# Assignment 1 Report

**Eray Taymaz**
**b2210356123**

## Introduction

The goal of this assignment is to implement a perspective correction pipeline that can automatically detect and rectify such distortions. To achieve this, I used edge detection, the Hough Transform, and the RANSAC algorithm to detect the document boundaries, followed by geometric transformations to warp the distorted document into a flat, frontal view. This report explains the methods used, provides visual and quantitative results, and discusses how the quality of the correction can be further improved.

## Methods

### Hough Transform

The Hough Transform is a classical technique used for detecting straight lines in an image. It works by transforming points from the image space into a parameter space, where each point votes for all possible lines that could pass through it. In this assignment, after applying edge detection, I implemented the Hough Transform to identify candidate lines representing the document's edges. Each edge pixel contributes to a sinusoidal curve in the Hough space, and peaks in this space correspond to the most likely lines in the image. By identifying these peaks, I was able to detect strong line candidates. However, since the Hough Transform can detect many redundant or noisy lines, I used RANSAC afterward to refine the line selection and improve accuracy.

### RANSAC

RANSAC is a algorithm used to fit a model to data that contains outliers. In this project, I used RANSAC to refine the line detections obtained from the Hough Transform. While the Hough Transform provides many potential lines, not all of them accurately represent the document edges due to noise or irrelevant structures in the image. RANSAC improves this by randomly sampling subsets of edge points and fitting line models to them. It then evaluates how well these models fit the rest of the data, keeping only those lines that have a large number of inliers. This process helps filter out incorrect detections and retain only the most reliable lines, which are later used to compute the papers boundary.

### Geometric Transformations

Intersection points are computed between pairs of lines and then these points are filtered using convex hull and polygon approximation to extract a quadrilateral representing the papers edges. When quadrilateral is detected, its corners are ordered and a homography matrix is computed to transform the document from its distorted shape to a flat rectangle. "warp_perspective" applies this transformation to the entire image to generate the corrected version, producing a frontal version of the paper. To accurately map pixel values during this transformation, bilinear interpolation is used, which computes each new pixel by blending the values of its four nearest neighbors in the original image.

### How to Improve

Custom implementations of Hough Transform and RANSAC may not be accurate as pre-defined versions of them. This may lower the overall SSIM score by missing some of the accurate lines, leading wrong or not even finding valid quadrilateral. Other thing that can be improved is the hyperparameter testing, refining the parameter values may actually increase the algorithms precision.

# Some Images

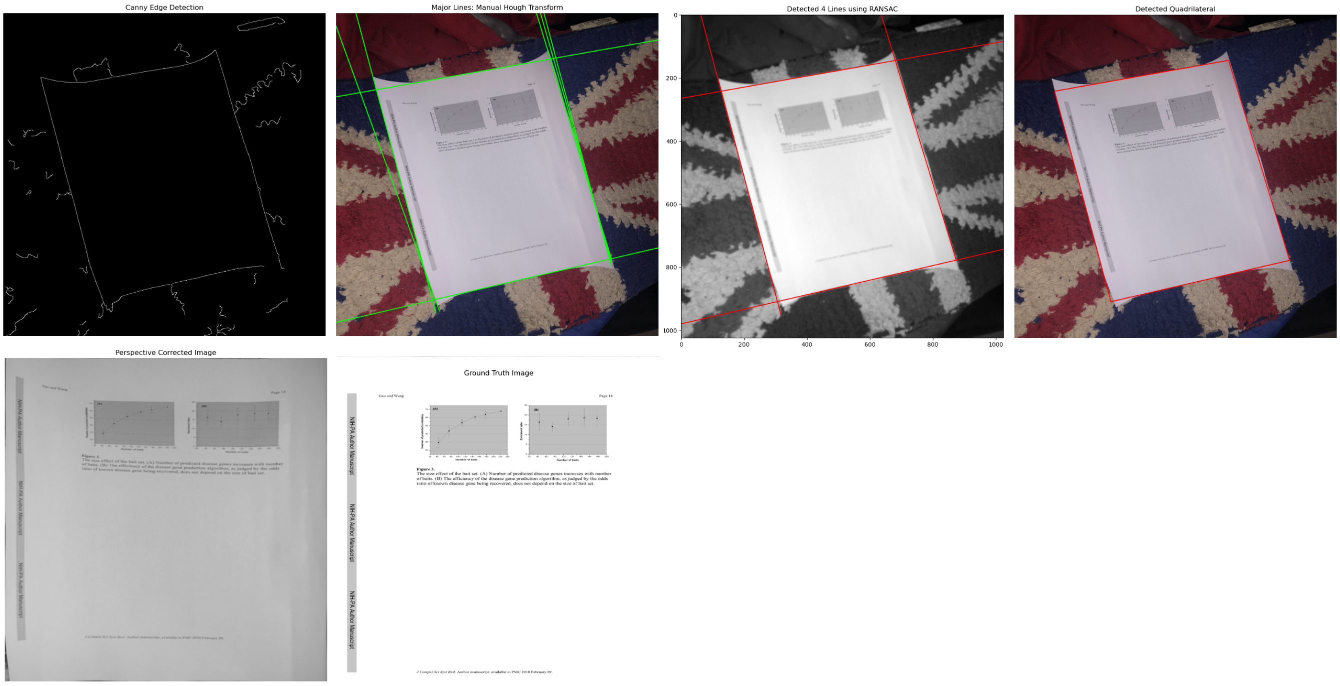


Figure 1: Curved, Index: 2, SSIM: 0.84

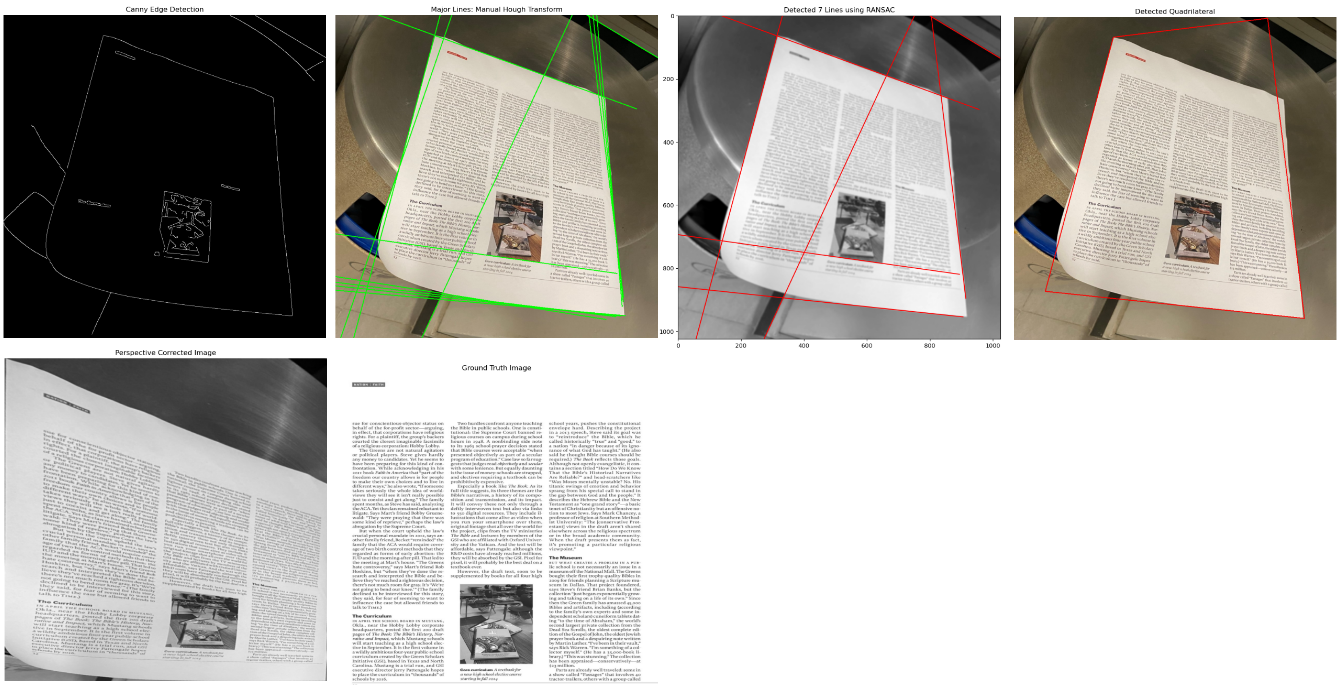


Figure 2: Curved, Index: 96, SSIM: 0.23

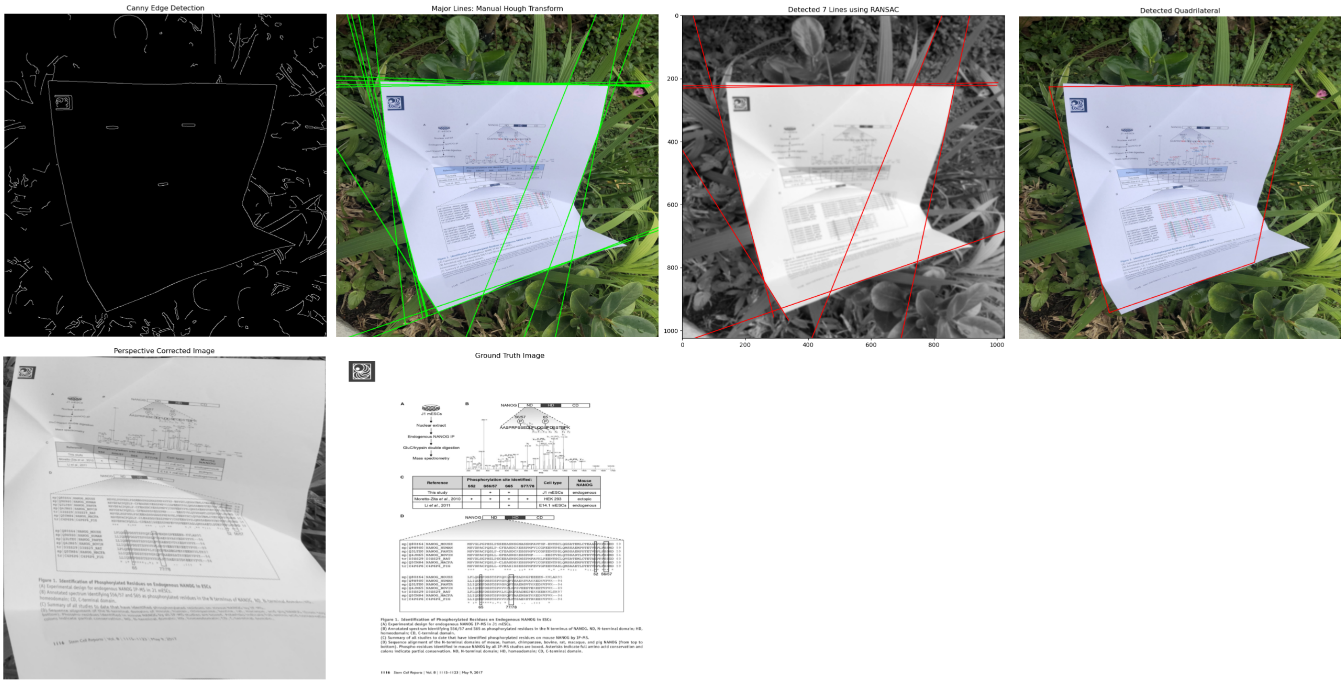


Figure 3: Fold, Index: 32, SSIM: 0.58

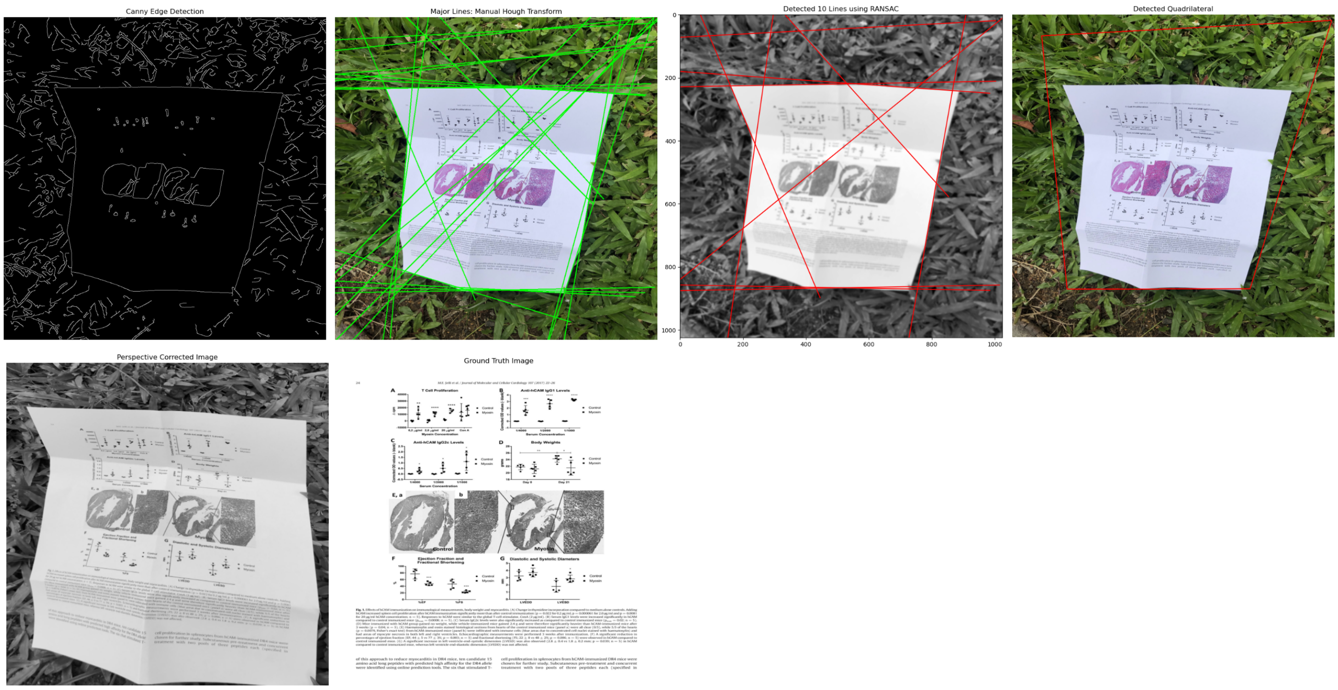


Figure 4: Fold, Index: 61, SSIM: 0.37

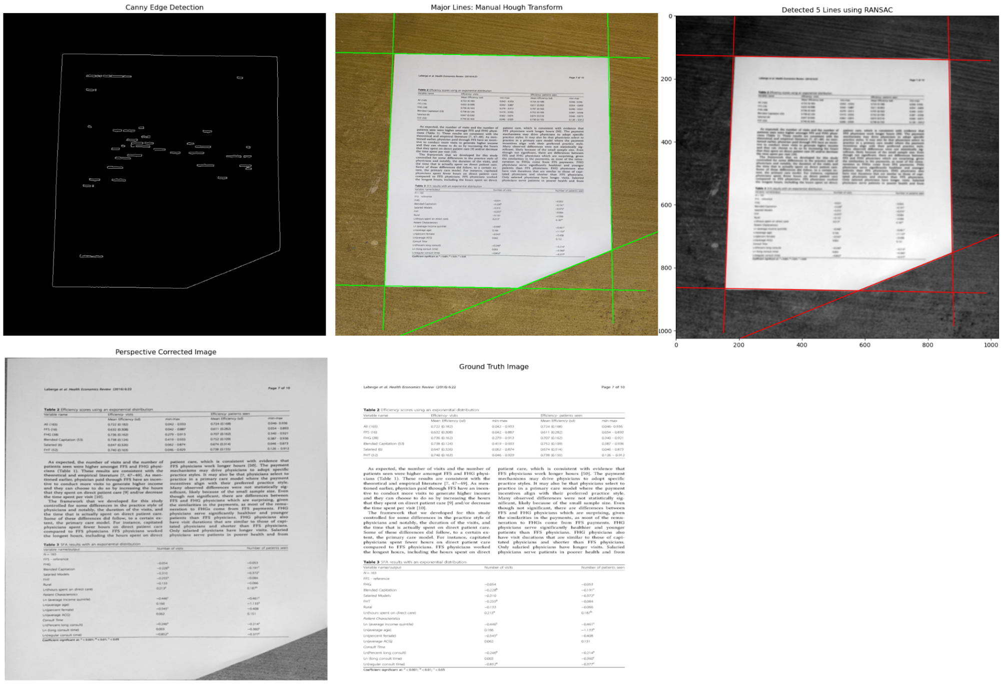


Figure 5: Incomplete, Index: 69, SSIM: 0.58

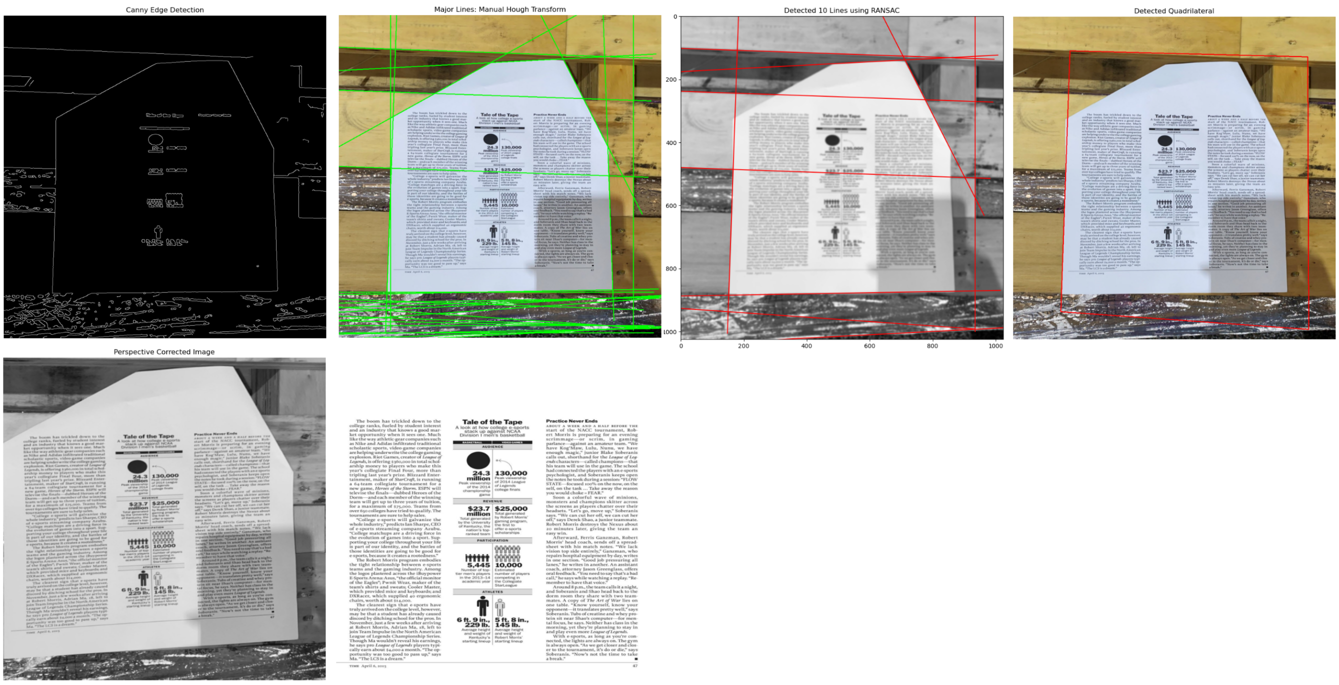


Figure 6: Incomplete, Index: 88, SSIM: 0.29

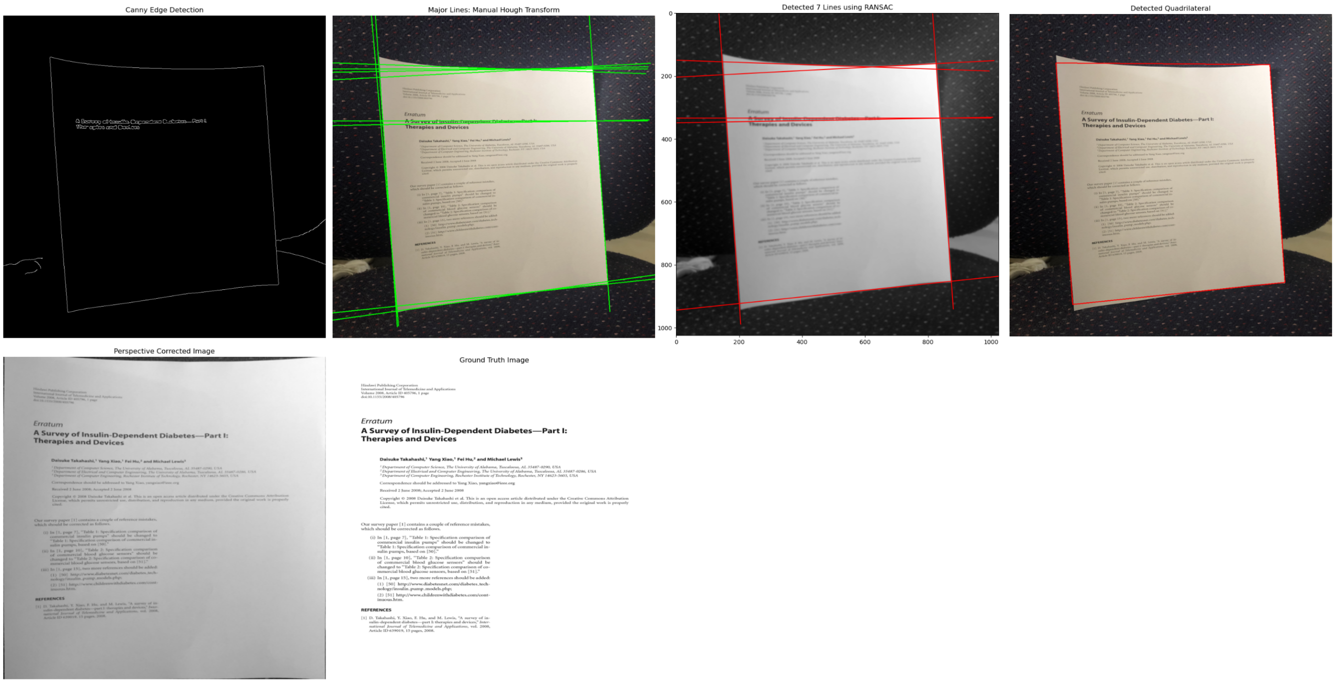


Figure 7: Perspective, Index: 49, SSIM: 0.72

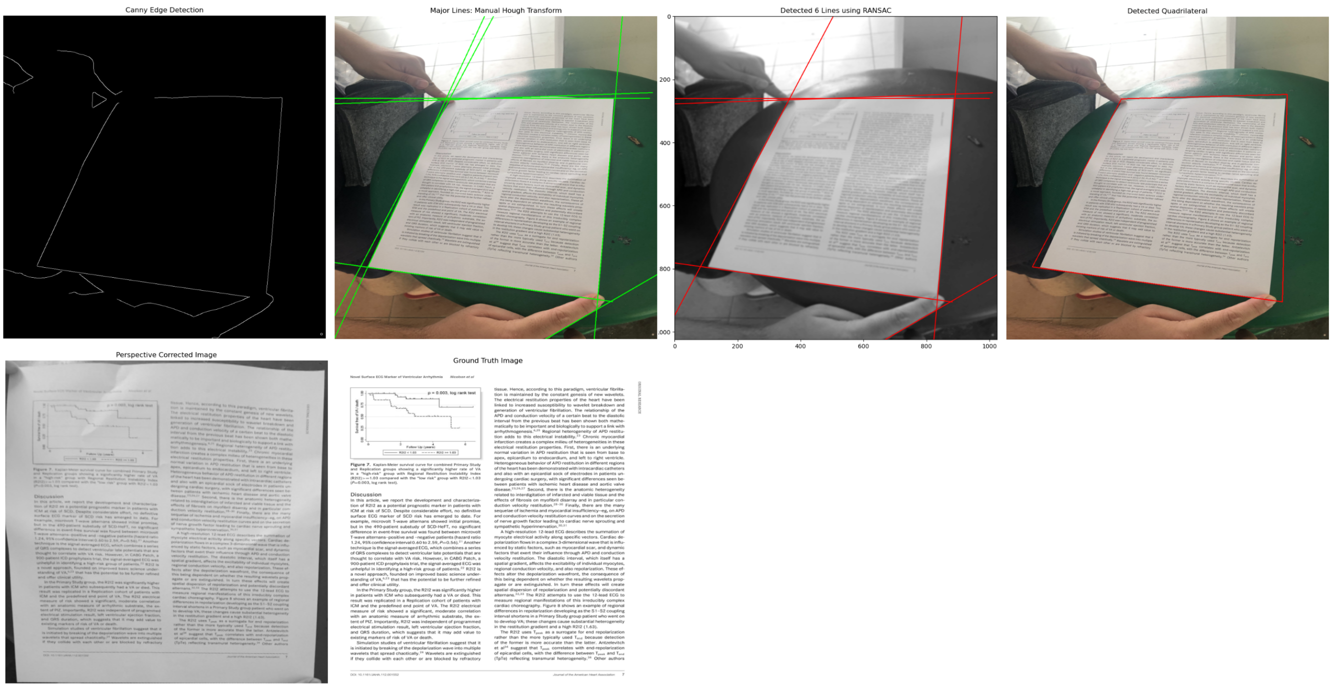


Figure 8: Perspective, Index: 58, SSIM: 0.34

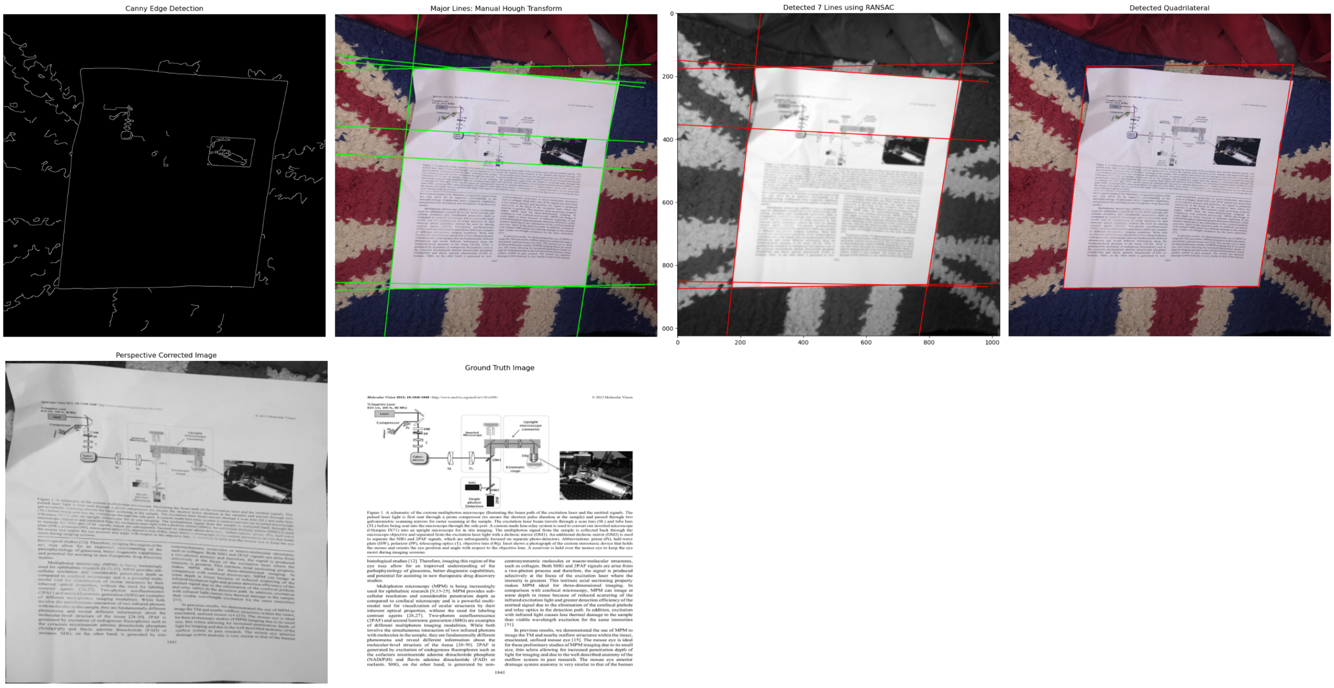


Figure 9: Random, Index: 61, SSIM: 0.46

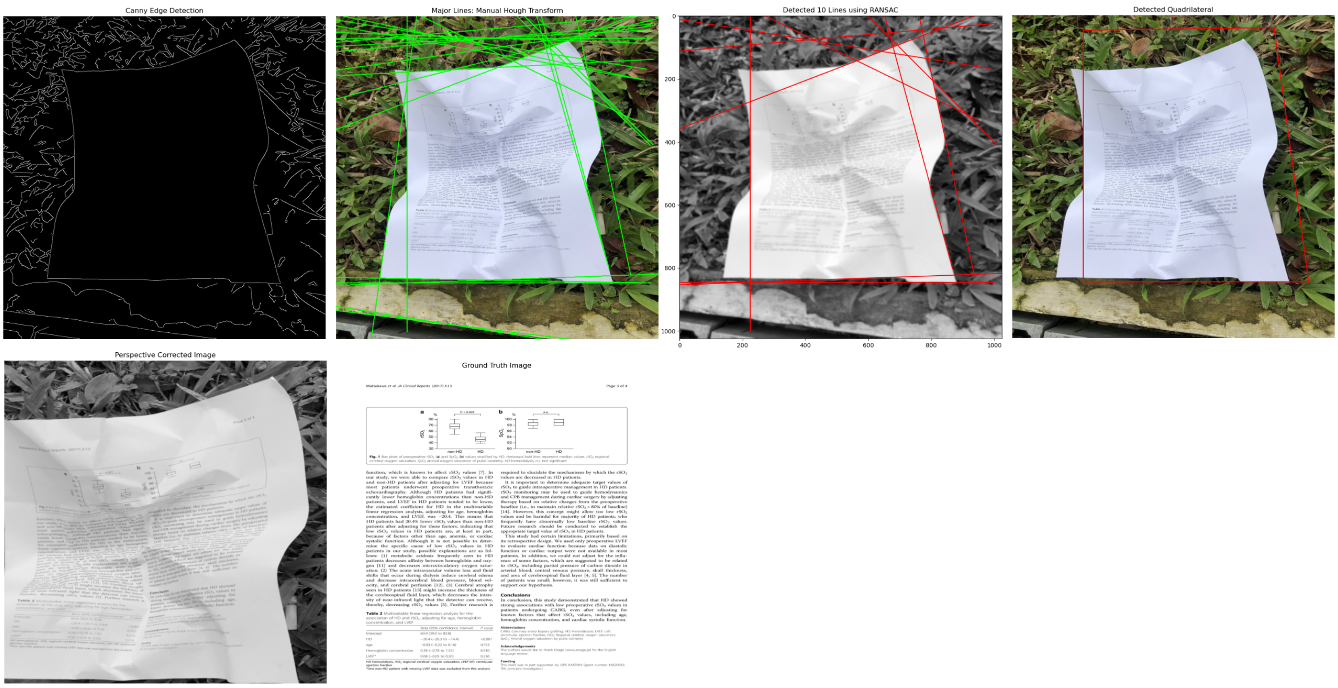


Figure 10: Random, Index: 28, SSIM: 0.37

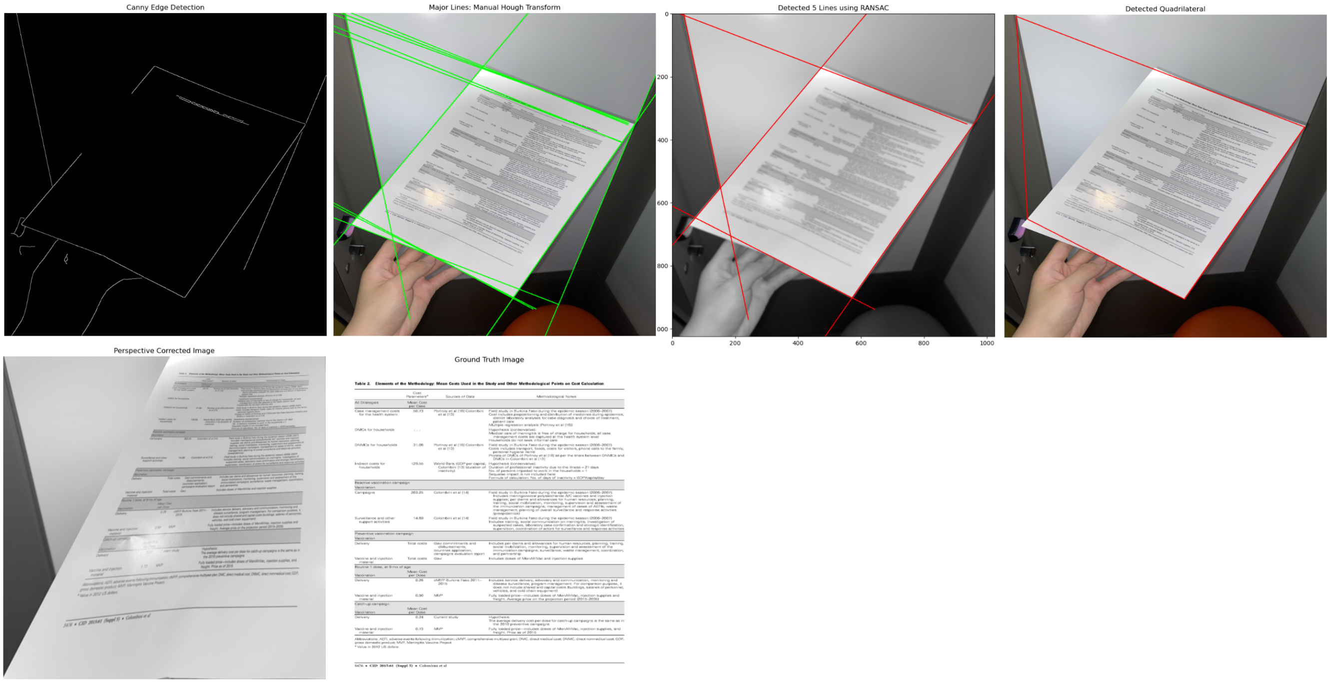


Figure 11: Rotate, Index: 77, SSIM: 0.45

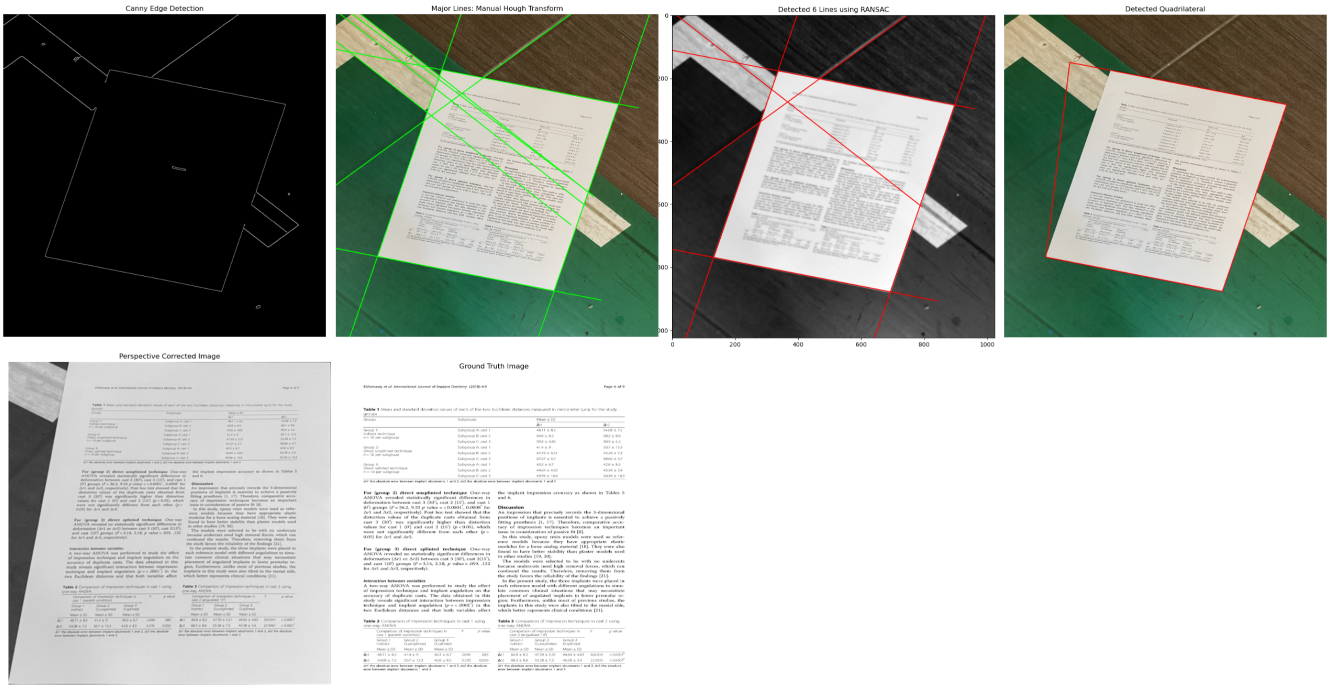


Figure 12: Rotate, Index: 60, SSIM: 0.53

## Results

| | *Average SSIM* |
|---|---|
| Curved | *0.45* |
| Fold | *0.41* |
| Incomplete | *0.39* |
| Perspective | *0.42* |
| Random | *0.40* |
| Rotate | *0.35* |

Table 1: Average SSIM scores per category

To evaluate the performance of the perspective correction system, I tested it on six different categories of distorted document images: curved, fold, incomplete, perspective, random, and rotate. For each image, the corrected version was compared against its corresponding ground truth using the Structural Similarity Index (SSIM), which measures the visual similarity between two images. On average, the system produced SSIM scores in range of 0.3 to 0.5, especially on curved and perspective categories, indicating successful correction. Some challenging cases, such as rotate due to class images contains much more noise compared to other classes, resulted in lower SSIM scores and affected quadrilateral detection. Despite these challenges, the method consistently flattened documents and improved their alignment. Visual results confirmed that the pipeline effectively restored most distorted documents to a readable, frontal view. To improve the SSIM scores, better edge detection and noise removal can be used, and more accurate corner detection might help fix challenging images.