((Image processing filters))

* In addition to color that reveals some information, the changing patters of grayscale intensity in an image can enhance our knowledge about the image. → These info reveal edges and borders of images → To reveal such features, we use filters.

* One good measure is frequency! since the frequency reveals the rate of changes, high frequency image in one where the intensity changes alot. This change can be due to the brightness change or edges. Images typically have both high and low frequency sections.

* Filters in CV are used to:

1/ Filter out unwanted information

2/ Amplify features of interest

* High-pass filters:

1/ Sharpen an image

2/ Enhance high-frequency parts of an image.

→ It emphasizes the edges.

↳ example → kernel conv. →

0	-1	0
-1	4	-1
0	-1	0

* Another example of filters is sobel filters for edge detection: $S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$ $S_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$

* All images have noise that might be the result of speckle or discoloration in an image which contains no information about the image and even might mess with the image processing steps. → most common way to remove noise is using low-pass filter which mainly:

1/ Blur/smooth an image

2/ Block high-frequency parts of an image.

→ This is very useful for improving quality in medical imaging → example: averaging filter

$\frac{1}{9} \times$

1	1	1
1	1	1
1	1	1

⊗ The averaging filter blures the image uniformly. But, we want to low-pass filter the image and preserve the edges as much as possible. ⇒ we have to do the weighted average with the most significant weight to the center:

$\frac{1}{16} \times$

1	2	1
2	4	2
1	2	1

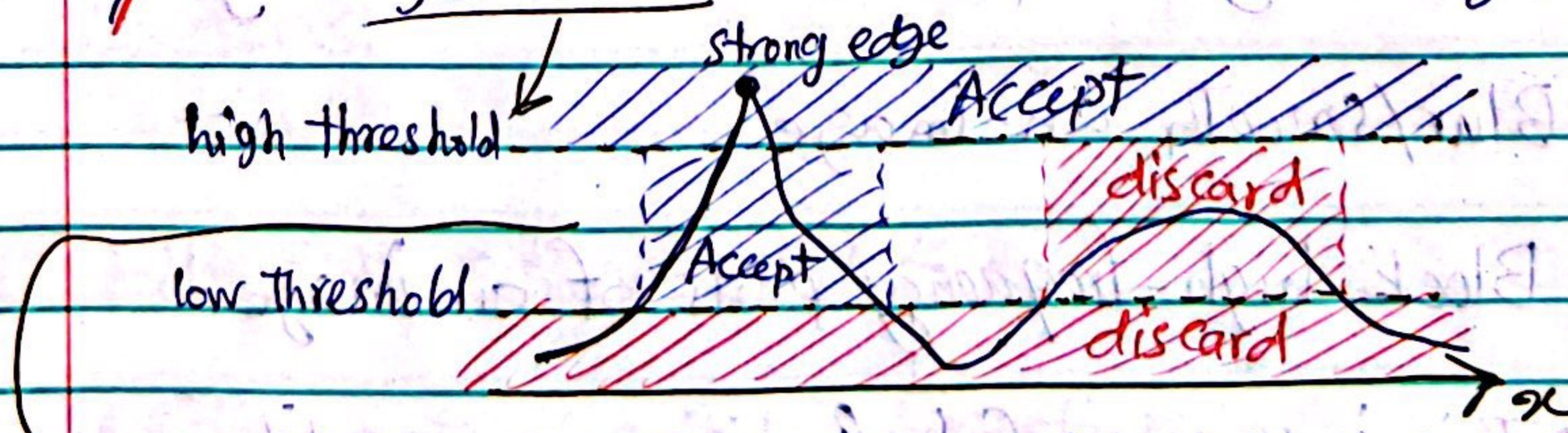
⊗ Canny edge detection:

1/ Filters out noise with Gaussian Blur.

2/ Finds the strength and direction of edges using sobel filters.

3/ Applies non-maximum suppression to isolate the strongest edges and thin them to one-pixel wide lines.

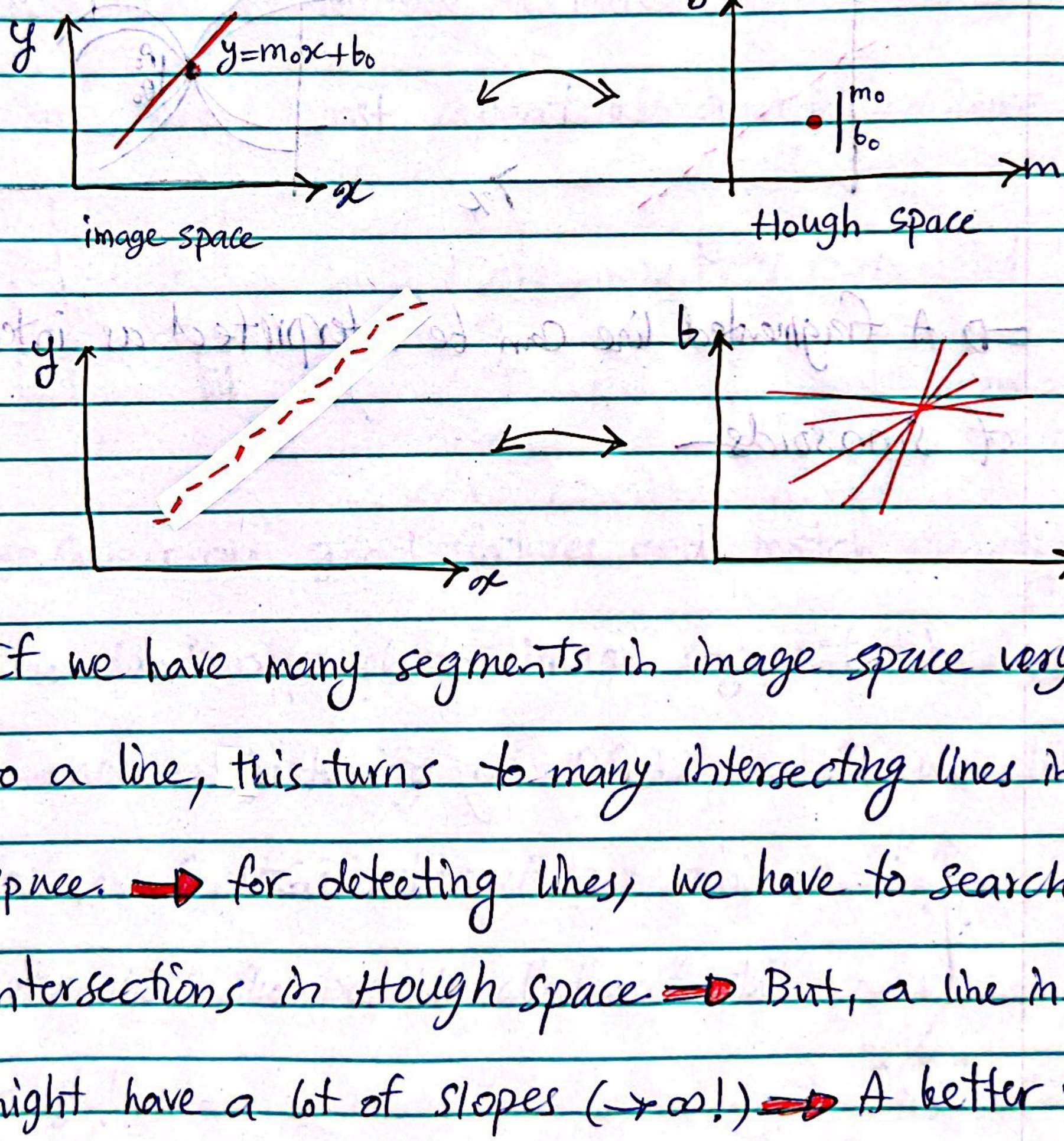
4/ Uses hysteresis to isolate the best edges



→ All edges in between high and low thresholds only will be kept if they are connected to a strong edge

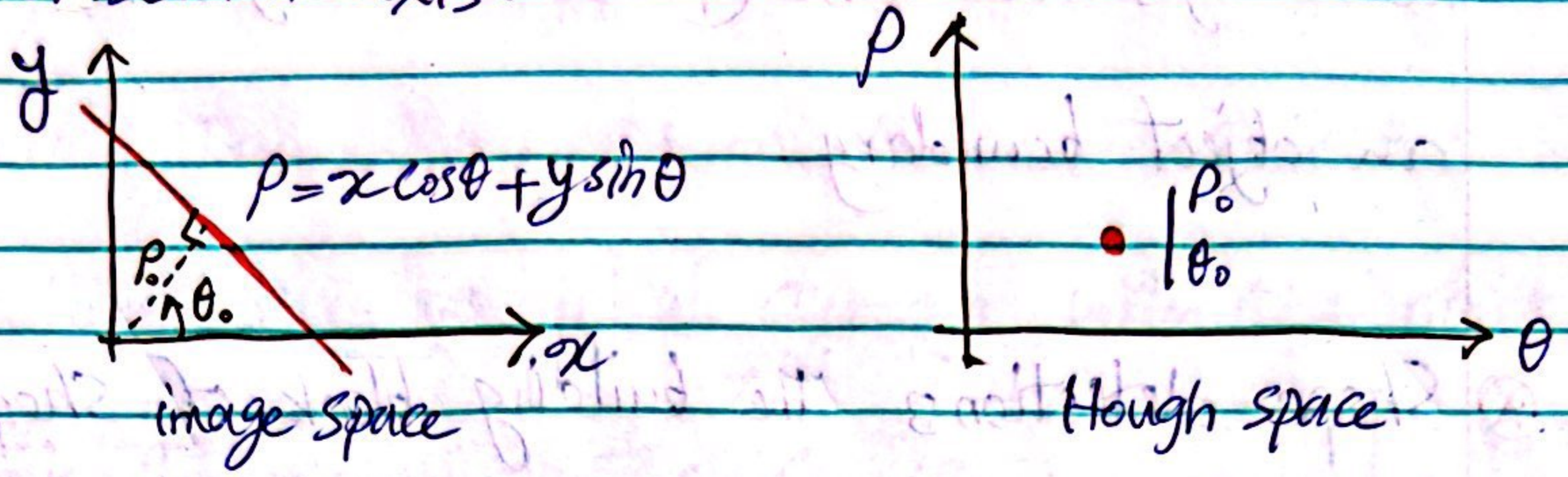
→ The hysteresis part of the canny algorithm, eliminates weak edges and noise ① isolates edges that are part of an object boundary.

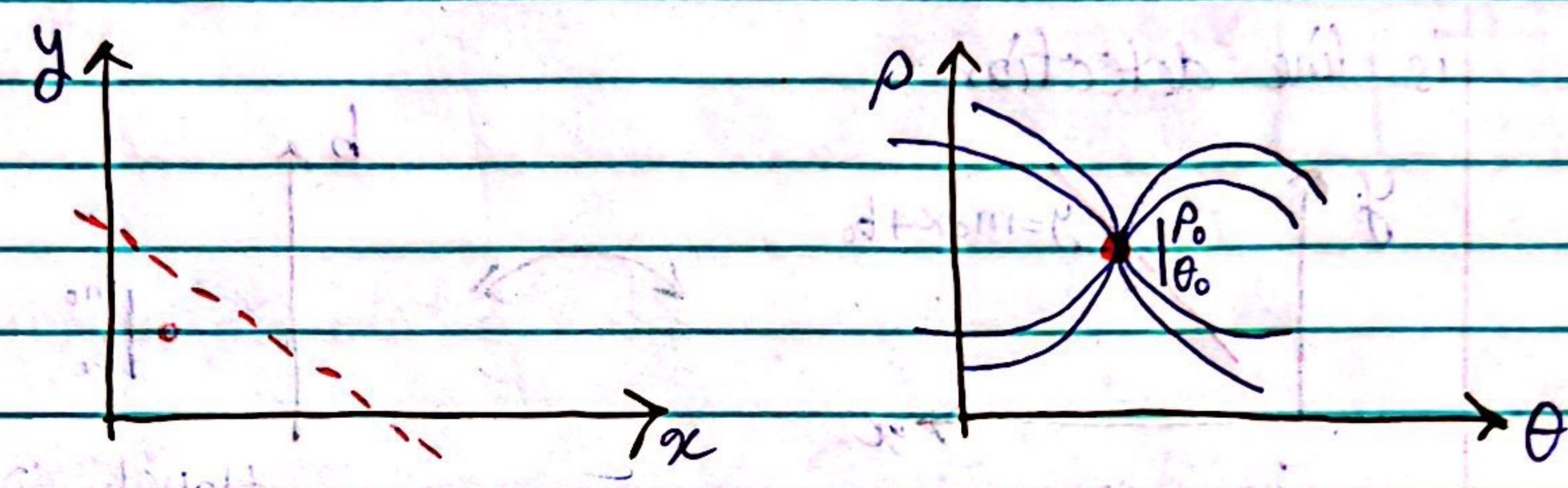
* Shape detection: the building block of shape detection is line detection:



* If we have many segments in image space very close to a line, this turns to many intersecting lines in Hough space. → for detecting lines, we have to search for intersections in Hough space → But, a line in an image might have a lot of slopes ($\to\infty$!) → A better transform is turning Hough space to polar coordinate! → instead of m and b we have ρ which is the distance of the line

from the origin and θ which is the angle from the horisontal axis.





→ A fragmented line can be interpreted as intersection of sinosoids