CHAPTER 10

IMAGE SEGMENTATION

What you will learn

- Practical edges
- Edge detection
- Edge linking
- Image segmentation based on the combination of thresholding and spatial filtering
- Region-based segmentation
- Image segmentation using motion cues

Background

- Segmentation ⇨ subdivide an image into constituent regions or objects
- Factors used by segmentation ⇨ *discontinuity* and *similarity* of gray levels
- *Discontinuity* ⇨ points, lines, edges …
- *Similarity* ⇨ similar regions
- Thresholding ⇨ fundamental approach

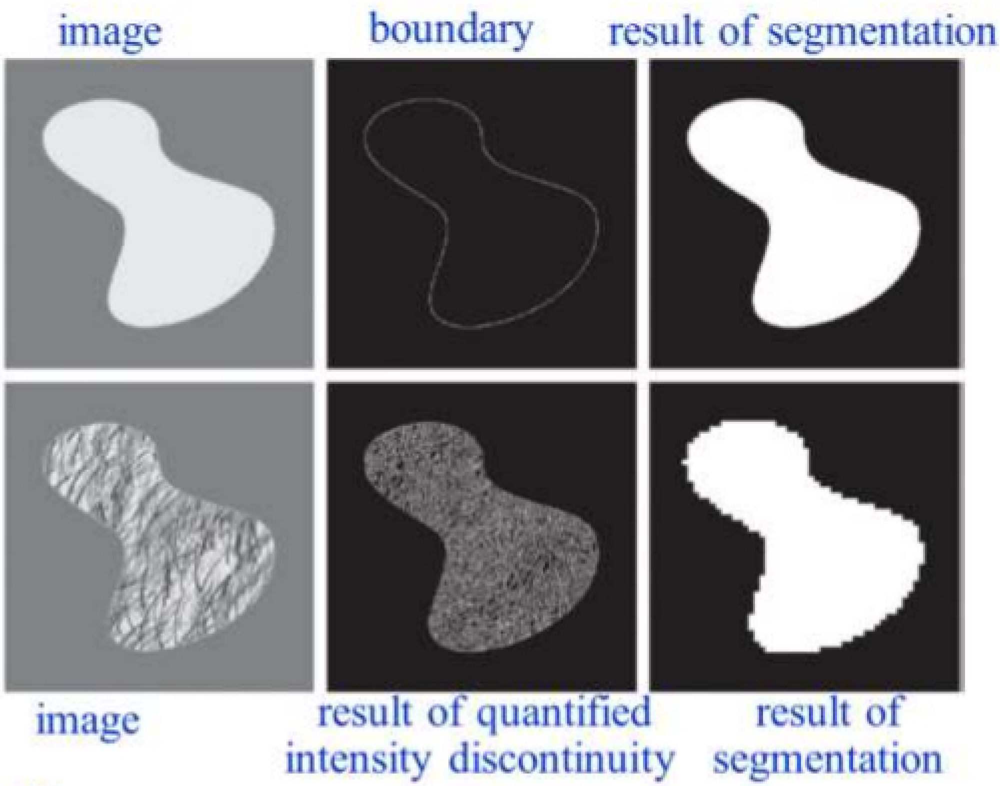


Figure 10.1

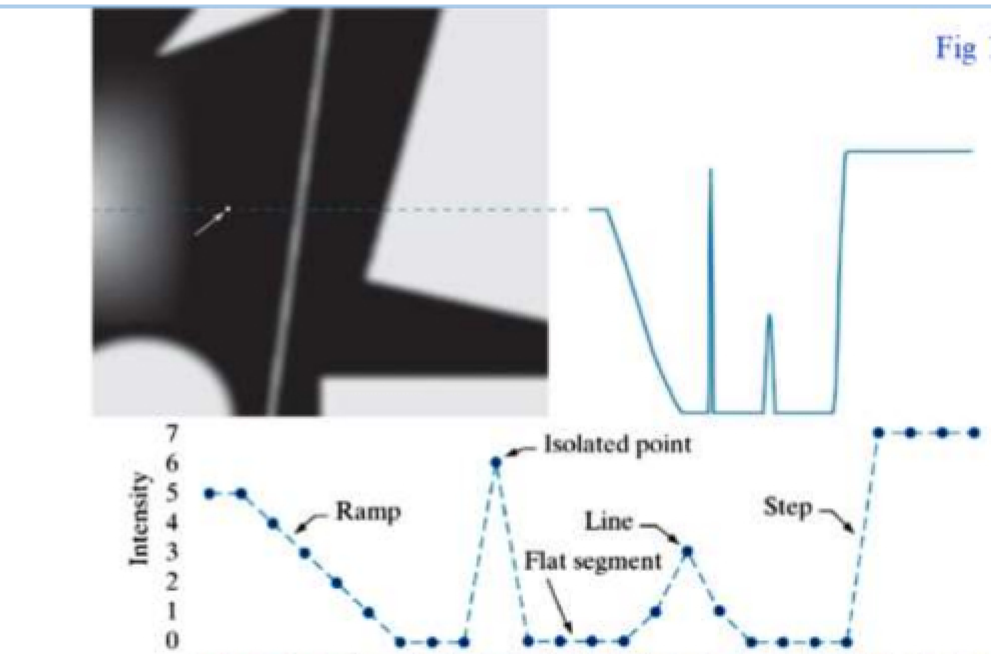
Outlines

- Detection of Discontinuities 10.2
- Edge Linking and Boundary Detection 10.2.7
- Thresholding 10.3
- Region-based Segmentation 10.4
- Segmentation based on Clustering 10.5
- Use of Motion in Segmentation 10.8

Detection of Discontinuities

- Point detection
- Line detection
- Edge detection

Fig 10.2



Intensity values	5	5	4	3	2	1	0	0	0	6	0	0	0	0	1	3	1	0	0	0	0	7	7	7	7	•	•

First derivative -1 -1 -1 -1 -1 0 0 6 -6 0 0 0 1 2 -2 -1 0 0 0 7 0 0 0

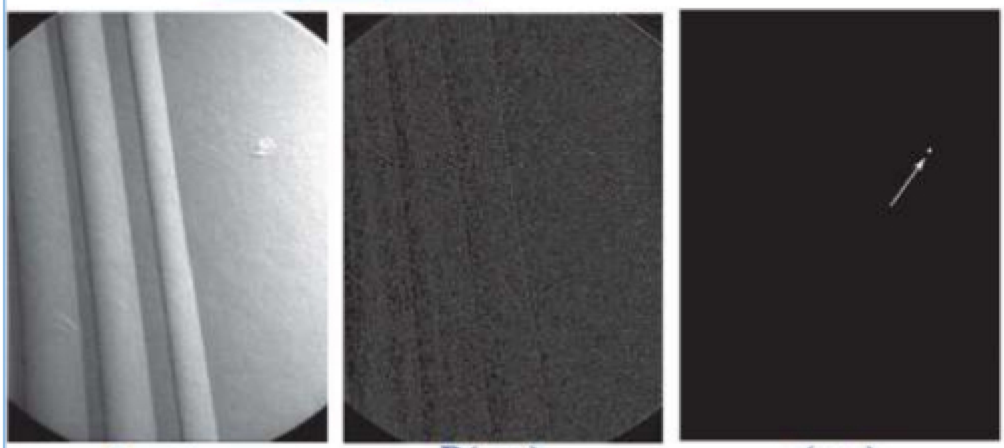
Second derivative -1 0 0 0 0 1 0 6 -12 6 0 0 1 1 -4 1 1 0 0 7 -7 0 0

$$R = z_1+z_2+z_3+z_4-8z_5+z_6+z_7+z_8+z_9$$

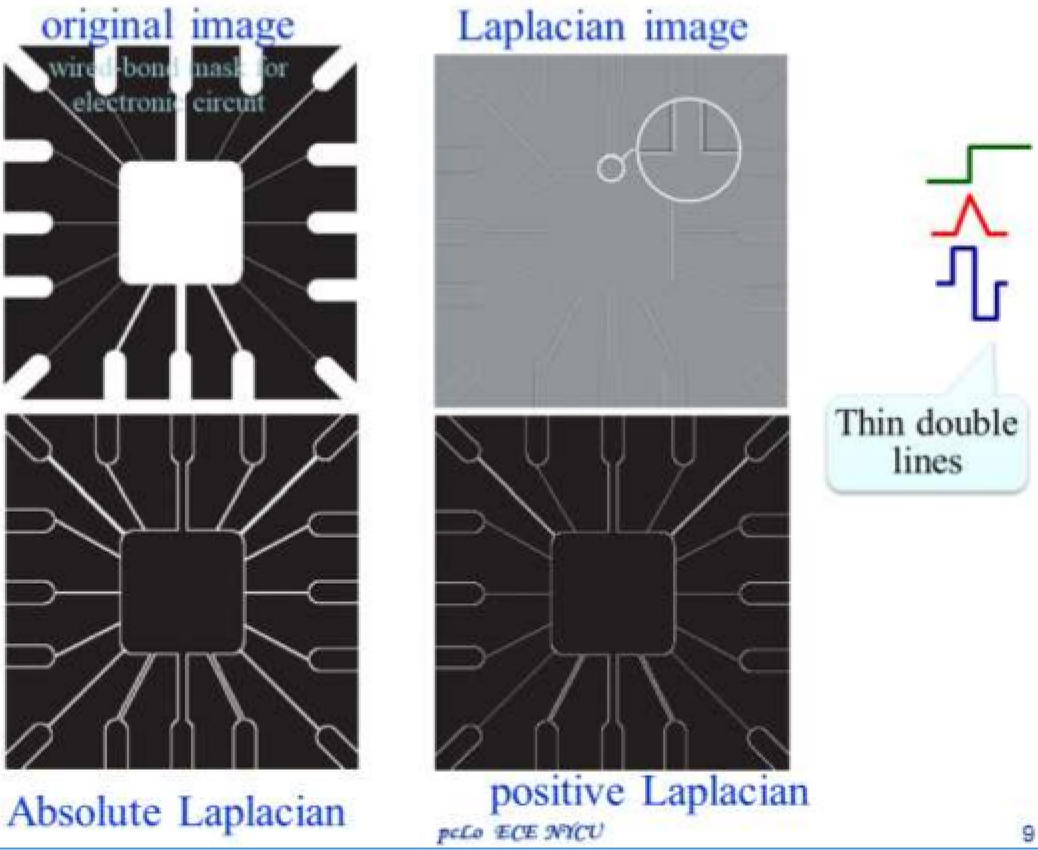
1	1	1
1	–8	1
1	1	1

$$g(x,y) = \begin{cases} 1, |R(x,y)| \geq T \\ 0, otherwise \end{cases}$$

Here, T=0.9R_{max}



$f(x,y)$ $R(x,y)$ $g(x,y)$



R_1 — Horizontal

−1	−1	−1
2	2	2
−1	−1	−1

R_2 — 45°

2	−1	−1
−1	2	−1
−1	−1	2

R_3 — Vertical

−1	2	−1
−1	2	−1
−1	2	−1

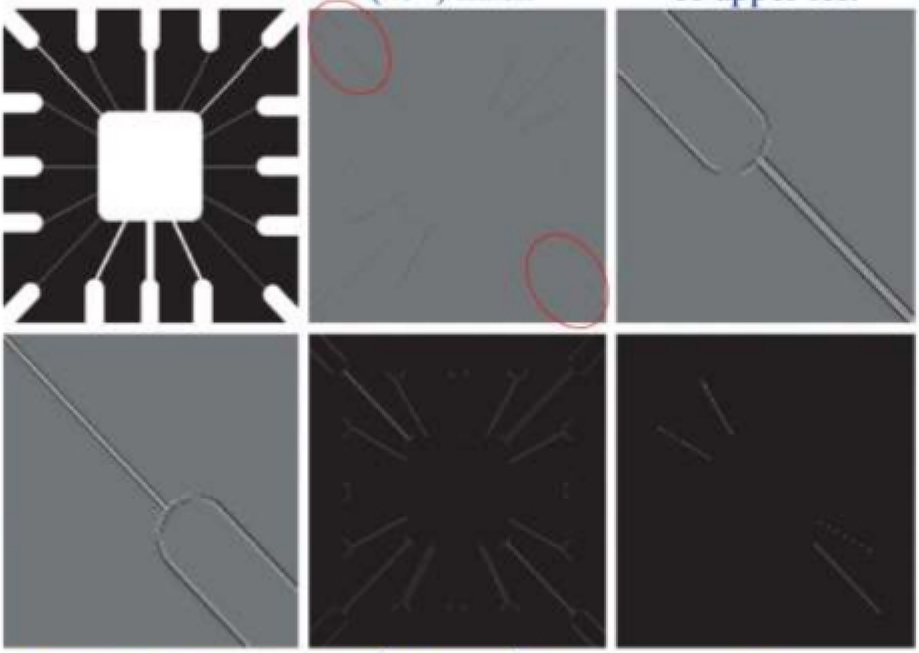
R_4 — −45°

−1	−1	2
−1	2	−1
2	−1	−1

486×486 binary image

Process with R_2 (45°) mask

Zoomed view of upper left



Zoomed view of lower right

g: Set negative values → 0

$g \geq T$ (254)

Fig 10.7

Edge model

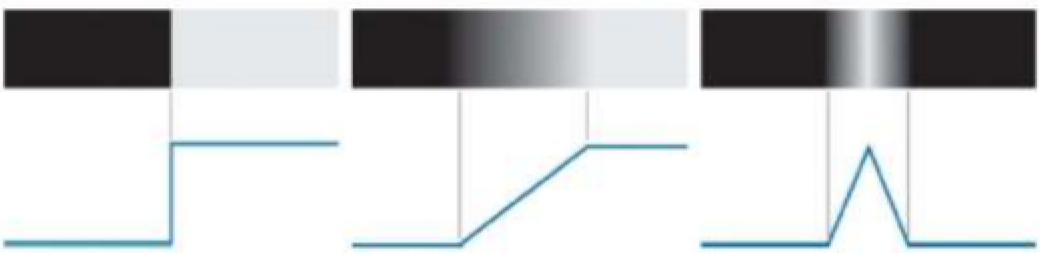
Fig. 10.8

Practical edges ⇨ edges often blurred due to image acquisition imperfections; a "ramplike" profile

Ideal-edge model

Practical-edge model

Roof-edge model



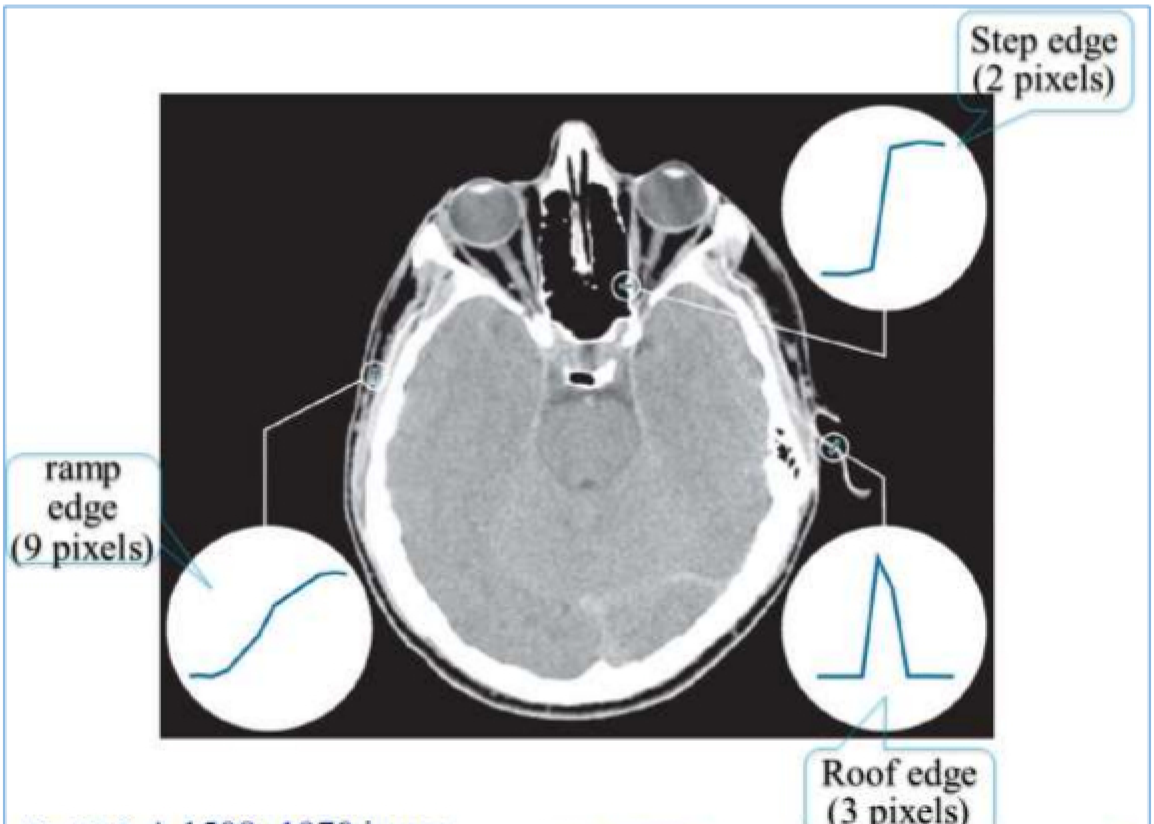
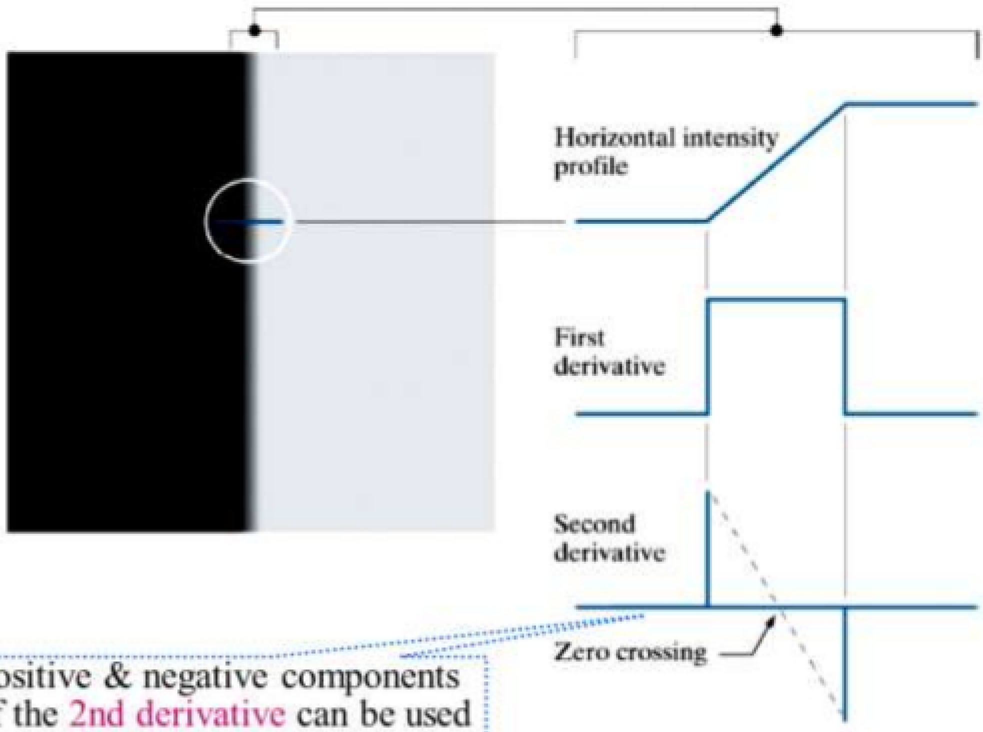
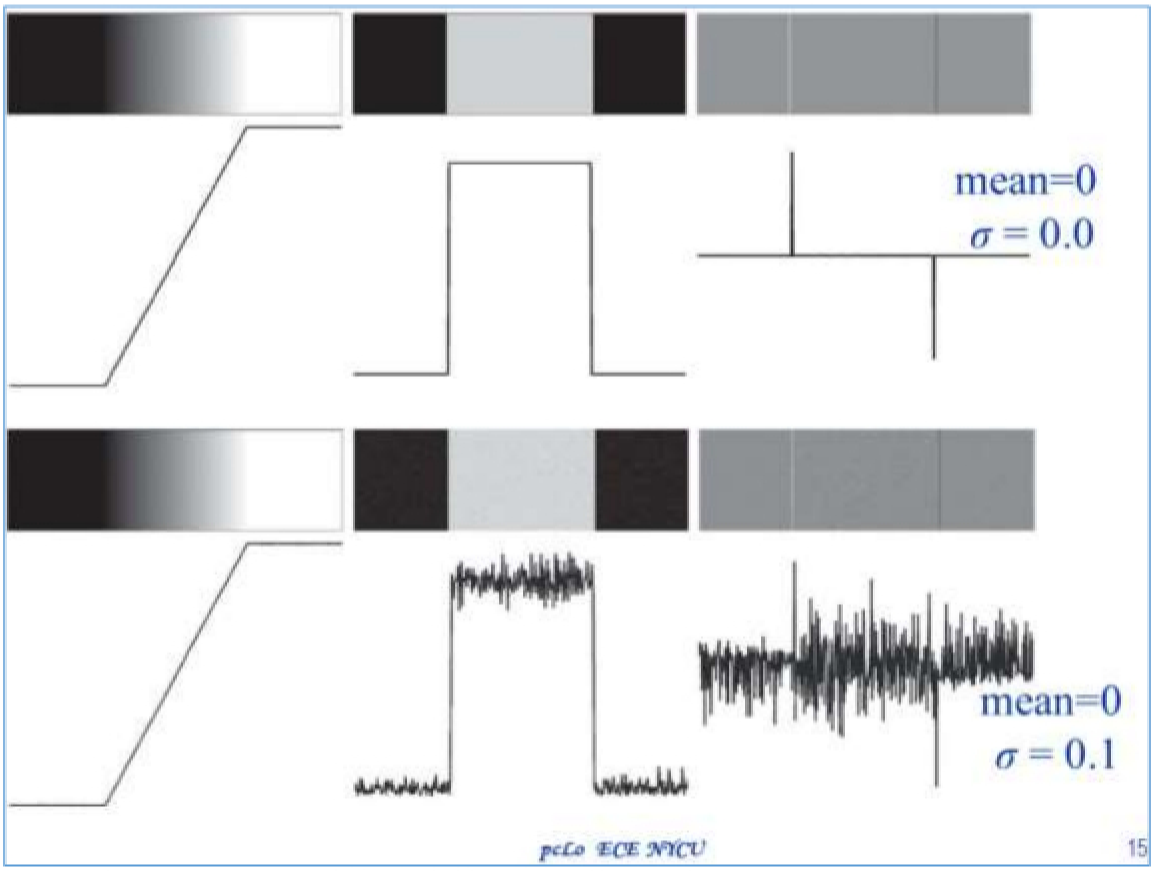
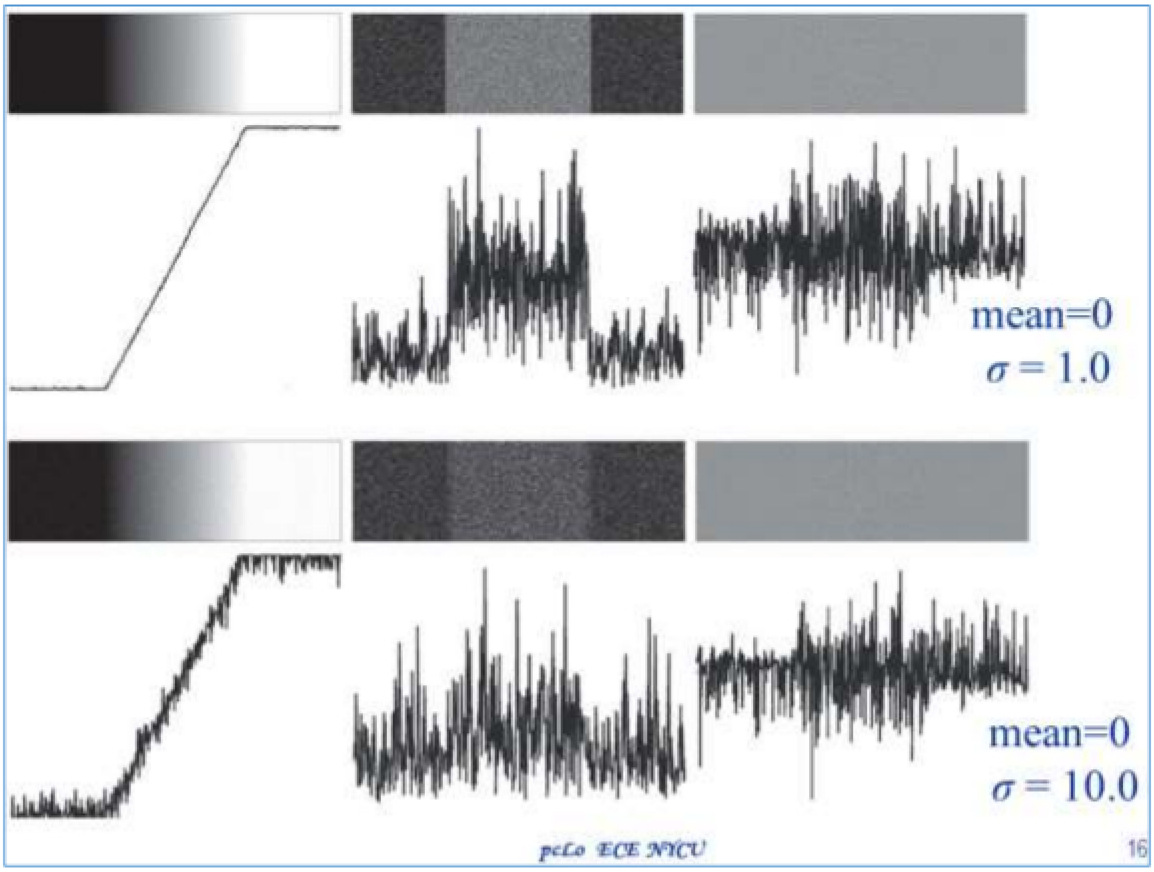


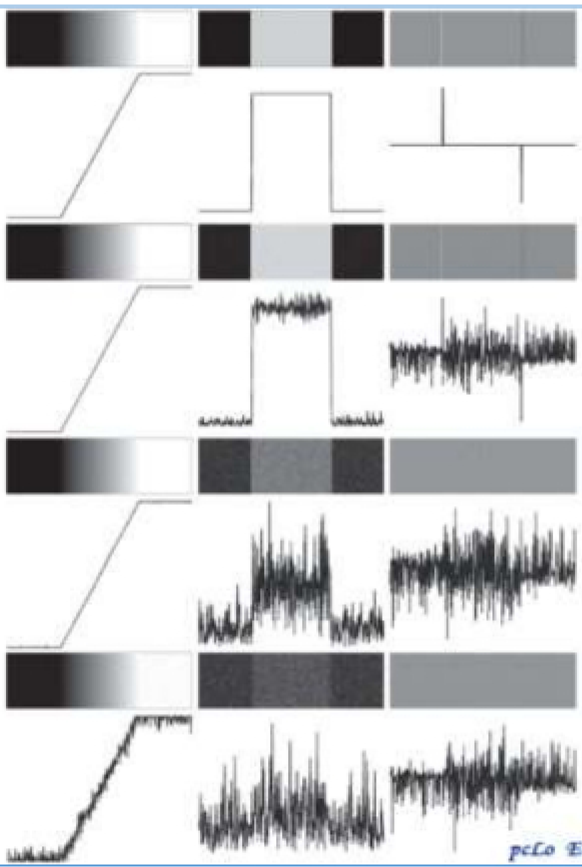
Fig 10.9 A 1508×1970 image



Positive & negative components of the 2nd derivative can be used by the edge detection


mean=0
σ = 0.0
mean=0
σ = 0.1
pcLo ECE NYCU
15


mean=0
σ = 1.0
mean=0
σ = 10.0
pcLo ECE NYCU
16



mean=0
$\sigma = 0.0$

$\sigma = 0.1$

$\sigma = 1.0$

$\sigma = 10.0$

Gradient of $f(x,y)$ at (x,y) :

$$\nabla \mathbf{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} \qquad G_x = \frac{\partial f}{\partial x} \qquad G_y = \frac{\partial f}{\partial y}$$

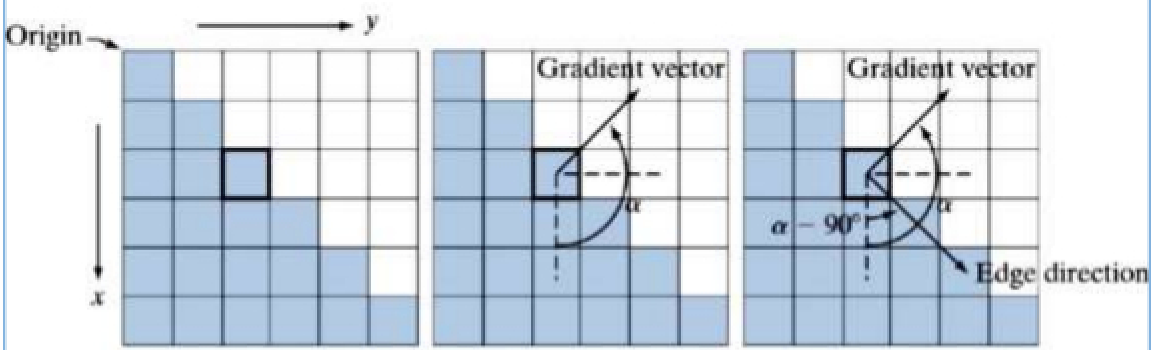
$$\nabla \mathbf{f} = \begin{bmatrix} g_x(x,y) \\ g_y(x,y) \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f(x,y)}{\partial x} \\ \dfrac{\partial f(x,y)}{\partial y} \end{bmatrix} = \begin{bmatrix} G_x \\ G_y \end{bmatrix}$$

∇f : Magnitude of the gradient vector $\nabla \mathbf{f}$

$$M(x,y) = \nabla f(x,y) = \sqrt{G_x^2 + G_y^2}$$

Alternatively, $\nabla f = |G_x| + |G_y|$

Angle (direction) of $\nabla \mathbf{f}$ $\alpha(x,y) = \tan^{-1}\left[\frac{G_y}{G_x}\right]$



edge strength and direction at a point

an edge map $$\boxed{\varepsilon(x,y)=\begin{cases}1, \nabla f > T\\ 0, \text{otherwise}\end{cases}}$$

Sobel operators:

G_x

−1	−2	−1
0	0	0
1	2	1

G_y

−1	0	1
−2	0	2
−1	0	1

G_{-45}

0	1	2
−1	0	1
−2	−1	0

G_{45}

−2	−1	0
−1	0	1
0	1	2

Prewitt operators:

G_x

−1	−1	−1
0	0	0
1	1	1

G_y

−1	0	1
−1	0	1
−1	0	1

G_{-45}

0	1	1
−1	0	1
−1	−1	0

G_{45}

−1	−1	0
−1	0	1
0	1	1

Compass operators

G_2 northwest

1	1	0
1	0	−1
0	−1	−1

G_1 north

1	1	1
0	0	0
−1	−1	−1

G_8 northeast

0	1	1
−1	0	1
−1	−1	0

G_3 west

1	0	−1
1	0	−1
1	0	−1

G_7 east

−1	0	1
−1	0	1
−1	0	1

G_4 southwest

0	−1	−1
1	0	−1
1	1	0

G_5 south

−1	−1	−1
0	0	0
1	1	1

G_6 southeast

−1	−1	0
−1	0	1
0	1	1

Fig10.15 Kirsch compass kernels (arrow: edge direction)

−3	−3	5
−3	0	5
−3	−3	5
N ↑		

−3	5	5
−3	0	5
−3	−3	−3
NW ↖		

5	5	5
−3	0	−3
−3	−3	−3
W ←		

5	5	−3
5	0	−3
−3	−3	−3
SW ↙		

5	−3	−3
5	0	−3
5	−3	−3
S ↓		

−3	−3	−3
5	0	−3
5	5	−3
SE ↘		

−3	−3	−3
−3	0	−3
5	5	5
E →		

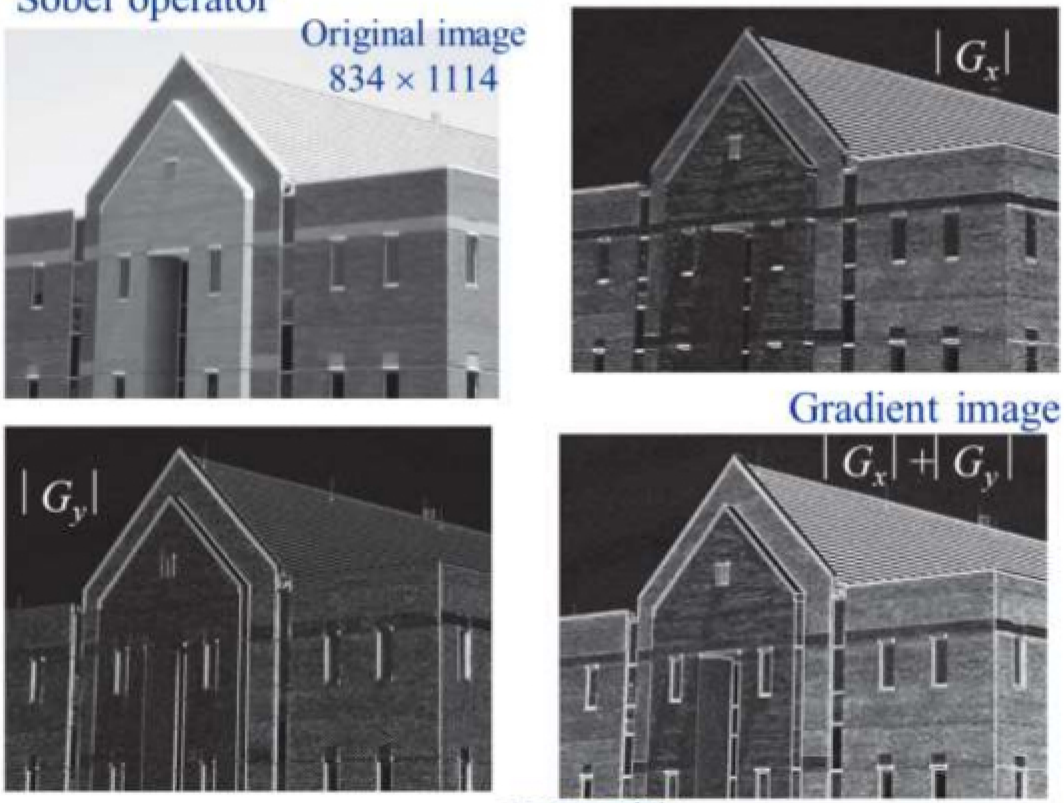
−3	−3	−3
−3	0	5
−3	5	5
NE ↗		

Based on Kirsch's approach

$$\nabla f = \max\{\,|G_k|\,\}_{k=1\sim 8}$$

$$\varepsilon(x,y) = \begin{cases} 1, \nabla f > T \\ 0, \text{otherwise} \end{cases}$$

Sobel operator



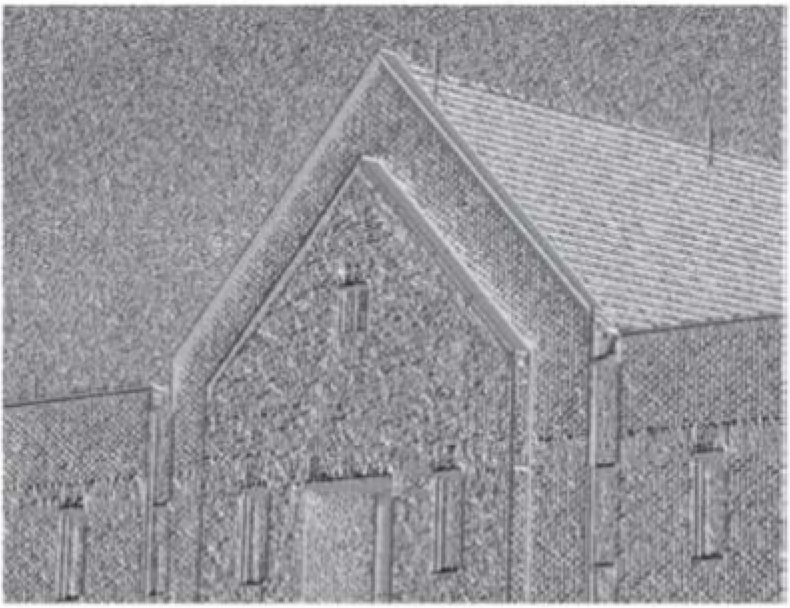


Figure 10.17 Gradient angle image.

$$\alpha(x,y) = \tan^{-1}\left[\frac{G_y(x,y)}{G_x(x,y)}\right]$$

Image smoothed by a 5×5 averaging filter

Gradient image: $|G_x| + |G_y|$

$|G_x| + |G_y|$ for original image

$|G_x| + |G_y|$ for smoothed image

NW

−3	5	5
−3	0	5
−3	−3	−3

SW

5	5	−3
5	0	−3
−3	−3	−3

$|G_x| + |G_y|$ original

$|G_x| + |G_y|$ smoothed

33

Marr-Hildreth edge detector

Marr-Hildreth algorithm: A differential operator

- Compute a digital approximation of the 1st or 2nd derivative at every point
- Tuned to act at any desired scale
- deal with the noise problem

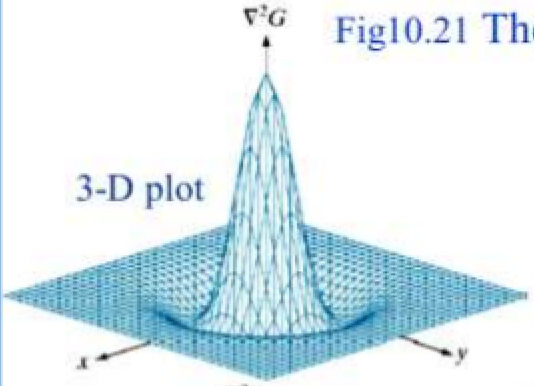
$$G(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$$

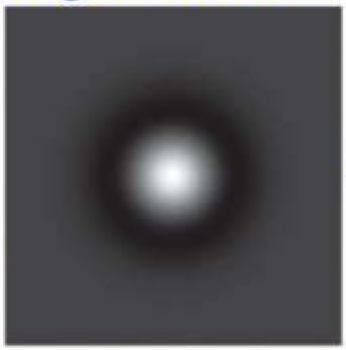
$$\nabla^2 G(x,y) = \left[\frac{x^2+y^2-2\sigma^2}{\sigma^4}\right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$

LoG: Laplacian of a Gaussian
Also called *Mexican hat* operator

Fig10.21 The negative of LoG.

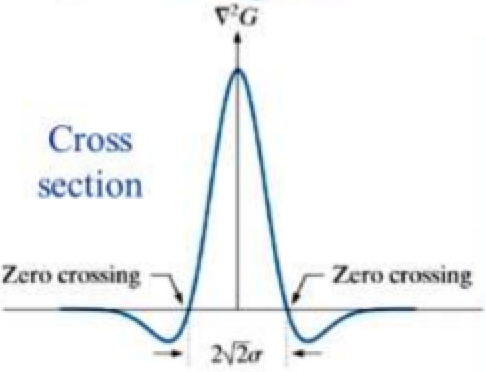
3-D plot





2D image form
−: black
+: whitish
0: gray

Cross section



0	0	−1	0	0
0	−1	−2	−1	0
−1	−2	16	−2	−1
0	−1	−2	−1	0
0	0	−1	0	0

5×5 mask approximation

$$\nabla^2 G(x,y) = \left[\frac{x^2+y^2-2\sigma^2}{\sigma^4}\right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Marr-Hildreth edge-detection algorithm:

$f(x,y)$ ↓ **$n \times n$ Gaussian LPF** ↓ Laplacian operation ↓ zero crossing ↓

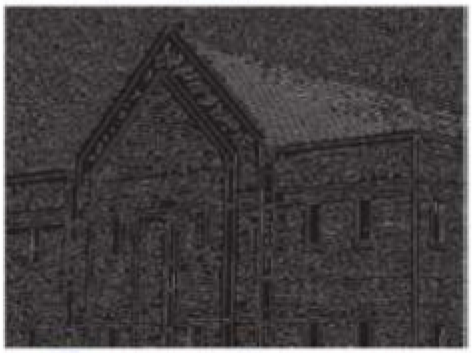
$$G(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$$

1	1	1
1	−8	1
1	1	1

Zero crossing: 3×3 mask, at least 2 opposing pixels of center pixel p
(1) opposite signs, and
(2) absolute difference ≥ threshold

Original image 824×1114

Result of LoG (σ=4, n=25)



Zero crossings
threshold = 0

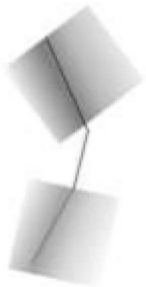
Zero crossings
threshold= 4% of max LoG

Canny edge detector

Superior performance than the others

Three objectives:

1. Low error rate
2. Well localized edge points
3. Single edge point response



Canny edge detector

$f(x,y)$

↓

Gaussian LPF + Gradient

↓ $f_s(x,y)$

$M_s(x,y)$ and $\alpha(x,y)$

↓

Nonmaxima suppression

↓

Hysteresis thresholding

↓

Edge map

Gradient ***magnitude*** $M_s(x,y)$: edge strength

Gradient ***direction*** $\alpha(x,y)$: edge direction at (x,y)

$f(x,y)$: input image $G(x,y)$: Gaussian function

$$G(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$$

$$f_s(x,y) = G(x,y) * f(x,y) \text{ Linear convolution}$$

$$\begin{cases} G_x = \dfrac{\partial f_s(x,y)}{\partial x} & M_s(x,y) = \|\nabla f_s(x,y)\| = \sqrt{G_x^2(x,y) + G_y^2(x,y)} \\ G_y = \dfrac{\partial f_s(x,y)}{\partial y} & \alpha(x,y) = \tan^{-1}\left[\dfrac{G_y(x,y)}{G_x(x,y)}\right] \end{cases}$$

Nonmaxima suppression

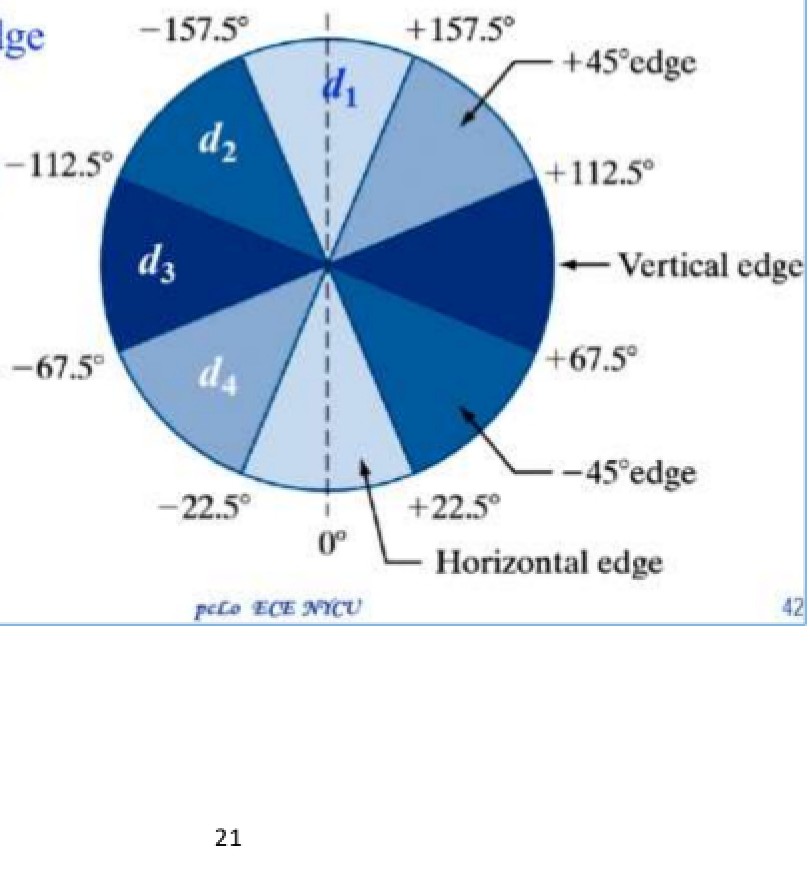
Angle ranges of the edge normal (gradient vector) for four edge directions in 3×3 region

d_1: horizontal edge

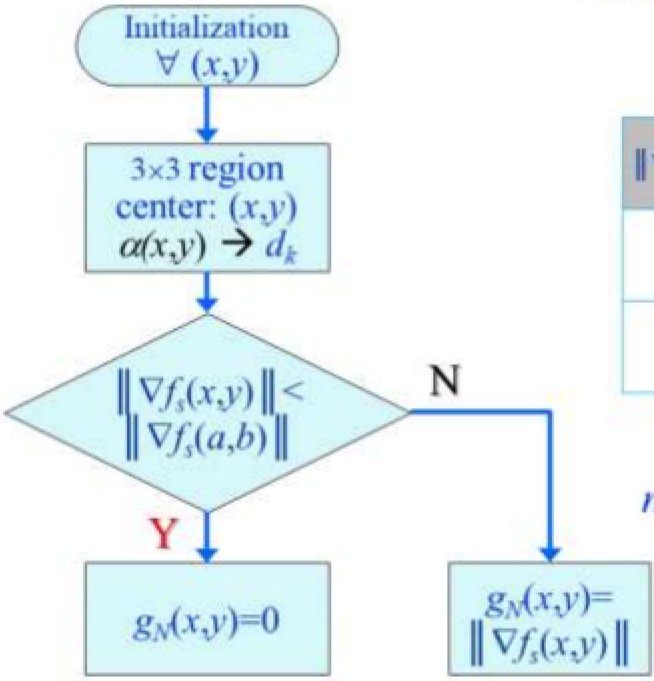
d_2: -45° edge

d_3: vertical edge

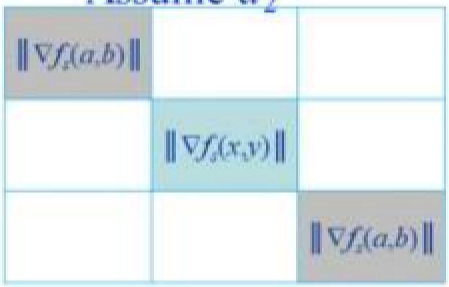
d_4: +45° edge



Nonmaxima Suppression

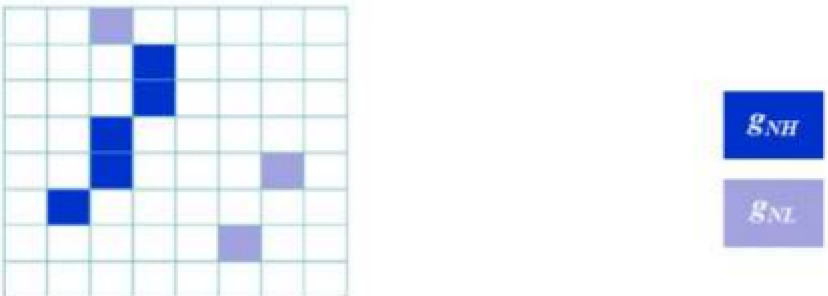


Assume d_2



nonmaxima suppressed
image $g_N(x,y)$

Canny's algorithm uses ***hysteresis thresholding***
for *edge trimming* and *edge linking*



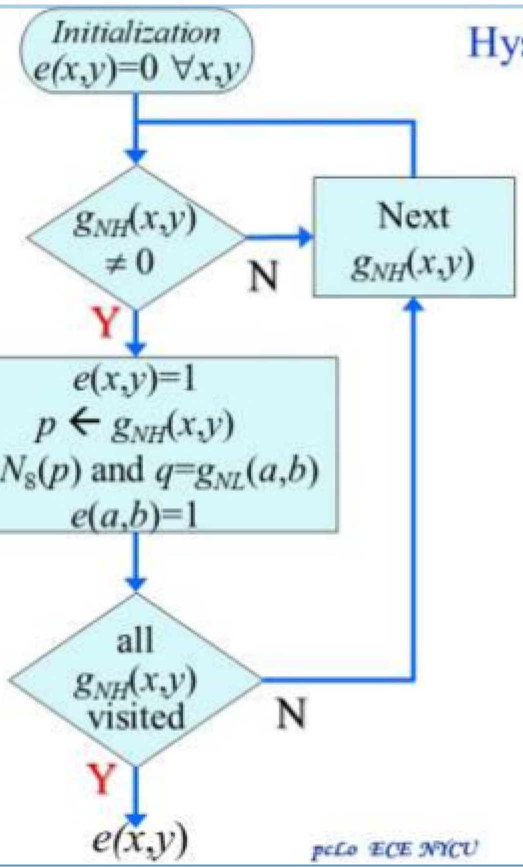
T_L (low threshold) and T_H (high threshold)

$T_H : T_L \approx$ 2:1 to 3:1

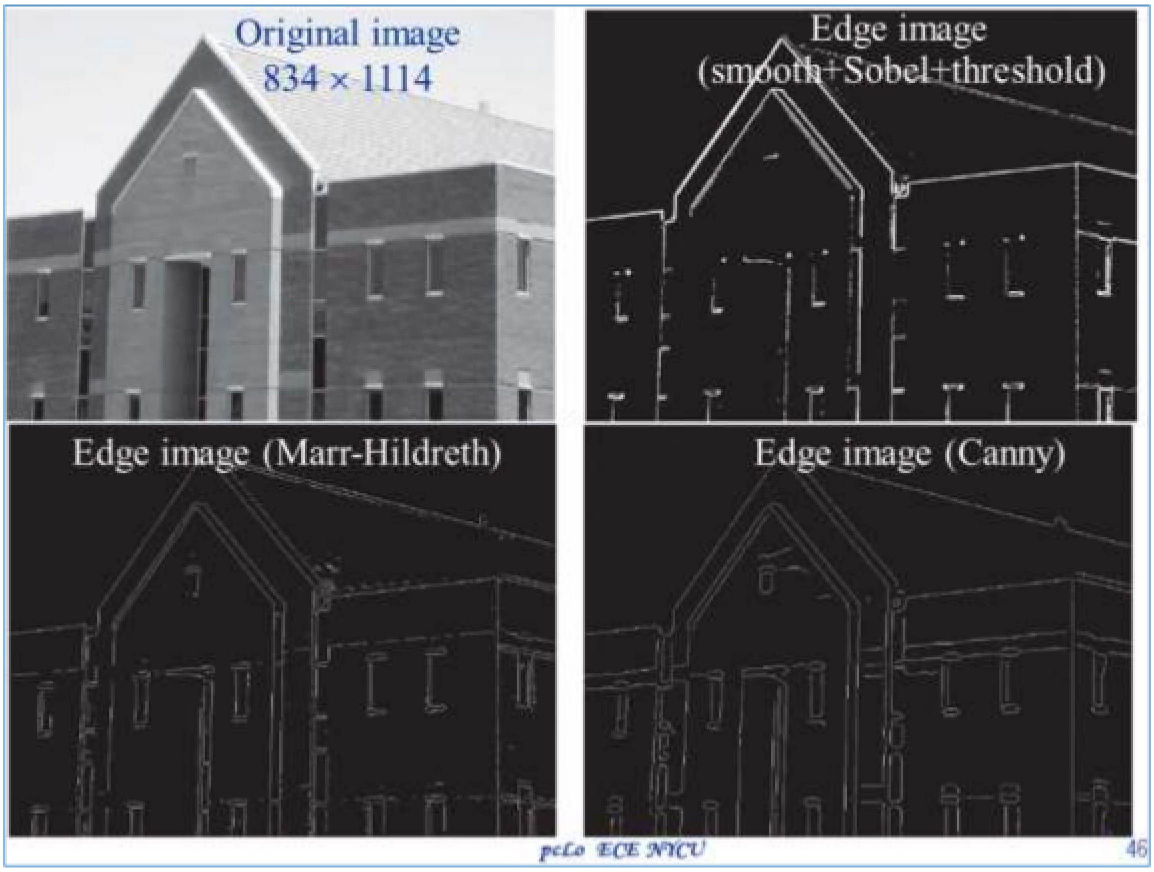
$$g_{NH}(x,y) = g_N(x,y) \text{ if } g_N(x,y) \geq T_H$$

$$g_{NL}(x,y) = g_N(x,y) \text{ if } T_H > g_N(x,y) \geq T_L$$

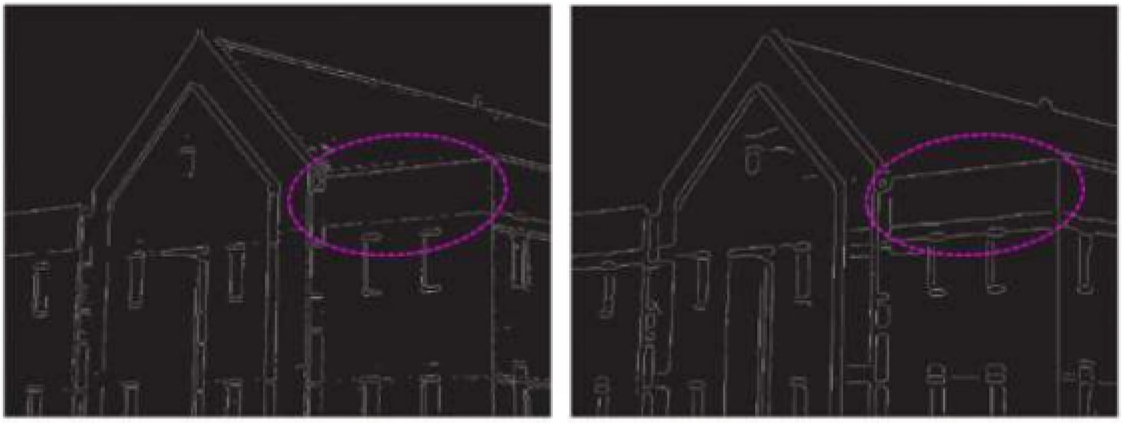
Hysteresis thresholding algorithm



pcLo ECE NYCU 45



pcLo ECE NYCU 46

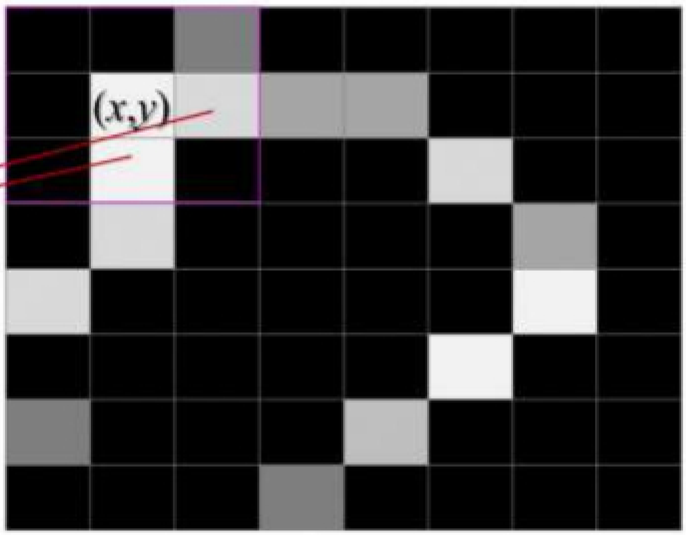


Edge image (Marr-Hildreth) Edge image (Canny)

$T_H = 0.10$ $T_L = 0.04$
LoG: 25×25 $\sigma = 4$

Edge Linking and Boundary Detection

Local Processing

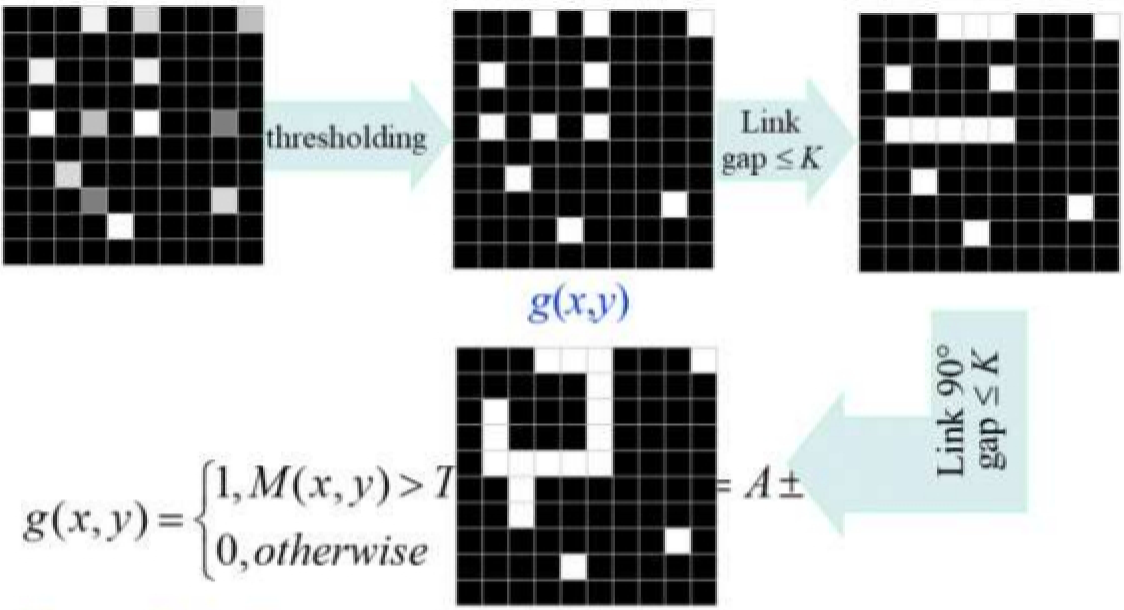


$$|M(a,b)-M(x,y)| \le T$$
$$|\alpha(a,b)-\alpha(x,y)| \le A$$

$$M(x,y) = \|\nabla f(x,y)\| = \sqrt{G_x^2(x,y) + G_y^2(x,y)}$$

$$\alpha(x,y) = \tan^{-1}\left[\frac{G_y(x,y)}{G_x(x,y)}\right]$$

$$\begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f(x,y)}{\partial x} \\ \dfrac{\partial f(x,y)}{\partial y} \end{bmatrix}$$



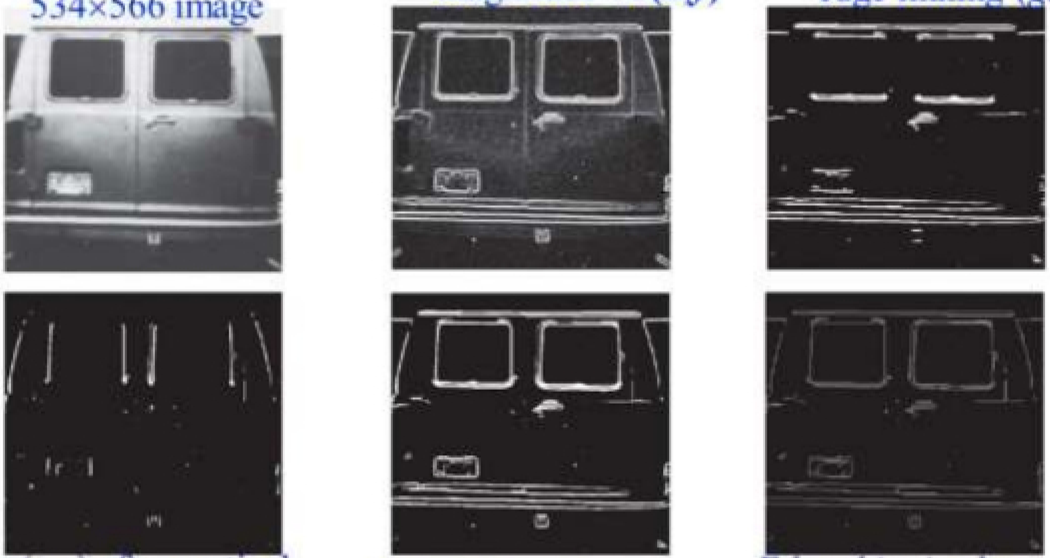
$$g(x,y) = \begin{cases} 1, M(x,y) > T \\ 0, otherwise \end{cases}$$

T_M : a threshold

A : a specified angle direction

$\pm T_A$: a band of acceptable directions about A

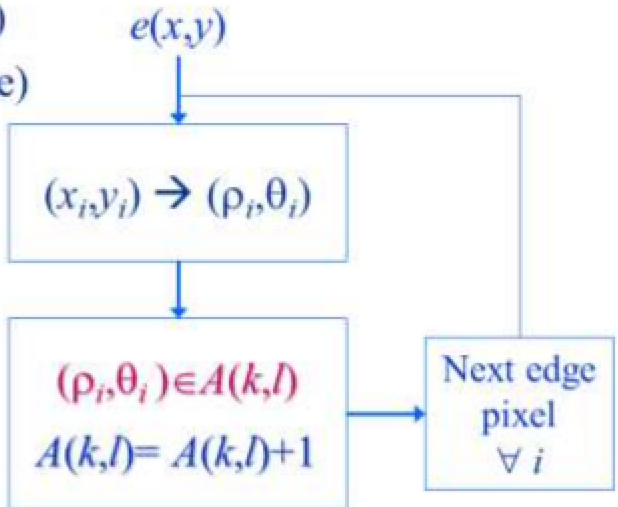
T_M=0.3max{$M(x,y)$}
T_A=45°, A=90°
K: 5% of image width (25 pixels)

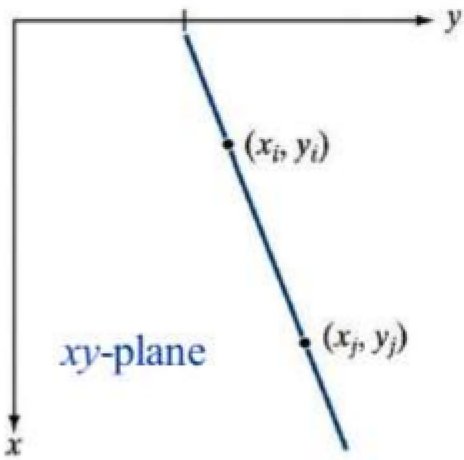


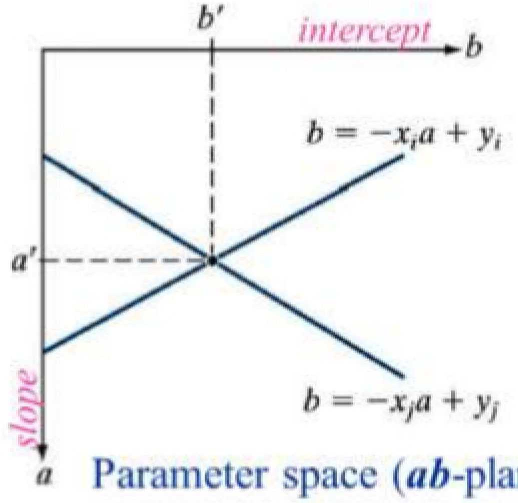
Global Processing (Hough Transform)

image coordinates (x_i, y_i)
parameter plane (ρθ-plane)

accumulator cells →







Parameter space (***ab***-plane)

$y_i = a\,x_i + b$

a: slope

b: intercept

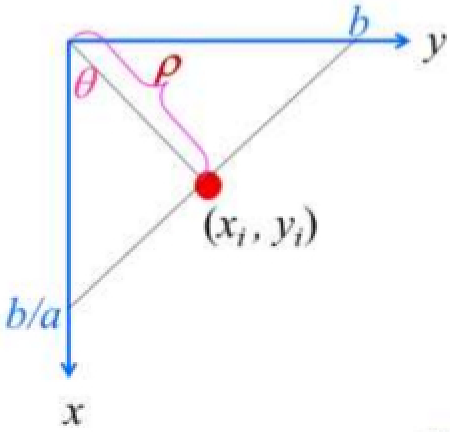
line connecting $(x_i\,, y_i)$ & $(x_j\,, y_j)$, $i \neq j$:

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

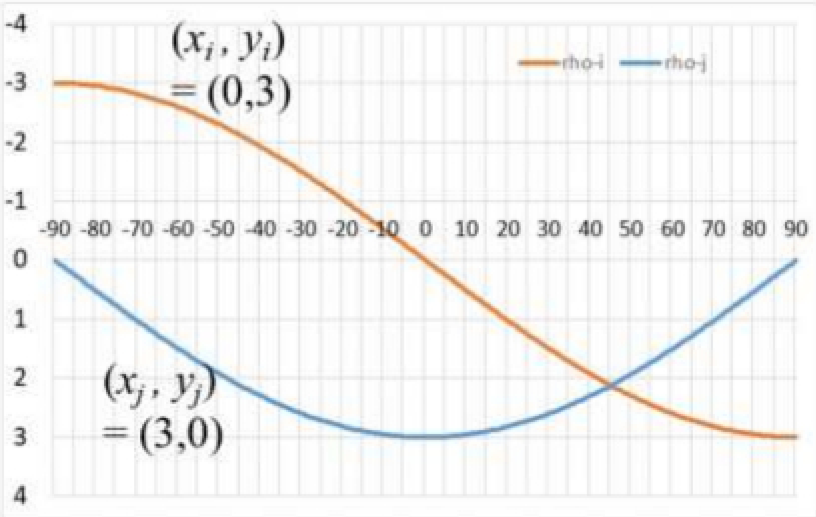
$$y_i = ax_i + b \xrightarrow{consider} ax_i + y_i = b$$

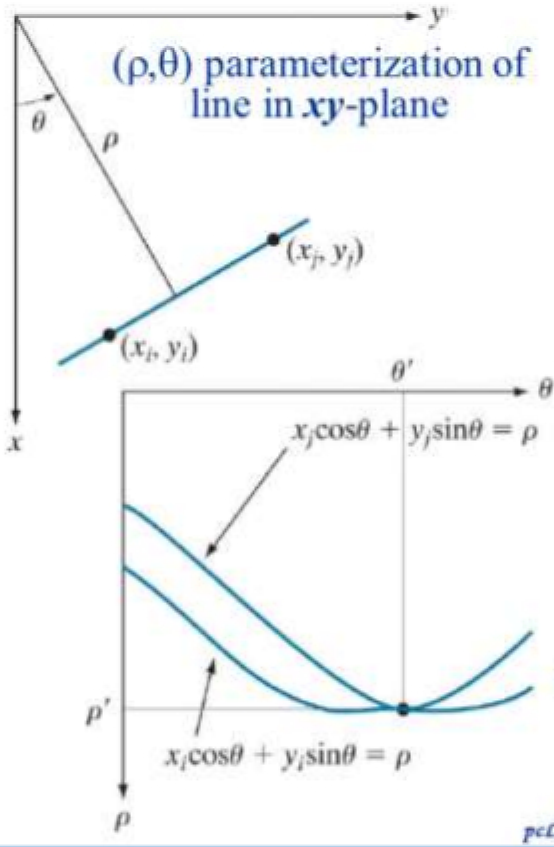
$$\frac{a}{\sqrt{a^2+1^2}}x_i + \frac{1}{\sqrt{a^2+1^2}}y_i = \frac{b}{\sqrt{a^2+1^2}}$$

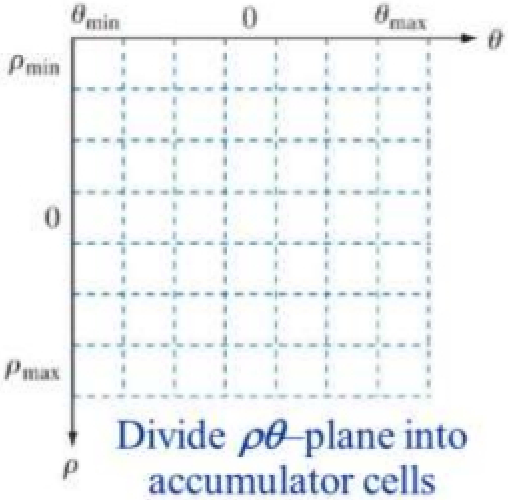
$$x_i \cos\theta + y_i \sin\theta = \rho$$



$$x_i\cos\theta + y_i\sin\theta = \rho$$







Divide $\rho\theta$–plane into accumulator cells

← Sinusoidal curves in $\rho\theta$–plane

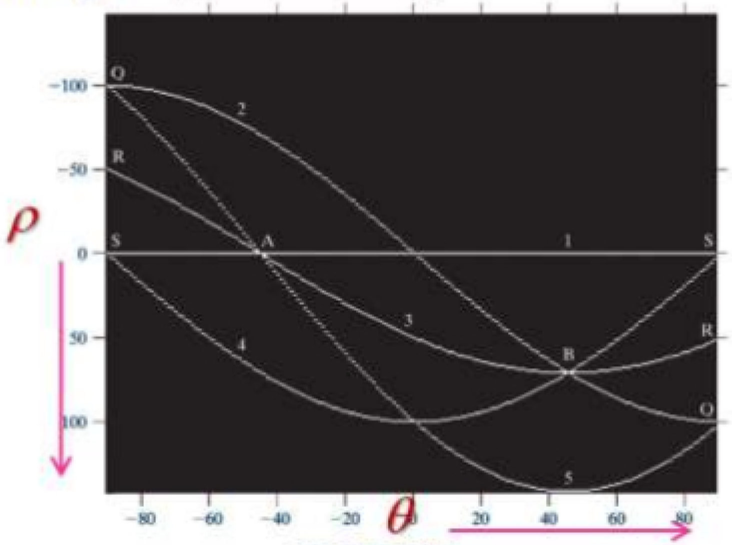
D=100

$$-\sqrt{2}D \le \rho \le \sqrt{2}D$$

$$-90^\circ \le \theta \le 90^\circ$$

$$x_i \cos\theta + y_i \sin\theta = \rho$$

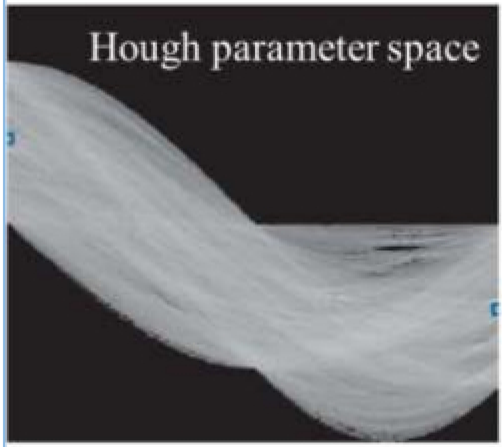
←Image 101×101, 5 white points



A 502×564 aerial image of an airport

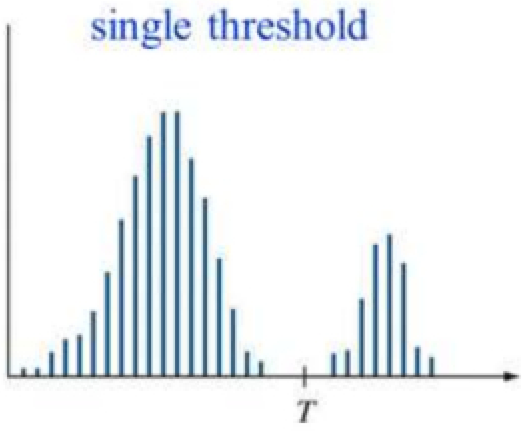
Edge image by Canny algorithm

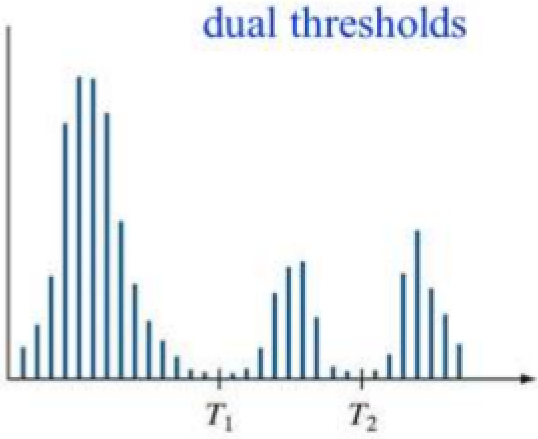
1° increments for θ
1 pixel increments for ρ



Lines superimposed on original image →

Thresholding





$$g(x,y)=\begin{cases}1, & f(x,y)>T\\ 0, & f(x,y)\le T\end{cases}\qquad g(x,y)=\begin{cases}a, & T_2<f(x,y)\\ b, & T_1<f(x,y)\le T_2\\ c, & f(x,y)\le T_1\end{cases}$$

$T=T[x,y,p(x,y),f(x,y)]$

$f(x,y)$: gray level of (x,y)

$p(x,y)$: local property of (x,y)

thresholded image: $g(x,y)=\begin{cases}1, & f(x,y)>T\\ 0, & f(x,y)\le T\end{cases}$

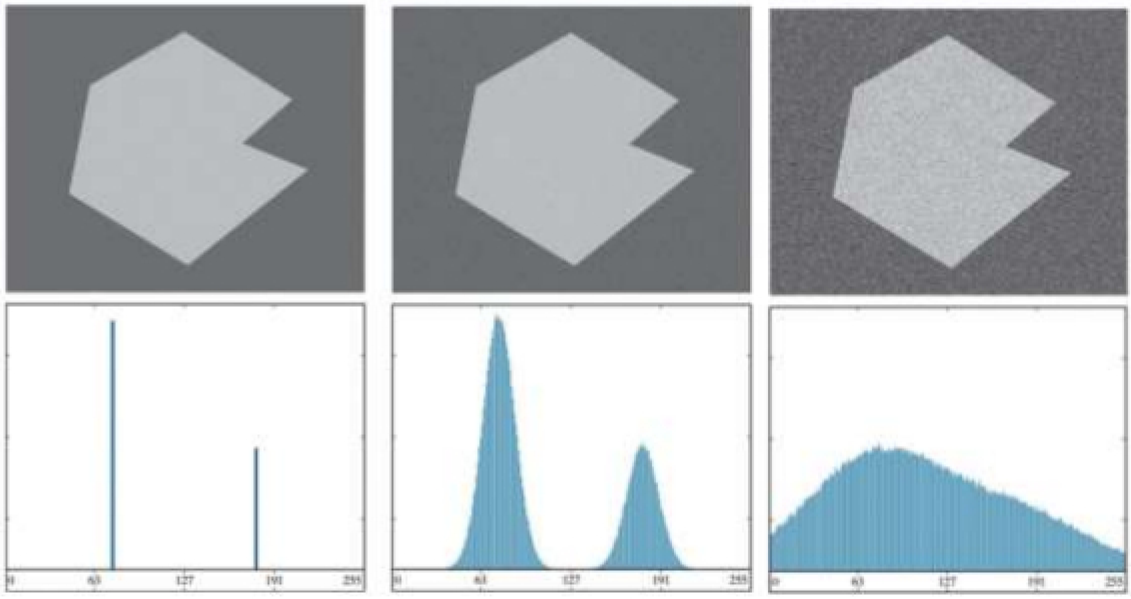
Global thresholding: T depends on $f(x,y)$ only

Local thresholding: T depends on $f(x,y)$ and $p(x,y)$

Dynamic (adaptive) thresholding: T depends on (x,y)

Effect of noise on histogram

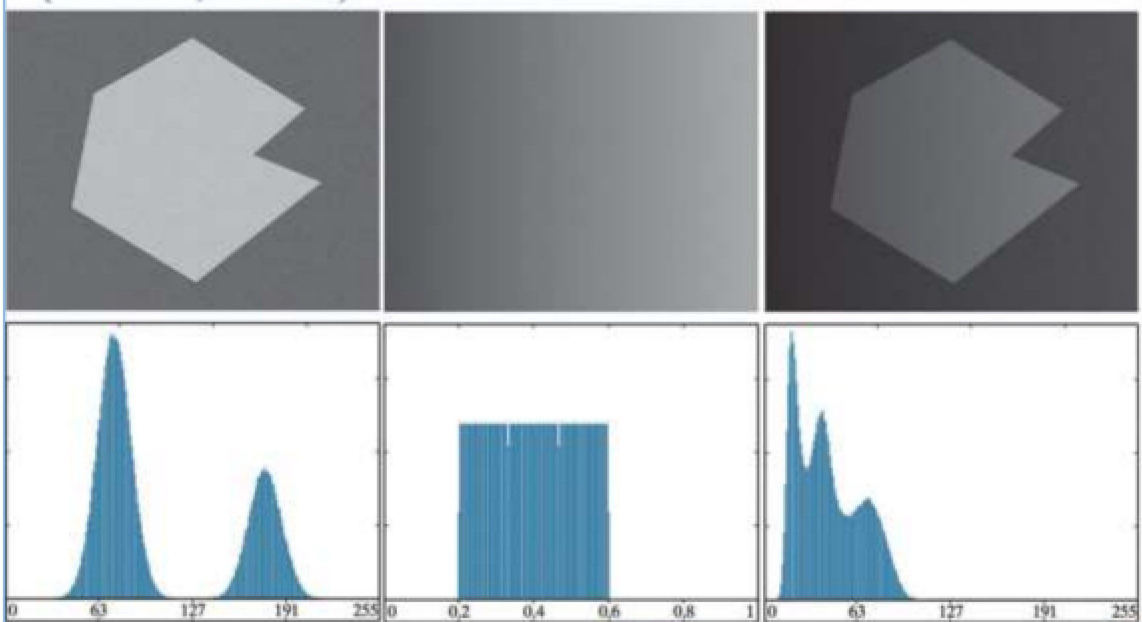
(noiseless) | Gaussian noise (mean=0, std=10) | Gaussian noise (mean=0, std=50)



pcLo ECE NYCU 63

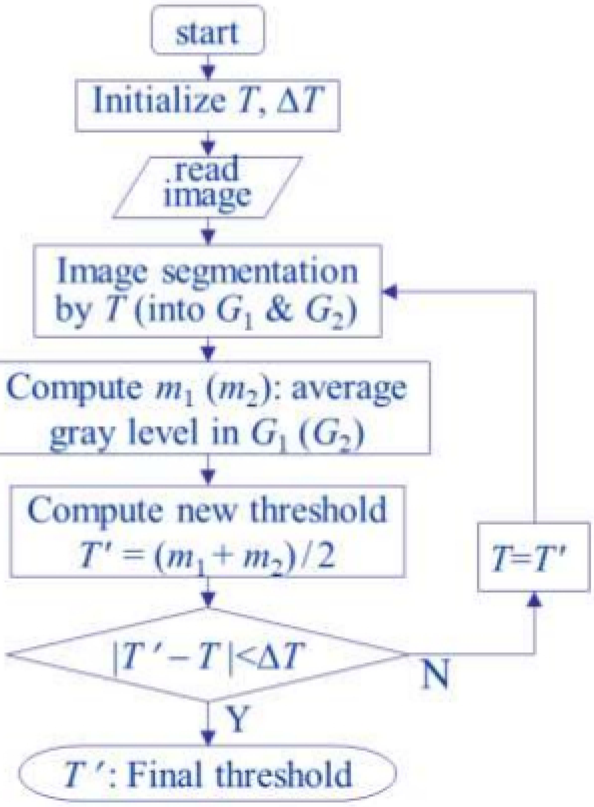
Effect of nonuniform illumination $i(x,y)$

Noisy image $r(x,y)$ (mean=0, std=10) | Illumination $i(x,y)$ | $f(x,y)=i(x,y)\cdot r(x,y)$



pcLo ECE NYCU 64

Global thresholding algorithm

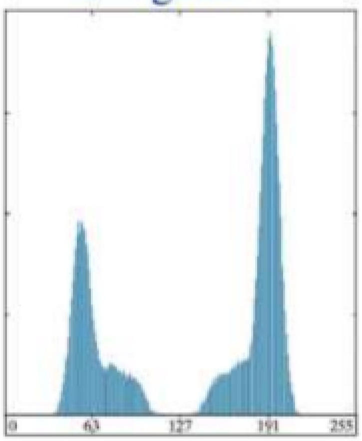


Normally, error $\Delta T = 0$

Initial T: average gray level

Original image
Noisy fingerprint

histogram



Result of segmentation with threshold (=125)

$\Delta T = 0$

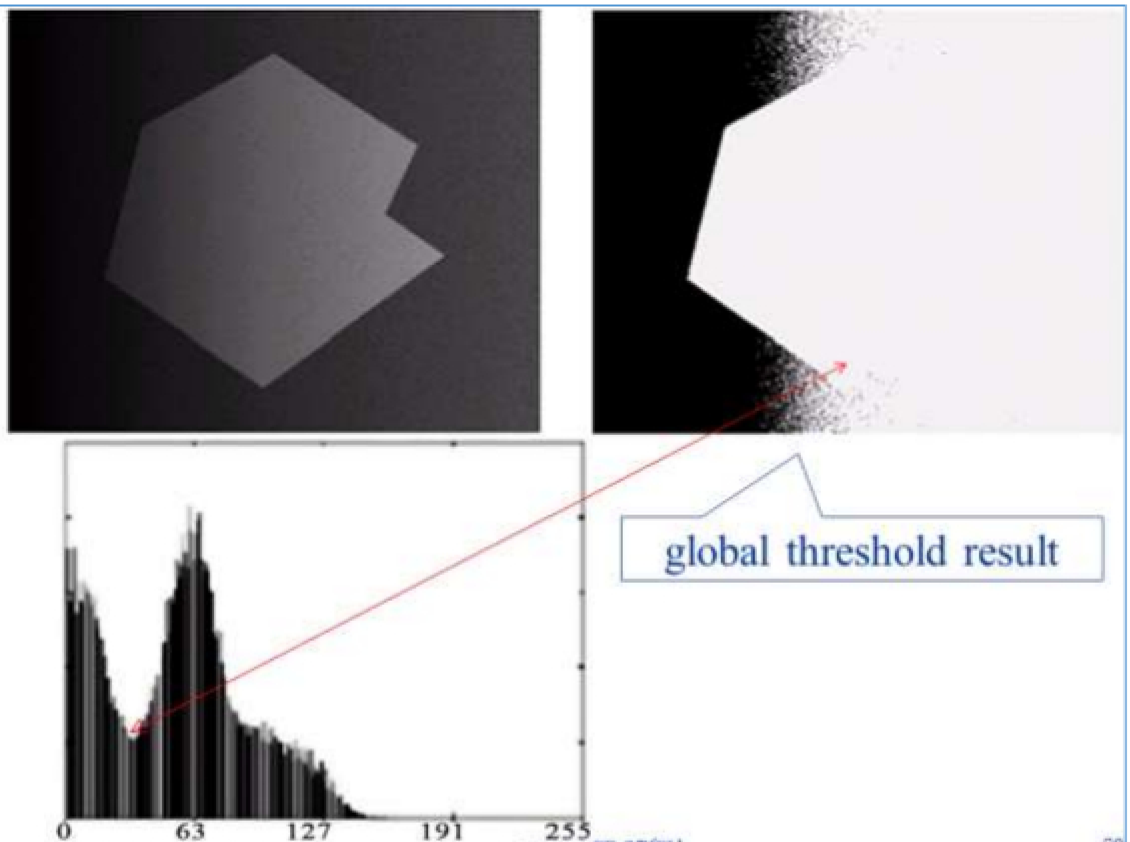
T=average gray level

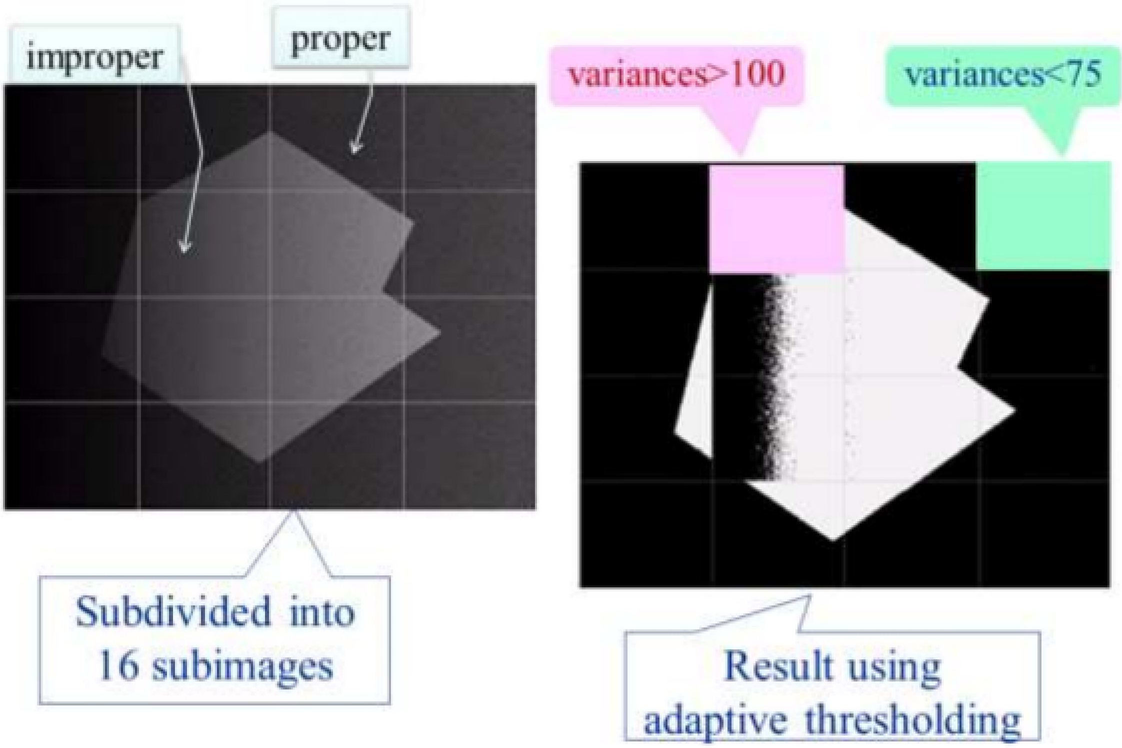
final threshold=125.4 (3 iterations)

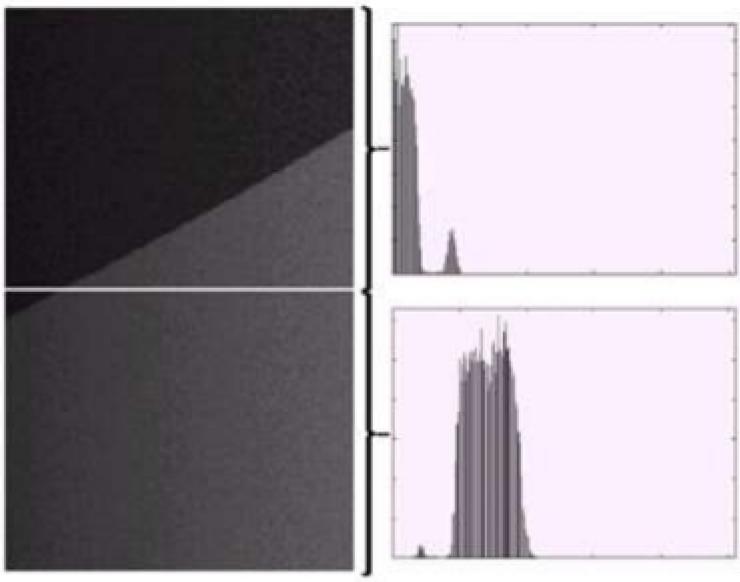
Adaptive Thresholding

Nonuniform illumination → a single global threshold cannot work

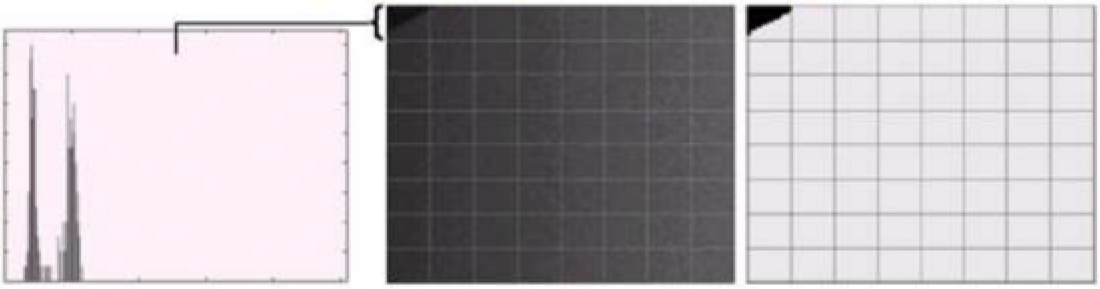
Divide the image → use different threshold for each subimage




improper
proper
variances>100
variances<75
Subdivided into 16 subimages
Result using adaptive thresholding



Further subdivision of the
improperly segmented subimage



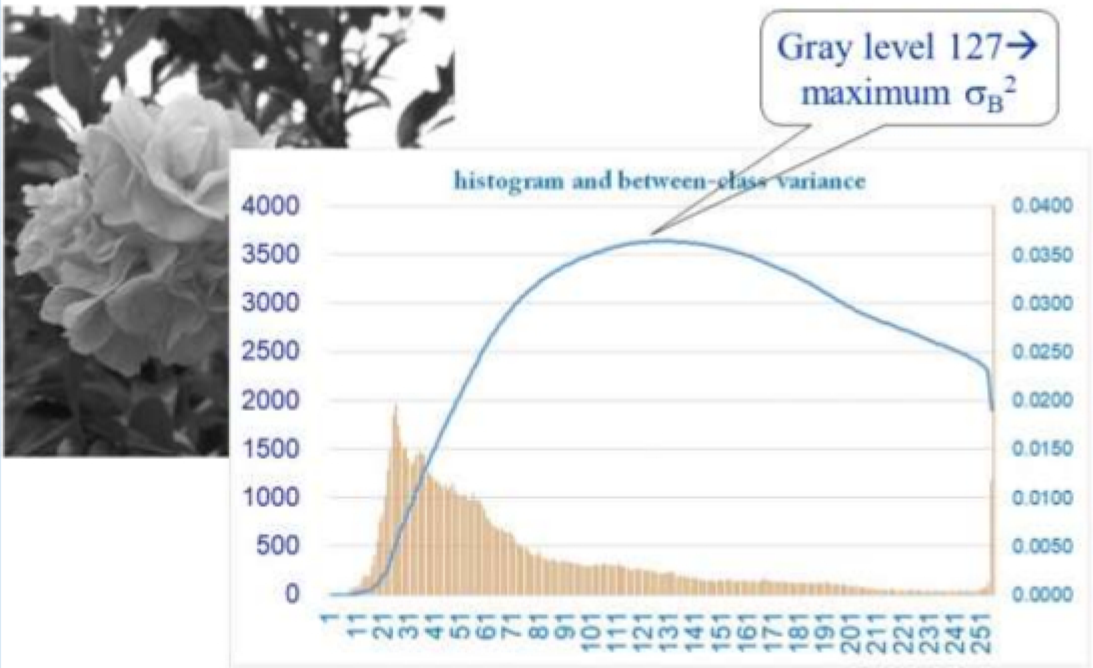
Optimal Global Thresholding — Otsu's method

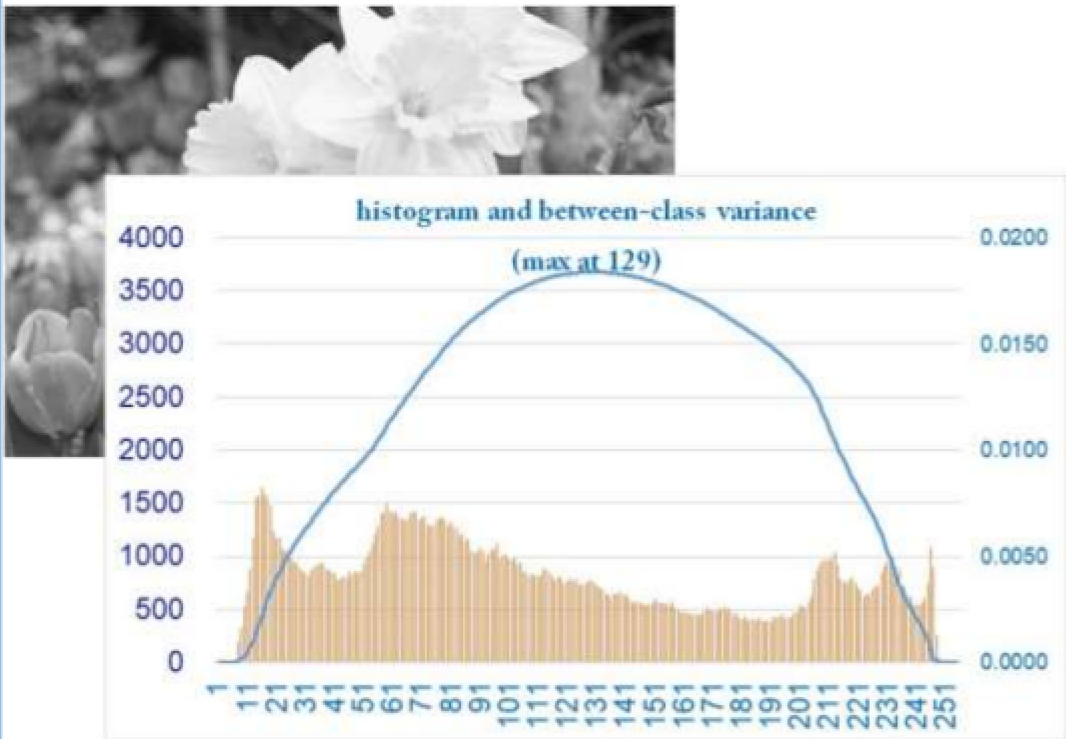
Optimization by maximizing the between-class variance

$$\eta = \frac{\sigma_B^2}{\sigma_G^2}, \begin{cases} \sigma_B^2 : \text{between - class variance} \\ \sigma_G^2 : \text{global variance} \end{cases}$$

η: separability measure

between-class variance: $\sigma_B^2 = \sum_{k=1}^{K} P_k \left(m_k - m_G \right)^2 \qquad P_k = \sum_{i \in C_k} p_i$


Gray level 127→ maximum σ_B^2
histogram and between-class variance
4000
3500
3000
2500
2000
1500
1000
500
0
0.0400
0.0350
0.0300
0.0250
0.0200
0.0150
0.0100
0.0050
0.0000
1 11 21 31 41 51 61 71 81 91 101 111 121 131 141 151 161 171 181 191 201 211 221 231 241 251


histogram and between-class variance
(max at 129)
4000
3500
3000
2500
2000
1500
1000
500
0
0.0200
0.0150
0.0100
0.0050
0.0000
1 11 21 31 41 51 61 71 81 91 101 111 121 131 141 151 161 171 181 191 201 211 221 231 241 251

$M \times N$ pixels, L intensity levels: $\{0, 1, 2, \ldots, L-1\}$

n_i: number of pixels with intensity level i

$$MN = n_0 + n_1 + n_2 + \cdots + n_{L-1}$$

Normalized histogram

$$\left\{ p_i = \frac{n_i}{MN} \middle| \ i = 0, 1, \cdots, L-1 \right\} \quad \text{So, } \sum_{i=0}^{L-1} p_i = 1$$

Segment the image into C_1 and C_2 by threshold k, $0 \le k < L-1$

$$\{ f(x,y) \in C_1 | \ 0 \le f(x,y) \le k \} \text{and}$$
$$\{ f(x,y) \in C_2 | \ k+1 \le f(x,y) \le L-1 \}$$

$P_1(k)/P_2(k)$: Probability of class C_1 / C_2 using threshold k

$$P_1(k) = \sum_{i=0}^{k} p_i \ \text{ and } \ P_2(k) = \sum_{i=k+1}^{L-1} p_i = 1 - P_1(k)$$

Mean intensity in C_1

$$m_1(k) = \sum_{i=0}^{k} i P\left(i / C_1 \right) \xrightarrow{\text{Bayes's theorem}} m_1(k) = \sum_{i=0}^{k} i \cdot \frac{P\left(C_1 / i \right) p_i}{P(C_1)}$$
$$= \frac{1}{P_1(k)} \sum_{i=0}^{k} i p_i$$

Mean intensity in C_2

$$m_2(k) = \sum_{i=k+1}^{L-1} i P\left(i / C_2 \right) = \frac{1}{P_2(k)} \sum_{i=k+1}^{L-1} i p_i$$

Cumulative mean up to level k → $m(k)=\sum_{i=0}^{k} ip_i$

Global mean→ $m_G=\sum_{i=0}^{L-1} ip_i$

$$\left\{\begin{aligned} m_1(k) &= \frac{1}{P_1(k)}\sum_{i=0}^{k} ip_i \\ m_2(k) &= \frac{1}{P_2(k)}\sum_{i=k+1}^{L-1} ip_i \end{aligned}\right.$$

$$m_G = P_1\cdot m_1 + P_2\cdot m_2$$

$$\text{Let}\begin{cases} P_1(k)\to P_1,\ \ m_1(k)\to m_1 \\ P_2(k)\to P_2,\ \ m_2(k)\to m_2 \end{cases}$$

Global variance $\sigma_G^2=\sum_{i=0}^{L-1}(i-m_G)^2 p_i$

Between-class variance

$$\sigma_B^2 = P_1(m_1-m_G)^2 + P_2(m_2-m_G)^2$$

$$\sigma_B^2 = P_1P_2(m_1-m_2)^2 = \boxed{\frac{(m_GP_1-m)^2}{P_1(1-P_1)}}$$

$$m=m(k)=\sum_{i=0}^{k} ip_i = m_1P_1$$

$$\boxed{\sigma_B^2(k^*) = \max_{0\le k\le L-1}\sigma_B^2(k)}$$

The *goodness* of the threshold at level k $\boxed{\eta=\frac{\sigma_B^2}{\sigma_G^2}}$

Otsu's Algorithm

$$\left\{ p_i = \frac{n_i}{MN} \middle| \, i = 0,1,\cdots,L-1 \right\}$$

$$m(k) = \sum_{i=0}^{k} i p_i \,, k = 0,\ldots,L-2 \qquad m_G = \sum_{i=0}^{L-1} i p_i$$

Normalized histogram → Cumulative sums $P_1(k)$ → Cumulative means $m(k)$ → Global mean m_G

$$P_1(k) = \sum_{i=0}^{k} p_i, k = 0,\ldots,L-2$$

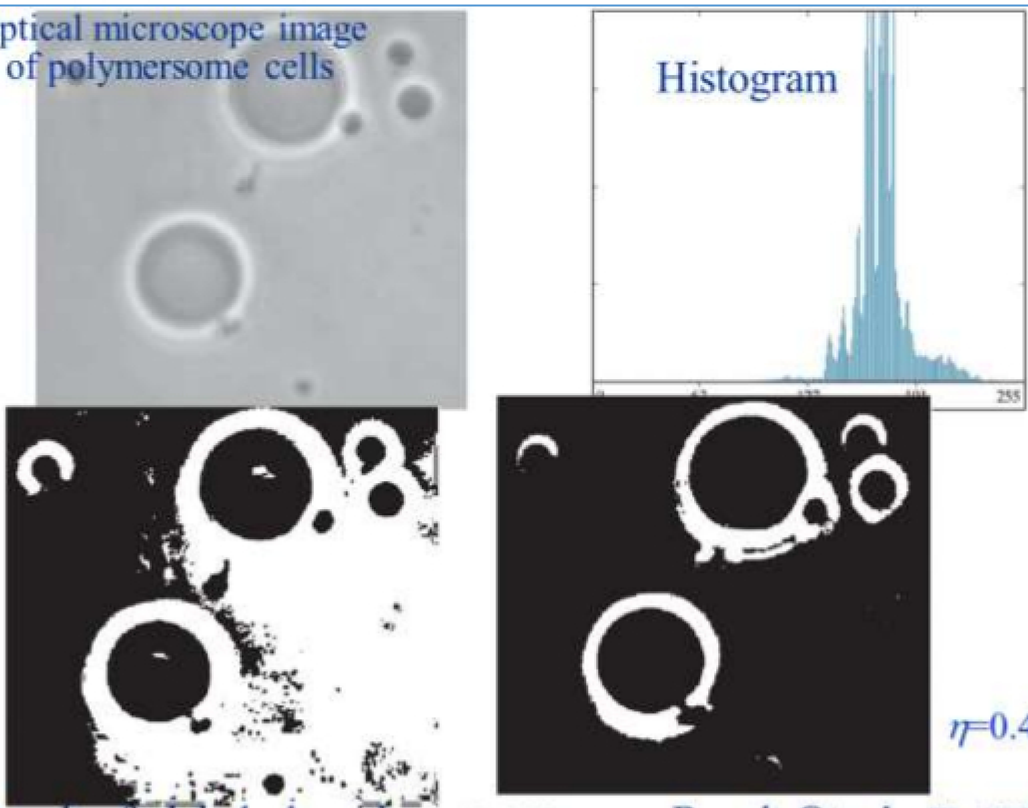
$$\sigma_B^2(k^*) = \max_{0 \le k \le L-1} \sigma_B^2(k)$$

Between-class variance → Otsu threshold k^* → Separability measure η^*

$$\sigma_B^2 = \frac{(m_G P_1 - m)^2}{P_1(1-P_1)}, k = 0,\ldots,L-2$$

$$\eta(k) = \frac{\sigma_B^2(k)}{\sigma_G^2}$$

pcLo ECE NYCU 79



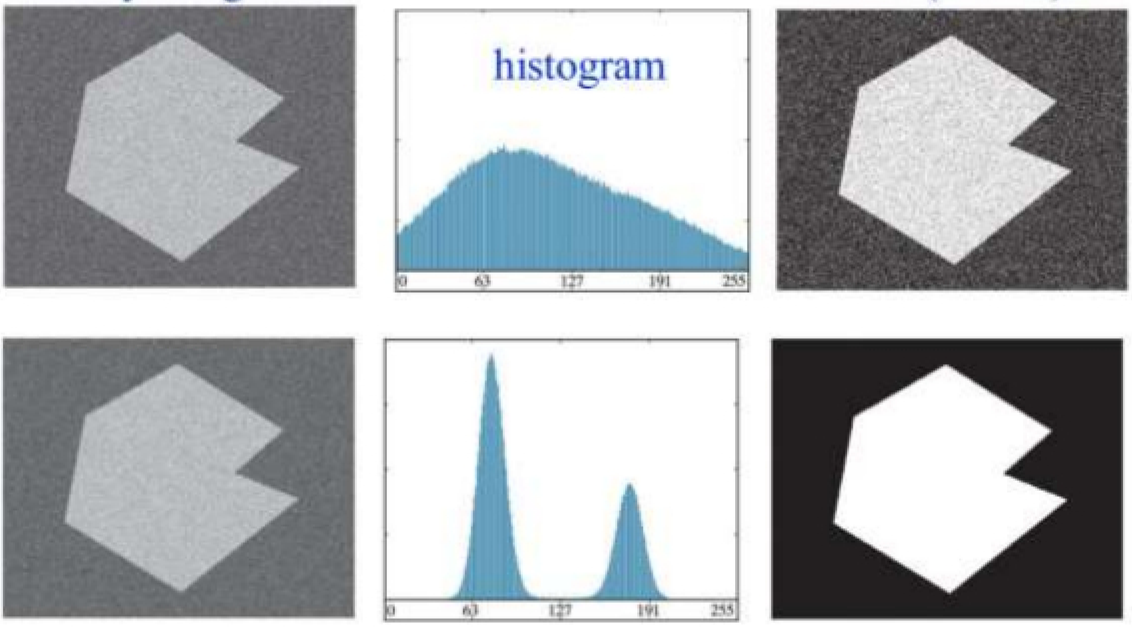
η=0.467

Result of global algorithm (T=169) Result Otsu's (T=182)

pcLo ECE NYCU 80

Noisy image

result (Otsu's)



Noisy image smoothed by 5×5 averaging mask

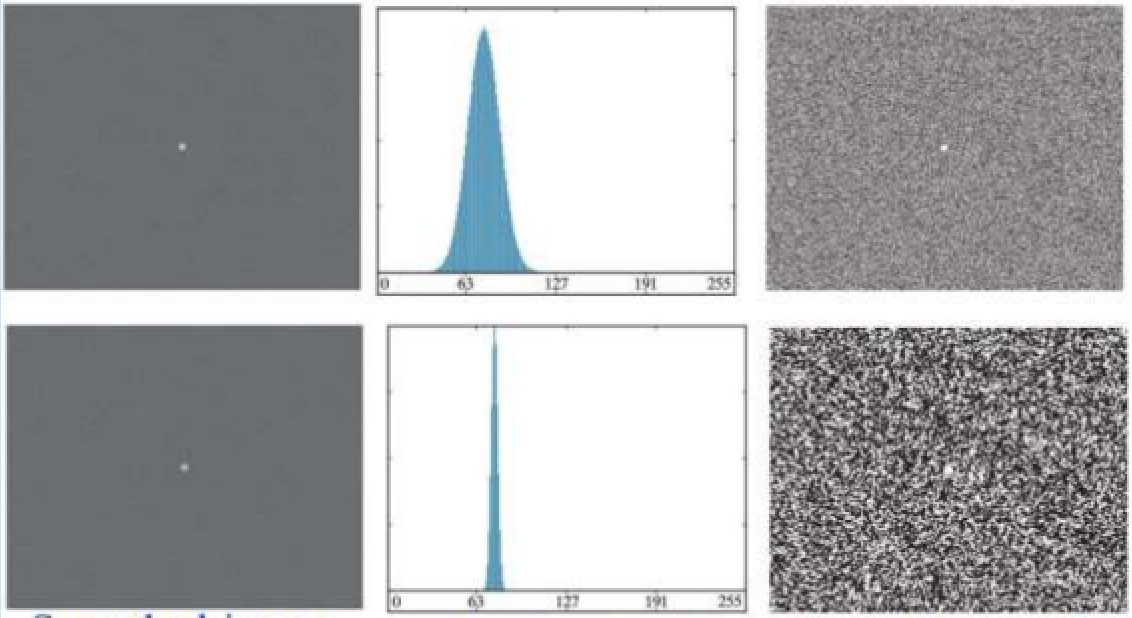
histogram

result (Otsu's)

Thresholding based on Boundary characteristics

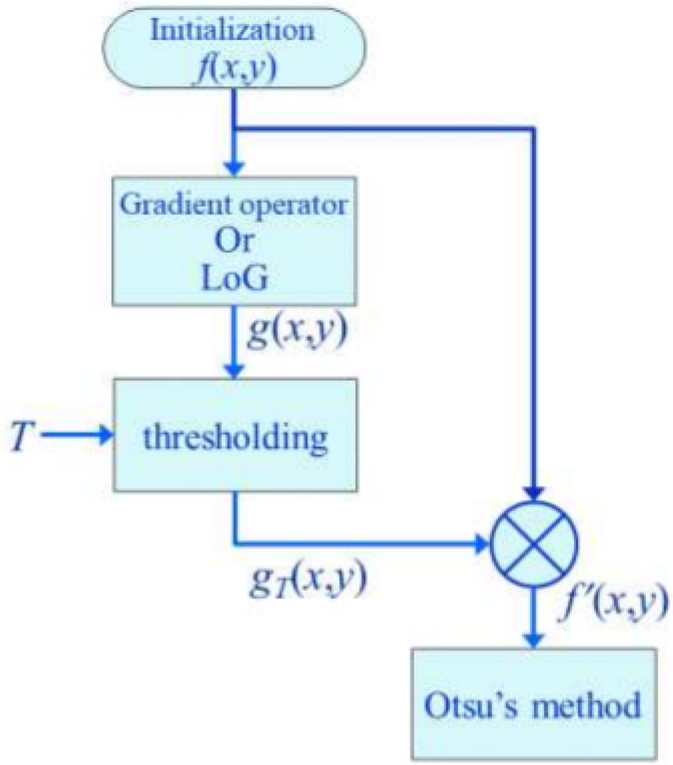
- Foreground region is comparably smaller than the background
- Significance of good shape of histogram
- *Gradient* or *Laplacian* improves shape of histogram

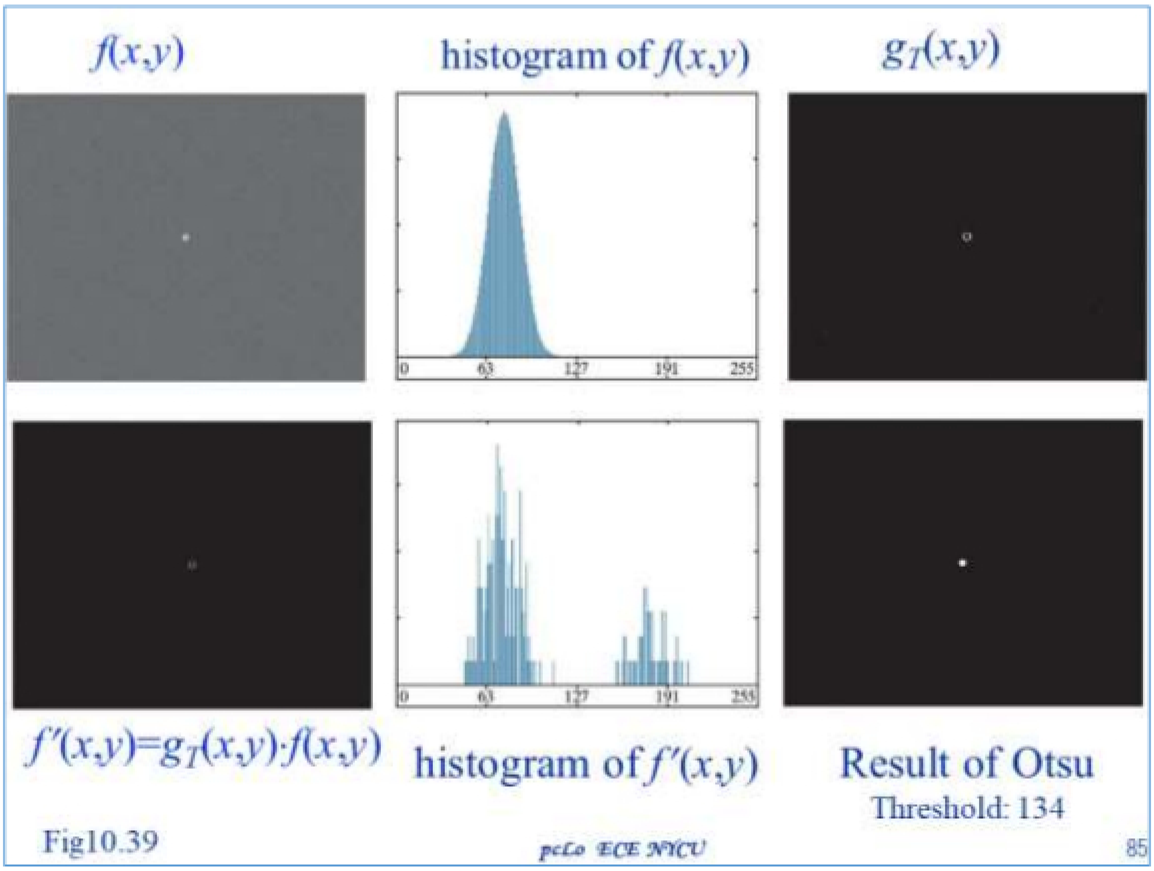
noisy image | histogram | result of Otsu's method

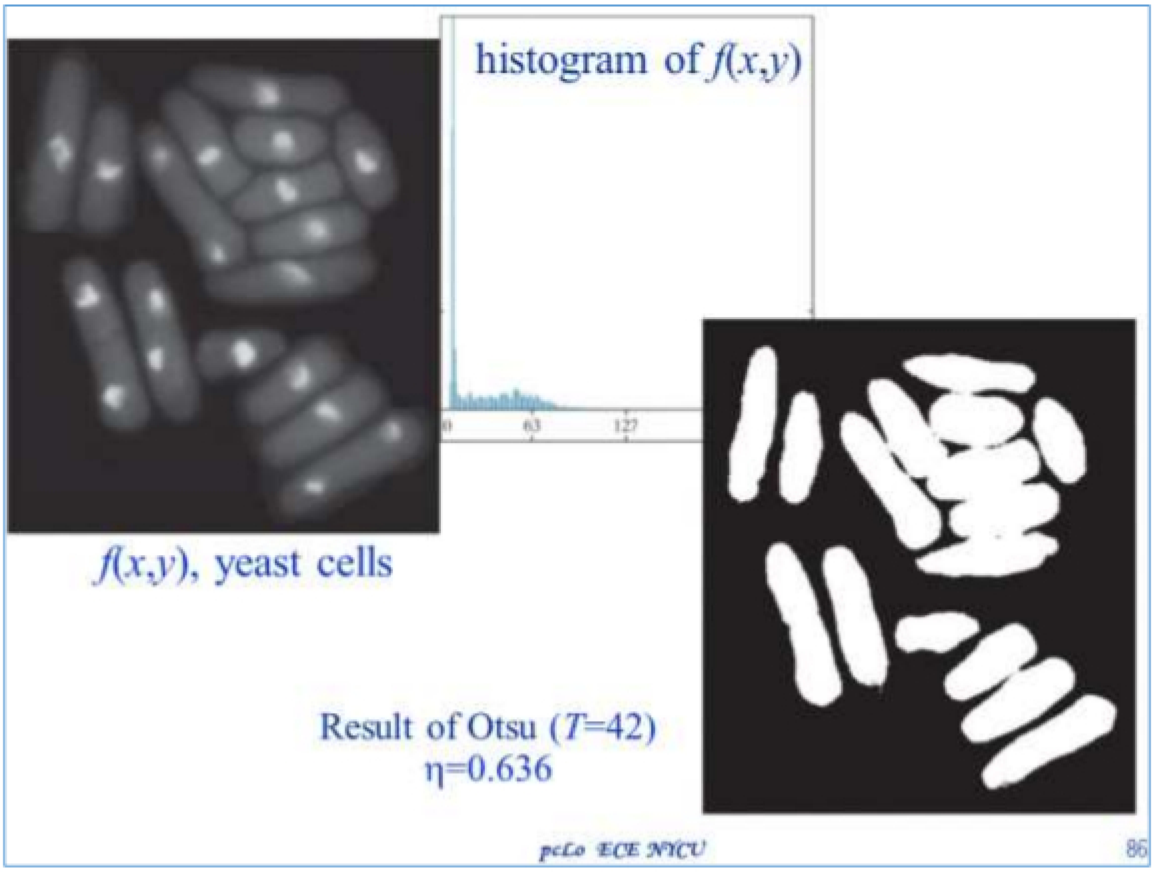


Smoothed image (5×5 averaging) | histogram | Result of Otsu's method

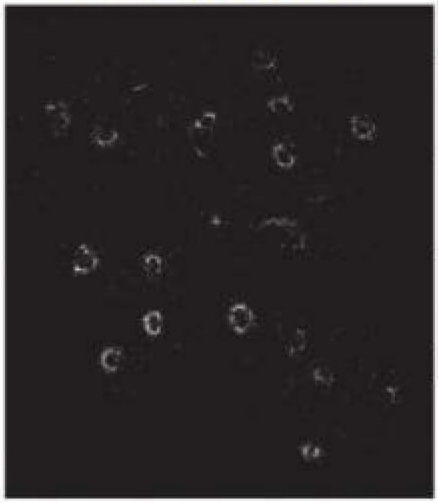
Fig10.38



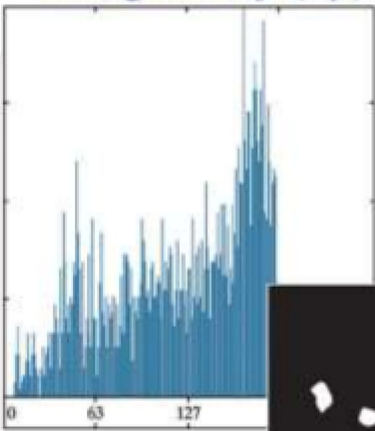

f(x,y)
histogram of f(x,y)
g_T(x,y)
0
63
127
191
255
f′(x,y)=g_T(x,y)·f(x,y)
histogram of f′(x,y)
Result of Otsu
Threshold: 134
Fig10.39
pcLo ECE NYCU
85

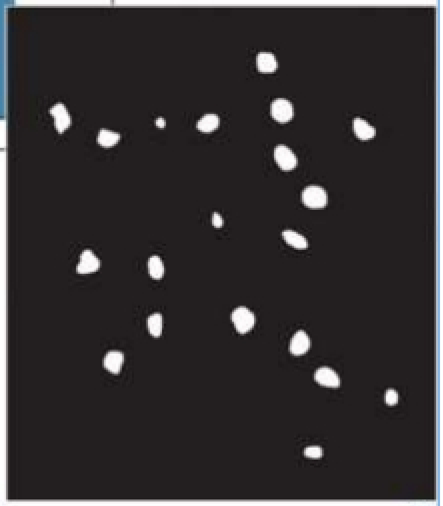

histogram of f(x,y)
0
63
127
f(x,y), yeast cells
Result of Otsu (T=42)
η=0.636
pcLo ECE NYCU
86

$f'(x,y)=f(x,y)g_T(x,y)$



histogram of $f'(x,y)$





Otsu's (T=115)
η=0.762

Form a 3-level image $s(x,y)$

$$s(x,y)=\begin{cases}0, \nabla f < T\\ +, \nabla f \geq T \text{ and } \nabla^2 f \geq 0\\ -, \nabla f \geq T \text{ and } \nabla^2 f < 0\end{cases}$$

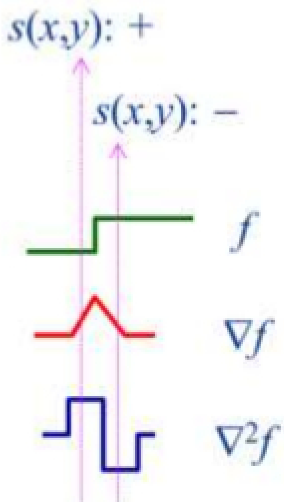
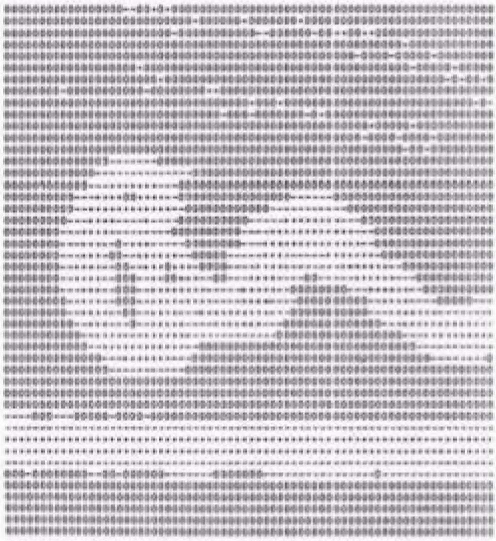
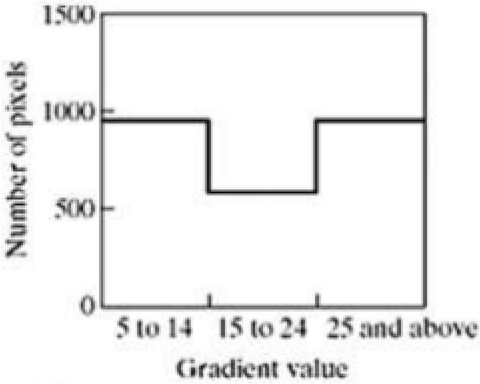


Image of a handwritten stroke

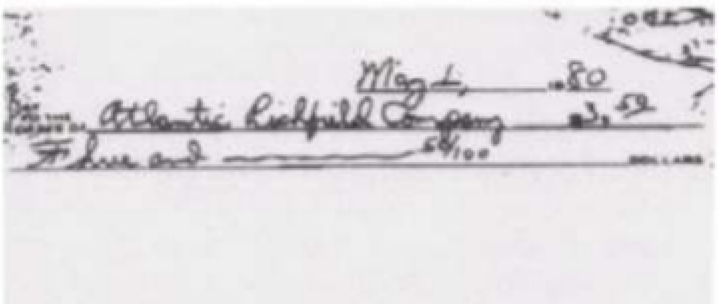


$s(x,y)$



histogram of gradient values (T>5)

scenic bank check image



segmented image (T≈20)

Otsu's algorithm for multiple thresholds

For K classes : $C_1, C_2, \ldots, C_K$

between-class variance: $\sigma_B^2 = \sum_{k=1}^{K} P_k \left(m_k - m_G\right)^2$

$$P_k = \sum_{i \in C_k} p_i \qquad m_k = \frac{1}{P_k} \sum_{i \in C_k} i p_i$$

m_G: global mean

K-1 thresholds $\{k_1^*, k_2^*, \ldots, k_{K-1}^*\}$ to maximize

$$\max_{0<k_1<k_2<\ldots<k_{K-1}<L-1} \sigma_B^2\left(k_1, k_2, \ldots, k_{K-1}\right)$$

Consider 3 classes → 2 thresholds k_1 & k_2

$$\sigma_B^2 = P_1\left(m_1 - m_G\right)^2 + P_2\left(m_2 - m_G\right)^2 + P_3\left(m_3 - m_G\right)^2$$

$$\begin{cases} P_1 = \sum_{i=0}^{k_1} p_i \\ P_2 = \sum_{i=k_1+1}^{k_2} p_i \\ P_3 = \sum_{i=k_2+1}^{L-1} p_i \end{cases} \qquad \begin{cases} m_1 = \frac{1}{P_1} \sum_{i=0}^{k_1} i p_i \\ m_2 = \frac{1}{P_2} \sum_{i=k_1+1}^{k_2} i p_i \\ m_3 = \frac{1}{P_3} \sum_{i=k_2+1}^{L-1} i p_i \end{cases}$$

$$P_1 m_1 + P_2 m_2 + P_3 m_3 = m_G$$
$$P_1 + P_2 + P_3 = 1$$

$$\sigma_B^2\left(k_1^*, k_2^*\right) = \max_{0<k_1<k_2<L-1} \sigma_B^2\left(k_1, k_2\right)$$

k_1^*, k_2^*: optimal threshold values

$k_1 = 0 - L\text{-}3$

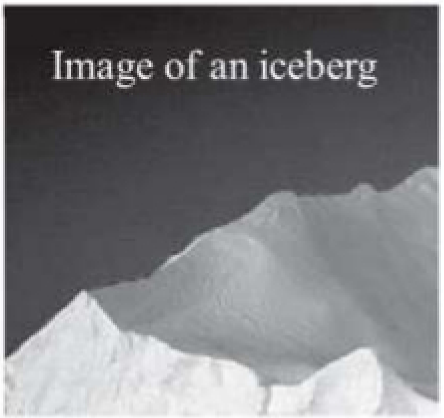
$k_2 = 1 - L\text{-}2\ (> k_1)$

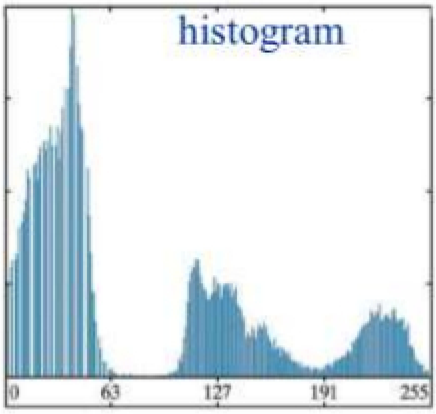
$$g(x,y) = \begin{cases} a & \text{if } f(x,y) \le k_1^* \\ b & \text{if } k_1^* < f(x,y) \le k_2^* \\ c & \text{if } f(x,y) > k_2^* \end{cases}$$

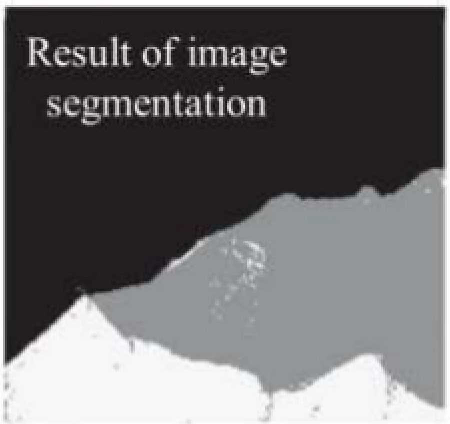
a, b, c: 3 distinct intensity values

$$\eta(k_1^*, k_2^*) = \frac{\sigma_B^2(k_1^*, k_2^*)}{\sigma_G^2} \quad \text{separability measure}$$

σ_G^2: variance of the entire image







$k_1^* = 80$, $k_2^* = 177$

separability = 0.954

Fig10.42

Region-Based Segmentation

Basic formulation

R: entire image region

R_i i=1,…, K: subregions satisfy:

① $\bigcup_{i=1}^{K} R_i = R$ ⇨ segmentation must be complete

② R_i (i=1,…,K) is a connected region ⇨ points in a region must satisfy some connectivity criterion

③ $R_i \cap R_j = \varnothing$ for all $i \neq j$ ⇨ regions must be disjoint

④ $P(R_i)$=TRUE for i=1,…, K ⇨ all pixels in R_i must satisfy predicate P

⑤ $P(R_i \cup R_j)$=FALSE for $i \neq j$ ⇨ regions R_i and R_j are different according to the criteria given in predicate P

Region growing

- *region growing*: group pixels or subregions into larger regions
- *pixel aggregation*:
  - select **seed** points,
  - grow regions by similarity

seed points $f(2,1)=1$ $f(2,3)=7$
predicate P: $|f(x,y)-f(\text{seed})|<T$

$y \rightarrow$	**0**	**1**	**2**	**3**	**4**
0	0	0	5	6	7
1	1	1	5	8	7
2	0	1	6	7	7
3	2	0	7	6	6
4	0	1	5	6	5
$\uparrow x$					

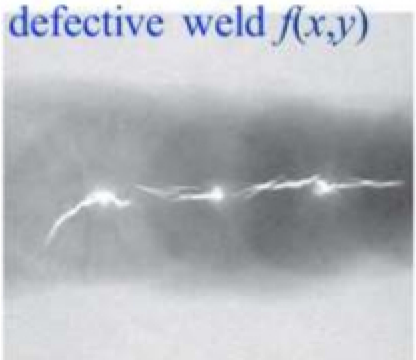
0	0	5	6	7
1	1	5	8	7
0	1	6	7	7
2	0	7	6	6
0	1	5	6	5

$T=3$
result: 2 disjointed regions

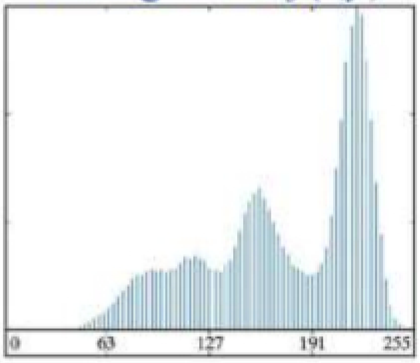
0	0	5	6	7
1	1	5	8	7
0	1	6	7	7
2	0	7	6	6
0	1	5	6	5

$T=8$
result: a single region

X-ray image of a defective weld $f(x,y)$



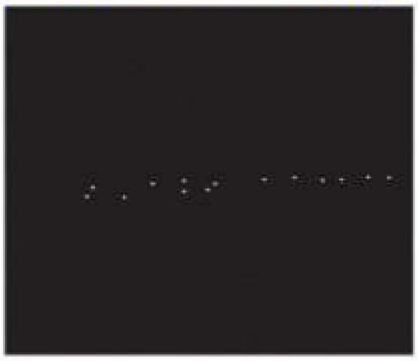
Histogram of $f(x,y)$



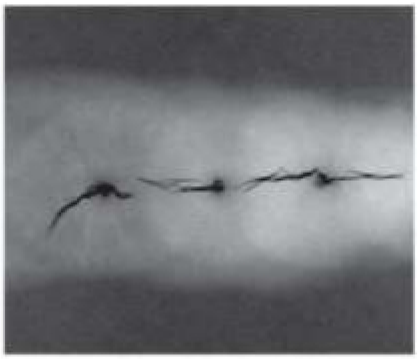
Thresholded image (T_S= 254)

Fig10.46 (a)-(c)

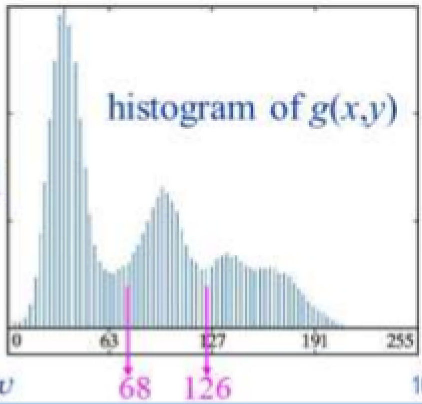
Final seed image $s(x,y)$, morphological erosion of thresholded image



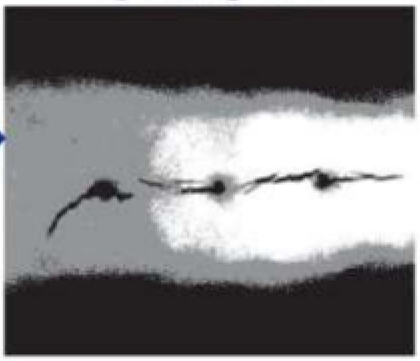
$g(x,y)= |f(x,y)–255|$



3 principal modes →

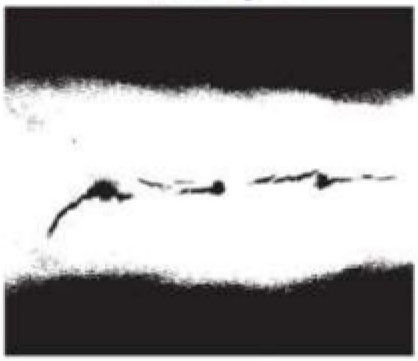


$g(x,y)$ thresholded by T_1=68, T_2=126



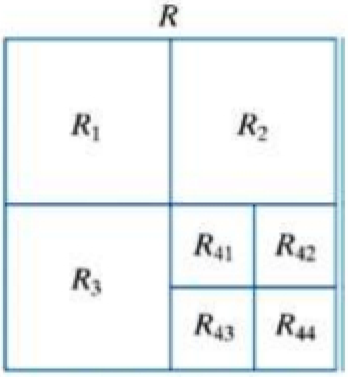
Fail→

$g(x,y)$ thresholded with T_1=68

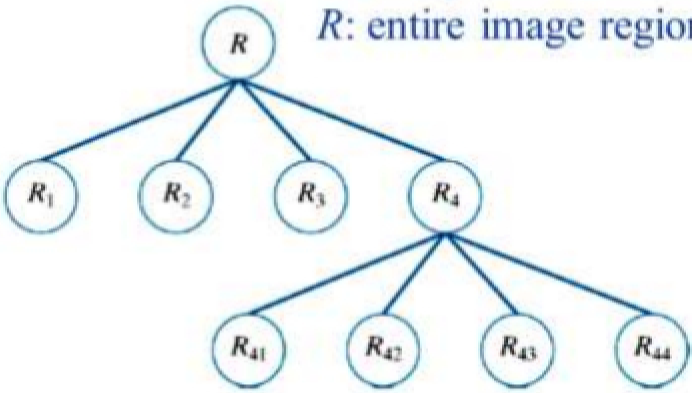


← Segmentation result by region growing

Region splitting and merging



partitioned image

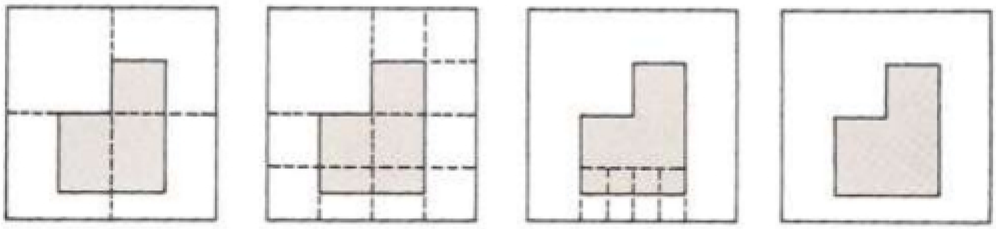


quadtree.
R: entire image region

(*quadtree*: each node is subdivided into 4 descendants):

❶$P(R_i)$=FALSE → R_i is split into 4 disjointed quadrants
❷$P(R_j \cup R_k)$=TRUE→ merge adjacent regions R_j and R_k
❸stop when no further merging or splitting

Example of split-and-merge algorithm



$P(R_i)$=TRUE if all pixels in R_i have the same intensity

(a)$P(R)$=FALSE (R: entire image)$\Rightarrow$ split

(b)$P(R_{UL})$=TRUE$\Rightarrow$skip,

$P(R_{UR})$, $P(R_{LL})$, $P(R_{LR})$=FALSE$\Rightarrow$split

Original image	Result of split-merge	Result of thresholding

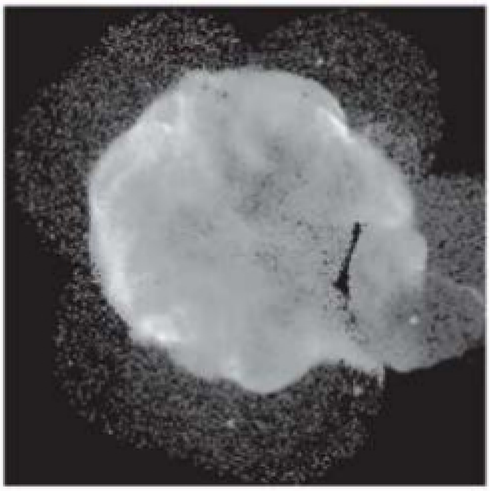
$P(R_i)$: at least 80% pixels in R_i have gray level z_j

$$|z_j - m_i| \le 2\sigma_i$$

m_i : mean gray level of R_i

σ_i : std of gray levels in R_i

566×566 X-ray image of Cygnus Loop supernova
Aim: segment the *ring* of less dense matter



$$Q(R) = \begin{cases} True, \text{if } \sigma_R > a \text{ and } 0 < m_R < b \\ False, \text{ otherwise} \end{cases}$$

σ_R / m_R: std/mean of R

Fig10.48

Segmentation result
quadregion: 32×32

8×8

16×16

$Q(R)$: $\sigma_R > 10$ and $0 < m_R < 125$
Min size of quadregions:
32×32, 16×16, 8×8
$Q(R)$=True → white
$Q(R)$= False → black

Segmentation using Clustering and Superpixels

Region segmentation using K-means clustering

$\{z_1, z_2, \ldots, z_Q\}$: a set of vector observations

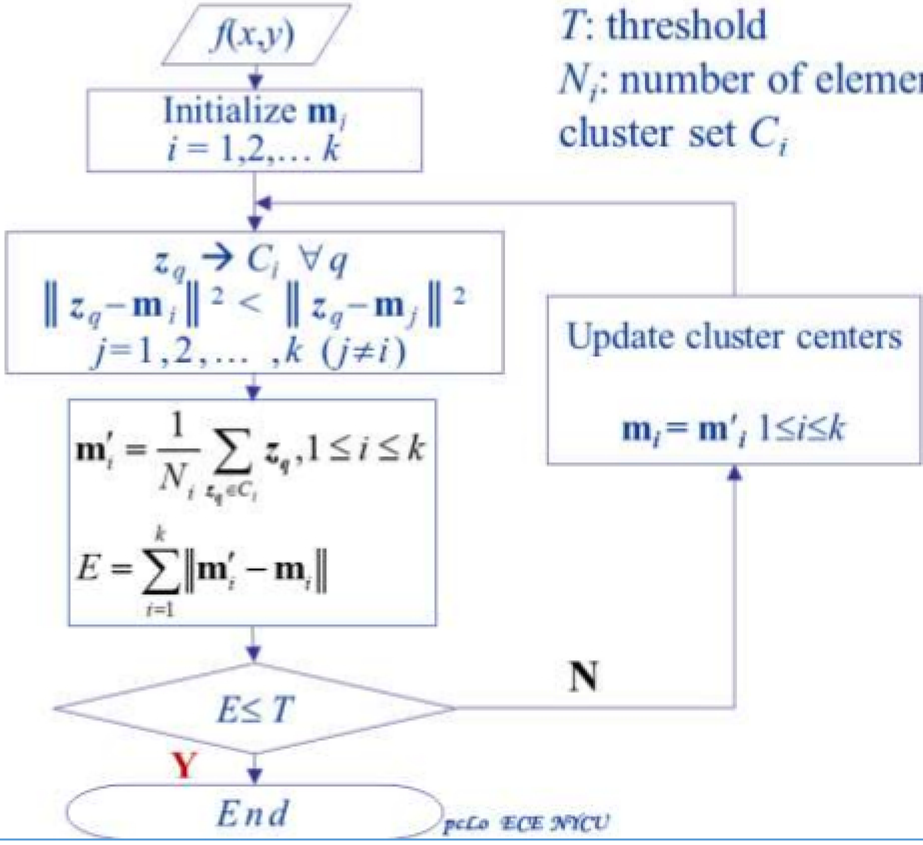
Find $C = \{C_1, C_2, \ldots, C_k\}$ to minimize the sum of distances from each point in the set to the *centroid* of the set ($k \le Q$)

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix}$$

$$\arg\min_{C} \left(\sum_{q=1}^{Q} \sum_{j=1}^{k} \| z_q - \mathbf{m}_j \| \right)$$

$\mathbf{m}_j$: mean vector (centroid) in C_j

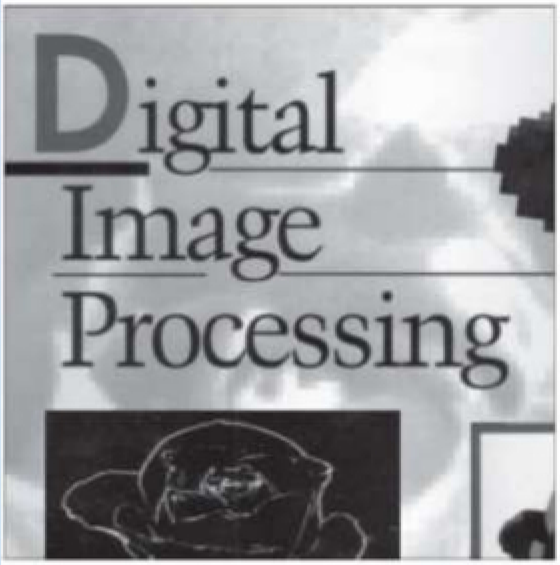
$\| z_q - \mathbf{m}_j \|$: Euclidean norm



T: threshold

N_i: number of elements in cluster set C_i

Image of size 688×688



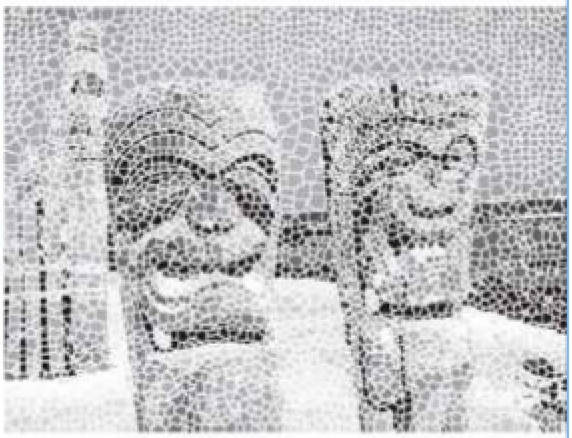
Result of k-means, k=3

(z: gray-level intensity)

Fig10.49

Region segmentation using Superpixels

Image of size 600×480
(480,000) pixels



4,000 superpixels and
their boundaries

Superpixel image
Generated by 4000 superpixels

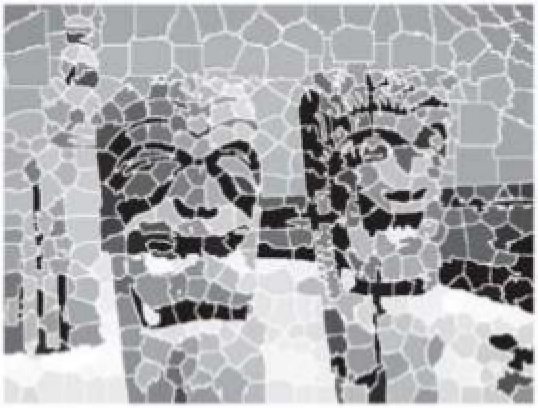
Image of size 600×480
(480,000) pixels

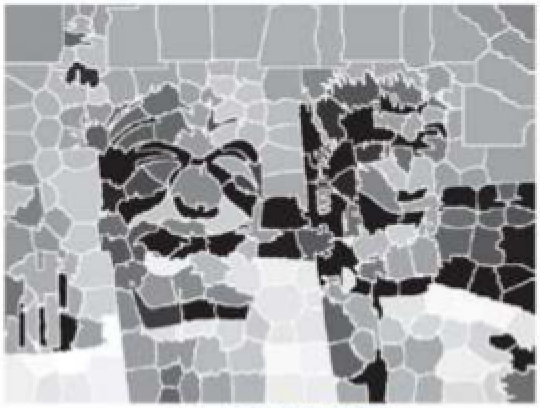
Superpixel image
Generated by 40000 superpixels

Original image

difference image

113





114



SLIC (simple linear iterative clustering) algorithm

$$z = \begin{bmatrix} r \\ g \\ b \\ x \\ y \end{bmatrix}$$

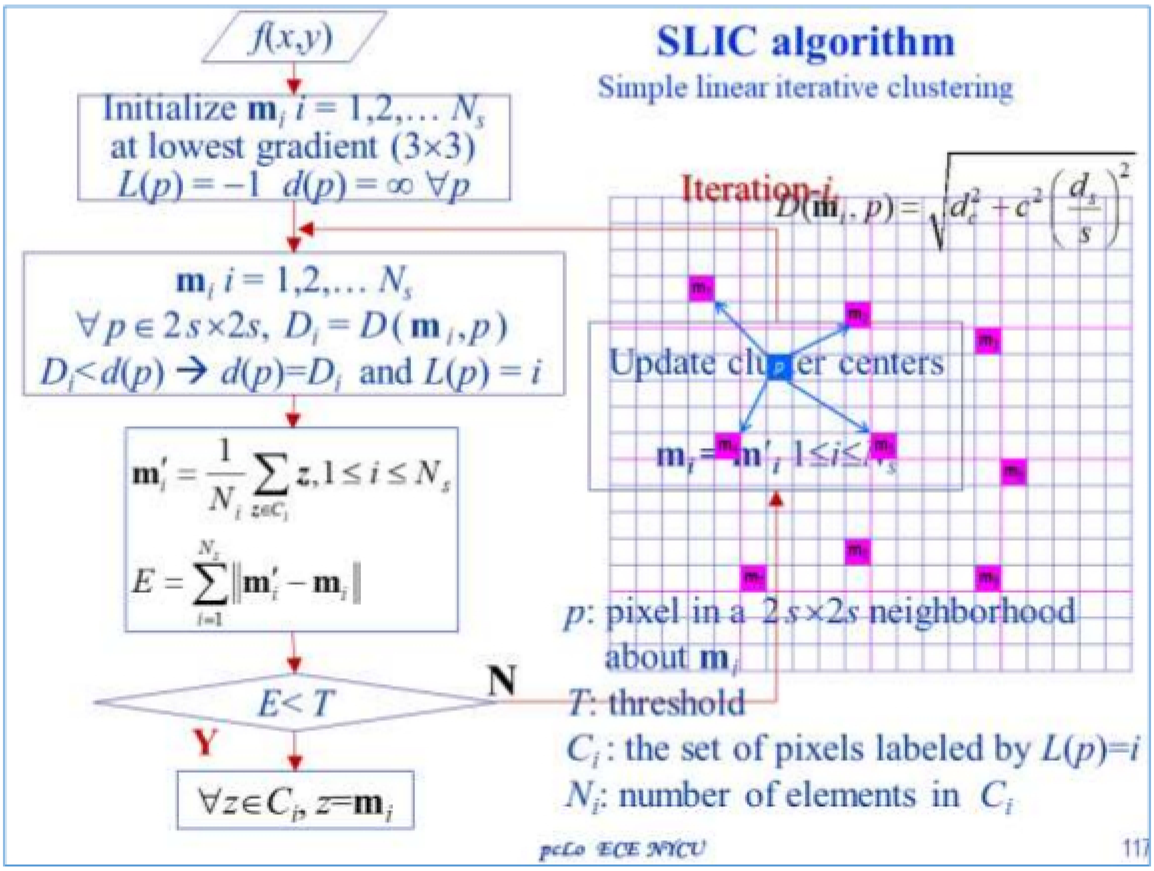
N_s: desired number of superpixels

N_t: total number of pixels in the image

$\mathbf{m}_i=[r_i\ g_i\ b_i\ x_i\ y_i]^T$, $i=1,\ldots, N_s$

Initial location of $\mathbf{m}_i$:
Sample image at grid spacing $s\times s$; select the lowest gradient position in the 3×3 neighborhood about center

$$s = \sqrt{\frac{N_t}{N_s}}$$



$$d_c = \sqrt{(r_j - r_i)^2 + (g_j - g_i)^2 + (b_j - b_i)^2} \xrightarrow{\text{gray-scale}} d_c = |l_j - l_i|$$

$$d_s = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}$$

$$D = \sqrt{\left(\frac{d_c}{d_{cm}}\right)^2 + \left(\frac{d_s}{d_{sm}}\right)^2} \Rightarrow \text{composite distance}$$

d_{cm} (d_{sm}): maximum expected values of d_c (d_s)

$$d_{sm} = s = \sqrt{\frac{N_t}{N_s}}$$

$$D = \sqrt{\left(\frac{d_c}{c}\right)^2 + \left(\frac{d_s}{s}\right)^2} = \sqrt{d_c^2 + c^2\left(\frac{d_s}{s}\right)^2}$$

pcLo ECE NYCU 118

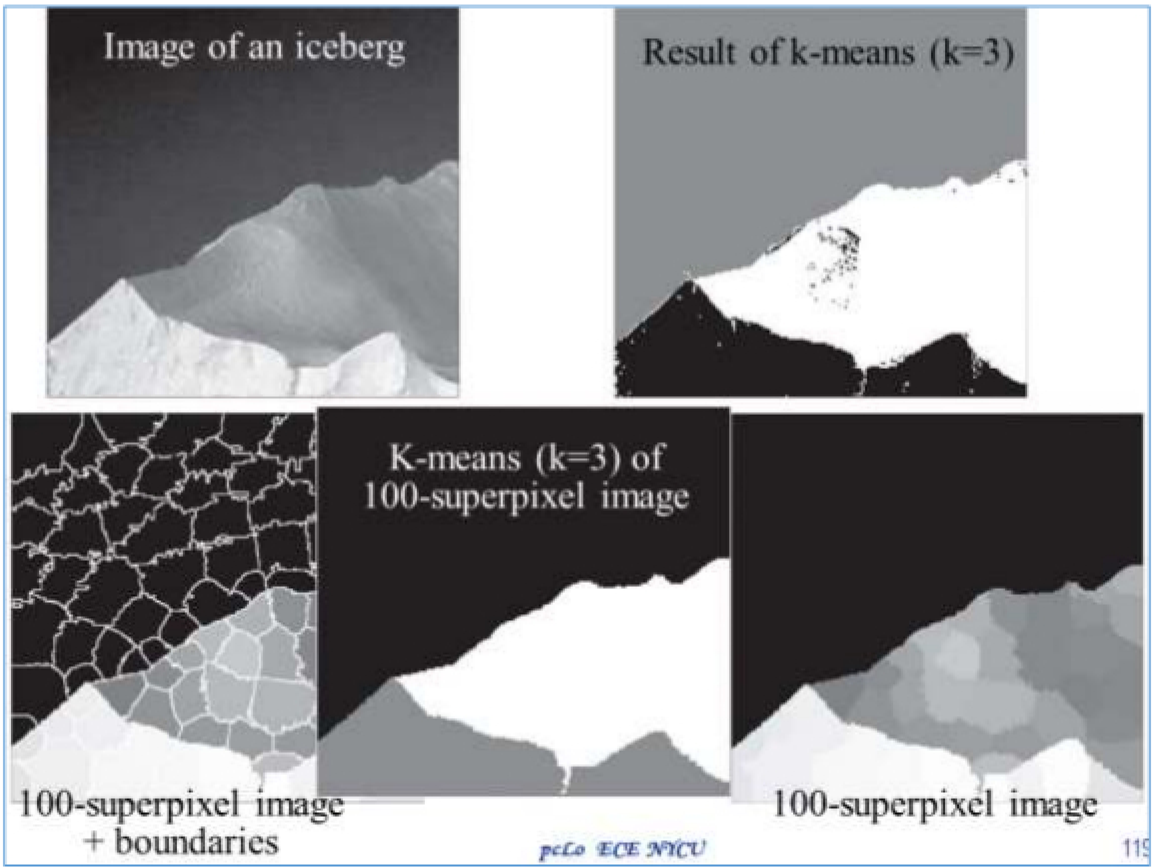


Image of an iceberg
Result of k-means (k=3)
K-means (k=3) of
100-superpixel image
100-superpixel image
+ boundaries
100-superpixel image

The End