Critical Applications: Enhancing Gravitational Wave Data Quality through Glitch Classification

Anushka Doke

13 December 2024

Abstract

Gravitational wave (GW) detectors, such as LIGO and Virgo, are highly sensitive instruments that not only detect astrophysical signals but also capture a significant amount of background noise. This noise often manifests as short-lived artifacts, known as "glitches," which can obscure or mimic signals. In this project, I aim to improve GW data quality by developing a deep learning model to classify various types of glitches using data from Gravity Spy. By testing models including convolutional neural networks (CNNs), this study seeks to determine an effective approach for glitch classification. Improved glitch identification will help reduce false positives, enabling more accurate and rapid detection of true GW events.

1 Introduction

Gravitational waves (GWs), ripples in spacetime caused by cosmic events, are critical for understanding astrophysical phenomena. The detection of gravitational waves [1] represents a significant milestone in physics, enabling researchers to study phenomena such as black hole mergers and neutron star collisions. Figure 1 shows the first-ever detection of a gravitational wave event, GW150914 [1]. Detecting these waves allows scientists to probe the universe in ways previously impossible, revealing information about the structure, composition, and evolution of celestial objects. Ground-based gravitational wave detectors such as Advanced LIGO and Advanced Virgo are highly sensitive instruments. Thus, they inadvertently capture significant background noise, including glitches. Glitches are transient noise artifacts. They pose significant challenges as they can obscure or mimic true signals, as seen during the detection of the binary neutron star merger GW170817 [2]. The presence of a loud glitch in the LIGO-Livingston data delayed prompt detection of the event 2. Efficient classification and mitigation of glitches are therefore essential for improving GW data quality, reducing false positives, and enabling faster identification of true signals to promote multimessenger astronomy.

This project aims to address these challenges by developing a deep learning model capable of classifying various types of glitches using data from the Gravity Spy project. By exploring transfer learning, this study seeks to enhance the robustness and accuracy of glitch classification.

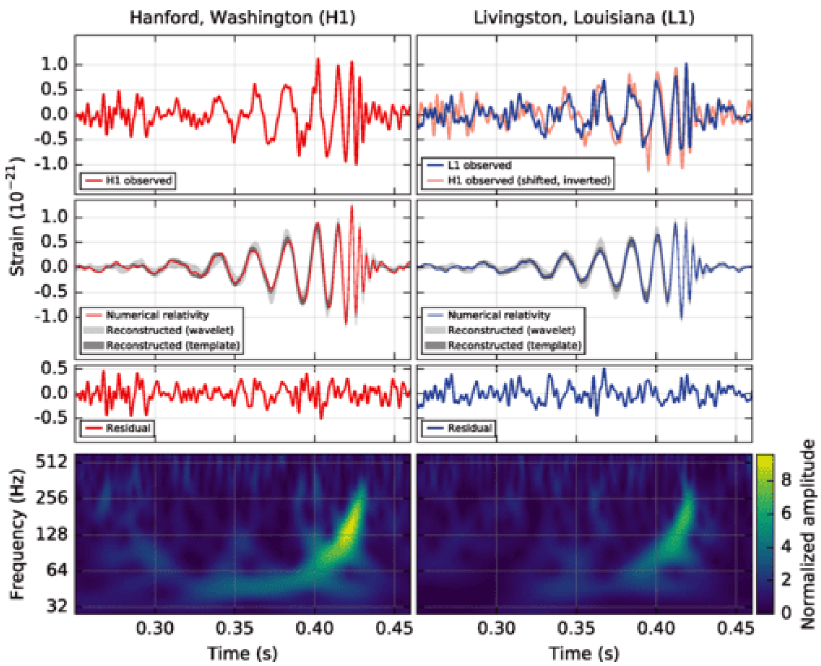


Figure 1: An example of a gravitational wave signal. The first signal detected by the LIGO detectors, GW150914. The topmost row show the signal in time-domain as observed by the two LIGO detectors in Hanford and Livingston. The bottommost row shows the signal in the frequency domain [1].

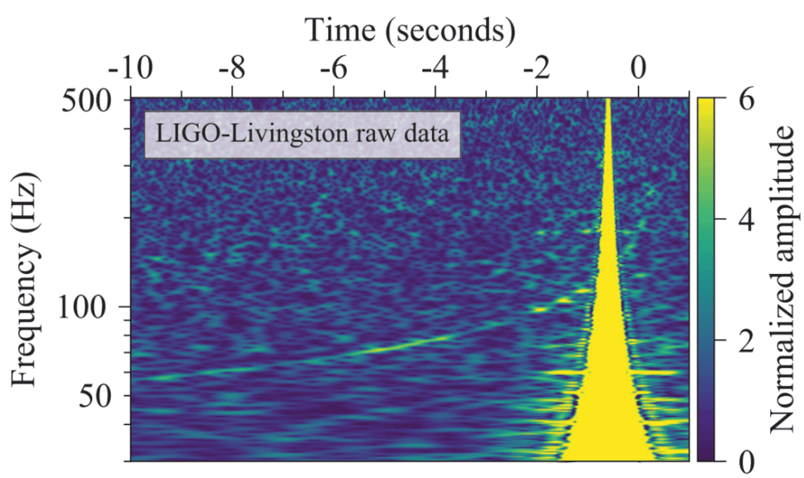


Figure 2: Glitch in LIGO-Livingston data at the time of the signal GW170817 [2].

2 Related Work

The classification of glitches in gravitational wave data has been a focus of recent studies, utilizing both traditional and advanced machine learning techniques. D. George et al. (2017) [3] introduced the application of deep learning combined with transfer learning for glitch classification. By using spectrograms of glitches labeled by Gravity Spy, they demonstrated that knowledge from pre-trained models on real-world object recognition could be effectively transferred for glitch detection. Their methodology achieved state-of-the-art accuracy of 98.8% and showed that transfer learning significantly reduced training time and enabled the use of small, unbalanced datasets.

Building on this, T. Fernandes et al. (2023) [4] investigated the use of convolutional neural networks for classifying glitches. They explored both supervised and self-supervised approaches, employing transfer learning to fine-tune pre-trained models on the Gravity Spy dataset. Their findings revealed that transfer learning improved classification performance, achieving an F1 score of 97.18% for supervised models. These models demonstrated strong generalization capabilities, performing well on data from LIGO-Virgo's O3 run despite being trained on data from earlier runs (O1 and O2). This highlighted the potential of transfer learning for both glitch and signal classification tasks.

Y. Wu et al. (2024) [5] proposed an advanced classifier aimed at overcoming the limitations of existing architectures, particularly in handling the increased complexity of glitches observed during LIGO's O4 run. They introduced a fusion strategy for multi-time window inputs, label smoothing to counter noisy labels, and an attention module to enhance interpretability. Their work addressed the evolving challenges in glitch classification, ensuring robust performance despite the diverse and complex nature of the glitches.

Together, these studies outline the significant progress in using deep learning for glitch classification.

3 Problem Definition

The primary goal of this study is to develop a machine learning model capable of accurately classifying glitches present in gravitational wave data. This involves designing a model that can map input images of glitches to their respective categories while minimizing classification errors. Achieving this objective is critical for enhancing the performance of gravitational wave detection pipelines.

4 Methodology

We use transfer learning with Inception V3 [6], a pre-trained deep learning model. The choice of Inception V3 is motivated by its proven capability to capture intricate features in image classification tasks. The model architecture includes global average pooling and a dense softmax layer tailored for multi-class classification 3. The Adam optimizer was employed

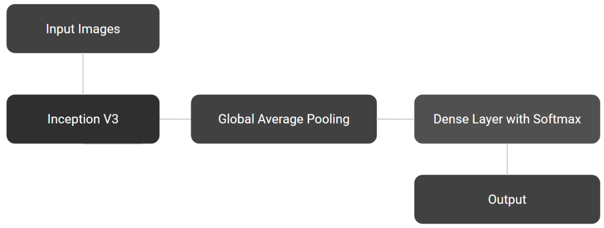


Figure 3: Illustration of the model architecture.

with a learning rate of 0.001. Sparse categorical cross-entropy served as the loss function since it is suitable for multi-class classification.

The implementation utilized TensorFlow 2.18.0 and Keras 3.7.0, executed on an NVIDIA A100 GPU to expedite training on the large dataset. Three distinct configurations were tested to evaluate the model's adaptability to various data settings (see Table 1):

Analysis-I: The batch size was set to 32, and the input images retained their original resolution of 569 x 479 pixels. This configuration aimed to assess the baseline performance of the model on the original unbalanced dataset.

Analysis-II: The batch size was increased to 64 while keeping other parameters unchanged. This adjustment was made to evaluate the impact of batch size on the model's performance.

Analysis-III: A reduced image size of 448x448 pixels was used. A new balanced dataset was created where each class contained 2000 images. For underrepresented classes, images were duplicated, while classes with an excess were randomly sampled to meet the threshold. This was done to study if removing class imbalance improved the model's performance.

We used the open-source weights from Inception V3 to initialize our model, then fine-tuned (re-trained) the model on our training dataset of glitches.

	Analysis-I	Analysis-II	Analysis-III
Batch size	32	64	64
Epochs	30	30	30
Image size	(569, 479)	(569, 479)	(448, 448)
Training data	original imbalanced dataset		each class = 2000 images

Table 1: Hyperparameters for the three analyses.

5 Data

We used data from Gravity Spy dataset [7, 8], a publicly available repository comprising 22 glitch classes (se Figure 4). The dataset was partitioned into training (80%), validation (10%), and test (10%) subsets. Each class in the dataset was represented by a varying number of samples. Table 2 shows the number of samples per class in each of the datasets.

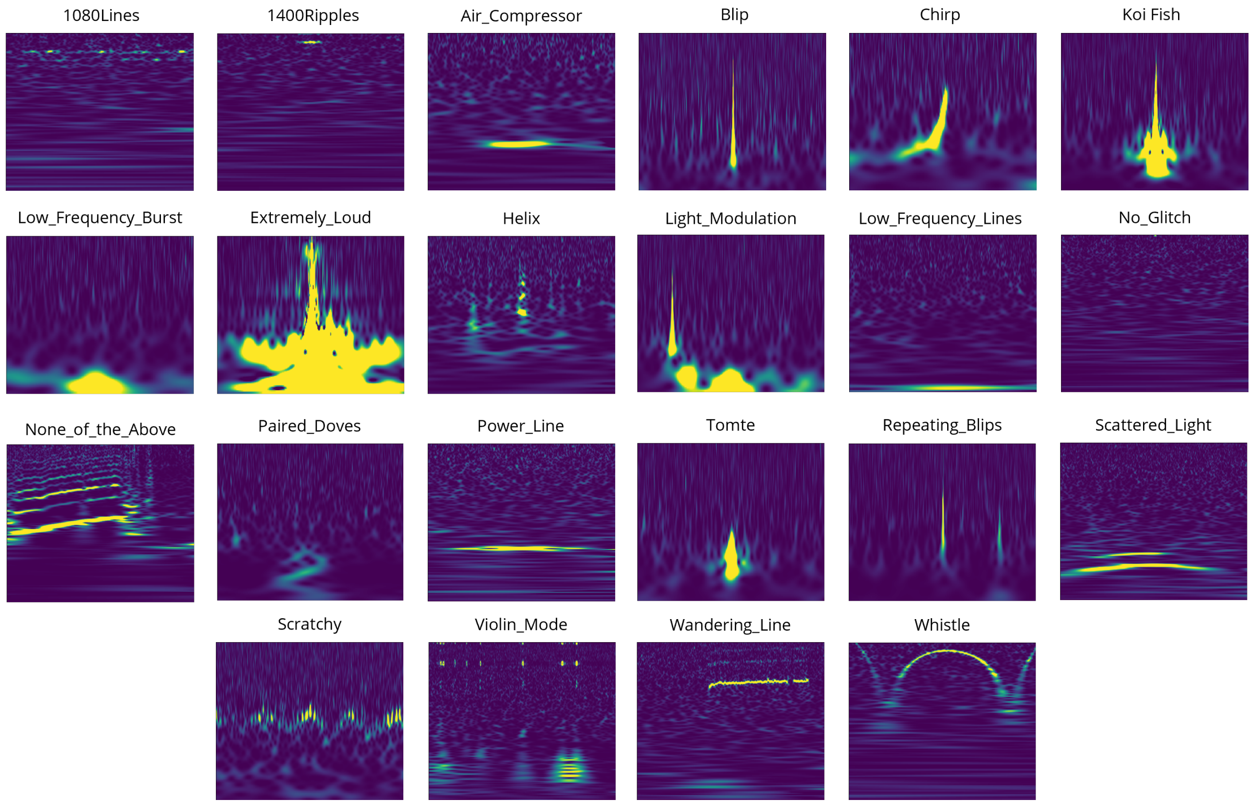


Figure 4: All 22 classes of glitches used in this analysis. The y-axis is the frequency and x-axis is the time. The colormap shows the signal-to-noise ratio.

Evaluation metrics included accuracy, confusion matrices, ROC curves, and F1 score. Accuracy provided an overall measure of classification performance, while confusion matrices offered insights into specific class-level misclassifications. ROC curves were used to assess the model's ability to distinguish between classes at various classification thresholds. The F1-score balanced precision and recall.

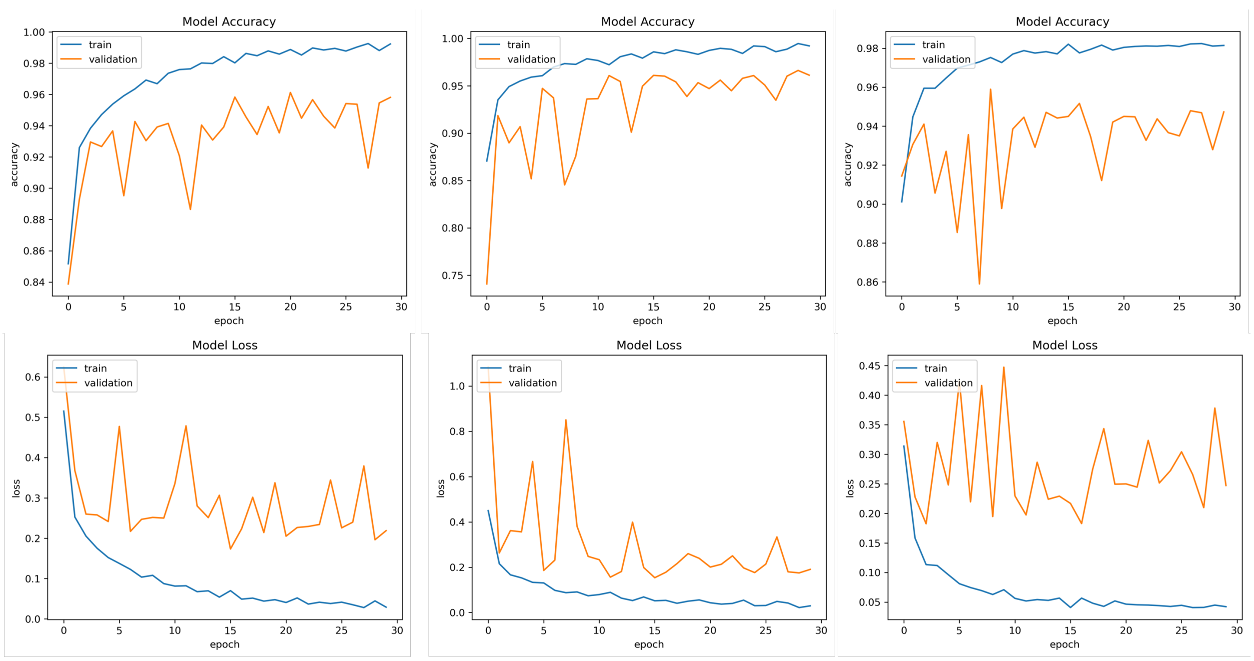
Class	Train	Validation	Test	Class	Train	Validation	Test
1080Lines	916	196	200	None_of_the_Above	228	52	44
1400Ripples	236	52	36	No_Glitch	428	88	84
Air_Compressor	164	32	36	Paird_Doves	76	16	16
Blip	5096	1100	1092	Power_Line	1256	268	272
Chirp	164	36	40	Repeating_Blips	740	164	148
Extremely_Loud	1264	268	256	Scattered_Light	1232	272	268
Helix	780	168	168	Scratchy	948	200	200
Koi_Fish	1992	424	408	Tomte	292	68	52
Light_Modulation	1444	292	312	Violin_Mod	1136	256	256
Low_Frequency_Burst	1748	376	360	Wandering_Line	116	24	28
Low_Frequency_Lines	1260	264	264	Whistle	832	184	180

Table 2: The number of glitches per class in the training, testing and validation datasets.

6 Results and Analysis

The results showed interesting trends across the tested configurations. The first analysis exhibited overfitting, with high training accuracy but low validation accuracy. A batch size of 64 and an image resolution of 569x479 yielded the best results. An improved generalization was observed, with a significant reduction in overfitting. The third configuration, which reduced the input resolution to 448x448, achieved performance comparable to the first analysis, with reduced computational overhead. Figure 5 compares the accuracy and loss across 30 epochs for all three analyses.

Error analysis highlighted recurring patterns of misclassification. For instance, frequent misclassifications occurred between glitch types with overlapping spectral features, such as “Blip” and “Repeating Blips.” Similarly, “Low-Frequency Burst” and “Low-Frequency Lines” were often confused due to their shared frequency-domain characteristics. Balancing the dataset mitigated overfitting to majority classes but introduced new misclassifications, such as that between “Paired Doves” and “Extremely Loud” glitches. The “Paired Doves” and “None of the Above” glitch classes had the lowest F1 scores across all three analyses. Figures **??**, show the confusion matrices for Analysis I, II and III, while Figures **??** depict the ROC curves.



Analysis-I Analysis-II Analysis-III

Figure 5: Accuracy and loss for each of the three analyses.

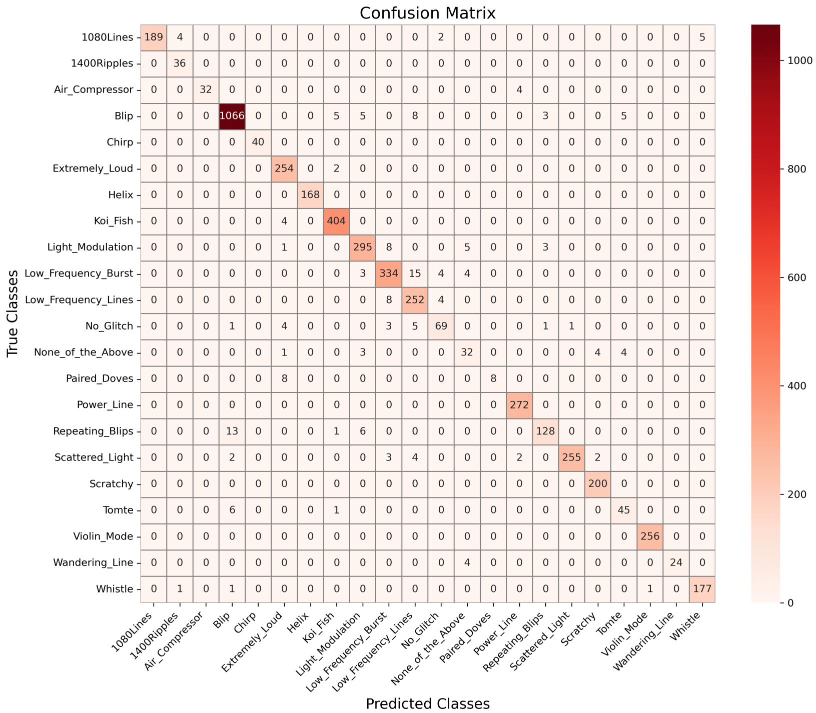


Figure 6: Confusion Matrix for Analysis I.

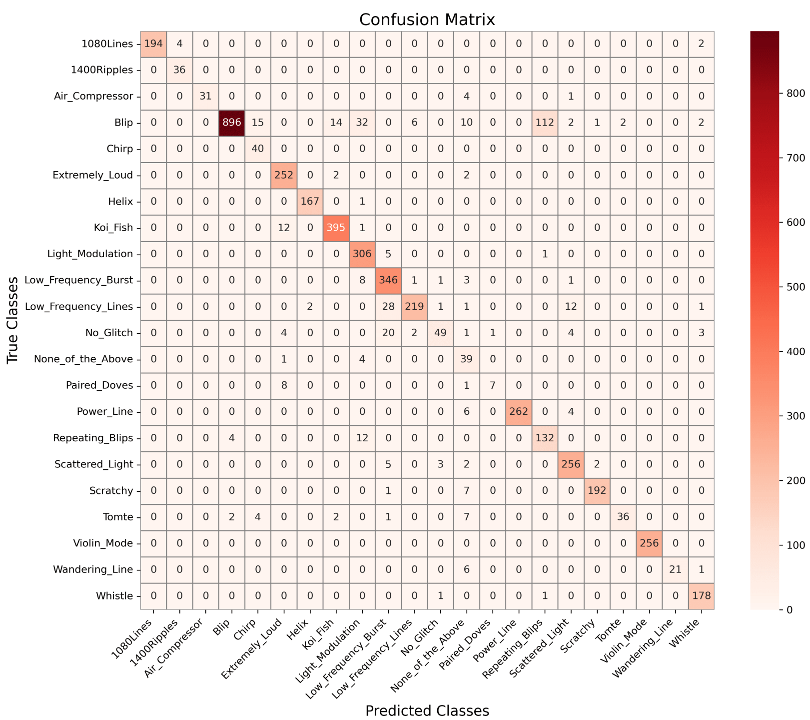


Figure 7: Confusion Matrix for Analysis II.

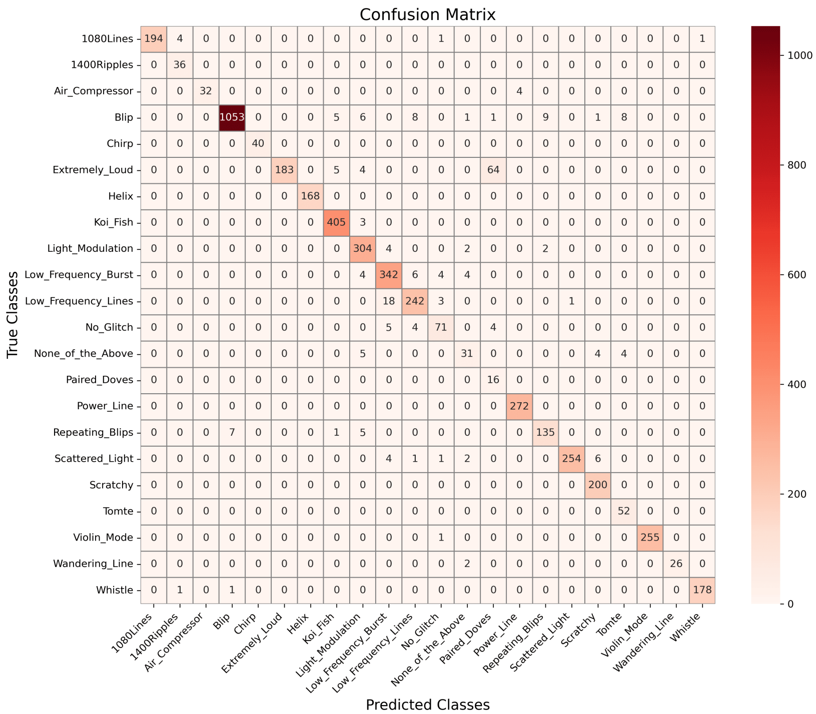


Figure 8: Confusion Matrix for Analysis III.

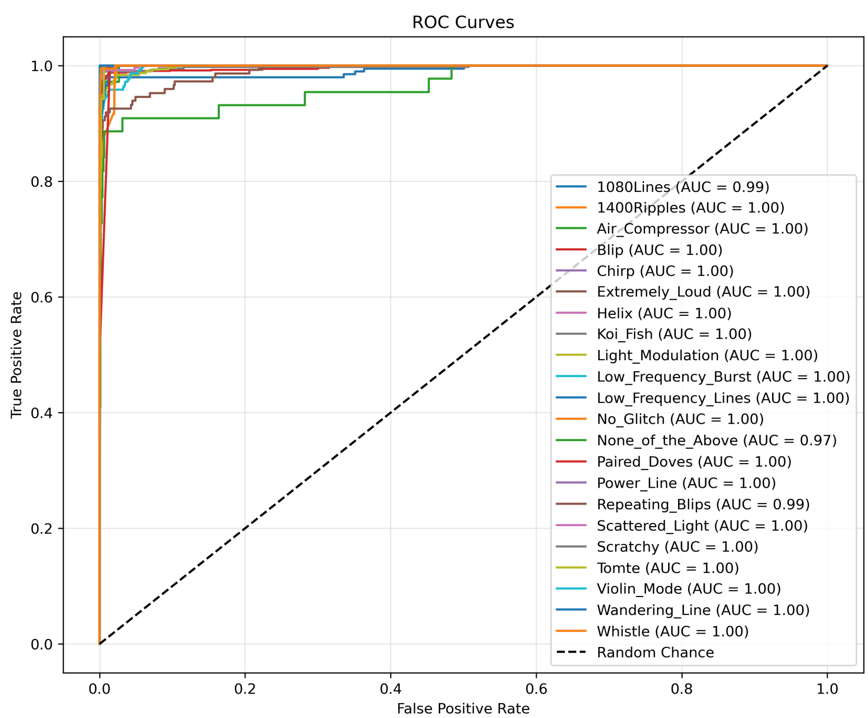


Figure 9: ROC Curve for Analysis I.

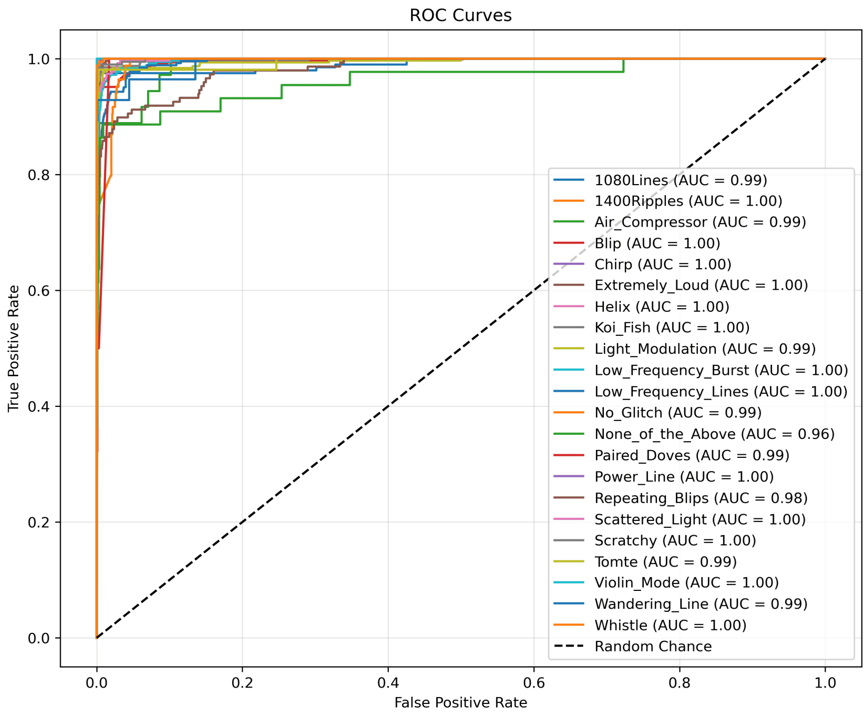


Figure 10: ROC Curve for Analysis II.

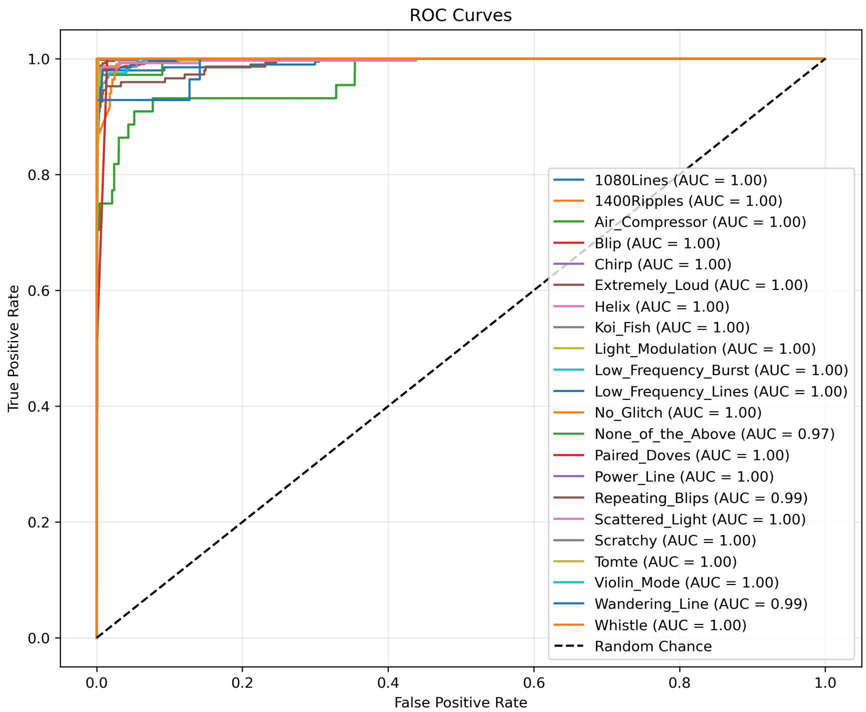


Figure 11: ROC Curve for Analysis III.

7 Future Work and Conclusion

In the future, we can focus on exploring alternative deep learning architectures, such as ResNet, and integrating fusion strategies to capture both spatial and temporal features more effectively. Additionally, advanced data augmentation techniques and synthetic data generation through techniques such as generative adversarial networks can be investigated to address persistent class imbalances. Refining feature extraction methods also remains a priority to enhance the model's ability to differentiate between similar glitch types. These refinements aim to address limitations observed in current experiments and ensure robust generalization to diverse glitch types and detector configurations.

In conclusion, this study demonstrates the potential of deep learning to improve gravitational wave data quality by effectively classifying glitches. The findings underscore the importance of transfer learning in achieving reliable performance. While transfer learning and advanced CNN architectures have proven effective, challenges such as class imbalances and the need for enhanced generalization remain focal points for future research. By addressing the challenges identified, future advancements in this domain will contribute significantly to enhancing the detection of gravitational waves.

Note

The glitch_classification.ipynb notebook contains the code for training the model. The glitch_classification_analysis.ipynb notebook has code for testing the model, as well as for other analysis tools.

References

[1] B. P. Abbott et al. “Observation of Gravitational Waves from a Binary Black Hole Merger”. In: *Phys. Rev. Lett.* 116 (6 2016), p. 061102.

[2] B. P. Abbott et al. “GW170817: Observation of Gravitational Waves from a Binary Neutron Star Inspiral”. In: *Phys. Rev. Lett.* 119 (16 2017), p. 161101.

[3] Hongyu Shen, Daniel George, Eliu Huerta, and Gravity Group/ LIGO Team. “Glitch Classification and Clustering for LIGO with Deep Transfer Learning”. In: *APS April Meeting Abstracts*. Vol. 2018. APS Meeting Abstracts. 2018, L01.027, p. L01.027.

[4] Tiago Fernandes et al. “Convolutional neural networks for the classification of glitches in gravitational-wave data streams”. In: *Classical and Quantum Gravity* 40.19 (2023), p. 195018.

[5] Yunan Wu et al. *Advancing Glitch Classification in Gravity Spy: Multi-view Fusion with Attention-based Machine Learning for Advanced LIGO's Fourth Observing Run*. 2024.

[6] Christian Szegedy et al. *Going Deeper with Convolutions*. 2014.

[7] M Zevin et al. “Gravity Spy: integrating advanced LIGO detector characterization, machine learning, and citizen science”. In: *Classical and Quantum Gravity* 34.6 (2017), p. 064003.

[8] Michael Zevin et al. “Gravity Spy: lessons learned and a path forward”. In: *The European Physical Journal Plus* 139.1 (2024).