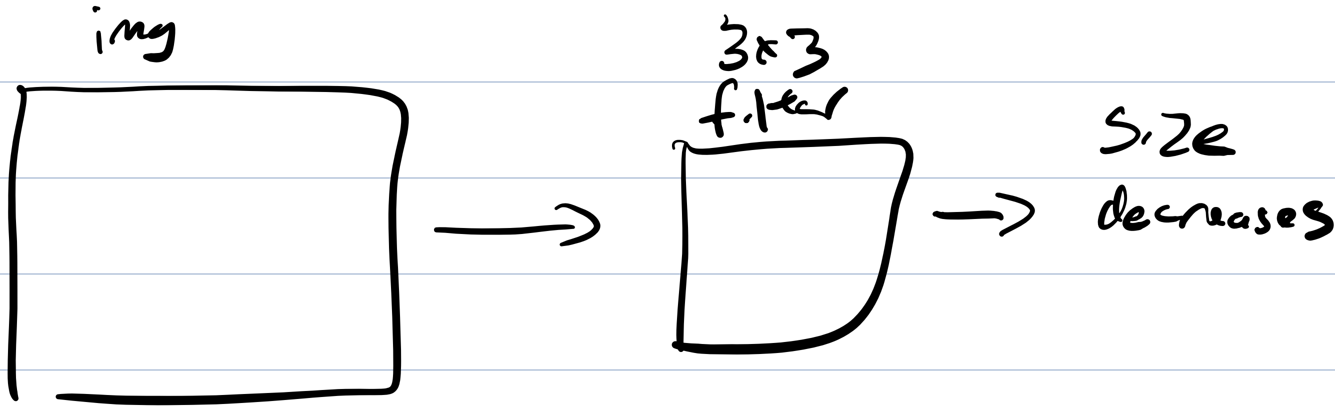
Filtering Cont.

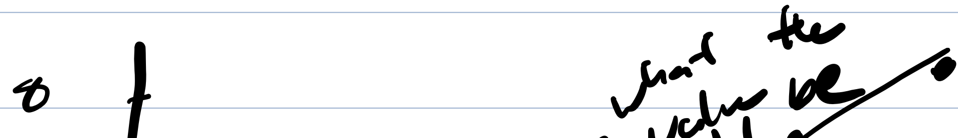
linear filter

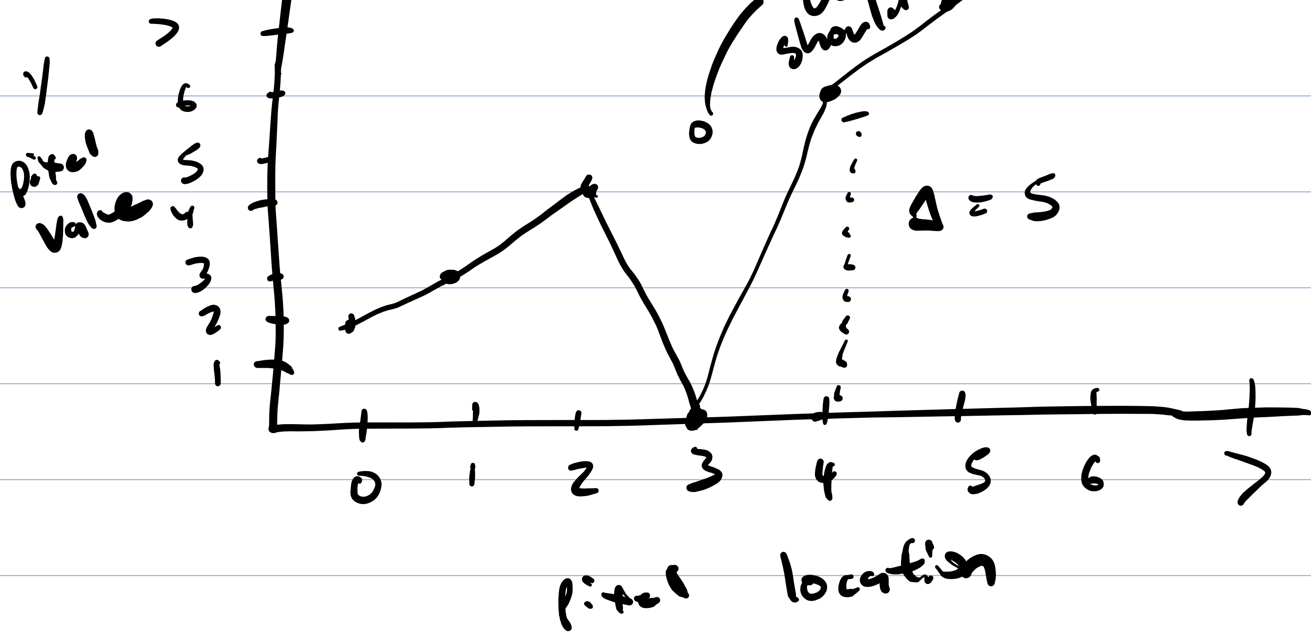
$$\overset{3\times3}{\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}} \quad \text{or} \quad \overset{3\times3}{\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}}$$



"Smoothing an image"

pixel values = [2, 3, 4, 0, 6, 7, 8]





filter = [1 0 1] $\frac{1}{2}$

x = [2, 3, 4, 0, 6, 7, 8]

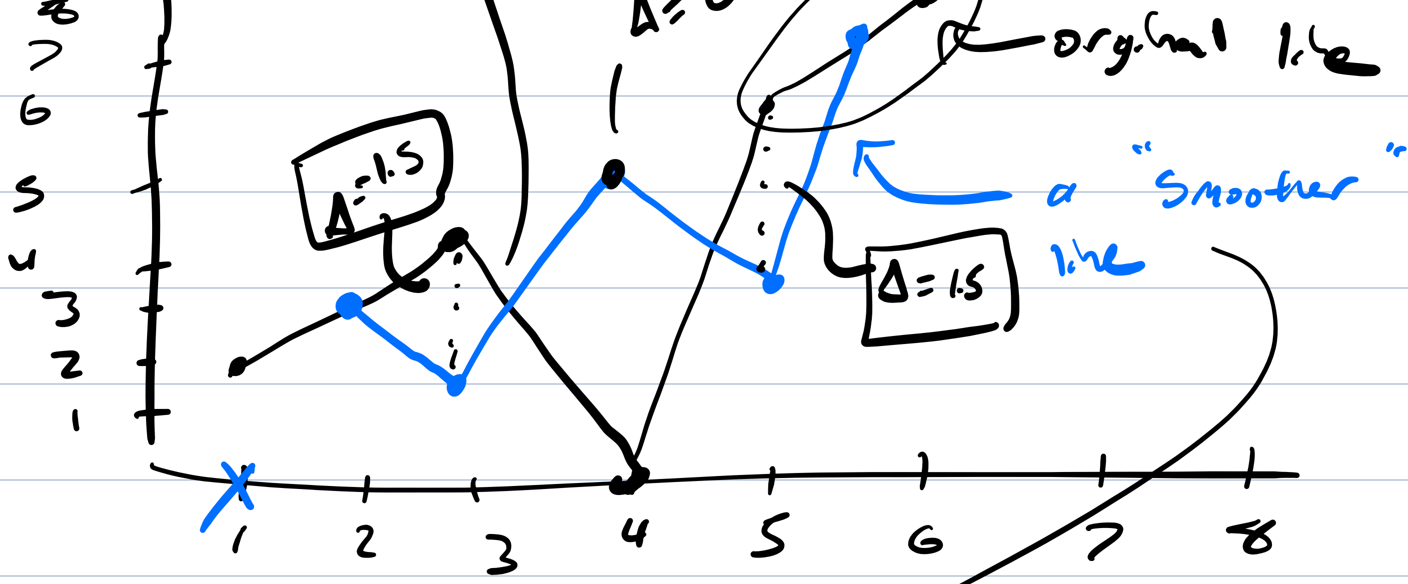
filter

Value 1 = [1, 0, 1] $\frac{1}{2}$

$$\frac{(2\cdot 1) + (3\cdot 0) + (4\cdot 1)}{2}$$

new Values = [X, 3, 1.5, 5, 3.5, 7, X]

$$\text{Value2} = \frac{(3\cdot 1) + \cancel{(4\cdot 0)} + \cancel{(0\cdot 1)}}{2} = 1.5$$



low pass filter

- removes noise /
- reduces high frequency content.
- only low frequency remain

$[1, 0, 1]\frac{1}{2}$

$[1, 1, 0, 1, 1]\frac{1}{4}$ — larger the filter the smoother / blurry the image will be

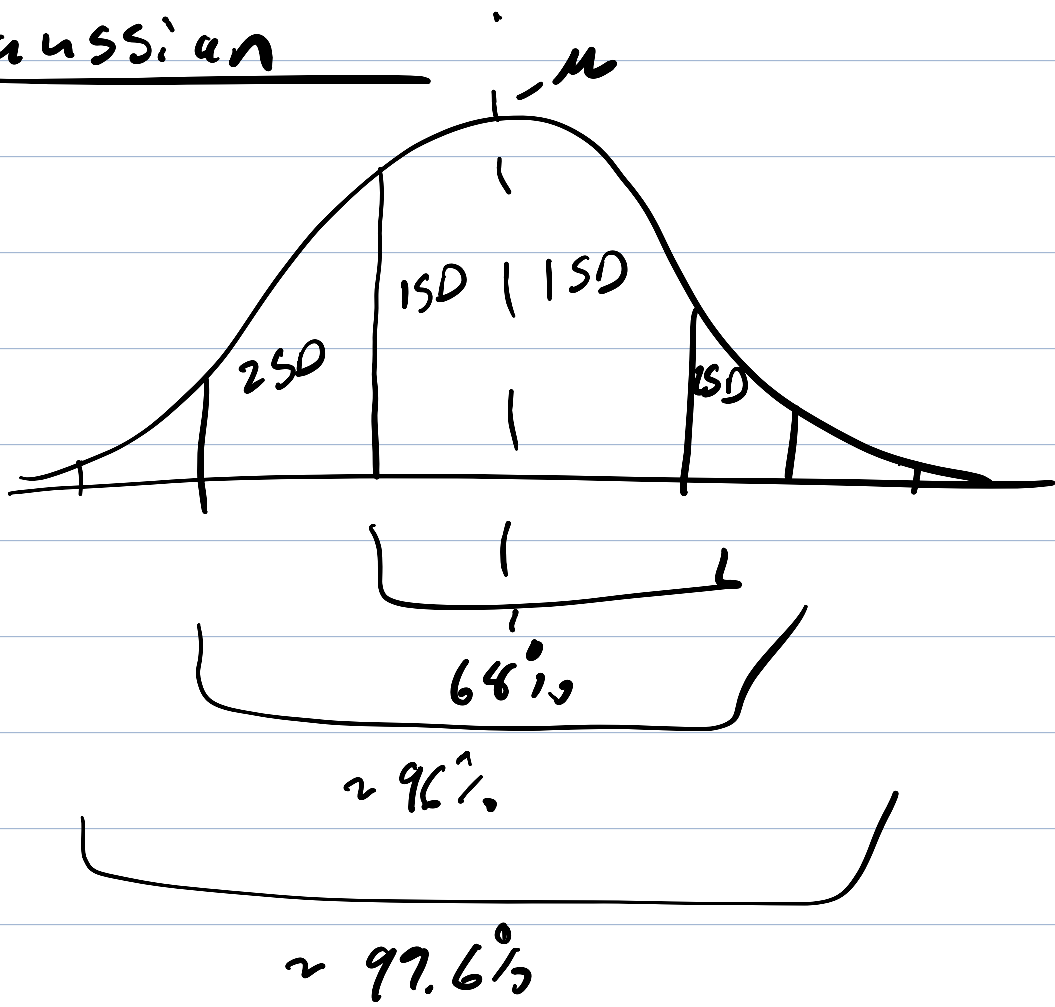
Other blurring filters

3x3

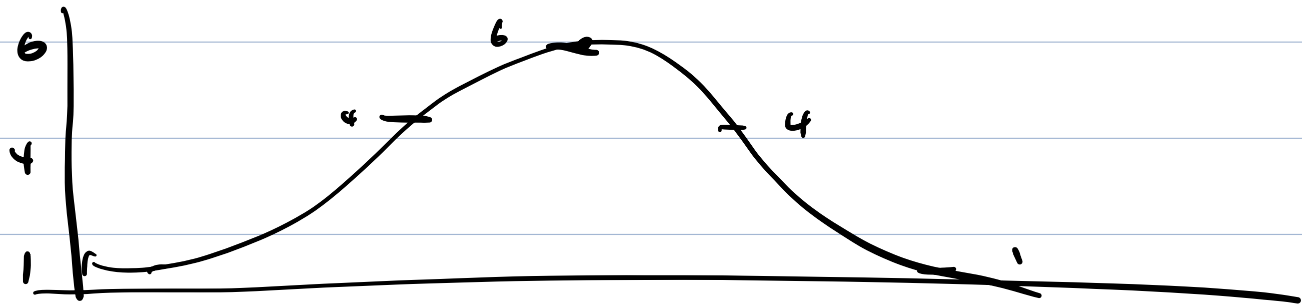
$$\text{bilinear} = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \frac{1}{16}$$

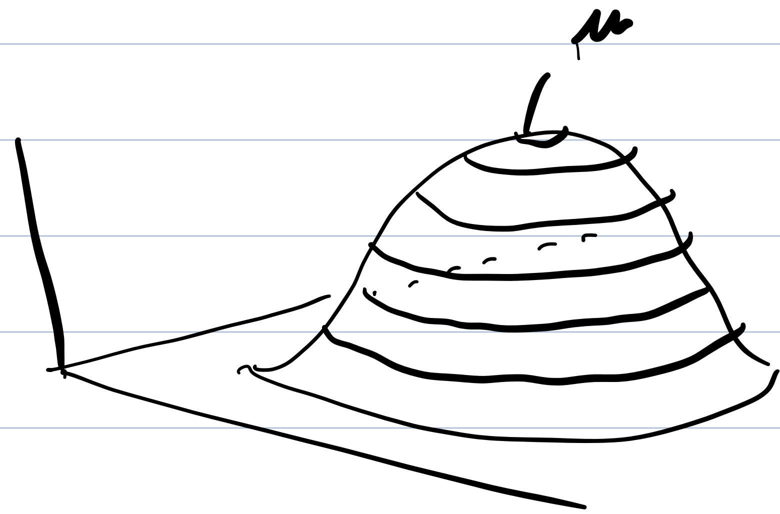
- Weights are prioritized by how close they are to the center.

Gaussian



$1D = [1, 4, 6, 4, 1]$





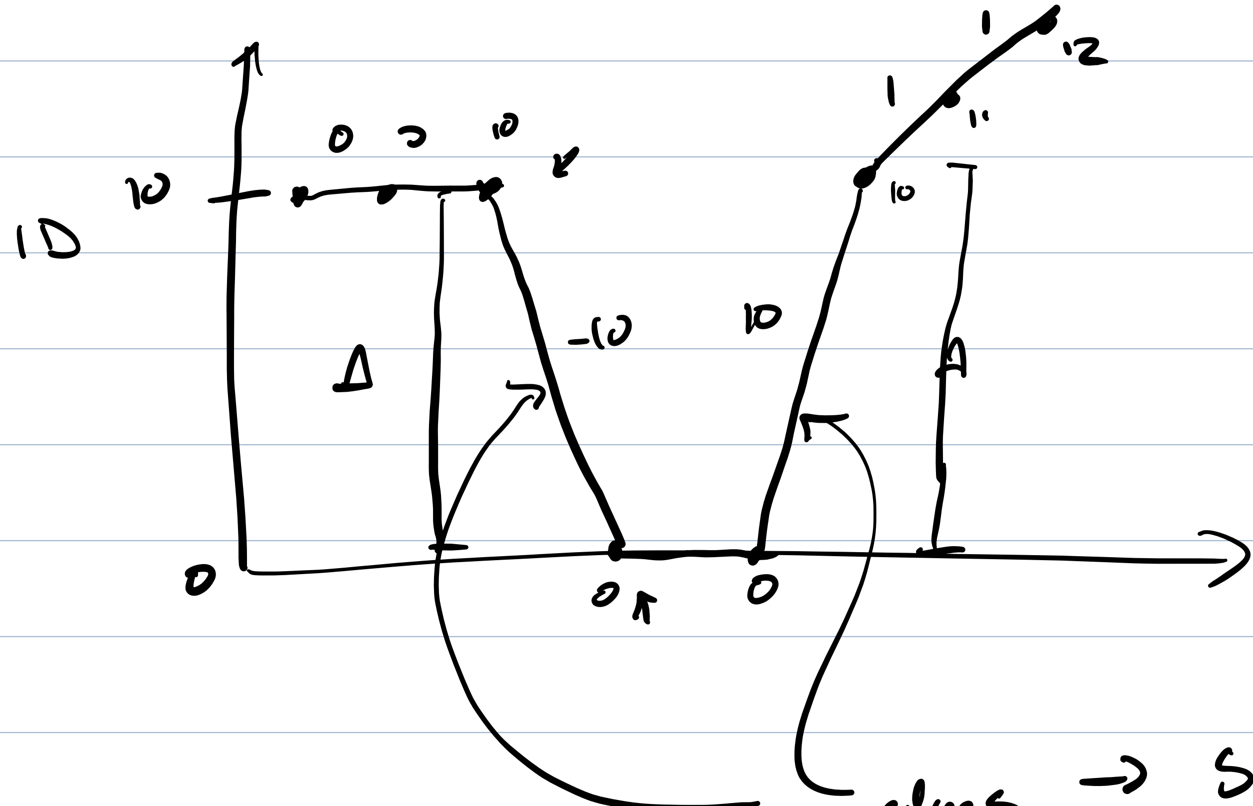
5x5

$$2D = \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix} \frac{1}{256}$$

* The most common filter for

Smoothing

Edge detection



edges → sharp change in the **brightness** of an image.

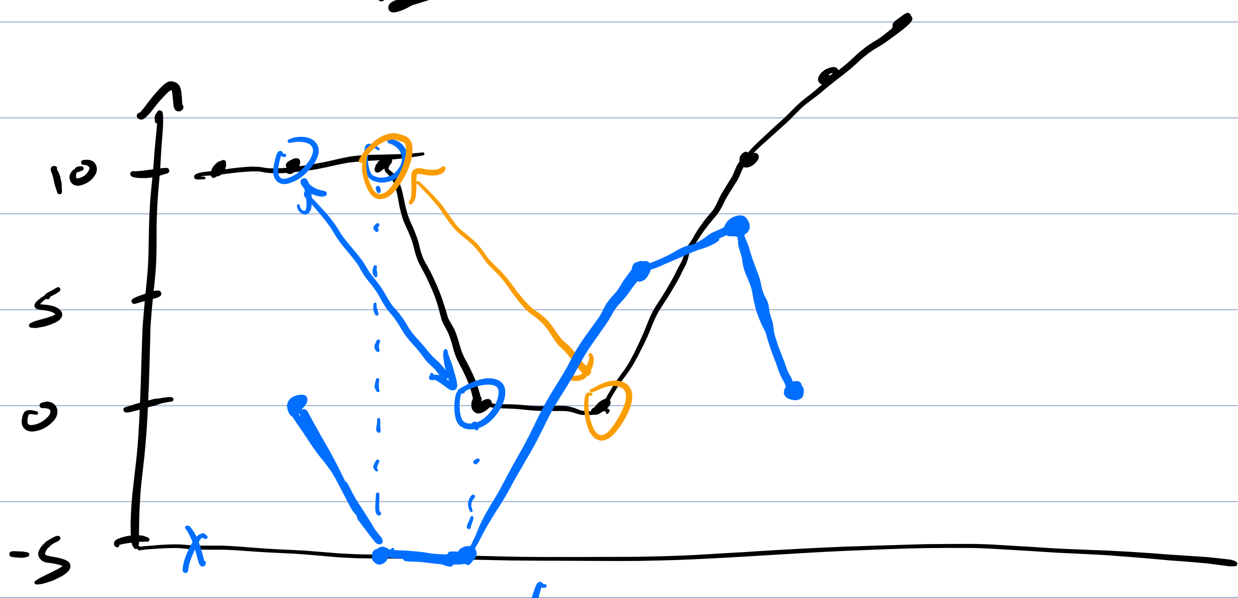
$x = [10, 10, 10, 0, 0, 10, 11, 12]$

sobel filter $= [-1, 0, 1]\,\frac{1}{2}$

$$\frac{\cancel{-10} + 0 + \cancel{10}}{2} = 0$$

results = [X, 0, -5, -5, 5, 5.5, 1, X]

$$\frac{-10+0+0}{2}$$



|abs| of the X'/edge

X = [X, 0, 5, 5, 5, 5.5, 1, X]

- once you compute the edge "strength"
  - set a lower threshold to remove "non" edges
  - the lower threshold, the more edges there will be

- the higher the threshold, the lower of edges.

$$\text{Sobel} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \frac{1}{8} \leftarrow gx$$

edges in the x direction

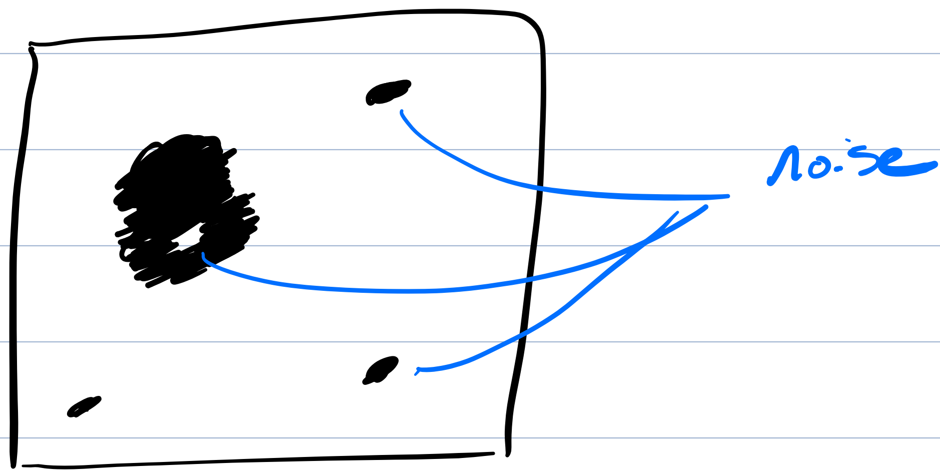
$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad gy$$

$$\begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -2 \end{bmatrix}$$

Morphological Filters.

- Dilation and Erosion of binary images.



- Dilation - Increasing object area
  - fills in "holes"

  - takes in an n x n kernel

  - if there is at least one pixel in the kernel of a certian value (1). Set the

value of pixel (i,j) to the
Same value.

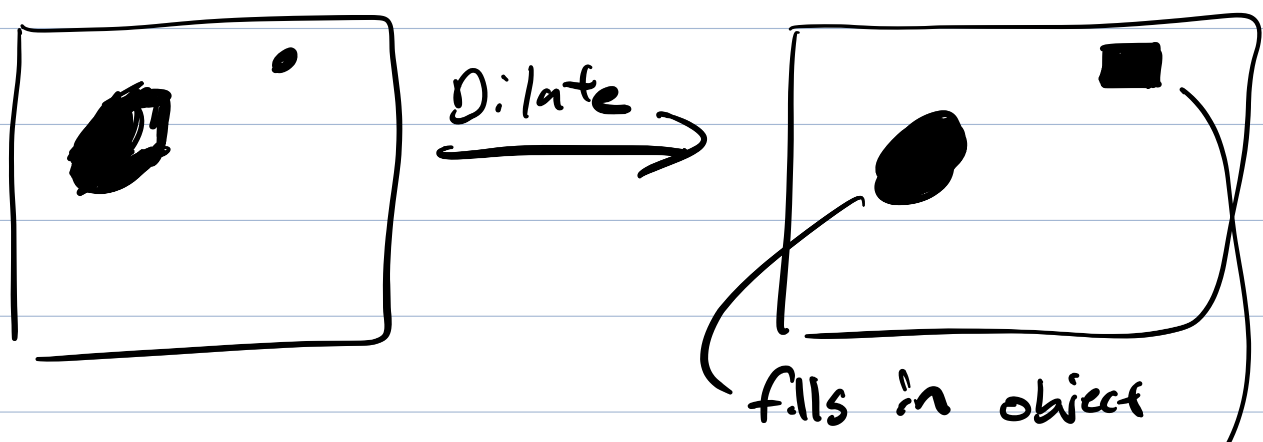
image

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \longrightarrow \text{Ceiling round}\left(\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \frac{1}{9} \right)$$

$$0.01 \xrightarrow[\text{round}]{\text{Ceiling}} 1$$

- every value has to be $\emptyset$ for it to be $\emptyset$.

- if there is at least one 1 then the output will be 1.



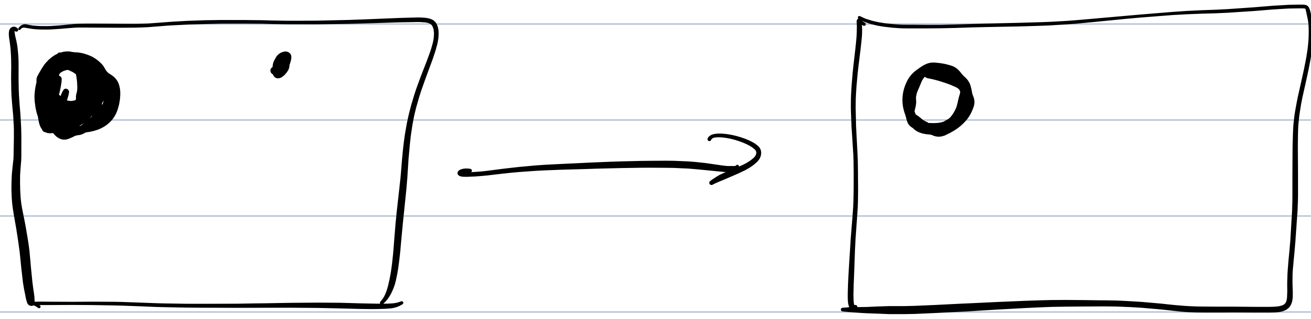
can amplify
other noise

Erosion — decreases object area
— removes small noise

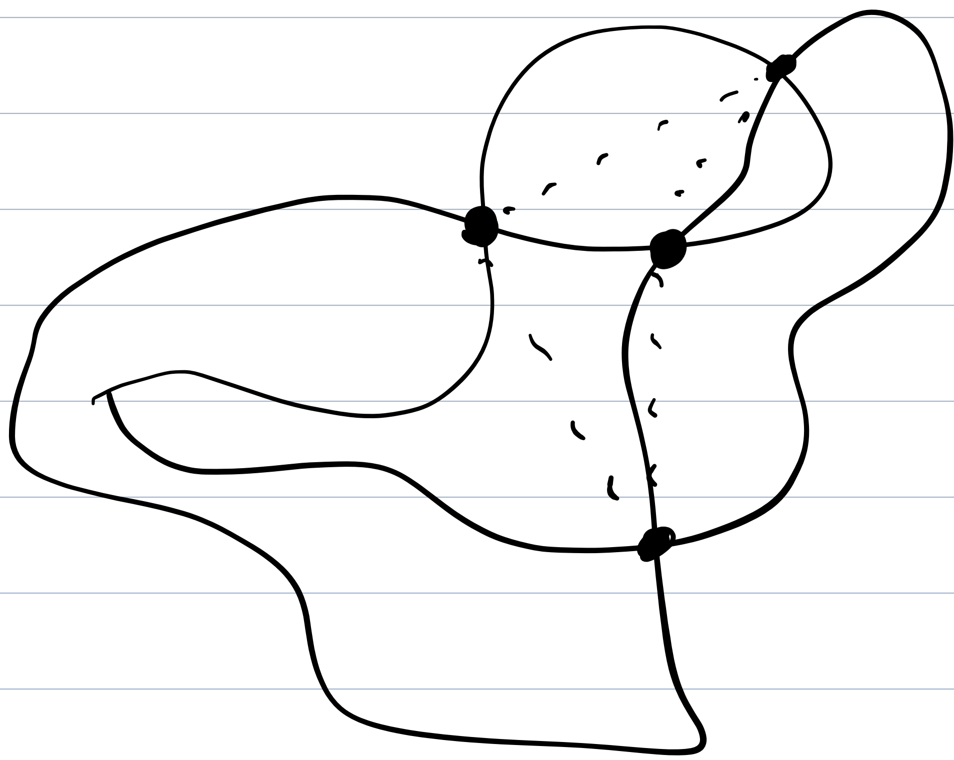
- if any pixel is set to zero in the kernel, set value to zero.

floor
round

$$\left(\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}\right)\frac{1}{9}$$



Opening → erode then dilate
Closing → dilate then erode



Steps of edge detection.

Sobel

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \rightarrow g_x \qquad \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \rightarrow g_y$$

Prewitt

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

<u>Step 1</u> Calculate both g_x and g_y of a given pixel (i,j).

<u>Step 2</u> Calculate the local edge magnitude.

- $g_{x,y} = |g_x| + |g_y|$

- $g_{x,y} = (g_x^2 + g_y^2)^{\frac{1}{2}} = \sqrt{g_x^2 + g_y^2}$

- $g_{x,y} = (\max(|g_x|, |g_y|))$

<u>optional step 3</u>

- implement a threshold.
- keep only the more extreem edges.

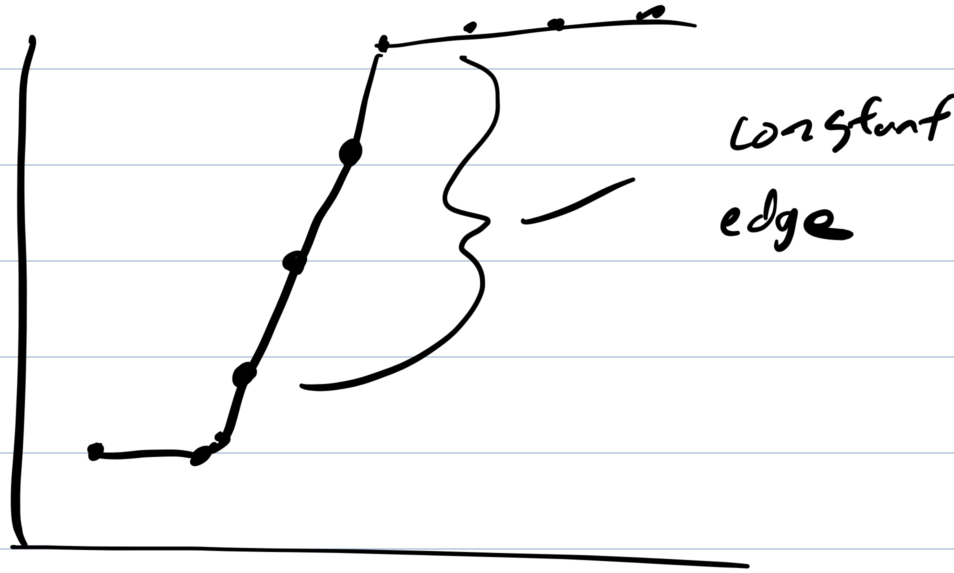
if ($g_{x,y}$ > threshold):

img[i, j] = 255 ← white.

else:

img[i, j] = 0

- issues with Sobel



↓

- will create a large "thick" edge.