

Image Processing (Pengolahan Citra)

Semester Genap Tahun 2019-2020

Jam 08:00 s.d. 10:30

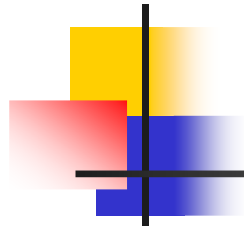
Pengajar: Mohammad Agung Wibowo, M.Kom.

STT Nurul Fikri

Slides by: Prof. Dr. Anianti Murni Arymurthy (FASILKOM UI)

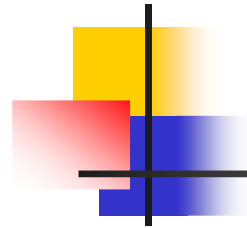
Image Enhancement

Spatial Domain



Tujuan Peningkatan Mutu Citra

- Sumber Pustaka: Gonzalez and Woods, Bab 3
- Tujuan dari teknik peningkatan mutu/kualitas citra (*image enhancement*) adalah untuk melakukan pemrosesan terhadap citra agar hasil interpretasinya mempunyai kualitas relatif lebih *baik* dari citra awal untuk aplikasi tertentu.
- Kata *baik* disini tergantung pada jenis aplikasi dan problem yang dihadapi.



Dua Pendekatan Image Enhancement

- Dapat dilakukan pada domain spasial, dan
- Dapat juga dilakukan pada domain frekwensi
- Dapat juga dilakukan dengan transformasi warna (akan dibahas secara khusus pada model transformasi warna)

Contoh Hasil Image Enhancement

(Source: <http://www.google.com/search?q=image+enhancement>)





Fourier Transform (akan dipelajari)

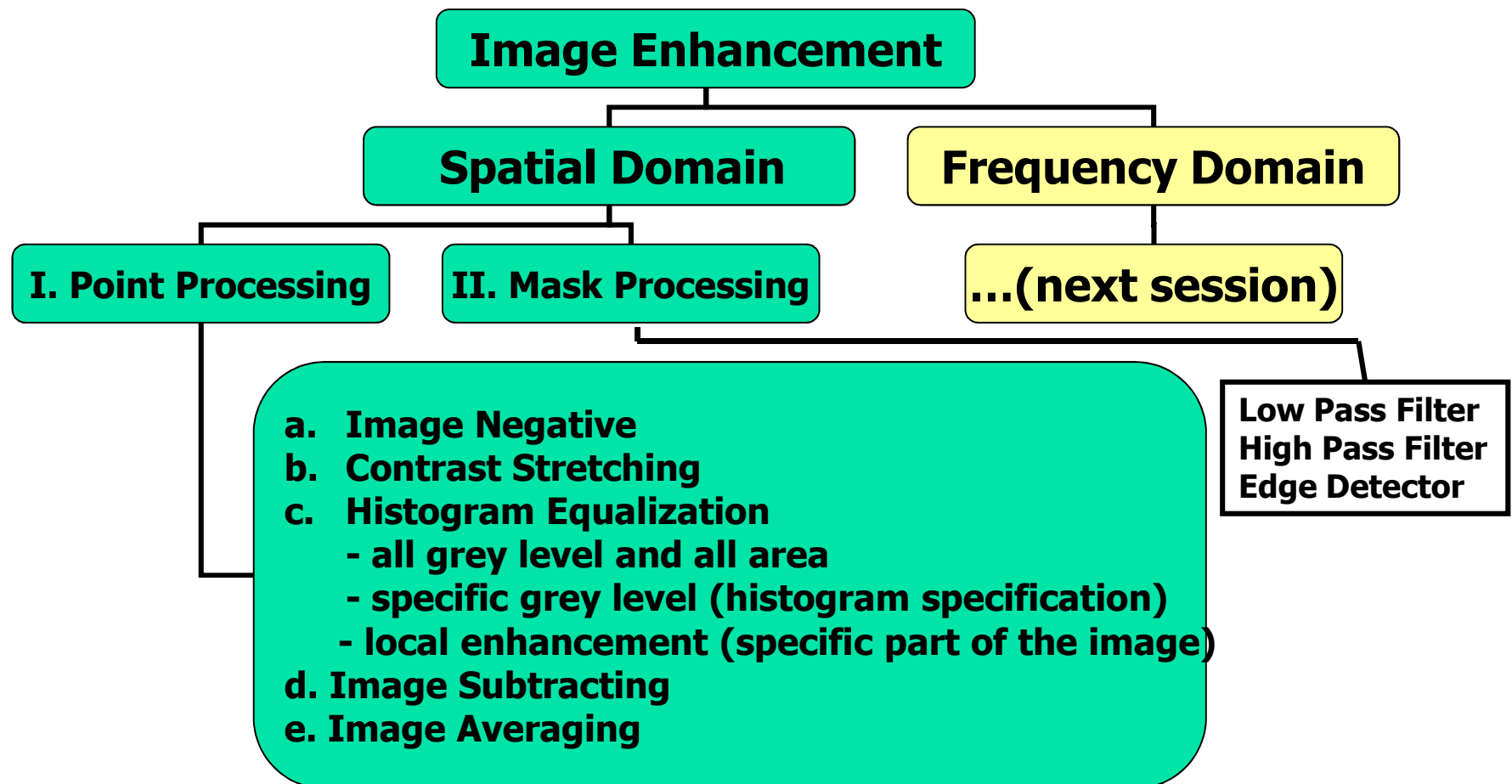
- Akan dipelajari secara khusus pada topik Image Transform
- **Mengubah representasi citra dari domain spasial ke domain frekwensi**
 - Koreksi radiometrik (filtering) dapat dilakukan pada domain spasial maupun domain frekwensi (akan dipelajari)
- Sebaliknya Inverse Fourier Transform akan mengubah representasi citra dari domain frekwensi ke domain spasial
- Memudahkan proses konvolusi dari bentuk integral menjadi bentuk perkalian biasa

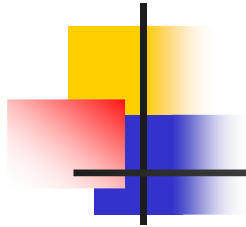
- *Konvolusi :*
$$h(x) = f(x) * g(x) = \int_{-\infty}^{\infty} f(\alpha) g(x - \alpha) d\alpha$$

 $\alpha = \text{dummy variable}$



Lingkup Pembahasan





I. Point Processing

- Cara paling mudah untuk melakukan peningkatan mutu pada domain spasial adalah dengan melakukan pemrosesan yang **hanya melibatkan satu piksel** saja (tidak menggunakan jendela ketetanggaan)
- Pengolahan biasanya menggunakan histogram, juga termasuk dalam bagian point processing



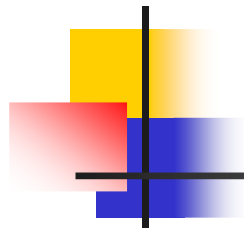
Image Negative

- Mengubah nilai grey-level piksel citra input dengan:

$$G_{\text{baru}} = 255 - G_{\text{lama}}$$

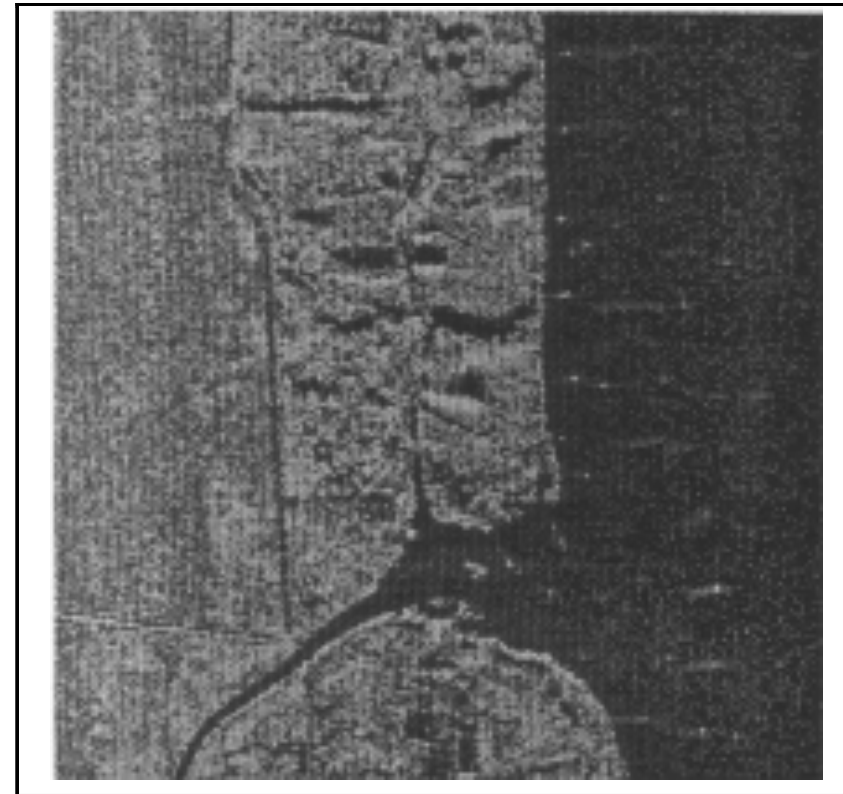
- Hasilnya seperti klise foto





Aplikasi Penginderaan Jarak Jauh

(Sumber: Murni, 1997)



Citra Optik sering dianggap sebagai negatif dari citra SAR
(Source: Bakosurtanal RI)

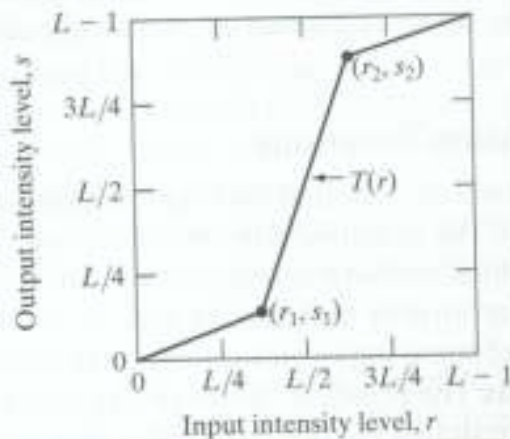
Contrast Stretching

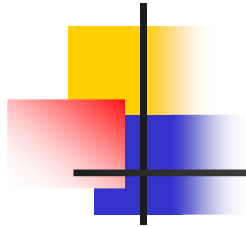
a b
c d

FIGURE 3.10

Contrast stretching.
(a) Form of
transformation
function. (b) A
low-contrast image.
(c) Result of
contrast stretching.
(d) Result of
thresholding.

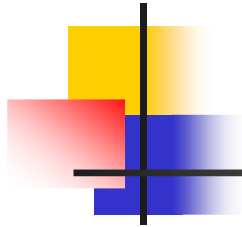
(Original image
courtesy of Dr.
Roger Heady,
Research School of
Biological Sciences,
Australian National
University,
Canberra,
Australia.)





Contrast Stretching

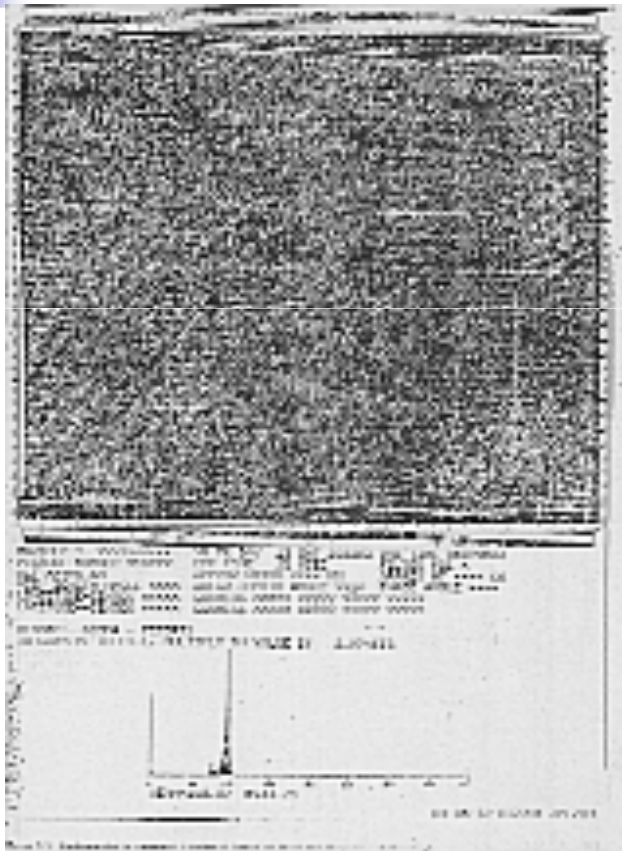




Contrast Stretching

- Fungsi lain yang baik digunakan adalah:
$$f_{\text{out}} = (f_{\text{in}} - a) * b$$
- $a = \min(f_{\text{in}})$
- $b = 255 / (\max(f_{\text{in}}) - \min(f_{\text{in}}))$
- Citra masukan yang grey level nya tidak penuh dari 0 – 255 (low contrast) diubah menjadi citra yang grey level nya berkisar dari 0 – 255 (high contrast)

Contrast Stretching



Input/Degraded Image



Output/Enhanced Image

(JPL, 1972)

Bit-Plane Slicing

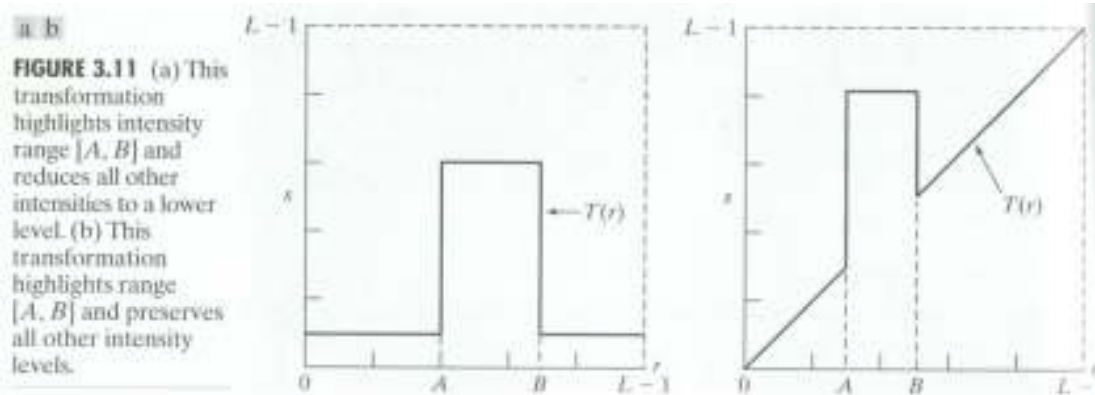
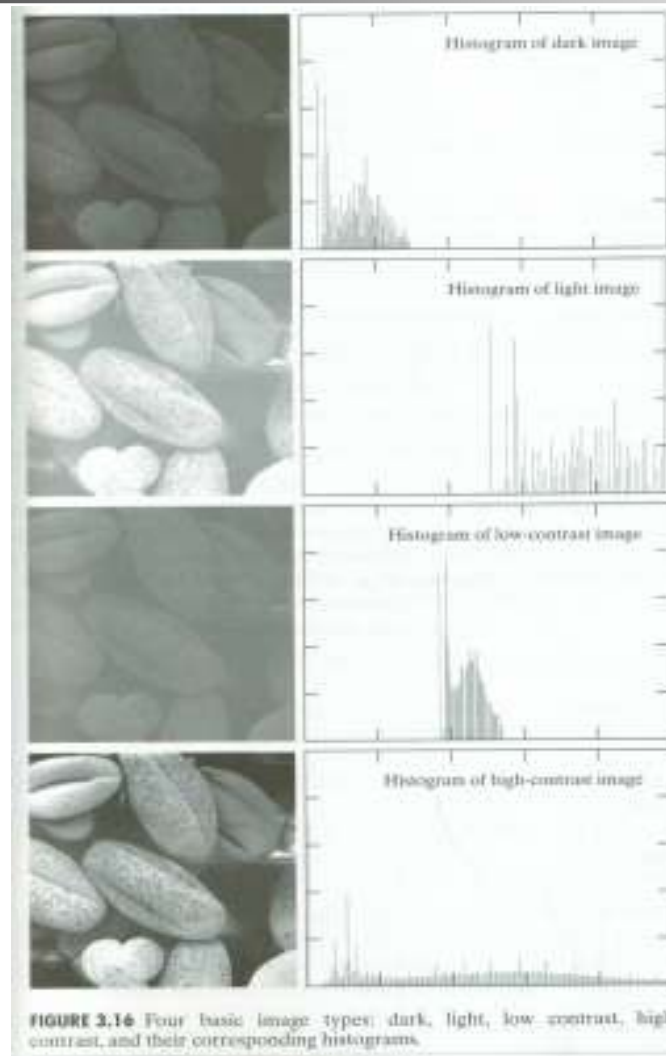


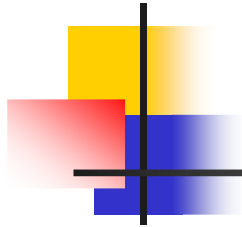
FIGURE 3.12 (a) Aortic angiogram. (b) Result of using a slicing transformation of the type illustrated in Fig. 3.11(a), with the range of intensities of interest selected in the upper end of the gray scale. (c) Result of using the transformation in Fig. 3.11(b), with the selected area set to black, so that grays in the area of the blood vessels and kidneys were preserved. (Original image courtesy of Dr. Thomas R. Gest, University of Michigan Medical School.)

(Sumber: Gonzalez & Woods, 2008)

Histogram Processing



(Sumber: Gonzalez & Woods, 2008)

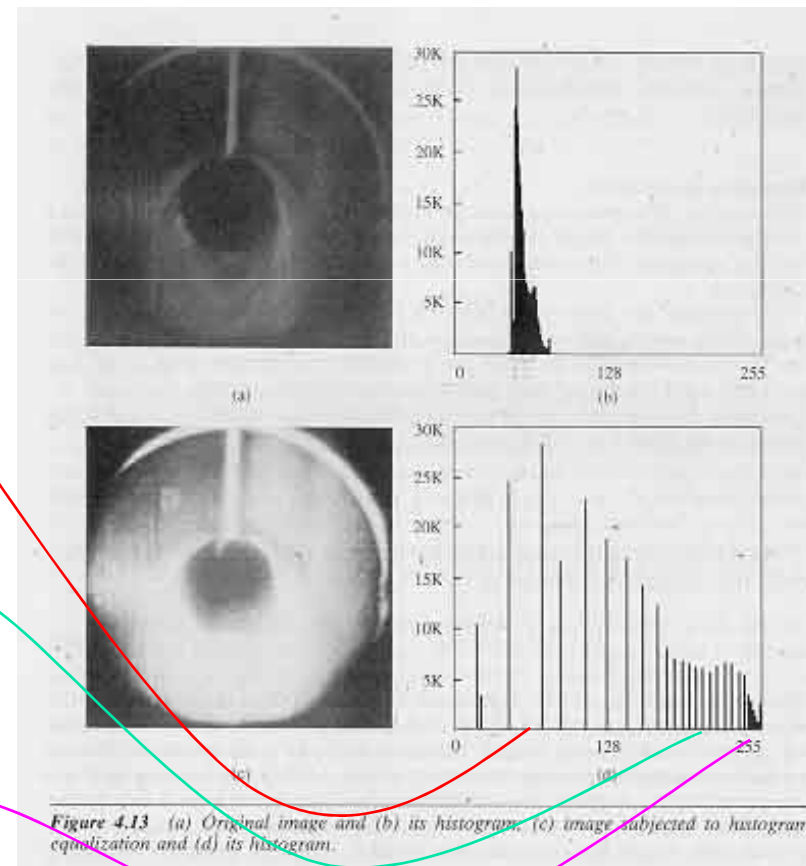


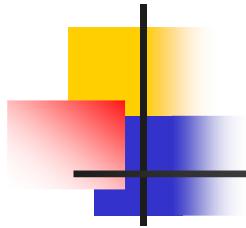
Histogram Equalization

- Histogram: diagram yang menunjukkan jumlah kemunculan grey level (0-255) pada suatu citra
- Histogram processing:
 - Gambar gelap: histogram cenderung ke sebelah kiri
 - Gambar terang: histogram cenderung ke sebelah kanan
 - Gambar low contrast: histogram mengumpul di suatu tempat
 - Gambar high contrast: histogram merata di semua tempat
- Histogram processing: mengubah bentuk histogram agar pemetaan gray level pada citra juga berubah

Histogram Equalization in all grey level and all area (1)

- Ide: mengubah pemetaan greylevel agar sebarannya (kontrasnya) lebih menyebar pada kisaran 0-255
- Sifat:
 - Grey level yang sering muncul lebih dijarangkan jaraknya dengan grey level sebelumnya
 - Grey level yang jarang muncul bisa lebih dirapatkan jaraknya dengan grey level sebelumnya
 - Histogram baru pasti mencapai nilai maksimal keabuan (contoh: 255)





Histogram Equalization in all grey level and all area (2)

- mengubah pemetaan grey level pada citra, dengan rumus (penerapan lihat next slide):

$$s_k = T(r_k) = \sum_{j=0}^k \frac{n_j}{n} = \sum_{j=0}^k p(r_j)$$

$$0 \leq r_k \leq 1 \quad \text{dan} \quad k = 0, 1, \dots, L-1$$

L adalah grey level maksimal yang ada pada citra

Histogram Equalization in all grey level and all area (3)

- Contoh : citra dengan derajat keabuan hanya berkisar 0-10, ada 20 piksel

- Citra awal:

3	5	5	5	4
5	4	5	4	4
5	3	4	4	4
4	5	6	6	3

- Citra Akhir:

1	9	9	9	5
9	5	9	5	5
9	1	5	5	5
5	9	10	10	1

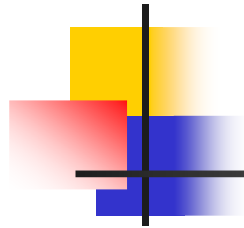
Derajat Keabuan	0	1	2	3	4	5	6	7	8	9	10
Kemunculan	0	0	0	3	8	7	2	0	0	0	0
Probabilitas Kemunculan	0	0	0	0.15	0.40	0.35	0.1	0	0	0	0
Sk	0	0	0	0.15	0.55	0.90	1	1	1	1	1
SK * 10	0	0	0	1.5	5.5	9	10	10	10	10	10
Derajat keabuan baru	0	0	0	1	5	9	10	10	10	10	10

Histogram Equalization specific area (local enhancement)

- Histogram equalization hanya dilakukan pada bagian tertentu dari citra



Citra masukan over exposed Citra keluaran lebih tajam



Operasi Aritmetik/Logic pada Citra

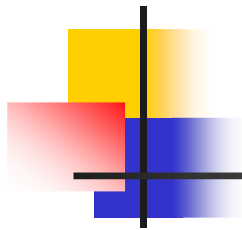
- Peningkatan mutu citra yang berikut sebenarnya termasuk pada operasi aritmetik atau logik antara dua atau lebih citra



Image Subtraction

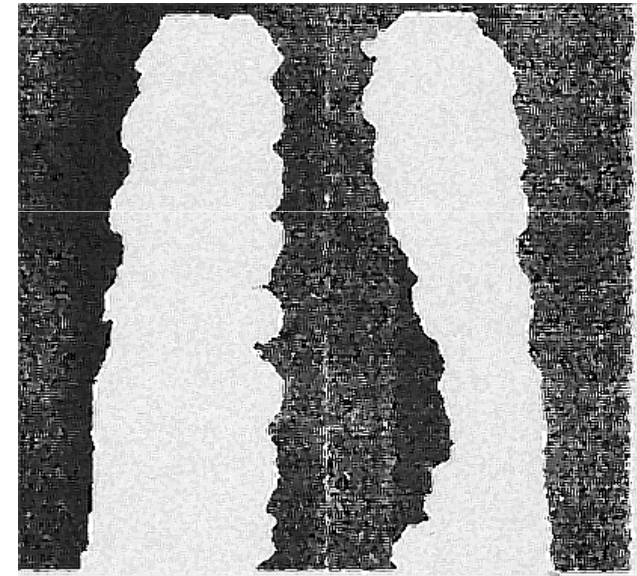
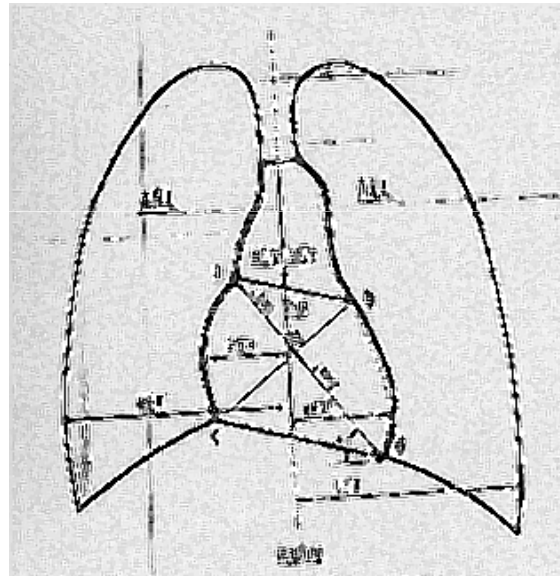
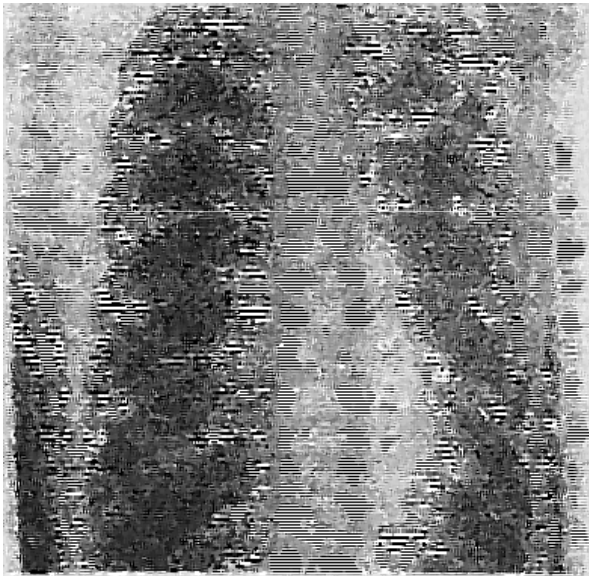
- Dilakukan jika kita ingin mengambil bagian tertentu saja dari citra





Aplikasi Kedokteran (Biomedik)

(Sumber: Thesis S2 Kartono)



(a) Thorax X-Ray

(b) Standard Landmarks

(c) Thorax Tissue

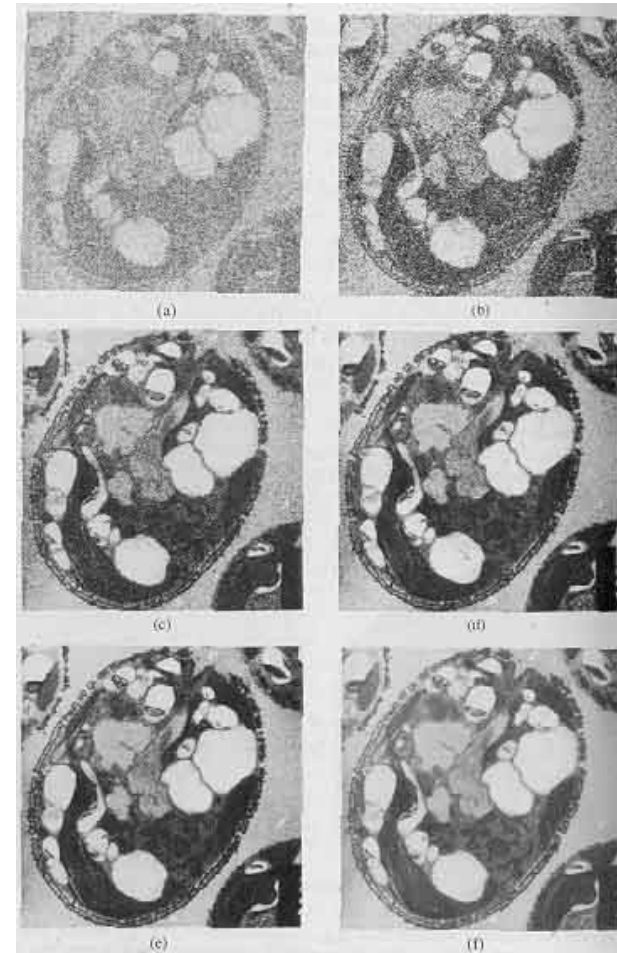
Kita bisa mengambil bagian jaringan parunya saja, dengan operasi AND citra (a) dan citra (c). Citra (c) dapat diperoleh melalui proses clustering dan thresholding

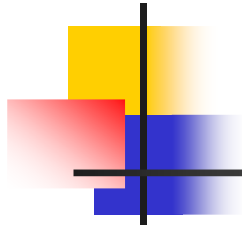
Image Averaging

Dilakukan jika kita memiliki beberapa citra yang bergambar sama, namun semua citra memiliki noise (gangguan)

Noise satu citra berbeda dengan noise citra lainnya (tidak berkorelasi)

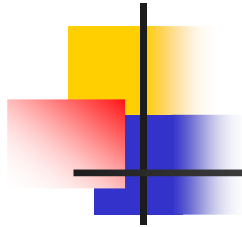
Cara memperbaikinya adalah dengan melakukan operasi rata-rata terhadap semua citra tersebut (yang masing-masing mengandung white noise)





II. *Mask Processing*

- Jika pada *point processing* kita hanya melakukan operasi terhadap masing-masing piksel, maka pada *mask processing* kita melakukan operasi terhadap suatu jendela ketetanggaan pada citra.
- Kemudian kita menerapkan (mengkonvolusikan) suatu *mask* terhadap jendela tersebut.
- *Mask* sering juga disebut *filter* atau *kernel*.



Mask Processing

1	2	3
8	x	4
7	6	5

Contoh:

Jendela ketetanggan 3x3,

Nilai piksel pada posisi **x** dipengaruhi oleh nilai 8 tetangganya

→ Perbedaan dengan *point processing* :
pada *point processing*, nilai suatu piksel tidak dipengaruhi oleh nilai tetangga-tetangganya



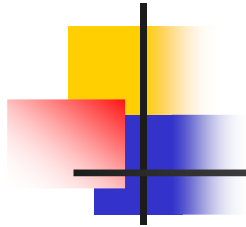
Mask Processing

W_1	W_2	W_3
W_4	W_5	W_6
W_7	W_8	W_9

**Contoh sebuah mask berukuran 3x3.
Filter ini akan diterapkan /
dikonvolusikan pada setiap jendela
ketetanggaan 3x3 pada citra (anggap
filter sudah di flip)**

G_{11}	G_{12}	G_{13}	G_{14}	G_{15}
G_{21}	G_{22}	G_{23}	G_{24}	G_{25}
G_{31}	G_{32}	G_{33}	G_{34}	G_{35}
G_{41}	G_{42}	G_{43}	G_{44}	G_{45}
G_{51}	G_{52}	G_{53}	G_{54}	G_{55}

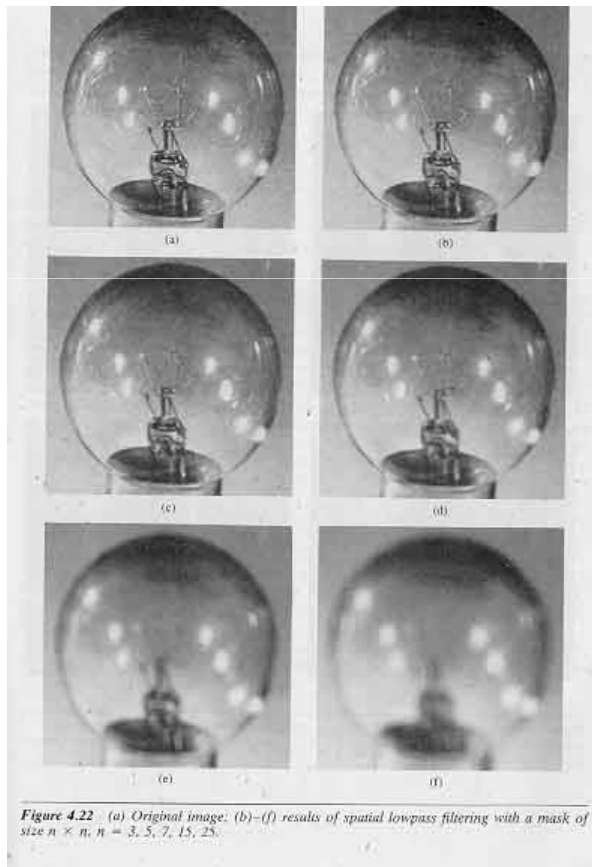
$$G_{22}' = w_1 G_{11} + w_2 G_{12} + w_3 G_{13} + \\ w_4 G_{21} + w_5 G_{22} + w_6 G_{23} + \\ w_7 G_{31} + w_8 G_{32} + w_9 G_{33}$$



Jenis-jenis filter spasial

- Smoothing filters:
 - Lowpass filter (linear filter, mengambil nilai rata-rata)
 - Median filter (non-linear filter, mengambil median dari setiap jendela ketetanggaan)
- Sharpening filters:
 - Roberts, Prewitt, Sobel (edge detection)
 - Highpass filter

Smoothing



$1/9 \times$

1	1	1
1	1	1
1	1	1

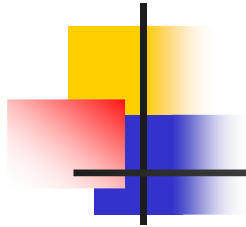
Average lowpass filter

(a) Gambar Asli
(b)-(f) hasil dari spatial lowpass
filtering dengan ukuran mask
3,5,7,15,25

Penerapan low pass filter dan filter median

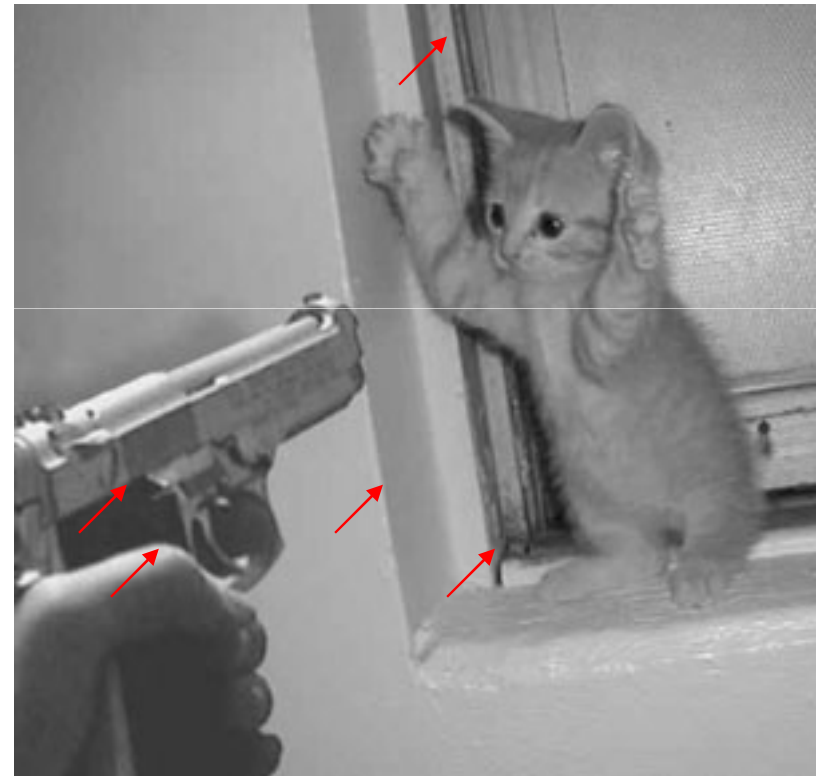
- (a) Gambar asli**
- (b) Gambar yang diberi noise**
- (c) Hasil dari 5×5 low pass average filtering**
- (d) Hasil dari 5×5 median filtering**

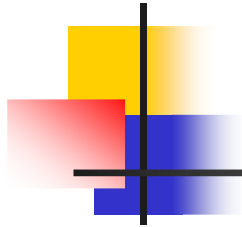




Edge detection

- Pada suatu citra monokrom, suatu edge (sisi) dapat ditandai dengan adanya *suatu perbedaan intensitas* yang besar

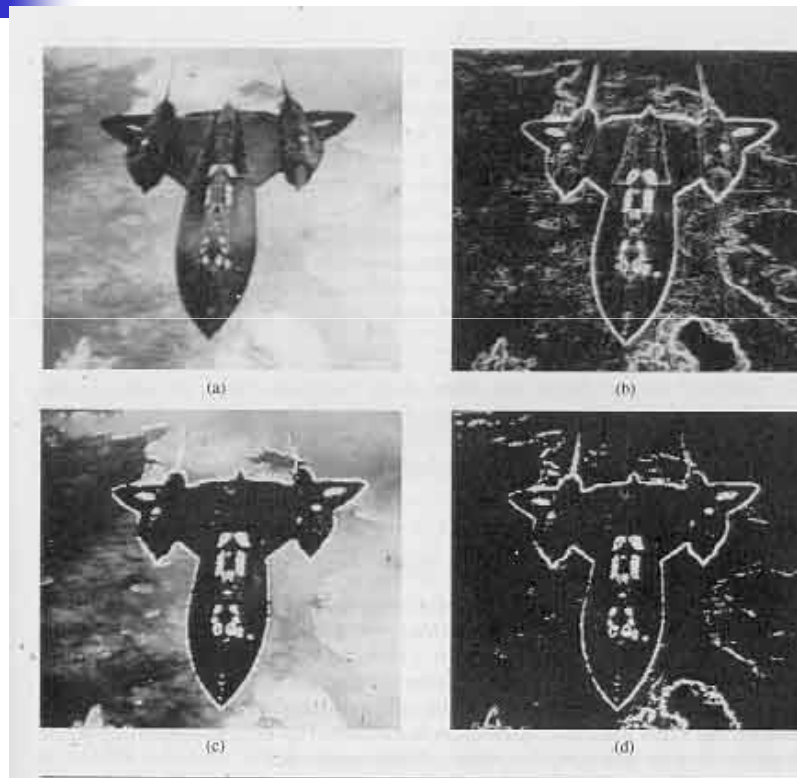




Edge detection

- Bagaimana 'mendeteksi' perbedaan intensitas tersebut?
 - Dengan mempertegas perbedaan (kalikan satu intensitas dengan nilai negatif, kemudian kalikan nilai positif pada intensitas lainnya)
 - Kasus A: 2 bersisian dgn 100 (edge) $\rightarrow 2*(-1) + 100*(1) = 99$
 - Kasus B: 2 bersisian dgn 4 (not edge) $\rightarrow 2*(-1) + 4*(1) = 2$
 - Lakukan thresholding untuk memperjelas mana bagian sisi dan mana yang bukan
 - Ambil threshold = 90, maka Kasus A akan dianggap sebagai sisi, Kasus B tidak dianggap sisi

Contoh edge detection



-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

Sobel

-1	-1	-1
0	0	0
1	1	1

-1	0	1
-1	0	1
-1	0	1

Prewitt

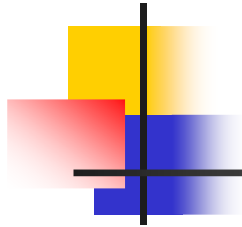
(a) Gambar awal, (b) hasil dari Prewitt mask, (c) thresholding dari (b) pada nilai > 25 (white) (d) thresholding dari (b) pada nilai > 25 (white) dan < 25 (black)



Proses Filtering dengan High Pass Filter

<i>Operator</i>	<i>Image</i>	<i>Hasil Filtering</i>
<i>1 -1 1</i>	<i>1 2 3 4 3</i>	<i>x x x x x</i>
<i>-1 8 -1</i>	<i>1 1 1 9 8</i>	<i>x 8 11 66 x</i>
<i>1 -1 1</i>	<i>1 2 1 9 9</i>	<i>x 16 10 69 x</i>
	<i>2 2 1 3 9</i>	<i>x 21 13 26 x</i>
	<i>1 2 9 7 9</i>	<i>x x x x x</i>

x = undefined



Gradient

- *Brightness gradient of image $f(x,y)$:*

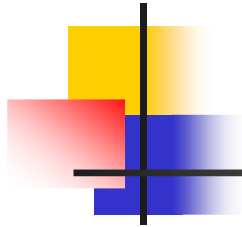
$$\Delta f = \left(\frac{\partial f(x, y)}{\partial x}, \frac{\partial f(x, y)}{\partial y} \right)$$

- *Digital derivative:*

$$\Delta x = f(x + n, y) - f(x, y)$$

$$\Delta y = f(x, y + n) - f(x, y)$$

umumnya $n=1$.



Magnitude of gradient vector

- *Rumus 1:*

$$\| \nabla f \| = \sqrt{\Delta x^2 + \Delta y^2}$$

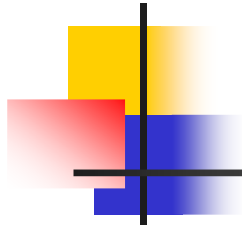
- *Rumus 2:*

$$\| \nabla f \| = \max(\text{abs}(\Delta x), \text{abs}(\Delta y))$$

- *Rumus 3:*

$$\| \nabla f \| = \text{abs}(\Delta x) + \text{abs}(\Delta y)$$

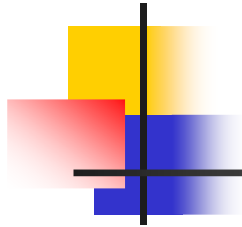
- *The quickest speed with which the intensity changes at $f(x,y)$*



Direction of gradient vector

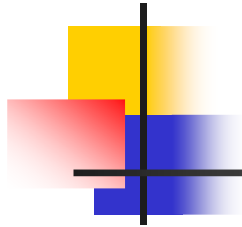
- *The direction in which the intensity changes the quickest at $f(x,y)$*
- *Direction*

$$\phi = \tan^{-1} \text{ atau } \arctan \frac{\Delta y f(i, j)}{\Delta x f(i, j)}$$

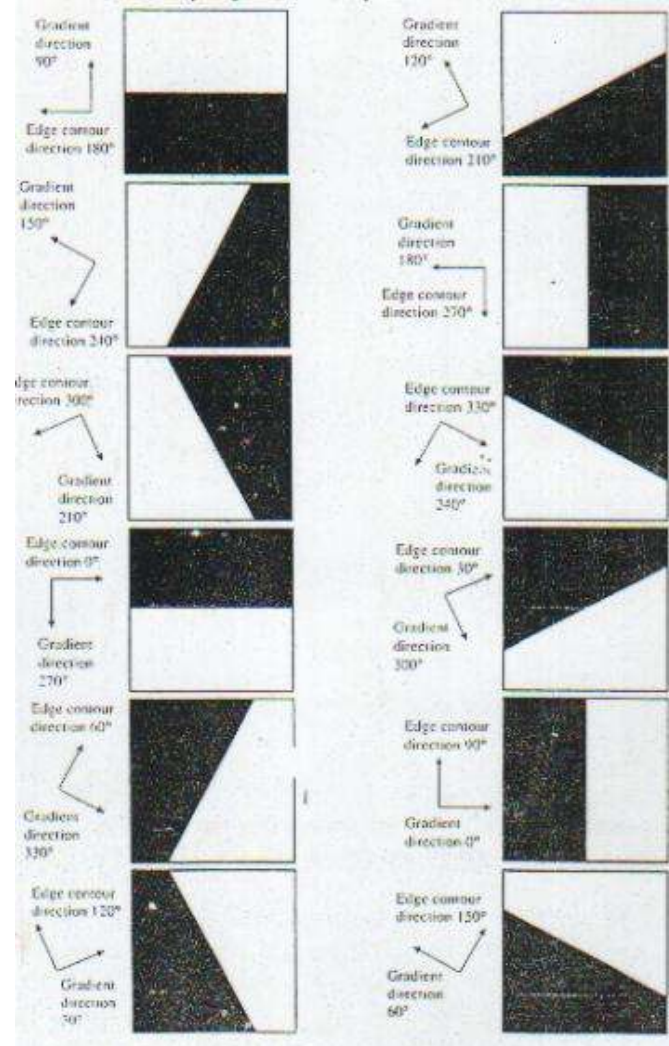


Direction of gradient vector

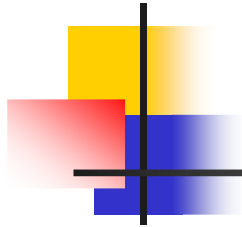
- *Edge contour direction: along the contour, right side is white (high value)*
- *Edge gradient direction: orthogonal to the contour, towards white (high value)*



Direction of gradient vector



Sumber: MSU



1st derivative and 2nd derivative

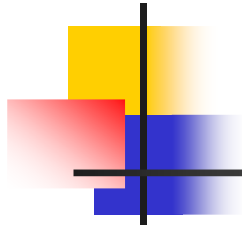
	$f(i, j-1)$	
$f(i-1, j)$	$f(i, j)$	$f(i+1, j)$
	$f(i, j+1)$	

$$\Delta_x f(i, j) = f(i, j) - f(i-1, j)$$

$$\Delta_y f(i, j) = f(i, j) - f(i, j-1)$$

$$\Delta^2_x f(i, j) = \Delta_x f(i+1, j) - \Delta_x f(i, j)$$

$$\Delta^2_y f(i, j) = \Delta_y f(i, j+1) - \Delta_y f(i, j)$$



Laplacian Operator (second derivative)

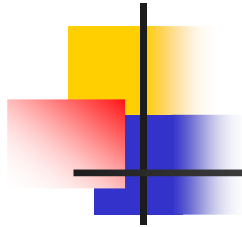
Citra Kontinue: $\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$

Citra Dijital: $\nabla^2 f(i, j) = \Delta^2 x f(i, j) + \Delta^2 y f(i, j)$

$$\begin{aligned}\Delta^2 x f(i, j) &= f(i+1, j) - f(i, j) - f(i, j) + f(i-1, j) \\ &= f(i+1, j) + f(i-1, j) - 2f(i, j)\end{aligned}$$

$$\Delta^2 y f(i, j) = f(i, j+1) + f(i, j-1) - 2f(i, j) \quad \text{Laplacian Mask}$$

$$\begin{aligned}\nabla^2 f(i, j) &= \overbrace{f(i+1, j) + f(i-1, j) + f(i, j+1) + f(i, j-1) - 4f(i, j)}^{\leftarrow} \\ &= \overbrace{a}^{\leftarrow} - 4f(i, j) = f(i, j) - 5f(i, j) + a = -5f(i, j) + b\end{aligned}$$



Laplacian Operator

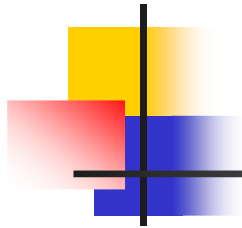
$$-1/5 * \nabla^2 f(i, j) = f(i, j) - 1/5 * b \quad b = \text{piksel tengah} + 4\text{-tetangga}$$

$$-k * \nabla^2 f(i, j) = f - f_{average}$$

$$k * \nabla^2 f(i, j) = f_{average} - f$$

$$f = g - k * \nabla^2 g(i, j) = g - (g_{average} - g) = 2g - g_{average}$$

$g_{average}$ menguatkan respon frekwensi rendah dan melemahkan respon frekwensi tinggi $\rightarrow (2g - g_{average})$ akan menguatkan respon frekwensi tinggi relatif terhadap frekwensi rendah.

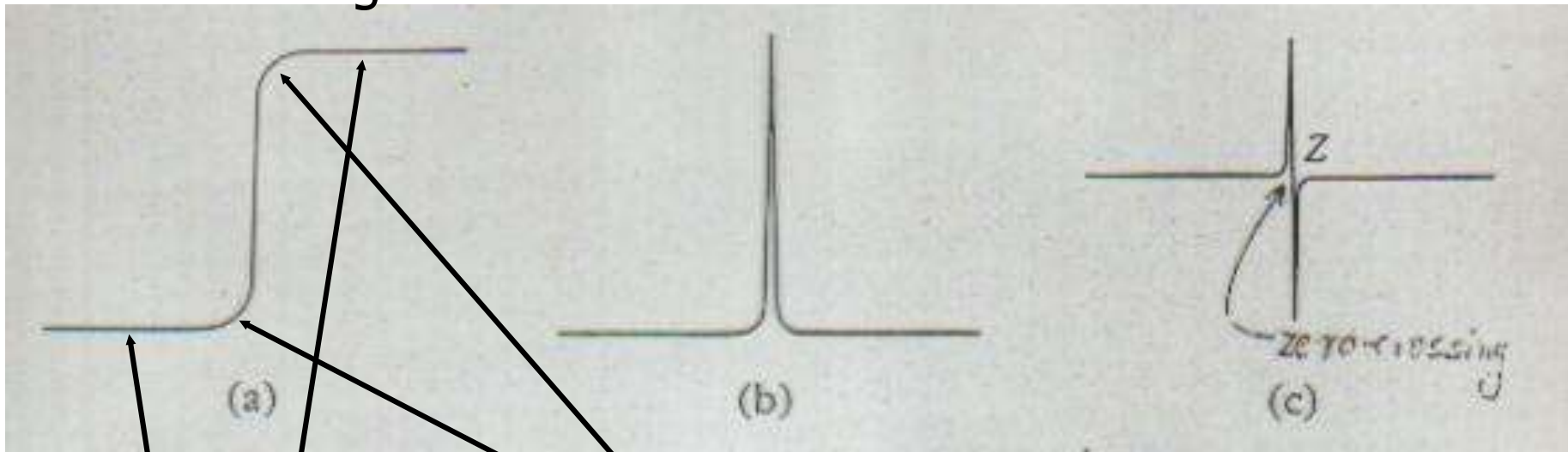


Konsep Zero-Crossing

1-D image

1st derivative

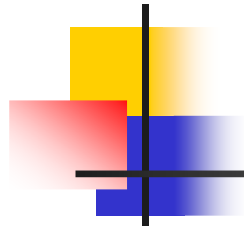
2nd derivative



Frekwensi rendah dan frekwensi tinggi.

(a) Perubahan intensitas; (b) Mempunyai *peak*; (c) *Steep zero-crossing*.

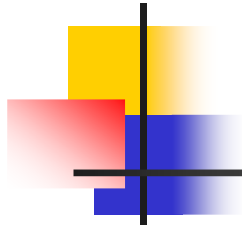
Sumber: MSU



Contoh Kernel Edge Detector

Gerald K. Moore: directional edge detection

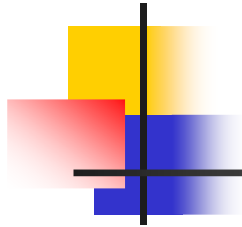
E-W			N-S			NW-SE			NE-SW		
-1	-2	-1	-1	2	-1	2	-1	-2	-2	-1	2
2	4	2	-2	4	-2	-1	4	-1	-1	4	-1
-1	-2	-1	-1	2	-1	-2	-1	2	2	-1	-2
WNW-ESE			NNW-SSE			ENE-WSW			NNE-SSW		
1	-2	-2	1	1	-2	-2	-2	1	-2	1	1
1	4	1	-2	4	-2	1	4	1	-2	4	-2
-2	-2	1	-2	1	1	1	-2	-2	1	1	-2



Contoh Kernel Edge Detector

Baxter: directional edge detection

Utara			U-T			Timur			S-T		
1	1	1	1	1	1	-1	1	1	-1	-1	1
1	-2	1	-1	-2	1	-1	-2	1	-1	-2	1
-1	-1	-1	-1	-1	1	-1	1	1	1	1	1
Selatan			S-B			Barat			U-B		
-1	-1	-1	1	-1	-1	1	1	-1	1	1	1
1	-2	1	1	-2	-1	1	-2	-1	1	-2	-1
1	1	1	1	1	1	1	1	-1	1	-1	-1



Contoh Kernel Edge Detector

Robert (1962):

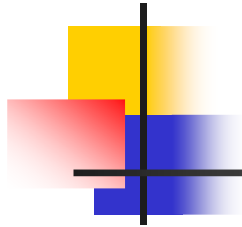
-1	0	0	-1
0	1	1	0

Prewitt (1970):

-1	-1	-1	-1	0	1
0	0	0	-1	0	1
1	1	1	-1	0	1

Sobel (1970):

-1	-2	-1	-1	0	1
0	0	0	-2	0	2
1	2	1	-1	0	1



Contoh Kernel Edge Detector

Kirsh (1977):

-1	1
----	---

1
-1

	1
-1	

1	
	-1

-1	0	1
-1	0	1
-1	0	1

1	1	1
0	0	0
-1	-1	-1

0	1	1
-1	0	1
-1	-1	0

1	1	0
1	0	-1
0	-1	-1

-1	-1	0	1	1
-1	-1	0	1	1
-1	-1	0	1	1
-1	-1	0	1	1
-1	-1	0	1	1

1	1	1	1	1
1	1	1	1	1
0	0	0	0	0
-1	-1	-1	-1	-1
-1	-1	-1	-1	-1

0	1	1	1	1
-1	0	1	1	1
-1	-1	0	1	1
-1	-1	-1	0	1
-1	-1	-1	-1	0

1	1	1	1	0
1	1	1	0	-1
1	1	0	-1	-1
1	0	-1	-1	-1
0	-1	-1	-1	-1



Edge Detection Turunan Pertama

$$f(x) = [0 \ 0 \ 1 \ 2 \ 3 \ 4 \ 0]$$

$$g(x) = [1 \ -1]$$

$$\rightarrow [\ 0 \ 0 \ 1 \ 2 \ 3 \ 4 \ 0 \ 0 \ 0]$$

$$\text{di-flip } [-1 \ 1]$$

$$\rightarrow [-1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]$$

maka $f(x) * g(x) =$

$$0x-1 + 0x1 + 1x0 + 2x0 + 3x0 + 4x0 + 0x0 + 0x0 + 0x0 = 0$$

$$0x0 + 0x-1 + 1x1 + 2x0 + 3x0 + 4x0 + 0x0 + 0x0 + 0x0 = 1$$

$$0x0 + 0x0 + 1x-1 + 2x1 + 3x0 + 4x0 + 0x0 + 0x0 + 0x0 = 1$$

$$0x0 + 0x0 + 1x0 + 2x-1 + 3x1 + 4x0 + 0x0 + 0x0 + 0x0 = 1$$

$$0x0 + 0x0 + 1x0 + 2x0 + 3x-1 + 4x1 + 0x0 + 0x0 + 0x0 = 1$$

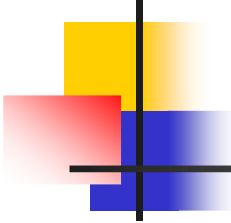
$$0x0 + 0x0 + 1x0 + 2x0 + 3x0 + 4x-1 + 0x1 + 0x0 + 0x0 = -4$$

$$0x0 + 0x0 + 1x0 + 2x0 + 3x0 + 4x0 + 0x-1 + 0x1 + 0x0 = 0$$

$$0x0 + 0x0 + 1x0 + 2x0 + 3x0 + 4x0 + 0x0 + 0x-1 + 0x1 = 0$$

$$f(x) * g(x) = [0 \ 1 \ 1 \ 1 \ 1 \ -4 \ 0 \ 0]$$

Operator Robert melakukan outlining (informasi asli hilang)



Edge Detection Turunan Kedua

$$f(x) = [0 \ 0 \ 1 \ 2 \ 3 \ 4 \ 0] \quad \rightarrow [\ 0 \ 0 \ 1 \ 2 \ 3 \ 4 \ 0 \ 0 \ 0]$$

$$g(x) = [-1 \ 4 \ -1] \text{ karena simetri di-flip tetap } [-1 \ 4 \ -1]$$

$$\rightarrow [-1 \ 4 \ -1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]$$

$$\text{maka } f(x) * g(x) =$$

$$0x-1 + 0x4 + 1x-1 + 2x0 + 3x0 + 4x0 + 0x0 + 0x0 + 0x0 = -1$$

$$0x0 + 0x-1 + 1x4 + 2x-1 + 3x0 + 4x0 + 0x0 + 0x0 + 0x0 = 2$$

$$0x0 + 0x0 + 1x-1 + 2x4 + 3x-1 + 4x0 + 0x0 + 0x0 + 0x0 = 4$$

$$0x0 + 0x0 + 1x0 + 2x-1 + 3x4 + 4x-1 + 0x0 + 0x0 + 0x0 = 6$$

$$0x0 + 0x0 + 1x0 + 2x0 + 3x-1 + 4x4 + 0x-1 + 0x0 + 0x0 = 13$$

$$0x0 + 0x0 + 1x0 + 2x0 + 3x0 + 4x-1 + 0x0 + 0x0 + 0x0 = -4$$

$$0x0 + 0x0 + 1x0 + 2x0 + 3x0 + 4x0 + 0x-1 + 0x4 + 0x-1 = 0$$

$$0x0 + 0x0 + 1x0 + 2x0 + 3x0 + 4x0 + 0x0 + 0x-1 + 0x4 = 0$$

$$0x0 + 0x0 + 1x0 + 2x0 + 3x0 + 4x0 + 0x0 + 0x0 + 0x-1 = 0$$

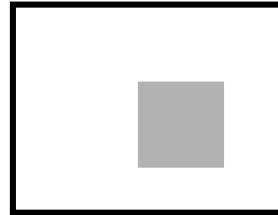
$$f(x) * g(x) = [-1 \ 2 \ 4 \ 6 \ 13 \ -4 \ 0 \ 0 \ 0]$$

Operator Laplace mempertahankan informasi aslinya

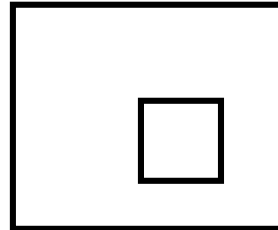


1st derivative and 2nd derivative

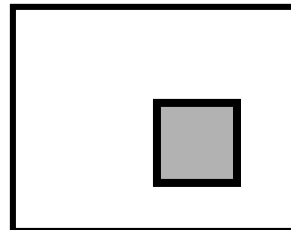
- *Contoh image:*

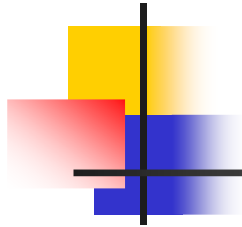


- *Hasil 1st derivative (outlining):*



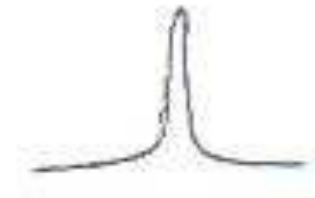
- *Hasil 2nd derivative (retaining original image):*





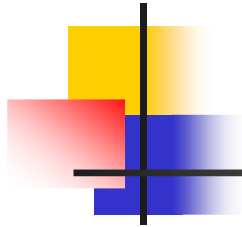
Laplacian of Gaussian Filtering

- *Gaussian operator (LPF):*
Gaussian blurring adalah $G_{\sigma}(x, y) * F(x, y)$
Smoothing untuk menghilangkan noise,
dengan nilai σ yang besar *atau σ yang kecil*



1-D: $G_{\sigma}(x) = \sigma^2 e^{-(x^2)/(2\sigma^2)}$

2-D: $G_{\sigma}(x, y) = \sigma^2 e^{-(x^2 + y^2)/(2\sigma^2)}$

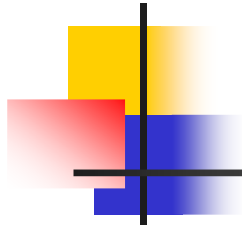


Laplacian of Gaussian Filtering

- *Laplacian operator (HPF):*

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

- *Laplacian bertujuan untuk meningkatkan kualitas detil (detail enhancement)*
- *Laplacian of Gaussian filtering bertujuan untuk menghilangkan noise dan meningkatkan kualitas detil.*



Laplacian of Gaussian Filtering

- *Laplacian of Gaussian:* $\nabla^2 G_\sigma(x, y, \sigma) = \nabla^2 G_\sigma * F(x, y)$

$$\nabla^2 G_\sigma(x, y, \sigma) = \left(\frac{r^2 - 2\sigma^2}{\sigma^4} \right) \exp\left(\frac{-r^2}{2\sigma^2}\right)$$

dengan $r = \sqrt{x^2 + y^2}$

- *Selanjutnya dicari lokasi zero-crossing untuk menentukan garis batas antara hitam dan putih.*



Laplacian of Gaussian Filtering



(a)

(b)

(c)

(a) Original image (320 x 320 pixels)

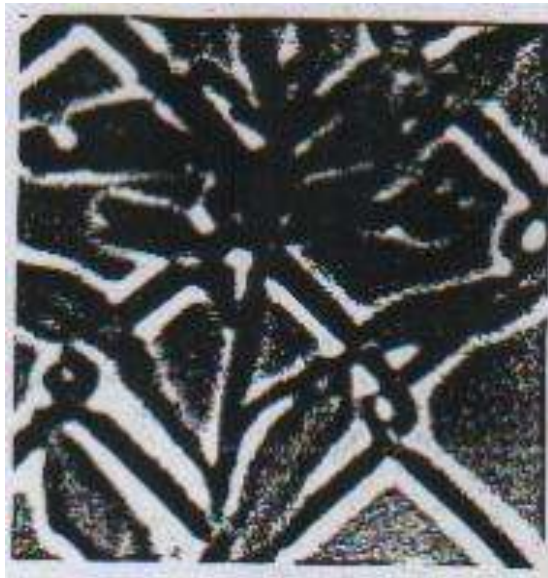
(b) Gaussian filtering dengan $\sigma = 8$ piksel

(Sumber: MSU)

(c) Gaussian filtering dengan $\sigma = 4$ piksel



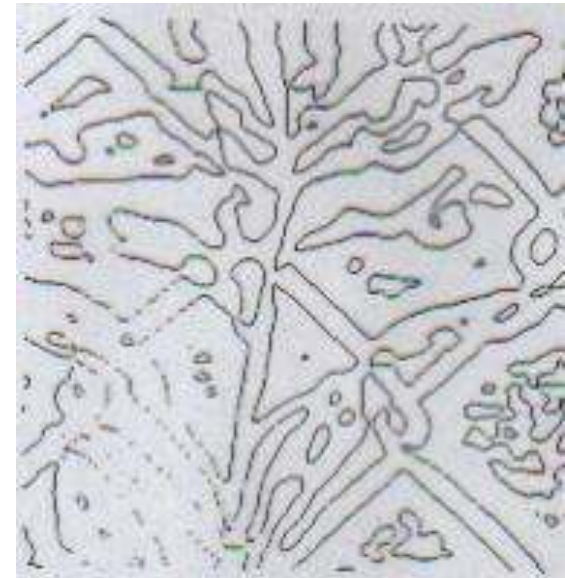
Laplacian of Gaussian Filtering



(a)



(b)



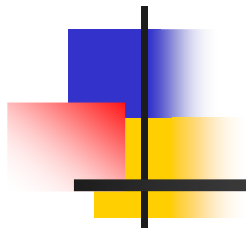
©

(a) Laplacian of Gaussian

(b) Positive = putih dan negative = hitam

(c) zero-crossings

(Sumber: MSU)



The End of Presentation