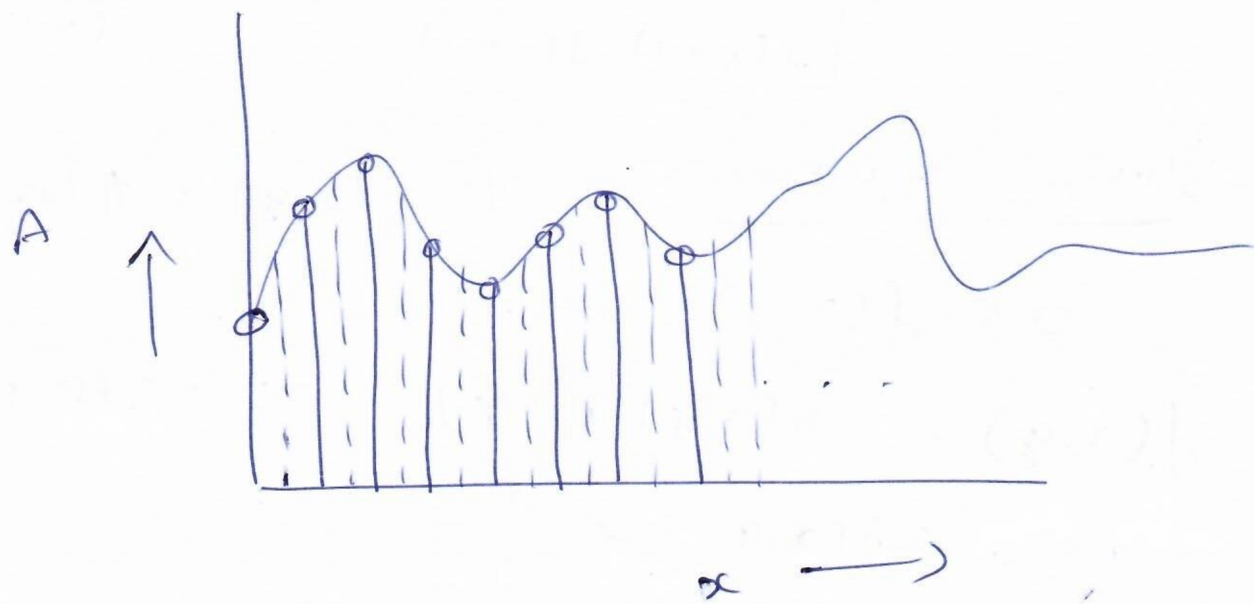
Image Digitization & Sampling 28/7/18

Let us consider single horizontal scan line



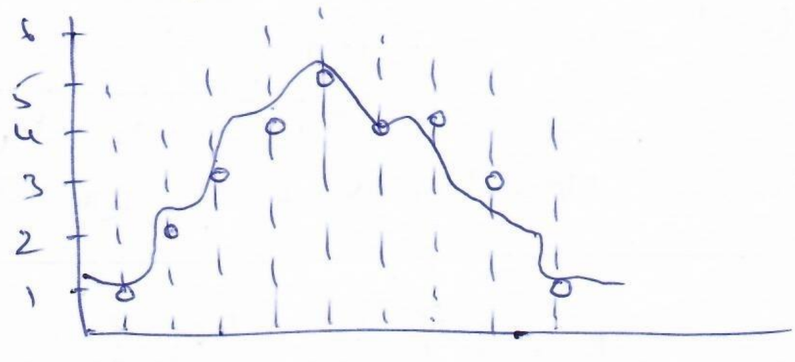
Sampling freq = 2 fmax

= 2 × max spacial freq.

Intensity levels ≈ 256

Low resolution images — computation efficiency

Spacial sampling & Intensity quantization



Max quantization err = ±0.5V

Mathematical rep of Images $f(x,y) \Rightarrow$ $x = 0, 1, \ldots N-1$; $y = 0, 1, \ldots M-1$

$$f(x,y) = \begin{bmatrix} f(0,0) & f(1,0) & \cdots & f(N-1,0) \\ f(0,1) & f(1,1) & \cdots & f(N-1,1) \\ f(0,2) & f(1,2) & \cdots & f(N-1,2) \\ \vdots & & & \\ f(0,M-1) & f(1,M-1) & \cdots & f(N-1,M-1) \end{bmatrix}$$

$N \times M$

Storage requirements: $\# f(x,y) \times \#$ bits/element

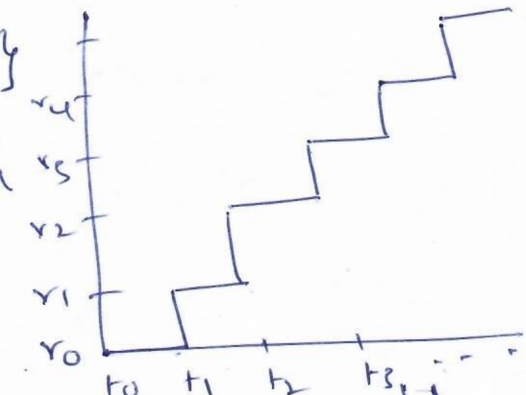
$$0 < f(x,y) < \infty$$

$$f(x,y) = i(x,y) \cdot r(x,y) \rightarrow 0 < f(x,y) < \infty$$

$$\left.\begin{aligned} 0 < i(x,y) < \infty \\ 0 \le r(x,y) \le 1 \end{aligned}\right\}$$

Analytical Expression on Image quantization

$$\xrightarrow[\text{continuous}]{u} \boxed{\text{quantizer}} \xrightarrow[\text{Discrete}]{\dot{u}} \{r_1, r_2 \cdots r_L\}$$



t_k: $k = 1, 2 \cdots L+1$

transition / Decision level

$\left.\begin{matrix} t_1 \\ t_2 \end{matrix}\right\} r_1$

$\left.\begin{matrix} t_2 \\ t_3 \end{matrix}\right\} r_2$

Optimum Mean Square Error quantizer

$$\mathcal{E} = E\left[(u-\dot{u})^2\right]; \quad \text{pdf of } u = p_u(u)$$

$$= \int_{t_1}^{t_{L+1}} (u-\dot{u})^2 \, p_u(u)\, du$$

$$= \sum_{i=1}^{L} \int_{t_i}^{t_{i+1}} (u - r_i)^2 \, p_u(u) \, du \qquad \text{Minimization of MSE}$$

$$\frac{\partial E}{\partial t_k} = (t_k - r_{k-1})^2 p_u(t_k) - (t_k - r_k)^2 p_u(t_k) = 0$$

$$\frac{\partial E}{\partial r_k} = 2 \int_{t_k}^{t_{k+1}} (u - r_k) \, p_u(u) \, du = 0$$

After Simplification $\quad t_k = \dfrac{r_k + r_{k-1}}{2}$

$$r_k = \frac{\int_{t_k}^{t_{k+1}} u \, p_u(u) \, du}{\int_{t_k}^{t_{k+1}} p_u(u) \, du}$$

uniform pdf $\quad p_u(u) = \dfrac{1}{t_{L+1} - t_1} \qquad t_1 \le u \le t_{L+1}$

$\qquad\qquad = 0 \qquad$ otherwise

$$r_k = \frac{t_{k+1} + t_k}{2}$$

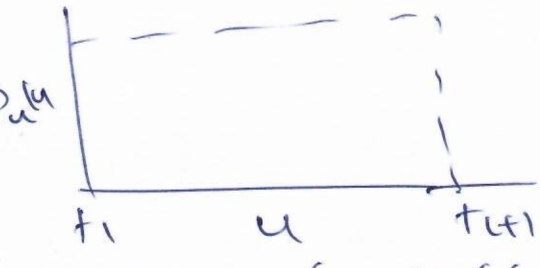
$t_k - t_{k-1} = t_{k+1} - t_k = \text{const } \Delta \simeq q$

$q = \dfrac{t_{L+1} - t_1}{L} \; ; \; t_k = t_{k-1} + q$

$$r_k = t_k + \frac{q}{2} = t_{k+1} - \frac{q}{2}$$

QE is uniformly distributed over $\left(-\frac{q}{2} \text{ to } \frac{q}{2}\right)$

$$E = \frac{1}{q}\int_{-q/2}^{q/2} u^2\,du = \frac{q^2}{12}$$



If 'A' is the range of variable u (intensity)
'B' is # bits in a quantizer

$$q = \frac{A}{2^B} \quad \text{Step size}$$

$$\sigma_u^2 = \frac{A^2}{12} \quad [\text{uniform pdf of } u]$$

$$\frac{E}{\sigma_u^2} = \frac{q^2}{12}\Big/\frac{A^2}{12} = \frac{q^2}{A^2} = \frac{A^2}{2^{2B}\cdot A^2} = \frac{1}{2^{2B}}$$

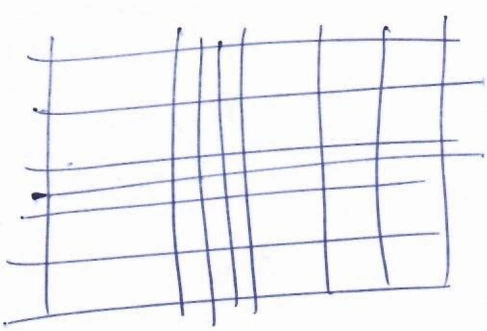
$$= 2^{-2B}$$

$$SNR = 10\log_{10}\frac{\sigma_u^2}{E} = 10\log_{10} 2^{2B} = \underline{6B \text{ dB}}$$

Uniform / Linear / Lloyd-max quantizer

Uniform spatial sampling

Non-uniform spatial sampling



Non-uniform quantization

Re-quantization

Image Interpolation & Resampling

Geometrical transformation $(x,y) \longrightarrow (x', y')$

Affine Transformation

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

Translation $\begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

Scaling $\begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix}$; $\begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

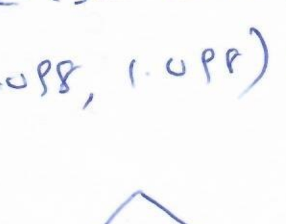
$s < 1$ — reduction

$s > 1$ — enlargement

Rotation by a $\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos a & \sin a \\ -\sin a & \cos a \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

Example

$a = 30^\circ$

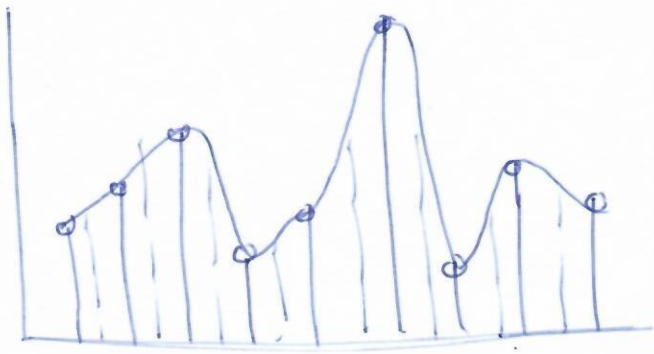


$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos 30 & \sin 30 \\ -\sin 30 & \cos 30 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$(2,2) \longrightarrow (2.732 \quad 0.732)$ $(3,2) \longrightarrow (3.598, 0.232)$

$(3,3) \longrightarrow (4.098, 1.098)$ $(2,3) \longrightarrow (3.232, 1.598)$

Approximation of continuous image



interpolation => finite number of neighbouring pixels

Basic requirement of interpolation

1. Finite support region
2. Smooth interpolation
3. Shift invariant

Finite support region and smooth interpolation] Spline function

B-Spline function

Let $\pi : \xi_0 < \xi_1 < \xi_2 < \dots < \xi_m < \xi_{m+1}$

be a partition of the interval $[\xi_0 \; \xi_{m+1}]$

$$B_m(\xi : \xi_0, \xi_1 \dots \xi_{m+1})$$

$$= (m+1) \sum_{k=0}^{m+1} \frac{(\xi - \xi_k)^m \, U(\xi - \xi_k)}{\omega(\xi_k)}$$

unit step

B_m - Spline fn of order 'm'

$\xi_0 \; \xi_1 \dots \xi_{m+1} \longrightarrow$ samples

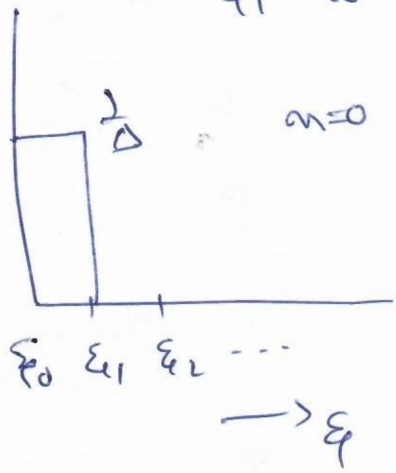
$$\omega(\xi_{ek}) = \prod_{\substack{J=0\\ J\neq k}}^{n+1} (\xi_{ek}-\xi_{eJ})$$

$$U(\xi_e - \xi_{ek}) = \begin{cases} (\xi_e-\xi_{ek})^0 & \text{for } \xi_e > \xi_{ek} \\ 0 & \text{for } \xi_e \le \xi_{ek} \end{cases}$$

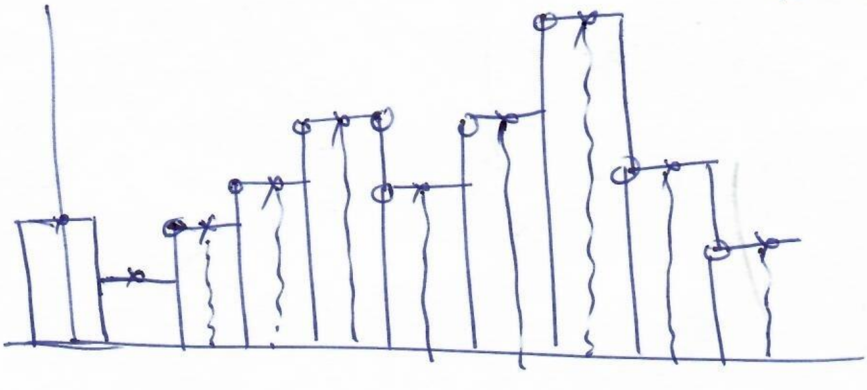
$n=0$; spline fn of order 0

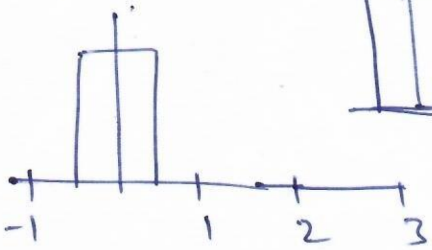
$$B_0 = \sum_{k=0}^{1} \frac{(\xi_e-\xi_{ek})\, U(\xi_e-\xi_{ek})}{\omega(\xi_{ek})}$$

$\Delta = \xi_{e1} - \xi_{e0}$



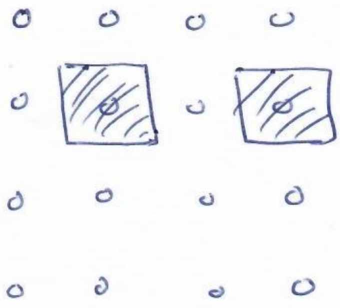
Nearest neighbour



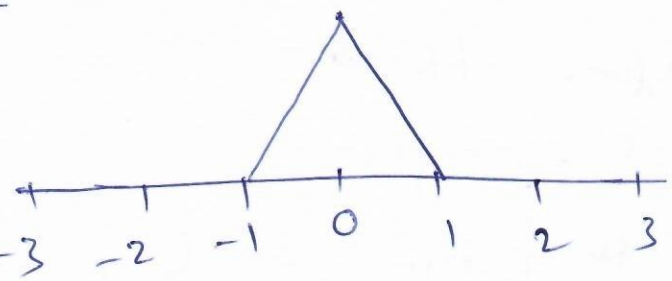


$-0.5 < \xi_e < 0.5 \quad f(\xi_e) = \xi_{e0}$

$0.5 < \xi_e < 1.5 \quad f(\xi_e) = \xi_{e1}$



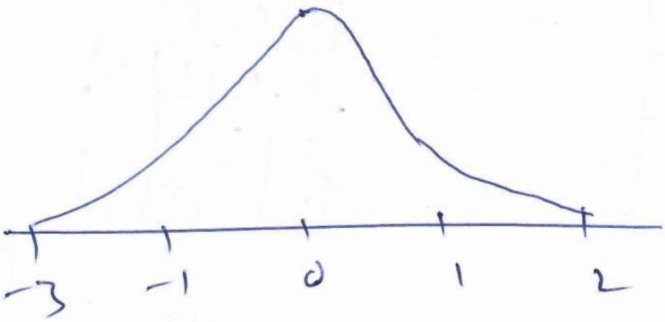
$n = 1 \Rightarrow B_1 = B_0 * B_0$ Linear interpolation

$B_1 =$ 

$B_2 = B_0 * B_0 * B_0$ quadratic interpolation

not symmetric => not used

$B_3 = B_0 * B_0 * B_0 * B_0 \Rightarrow B_1 * B_1$



cubic B-spline interpolation

Comparison of B_0, B_1 & B_3

$B_0 \Rightarrow$ Simple to implement & less computational cost

LPF on image, very good performance on passband

sharp transitions, high freq. are retained

$B_1 \Rightarrow$ linear interpolation, poor passband performance

good stopband performance

suitable for low freq. image data

Applications of interpolation functions

1. Geometric corrections of satellite images
2. compare the images of diff. sensors, diff. resolutions (interpolate & relate)

~~Registration~~

(2) Registration of images from different sensors
defense application

(3) Medical images

(4) Magnification / Minification } of images by real factors.
Enlargement / reduction

Image Magnification Technique

A	B		
C	D		

A	A	B	B
A	A	B	B
C	C	D	D
C	C	D	D

Box structure effect

$256 \times 256 \longrightarrow 512 \times 512 \dashrightarrow$ pixel replication

$256 \times 256 \longrightarrow 320 \times 320$ (interpolation)

Linear interpolation

$f(k)$ & $f(k+1)$

$0 \leq r \leq 1$

$$f(k+r) = (1-r)\,f(k) + r\,f(k+1)$$

Cubic B-spline interpolation

$-2 \quad -1 \quad 0 \quad 1 \quad 2$

$$f(x) = \frac{x^3}{2} - x^2 + \frac{4}{6} \quad - (0,1)$$

$$= -\frac{x^3}{6} + x^2 - 2x + \frac{8}{6} \quad - (1,2)$$

① Hsieh Hou & H. Andrews, " Cubic splines for image interpolation and digital filtering IEEE Trans ASSP, V.26, Dec 1978.

② Comparison of interpolation methods for image resampling, J. A. Parker, R. V. Kenyon & Donald. E. Troxel, IEEE Trans on Medical Imaging, March 1983

③ Cubic convolution interpolation for DIP, Robert G Keys, IEEE Trans ASSP, 1981

Basic Relationship between Pixels

4-Neighbour $(x, y) \Rightarrow (x-1, y)\ (x+1, y)$

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

$o \rightarrow (x, y+1)$

$(x+1, y) \leftarrow o \quad x \quad o \rightarrow (x+1, y)$

$x \rightarrow (x, y)$

$o \rightarrow (x, y-1)$

$N_4(p)$

$N_D(p)$ $\left.\begin{array}{ll}(x+1, y+1) & (x-1, y-1) \\ (x+1, y-1) & (x-1, y+1)\end{array}\right\}$ Diagonal neighbours

$$N_8(p) = N_4(p) \cup N_D(p)$$

$$\begin{array}{ccc} \square & o & \square \\ o & x & o \\ \square & o & \square \end{array}$$

Eight Neighbours

Connectivity of pixels $\quad V$ - set of intensity values

4- connectivity $\quad q \in N_4(p)\ ;\ f(p) \in V$

$f(q) \in V$

p & q have 4-connectivity

8- connectivity $\quad f(p) \in V$

$q \in N_8(p)\ :\ f(q) \in V$

q is having 8-connectivity with p

(mixed)
m- connectivity $\quad f(q) \in V \quad f(p) \in V$

(1) If $q \in N_4(p)$

or (2) $q \in N_D(p)$ & the set $N_4(p) \cap N_4(q) = \phi$ (empty)

$V = \{59, 60, 61\}$

100   60
o   o q
x   o
59 p 101

p & $q \in V$

$q = N_D(p)$

$N_4(p) \cap N_4(q) = \phi$

$\therefore$ p & q are m- connected

60   60
o   o q
59 x   o
p

X

p & $q \in V$; $q \in N_D(p)$

$N_4(p) \cap N_4(q) \neq \phi$

$\therefore$ p & q are not m-connected

Adjacent Pixel

A pixel p is adjacent to q if they are connected

4, 8, m- adjacency

Image Subset adjacency

S_1 image subset S_2 image subset

Path

$p(x, y)$ $q(s, t)$

path from p to q is a sequence of distinct pixels with coordinates $(x_0 y_0)$ $(x_1 y_1)$ ------ $(x_n y_n)$

$(x_0 y_0) = (x, y)$ & $(x_n y_n) = (s, t)$

$(x_i y_i)$ is adjacent to $(x_{i-1} y_{i-1})$

Distance Measures

D to be distance metric if it fulfils

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

(2) $D(p\ q) = D(q\ p)$

(3) $D(p\ z) \leq D(p\ q) + D(q\ z)$

Euclidean distance

p & q

(x,y) (s,t)

$$D_e(p,q) = \left[(x-s)^2 + (y-t)^2\right]^{\frac{1}{2}}$$

City-block distance (over estimation)

$$D(p\ q) = |x-s| + |y-t|$$

$$\begin{matrix} & & 2 & & \\ & 2 & 1 & 2 & \\ 2 & 1 & 0 & 1 & 2 \\ & 2 & 1 & 2 & \end{matrix}$$

Chess-board distance (under estimation)

$$D(p\ q) = \max(|x-s|, |y-t|)$$

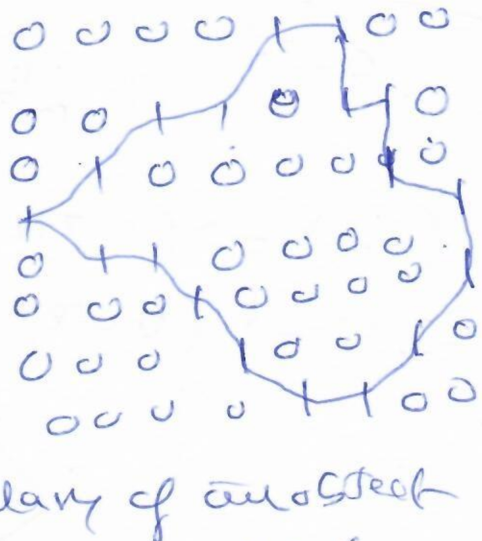
$$\begin{matrix} & & 2 & & \\ 2 & 2 & 2 & 2 & 2 \\ 2 & 1 & 1 & 1 & 2 \\ 2 & 1 & 0 & 1 & 2 \\ 2 & 1 & 1 & 1 & 2 \\ 2 & 2 & 2 & 2 & 2 \end{matrix}$$

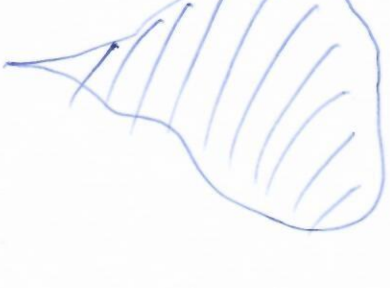
Easy to compute & efficient

integer values

Eg $\quad V = \{1\}$

8-connectivity





boundary of an object

edge of an object

Tracing the object