

A series of thin, black, overlapping geometric lines forming various polygons and triangles, located in the top-left quadrant of the slide.

# **OPERASI MORFOLOGI**

Pengolahan Citra

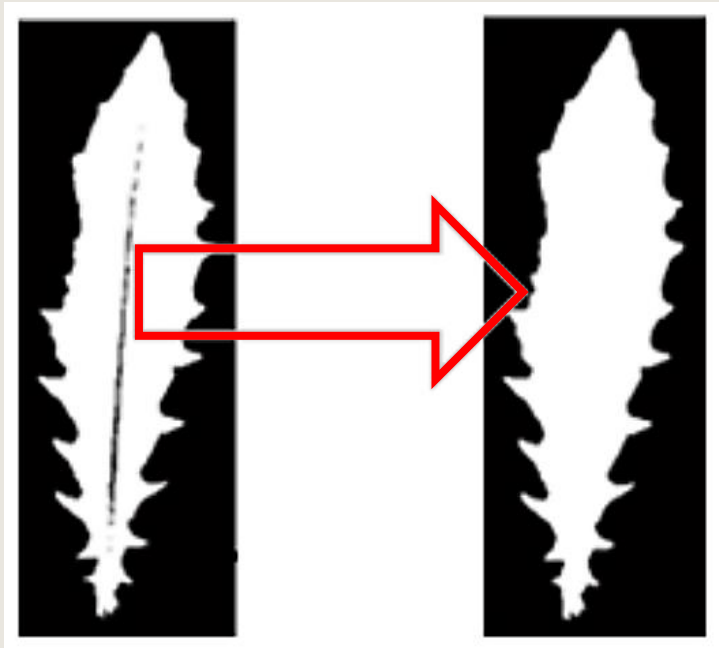
**PERTEMUAN KE 11 dan 12**

# PENGERTIAN OPERASI MORFOLOGI

Operasi morfologi merupakan operasi yang umum dikenakan pada citra biner (hitam – putih) untuk **mengubah struktur bentuk obyek yang terkandung dalam citra.**



# APLIKASI OPERASI MORFOLOGI



Gambar 10.1 Melalui morfologi tulang daun dianggap sebagai bagian dari daun



(a) (b)  
Gambar 10.2 Melalui morfologi daun-daun yang bersinggungan (a), dapat dipisahkan (b)

# BEBERAPA CONTOH APLIKASI MORFOLOGI



- Membentuk filter Spasial
- Memperoleh skeleton (rangka) objek
- Menentukan letak obyek di dalam citra
- Memperoleh bentuk struktur obyek

# OPERASI MORFOLOGI

Inti dari operasi morfologi melibatkan dua larik piksel, larik pertama berupa citra yang akan dikenai operasi morfologi, sedangkan larik kedua dinamakan kernel atau *structuring element* (elemen penstruktur) (Shih, 2009). Contoh kernel ditunjukkan di Gambar 10.3

0	1	0
1	1	1
0	1	0

1	1	1
1	1	1
1	1	1

1	1
1	1

1
1
1
1
1

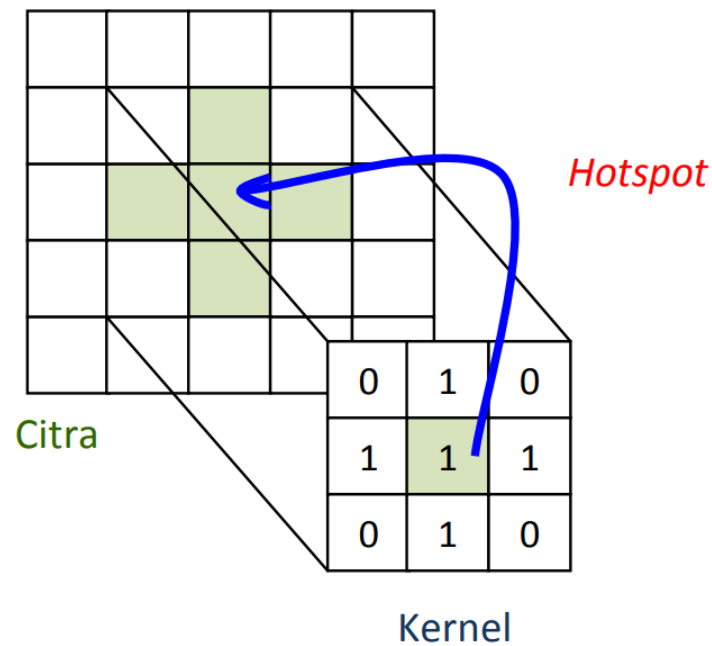
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |

Gambar 10.3 Contoh beberapa kernel



## OPERASI MORFOLOGI

Pada contoh kernel yang ditunjukkan Gambar 10.3, piksel pusat (biasa diberi nama *hotspot*) sebagaimana diilustrasikan Gambar 10.4, ditandai dengan warna hijau. Piksel pusat ini yang menjadi pusat dalam melakukan operasi terhadap citra.



- Dua operasi yang mendasari morfologi yaitu dilasi dan erosi. Dua operasi lain yang sangat berguna dalam pemrosesan citra adalah opening dan closing, yang dibentuk melalui operasi dasar itu.

Gambar 10.4 Operasi kernel terhadap citra

# Untuk Memahami Operasi Morfologi

Dalam memahami operasi morfologi, tentu perlu mengetahui tentang operasi himpunan, seperti interseksi dan gabungan. Selain itu pemahaman terhadap logika seperti “atau”, “dan” juga diperlukan.



# TEORI HIMPUNAN

$$a \in A \text{ ..... (10.1)}$$

Arti notasi di atas, bahwa **a** adalah anggota himpunan **A**. kebalikannya, jika **a** bukan anggota himpunan **A**, **a** ditulis seperti berikut:

$$a \notin A \text{ ..... (10.2)}$$

Sebagai contoh,  $s = (1, 2)$  dan  $t = (1, 4)$ , sedangkan himpunan  $A$  berisi seperti berikut:  $A = \{ (1,1), (1,2), (1, 3), (2, 1), (2, 2) \}$  Pada contoh tersebut,  $A$  memiliki 5 anggota. Berdasarkan contoh tersebut, dapat dituliskan fakta

berikut:

$$s \in A$$

$$t \notin A$$





# TEORI HIMPUNAN

$$a \in A \text{ ..... (10.1)}$$

Arti notasi di atas, bahwa **a** adalah anggota himpunan **A**. kebalikannya, jika **a** bukan anggota himpunan **A**, **a** ditulis seperti berikut:

$$a \notin A \text{ ..... (10.2)}$$

Sebagai contoh,  $s = (1, 2)$  dan  $t = (1, 4)$ , sedangkan himpunan **A** berisi seperti berikut:  $A = \{ (1, 1), (1, 2), (1, 3), (2, 1), (2, 2) \}$ . Pada contoh tersebut, **A** memiliki 5 anggota. Berdasarkan contoh tersebut, dapat

dituliskan fakta berikut:

$$s \in A$$

$$t \notin A$$



# TEORI HIMPUNAN

Perlu diketahui, **setiap elemen hanya dapat menjadi anggota himpunan satu kali**. Dengan demikian,

$$A = \{(1,1), (1,1), (2,1), (2,3), (2,1)\}$$

Sesungguhnya hanya mempunyai 3 anggota, yaitu

$$A = \{(1,1), (2,1), (2,3)\}$$



	1	2	3	4	5		1	2	3	4	5
1	0	1	0	0	0		1	1	0	0	0
2	0	1	1	0	0		0	1	0	0	0
3	0	1	1	1	0		0	0	1	0	0
4	0	1	1	0	0		0	1	0	0	0
5	0	1	0	0	0		1	0	0	0	0

$A = \{(1,2), (2,2), (2,3), (3,2), (3,3), (3,4), (4,2), (4,3), (5,2)\}$

$B = \{(1,1), (2,1), (2,2), (3,3), (4,2), (5,1)\}$

$C = A \cup B$

1	1	0	0	0
0	1	1	0	0
0	1	1	1	0
0	1	1	0	0
1	1	0	0	0

$C = \{(1,1), (1,2), (2,2), (2,3), (3,2), (3,3), (3,4), (4,2), (4,3), (5,1), (5,2)\}$

Gambar 10.5 Operasi Union pada citra biner



	1	2	3	4	5
1	0	1	0	0	0
2	0	1	1	0	0
3	0	1	1	1	0
4	0	1	1	0	0
5	0	1	0	0	0

	1	2	3	4	5
1	1	1	0	0	0
2	0	1	0	0	0
3	0	0	1	0	0
4	0	1	0	0	0
5	1	0	0	0	0

$A = \{(1,2), (2,2), (2,3), (3,2), (3,3), (3,4), (4,2), (4,3), (5,2)\}$

$B = \{(1,1), (2,1), (2,2), (3,3), (4,2), (5,1)\}$



$C = A \cap B$

1	1	0	0	0
0	1	1	0	0
0	1	1	1	0
0	1	1	0	0
1	1	0	0	0

$C = \{(1,2), (2,2), (3,3), (4,1)\}$



Gambar 10.6 Operasi Interseksi pada citra biner

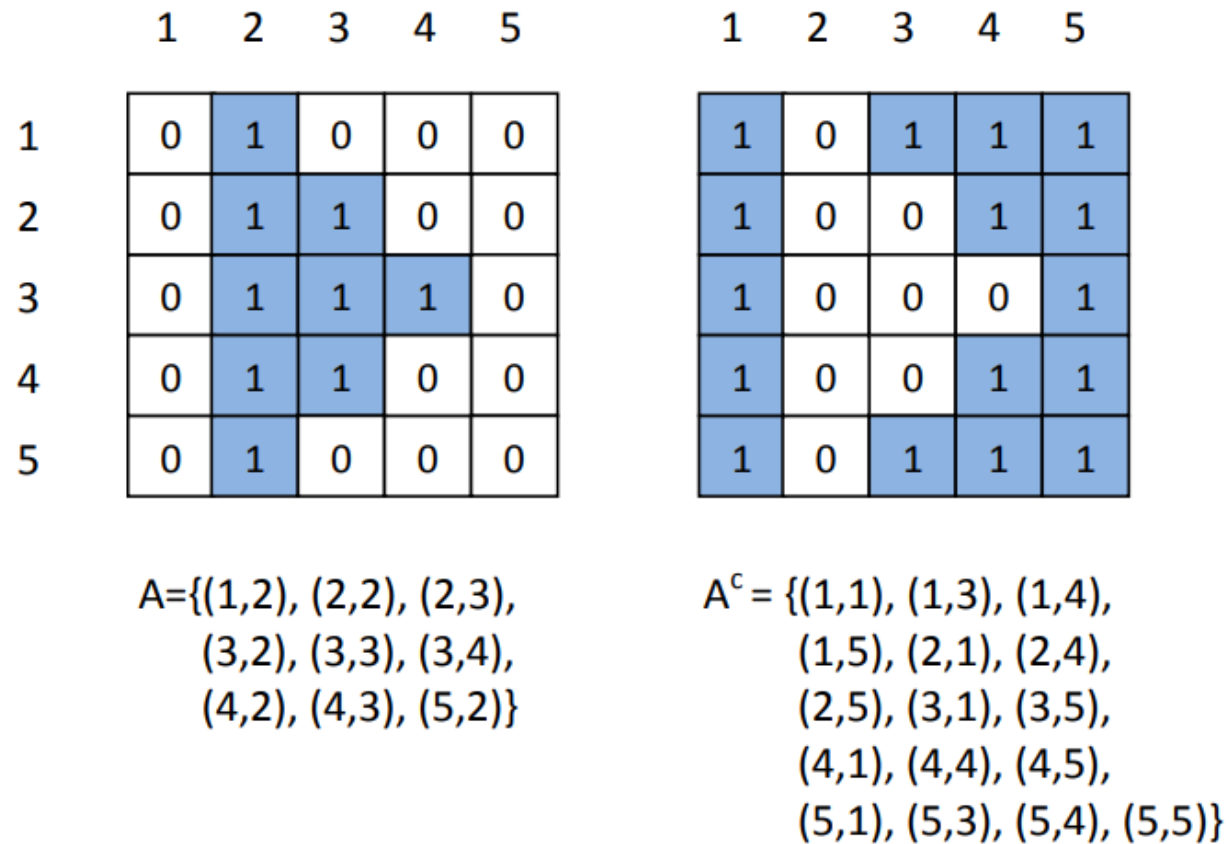
# TEORI HIMPUNAN

**Komplemen** himpunan  $A$  biasa dinotasikan dengan  $A^c$  dan menyatakan semua elemen yang tidak terdapat pada  $A$ . Secara matematis, komplemen ditulis seperti berikut:

$$A^c = \{ w \mid w \notin A \} \dots\dots\dots (10.3)$$

- Notasi di atas dibaca “semua elemen yang tidak menjadi anggota  $A$ ”.





Gambar 10.7 Operasi komplemen pada citra biner

Komplemen atau juga disebut inversi dapat dibayangkan seperti saling menukarkan warna hitam dan putih. Nilai yang semula berupa nol diganti satu dan nilai satu diganti dengan nol.

Contoh dapat dilihat di Gambar 10.7. Di bidang fotografi dengan film, inversi menghasilkan gambar negatif.



# TEORI HIMPUNAN

**Operasi selisih** dua himpunan dapat ditulis seperti berikut:

$$A - B = \{ w \mid w \in A, w \notin B \} = A \cap B^c \dots\dots\dots (10.4)$$

Contoh ditunjukkan di Gambar 7.8.



	1	2	3	4	5
1	0	1	0	0	0
2	0	1	1	0	0
3	0	1	1	1	0
4	0	1	1	0	0
5	0	1	0	0	0

	1	2	3	4	5
1	1	1	0	0	0
2	0	1	0	0	0
3	0	0	1	0	0
4	0	1	0	0	0
5	1	0	0	0	0

A={{(1,2), (2,2), (2,3),  
(3,2), (3,3), (3,4),  
(4,2), (4,3), (5,2)}}

B={{(1,1), (2,1), (2,2),  
(3,3), (4,2), (5,1)}}



C = A - B

C = B - A

0	0	0	0	0
0	0	1	0	0
0	0	0	1	0
0	0	1	0	0
0	1	0	0	0

1	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
1	0	0	0	0

C = { (2,3), (3,4),  
(4,3), (5,2) }

C = { (1,1), (5,1) }

Contoh di samping  
menunjukkan bahwa  
 $A - B \neq B - A$ .



Gambar 10.8 Contoh selisih dua himpunan



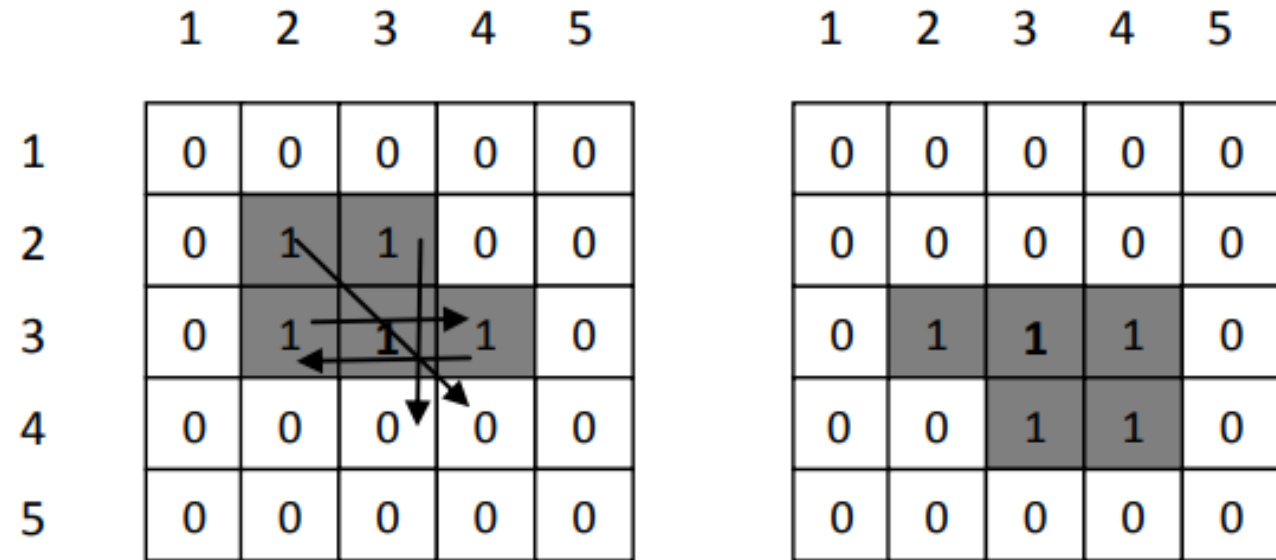
# TEORI HIMPUNAN

**Refleksi** B dinotasikan dengan  $B^{\wedge}$  dan didefinisikan sebagai berikut:

$$B^{\wedge} = \{w | w = -b, \text{ untuk } b \in B\} \dots\dots\dots (10.4)$$

Refleksi sebenarnya menyatakan pencerminan terhadap piksel pusat. Contoh ditunjukkan pada Gambar 10.9. Bayangan cermin 2-D terjadi melalui pencerminan pada arah X dan dilanjutkan pada arah Y. namun, ternyata hasilnya sama dengan pemutaran di bidang citra 180°.





$$A=\{(2,2), (2,3), (3,2), (3,3), (3,4)\}$$

$$\hat{A} = \{(3,2), (3,3), (3,4), (4,3), (4,4)\}$$

Gambar 10.9 Contoh refleksi



# OPERASI NALAR




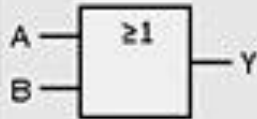


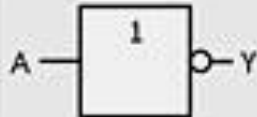




JAWA TENGAH

Operator nalar didasarkan pada aljabar Boolean. Sebagaimana diketahui, aljabar Boolean adalah pendekatan matematis yang berhubungan dengan nilai kebenaran (benar atau salah). Ada tiga operator nalar dasar yang akan dibahas, yaitu AND, OR, serta NOT.

# GERBANG LOGIKA DAN TABEL KEBENARAN



Nama	Fungsi	Lambang dalam rangkaian			Tabel kebenaran															
		IEC 60617-12	US-Norm	DIN 40700 (sebelum 1976)																
<b>Gerbang-AND</b> (AND)	$Y = A \wedge B$ $Y = A \cdot B$ $Y = AB$				<table><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	Y	0	0	0	0	1	0	1	0	0	1	1	1
A	B	Y																		
0	0	0																		
0	1	0																		
1	0	0																		
1	1	1																		
<b>Gerbang-OR</b> (OR)	$Y = A \vee B$ $Y = A + B$				<table><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	Y	0	0	0	0	1	1	1	0	1	1	1	1
A	B	Y																		
0	0	0																		
0	1	1																		
1	0	1																		
1	1	1																		
<b>Gerbang-NOT</b> (NOT, Gerbang-komplemen, Pembalik(/nverter))	$Y = \overline{A}$ $Y = \neg A$				<table><tr><th>A</th><th>Y</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	A	Y	0	1	1	0									
A	Y																			
0	1																			
1	0																			

# GERBANG LOGIKA DAN TABEL KEBENARAN



20XX

<b>Gerbang-NAND</b> (Not-AND)	$Y = \overline{A \wedge B}$ $Y = A \overline{A} B$ $Y = \overline{A} \overline{B}$				<table><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	Y	0	0	1	0	1	1	1	0	1	1	1	0
A	B	Y																		
0	0	1																		
0	1	1																		
1	0	1																		
1	1	0																		
<b>Gerbang-NOR</b> (Not-OR)	$Y = \overline{A \vee B}$ $Y = A \overline{A} B$ $Y = \overline{A} + \overline{B}$				<table><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	Y	0	0	1	0	1	0	1	0	0	1	1	0
A	B	Y																		
0	0	1																		
0	1	0																		
1	0	0																		
1	1	0																		
<b>Gerbang-XOR</b> (Antivalen, Exclusive-OR)	$Y = A \underline{\vee} B$ $Y = A \oplus B$			 atau 	<table><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	Y	0	0	0	0	1	1	1	0	1	1	1	0
A	B	Y																		
0	0	0																		
0	1	1																		
1	0	1																		
1	1	0																		
<b>Gerbang-XNOR</b> (Ekuivalen, Not-Exclusive-OR)	$Y = \overline{A \underline{\vee} B}$ $Y = A \underline{\vee} B$ $Y = \overline{A \oplus B}$			 atau 	<table><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	Y	0	0	1	0	1	0	1	0	0	1	1	1
A	B	Y																		
0	0	1																		
0	1	0																		
1	0	0																		
1	1	1																		

# OPERASI NALAR

Operasi AND melibatkan dua masukan dan mempunyai sifat bahwa hasil operasinya bernilai 1 hanya jika kedua masukan bernilai 1. Pada operasi OR, hasil berupa 1 kalau ada masukan yang bernilai 1.



20XX  
JAWA TIMUR

Tabel10.1 kebenaran AND dan OR

Masukan 1	Masukan 2	AND	OR
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

Tabel10.2 kebenaran NOT

Masukan	Keluaran
0	1
1	0

Berbeda dengan AND dan OR, operasi NOT hanya melibatkan satu masukan. Hasil NOT berupa 1 kalau masukan berupa 0 dan sebaliknya akan menghasilkan nilai 0 kalau masukan berupa 1.

# OPERASI NALAR

Selain ketiga operator yang disebut di depan, operator lain yang kadang-kadang digunakan adalah XOR dan NAND. Sifat XOR dan NAND ditunjukkan pada Tabel 10.3.

Tabel 10.3 Tabel kebenaran XOR dan NAND

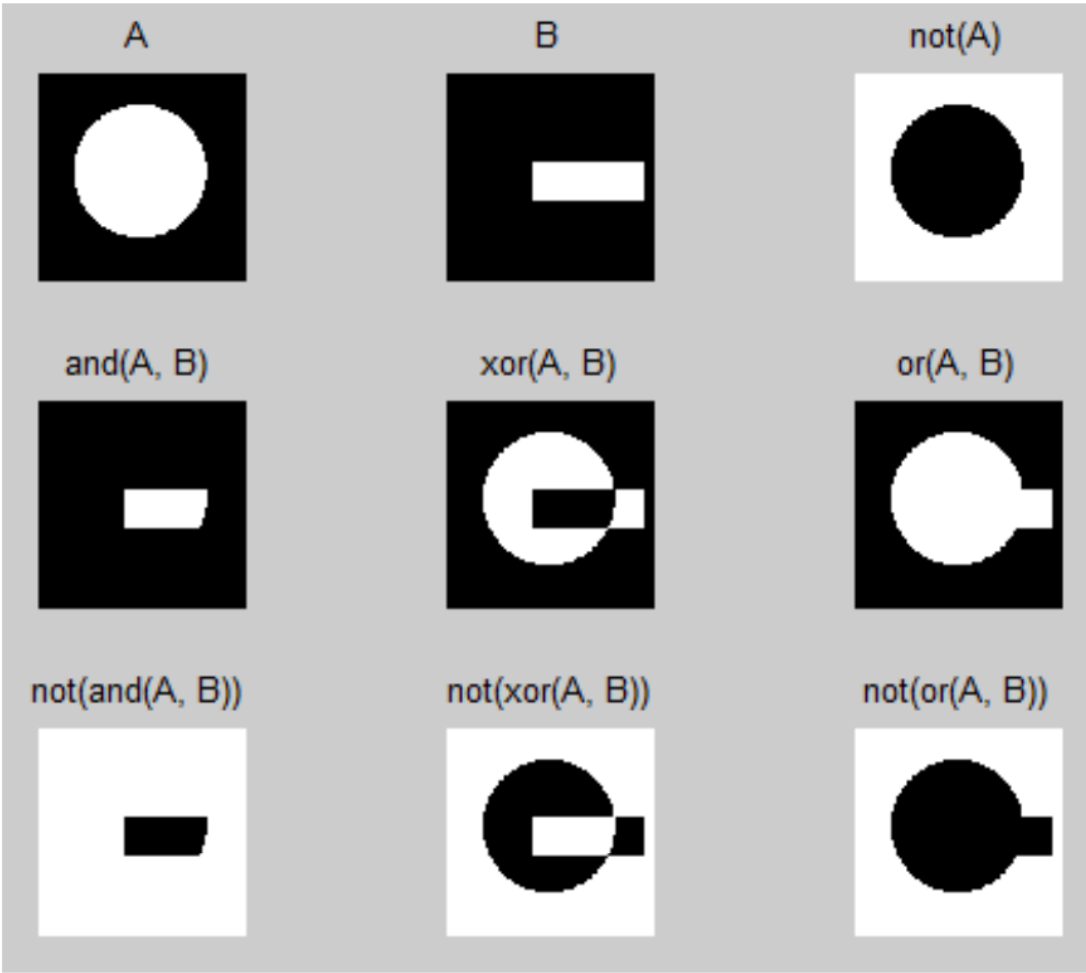
Masukan 1	Masukan 2	XOR	NAND
0	0	0	1
0	1	1	1
1	0	1	1
1	1	0	0





# OPERASI NALAR

Berbagai efek operasi AND, OR, NOT, XOR, dan NAND ditunjukkan pada Gambar 10.10.



Gambar 10.10 Hasil-hasil operasi nalar terhadap citra A dan B





# OPERASI DILASI

Operasi dilasi biasa dipakai untuk mendapatkan efek pelebaran, terhadap piksel bernilai 1.

Burger dan Burge (2008) mendefinisikan operasi dilasi sebagai berikut :

$$A \oplus B = \{z | z = a + b, \text{ dengan } a \in A$$

$$\text{dan } b \in B\} \dots\dots\dots (10.5)$$

Hasil dilasi berupa penjumlahan seluruh pasangan koordinat dari A dan B.



# OPERASI DILASI

Contoh operasi dilasi dengan menggunakan Persamaan 10.5, Pada contoh tersebut, dapat dilihat pada Gambar 10.11.

$$A = \{ (2,2), (2,3), (2,4), (3,2), (3,3), (3,4), (4,3) \}$$

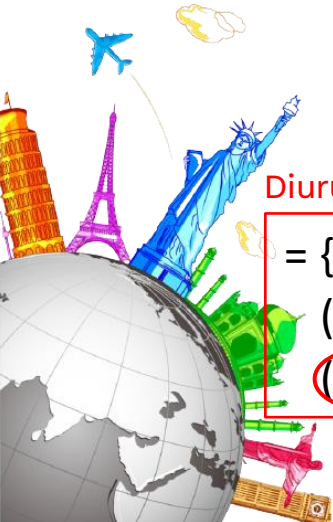
$$B = \{ (-1, 0), (0,0), (1,0) \}$$

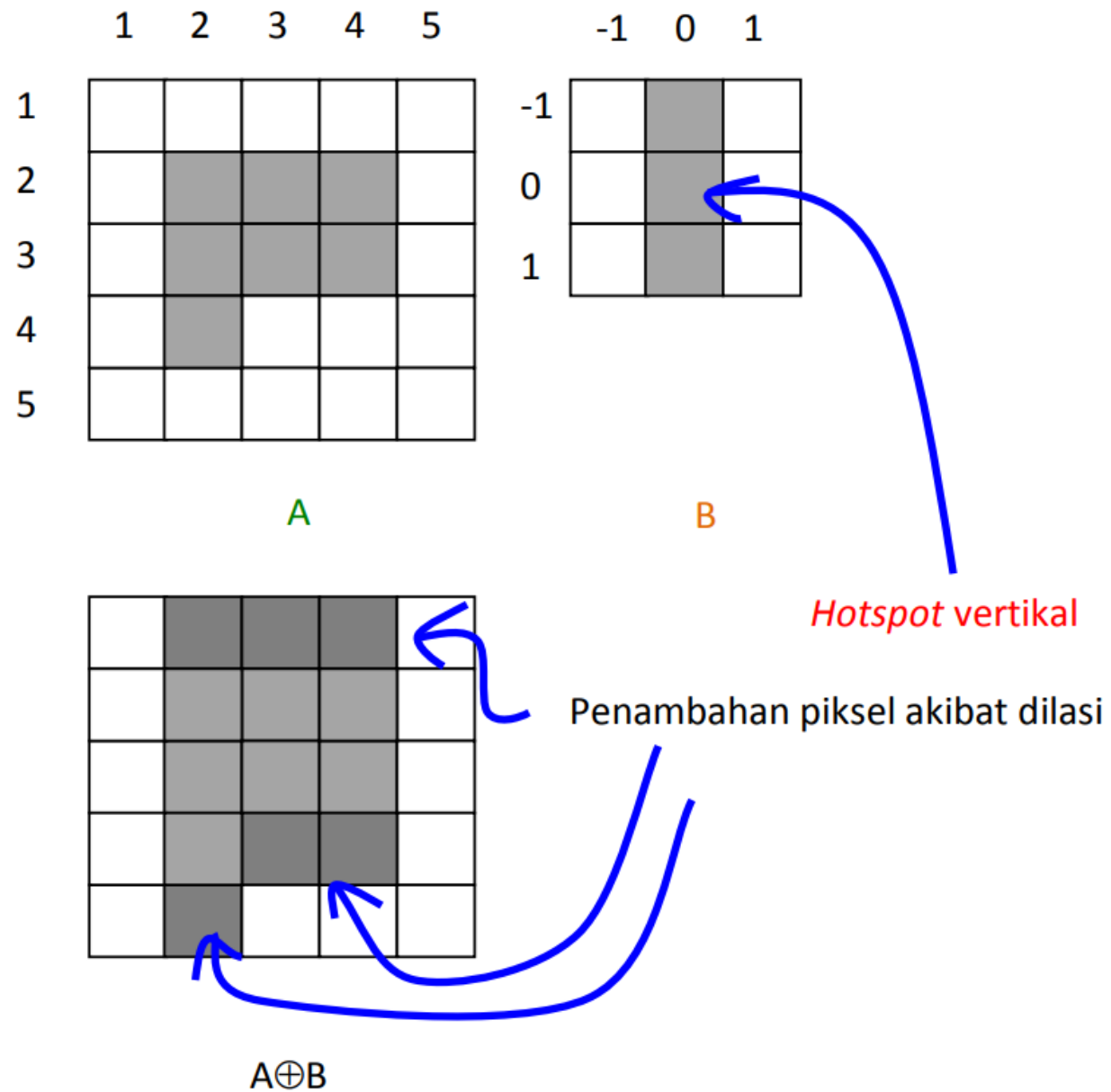
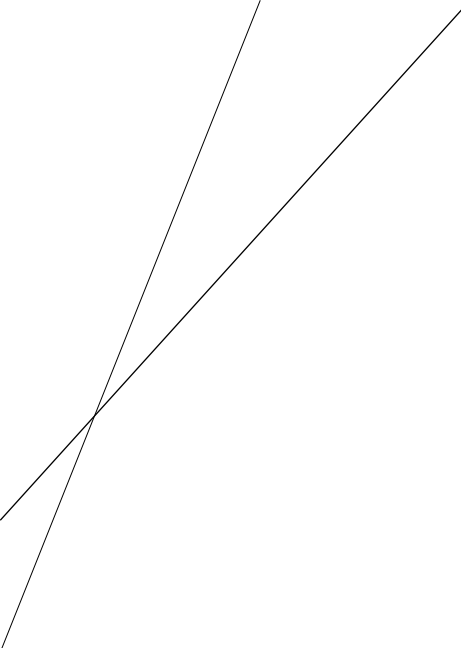
Dengan demikian,

$$\begin{aligned} A \oplus B &= \{ (2,2) + (-1, 0) , (2,2) + (0, 0) + (2,2) + (1, 0), \\ &\quad (2,3) + (-1, 0) , (2,3) + (0, 0) + (2,3) + (1, 0), \\ &\quad (2,4) + (-1, 0) , (2,4) + (0, 0) + (2,4) + (1, 0), \\ &\quad (3,2) + (-1, 0) , (3,2) + (0, 0) + (3,2) + (1, 0), \\ &\quad (3,3) + (-1, 0) , (3,3) + (0, 0) + (3,3) + (1, 0), \\ &\quad (3,4) + (-1, 0) , (3,4) + (0, 0) + (3,4) + (1, 0), \\ &\quad (4,3) + (-1, 0) , (4,3) + (0, 0) + (4,3) + (1, 0) \} \\ &= \{ (1,2), (2,2), (3,2), (1,3), (2,3), (3,3), (1,4), \\ &\quad (2,4), (3,3), (2,2), (3,2), (4,2), (2,3), (3,3), \\ &\quad (4,3), (2,4), (3,4), (4,4), (3,3), (4,3), (5,3) \} \\ &= \{ (1,2), (1,3), (1,4), (2,2), (2,3), (2,4), (3,2), \\ &\quad (3,3), (3,4), (4,2), (4,3), (4,4), (5,3) \} \end{aligned}$$

Diurutkan dan untuk angka yang kembar di tulis sekali saja

$$\begin{aligned} &= \{ (1,2), (2,2), (3,2), (1,3), (2,3), (3,3), (1,4), \\ &\quad (2,4), (3,3), (2,2), (3,2), (4,2), (2,3), (3,3), \\ &\quad (4,3), (2,4), (3,4), (4,4), (3,3), (4,3), (5,3) \} \end{aligned}$$





Gambar 10.11 Efek dilasi dengan hotspot vertikal



(a) citra bravo.png beraras keabuan



(b) Hasil konversi ke biner menggunakan im2bw

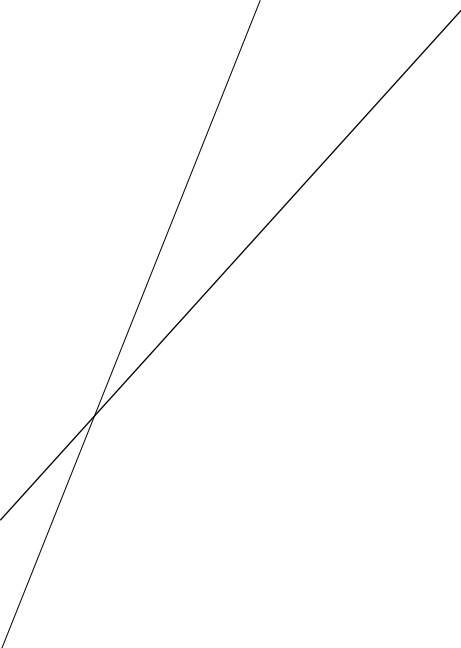


(c) Hasil dilasi dengan elemen penstruktur 4 x 4



(d) Hasil dilasi dengan elemen penstruktur 7 x 7

Gambar 10.12 Contoh operasi dilasi pada citra



# OPERASI EROSI

Operasi erosi mempunyai efek memperkecil struktur citra.

$$A \ominus B = \{Z \mid (B)_Z \subseteq A \dots\dots\dots(10.6)$$

atau menggunakan persamaan Burger dan Burge

$$A \ominus B = \{p \in Z^2 \mid (a + b) \in I, \text{ untuk} \\ \text{setiap } b \in B\} \dots\dots\dots(10.7)$$

Berdasarkan persamaan 10.6 dan 10.7, posisi p terdapat pada  $A \ominus B$  jika seluruh nilai 1 di B terkandung di posisi p tersebut.

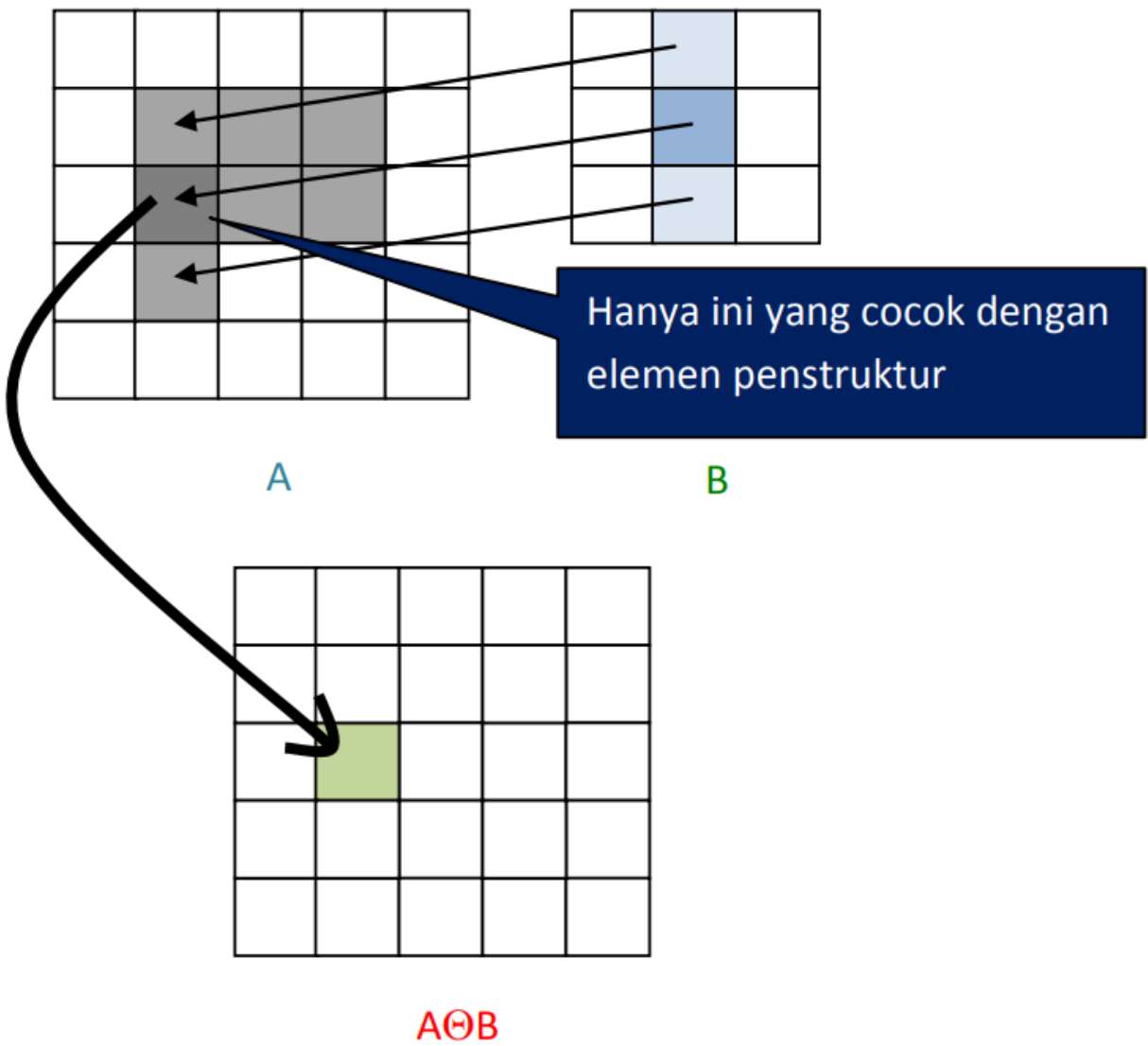


# OPERASI EROSI

Implementasi fungsi erosi berikut didasarkan makna operasi erosi, ditunjukkan pada Gambar 10.13.



20XX



Gambar 10.13 Contoh visualisasi operasi erosi





(a) Citra asli daun.png



(b) Hasil konversi ke citra biner



(c) Erosi dengan  $H = \text{ones}(4)$



(d) Erosi dengan  $H = \text{ones}(6)$

Gambar 10.14 Contoh operasi erosi pada citra

# OPERASI OPENING

Operasi **opening** adalah operasi erosi yang diikuti dengan dilasi dengan menggunakan elemen penstruktur yang sama. Operasi ini berguna untuk menghaluskan kontur objek dan menghilangkan seluruh piksel di area yang terlalu kecil untuk ditempati oleh elemen penstruktur.

$$A \circ B = (A \ominus B) \oplus B \dots\dots\dots (10.7)$$

Dengan kata lain, semua struktur latar depan yang berukuran lebih kecil dari pada elemen penstruktur akan tereliminasi oleh erosi dan kemudian penghalusan dilakukan melalui dilasi.





# OPERASI OPENING

Ditunjukkan pada Gambar 10. 15 bahwa operasi erosi membuat objek mengecil dan bahkan ada yang hilang. Adapun operasi opening membuat ukuran objek relatif tetap sama, walaupun juga menghilangkan objek yang berukuran kecil (kurus). Operasi opening

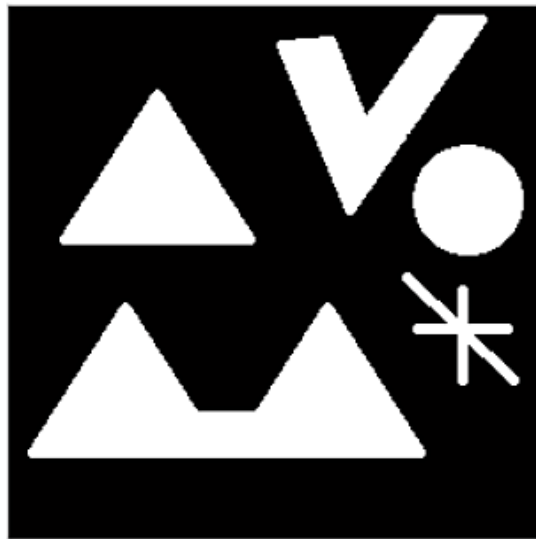
membuat penghalusan di bagian tepi. Perhatikan, ujung segitiga tidak tajam setelah dikenai operasi opening. Sebagai pembandingan, Gambar 7.21(d) menunjukkan hasil penggunaan operasi closing





20XX

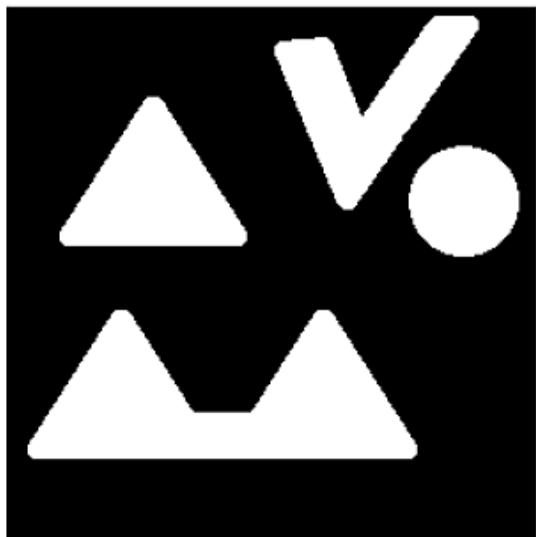
MALUKU UTARA



(a) Citra struktur.png



(b) Hasil erosi dengan strel('disk', 5)



(c) Operasi opening  
dengan strel('disk', 5)



(d) Operasi closing  
dengan strel('disk', 5)

Operasi opening sering dikatakan sebagai idempotent. Artinya, jika suatu citra telah dikenai operasi opening, pengenalan opening dengan elemen penstruktur yang sama tidak membawa efek apapun.

Gambar 10.15 Perbandingan operasi erosi, opening dan closing pada citra

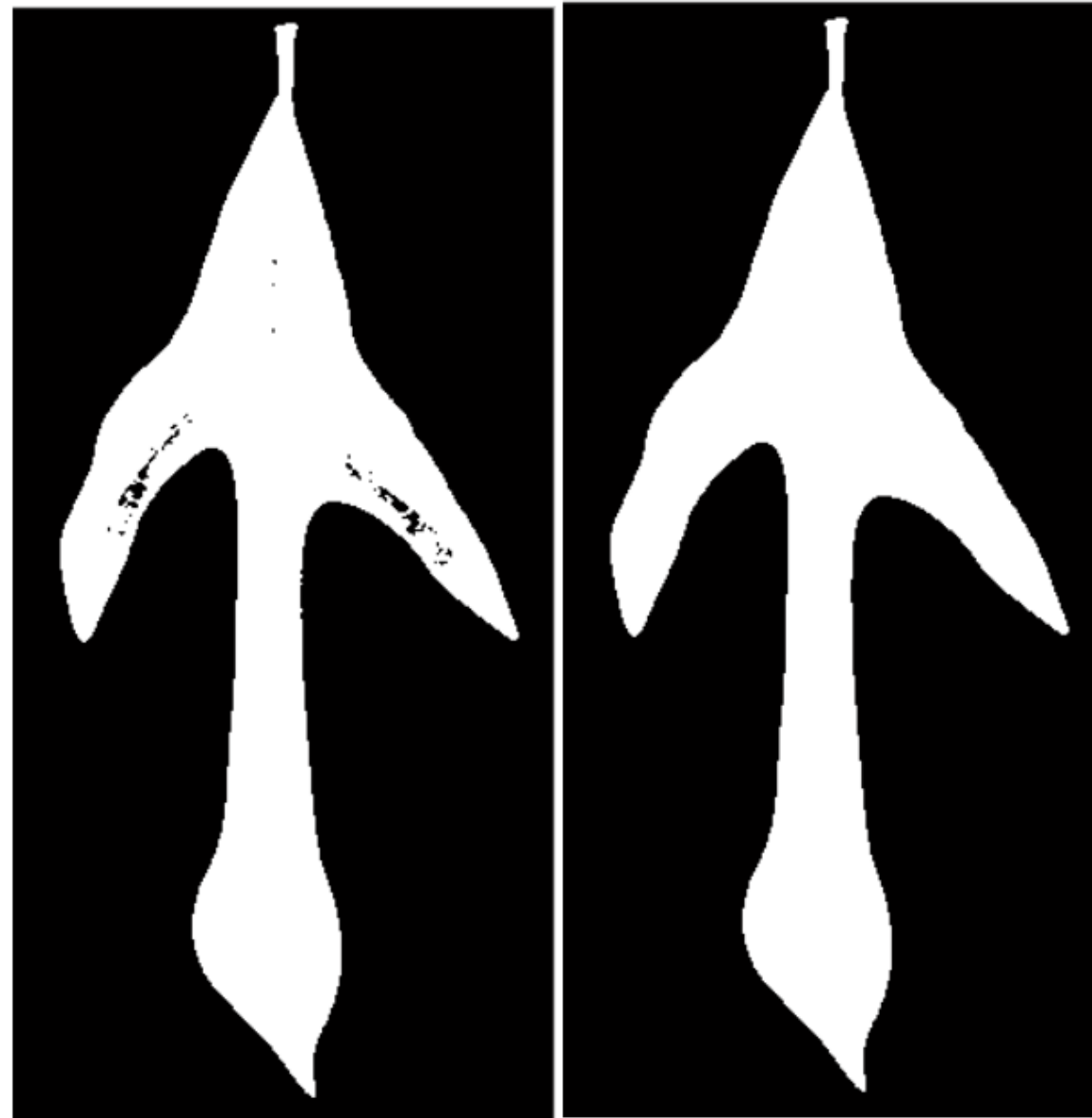
# OPERASI CLOSING

Operasi closing berguna untuk menghaluskan kontur, dan menghilangkan lubang-lubang kecil. Definisinya seperti berikut:

$$A \bullet B = (A \oplus B) \ominus B \quad \dots\dots\dots (10.8)$$

Operasi closing dilaksanakan dengan melakukan operasi dilasi terlebih dahulu dan kemudian diikuti dengan operasi erosi.





(a) Hasil konversi ke biner

(b) Hasil operasi *closing*

Gambar 10.16 Lubang kecil pada citra tertutup oleh operasi closing





# TERIMAKASIH



PRESENTATION TITLE

