixel coordinates to rectangular coordinates
- Image registration
  - Geometrically match two images or an image and a map; stationary objects should have same position in both images
- Map projections

38

Image registration example



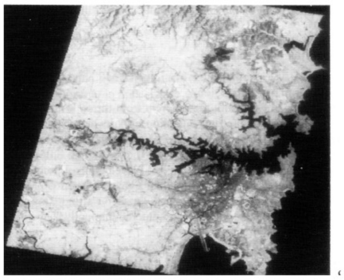


Fig. 4.10: *Image registration. (a) Map; (b) Landsat MSS image to be registered; (c) Landsat image registered to map using 2nd order polynomials (Fig. 2.16 from Richards, 1986)*

39

Example of image rectification

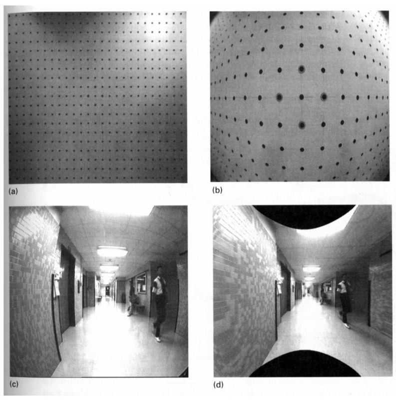


Fig. 4.8: *Geometric rectification of an image taken with a fish-eye lens: (a) test target, (b) fish-eye image of test target, (c) fish-eye image (d) rectified image (Fig 8.9 from Castleman, 1996)*

40

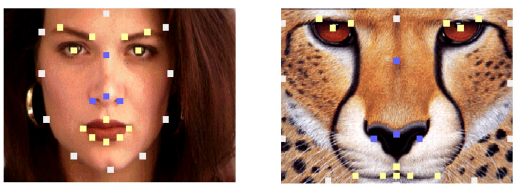
References

- Gonzalez book
- (Chap 4) Geometric transformations
  - http://www.iup.uni-bremen.de/~melsheim/dip/dipSS17-L3.pdf
- Image warping/morphing
  - https://www.csie.ntu.edu.tw/~cyy/courses/vfx/18spring/lectures/handouts/lec05_morphing.pdf

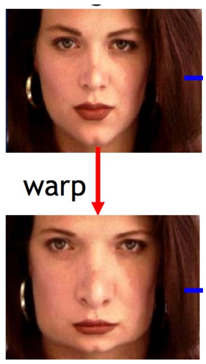
41

Mesh warping

Specify corresponding points



Interpolate to a complete warping function

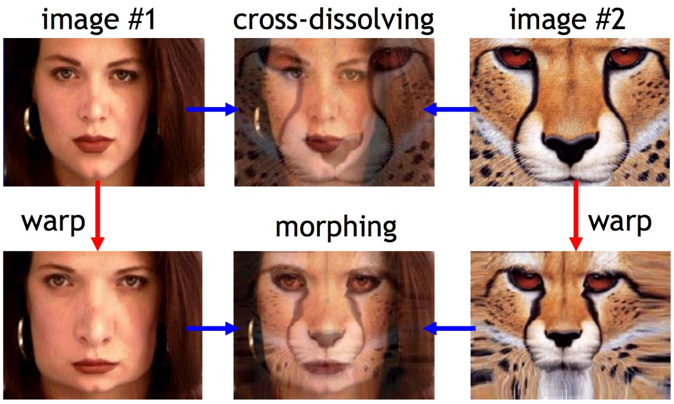


Convert to mesh warping

42

Image morphing

- To synthesize a fluid transformation from one image to another



43