Seminar 2 - Lectures 4 and 5
Computer Vision 1, Master AI, 2025

Martin Oswald, Dimitris Tzionas, Qi Bi, Roan van Blanken

1 Exercise 1: Mathematical Morphology

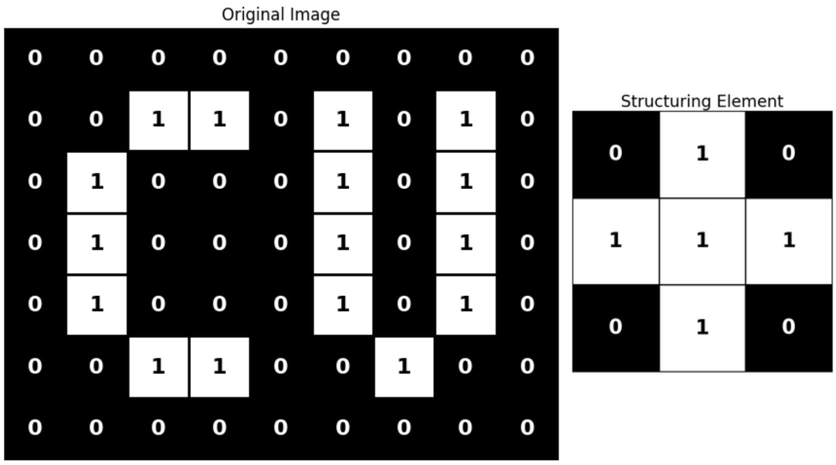

Figure 1: Input binary image and structuring element.

Q.1.a Consider a 7x9 binary image shown in Figure 1, where the value '1' represents the foreground, and '0' represents the background. The 3x3 matrix next to the image is the structuring element.

Perform a dilation operation on the image using the given structuring element, and assume zero padding around the edges of the image.

Answer: See Figure 2:

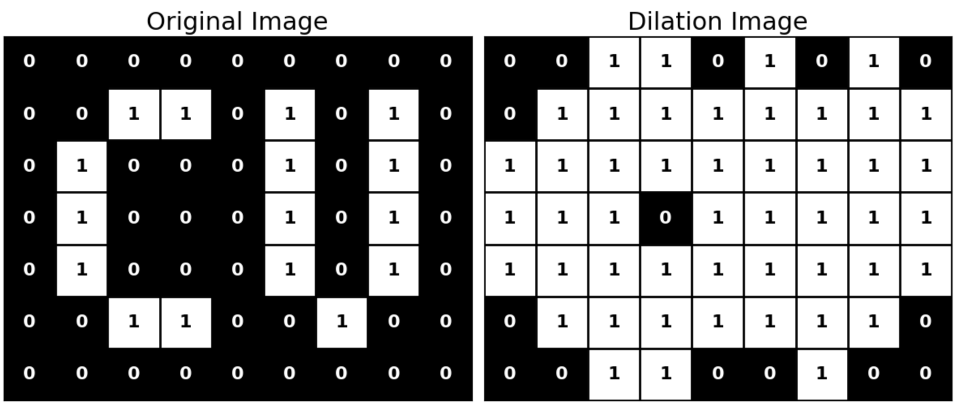

Figure 2: Result of the dilation operation on the 7x9 binary image, using the given structuring element and assuming zero padding around the edges.

Q.1.b Consider the same 7x9 binary image shown in Figure 1. Perform an erosion operation on the image using the given structuring element, and assume zero padding around the edges of the image.

Answer: See Figure 3:

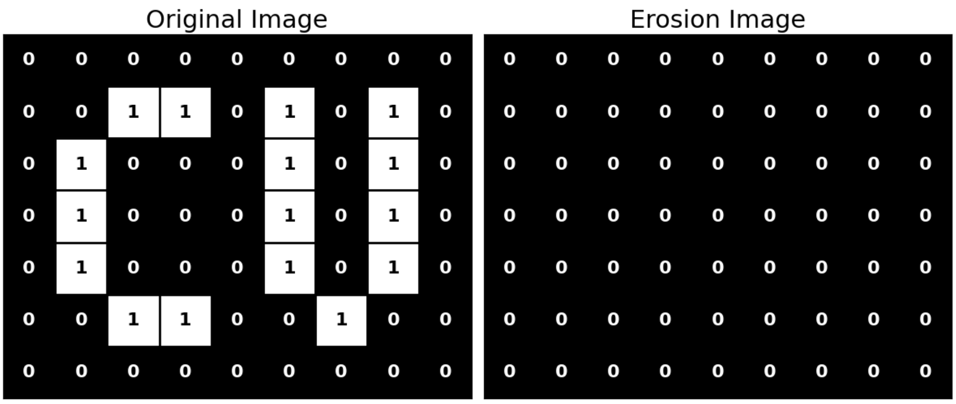

Figure 3: Result of the erosion operation on the 7x9 binary image, using the given structuring element and assuming zero padding around the edges.

Q.1.c Consider the same 7x9 binary image shown in Figure 1. Perform a closing operation on the image using the given structuring element, and assume zero padding around the edges of the image.

Answer: See Figure 4:

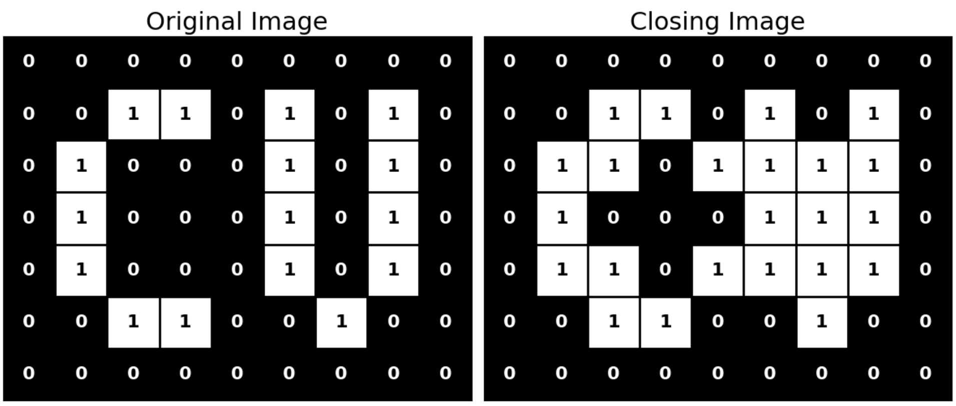

Figure 4: Result of the closing operation on the 7x9 binary image, using the given structuring element and assuming zero padding around the edges.

Q.1.d Would the result of the closing operation in the previous question have been different if a different padding method (such as mirror, clamp, or wrap) had been used instead of zero padding? Explain your reasoning.

Answer: No, the result would not have been different with any of these padding methods. Since the pixels at the edges of the image are all zeros, the choice of padding (zero, mirror, clamp, or wrap) would not affect the outcome of the closing operation. The structuring element would not interact with any non-zero pixels at the edges, so the final result remains the same.

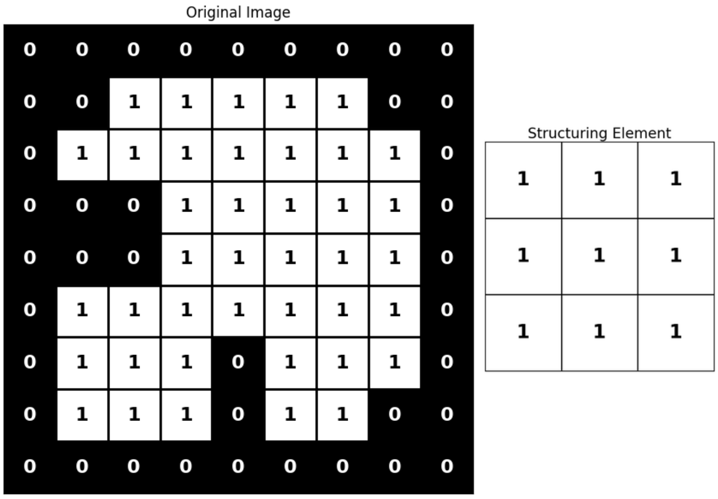

Figure 5: Input binary image and structuring element.

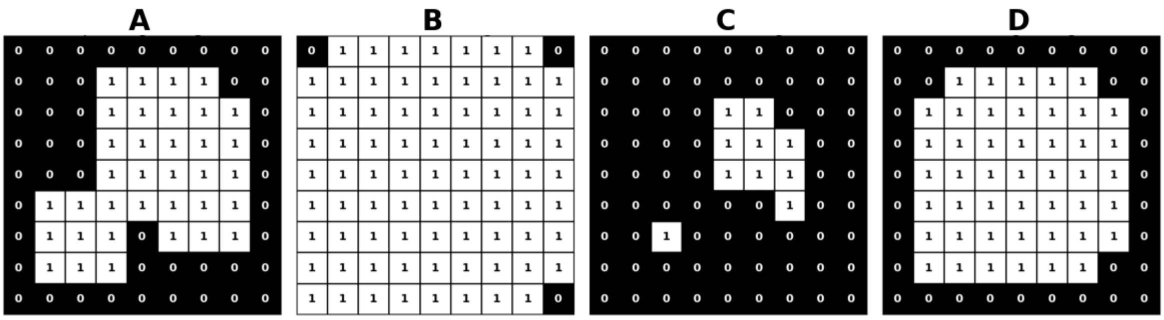

Figure 6: Morphological operation results on the input binary image.

Q.1.e Consider a different 9x9 binary image and structuring element, as shown in Figure 5. Four different morphological operations (Dilation, Erosion, Opening, and Closing) have been applied to this image using the given structuring element. Assume zero padding around the edges of the image.

The results of these operations are shown in Figure 6, with the outputs labeled A, B, C, and D. Match each output to the correct morphological operation.

Answer: Subplots A, B, C, and D correspond to Opening, Dilation, Erosion, and Closing, respectively.

2 Exercise 2: Connected Component Analysis

Connected Component Analysis, or more broadly, Connected Component Labeling, is a set of algorithms used to identify and label (sub)sets of connected components within a graph. This concept can be naturally applied to an image by treating the image as a graph.

Q.2.a How would you construct an image graph? How are pixels connected?

Answer: To construct an image graph, the vertices represent the pixels of the image, and the edges represent the connections between neighboring pixels. There are two common ways to define pixel connectivity:

- **4-connected:** Each pixel is connected only to its horizontal and vertical neighbors (up, down, left, right).
- **8-connected:** Each pixel is connected to all its eight neighbors (including diagonals).

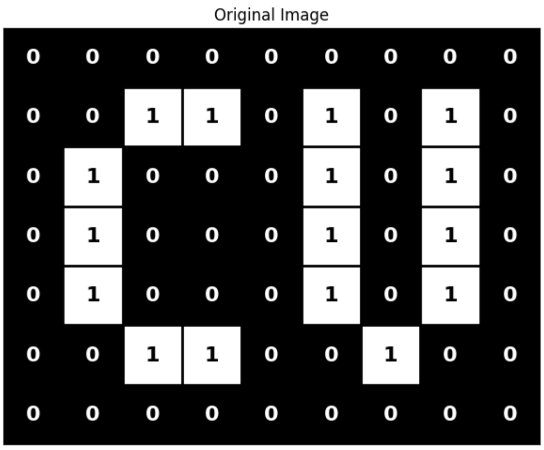

Figure 7: A binary image for connected component analysis.

Q.2.b Identify and label the connected components in the binary image shown in Figure 7. First, use 4-connected neighbors, then use 8-connected neighbors.

Answer: When using 4-connected neighbors, there are 6 distinct connected components in the image. When using 8-connected neighbors, there are only 2 distinct connected components. See the results in Figure 8.

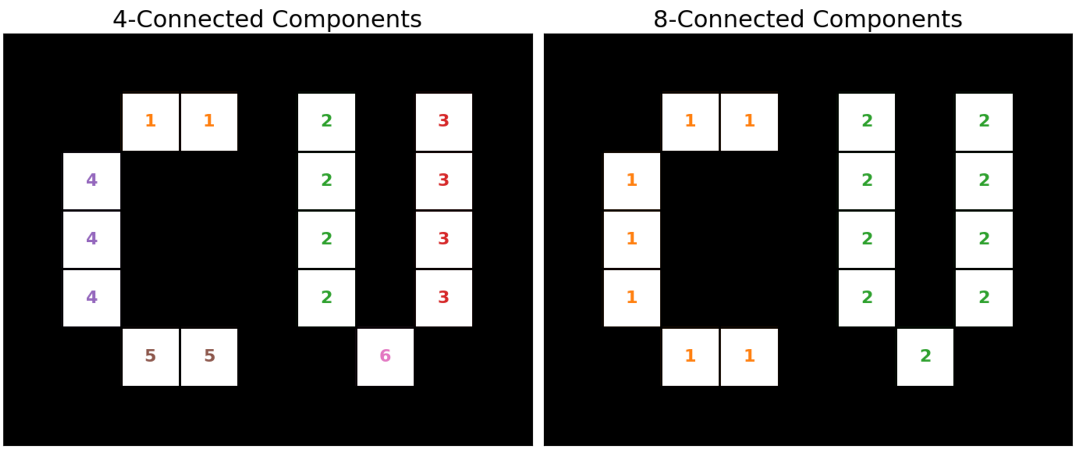

Figure 8: Connected components labeled in the binary image using 4-connected and 8-connected neighbors.

3 Exercise 3: Image Filtering

In image processing, kernels are often applied to an image to alter its appearance—a process known as image filtering. A kernel is a local operator that depends only on the neighborhood of pixels around each point. Below are some types of commonly used kernels.

3x3 Uniform (Box) Kernel

$$T = \frac{1}{9}\begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & \underline{1} & 1 \\ \hline 1 & 1 & 1 \\ \hline\end{array}$$

5x5 Uniform (Box) Kernel

$$U = \frac{1}{25}\begin{array}{|c|c|c|c|c|}\hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & \underline{1} & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline\end{array}$$

Another 3x3 kernel

$$V = \frac{1}{6}\begin{array}{|c|c|c|}\hline 1 & 0 & -1 \\ \hline 1 & \underline{0} & -1 \\ \hline 1 & 0 & -1 \\ \hline\end{array}$$

Yet another 3x3 kernel

$$W = \begin{array}{|c|c|c|}\hline 1 & -2 & 1 \\ \hline -2 & \underline{4} & -2 \\ \hline 1 & -2 & 1 \\ \hline\end{array}$$

3x3 Laplacian Kernel

$$L = \begin{array}{|c|c|c|}\hline 0 & 1 & 0 \\ \hline 1 & \underline{-4} & 1 \\ \hline 0 & 1 & 0 \\ \hline\end{array}$$

3x3 Sobel Kernel

$$S = \begin{array}{|c|c|c|}\hline 1 & 0 & -1 \\ \hline 2 & \underline{0} & -2 \\ \hline 1 & 0 & -1 \\ \hline\end{array}$$

Q.3.a Which of the following kernels would make an image blurrier: a 3x3 uniform kernel or a 5x5 uniform kernel? Explain your reasoning.

Answer: A 5x5 uniform kernel would make the image blurrier. This is because it averages the pixel values over a larger area, leading to a greater smoothing effect and thus more blurring.

Q.3.b What types of image features are detected by the 3x3 kernels V and W? Describe their functionality.

Answer: V: This kernel detects vertical edges in an image. It is composed of a smoothing component and a differentiation component, which highlights changes in pixel intensity along the vertical direction.
W: This kernel is used to detect corners and, for larger kernels, it can detect blobs. It is a symmetric kernel that represents a fourth derivative operation, emphasizing areas in the image where intensity changes occur in multiple directions.

Q.3.c How can the transformation of an image, achieved by convolving a 3x3 averaging **kernel** T followed by the 3x3 **kernel** W, be implemented using a single 5x5 kernel? Calculate the elements of this resulting kernel.

Answer: When applying a kernel to an image, the operation typically involves convolution, which is both commutative and associative. This allows us to first convolve the two kernels and then apply the resulting kernel to the image.

Due to the associativity of convolution, we can convolve kernels T and W to obtain a single kernel that combines their effects. In this case—since both kernels are isotropic—convolution and cross-correlation yield the same result.

The combined 5x5 kernel M is calculated as follows:

$$M = \frac{1}{9}\begin{array}{|c|c|c|c|c|}\hline 1 & -1 & 0 & -1 & 1 \\ \hline -1 & 1 & 0 & 1 & -1 \\ \hline 0 & 0 & 0 & 0 & 0 \\ \hline -1 & 1 & 0 & 1 & -1 \\ \hline 1 & -1 & 0 & -1 & 1 \\ \hline\end{array}$$

This 5x5 kernel M can now be used to achieve the same transformation as convolving the image with T followed by W.

Q.3.d Demonstrate that the Sobel filter S can be separated into two different 1-dimensional filters. Identify each of these 1-dimensional filters as either a low-pass filter, a high-pass filter, or a band-pass filter.

Answer: The Sobel filter S can be separated as follows:

$$S = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \circledast \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}.$$

Note that the separation uses the convolution operator $\circledast$ and includes the necessary flipping of the filter.

The first filter

$$\begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}$$

is a 1D Gaussian-like smoothing filter and acts as a low-pass filter, which smooths the image and suppresses high-frequency noise.

The second filter

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

is a 1D derivative filter and acts as a high-pass filter, emphasizing edges and high-frequency components.

Consider the following image matrix A:

$$A = \begin{array}{|c|c|c|c|} \hline 0 & 0 & 0 & 0 \\ \hline 1 & 1 & 0 & 1 \\ \hline 0 & 1 & 0 & 0 \\ \hline 1 & 1 & 1 & 0 \\ \hline \end{array}$$

Q.3.e Correlate Image A with kernel T. Use zero padding if necessary.

Answer: To perform the correlation of image A with the 3x3 averaging kernel T, zero padding is applied around the borders of A to handle the edges of the image. The kernel T averages the pixel values within its window, and the operation results in a smoother, blurred version of the image. The resulting image after the convolution is:

$$\frac{1}{9} \begin{array}{|c|c|c|c|} \hline 2 & 2 & 2 & 1 \\ \hline 3 & 3 & 3 & 1 \\ \hline 5 & 6 & 5 & 2 \\ \hline 3 & 4 & 3 & 1 \\ \hline \end{array}$$

This output represents the pixel values after applying the smoothing effect of the averaging kernel.

Q.3.f Apply a 3x3 median kernel on Image A. Use mirror padding if necessary.

Answer: First, apply mirror padding to Image A to handle the edges. Mirror padding replicates the edge pixels around the border of the image. The padded image A is as follows:

Padded Image A

0	**0**	**0**	**0**	**0**	**0**
0	0	0	0	0	**0**
1	1	1	0	1	**1**
0	0	1	0	0	**0**
1	1	1	1	0	**0**
1	**1**	**1**	**1**	**0**	**0**

Image after Median Filter

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

4 Exercise 4: Hough Transform

The Hough Transform is an image processing technique used to detect instances of shapes within an image. Originally, it was designed to detect lines, but it has since been extended to detect more complex shapes such as circles, ellipses, and arbitrary shapes. For the purposes of these questions, we will focus on the Hough Transform for line detection.

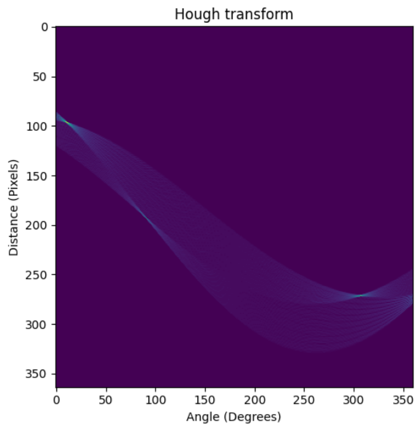

Figure 9: The accumulator matrix depicted as an image. This matrix results from applying a Hough Transform to an image with lines.

Q.4.a Consider the accumulator matrix after applying the Hough Transform to an image, as shown in Figure 9. How many lines are present in the original image?

Answer: In this example, the parameter space (Hough space) is dense and noise-free, making it easy to identify three local maxima. Therefore, the original image contains 3 lines. The original image is shown in Figure 10. Note that in most cases, the Hough space can be undersampled and noisy, making it more challenging to find local maxima.

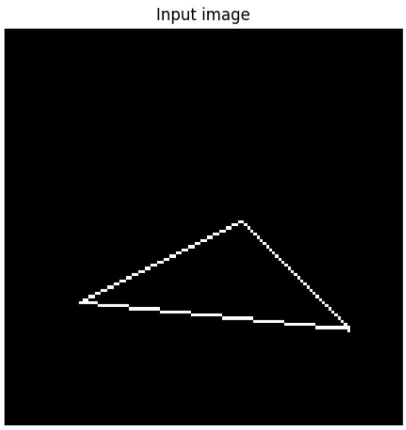

Figure 10: The original image that produces the accumulator matrix shown in Figure 9 after applying a Hough Transform.

Q.4.b The accumulator space of the Hough Transform is often expressed in polar coordinates (ρ, θ) instead of line parameters (a, b). Why is this?

Answer: When dealing with vertical lines, the slope parameter a goes to infinity, which poses a problem. To address this, the coordinates are transformed into polar coordinates: $\rho = x\cos\theta + y\sin\theta$. In (ρ, θ)-space, a sinusoidal curve represents the set of all straight lines passing through a specific point in the image. When detecting points that lie on the same line in the image, we look for points in (ρ, θ)-space where many curves intersect, known as concurrent curves. This can be observed in Figures 9 and 10.

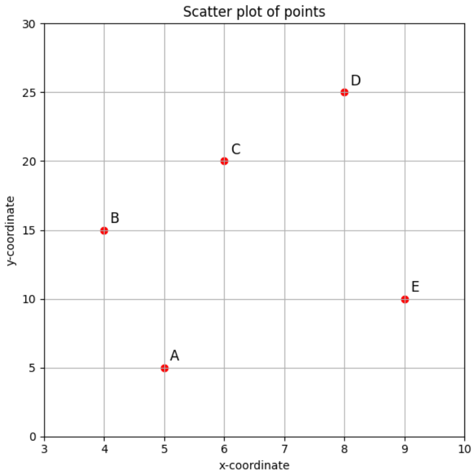

Figure 11: Scatter plot of points

Q.4.c Five points in an image, as shown in Figure 11, are given with the following coordinates: $A = (A_x, A_y) = (5, 5)$, $B = (4, 15)$, $C = (6, 20)$, $D = (8, 25)$, $E = (9, 10)$. Demonstrate how you can fit a line to these points using the Hough Transform. For simplicity, use the linear version of the Hough Transform: $y = ax + b$.

Answer: To solve this problem using the Hough Transform, first, write the equation $y_i = ax_i + b$ for each point (x_i, y_i). For the points $A = (5, 5)$, $B = (4, 15)$, $C = (6, 20)$, $D = (8, 25)$, and $E = (9, 10)$, the corresponding equations in the (a, b) space are:

- For $A(5, 5)$: $y = ax + b \Rightarrow 5 = 5a + b \Rightarrow b = -5a + 5$
- For $B(4, 15)$: $y = ax + b \Rightarrow 15 = 4a + b \Rightarrow b = -4a + 15$
- For $C(6, 20)$: $y = ax + b \Rightarrow 20 = 6a + b \Rightarrow b = -6a + 20$
- For $D(8, 25)$: $y = ax + b \Rightarrow 25 = 8a + b \Rightarrow b = -8a + 25$
- For $E(9, 10)$: $y = ax + b \Rightarrow 10 = 9a + b \Rightarrow b = -9a + 10$

These equations represent lines in the (a, b) space. Plotting these lines, we observe that the lines corresponding to points B, C, and D intersect at the point $(a, b) = (2.5, 5)$. Since this point has the most intersections (3 lines), we choose it as the best fit. Therefore, the line that best fits the points A, B, C, D, and E has the equation $y = 2.5x + 5$ (see Figure 12).

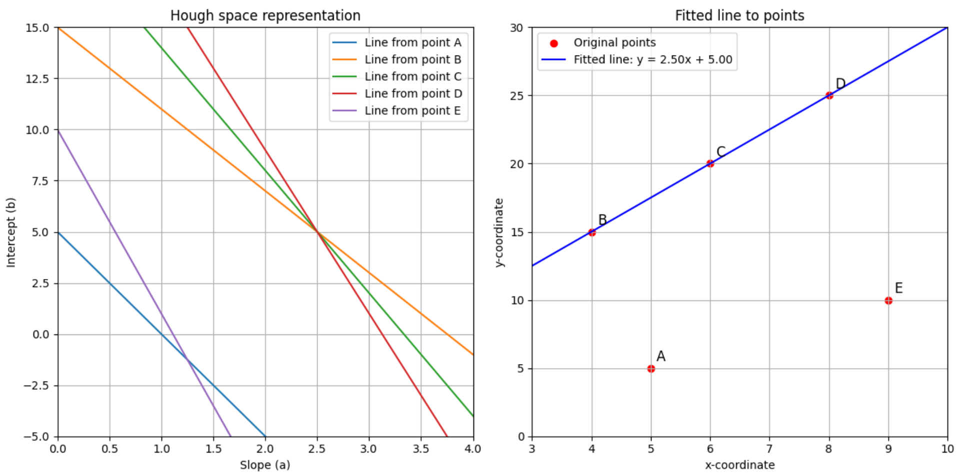

Figure 12: Hough Transform space and fitted line