MP5 Report

In this project the objective is to implement the Canny edge detector and evaluate its performance across different test images and parameter configurations. Comparisons are made with classical edge detection techniques including Sobel, Roberts, and Zero-Crossing, with a focus on output quality and clarity.

For the Canny edge detector, I followed a multi-step approach:

- Gaussian Smoothing is performed to remove noise.
- Gradient Computation is performed to detect edges.
- Non-Maximum Suppression is implemented to thin edges.
- Double Thresholding is done to separate strong and weak edges.
- Finally edge linking (hysteresis) is performed to build continuous contours.

The detector was tested finally using different sigma(smoothing) values and non-edge percentages (threshold tuning) to understand its behavior.

Results

Image	Best Params (σ, pct)	Canny Result Summary	Best Performing Detector	Notes on Zero-Crossing
Joy	1.5, 0,6	Clear edges, balanced detail and noise	Roberts	Too dark, edges poorly defined.
Lena	1.5, 0,6	Smooth facial contours captured well	Roberts	Weak output, edges barely visible.
Pointer	1.5, 0,6	Strong pointer outlines, some blue at $\sigma = 2.0$	Roberts	Fails to outline key object lines
Test	1.5, 0,6	Structured, thin edges across features.	Roberts	Lacks contrast, misshapen edges.

Comparative Analysis

- **Canny:** Strong and balanced across most tests but sensitive to parameter tuning.
- **Roberts:** Surprisingly produced the sharpest and clearest results across all images.
- **Sobel:** Edges appeared thick and noisy in low-sigma cases.
- **Zero Crossing:** Consistently underperformed with dim, vague edge outlines.

The canny edge detector, when tuned correctly (sigma =1.5, non edge percentages = 0.6), provided reliable and clean edge maps. However, in this specific test suite, Roberts outperformed Canny in final output quality, producing sharper and more visually accurate contours. The experiment highlights the importance of parameter tuning and the strengths of older detectors in specific cases. Although it depends on the requirement for the system or user.