NEURAL CANNY-NET EDGE DETECTION WITH SINGLE-IMAGE LEARNING

1 Problem Definition

This project explores the limits of edge detection data efficiency by learning from a single image. Extending the known operator learning approach in [1], I implemented a multi-channel architecture with trainable filters and enhanced non-maximum suppression. The input is a single image and its augmentations; the output is optimized edge detection parameters. Success is measured by quantitative improvement over traditional Canny detection. This investigation challenges conventional deep learning paradigms and could enable applications where collecting large datasets is impractical, such as rare medical conditions or industrial inspection.

2 Related Work

Wittmann and Herl [1] introduced Canny-Net, which reformulates the traditional Canny edge detector into a trainable neural network using known operator learning by embedding the steps involved in Canny Edge Detection [2] as trainable layers, they achieved an 11% improvement in F1 score with only 29 parameters.

DexiNed [3] approached edge detection with a dense architecture using parallel skip-connections and a specialized loss function that addresses class imbalance through differential weighting of edge versus non-edge pixels.

My work extends Canny-Net with multi-channel architecture and enhanced non-maximum suppression while adopting a modified version of the BDCN (Bi-Directional Cascade Network) loss function from DexiNed. This loss function dynamically balances edge and non-edge pixels using class-specific weights calculated from their proportions.

3 Method

My approach extends the Canny-Net architecture to handle the extreme constraint of single-image learning while incorporating elements from DexiNed's loss function. Figure 1 shows the overall methodology.

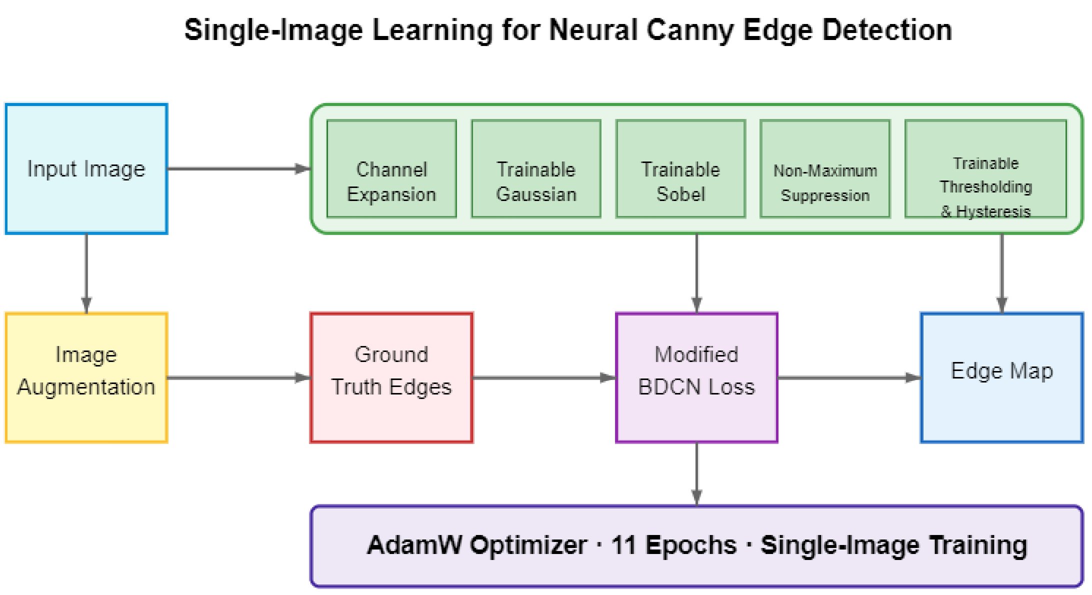

Figure 1: End-to-End Architecture of Single-Image Training for Neural Canny Edge Detection

3.1 Loss Function

I adopted a modified BDCN loss function [3] that addresses class imbalance by dynamically weighting edge and non-edge pixels based on their proportions in the image:

$$\texttt{mask[mask > 0.5]} = 1.0 \times \frac{\texttt{num\_negative}}{\texttt{num\_positive} + \texttt{num\_negative}}$$
$$\texttt{mask[mask} \leq \texttt{0.5]} = 1.1 \times \frac{\texttt{num\_positive}}{\texttt{num\_positive} + \texttt{num\_negative}}$$

3.2 Dataset Generation

From a single image of the John Curtin School building, I created a comprehensive dataset through:

- **Standard Augmentations:** Random rotations (-30° to 30°), horizontal and vertical flips, and brightness/contrast adjustments.
- **Challenging Scenarios:** Low contrast, noise, complex textures, partial occlusions, and gradient illumination variations.
- **Train/Test Split:** 80% for training, 20% for testing.

4 Results

My single-image-trained neural Canny edge detector achieved an F1 score of 0.5054, demonstrating that meaningful edge detection parameters can be learned from extremely limited data.

4.1 Quantitative Analysis

Comparing my model with the traditional Canny edge detector in different test scenarios revealed consistent improvements:

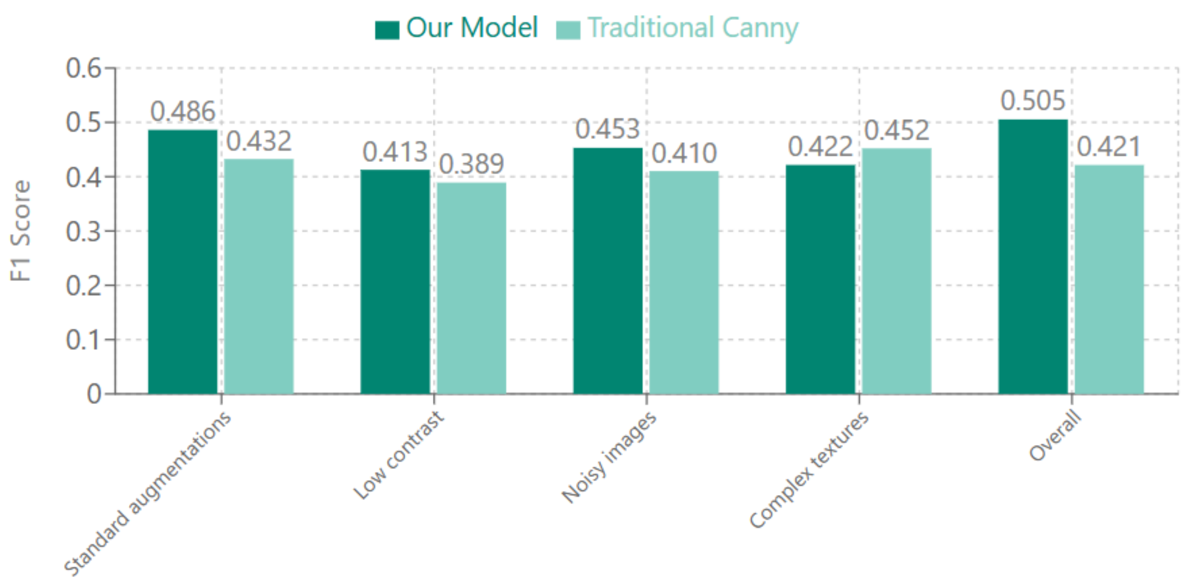

Figure 2: F1 Score Comparison: My Model vs Traditional Canny

My model outperformed the traditional algorithm in most scenarios, particularly for low-contrast and noisy images where adaptivity provides an advantage.

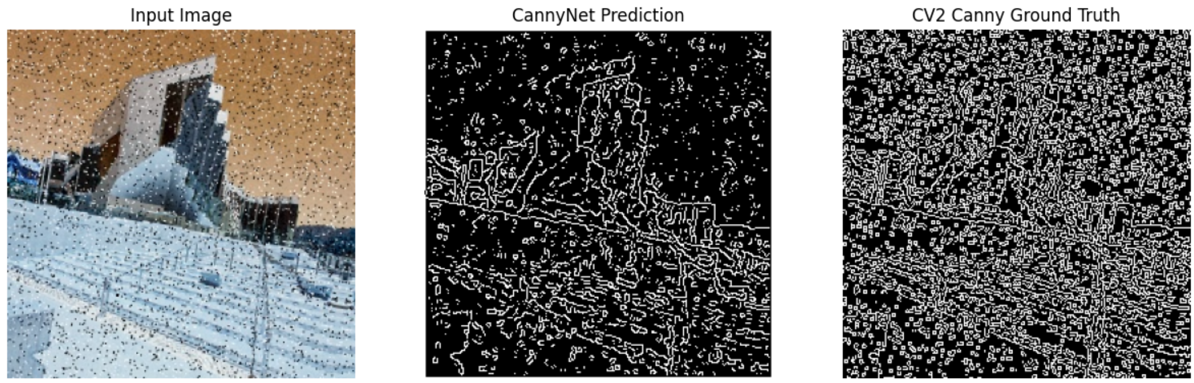

Figure 3: Visual Comparison: Noisy Input Image with my CannyNet model and CV2 Canny Edge Detection Results

4.2 Learning Rate Optimization

I tested multiple learning rates to identify optimal training parameters, with 2e-4 providing the best balance between performance (F1 score of 0.5054) and stability.

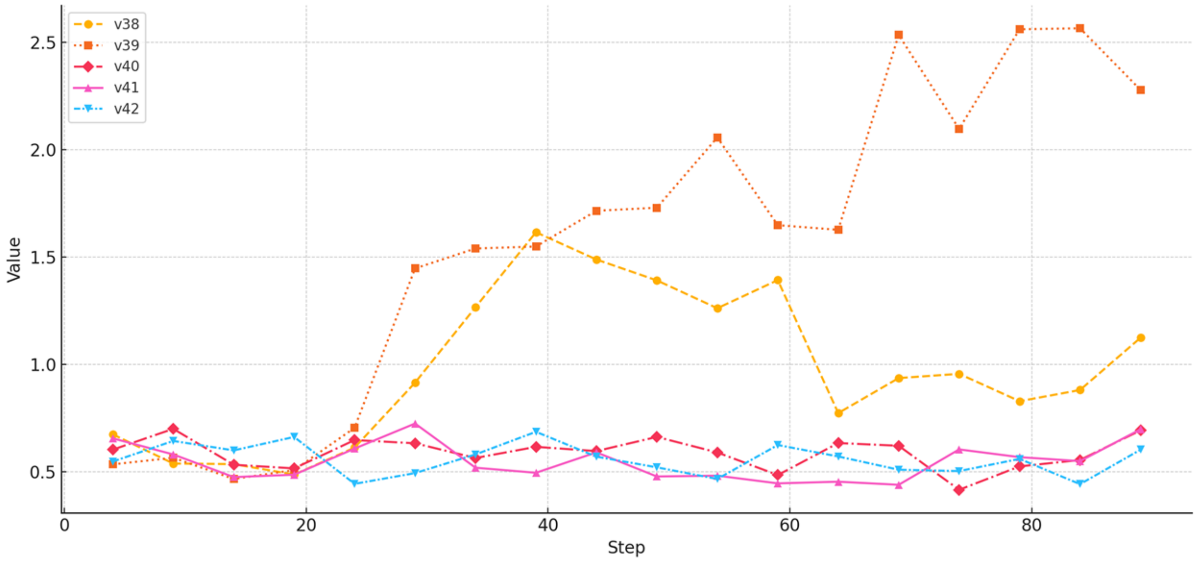

Figure 4: Training Progress over varying Learning Rates

Table 1: Model Performance with Varying Learning Rates

Version	Learning Rate	F1 Score	Precision	Recall	Stability	Overall Performance
38	5e-4	0.5000	0.5851	0.4538	Moderate	Baseline
39	1e-3	0.4917	0.5840	0.4422	Volatile	Not Recommended
40	2e-4	0.5054	0.5856	0.4619	Good	Best Performance
41	1e-4	0.4880	0.5842	0.4365	Stable	Slightly Suboptimal
42	3e-4	0.4676	0.5795	0.4098	Moderate	Less Effective

4.3 Ablation Study

The contributions of individual components were evaluated through an ablation study:

Table 2: Ablation study showing the impact of each component on the F1 score.

Model Configuration	**F1 Score**
Full model	0.5054
Without multi-channel architecture	0.4231
Without enhanced NMS	0.4453
Without BDCN loss	0.4105
Without hysteresis	0.4522

The multi-channel architecture and modified BDCN loss function provided the most significant contributions to performance, confirming that my extensions to the original Canny-Net were effective for the single-image learning task.

5 Reflection

This project raises important ethical considerations. First, single-image learning has privacy benefits by reducing data collection needs, but could enable rapid development of surveillance systems with minimal training data. Second, while potentially valuable for assistive technologies like navigation aids for visually impaired individuals, the model's inconsistent performance with complex textures creates reliability concerns that could impact safety in critical applications.

6 Conclusion

I demonstrated that meaningful edge detection parameters can be learned from a single image by extending Canny-Net with significant improvements. My multi-channel architecture, enhanced non-maximum, and modified BDCN loss function collectively improved performance by up to 14.1% in ablation studies. The model achieved an F1 score of 0.5054, outperforming traditional Canny edge detection particularly in low contrast and noisy scenarios. Future work could explore transfer learning approaches, investigate data quality versus quantity tradeoffs, and develop adaptive inference techniques to enhance generalization to more diverse architectural styles.

References

[1] J. Wittmann and G. Herl. Canny-net: Known operator learning for edge detection. In *12th Conference on Industrial Computed Tomography (iCT) 2023*, volume 28, Furth, Germany, February 27–March 2 2023. doi: 10.58286/27751. URL `https://doi.org/10.58286/27751`.

[2] OpenCV Team. Canny edge detection. `https://docs.opencv.org/4.x/da/d22/tutorial_py_canny.html`, 2024. Accessed: 2025-04-23.

[3] X. Soria, E. Riba, and A. Sappa. Dense extreme inception network: Towards a robust cnn model for edge detection. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1912–1921, Los Alamitos, CA, USA, mar 2020. IEEE Computer Society. doi: 10.1109/WACV45572.2020.9093290. URL `https://doi.ieeecomputersociety.org/10.1109/WACV45572.2020.9093290`.