

رسالة محمد



مبانی بینایی کامپیوتر

مدرس: محمدرضا محمدی

بهار ۱۴۰۲

پردازش‌های مورفولوژی

Morphological Image Processing

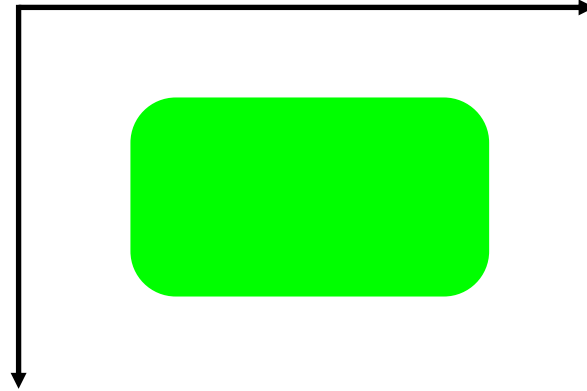
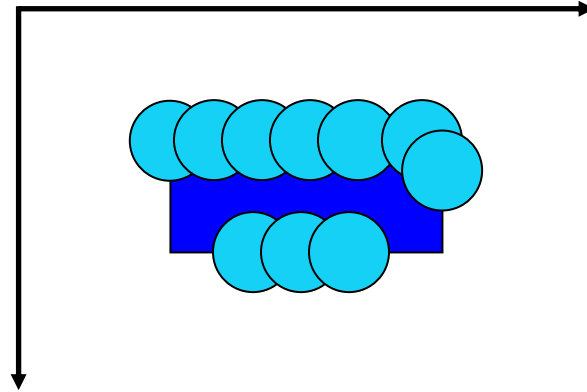
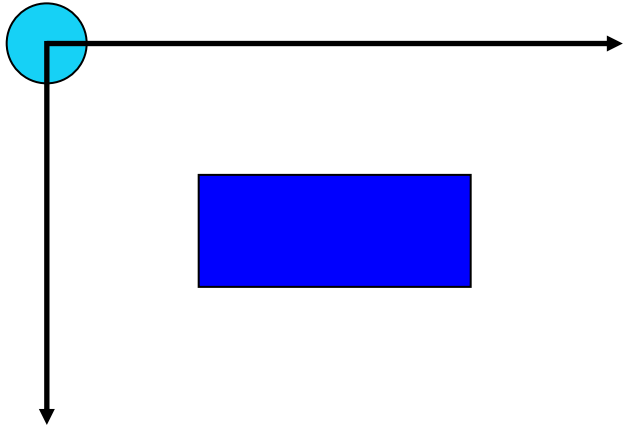
عملگر گسترش

- عملگر گسترش (dilate) برای گسترش مجموعه A توسط B به صورت زیر تعریف می‌شود:

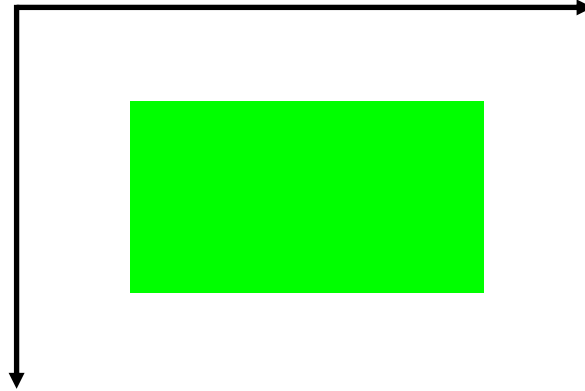
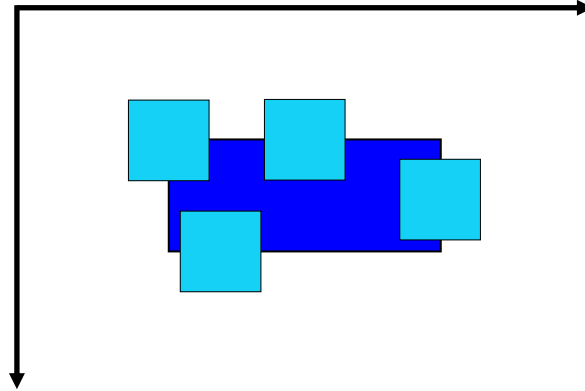
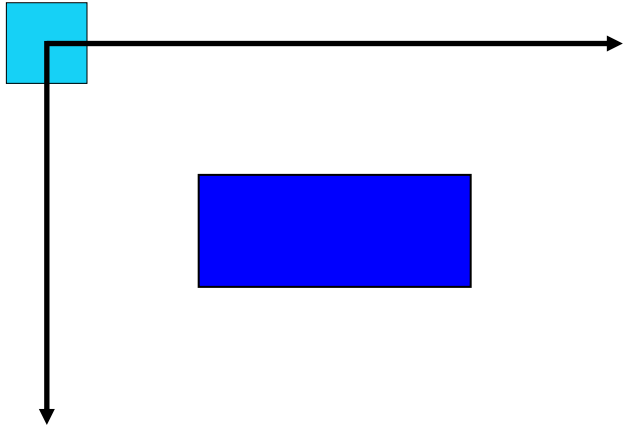
$$A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \emptyset\}$$

- این رابطه به مفهوم بدست آوردن انعکاس B حول مرکز (لنگر) خودش و جابجایی آن به اندازه z است که اگر این نسخه از B دارای اشتراک با A بود، z جزء مجموعه جدید خواهد بود

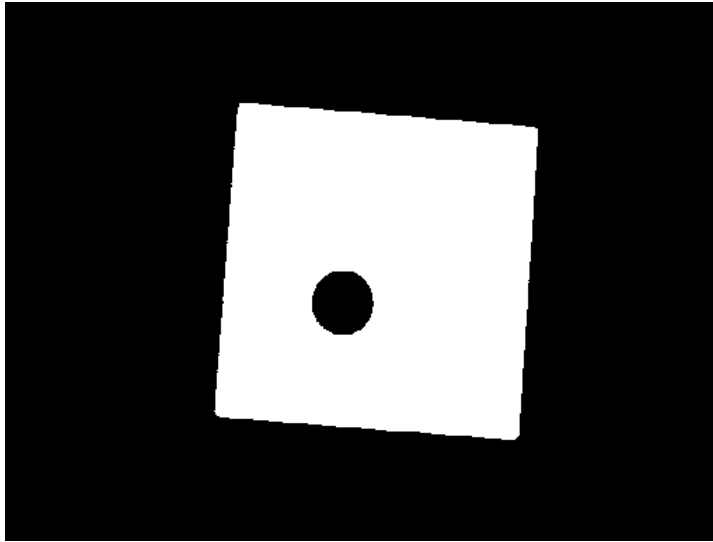
مثال: گسترش 2D



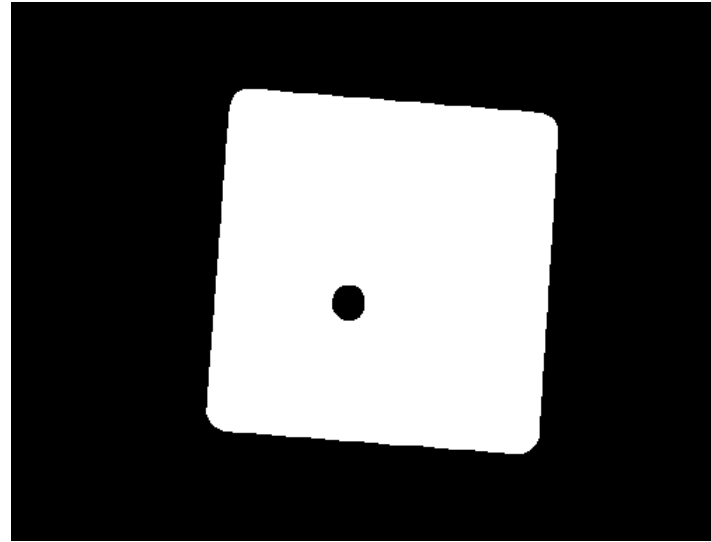
مثال: گسترش 2D



مثال: گسترش 2D



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



عملگر سایش

- عملگر سایش (erode) برای فرسایش مجموعه A توسط B به صورت زیر تعریف می‌شود:

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

- بنابراین سایش مجموعه A توسط B شامل مجموعه نقاطی است که به ازای آنها B به طور کامل درون A قرار می‌گیرد

مثال: سایش 1D

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

Input image

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



Structuring Element

1	1	1
---	---	---



Output Image

0										
---	--	--	--	--	--	--	--	--	--	--

مثال: سایش 1D

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

Input image

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



Structuring Element

1	1	1
---	---	---



Output Image

0	0								
---	---	--	--	--	--	--	--	--	--

مثال: سایش 1D

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

Input image

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



Structuring Element

1	1	1
---	---	---



Output Image

0	0	0							
---	---	---	--	--	--	--	--	--	--

مثال: سایش 1D

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

Input image

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



Structuring Element

1	1	1
---	---	---



Output Image

0	0	0	0						
---	---	---	---	--	--	--	--	--	--

مثال: سایش 1D

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

Input image

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



Structuring Element

1	1	1
---	---	---



Output Image

0	0	0	0	0					
---	---	---	---	---	--	--	--	--	--

مثال: سایش 1D

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

Input image

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



Structuring Element

1	1	1
---	---	---



Output Image

0	0	0	0	0	0				
---	---	---	---	---	---	--	--	--	--

مثال: سایش 1D

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

Input image

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



Structuring Element

1	1	1
---	---	---



Output Image

0	0	0	0	0	0	0			
---	---	---	---	---	---	---	--	--	--

مثال: سایش 1D

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

Input image

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



Structuring Element

1	1	1
---	---	---



Output Image

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

مثال: سایش 1D

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

Input image

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



Structuring Element

1	1	1
---	---	---



Output Image

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

مثال: سایش 1D

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

Input image

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



Structuring Element

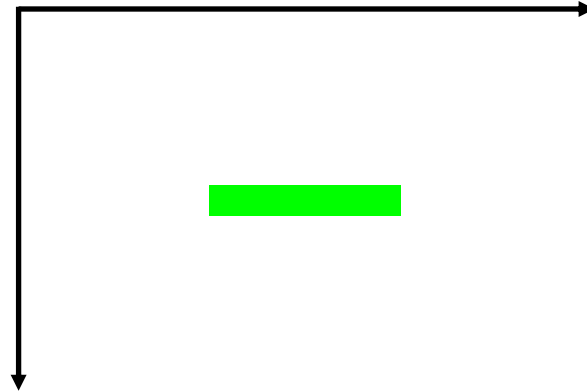
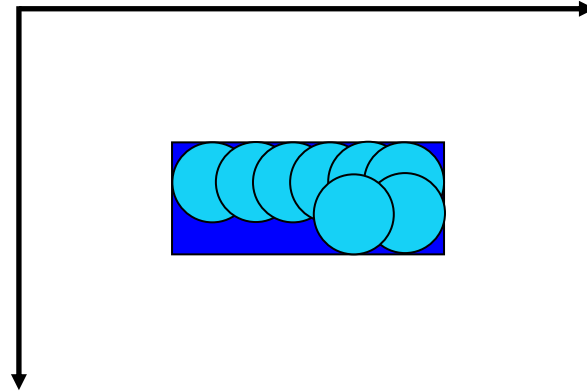
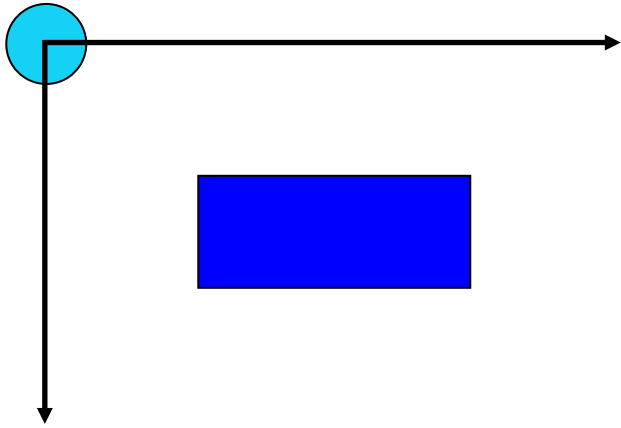
1	1	1
---	---	---



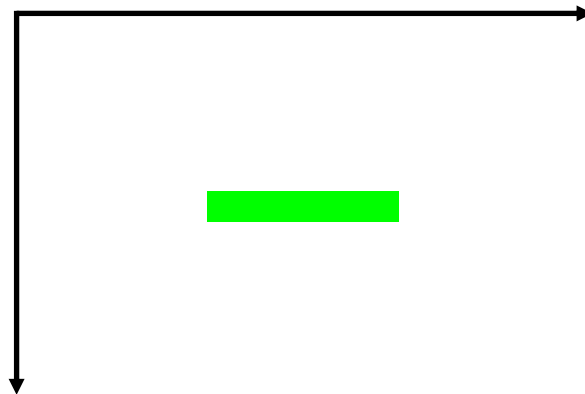
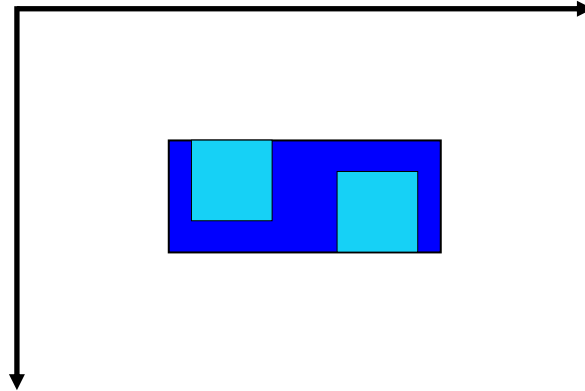
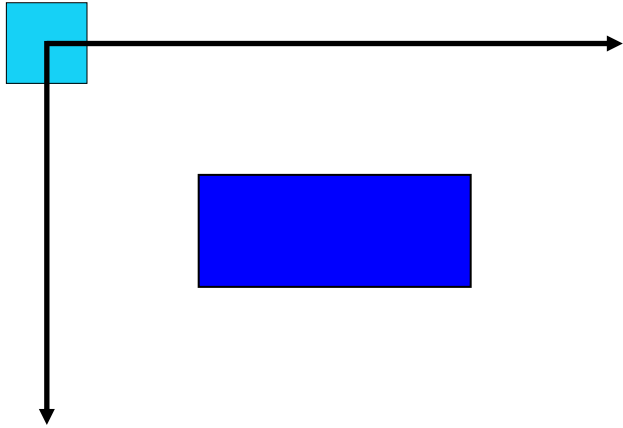
Output Image

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

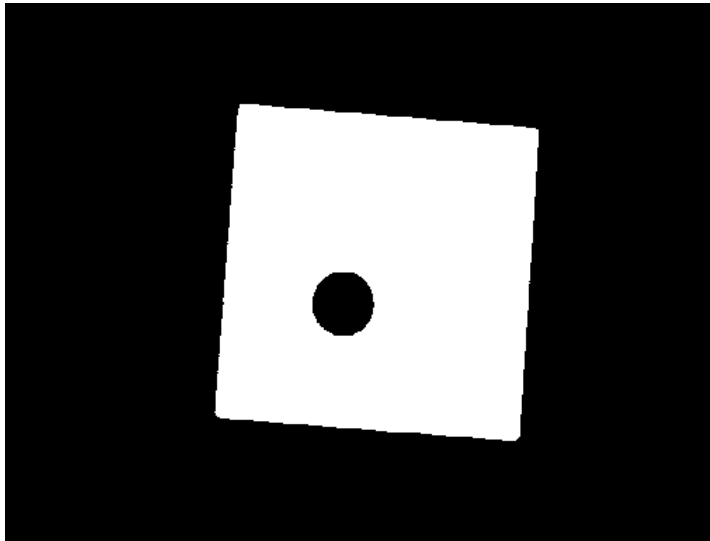
مثال: سایش 2D



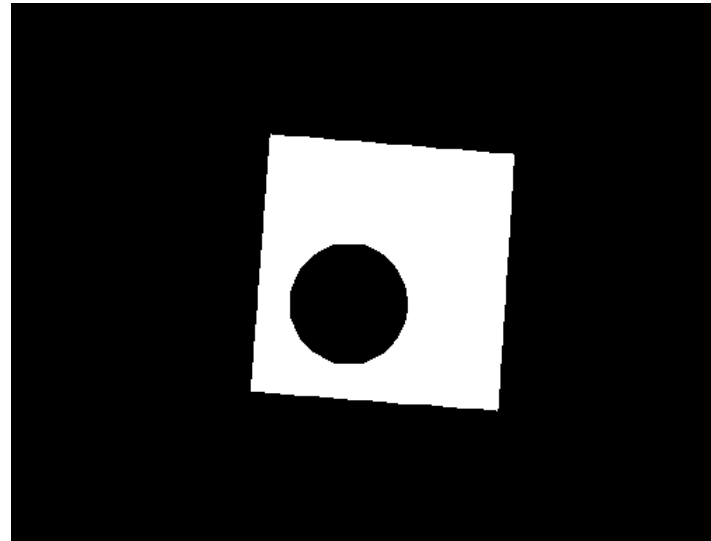
مثال: سایش 2D



مثال: سایش 2D

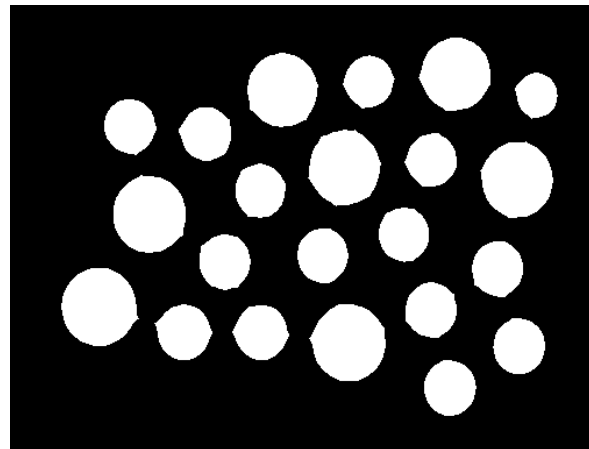
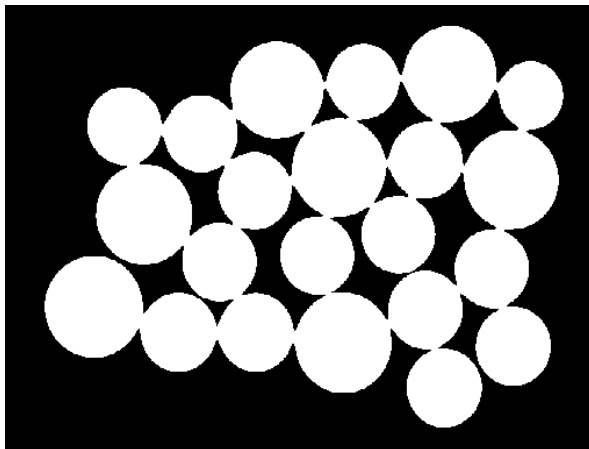
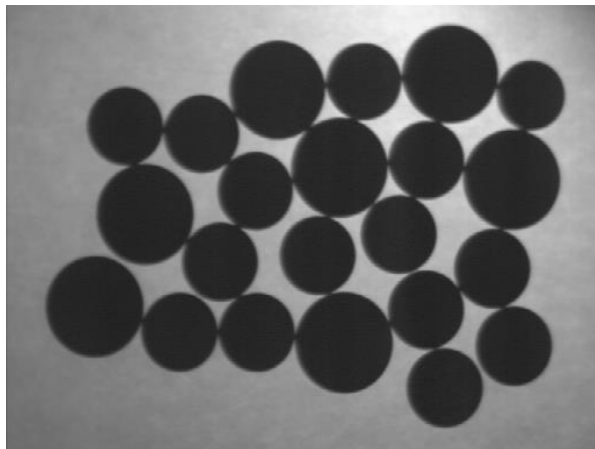


0	0	1	1	1	0	0
0	1	1	1	1	1	0
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
0	1	1	1	1	1	0
0	0	1	1	1	0	0



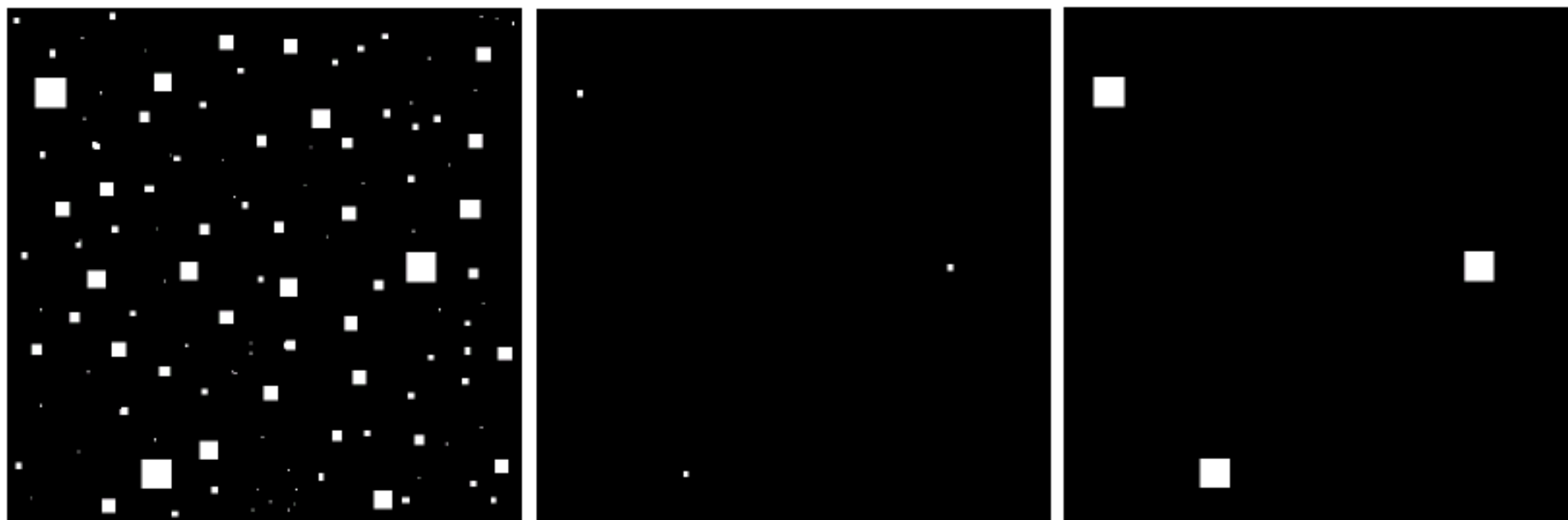
مثال: شمارش سکه‌ها

- چگونه می‌شود تعداد سکه‌هایی را شمرد که با یکدیگر در تماس هستند؟
- می‌توان تصویر را دوسطحی کرد
- سپس، توسط عملگر سایش آنها را جدا نمود



حذف جزئیات غیر ضروری

- یکی از ساده‌ترین کاربردهای سایش حذف جزئیات غیر ضروری است

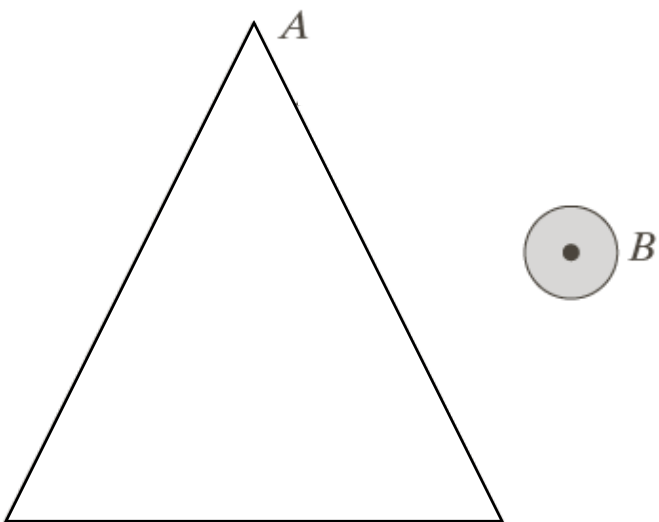


عملگر باز

- عملگر باز (opening) برای حذف جزئیات کوچک و هموار کردن محیط نواحی تعریف شده است

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

- این عملگر ناحیه‌های سفید که در احاطه پیکسل‌های سیاه هستند را حذف می‌کند

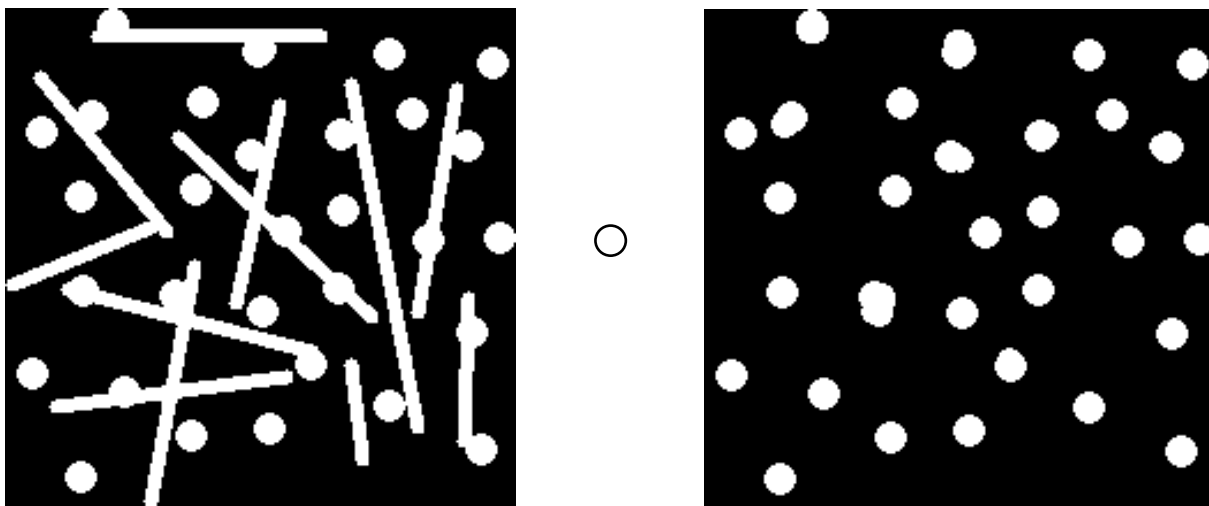


عملگر باز

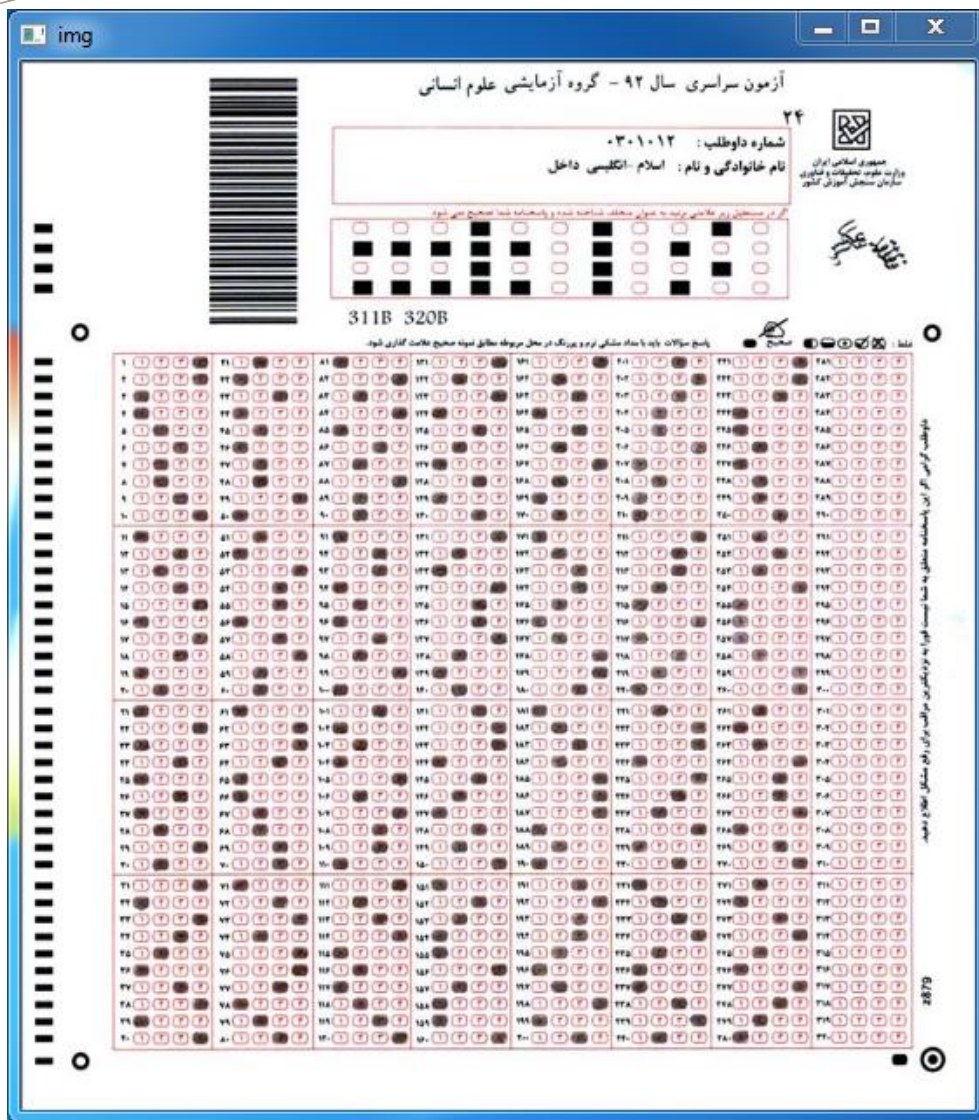
- عملگر باز (opening) برای حذف جزئیات کوچک و هموار کردن محیط نواحی تعریف شده است

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

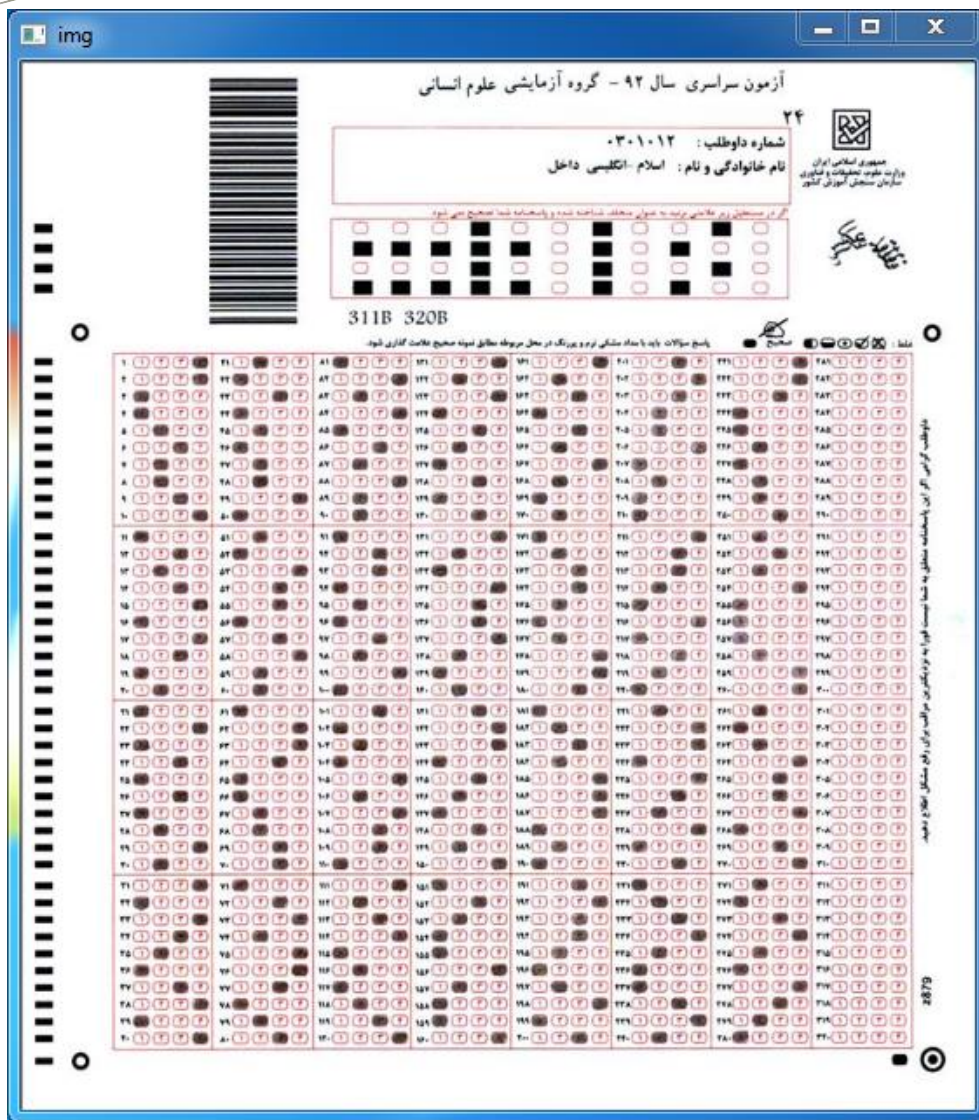
- این عملگر ناحیه‌های سفید که در احاطه پیکسل‌های سیاه هستند را حذف می‌کند



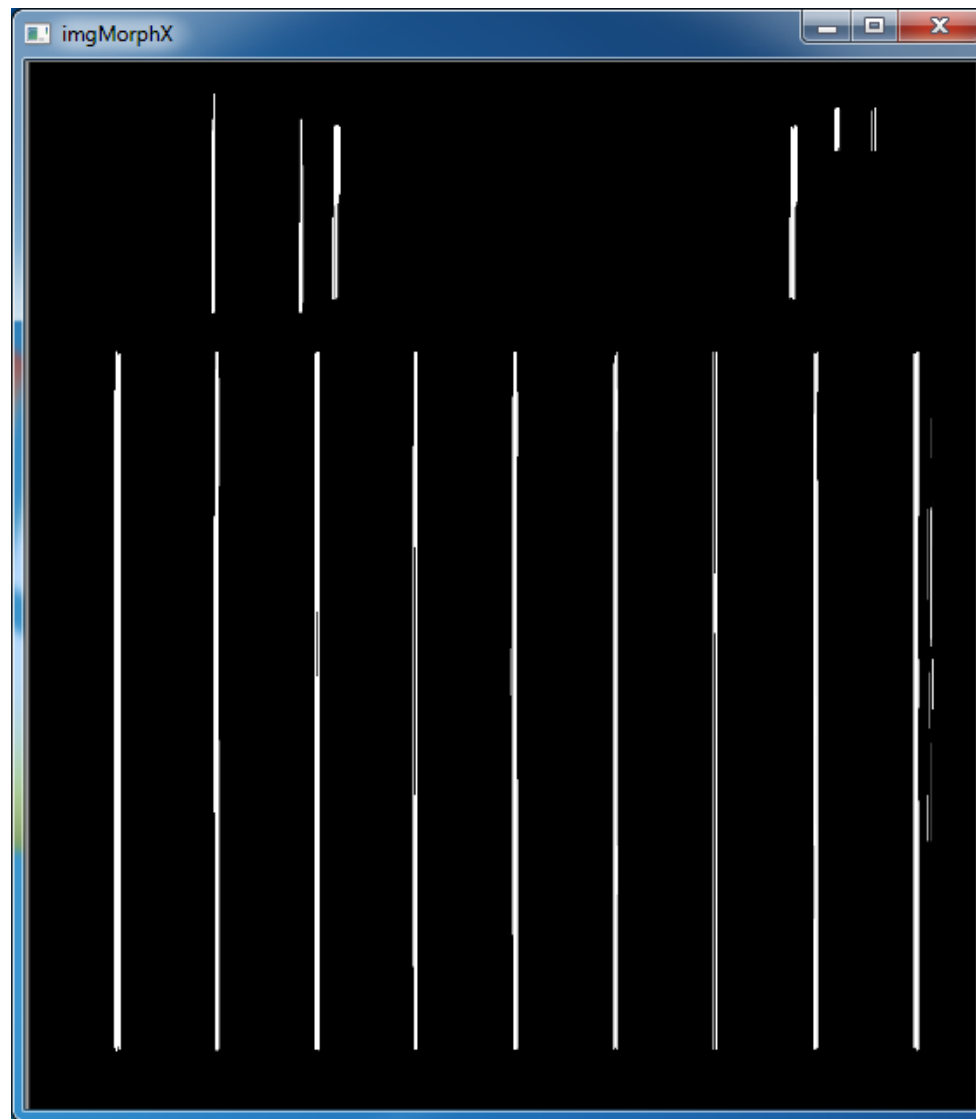
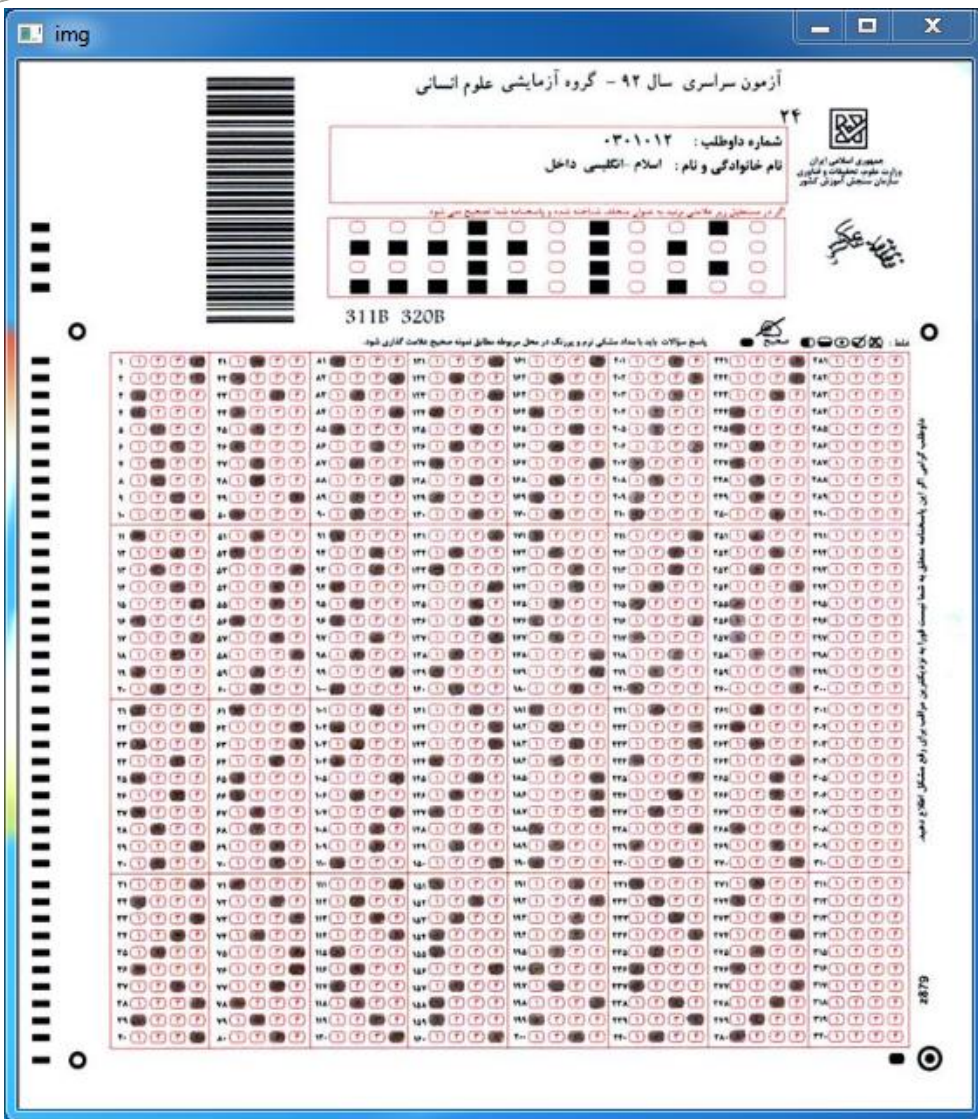
عملگر باز



عملگر باز



عملگر باز

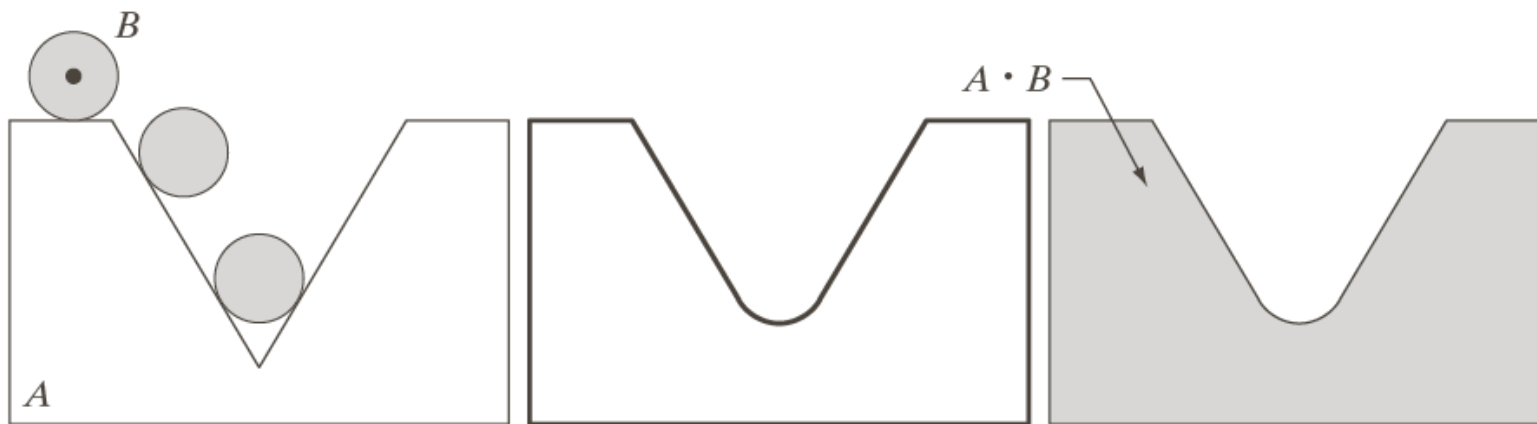


عملگر بسته

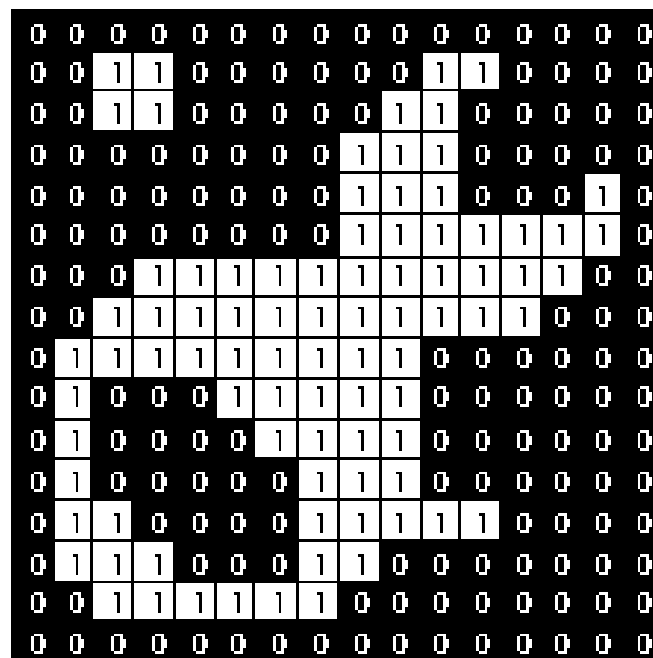
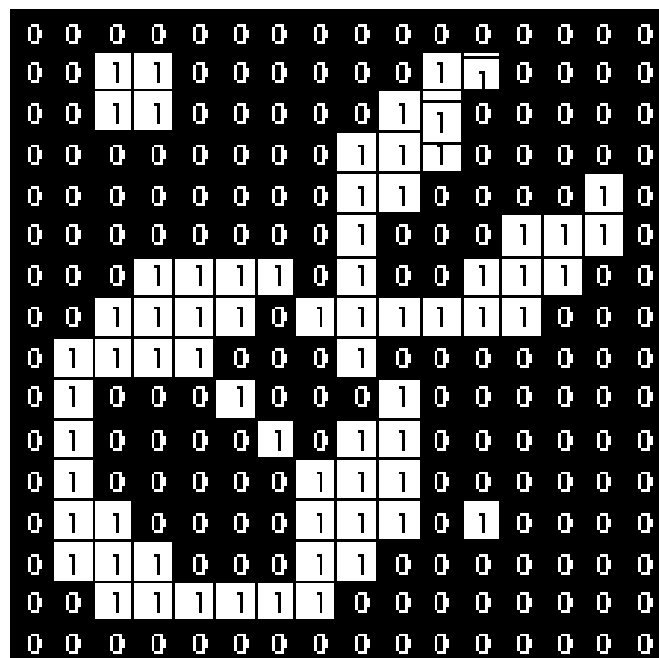
- عملگر بسته (closing) برای حذف حفره‌های کوچک و هموار کردن محیط نواحی تعریف شده است

$$A \cdot B = (A \oplus B) \ominus B$$

- این عملگر ناحیه‌های سیاه که در احاطه پیکسل‌های سفید هستند را حذف می‌کند

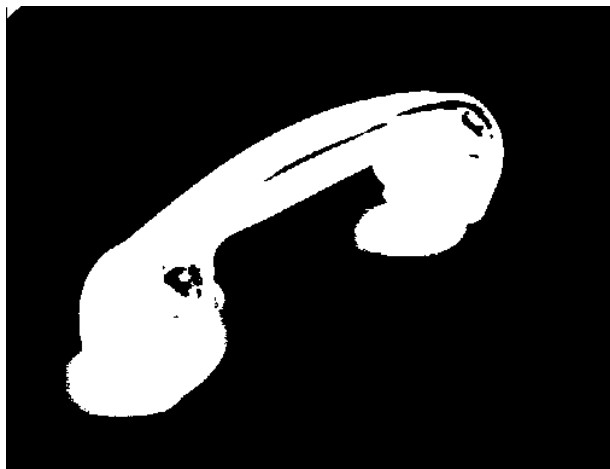
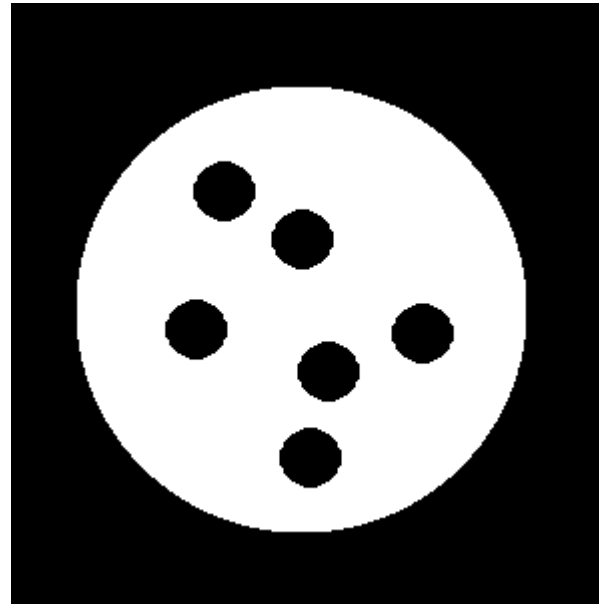
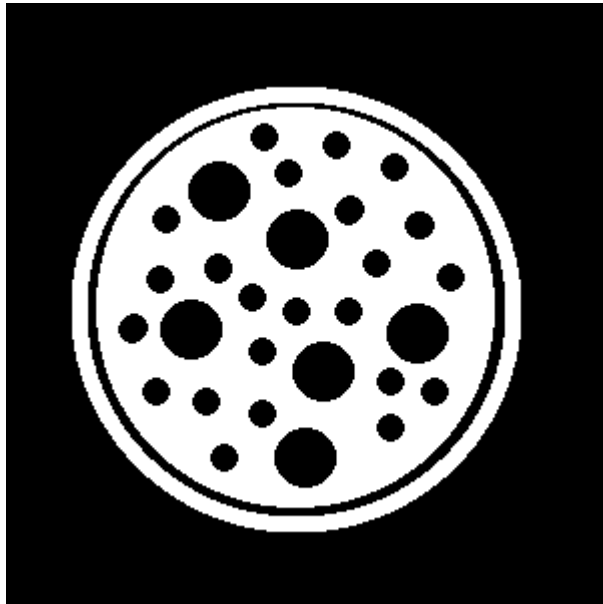


عملگر بسته

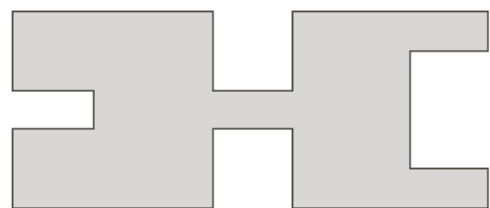


1	1	1
1	1	1
1	1	1

عملگر بسته



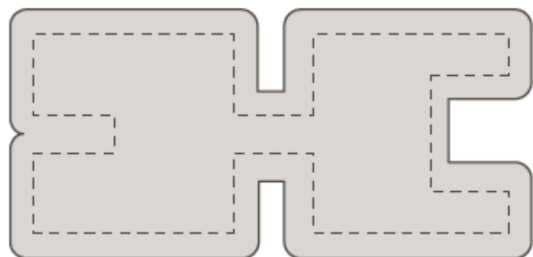
عملگرهای باز و بسته



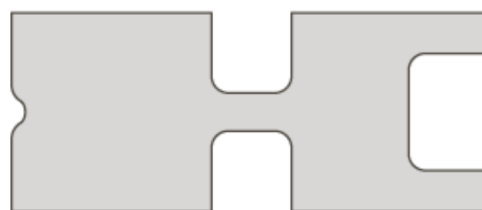
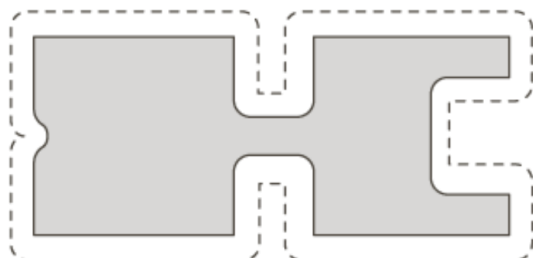
A



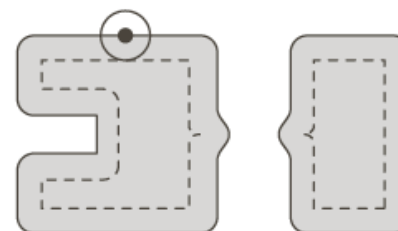
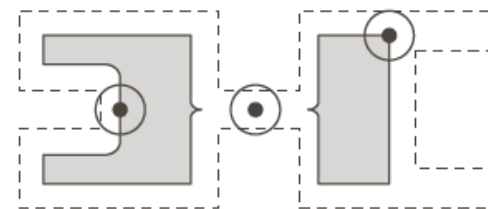
B



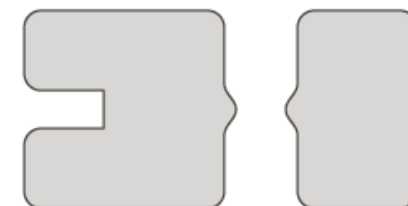
$A \oplus B$



$A \cdot B = (A \oplus B) \ominus B$



$A \ominus B$



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

عملگرهای باز و بسته

$$A \ominus B$$



$$B$$

1	1	1
1	1	1
1	1	1



$$((A \circ B) \oplus B) \ominus B = (A \circ B) \cdot B$$

$$(A \circ B) \oplus B$$

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



عملگر Hit-or-Miss

- عملگر Hit-or-Miss یک پردازش مورفولوژی برای تشخیص شکل یک ناحیه است و از آن برای استخراج الگویی در تصویر استفاده می‌شود
- تفاوت این عملگر با عملگر سایش آن است که پیکسل‌های سیاه نیز اهمیت پیدا می‌کنند
- به طور مثال، Hit-or-Miss با پنجره زیر یعنی ۵ عدد ۱ و اطراف آنها ۴ عدد صفر باشد

0	1	0
1	1	1
0	1	0

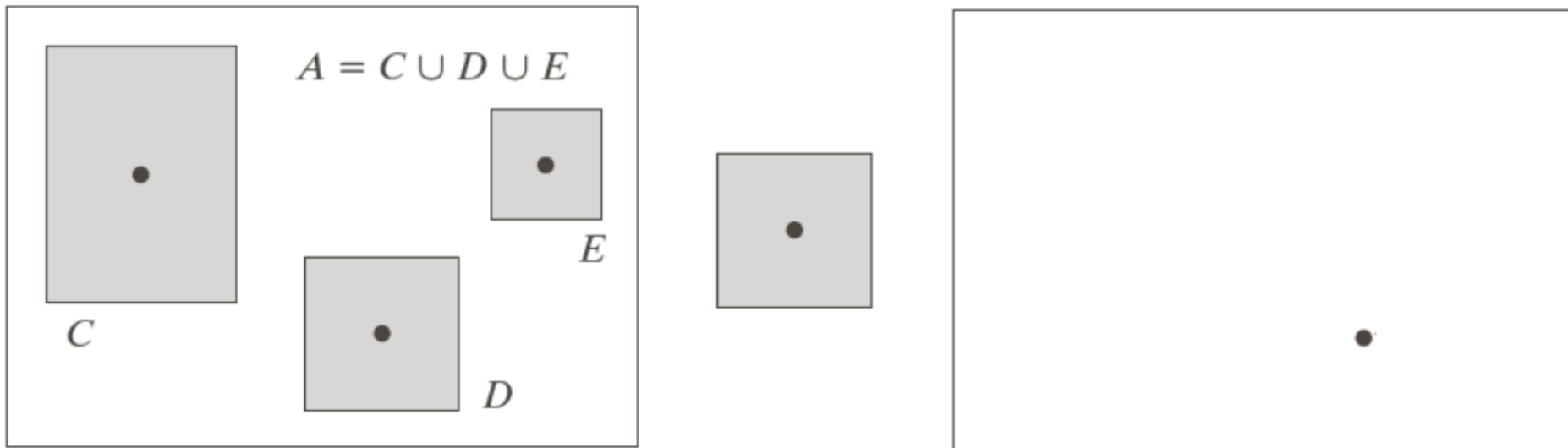
عملگر Hit-or-Miss

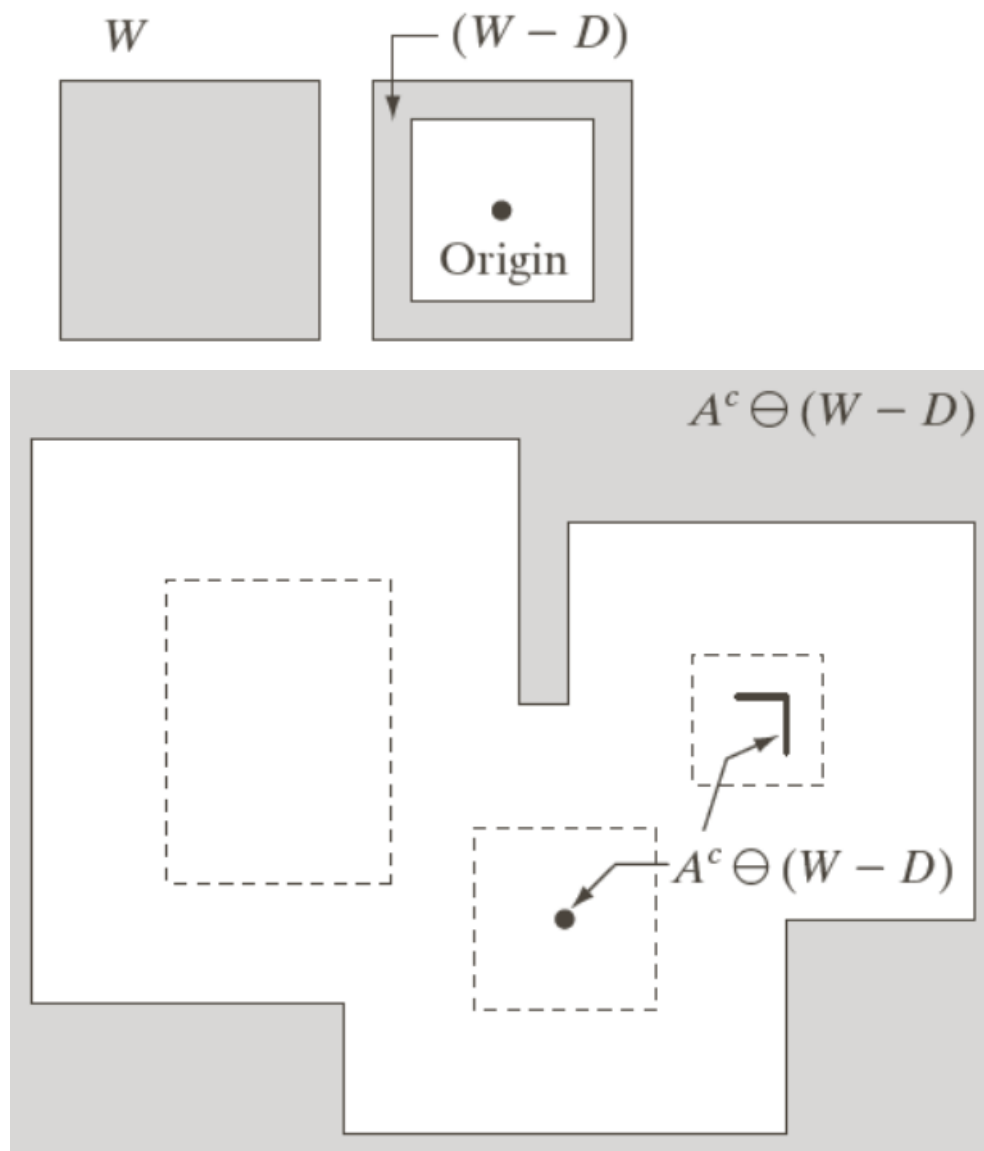
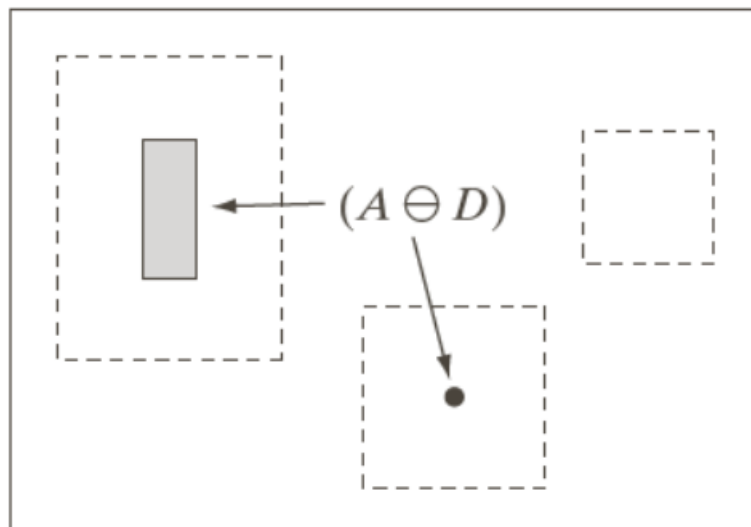
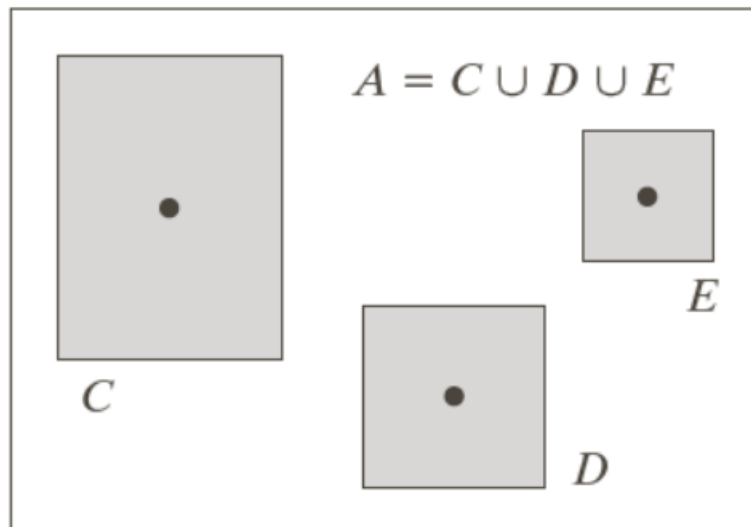
- عملگر Hit-or-Miss یک پردازش مورفولوژی برای تشخیص شکل یک ناحیه است و از آن برای استخراج

الگویی در تصویر استفاده می‌شود

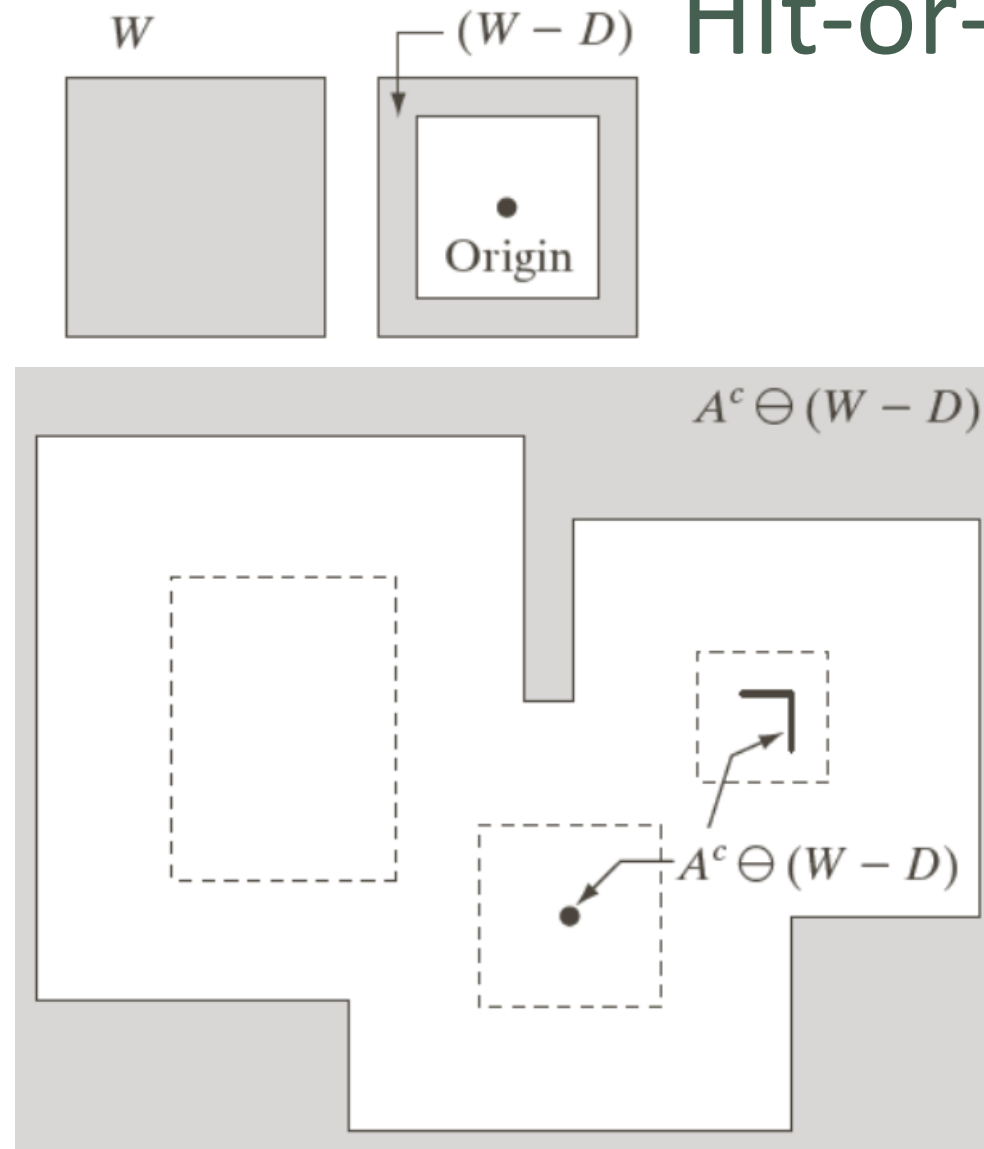
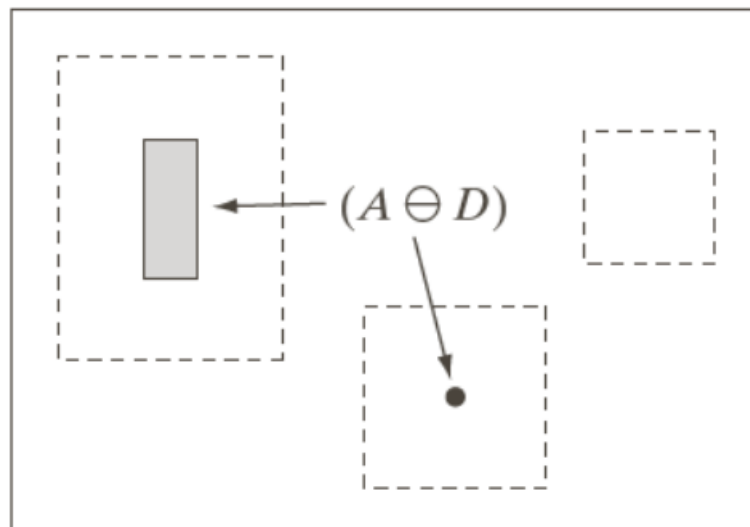
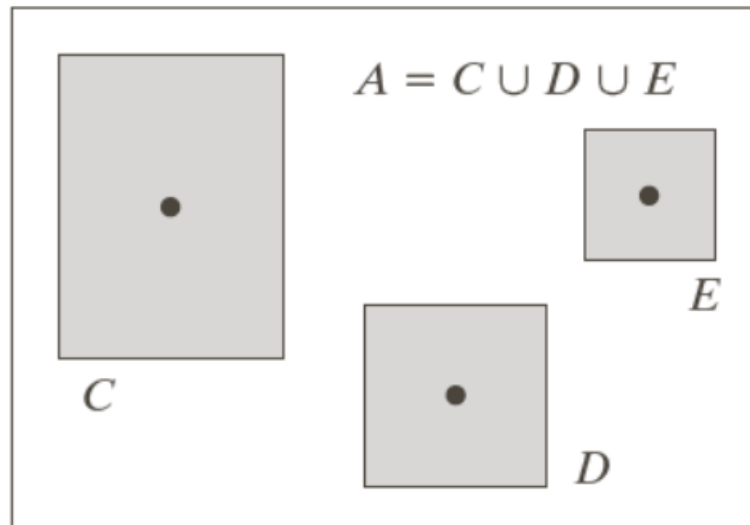
$$(A \circledast B) = (A \ominus X) \cap (A^c \ominus (W - X))$$

$$(A \circledast B) = (A \ominus B_1) \cap (A^c \ominus B_2)$$





عملگر Hit-or-Miss



عنصر ساختاری Hit-or-Miss

0	1	0
1	0	1
0	1	0

B_1

0	0	0
0	1	0
0	0	0

B_2

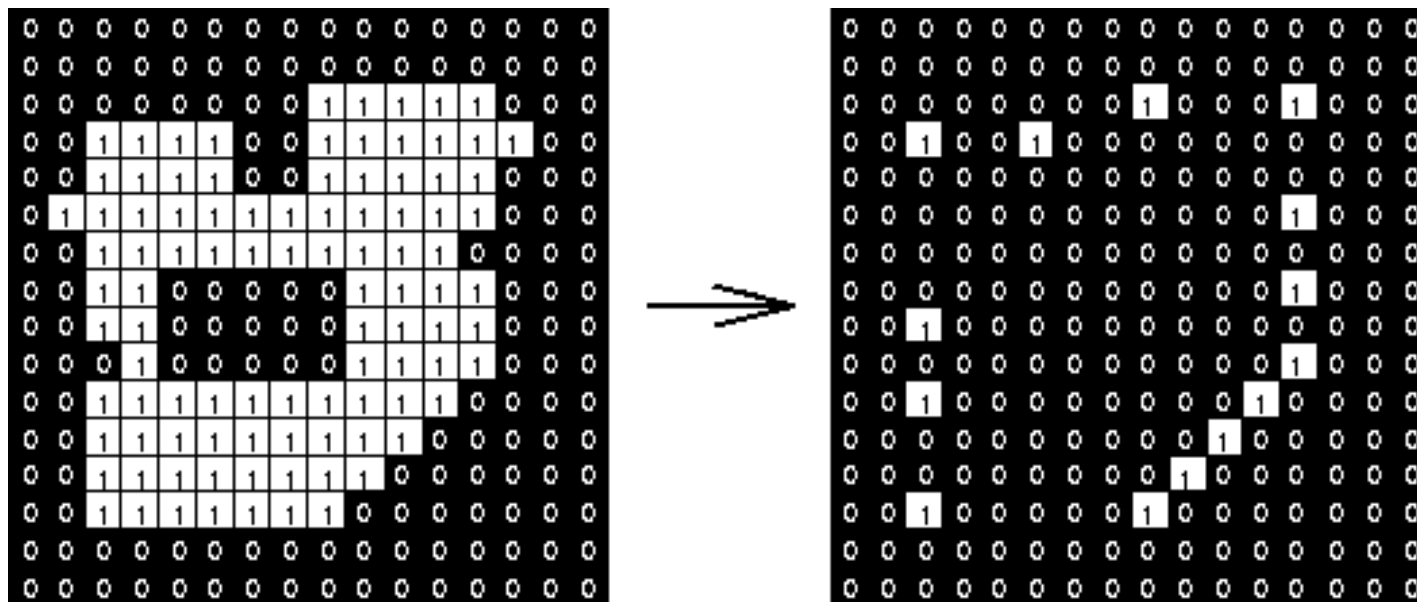
0	1	0
1	-1	1
0	1	0

B

0	0	0	0	0	0	0	0
0	255	255	255	0	0	0	255
0	255	255	255	0	0	0	0
0	255	255	255	0	255	0	0
0	0	255	0	0	0	0	0
0	0	255	0	0	255	255	0
0	255	0	255	0	0	255	0
0	255	255	0	0	0	0	0

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	255	0	0	0	0	0
0	0	0	0	0	0	0	0

تشخیص گوشه‌ها



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

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

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

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

استخراج مرز

- مرز مجموعه A را با $\beta(A)$ نمایش می‌دهیم که از طریق رابطه زیر قابل محاسبه است

$$\beta(A) = A \text{ and } (A \ominus B)^c$$

$$\beta(A) = A \text{ xor } (A \ominus B)$$

$$\beta(A) = A \text{ xor } (A \oplus B)$$

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



استخراج مرز

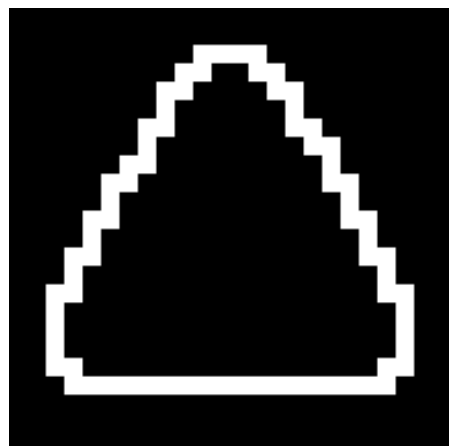


1	1	1
1	1	1
1	1	1

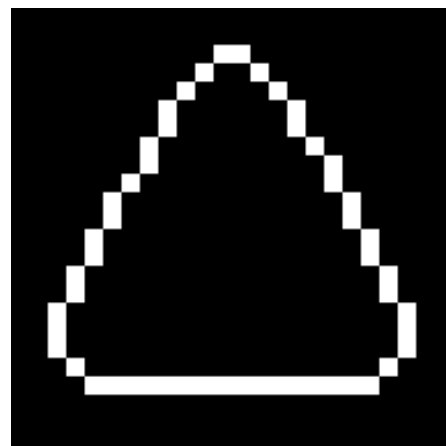
B_1

0	1	0
1	1	1
0	1	0

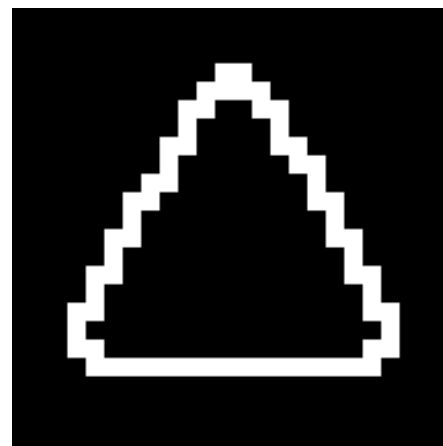
B_2



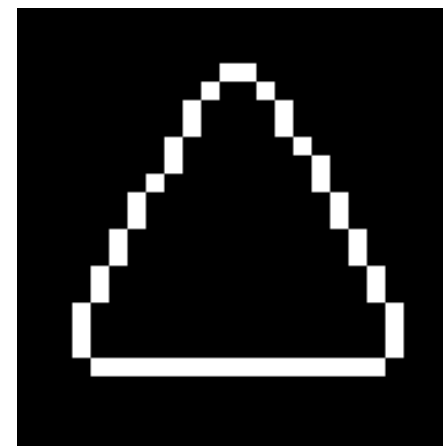
$A \text{ xor } (A \oplus B_1)$



$A \text{ xor } (A \oplus B_2)$



$A \text{ xor } (A \ominus B_1)$



$A \text{ xor } (A \ominus B_2)$