Final Project : Paper Homography

Computer Vision

Fall 2024

Chanakya Nalapareddy
Nosherwan Babar

12/11/2024

First Image Sample

Part 1: Input and Edge Image

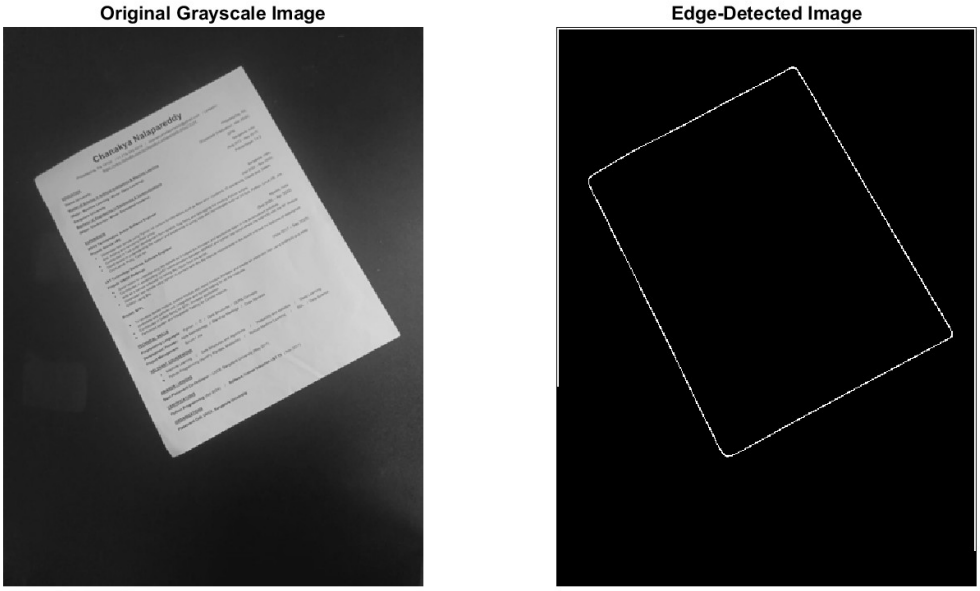

Figure 1: Input and Edge Image. The left shows the resized grayscale input image, and the right shows the edge-detected image after applying Canny edge detection and morphological operations.

Part 2: Hough Transform Histogram

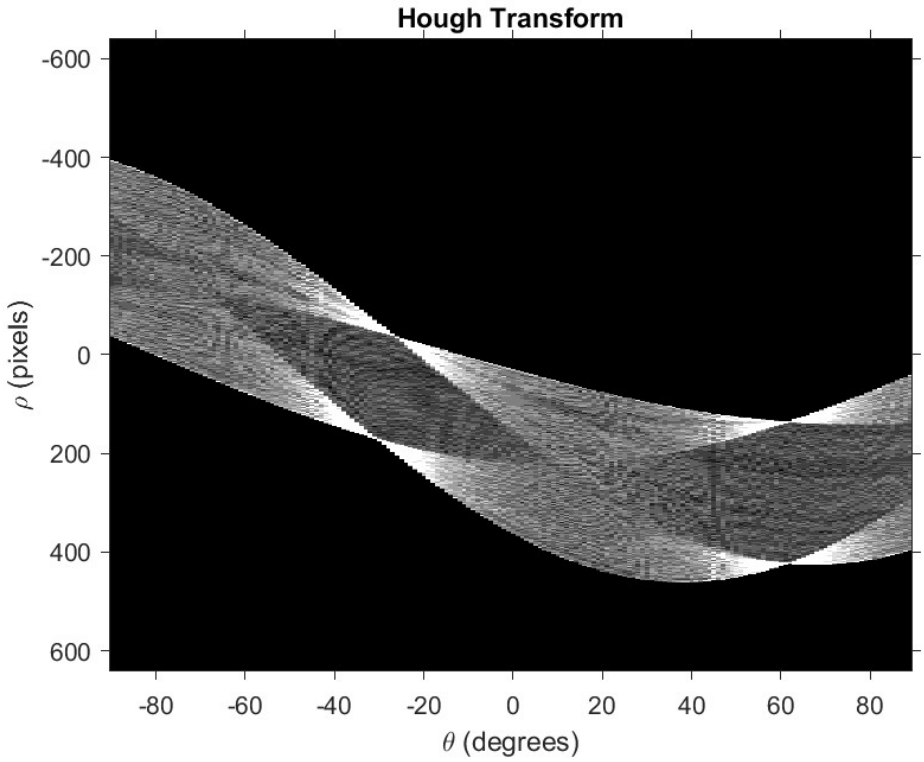

Figure 2: 2D Hough Transform Histogram. The accumulator space represents potential lines in the edge-detected image, with peaks corresponding to prominent line candidates.

Part 3: Extracted Points and Superimposed Lines

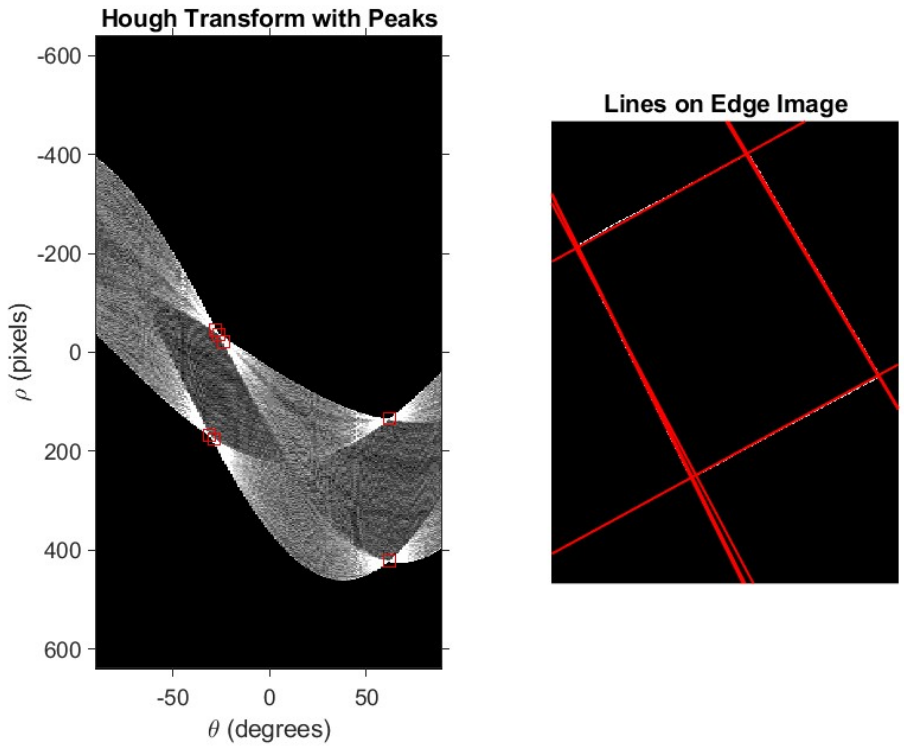

Figure 3: Selected Lines Superimposed on Edge-Detected Image. Red lines represent the detected edges of the paper.

Extracting Points from the Hough Transform and Superimposing Selected Lines

Hough Transform Peaks

The edge detected image is passed through Hough Transform to generate a 2D accumulator array $H(\rho, \theta)$, where:

- ρ is the perpendicular distance from the origin to the line.
- θ is the angle of line relative to the x-axis.

Peaks in $H(\rho, \theta)$ correspond to lines that accumulated significant votes from collinear edge pixels.

Peak Detection and Selection

The custom function, `myHoughPeaks`, is used to the extract peaks from $H(\rho, \theta)$:

- **Thresholding**: A threshold proportional to the maximum value in H is applied to filter out the weak peaks.
- **Non-Maximum Suppression**: Peaks are retained only if they are local maxima within a defined neighborhood, ensuring prominent lines were selected without overlapping.

The extracted peaks are stored as (ρ, θ) pairs.

Selecting Relevant Lines

From the extracted peaks, lines are filtered and prioritized based on:

- **Geometric Constraints**: Lines with similar θ values are grouped to identify vertical and horizontal edges of the document.
- **Line Separation**: Distinct lines are ensured by enforcing a minimum separation in ρ and θ values.

Superimposing Lines

Each selected (ρ, θ) pair was converted into Cartesian form using:

$$y = -\frac{\cos(\theta)}{\sin(\theta)}x + \frac{\rho}{\sin(\theta)}$$

The equations are extended to span the width and height of the edge-detected image, and the lines are drawn in red to superimpose them on the image.

Intersections Superimposed on the Original Image

Computing Intersections

Pairwise intersections of the selected lines are computed by solving their line equations:

$$\begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}$$

where a_1, b_1, c_1 and a_2, b_2, c_2 are coefficients of the two line equations in general form $ax + by = c$. Intersections (x, y) were calculated only for line pairs with sufficiently different θ values to ensure meaningful crossings.

Filtering Valid Intersections

- Intersections outside the image boundaries were discarded.
- Valid intersections are grouped into four quadrants defined by the bounding box center.

Selecting Document Corners

- In each quadrant, the intersection closest to the theoretical corner of the bounding box was selected as the document corner.
- This process ensured that the detected corners corresponded to the paper's physical edges.

Visualization of Intersections

The selected intersections are superimposed on the original grayscale image using red circles, clearly marking the detected corners. This step validated the accuracy of the detected lines and their intersections.

Part 4: Intersections Superimposed on Original Image

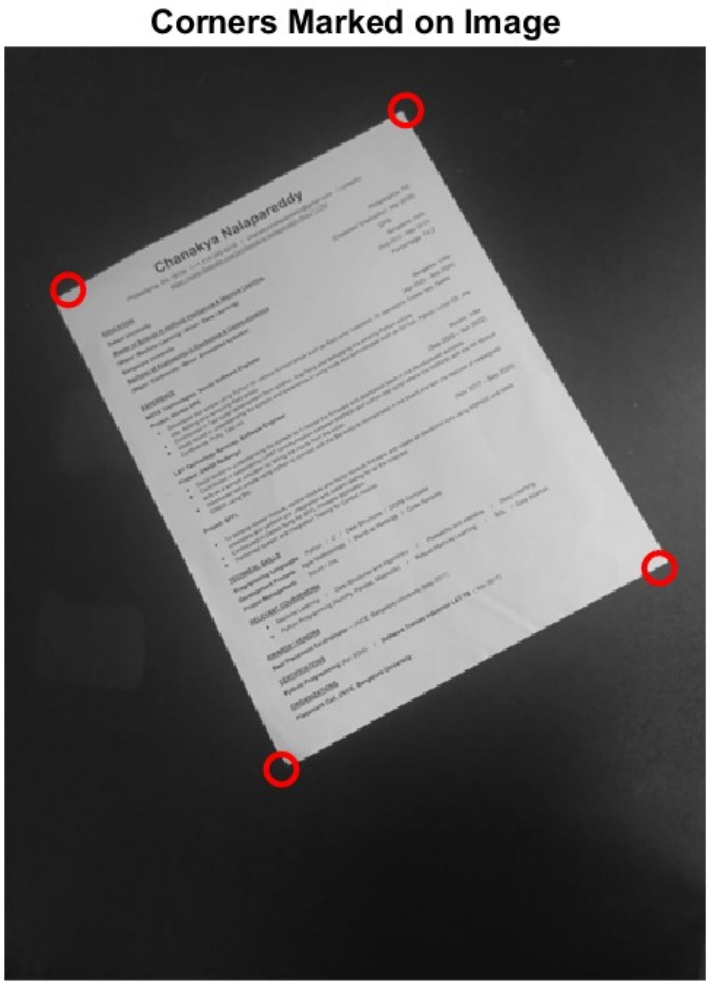

Figure 4: Intersections Superimposed on Original Image. Red circles mark the detected corners of the paper, aligned with the edge-detected image.

Part 5: Rectified Image

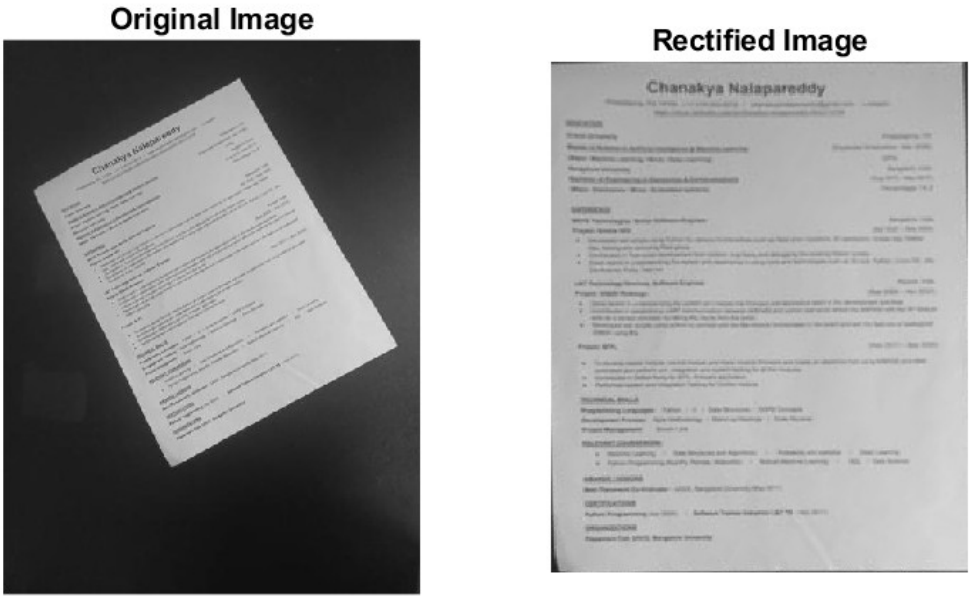

Figure 5: Rectified Image. The paper is rectified using the detected corners and a computed homography matrix.

Second Image Sample

Part 1: Input and Edge Image

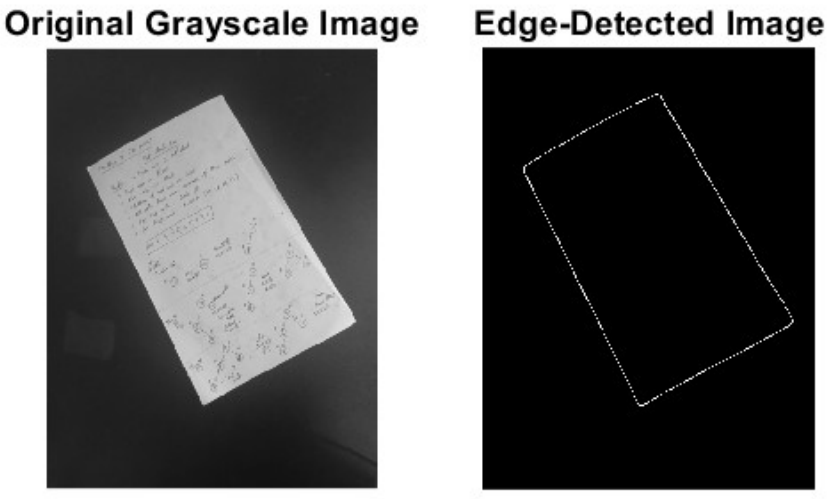

Figure 6: Input and Edge Image. The left shows the resized grayscale input image, and the right shows the edge-detected image after applying Canny edge detection and morphological operations.

Part 2: Hough Transform Histogram

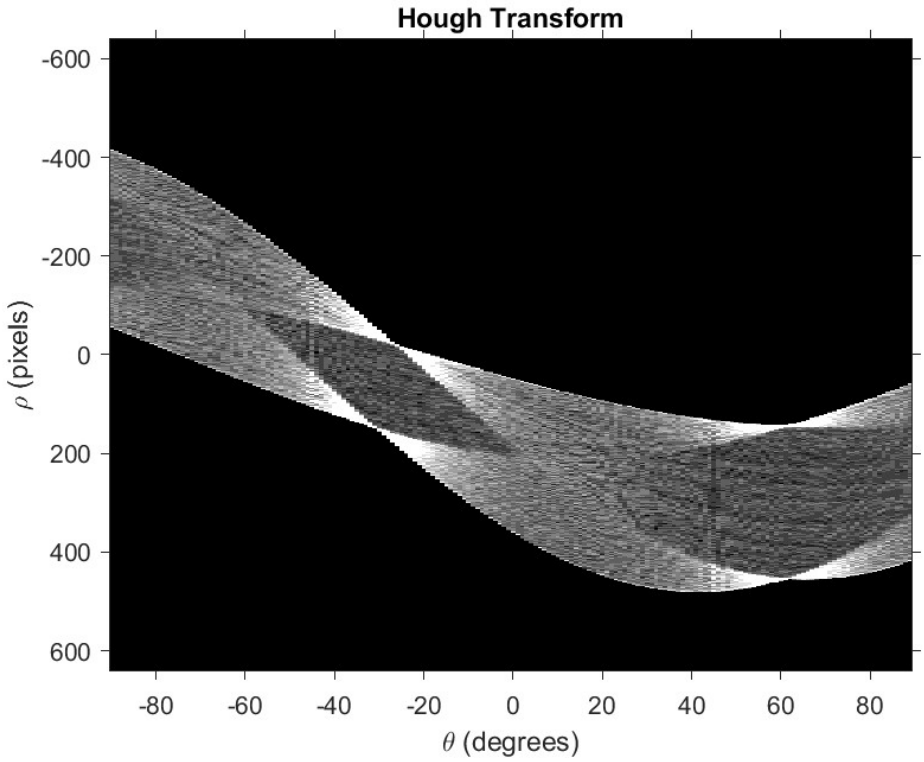

Figure 7: 2D Hough Transform Histogram. The accumulator space represents potential lines in the edge-detected image, with peaks corresponding to prominent line candidates.

Part 3: Extracted Points and Superimposed Lines

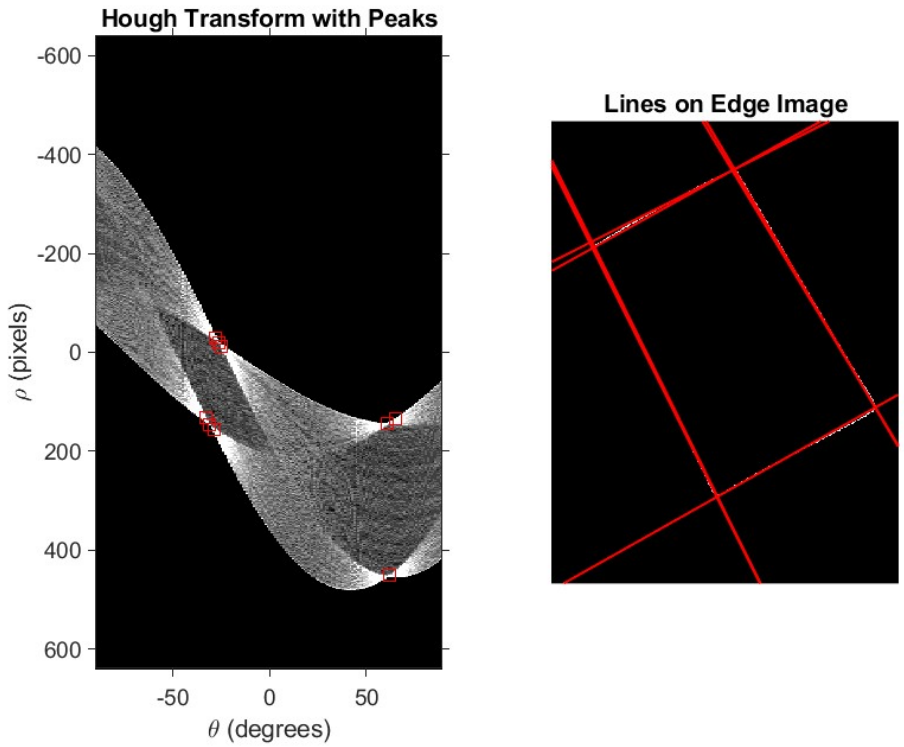

Figure 8: Selected Lines Superimposed on Edge-Detected Image. Red lines represent the detected edges of the paper.

Part 4: Intersections Superimposed on Original Image

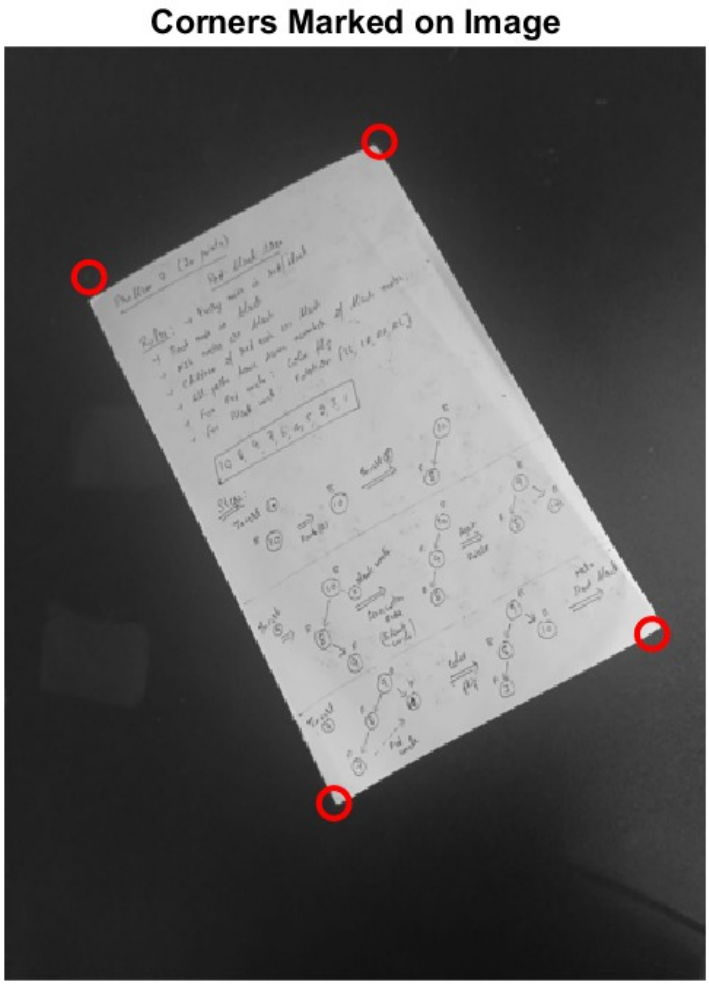

Figure 9: Intersections Superimposed on Original Image. Red circles mark the detected corners of the paper, aligned with the edge-detected image.

Part 5: Rectified Image

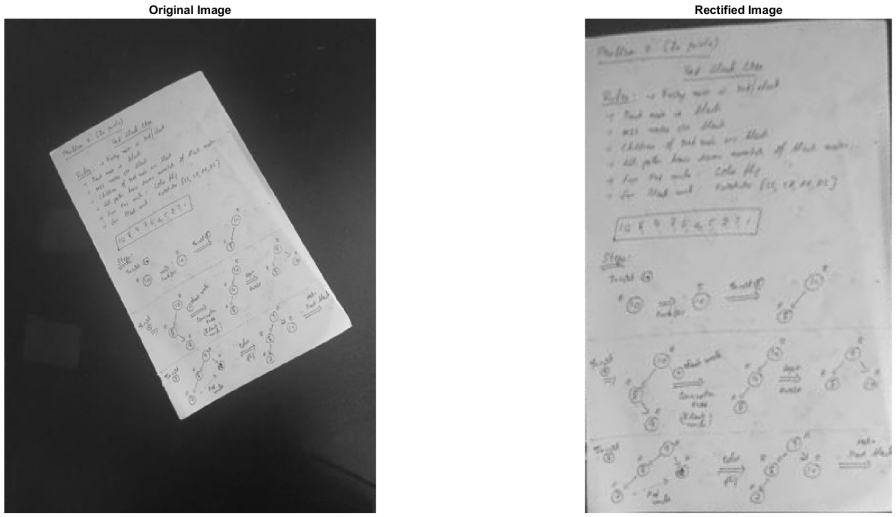

Figure 10: Rectified Image. The paper is rectified using the detected corners and a computed homography matrix.