Image Interpolation

Interpolation is the process of using known data to estimate values at unknown locations.

Nearest Neighbour Interpolation: Assigns new value to the location by using nearest neighbor in the original image.

- May lead to undesirable artifacts, such as severe distortion of straight edges

Bilinear Interpolation: Use four nearest neighbours to estimate the intensity at a given location.

Bicubic Interpolation: Involves 16 nearest neighbours of a point

★ Some basic Relationships between pixels

Neighbors of a pixel

A pixel p at coordinates (x, y) has two horizontal and two vertical neighbors with co-ordinates:

$$(x+1, y), (x-1, y), (x, y+1), (x, y-1)$$

This set of pixels, called 4-neighbors of p, is denoted $N_4(p)$.

The four diagonal neighbors of p have coordinates

$$(x+1, y+1), (x+1, y-1), (x-1, y+1), (x-1, y-1)$$

and are denoted by $N_D(p)$.

$N_4(p)$ and $N_D(p)$ together are called $N_8(p)$.

This is called neighborhood of p.

It is closed if it contains p, open if it doesnt.

Adjacency, connectivity, regions and boundaries

Connectivity/Adjacency: Two pixels that are neighbors and have the same gray-level/intensity level are called adjacent.

(a) 4-adjacency - Two pixels p and q with values from V are 4-adjacent if q is in the set $N_4(p)$.

Binary Image

$$\begin{matrix} 0 & 1 & 0 & 1 \\ 0 & 0 & \boxed{1} & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{matrix}$$

$V = \{1\}$

8-adjacency: 2 pixels p and q with values from V are 8-adjacent if q is in the set $N_8(p)$.

54	10	100	8
81	150	2	34
201	200	3	45
7	70	147	56

8-adjacency

$V = \{1, 2, 3, \ldots, 10\}$

4-adjacency

Not

m-adjacency (mixed adjacency)

2 pixels p and q are m-adjacent if

(i) q is in $N_4(p)$ or

(ii) q is in $N_D(p)$ and the set $N_4(p) \cap N_4(q)$ has no pixels whose values are from V.

Its a modification of 8-adjacency introduced to eliminate ambiguities resulting from 8-adjacency.

Eg.

0	1	1
0	1	0
0	0	1

8 path

3 pixels at the top show multiple 8-adjacency.

$V = \{1\}$

8-adjacency

m path

0	1	1
0	1	0
0	0	1

Digital path (or curve) from pixel p with coordinates (x_0, y_0) to pixel q with coordinates (x_n, y_n) is a sequence of distinct pixels with coordinates $(x_0, y_0), (x_1, y_1), \ldots (x_n, y_n)$.

where (x_i, y_i) and (x_{i-1}, y_{i-1}) are adjacent for $1 \le i \le n$.

Here n, is the length of path.

If $(x_0, y_0) = (x_n, y_n)$, The path is closed.

We can define 4-, 8- and m- paths depending on the type of adjacency.

Connected Set : Let S represent a subset of pixels in an image. Two pixels p and q are said to be connected in S if there exists a path between them consisting entirely of pixels in S.

For any pixel p in S, set of pixels that are connected to it in S is called a connected component.

If it has one component and that component is connected, then S is called a connected set.

Let R represent a sub-set of pixels of an image. R is a <u>region</u> if R is a connected set.

2 Regions R_i and R_j are said to be adjacent if their unions form a connected set.

Non-adjacent regions are called disjoint.

<u>Eg</u>

$$\left.\begin{matrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{matrix}\right\} R_i$$

$$\left.\begin{matrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix}\right\} R_j$$

2 regions with 8-adjacency

<u>Distance Measures</u>

For pixels p, q and s with coordinates (x,y), (u,v) and (w,z) respectively, D is a distance function or metric if.

(a) $D(p,q) \geq 0$ $\quad (D(p,q) = 0$ iff $p = q)$,

(b) $D(p,q) = D(q,p)$ and

(c) $D(p,s) \leq D(p,q) + D(q,s)$

<u>Euclidean Distance</u> between p and q is defined as—

$$D_e(p,q) = \left[(x-u)^2 + (y-v)^2\right]^{1/2}$$

For this distance measure, pixels having a distance <= some value r from (x,y) are the points contained in a disc of radius r centered at (x,y)

D_4 distance: city-block distance between p and q is defined as

$$D_4(p,q) = |x-u| + |y-v|$$

D_4 distance from (x,y) <= some distance d form a diamond centered at (x,y).

eg D_4 distance ≤ 2 from (x,y) form

		2		
	2	1	2	
2	1	0	1	2
	2	1	2	
		2		

$D_4 = 1$ and 4 neighbors of (x,y).

D_8 distance: chessboard distance between p and q is defined as

$$D_8(p,q) = \max(|x-u|, |y-v|)$$

A square centered at (x,y)

D_8 distance ≤ 2 forms -

2	2	2	2	2
2	1	1	1	2
2	1	0	1	2
2	1	1	1	2
2	2	2	2	2

$D_8 = 1$ are the 8-neighbors of pixel at (x,y)

- D_4 and D_8 ~~path~~ distances are independent of any paths that might exist between these points. ∵ it involves only coordinates
- D_m between 2 points is defined as the shortest m-path b/w the points

p_3 p_4 $V = \{1\}$ p, p_2, p_4 are 1. p_1 & p_3 may be 0 or 1.

p_1 p_2

p

If p_1 and p_3 are 0.

then length of shortest m-path = 2 b/w p and p_4

$p\,p_2\,p_4$.

If $p_1 = 1$

shortest m-path = 3 $p\,p_1\,p_2\,p_4$

If $p_3 = 1$

shortest m-path = 3 $p\,p_2\,p_3\,p_4$

If $p_1, p_3 = 1$

shortest m-path = 4 $p\,p_1\,p_2\,p_3\,p_4$.

INTRODUCTION TO BASIC MATHEMATICAL TOOLS USED IN DIP.

Elementwise V/s Matrix Operations.

- 2x2 images

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \quad \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

element wise product ($\odot$ or $\otimes$)

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \odot \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} & a_{12}b_{12} \\ a_{21}b_{21} & a_{22}b_{22} \end{bmatrix}$$

matrix product

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{bmatrix}$$

Linear Vs Non-linear operations

Consider a general operator H that produces an output image $g(x,y)$ from the given input image $f(x,y)$:

$$H[f(x,y)] = g(x,y)$$

given 2 arbitrary constants a and b, 2 arbitrary images $f_1(x,y)$ and $f_2(x,y)$,

H is said to be a linear operator if

$$\mathcal{H}[af_1(x,y)+bf_2(x,y)] = a\mathcal{H}[f_1(x,y)] + b\mathcal{H}[f_2(x,y)]$$
$$= ag_1(x,y)+bg_2(x,y) \quad \leftarrow \text{linearity}$$

Eg1 $\sum[af_1(x,y)+bf_2(x,y)] = \sum af_1(x,y) + \sum bf_2(x,y)$ {summation is distributive

$= a\sum f_1(x,y) + b\sum f_2(x,y)$

$= ag_1(x,y) + bg_2(x,y)$

$\therefore \sum$ is a linear operator.

Eg2 $f_1 = \begin{bmatrix} 0 & 2 \\ 2 & 3 \end{bmatrix}$ $f_2 \begin{bmatrix} 6 & 5 \\ 4 & 7 \end{bmatrix}$, $a=1, b=-1$

LHS $\max\left\{1\begin{bmatrix} 0 & 2 \\ 2 & 3 \end{bmatrix} + (-1)\begin{bmatrix} 6 & 5 \\ 4 & 7 \end{bmatrix}\right\} = \max\left\{\begin{bmatrix} -6 & -3 \\ -2 & -4 \end{bmatrix}\right\} = -2$

RHS

$(1)\max\begin{bmatrix} 0 & 2 \\ 2 & 3 \end{bmatrix} + (-1)\max\begin{bmatrix} 6 & 5 \\ 4 & 7 \end{bmatrix}$

$= 3 + (-1)7 = -4$.

$\therefore$ max is non-linear operator.

Arithmetic Operations

$s(x,y) = f(x,y) + g(x,y)$

$d(x,y) = f(x,y) - g(x,y)$

$p(x,y) = f(x,y) \times g(x,y)$

$v(x,y) = f(x,y) \div g(x,y)$

These are element-wise operator

Set & Logical operators

Set operations

A set is a collection of distinct objects

If a is an element of A, we write | if not

$a \in A$ | $a \notin A$

$A \subseteq B$

$C = A \cup B$

$D = A \cap B$

disjoint sets $A \cap B = \emptyset$

$A^c = \{ w \mid w \notin A \}$

$A - B = \{ w \mid w \in A, w \notin B \} = A \cap B^c$

Example images

Relation: a relation on a set A is a collection of ordered pairs of elements from A.

(a) Reflexive: for any $a \in S$ (universal set); aRa.

(b) transitive: aRb, bRc implies aRc.

(c) antisymmetric: aRb and bRa implies $a = b$.

Logical operations

Deal with TRUE and FALSE variables and expressions.

AND, OR, NOT.

Spatial operations

These are performed directly on pixels.

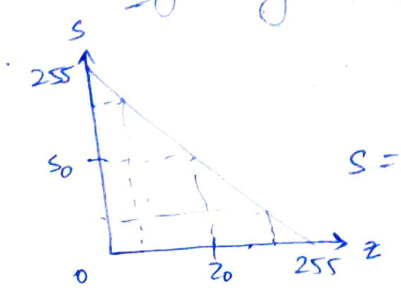
① single-pixel operations ② Neighborhood operations

③ Geometric spatial transformations.

(i) Single pixel operations

$s = T(z)$ where z is the intensity of a pixel in original image, s is the mapped intensity of corresponding pixel in processed image.

Eg negative of a binary image



$s = T(z)$

Neighborhood operations

For a pixel centered at coordinate (x,y), its value is determined by operation on its neighborhood

Eg
$$g(x,y) = \frac{1}{mn} \sum_{(r,c) \in S_{xy}} f(r,c)$$

where r and c are the row & column coordinates of the pixels whose coordinates are in set S_{xy}. Image g is created by varying (x,y) so that the centre of neighborhood moves from pixel to pixel in image f.

$m \times n$ is the rectangular neighborhood centered on (x,y).

Eg Averaging

Geometric Transformations

Modifies the spatial arrangement of pixels in an image

This includes

1. spatial transformation of coordinates.
2. Intensity interpolation that assigns intensity values to the spatially transformed pixels.

This transformation can scale, rotate, translate and sheer an image.

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

Mapping → Forward mapping
Mapping → Inverse mapping

$(x', y') = A(x,y)$

Forward mapping → scanning the pixels in input image and at each location (x,y), computing spatial coordinates location (x', y') of the corresponding pixel in the output image.

- 2 or more pixels may be mapped onto same location in O/P image
- Some output locations may not be assigned any values

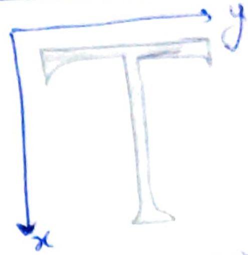
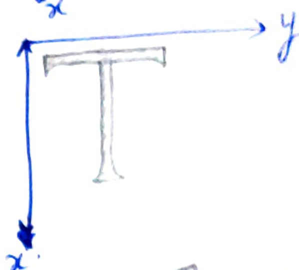
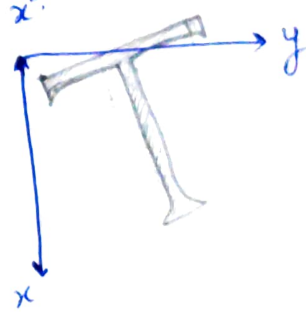
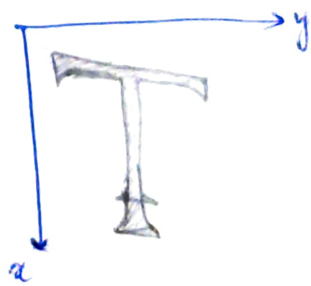
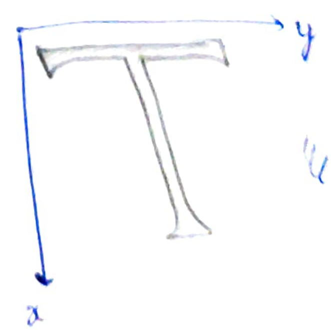
Inverse mapping

scans the O/P pixel location pixel-by-pixel at each location (x',y') and computes corresponding location in the input image using

$$(x,y) = A^{-1}(x,y).$$

It then interpolates to determine the intensity

MATLAB uses this approach.

TRANSFORMATION NAME	Affine Matrix	Coordinate equations	Example
• Identity	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x$ $y' = y$	
• Scaling	$\begin{bmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = c_x x$ $y' = c_y y$	
• Rotation (about the origin)	$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x\cos\theta - y\sin\theta$ $y' = x\sin\theta + y\cos\theta$	
• 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)	$\begin{bmatrix} 1 & 0 & 0 \\ s_h & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x$ $y' = s_h x + y$	

Q1 Consider 2 image subsets S1 and S2. For $V = \{1\}$, determine whether these 2 are (a) 4-adjacent (b) 8-adjacent (c) m-adjacent connected.

$$\begin{array}{cccc|cccc} \multicolumn{4}{c|}{S_1} & \multicolumn{4}{c}{S_2} \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1_q & 1 & 0 & 0 \\ 0 & 1 & 1 & 1_p & 0 & 0 & 0 & 0 \end{array}$$

Solⁿ

(a) S1 and S2 are not 4-connected ∵ q is not in $N_4(p)$.

(b) S1 and S2 are 8-connected ∵ q is in $N_8(p)$.

(c) S1 & S2 are m-connected because

(i) q is in $N_D(p)$ and $N_4(p) \cap N_4(q)$ is empty

Q2. Consider the image segment shown: Let $V = \{0, 1\}$. Compute the shortest 4-, 8- and m-path between p & q.

$$\begin{array}{cccc} 3 & 1 & 2 & 1_q \\ 2 & 2 & 0 & 2 \\ 1 & 2 & 1 & 1 \\ 1_p & 0 & 1 & 2 \end{array}$$

4-path: Not possible

$$\begin{array}{cccc} 3 & 1 & 2 & 1_q \\ 2 & 2 & 0 & 2 \\ 1 & 2 & 1 & 1 \\ 1_p & 0 & 1 & 2 \end{array}$$

8-path

$$\begin{array}{cccc} 3 & 1 & 2 & 1 \\ 2 & 2 & 0 & 2 \\ 1 & 2 & 1 & 1 \\ 1 & 0 & 1 & 2 \end{array}$$

shortest length = 4

m-path

$$\begin{array}{cccc} 3 & 1 & 2 & 1 \\ 2 & 2 & 0 & 2 \\ 1 & 2 & 1 & 1 \\ 1 & 0 & 1 & 2 \end{array}$$

shortest path = 5