Comparison of Image Enhancement and Edge-Detection Techniques on the Raspberry Pi 5

Brendan James Joyce & Nuri Yilmazer, Ph.D.

Department of Electrical Engineering and Computer Science

TEXAS A&M UNIVERSITY KINGSVILLE

MCNAIR SCHOLARS PROGRAM

ABSTRACT

In computer vision, edge detection is the characterization of boundaries and thus is paramount to the problem of automated feature extraction in images. This research project presents a comparative analysis of edge detection and image enhancement techniques implemented on the Raspberry Pi 5, a low-cost, resource-constrained platform. Similar comparative analyses focused on industrial settings utilizing top-tier hardware. Instead, this paper addresses the practical implementation and performance of edge detection and image enhancement techniques on affordable hardware, contributing to applications in fields such as surveillance, agriculture, and robotics. Here the performance of various common edge detection algorithms is evaluated, including the Prewitt, Sobel, and Canny operators. The goal is to determine the suitability of these techniques for real-time applications on the Raspberry Pi 5 by assessing their effectiveness in improving image quality, computational efficiency, and adaptability.

INTRODUCTION

Background

- The Raspberry Pi 5 is a small yet powerful computer that has revolutionized research, education, and home application. It boasts a high-performance processor and memory capable of running complex software [1][2].
- Edge-detection is crucial for further research because it identifies boundaries within images, allowing for extraction of essential features and shapes
- Widely used in computer vision, object recognition, medical imaging, and autonomous driving, where understanding contours is vital for accurate decision-making

Problem

- Missing application-specific performance comparisons.
- Many algorithms are developed and tested under ideal conditions in industrial environments.
- Practical implementations on affordable hardware in other environments are underexplored, particularly for real-time applications.

PURPOSE

- Compare edge-detection techniques performed on the Raspberry Pi 5 by image quality, runtime efficiency, and adaptability across image variation**.** By evaluating the various methods by limitations and suitability by applications, this research will inform future researchers about the deployment of edge detection algorithms on similarly resource-constrained environments.

METHODS

- **Implementation**
  - Raspberry Pi 5 with Pi Camera V3
  - Python-3
- **Evaluation**
  - **Efficiency**
    - Runtime (in seconds)
  - **Adaptability**
    - Natural scenes
    - Urban environment
    - Medical
    - Synthetically generated
  - **Image Quality**
    - Presence of noise
    - Subjective

- **Spatial Domain Filters**
  - Mean Filter
  - Median Filter
  - Gaussian Filter
  - Butterworth Low-pass Filter
- **Edge Detection Operators**
  - Ideal High-Pass Filter (IHPF)
  - Sobel
  - Prewitt
  - Canny
  - Laplacian of Gaussian (LoG)

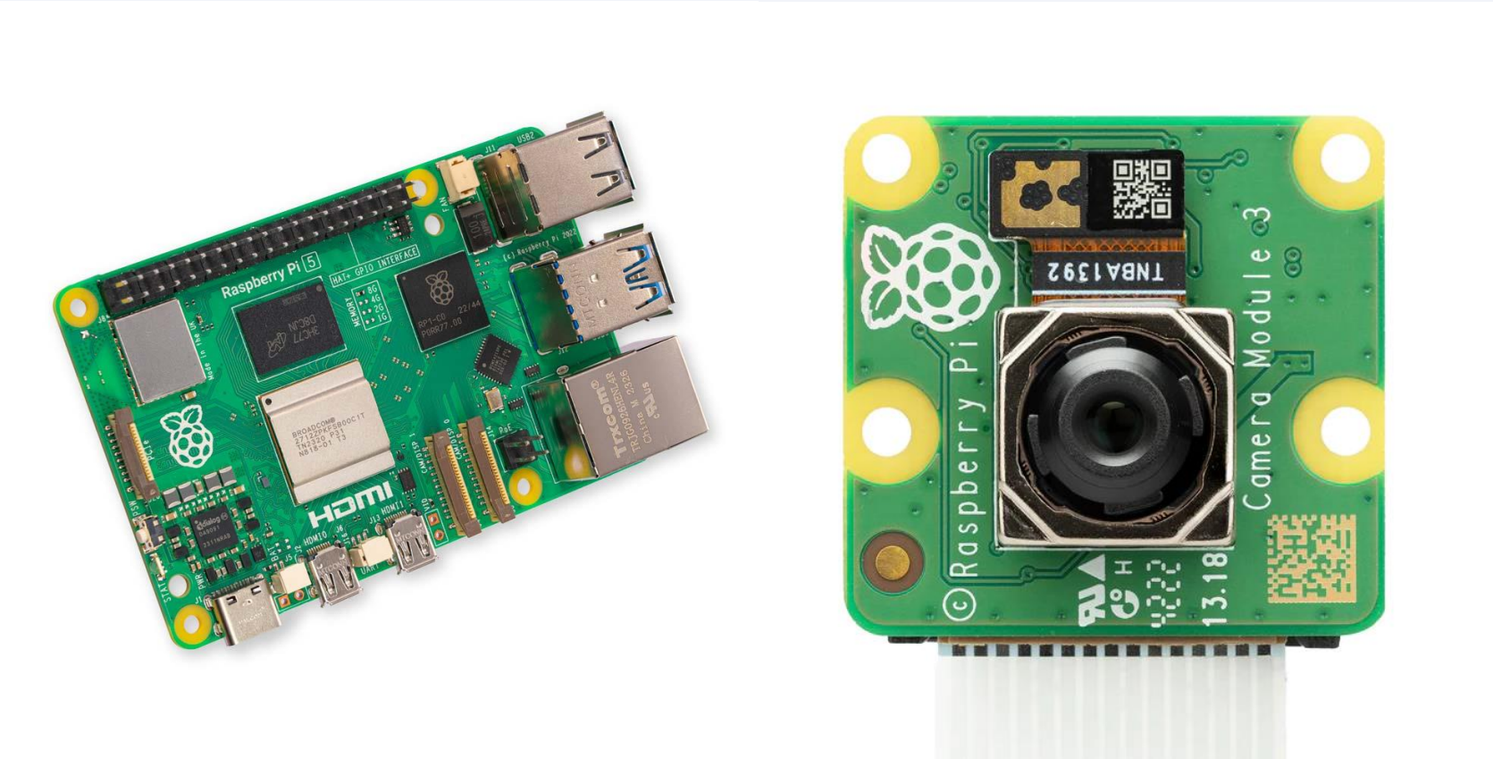

Figure 1 – Raspberry Pi 5 with Cam V3

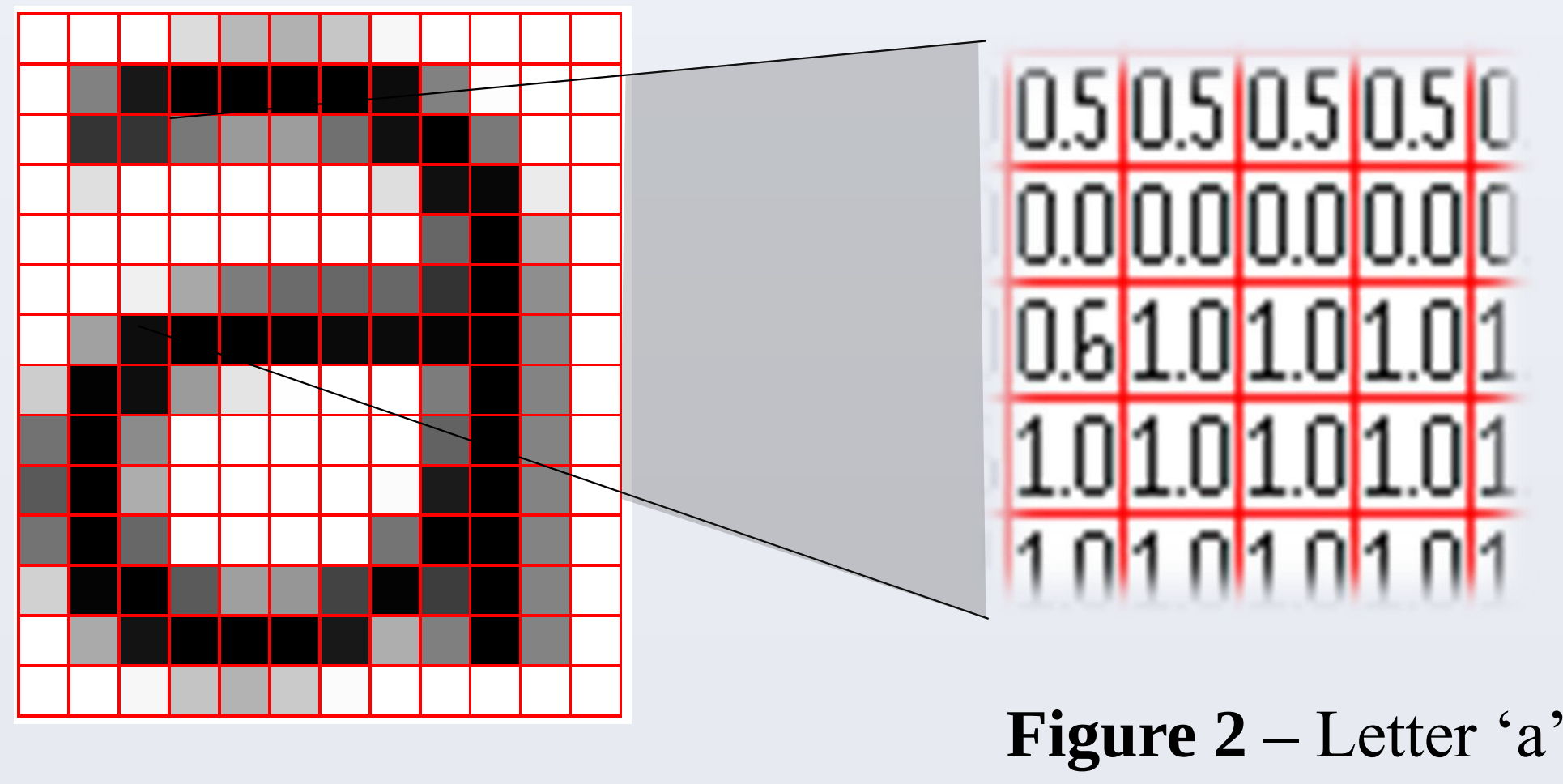

Figure 2 – Letter ‘a’ as a matrix

RESULTS

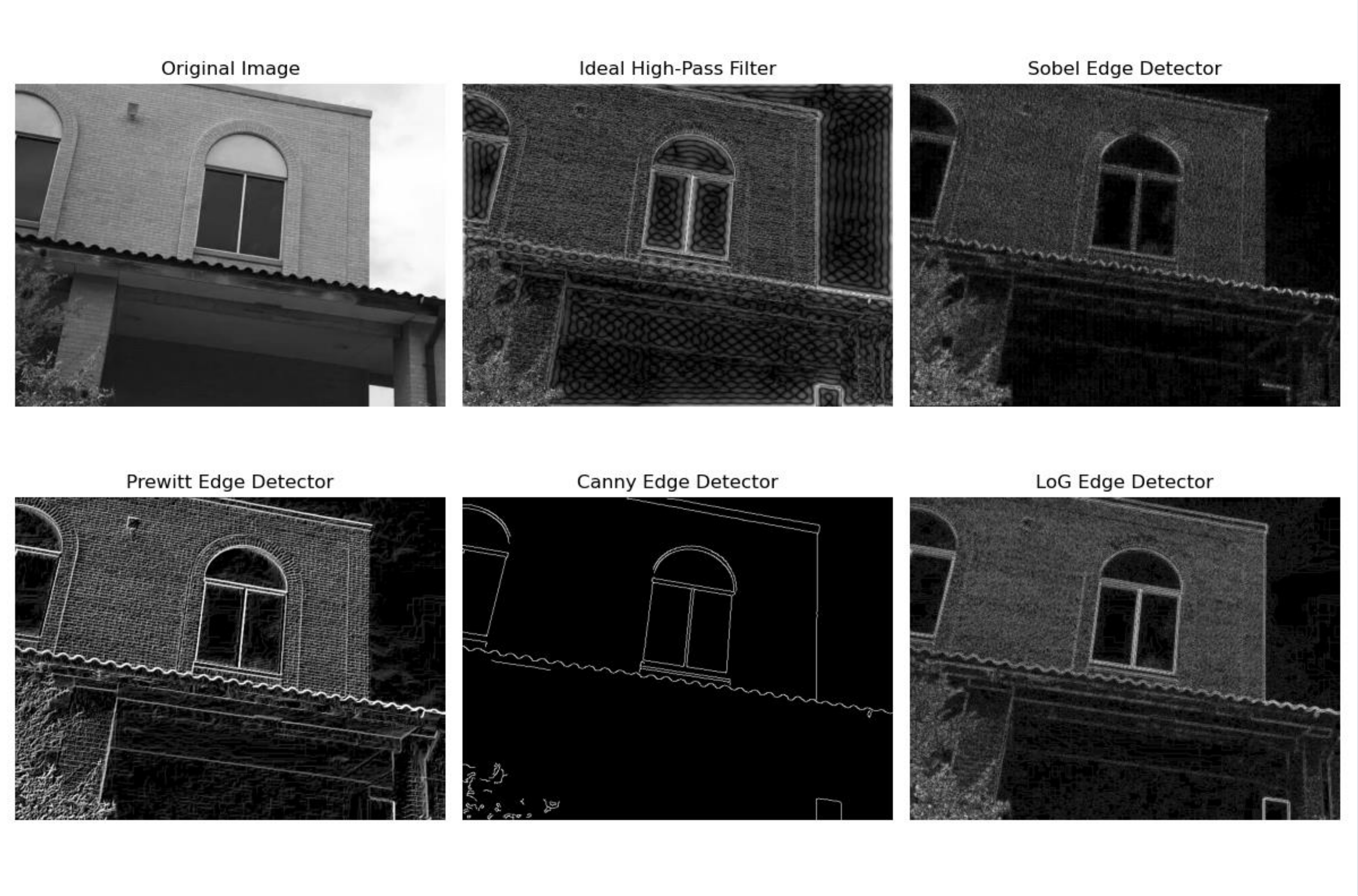

Figure 3 – Edge detection operators applied to urban environment imagery.

Resolution :[640 × 480],

Filters applied clockwise from top-left: Ideal High-Pass Filter, Sobel, Prewitt, Canny, Laplacian of Gaussian (LoG)

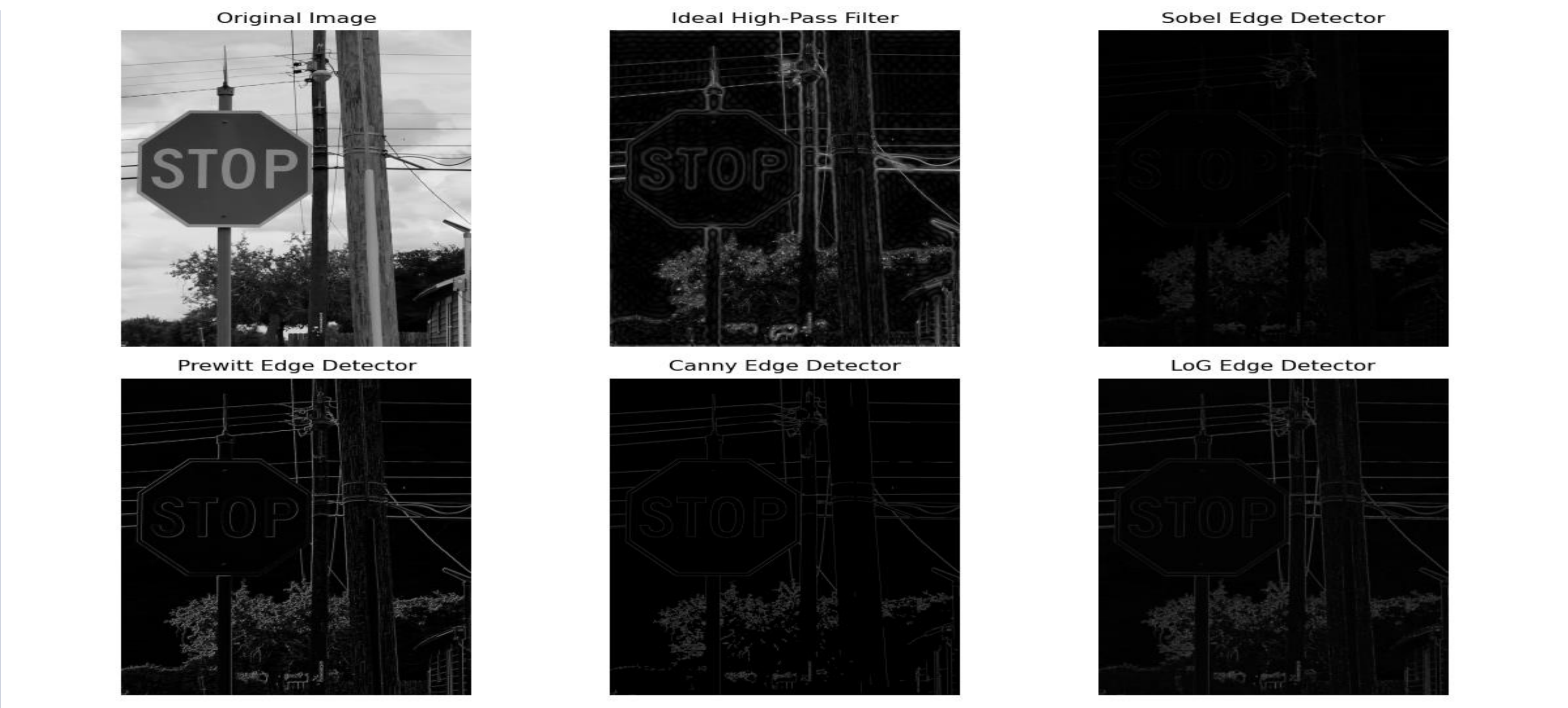

Figure 4 – Edge detection operators applied to stop signs.

Resolution: [3024 × 4032]

Filters applied clockwise from top-left: Ideal High-Pass Filter, Sobel, Prewitt, Canny, Laplacian of Gaussian (LoG)

DISCUSSION

- **Efficiency by Runtime (3024 x 4032 Resolution)**
  - **Spatial Domain Filters**
    - **Mean** – 27.76 ms
    - **Gaussian** – 108.1 ms
    - **Butterworth** – 3601 ms
    - **Median** – 280.1 ms
  - **Edge Detection Algorithms**
    - **IHPF** – 3298 ms
    - **Sobel** – 809.0 ms
    - **Prewitt** – 1098 ms
    - **Canny** – 1175 ms
    - **LoG** – 526.6 ms

CONCLUSIONS

- **Prewitt** and **Sobel** for efficiency; **Canny** for precision when prioritized over speed.
- **Sobel** while timely, less adaptable in complex noises. Fails to detect weak edges. Useful on noisy images.
- **Canny** requires manual tuning of parameters,
- **Prewitt** provides less accurate edges than **Sobel** due to the weighted filter of the latter.
- Edge detection on medical images inconclusive

- **Mean** filtering for speed; **Gaussian** for quality.
- Low-pass filters, **Sobel**, and **LoG** may be best suited for real-time applications.
- **LoG** is adaptable, efficient when not handling particularly noisy images
- The 5x5 filter dimension proved a sufficient balance between noise reduction and detail preservation.

- Limitations identified include the need for further testing of specifications between Raspberry Pi 4 and 5, inclusion of image quality metrics (e.g., PSNR, SSIM), and exploration of machine learning applications.

REFERENCES

SCAN ME

To access the full references, please scan the QR code

Acknowledgements

Nuri Yilmazer, Ph.D.
The McNair Program at Texas A&M Kingsville