بسمه‌تعالي

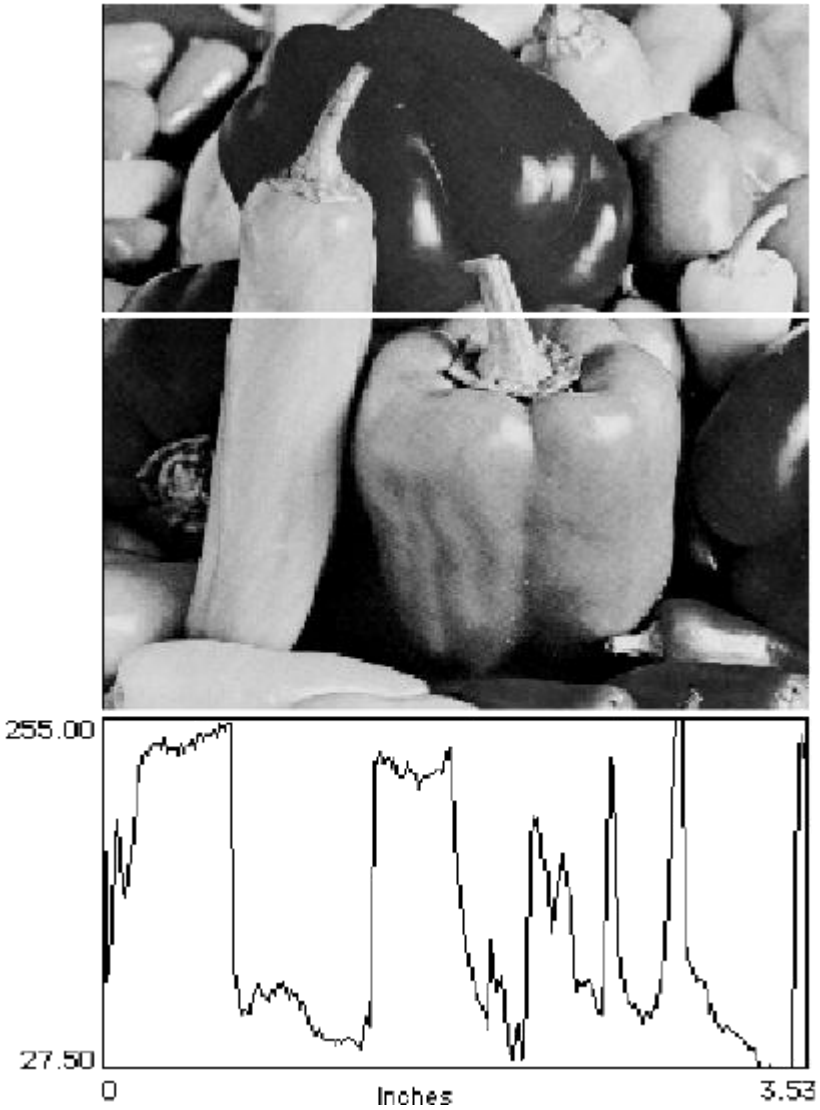
پردازش تصویر

Edge Detection

کشف لبه (Edge Detection)

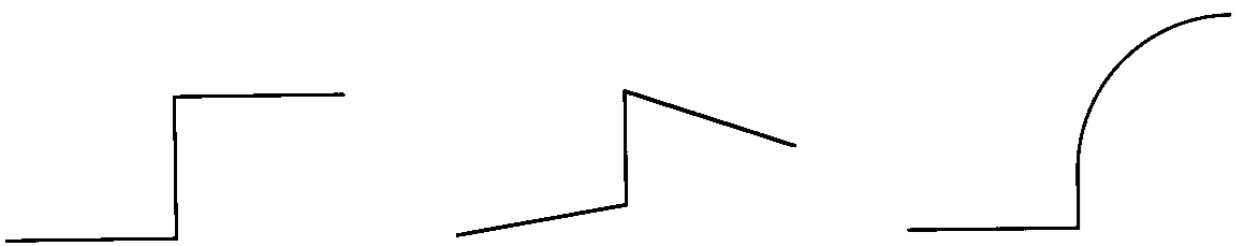
- انسان مي تواند بسياري از اشياء را از روي تصوير خطوط آنها شناسايي كند
  - مثال: تصاوير كارتوني
- سيستم بينايي انسان قبل از بازشناسي رنگ يا شدت روشنايي، نوعي كشف لبه انجام مي دهد.
- بنابراين انجام كشف لبه قبل از تفسير تصاوير در سيستمهاي خودكار منطقي به نظر مي رسد.
- انجام عمليات كشف لبه، پردازش مهمي در بسياري از سيستمهاي بينايي مصنوعي محسوب مي شود.

کشف لبه (Edge Detection)

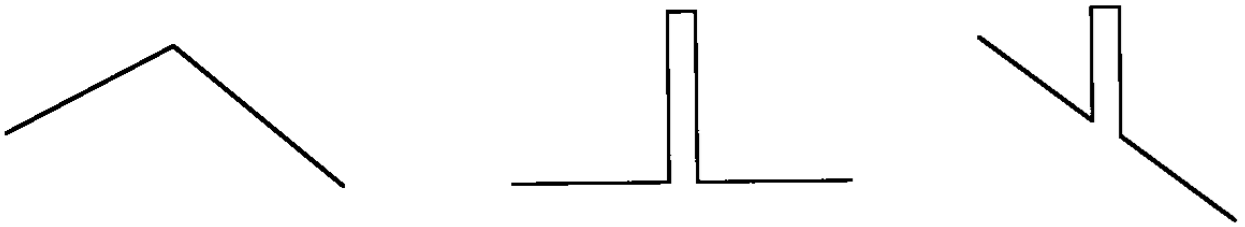


- **لبه چيست؟** به تغييرات ناگهاني محلي در روشنايي تصوير لبه گفته مي شود.
- لبه ها عموماً در مرز دو ناحيه اتفاق مي افتند.
- لبه ها ويژگي مهمي در پردازش تصوير هستند.
- تغييرات ناشي از نويز لبه نيستند
  - تصاوير واقعي نويزي هستند.
  - کشف لبه در تصاوير نويزي کار دشواري است.
- تغييرات ناشي از سايه نيز لبه نيست.

لبه هاي ايده آل



Step Edges



Roof Edge

Line Edges

لبه هاي نويزي

كشف لبه

- تعاريف
  - يك نقطة لبه، نقطه اي است كه در محل آن تغييرات محلي روشنايي وجود دارد.
  - يك آشكارساز لبه، الگوريتمي است كه لبه ها را در يك تصوير محاسبه مي كند.
  - يك كانتور، يك مسير بسته ناشي از مرزهاي يك شي است
  - دنبال كردن لبه، فرايند جستجوي تصوير براي يافتن كانتورها است.

- فرايند كشف لبه با دو پديده مواجه است
  - لبه هاي كشف شده
    - لبه هاي واقعي كه اطلاعات مفيد تصوير بوده و منطبق بر مرزهاي تصوير است.
    - لبه هاي كاذب كه مطابق لبه هاي واقعي نبوده و ناشي از نويز است
  - لبه هاي گم شده
    - لينك كردن لبه ها مي تواند مشكل لبه هاي گم شده را ترميم كند
    - دنبال كردن لبه، به كمك لبه هاي كشف و ترميم شده، كانتورها را پيدا مي كند.

تعيين محل لبه

1- كشف قطعه لبه هاي خطي كوچك (خرده لبه)

2- گردآوري خرده لبه ها به منظور ايجاد لبه

آشكارسازي خرده لبه

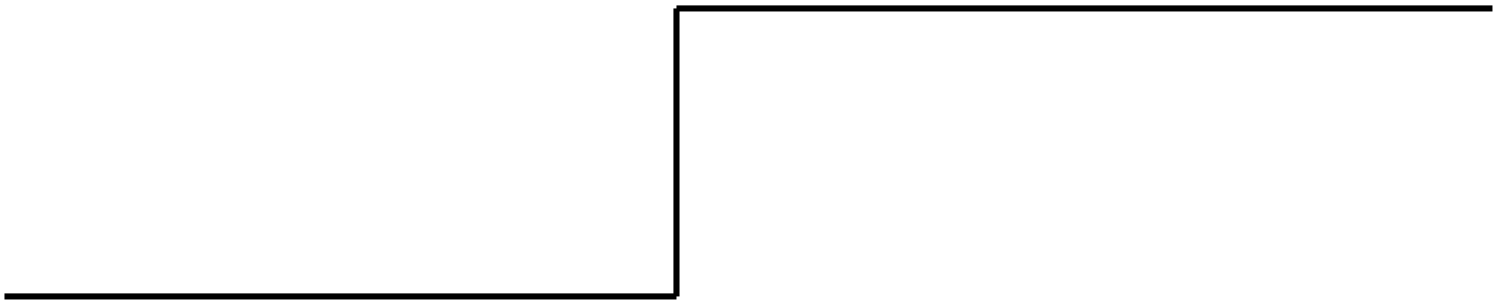
- با استفاده از عملگر تفاضل (از دو كليشة عمود بر هم استفاده ميكنند)
- با استفاده از كليشه هاي لبه (از چندين كليشه استفاده ميكنند)
- با انطباق مدلهاي پارامتريك

نويز مي تواند اثر منفي بر روي كشف لبه بگذارد.

كشف لبه به كمك تفاضل

لبه محلي است كه در آن تغييرات رخ مي دهد

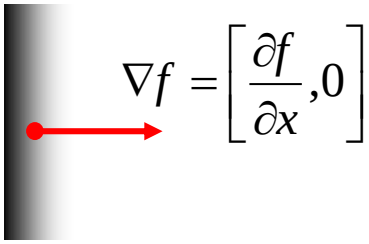
- اندازه گيري تغييرات مي تواند به كمك مشتق انجام شود.
- بيشترين تغييرات به معناي ماكزيمم شدن مشتق اول و صفر شدن مشتق دوم مي شود.



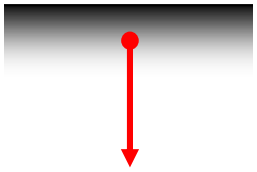
كشف لبه به كمك تفاضل

گراديان: برداري است كه حداكثر نرخ تغييرات روشنايي $f(x,y)$ را ارائه مي كند

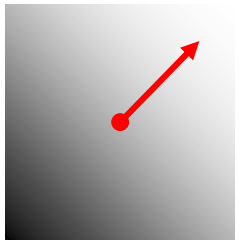
$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$



$$\nabla f = \left[\frac{\partial f}{\partial x}, 0\right]$$



$$\nabla f = \left[0, \frac{\partial f}{\partial y}\right]$$



$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$

کشف لبه به کمك تفاضل

گرادیان

$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$

جهت گرادیان:

$$\theta = \tan^{-1}\left(\frac{\partial f}{\partial x} \Big/ \frac{\partial f}{\partial y}\right)$$

این مقدار چه ارتباطي با جهت لبه دارد؟

شدت گرادیان با اندازه گرادیان بیان مي شود:

$$\|\nabla f\| = \sqrt{(\frac{\partial f}{\partial x})^2 + (\frac{\partial f}{\partial y})^2}$$

$$\|\nabla f\| \approx \left|\frac{\partial f}{\partial x}\right| + \left|\frac{\partial f}{\partial y}\right|$$

$$\|\nabla f\| \approx Max\left(\left|\frac{\partial f}{\partial x}\right|, \left|\frac{\partial f}{\partial y}\right|\right)$$

كشف لبه به كمك تفاضل

چگونه مي توان بر روي يك تصوير ديجيتال مشتق گرفت؟

- از تصوير ديجيتال، يك تصوير آنالوگ بسازيم و سپس از آن مشتق بگيريم
- از مشتق گسسته استفاده كنيم (Finite Difference)
- از دو كليشه متعامد استفاده شده و بر اساس آنها راستاي لبه مشخص ميشود.

مثلاً:

$$G_x \approx \Delta_x f(i,j) = f(i,j+1) - f(i,j)$$
$$G_y \approx \Delta_y f(i,j) = f(i,j) - f(i+1,j)$$

اين روابط متناظر با كانوالو كردن تصوير با ماسكهاي زير است:

$$G_x = \begin{bmatrix} -1 & 1 \end{bmatrix} \qquad G_y = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

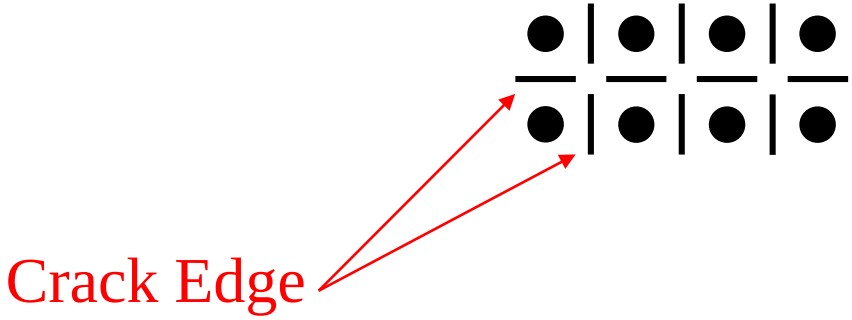
استخراج لبه به كمك تفاضل

- G_x مشتق را در نقطة $(i+1/2,j)$ بدست مي آورد
- G_y مشتق را در نقطة $(i,j+1/2)$ بدست مي آورد
- ماسك 2*2 براي بدست آوردن لبه در $(i+1/2,j+1/2)$

$$G_x = \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix} \qquad G_y = \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}$$

استخراج لبه به كمك تفاضل

به لبه هايي كه بين پيكسلها قرار گرفته Crack Edge گفته ميشود. در اين حال پيدا كردن لبه ها ساده است اما امتداد لبه ها تنها ميتواند افقي يا عمودي باشد



اپراتور Robert

- لبه رادر $(i+1/2,j+1/2)$ بدست مي آورد

$$G_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \qquad G_y = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

- براي بدست آوردن نرخ تغييرات در واحد طول بايستي مقادير G_x و G_y بر راديكال 2 تقسيم شود (محاسبة گراديان قطري انجام شده)

ايراد: حساسيت بالا به نويز به دليل استفاده از نقاط كم در تخمين مشتق

اپراتور Prewitt

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \qquad G_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

اپراتور Sobel

- بهترين و عمومي ترين اپراتور آشكارساز لبه است
- لبه را در **(*i,j*)** بدست مي آورد.
- همزمان عمل هموارسازي نيز انجام مي دهد.
- تاكيد بر نقاط نزديك به **(*i,j*)** است.

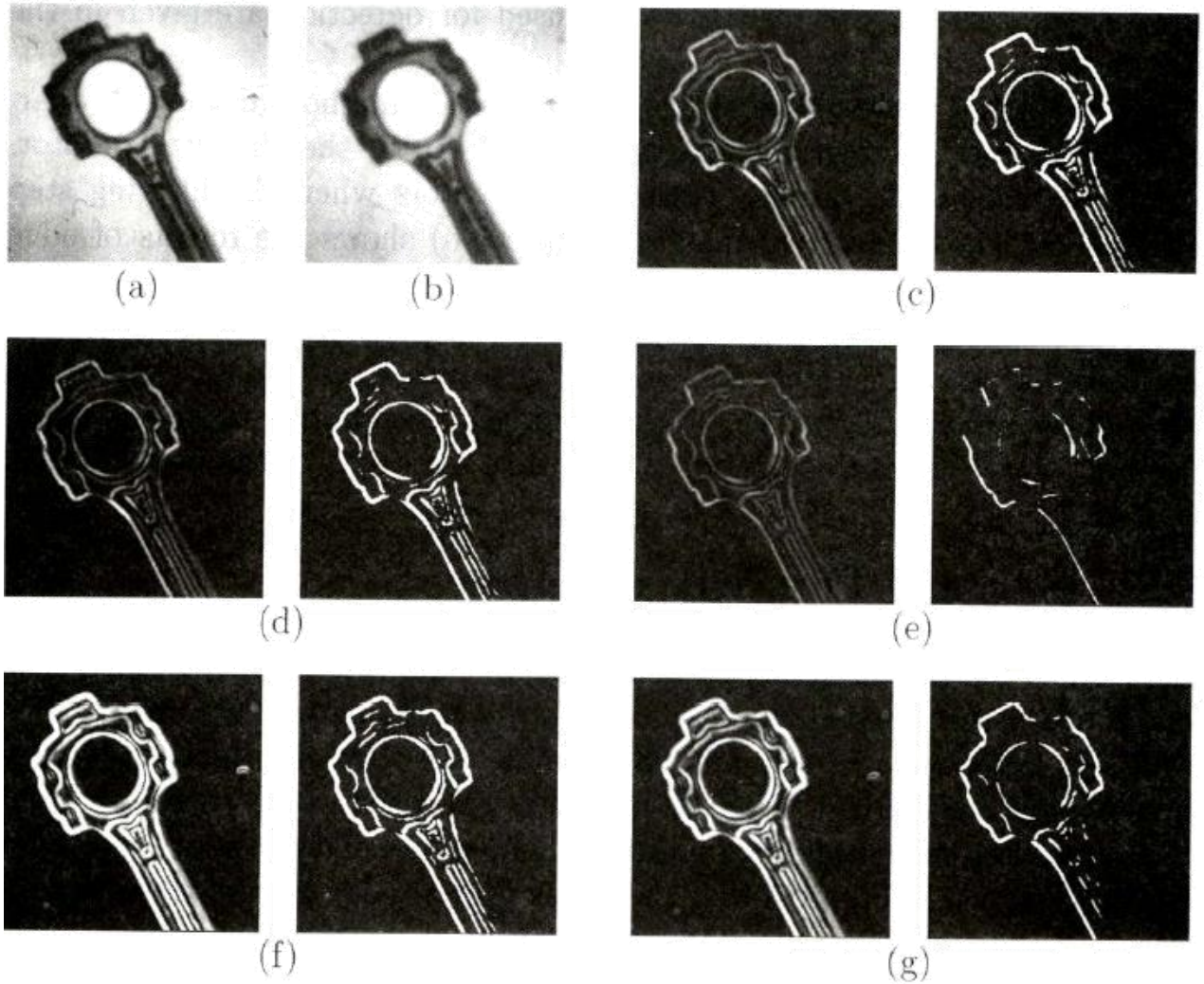
$$G_x = \frac{1}{8}\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad G_y = \frac{1}{8}\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

- عموماً از ضريب **1/8** صرفنظر مي شود، ولي براي محاسبه مقدار درست گراديان اين ضريب بايستي وجود داشته باشد.

اپراتور Robinson

$$G_x = \begin{bmatrix} -1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & 1 & 1 \end{bmatrix} \qquad G_y = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{bmatrix}$$

آشکارسازي لبه با پيش پردازش هموارسازي



a. Original

b. Smoothed image

c. Gradient with 1x2, 2x1 masks (T=32)

d. Gradient with 2x2 masks (T=64)

e. Roberts operator (T=64)

f. Sobel operator (T=128)

g. Prewit operator (T=128)

روشهاي مبتني بر كليشه هاي لبه

- در اين روش از تعدادي (4 يا 8) كليشه ميشود
  - هر كدام از اين كليشه ها لبه را در يك راستا مشخص مي كند
  - نتايج اعمال كليشه ها با يكديگر مقايسه مي شوند
  - كليشه اي كه بيشترين مقدار را ارائه كند، دامنه و راستاي لبه را مشخص خواهد كرد
  - به Thresholding هنوز هم احتياج است.

روشهاي مبتني بر كليشه هاي لبه: اپراتور Kirsch

$$G_1=\begin{bmatrix}-5 & 3 & 3\\ -5 & 0 & 3\\ -5 & 3 & 3\end{bmatrix} \qquad G_2=\begin{bmatrix}3 & 3 & 3\\ 3 & 0 & 3\\ -5 & -5 & -5\end{bmatrix} \qquad G_3=\begin{bmatrix}3 & 3 & 3\\ -5 & 0 & 3\\ -5 & -5 & 3\end{bmatrix}$$

A_0	A_1	A_2
A_7	$f(j,k)$	A_3
A_6	A_5	A_4

اندیس A ها بصورت **Modula-8** است.

$$S_i = A_i + A_{i+1} + A_{i+2}$$

$$T_i = A_{i+3} + A_{i+4} + A_{i+5} + A_{i+6} + A_{i+7}$$

$$f(j,k) = Max\left\{1, \underset{i=0}{\overset{7}{Max}}\left[\left|3T_i - 5S_i\right|\right]\right\}$$

اپراتور Kirsch

- با اعمال 8 اپراتور مختلف، اپراتوري كه پاسخ آن حداكثر است تعيين كنندة راستاي لبه است.
- يك كد 3 بيتي مي توان براي مشخص كردن امتداد لبه در نظر گرفت.
- استفاده از پنجرة بزرگتر
  - تعداد جهتهاي قابل تفكيك بيشتر ميشود
  - اثر نويز كاهش مي يابد
  - لبه هاي نزديك به هم قابل تشخيص نيستند
- با استفاده از ديگر اپراتورها نظير Sobel نيز ميتوان كليشه هاي لبه را ايجاد كرد.

ايجاد كليشه هاي لبه به كمك اپراتور Sobel

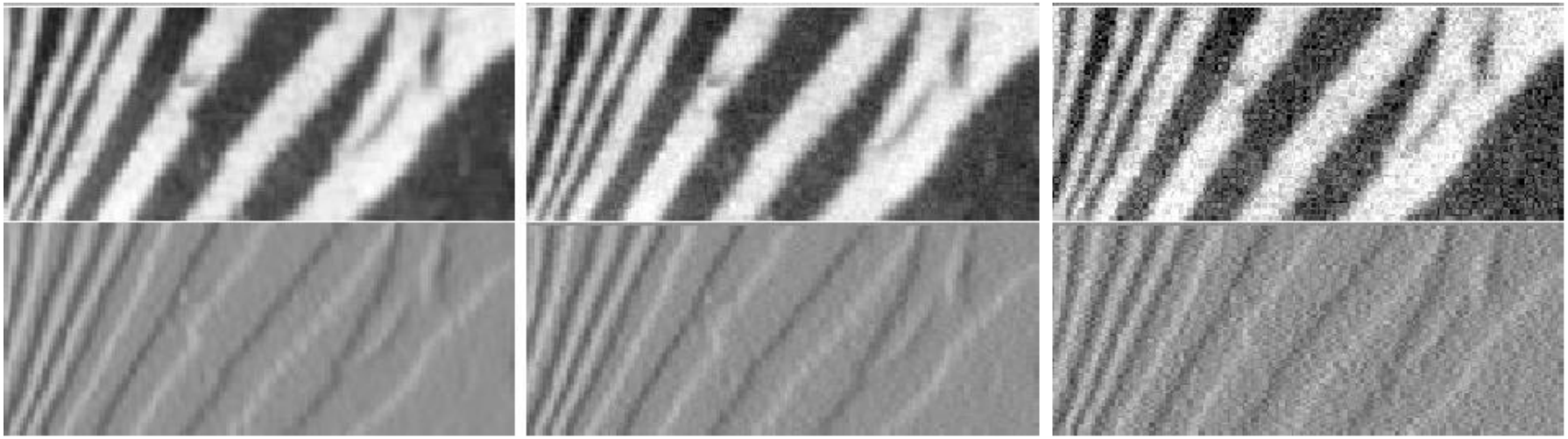
$$G_0=\begin{bmatrix}1&2&1\\0&0&0\\-1&-2&-1\end{bmatrix}\quad G_1=\begin{bmatrix}2&1&0\\1&0&-1\\0&-1&-2\end{bmatrix}\quad G_2=\begin{bmatrix}1&0&-1\\2&0&-2\\1&0&-1\end{bmatrix}\quad G_3=\begin{bmatrix}0&-1&-2\\1&0&-1\\2&1&0\end{bmatrix}$$

$$G_4=\begin{bmatrix}-1&-2&-1\\0&0&0\\1&2&1\end{bmatrix}\quad G_5=\begin{bmatrix}-2&-1&0\\-1&0&1\\0&1&2\end{bmatrix}\quad G_6=\begin{bmatrix}-1&0&1\\-2&0&2\\-1&0&1\end{bmatrix}\quad G_7=\begin{bmatrix}0&1&2\\-1&0&1\\-2&-1&0\end{bmatrix}$$

- اپراتورهاي Sobel و Kirsch عموماً بهتر از ديگر اپراتورها عمل مي كنند
- اپراتور Kirsch به تغييرات كوچك گراديان بسيار حساس است.
- اپراتورهاي با پنجرة كوچكتر، بدليل حجم محاسبات كمتر، بيشتر استفاده ميشوند.

اثر نویز بر کشف لبه

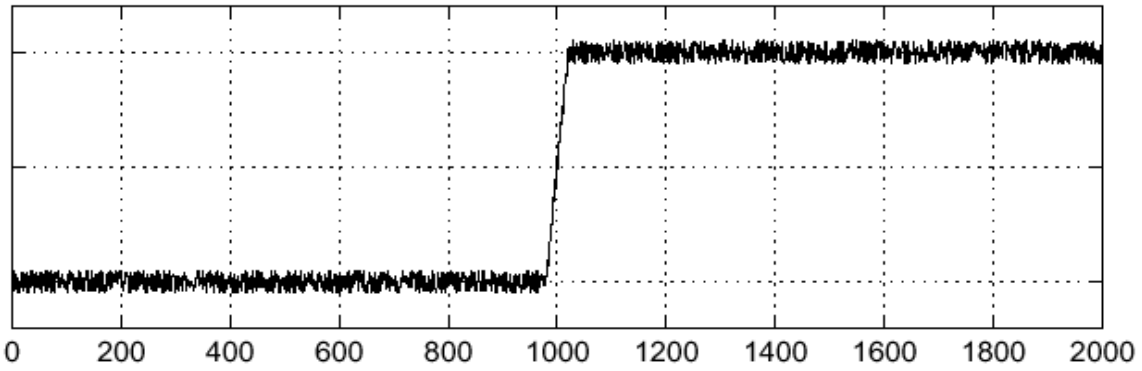
با توجه به تعریف لبه، نویز نیز مي تواند به عنوان خرده لبه آشکار شود.



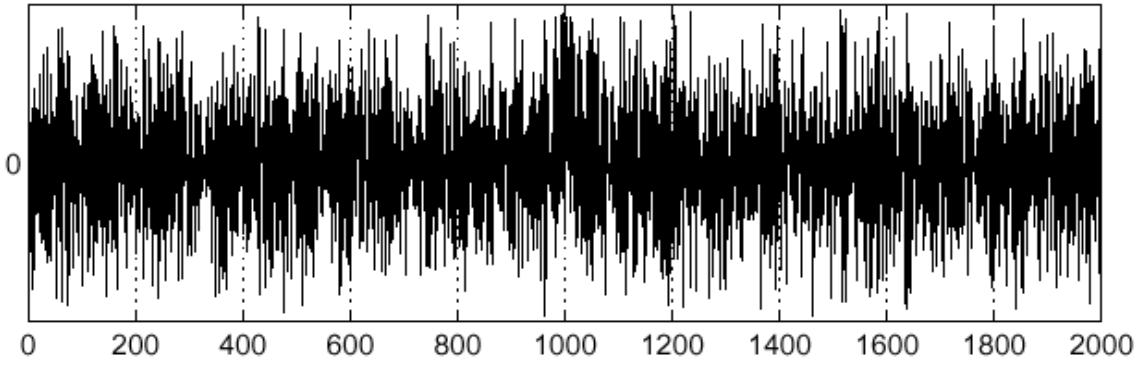
اثر نویز بر کشف لبه

یك سطر از یك تصویر را در نظر بگیرید:

$f(x)$



$\frac{d}{dx} f(x)$

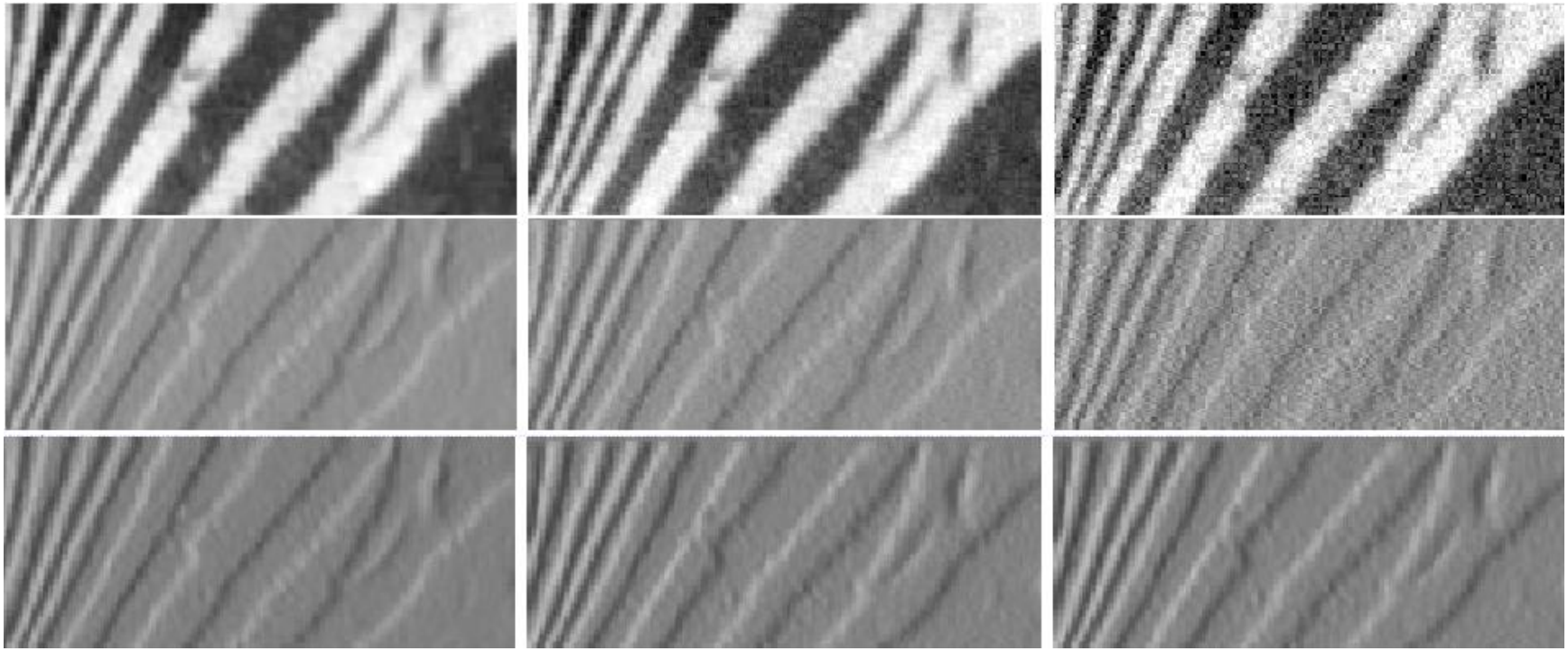


اثر نويز بر كشف لبه

نويز عموماً به كمك يك تابع چگالي احتمال مدل مي شود.

- **يك روش عمومي براي مدل كردن نويز، استفاده از يك مدل نويز گوسي جمع شوندة ايستا است. در اين حال عموماً ميانگين نويز صفر در نظر گرفته مي شود. بنابراين تنها پارامتر مدل نويز، انحراف معيار آن خواهد بود.**
- **مزيت استفاده از اين مدل، سهولت تخمين پاسخ فيلترها يا عملگرها به آن است.**
- **براي كاهش اثر نويز از هموارسازي استفاده مي شود.**

اثر همواسازي بر استخراج لبه

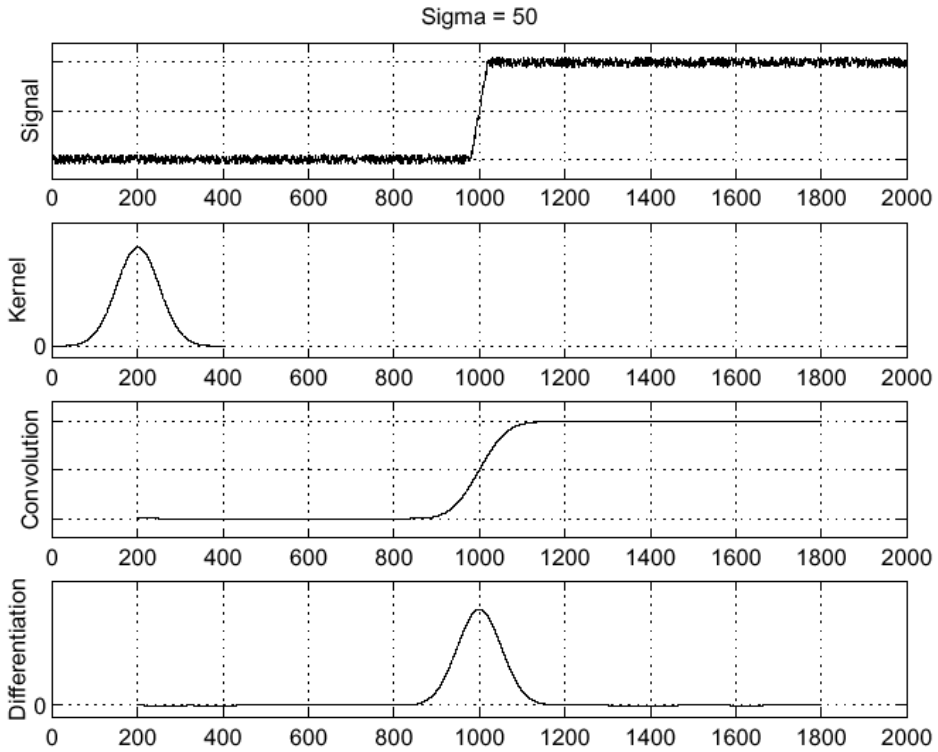


اثر هموارسازي بر استخراج لبه

f

h

$f * h$



$$\frac{\partial}{\partial x}(f * h)$$

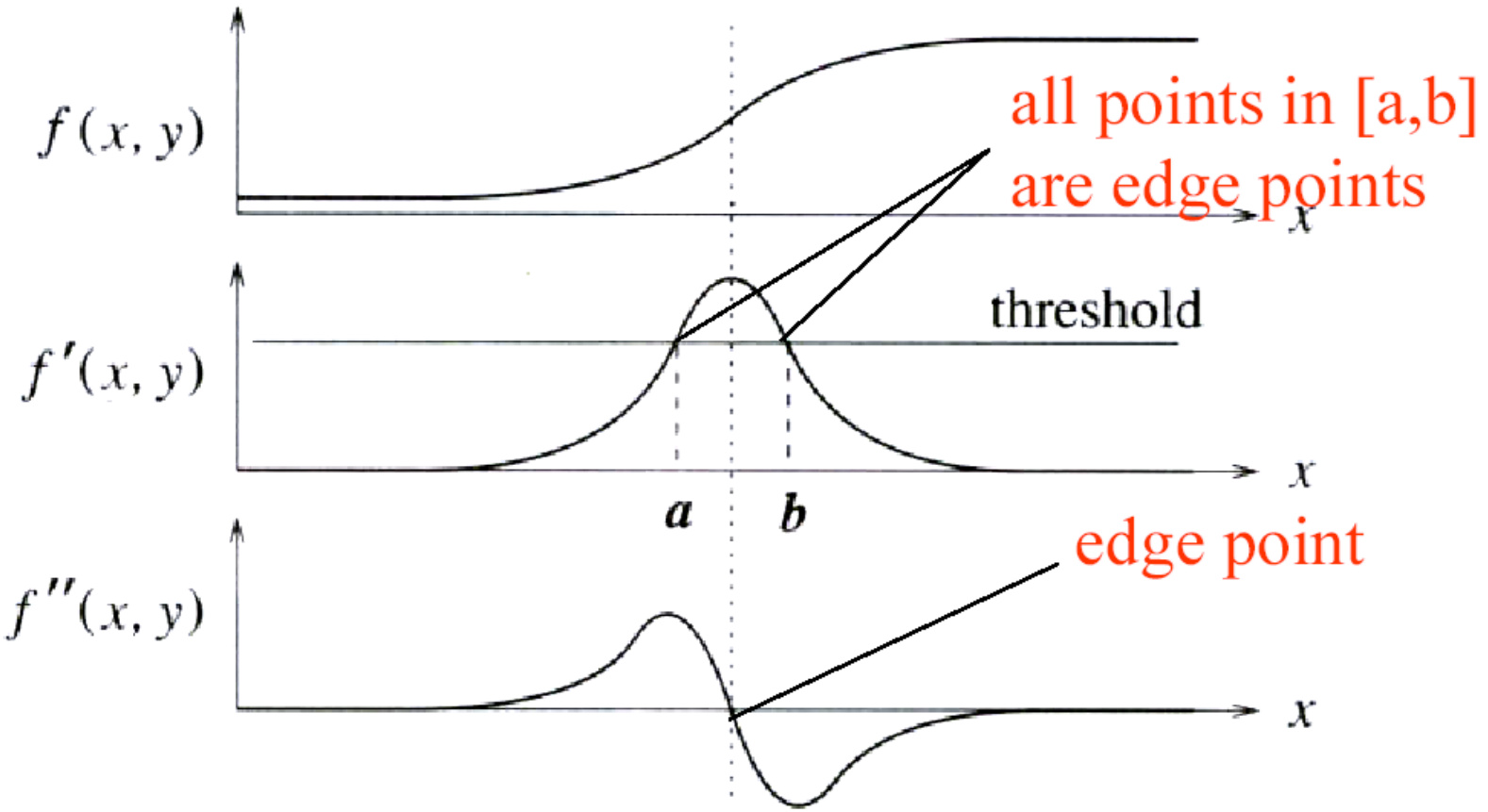
استفاده از مشتق مرتبة دوم

- ماکزیمم شدن مشتق اول (لبه) متناظر با صفر شدن مشتق دوم است.
- اشکالات استفاده از مشتق دوم:
  - حساسیت بالاتر به نویز
  - از بین رفتن اطلاعات جهت (مشتق دوم ایزوتروپیك است)
- بنابراین عموماً از ترکیب مشتق اول و دوم استفاده مي شود.
- روش ترکیبي:

يك نقطه لبه است اگر:

$$\nabla^2 f(x,y) \approx 0 \quad \& \quad \left|\nabla f(x,y)\right| > Threshold$$

استفاده از مشتق مرتبة اول و دوم



مشتق مرتبة دوم (لاپلاسين)

- **مشتق دوم دو بعدي:**

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

- **تقريب ديجيتالي:**

$$\nabla^2 f(i,j) \approx f(i+1,j) + f(i-1,j) + f(i,j+1) + f(i,j-1) - 4f(i,j)$$

- **كليشة لاپلاسين**

$$\nabla^2 \approx \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

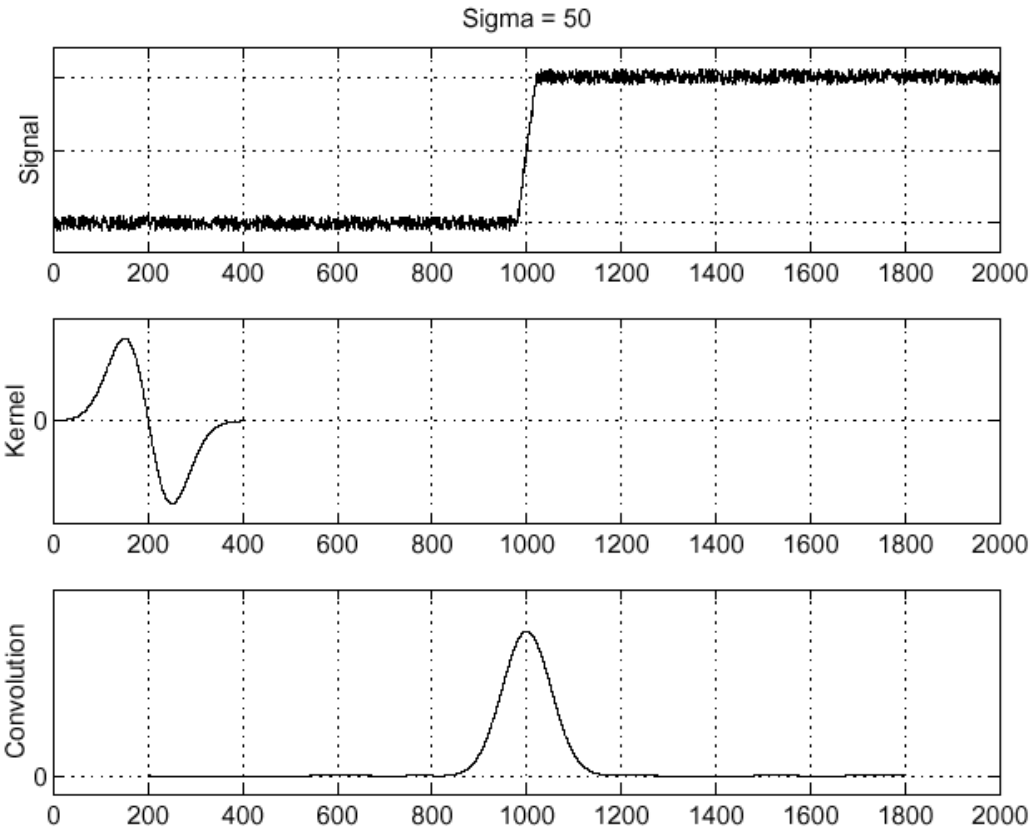
تئوري مشتق كانولوشن

$$\frac{\partial}{\partial x}(h * f) = (\frac{\partial}{\partial x} h) * f$$

f

$\frac{\partial}{\partial x} h$

$(\frac{\partial}{\partial x} h) * f$



اين مساله باعث
صرفه جويي در يك
عمل مي شود

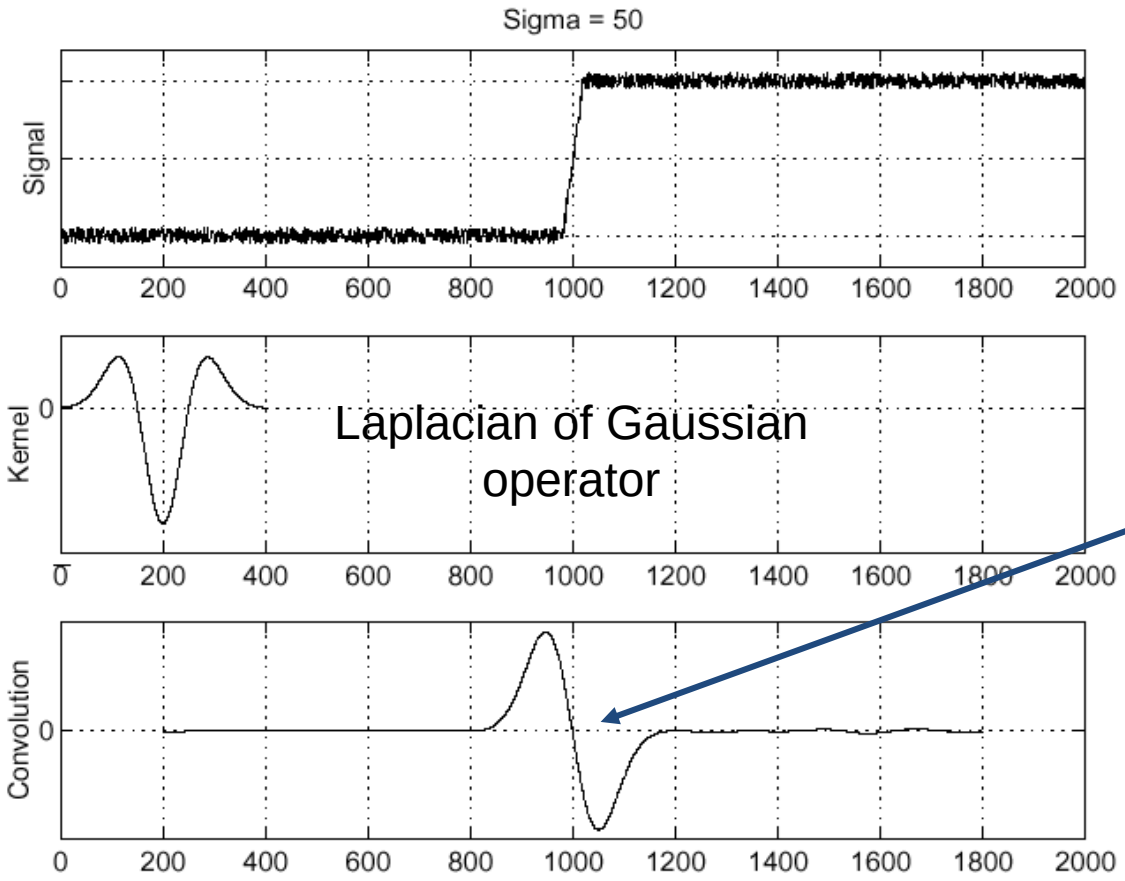
لاپلاسین گوسین LoG

$$\frac{\partial^2}{\partial x^2}(h * f)$$

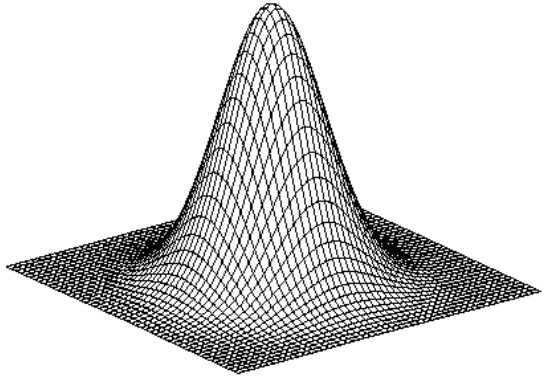
f

$$\frac{\partial^2}{\partial x^2} h$$

$$(\frac{\partial^2}{\partial x^2} h) * f$$

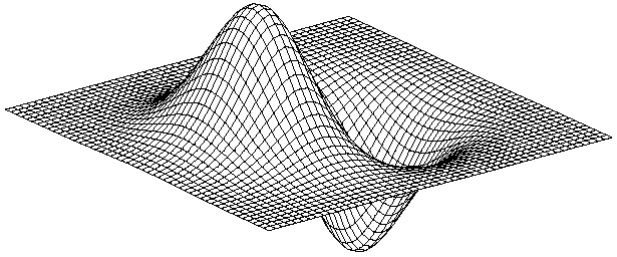


لاپلاسین گوسین LoG



Gaussian

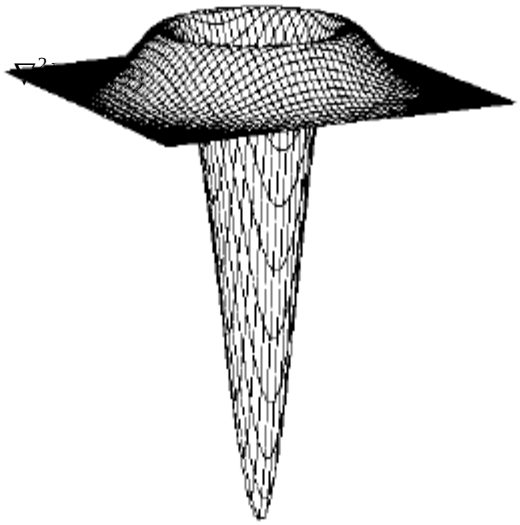
$$h_\sigma(u,v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2\sigma^2}}$$



Derivative of Gaussian

$$\frac{\partial}{\partial x} h_\sigma(u,v)$$

Laplacian of Gaussian



لاپلاسین گوسین LoG

تخمین دیجیتالي فیلتر LoG

5 × 5 Laplacian of Gaussian mask

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

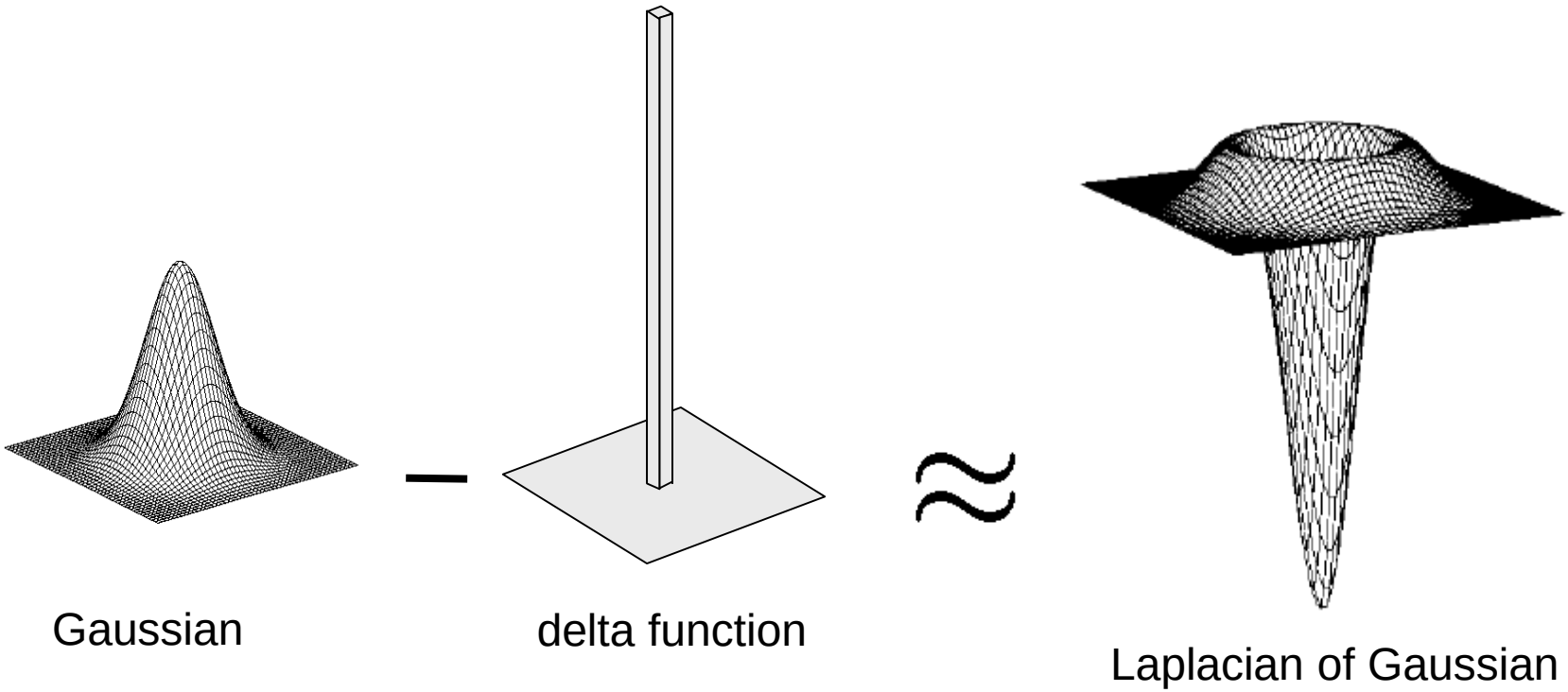
17 × 17 Laplacian of Gaussian mask

0	0	0	0	0	0	−1	−1	−1	−1	−1	0	0	0	0	0	0
0	0	0	0	−1	−1	−1	−1	−1	−1	−1	−1	−1	0	0	0	0
0	0	−1	−1	−1	−2	−3	−3	−3	−3	−3	−2	−1	−1	−1	0	0
0	0	−1	−1	−2	−3	−3	−3	−3	−3	−3	−3	−2	−1	−1	0	0
0	−1	−1	−2	−3	−3	−3	−2	−3	−2	−3	−3	−3	−2	−1	−1	0
0	−1	−2	−3	−3	−3	0	2	4	2	0	−3	−3	−3	−2	−1	0
−1	−1	−3	−3	−3	0	4	10	12	10	4	0	−3	−3	−3	−1	−1
−1	−1	−3	−3	−2	2	10	18	21	18	10	2	−2	−3	−3	−1	−1
−1	−1	−3	−3	−3	4	12	21	24	21	12	4	−3	−3	−3	−1	−1
−1	−1	−3	−3	−2	2	10	18	21	18	10	2	−2	−3	−3	−1	−1
−1	−1	−3	−3	−3	0	4	10	12	10	4	0	−3	−3	−3	−1	−1
0	−1	−2	−3	−3	−3	0	2	4	2	0	−3	−3	−3	−2	−1	0
0	−1	−1	−2	−3	−3	−3	−2	−3	−2	−3	−3	−3	−2	−1	−1	0
0	0	−1	−1	−2	−3	−3	−3	−3	−3	−3	−3	−2	−1	−1	0	0
0	0	−1	−1	−1	−2	−3	−3	−3	−3	−3	−2	−1	−1	−1	0	0
0	0	0	0	−1	−1	−1	−1	−1	−1	−1	−1	−1	0	0	0	0
0	0	0	0	0	0	−1	−1	−1	−1	−1	0	0	0	0	0	0

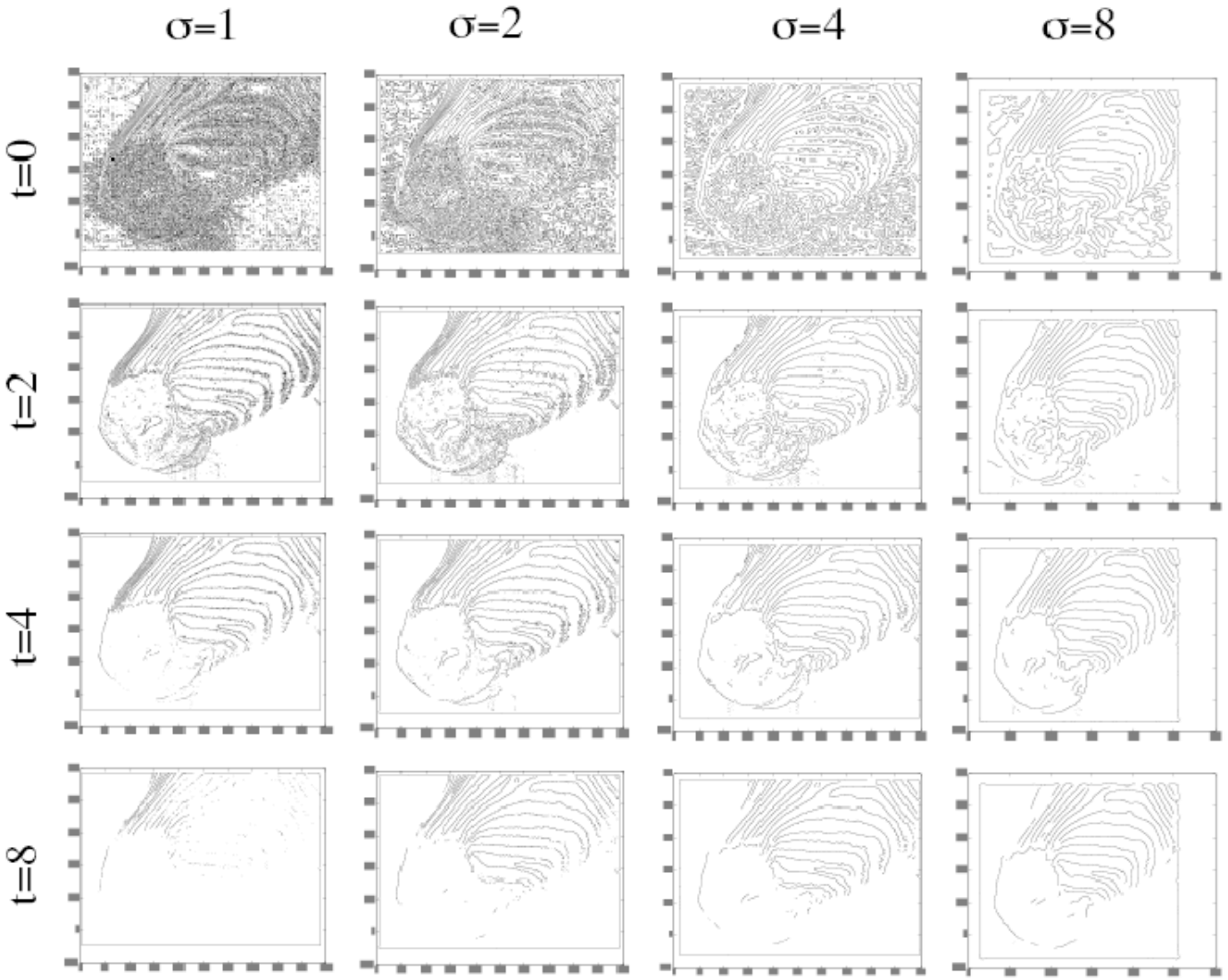
هموارسازي در LoG

- درجة هموارسازي را σ تعيين مي كند
  - σ بزرگتر موجب مي شود نويز بهتر فيلتر شود، ولي باعث تار شدن و جابجايي لبه مي شود
  - σ كوچكتر باعث ايجاد لبه هاي كاذب مي شود، ولي محل لبه ها بهتر تعيين مي شود.
- بهترين مقدار براي اندازه فيلتر نامشخص است
- LoG بخوبي توسط DoG تخمين زده مي شود. DoG كانوالو تفاضل دو ماسك گوسين با σ كاملاً متفاوت با تصوير است.

آشکارسازي لبه به کمك تفريق



اثر σ در هموارسازي LoG



آشکارسازي لبه به کمك تفريق

original

smoothed (5x5 Gaussian)

smoothed – original
(scaled by 4, offset +128)

مراحل آشكارسازي لبه

- **فيلترينگ:** به منظور كاهش نويز و افزايش كارايي كشف لبه
  - مصالحه بين شدت لبه و كاهش نويز انجام مي شود
  - فيلترها لبه ها را نيز هموار مي كنند
- **بهبود لبه** به كمك اپراتور گراديان
- **كشف لبه:** نگه داشتن لبه هاي واقعي و حذف لبه هاي كاذب
  - نگه داشتن لبه هاي قوي
  - با استفاده از يك مقدار آستانه
- **تعيين مكان لبه:** محاسبة مكان و راستاي لبه

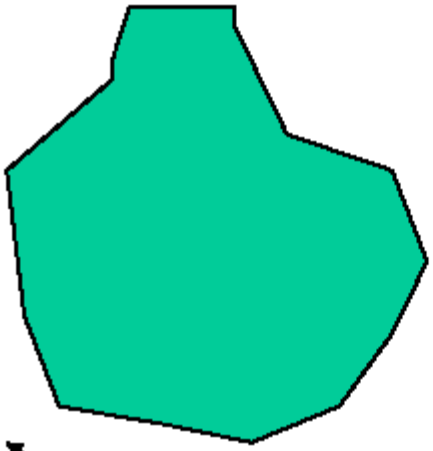
مراحل آشكارسازي لبه

- **هموارسازي:**
  - **استفاده از درجة بالاي هموارسازي (كليشه هاي بزرگتر يا استفاده از چند بار هموارسازي) نويز را حذف مي كند اما باعث تار شدن لبه ها نيز مي شود**
  - **استفاده از فيلترهاي با درجة پايين نويز را باقي مي گذارد.**

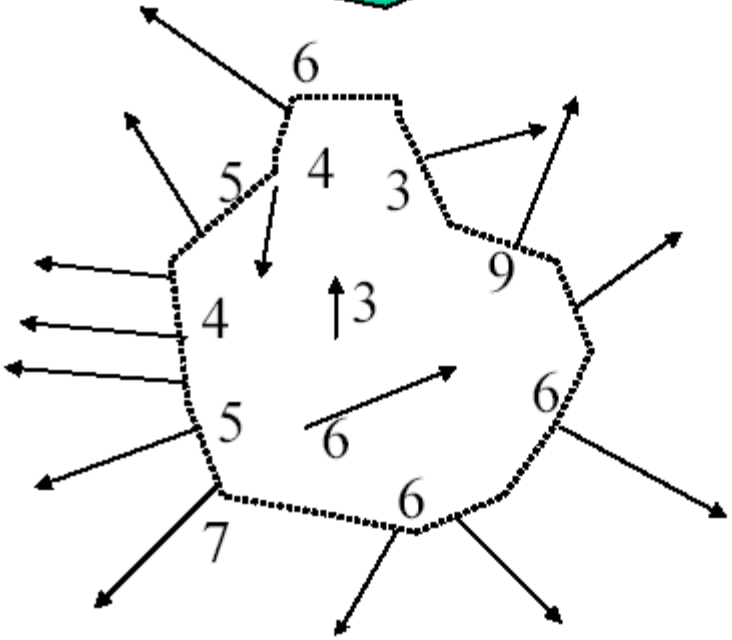
در هموارسازي بايستي يك مصالحه (Trade-off) بين آشكارسازي و دقت تعيين محل لبه صورت گيرد

مراحل آشکارسازي لبه

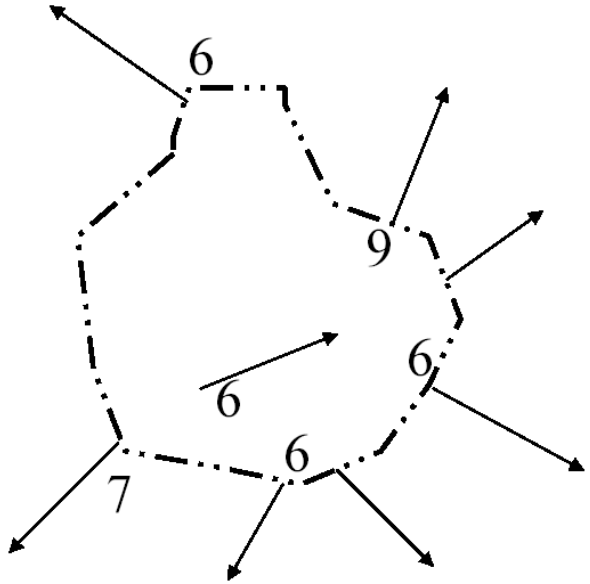
Thresholding



original image



gradient



T=5 ﺑﺎ ﺁﺳﺘﺎﻧﻪ

- ﻛﻪ ﺁﺳﺘﺎﻧﻪﮔﺬﺍﺭﻱ ﺑﺎ ﺁﺳﺘﺎﻧﻪ ﺑﺎﻻ ﺗﺮﻙ ﺑﻴﻦ ﻧﻮﺍﺣﻲ ﻣﻬﻢ ﻭ ﻏﻴﺮ ﻣﻬﻢ ﺗﻔﺎﻭﺕ ﻗﺎﺋﻞ ﺷﻮﺩ
- ﻛﻪ ﺁﺳﺘﺎﻧﻪﮔﺬﺍﺭﻱ ﺑﺎ ﺁﺳﺘﺎﻧﻪ ﭘﺎﻳﻴﻦ ﻧﻮﻳﺰ ﺭﺍ ﺣﺬﻑ ﻧﻤﻲﻛﻨﺪ ﻭ ﻟﺒﻪﻫﺎﻱ ﻛﺎﺫﺏ ﺗﻮﻟﻴﺪ ﻣﻲﻛﻨﺪ

مراحل آشکارسازي لبه

تعيين حد آستانه:

- آستانة ثابت
  - با استفاده از متوسط تصوير گراديان
  - با استفاده از هيستوگرام تصوير گراديان
- آستانۀ تطبيقی (Adaptive)
  - استفاده از متوسط محلي تصوير گراديان

اثر حد آستانه بر کشف لبه

آشكارساز بهينة لبه

- آشكارساز بايد براي لبة پله مانند آغشته به نويز، مناسب است.
- بهينه بودن آشكارسازمرتبط با سه معيار است:

1- آشكارسازي: لبه‌هاي مهم نبايستي گم شوند، ضمناً لبه‌هاي جعلي نيز نبايستي بوجود آيند.

2- صحت مكان لبه: فاصلة مكان واقعي لبه و مكان تعيين شده براي آن بايد حداقل باشد

3- تك پاسخي: پاسخ به يك لبه بايد يكسان باشد، يعني پديدة چند پاسخي بايستي حداقل شود.

الگوریتم آشکارسازی بهینه

original image (Lena)

الگوريتم كني (Canny)

norm of the gradient

الگوريتم كني (Canny)

thresholding

الگوريتم كني (Canny)

Thinning (non-maximal suppression)

نازك‌سازي خطوط (Line Thinning)

- Non-maximal Supression (قبل از آستانه‌گيري)
- استفاده از تعدادي كليشه (بعد از آستانه‌گيري)

$$\begin{bmatrix} 1 & x & 0 \\ 1 & 1 & 0 \\ x & 0 & 0 \end{bmatrix} \quad \begin{bmatrix} x & 1 & 1 \\ 0 & 1 & x \\ 0 & 0 & 0 \end{bmatrix} \quad \begin{bmatrix} x & 1 & x \\ 1 & 1 & x \\ x & x & 0 \end{bmatrix} \quad \begin{bmatrix} x & 1 & x \\ x & 1 & 0 \\ 0 & x & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 \\ x & 1 & 0 \\ 1 & 1 & x \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & x \\ 0 & 1 & 1 \\ 0 & x & 1 \end{bmatrix} \quad \begin{bmatrix} x & x & 0 \\ 1 & 1 & x \\ x & 1 & x \end{bmatrix} \quad \begin{bmatrix} 0 & x & x \\ x & 1 & 1 \\ x & 1 & x \end{bmatrix}$$

پس از انطباق Seed Pixel با صفر جايگزين ميشود

كشف خطوط در تصوير

كشف خطوط مي تواند به كمك تعدادي كليشه و رابطة زير انجام شود:

$$f(i,j) = \max[\,0, \max_k (g, h_k)\,]$$

$$h_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad h_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad h_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & -1 & 2 & -1 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$h_4 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad h_5 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad h_6 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$h_7 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad h_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & -1 & 0 \\ 0 & 2 & 2 & 2 & 0 \\ 0 & -1 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

پركردن لبه ها (Edge Filling)

- **استفاده از تعدادي كليشه (بعد از آستانه‌گيري)**

$$\begin{bmatrix}0&1&0\\0&0&0\\0&1&0\end{bmatrix}\quad\begin{bmatrix}0&0&0\\1&0&1\\0&0&0\end{bmatrix}\quad\begin{bmatrix}1&0&0\\0&0&0\\0&1&0\end{bmatrix}\quad\begin{bmatrix}0&0&1\\1&0&0\\0&0&0\end{bmatrix}$$

$$\begin{bmatrix}0&1&0\\0&0&0\\0&0&1\end{bmatrix}\quad\begin{bmatrix}0&0&0\\0&0&1\\1&0&0\end{bmatrix}\quad\begin{bmatrix}1&0&0\\0&0&0\\0&0&1\end{bmatrix}\quad\begin{bmatrix}0&0&1\\0&0&0\\1&0&0\end{bmatrix}$$

پس از انطباق Seed Pixel با يك جايگزين ميشود

روشهاي فوق تضميني براي يك پيكسل بودن عرض خطوط و يا پيوستگي آن ايجاد نمي كند

تناظر (Correspondence Problem)

- تناظر كاربردهاي متعددي دارد مانند تخمين حركت و تخمين عمق

- در حالت عمومي براي بدست آوردن تناظر بين نقاط دو تصوير، تناظر هر پيكسل با كليه نقاط تصوير ديگر الزامي است كه البته اين كار با محاسباتي بسيار زيادي را ايجاد مي‌كند
- كاهش محاسبات مي‌تواند به كمك ايجاد تناظر بين نقاط ويژه (مانند گوشه‌ها) حاصل شود

کشف گوشه (Corner Detection)

- آشکارساز **Morvaec**

$$MO(i,j)=\frac{1}{8}\sum_{k=i-1}^{i+1}\sum_{l=j-1}^{j+1}\left|g(k,l)-g(i,j)\right|$$

- با استفاده از کلیشه ها

وفقي سازي پردازش

- وفقي سازي ضرايب فيلتر

- وفقي سازي پيكسلهاي دخيل در فيلترينگ

(Adaptive Neighborhood Pre-processing)

- ارزيابي 8 همسايه براي تعيين تعلق به همسايگي

$$\left| f(k,l) - f(i,j) \right| \le T_1$$

Additive Tolerance Interval

$$\frac{\left| f(k,l) - f(i,j) \right|}{f(i,j)} \le T_2$$

Multiplicative Tolerance Interval

- انواع همسايگي:
  - **Foreground**
  - **Background**