

GWSkyNetLite: A Gravitational Wave Glitch Classifier Using High Speed Localization Skymaps

by

Eli Cohen

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

B.SC., UNIVERSITY OF BRITISH COLUMBIA

in

Department of Physics and Astronomy

(Honours Physics)

THE UNIVERSITY OF BRITISH COLUMBIA

(Vancouver)

October 2024

© Eli Cohen 2024

Abstract

Thanks to the efforts of the LIGO-Virgo-KAGRA (LVK) Collaboration, gravitational waves now form a crucial part of our understanding of astrophysical phenomena. Using an array of detectors, it is possible to characterize gravitational wave sources and probabilistically estimate their locations in space through representations called skymaps. These skymaps have been shown to contain valuable information regarding the authenticity of an event. However, there is still a lack of clarity on the specific characteristics or patterns within skymaps that are utilized by models to make predictions and distinguish between genuine events and detector noise. In this study, we introduce GWSkyNetLite, a machine learning model that integrates current architectures and image augmentation techniques to assess the credibility of event candidates localized using a new rapid method of localization. GwSkyNetLite achieves a true positive rate of 71.99% with a false positive rate of 19.97% and an accuracy of 87.11% on a simulated test dataset representative of current search pipeline parameters. These predictions are generated on the scale of milliseconds. Additionally, through use of saliency and occlusion techniques we characterize which features of skymaps are most relevant to classification. Our results suggest that alongside a large dependence on the size of the localization, geometry may play a role in the model's predictions.

Lay Summary

Gravitational waves are propagating ripples in space-time that are generated by astronomically massive objects. In order to learn more about the events that produce these waves researchers have to act quickly to obtain as much data as possible. However, the presence of glitches in the data, resulting from the incredible sensitivity of our detectors, complicates matters. Glitches can impede our ability to further investigate gravitational waves with observations of other emission products, a processes that has lead to a variety of exciting new insights about cosmology and astrophysics. This follow up must also be done quickly and accurately to ensure as much data is collected as possible. In order to improve the classification performance at low latency, we developed GWNetLite, a machine learning classification that has been tuned to work with high-speed localizations. These localizations are estimates of where an event has occurred in space and carry information about if a candidate is authentic. We also look at how this model decides if a candidate is a glitch or a gravitational wave in hopes of learning more about what is important in how our model makes decisions.

Table of Contents

Abstract	ii
Lay Summary	iii
Table of Contents	iv
List of Figures	vi
Acknowledgements	x
1 Introduction	1
1.1 Gravitational Waves	1
1.2 Ground Based Interferometric Detectors	2
1.3 Multi-Messenger Astronomy With Gravitational Waves	4
2 Theory	5
2.1 General Relativity	5
2.2 Event Detection	7
2.2.1 Glitches	8
2.3 Machine Learning	9
2.3.1 Convolutional Neural Networks	10
3 Methods	12
3.1 Gravitational Wave Localization Data	12
3.1.1 Datasets	12
3.1.2 Localization	14
3.1.3 Current Localization Algorithms	15

3.1.4	Skymaps and Projection	15
3.2	Construction of Models	17
3.2.1	Baseline Model	17
3.2.2	GWSkyNetLite	18
3.3	Architectural Improvements	18
3.3.1	Image Augmentation	18
3.3.2	Inception Blocks	19
3.3.3	Attention Blocks	20
3.3.4	Hyperparameter Tuning	21
3.3.5	Ensembling	22
3.4	Model Interpretability	22
3.4.1	Occlusion Maps	23
3.4.2	Saliency Maps	24
3.4.3	Feature Visualization	25
4	Results	27
4.1	Model Performance	27
4.2	Interpretations	29
4.2.1	Saliency	29
4.2.2	Occlusion	31
4.2.3	Visualization	32
4.2.4	Classifications	33
5	Discussion	36
	Bibliography	38

List of Figures

1.1	A diagram depicting an interferometer, which forms the foundation of the LIGO-Virgo detectors. The photodiode measures the interference resulting from the strain on the beam arms caused by a passing gravitational wave.	3
2.1	Two example cases of mountainous neutron stars. On the left there axis-asymmetric acceleration of mass, so no gravitational waves are produced. However on the right we can see that the mountain contributes a non-zero nonspherical contribution to the kinetic energy, resulting in the production of gravitational waves. We define R to be the distance from the source to our detector, in this example, LIGO, and M as the mass of our source.	6
2.2	A sample of different LIGO and Virgo glitches. These glitches represent different morphologies in the time-frequency domain that we expect to find in detector data.	9
2.3	An illustration of a 2D convolution. The purple matrix represents input image data. The red matrix represents a filter with a kernel size of 2x2 that can be trained using backpropagation. The gray matrix represents the output of convolving the purple matrix with the red matrix.	11
3.1	Two projections of the same event, G927737, generated using BAYESTAR and HSLM localization methods. Included in the plot are measurements of the area comprising 50% and 90% of the probability of the skymap.	16

3.2 (a) shows an unaugmented skymap of gravitational wave event. (b) is the same skymap translated horizontally, corresponding to a random physical shift in GPS time. Of important note is that both events share the same “event” label in training and will share the same metadata pertaining to SNR. 19

3.3 (a) is a schematic of an InceptionNeXt block. As input, the model receives 2D image data with N representing the batch size of the model and X and Y representing the dimensions of the image. The image is then processed by three depthwise convolutions of various kernel sizes with Z representing the number of filters. The output of the convolutional steps, along with a residual identity connection, are concatenated. The outputs are subsequently normalized across all features and combined through addition. (b) is a schematic of a residual block used in the baseline model and the original GWSkyNet. The same input as the InceptionNext block is used. This input is separably convolved by a 3×3 kernel with 16 filters. A non-linear activation function is then applied to the output of this convolution and the result is added back to the initial input. 20

3.4 The architecture of an attention block used in the model. Again, the input to the block is N X by Y pieces of 2D image data. This input is processed by a multi-head attention block with W heads. The result is then summed with the identity, and normalized with respect to the layer. In order to add additional non-linearity to the system we utilize a multilayer perceptron (MLP) with a residual connection before normalizing the data a final time. 21

3.5 Occlusion methods. Original skymap is illustrated on the left. Two different occlusion methods for the skymap are shown on the right. On the top, we occlude areas with the highest probability first and progress towards less probable areas as we iterate. On the bottom, we occlude regions with the lowest probability area first and work towards regions with higher probabilities. 24

3.6 An initial skymap and its corresponding saliency map. The saliency map was generated by considering the best single model after hyperparameter tuning with respect to the prediction score. 25

3.7 An example of feature visualization. On the left is an unaltered glitch skymap and on the right is the same skymap altered to decrease the prediction score of the skymap.	26
4.1 Confusion matrices describing the performance of the final baseline, best and ensemble models on the MDC 11 dataset. These are created with an imposed threshold corresponding to a false positive rate of 20%.	28
4.2 Model performance metrics. (a) An ROC curve highlighting the performance of the three models. The region highlight around the curves represents a 95% confidence interval. (b) The prediction speed of the best single model on the first 10000 MDC 11 candidates.	29
4.3 Correlation of MDC 11 saliency maps. Blue highlights correlation between saliency maps and their progenitor skymaps. Orange highlights correlation between saliency maps and random noise.	30
4.4 500 occlusion runs of MDC 11 data using the best single model. Each occlusion begins at the highest valued pixel. Each run is represented by a different colored line. The start of the run and the score of the unoccluded map is furthest to the right, and as more of the map is occluded the 90% area decreases and the new prediction score is plotted.	31
4.5 A figure showcasing the visualization of a “Scratchy” glitch, shown before in (b) and after in (c), adopting event-like characteristics through feature visualization (top) and a simulated gravitational wave event, shown before in (f) and after in (g), adopting glitch like characteristics. Plots (a) and (e) showcase the change in loss function as the skymaps are iteratively updated using backpropagation. These graphics reflect the intention to maximize our prediction score in (a) and minimize it in (e). The reduction in localization area seen in the transition from (b) to (c) seems to agree with this characteristic and plays a critical role in classification. On the other hand, figures (f) and (g) seem to imply that geometrical features in a sky map may play less of a role than overall localization area in the prediction of glitches.	33

4.6 The 10 events and glitches that the model correctly classified with the highest confidence. All candidates were correctly classified. 34

4.7 The 10 events and glitches most confidently incorrectly classified by the model. For all these candidates the model is very confident that they are described by the incorrect label. With prediction scores close to those observed in Fig 4.6. . . 35

Acknowledgements

I am deeply grateful to Professor Jess McIver for her support and affording me the opportunity to contribute to this collaboration. Being part of this team has been the highlight of my undergraduate journey at the University of British Columbia. Furthermore, I would like to thank Dr. Mervyn Chan for his invaluable mentorship that has shaped my project. Finally, I would like to thank Christina Gentle for her assistance and patience in the editing of this document.

Chapter 1

Introduction

1.1 Gravitational Waves

Gravitational waves are ripples in spacetime created by the asymmetrical acceleration of mass. When these ripples interact with matter they squeeze and stretch it in orthogonal directions. Moreover, due to the weak nature of gravitational interactions, gravitational waves can propagate with little disturbance, allowing their use as a tool to investigate the sources of these systems. Through their detection, our understanding of astrophysics and cosmology has been greatly benefited.

The study and detection of gravitational waves, starting with the first binary black hole merger directly observed on September 14, 2015 (GW150914) [1], has established itself as an important and exciting field in astrophysics. From the first event alone, the existence of black holes with mass on the order of $30 M_{\odot}$ was confirmed, along with the existence of binary black hole mergers. Moreover, this event allowed researchers to constrain the population of future events.

The rapid progress demonstrated in this field is exemplified by the LIGO Scientific Collaboration (LSC) and Virgo Collaboration's catalogue of gravitational wave events. After the latest observing run the third Gravitational-Wave Transient Catalog (GWTC) included 90 events, and one event with successful electromagnetic followup in 2017 [2]. Moreover, 81 significant alerts have been published in O4a with even more on the horizon with the start of O4b ¹.

This catalogue has provided insights into fundamental physics, astrophysics, and cosmology. An example of one of these findings are the constraints placed on the formation of compact binary systems. By observing the angular momenta of binary systems, researchers can draw conclusions on how these systems form and evolve [3, 4].

¹<https://gracedb.ligo.org/superevents/public/O4/>

The prospects for gravitational wave astronomy are bright, given the enhancements in sensitivity and expansion in size of the worldwide detector network. This includes the addition of KAGRA to the observational array and the planned construction of new detectors such as LIGO India. Moreover, the current successes of the array have inspired a planned next generation of ground-based gravitational wave detectors with increased sensitivity to astrophysical events, mainly the Cosmic Explorer [5] and Einstein Telescope [6]. This sensitivity would allow detectors to view events barely visible to our current detectors with incredible resolution, or the potential discovery of currently unseen gravitational wave sources like core-collapse supernovae and isolated neutron stars.

1.2 Ground Based Interferometric Detectors

These detectors are beam interferometers that function by detecting the interference of two beams of a split laser moving through 4 km beam arms. This is essential because the greater the length of an interferometer's arms, the more precise the measurements it can produce. Furthermore, this length is compounded by Fabry P  rot cavities that exploit the reflection of light to significantly increase the effective distance the light travels, which is essential due to the incredibly weak nature of gravitational waves.

As a gravitational wave interacts with the detector, depending on its polarization, the two arms experience strain resulting in stretching and compression. Differences in phase that arise from the two beams travelling a different distance can be observed by reuniting the beams and observing the resulting interference pattern with a photodetector. This allows for the calculation of a quantifiable strain, described as the change in length of one of the arms over the arm's total length, which provides evidence for a detection.

These detectors employ incredible sensitivity to detect these waves; the change in length of these arms resulting from a gravitational wave is on the order of 10,000 times less than of a proton and a fractional change of length on the order of 10^{-21} [7].

The remarkable sensitivity attained by interferometric gravitational wave detectors is constrained by various factors, including seismic disturbances, thermal fluctuations, and quantum disruptions. Seismic vibrations necessitate that the apparatus is shielded from harsh ground

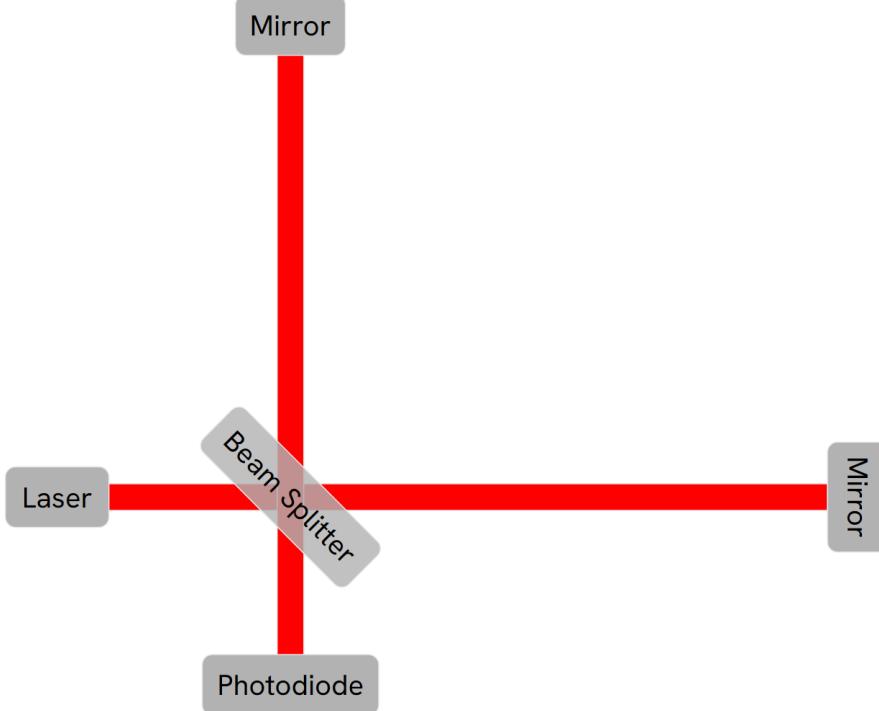


Figure 1.1: A diagram depicting an interferometer, which forms the foundation of the LIGO-Virgo detectors. The photodiode measures the interference resulting from the strain on the beam arms caused by a passing gravitational wave.

noise. The detector is so sensitive that vibrations from nearby traffic could impact the measurement. Thermal noise arises from minute oscillations of atoms within the mirrors and their mountings. Quantum effects arising from the quantized nature of light also impacts the sensitivity of the apparatus. Additionally, of humorous note was the introduction of noise from ravens pecking at ice on a pipe connected to one of the cooling systems [8]. These intrusions can serve as a severe hindrance to the analysis and classification of gravitational waves; we dub these short bursts of noise “glitches”.

The reliable detection of gravitational waves requires a network of multiple detectors. This is important for localization, as directional observations cannot be made with a single detector. Furthermore, by comparing the data of multiple detectors, candidates caused by glitches can be ruled out.

1.3 Multi-Messenger Astronomy With Gravitational Waves

One important aspect of the collaboration’s focus on reducing latency comes from the capacity to engage in observational collaboration with other types of detectors. Multi-messenger astronomy is a field in astrophysics that involves the coordinated observations of celestial events through various cosmic messengers, including electromagnetic radiation waves, neutrinos, cosmic rays, and now gravitational waves [9].

These different messengers provide complementary and comprehensive views of astrophysical phenomena and their environments, with sources such as solar flares, supernovae, compact binary mergers involving at least one neutron star, blazars, and tidal disruption events, furthering our understanding of the cosmos. A joint observation of these messengers may help improve our understanding and allow us to draw conclusions on a variety of topics, including the formation of heavy elements [10], constraints on the maximum mass of neutron stars [11], and indirect searches for dark matter [12].

Mergers of compact binaries that involve at least one neutron star can generate both gravitational waves and electromagnetic signals [13]. This was seen with the first direct observation of a binary neutron star inspiral (GW170817) by the LIGO-Virgo collaboration [14], which spurred on a variety of electromagnetic follow-ups. This included an independent observation 1.7 seconds after the merger time by the Fermi Gamma-ray Burst Monitor of a gamma-ray burst (GRB 170817A) [15], an independently verified optical transient detected by the Swope Telescope (SSS17a) [16] and X-ray followup by the Chandra telescope [17].

The joint observations of these emissions together provided a variety of insights, including an independent measurement of the Hubble constant [18], confirmation of gamma-ray bursts associated with binary neutron star mergers [19], and the synthesis of heavy elements [10]. Successful rapid electromagnetic follow-up to gravitational wave events, such as this one, are crucial, as certain signals may fade rapidly and longer-lasting signals may still convey unique information unsalvagable in the later parts of observation. This information is particularly useful because it serves to constrain the emission models of these phenomena [20].

Chapter 2

Theory

2.1 General Relativity

In 1916, Albert Einstein predicted the existence of gravitational waves using his general theory of relativity. These waves are disturbances in spacetime caused by an asymmetrical acceleration of mass. For observers at a large distance, Einstein's field equations, which describe the curvature of space-time, can be linearized, which allows the study of small perturbations in the metric tensor $g_{\mu\nu}$ [21], where the perturbation $|h_{\mu\nu}| \ll 1$, $\eta_{\mu\nu}$ is Minkowski metric and $O(h_{\mu\nu}^2)$ is a very small correction term.

$$g_{\mu\nu} \approx \eta_{\mu\nu} + h_{\mu\nu} + O(h_{\mu\nu}^2) \quad (2.1)$$

As an approximation we consider the space-time of a vacuum; we find that the Einstein Field equations give rise to a wave equation,

$$\left(\frac{\partial^2}{\partial t^2} - \nabla^2 \right) h_{\mu\nu} = 0 \quad (2.2)$$

Of particular note is the introduction of a source term to Eq. 2.2, resulting in the quadrupole formula, which describes the rate at which gravitational waves are emitted from a system, which is proportional to the second time derivative of the quadrupole moment, a quantity that scales with the mass of the system. It can be shown that

$$\bar{h}_{\mu\nu}(t, r) = \frac{2G}{c^4 r} (\ddot{I})_{\mu\nu}(t - r/c) \quad (2.3)$$

where $\bar{h}_{\mu\nu}$ represents the gravitational wave and $I_{\mu\nu}$ is the quadrupole moment.

This formula, along with other approximations, provides a method to roughly approximate

emission strengths for highly non-spherical sources, such as the merger of two compact objects,

$$h \approx \frac{2Mv_{ij}^2}{r} \quad (2.4)$$

Here, M is the mass of the source, v_{ij} is a non-spherical velocity and r is the distance from the source to our detector. Additionally, Mv_{ij}^2 is the magnitude of the non-spherical component of $(\ddot{\ell})_{ij}$, twice the kinetic energy of the non-spherical component of the source. An illustration of a neutron star mountain creating gravitational waves is shown in Fig 2.1.

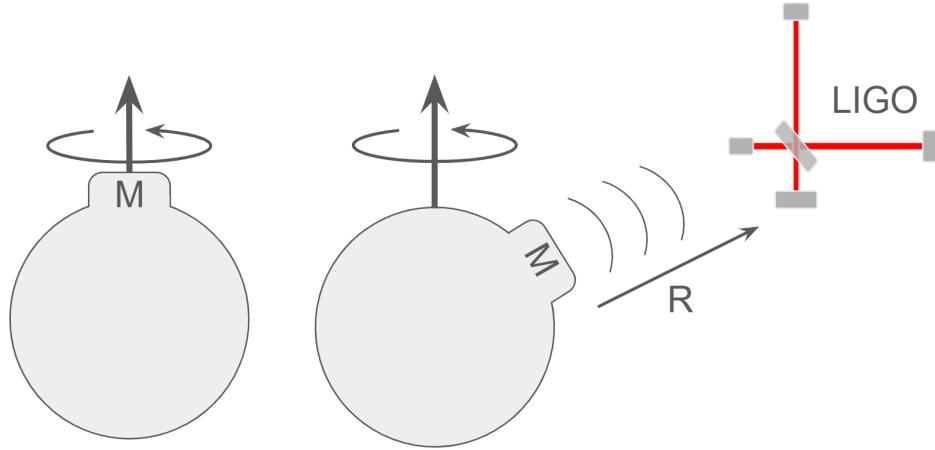


Figure 2.1: Two example cases of mountainous neutron stars. On the left there axis-asymmetric acceleration of mass, so no gravitational waves are produced. However on the right we can see that the mountain contributes a non-zero nonspherical contribution to the kinetic energy, resulting in the production of gravitational waves. We define R to be the distance from the source to our detector, in this example, LIGO, and M as the mass of our source.

This h can be thought of as the fraction of squeezing and stretching induced by the gravitational wave. The LIGO detectors are sensitive to waves larger than $h \sim 5 * 10^{-22}$. Moreover, using this Equation (2.3) we find that the only sources of detectable gravitational waves must be astrophysical, as a liberal estimate for the amplitude of a terrestrial gravitational wave lies around $h \sim 10^{-43}$, more than 40 orders of magnitude smaller than a binary black hole or neutron star merger that has an amplitude on the order of 10^{21} [22].

2.2 Event Detection

In order to detect gravitational waves from compact binary coalescences (CBCs), researchers employ a technique called matched filtering. This is achieved by comparing observed data to a bank of theoretical templates representing expected waveforms from astrophysical events. These expected waveforms also describe systems of known parameters such as the mass, localization, distance and spins of the compact objects, so this also provides a method to estimate the parameters of the merging objects. The data is further analyzed by a variety of pipelines to estimate additional collision parameters to select interesting candidates [23–27]. If an event is judged to have a lower enough false alarm rate (FAR) the results are published on a publicly available online database [28] alongside the estimated event parameters and additional metadata.

Effective electromagnetic follow-up observations of gravitational wave candidates depend on rapid release, and thus classification of said candidates. In the last LIGO-Virgo observing run, 80 gravitational wave candidates were publicly available in low latency, which entailed publication within minutes of detection time. However, 24 of them were later retracted and 11 of them were not confirmed by more refined offline analysis [29]. These false candidates could hinder electromagnetic follow-up efforts because of limited telescope time availability, especially if these potential targets initially appear attractive to astronomers because of precise localization or low likelihood of false alarms.

BAYESTAR is an important method used to follow up on candidates [30]. This is an algorithm that generates 3D sky localization information. These localizations are also included in the publication of the event. This gives astronomers the ability to know where to orient their detectors. Moreover, these skymaps encode information between detectors, as they consider a variety of parameters to estimate the source location of a candidate. This information can be leveraged in order to determine if an event is physical or a noise artifact.

To reduce the high false positive rate (FPR) in low-latency searches, Cabero et al. developed a machine learning algorithm, called GWSkyNet [31], which is capable of determining whether a gravitational wave candidate is astrophysical in real time using BAYESTAR localizations. An extension to GWSkyNet, called GWSkyNet-Multi, has also been developed by Abbott et al.

for the identification of gravitational wave candidates involving neutron stars [32].

As of yet, sky localization information has not been used as data in any current search pipelines. This is in part due to the low-latency nature of the pipelines compared to the relatively long time it takes to localize an event. However, due to a recent advance in localization speed, as part of this work, we seek to understand whether this input could potentially improve search pipeline classification accuracy.

2.2.1 Glitches

Due to the incredible sensitivity of the current generation of gravitational wave detectors [33], non-Gaussian artifacts, known as glitches, are frequent in their data output. Due to the relatively high occurrence of glitches a false alarm may be triggered when a glitch occurs in one detector alongside a large Gaussian noise fluctuation in another. These events could potentially impersonate astrophysical phenomena. Moreover, glitches overlapping gravitational wave signals have in the past significantly hindered the electromagnetic follow-up of candidates. In the case of the first binary neutron star merger (GW170817), the first hours of optical and ultraviolet follow-up were delayed due to an inability to localize the source because of the presence of a large glitch overlaying the data [34].

A variety of different techniques are employed to filter glitches in data at various stages of the detection timeline. For compact binary inspiral waves, processes known as DQ vetos and iDQ, serve to remove non-Gaussian noise transients. These processes ensure the initial quality of the data.

A selection of these glitches is shown in Fig 2.2. Their presence necessitates the development of sophisticated digital classification mechanisms to discern a large number of genuine astrophysical signals from instrumental and environmental disturbances.

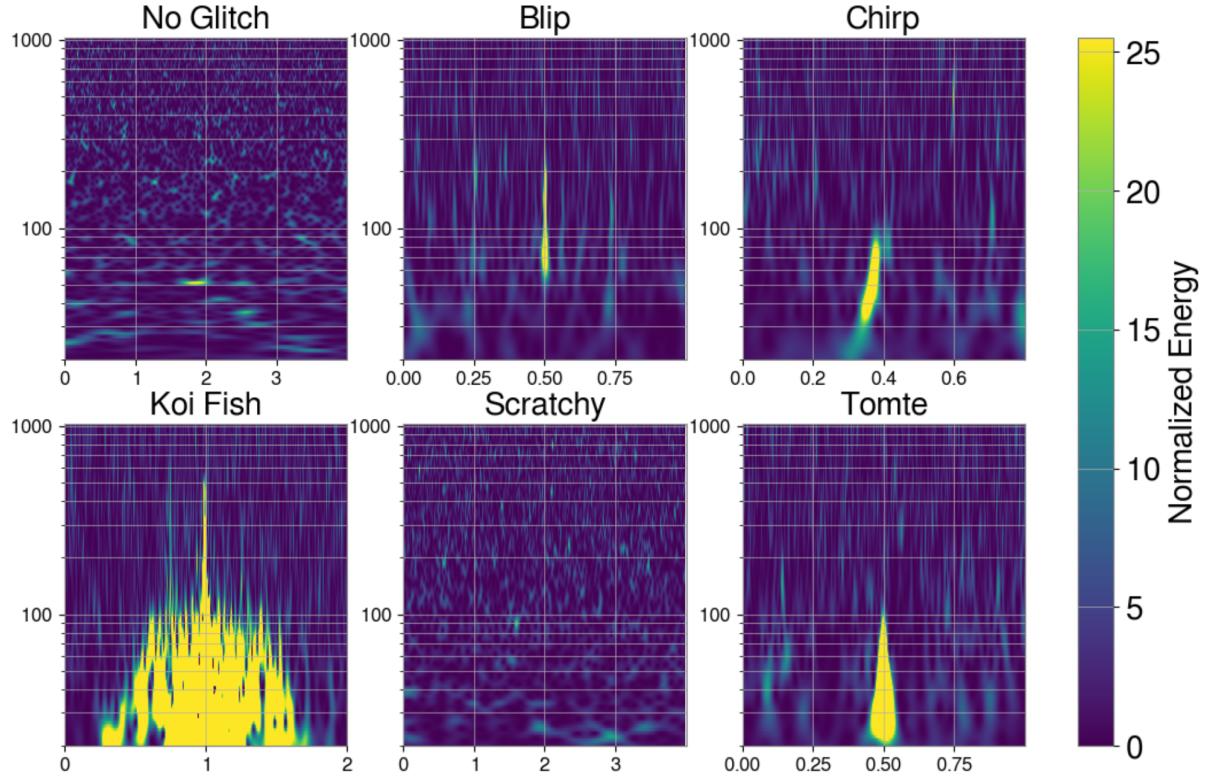


Figure 2.2: A sample of different LIGO and Virgo glitches. These glitches represent different morphologies in the time-frequency domain that we expect to find in detector data.

2.3 Machine Learning

Machine learning is a branch of statistics where systems learn from data patterns to make predictions or decisions without being explicitly programmed. These models are algorithms that iteratively learn from data, improving their performance over time. Machine learning models have been of immense interest in recent astrophysical research due to their ability to parse complex data with relative ease. The particular subset of models with which we are concerned are supervised methods that adjust the parameters of a highly complex function by comparing a model's response to input data to some known ground truth.

For classifying complicated input skymap data we seek to avoid difficult feature extraction while ensuring low latency. These requirements make a machine learning model an excellent candidate for this task.

2.3.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are a specific type of machine learning model that work with pixel-based data, such as images and videos. The main advantage of a CNN is its ability to learn hierarchical feature representations [35] and organize features in multiple layers to capture increasingly abstract concepts. In addition, CNNs accomplish this task with significantly less complexity than a fully connected model with comparable accuracy. For the purposes of our project, this is achieved through three types of layers in the network. First, convolutional layers identify spatial patterns in the data with the help of learned filters. Second, pooling layers that downsample the data while keeping essential information. Finally, fully connected layers that classify the data.

The convolutional layer is the defining component of the architecture of a CNN and serves as the primary mechanism for extracting spatial features from input data. The convolutions employed by the model are specified by a kernel size, which refers to the dimensions of the sliding window or filter matrix. This layer convolves across the input data, as seen in Figure 2.3, generating feature maps that highlight specific patterns and structures within the data. Using configurable settings such as padding, and strides, the convolutional layer can effectively control the spatial dimensions of the resulting feature maps, allowing the network to capture intricate details while keeping model complexity reasonable. Furthermore, activation functions, which act on the output of a neuron, can be used in tandem with these layers to introduce non-linearity that can further describe complicated features present in the data.

We expect that by training the convolutional layers they will capture the features that are necessary for the classification of the skymap. However, the actual prediction, arising from weighting these learned features, will be done by the fully connected layers. These are layers where every input is connected to every output by a weight.

In order to quantify how well our model describes the ground truth we utilize binary cross-entropy,

$$L = - \sum_i P^*(i) \log(P(i)) \tag{2.5}$$

where L is the loss score, $P^*(i)$ is the ground truth, and $P(i)$ is the prediction of the model,

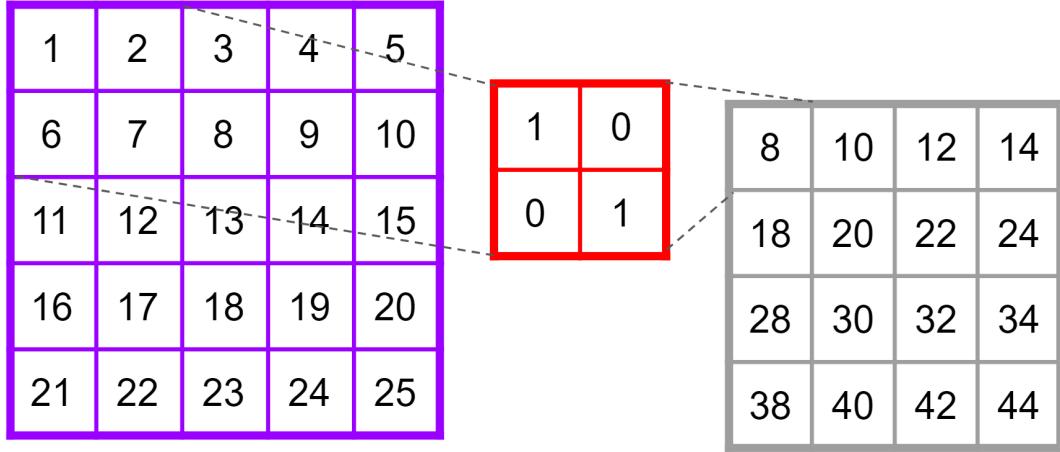


Figure 2.3: An illustration of a 2D convolution. The purple matrix represents input image data. The red matrix represents a filter with a kernel size of 2x2 that can be trained using backpropagation. The gray matrix represents the output of convolving the purple matrix with the red matrix.

which quantifies how close our predictions lie with respect to a given label. This loss is then used to iterate upon our predictions; after, an optimizer that repeatedly updates the weights and biases of our neurons that seeks to minimize our model’s loss is employed. Following supervised learning, we seek to accurately assess our model’s performance. In order to accomplish this, we avoid data leakage by preventing cross-contamination of testing and training data. As such, absolute separation of the data sets is required [36].

Moreover, We also seek to avoid overtraining, a phenomena wherein a model learns to fit the training data too closely, capturing noise rather than underlying patterns [37]. This often leads to poor generalization on unseen data. In order to combat this we seek to reduce model complexity, as this makes it more difficult for a model to completely describe our training dataset.

Chapter 3

Methods

In this chapter, we describe how we develop and interpret a machine learning algorithm for the purpose of classifying low-latency gravitational wave localization data. To do this, we first select relevant simulated training and testing datasets. These datasets are comprised of strain data that we then localize using a new high speed method and project into a format useable by a machine learning model. We then develop a baseline model for comparison, and our model GWSkyNetLite that builds upon the scaffold of the baseline with novel data and architectural modifications. We then employ hyperparameter tuning to optimize model performance. There will also be a discussion of the approaches employed to understand and reveal the features and pattern hidden in the data that the model has captured.

3.1 Gravitational Wave Localization Data

The data collected for this thesis was acquired by the LIGO-Virgo-KAGRA Collaboration [38]. To develop a machine learning model for the purpose of separating glitches and gravitational waves, we employed three different data sets. In particular, we are concerned with data from the LIGO [39] and Virgo detectors [40].

3.1.1 Datasets

Due to the inability to characterize an entire observing run, especially in regards to glitch transients, three datasets, consisting of simulated events and real glitches, are used in the development of GWSkyNetLite. The first for general training, the second for calibration, and the third as an independent and representative testing set.

The first dataset, was created by the UBC LIGO group for the development of GWSkyNet II. This was used primarily to allow our model to learn important skymap features from a

3.1. Gravitational Wave Localization Data

large set of data. This dataset was described by Chan et al. in prep, using the fourth Open Gravitational wave Catalog (4-OGC)² and O3 glitch classifications from Gravity Spy [41]. To generate this dataset, simulations were used to create a diverse population of real events using gravitational waveform models provided by LALSuite [42]. These events represent a variety of different mergers with different parameters like spin, mass, source location, and merger type. In order to ensure these waveforms were representative of real gravitational wave events we add them to a Gaussian noise background coloured using PSDs of noise observed in the LIGO and Virgo detectors. Glitches were identified through the analysis of real detector data and corresponding triggers from GW search pipelines.

The second and third datasets were generated by the LVK Collaboration for a Mock Data Challenge (MDC) [43] campaign. This MDC uses data collected by the LIGO and VIRGO detectors during a period of 40 days, from 5 January 2020 15:59:42 to 14 February 2020 15:59:42 UTC. The collected data was then enriched through the addition of 50,000 simulated CBC gravitational wave signals comprised of binary neutron star, neutron star black hole, and binary black hole mergers. For further details pertaining to the MDC and the source parameters for the simulated signals, see [43]. The data are then replayed and analyzed using search pipelines with parameters similar to those used in low latency during an observing run. Once the 40 days of data have been replayed and analyzed, this process is repeated with refined pipeline configurations informed by performance on the MDC and in observing runs. Each of these iterations is called a cycle. This is crucial, as even slight variations in pipeline parameters can result in the identification of different populations of gravitational waves and glitches. Differences in detection time incurred by different pipeline parameters may also produce slightly different localizations. Moreover, this methodology ensures that the datasets have events and glitches that are closely representative of those seen in an actual observing run in low latency. Furthermore, although the first dataset represents a reasonable population of gravitational wave data, we make use of the independently generated data to ensure the impartial evaluation of our model.

After initially training on the simulated dataset, the model is fine-tuned to an MCD-style dataset through an additional round of training on the dataset produced during the 10th cycle

²<https://github.com/gwastro/4-ogc>

3.1. Gravitational Wave Localization Data

of MDC (MDC 10). This fine-tuning was performed to allow for a reasonable generalization of our models [44]. We implement fine-tuning by freezing the convolutional layers of our model while allowing the dense layers to train. This reflects the idea that skymaps are comprised of features general to our localization method that are captured by our convolutional layers, whereas the interpretation and subsequent classification is done by the fully connected layers.

Lastly, we use the more recent 11th cycle of MDC (MDC 11) as our test dataset for the final evaluation of the models. This dataset was used for this purpose, as out of all the MDC cycle datasets, it has been dubbed the most reflective of current search pipeline parameters used in observation. This is because MDC 11 is the cycle that began closest to the start of a recent observing run. MDC 12, a later cycle of MDC overlapped with the start of the O4 run, and therefore certain pipelines were not available to participate in the cycle. Since both MDC10 and MDC11 are supplied with the same set of injections, events, and glitches in MDC 11 that have comparable GPS times taking into account the difference in time between cycles (a difference of 0.5s), candidates in MDC 10 are removed in order to prevent data leakage.

3.1.2 Localization

Gravitational waves are mainly localized using timing triangulation via a comparison between detectors [45]. By factoring the arrival time, the distance between detectors, and the number of detectors, an estimate of the source's location can be found. In principle, for a network of three detectors, we can also estimate the localization error using the formula [46]

$$\Delta\Omega = \frac{2c^2\Delta\tau_{12}\Delta\tau_{13}}{A\cos(\theta)} \quad (3.1)$$

where $\Delta\tau_{12}$ and $\Delta\tau_{13}$ are the uncertainties in timing delays between detectors 1 and 2, and 2 and 3 respectively. This uncertainty can be reduced by adding more detectors to a network. Localization can be further improved by comparing the amplitude, and phase of the relevant strain data.

3.1.3 Current Localization Algorithms

For the development of GWSkyNetLite, we generate skymaps: images showing gravitational wave candidate localizations in the sky. As of the current time, BAYESTAR represents the standard for low-latency localization. This software localizes events using a Bayesian method and serves as a significant improvement over previous Monte Carlo methods, such as LALINFERENCE [47]. BAYESTAR generates source sky location estimates on the order of seconds to minutes compared to the minutes to hours taken by previous algorithms. Additionally, BAYESTAR provides 3D distance estimates, localizing a candidate not only as a position on the sky but also in space. These localizations generated by BAYESTAR are also available on GraceDB for further analysis.

Due to the importance of swift classification of candidates, efforts in skymap localization have placed an emphasis on low latency. Takuya Tsutsui et al. have recently developed a high-speed localization method (HSLM), with even lower latency than BAYESTAR. By marginalizing over specific search parameters, including distance measurements, the HSLM has achieved localizations of the order of milliseconds [48]. However, the increase in the rapidity of this method comes at the cost of higher uncertainty in localization dimensions compared to current localization methods. Moreover, due to the marginalization over distance, the low latency approach loses information about 3D distance values.

We use HSLM to generate FITS (Flexible Image Transport System) localization files for all relevant datasets. FITS files are well suited for astronomical images and will be used in the proposed project, as they support the storage of multidimensional data, metadata, and complex structures.

3.1.4 Skymaps and Projection

After the gravitational wave candidate has been localized, it is formatted by the Hierarchical Equal Area isoLatitude Pixelation of a sphere (HEALPix) method [49]. HEALPix assigns a mesh to the sky with equal-sized regions. In each region, our localization algorithm assigns a value corresponding to the probability that the gravitational wave source is located directly above that location in the sky. In this form, the data are unsuitable for use in a CNN, so

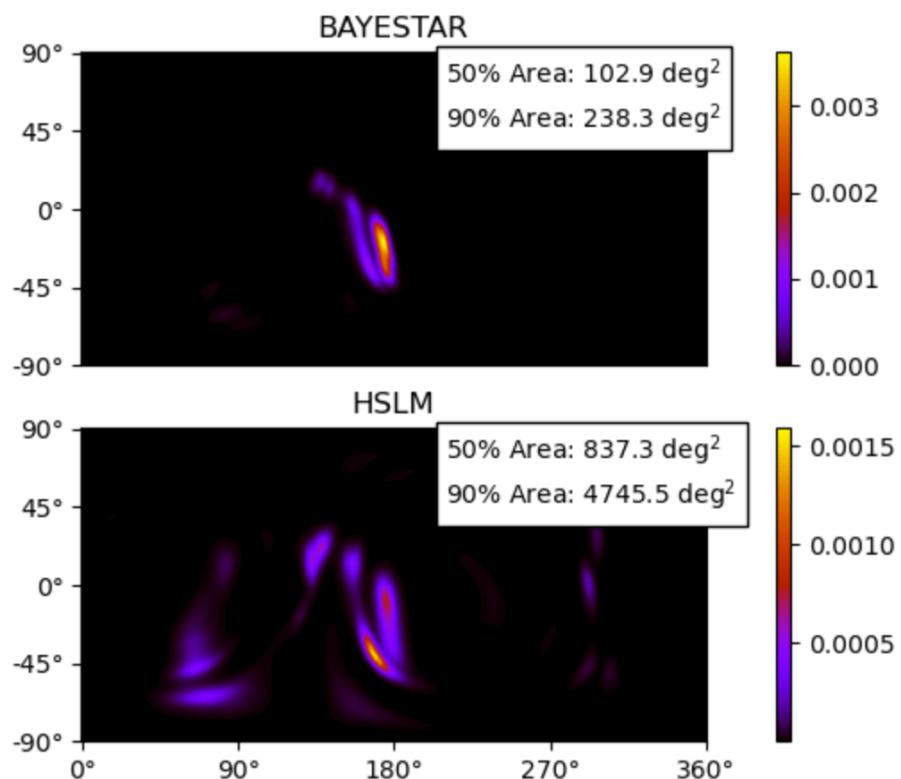


Figure 3.1: Two projections of the same event, G927737, generated using BAYESTAR and HSLM localization methods. Included in the plot are measurements of the area comprising 50% and 90% of the probability of the skymap.

they are projected in a rectangular format. Sky maps are then normalized by dividing by the maximum pixel value and this normalization value is recorded for use in the model. Finally, the normalization value is normalized by dividing it by the largest normalization value in the data set. Also known as feature scaling, normalization often improves convergence times and generalizability by preventing the dominance of large-valued inputs [50]. This preprocessing step has shown significant performance increases compared to unnormalized data.

The data used to generate these skymaps also provide signal-to-noise ratios (SNRs) for each detector parallel to the time series data. We find these benefit classifications and include them alongside a value characterizing the maximum value of the SNR prior to normalization. The normalization value is also included in the input data.

3.2 Construction of Models

The models discussed in this section were constructed using TensorFlow [51] using the Keras package [52].

3.2.1 Baseline Model

In order to evaluate our models and to provide a point of comparison, we construct a baseline. To do so we adapt GWSkyNet to the HLSM localization data along with its current weights and biases.

The current iteration of GWSkyNet uses distance estimations provided by the BAYESTAR localization algorithm; therefore, we will be concerned with a limited subset of branches in the model. The subset of interest that will be incorporated into the baseline comprises multiple branches of the model. First, a branch that takes 180x360 2D skymap projections. Second, a branch consisting of an input of the normalization factor of the said skymap. Finally, a branch that takes a hot-encoded 3x1 vector of SNRs as input, either one if the SNR exceeds a certain threshold or zero if the threshold is not met or the candidate is not detected by the detector.

The input of all of these branches can be provided by the SNR time series or produced by the HLSM. Therefore, in order to adapt GWSkyNet to our new dataset produced by the HLSM, we will select the above branches and introduce a final single sigmoid perceptron trained on

our dataset to classify events and glitches. This perceptron is necessary, as the one present in the current GWSkyNet requires all of the branches to make a prediction, therefore we must introduce a new one and train it on our dataset.

3.2.2 GWSkyNetLite

Our model mirrors the adapted baseline, consisting of a convolutional branch that accepts 180x360 skymap projections, a second branch that considers the normalization factor, a third, densely connected branch that considers SNRs without the addition of a threshold used by GWSkyNet, and a final branch that takes a normalization factor for the SNR vector.

We optimize our model for these new low-latency data through the implementation of various modifications including augmentation of our data, new architectures and a hyperparameter tuning process. Our focus lies on maintaining latency at the same magnitude as that of the localization method while maximizing performance. This is done by utilizing data augmentation to create a more robust dataset, employing a new architecture to better capture the intricacies of our data, and finally tuning the hyperparameters of the model to find an optimal configuration.

3.3 Architectural Improvements

3.3.1 Image Augmentation

Each piece of generated data is localized with respect to a GPS time (the number of seconds since the start of the GPS epoch at midnight on January 6th 1980), which determines the longitudinal position of the localization in the sky. An example of an initial piece of data is shown in Fig 3.2 (a), the horizontal position of this skymap is determined by the GPS time it was recorded at. Since the data contains known injections and glitches we augment the GPS times to generate additional samples with the same label as the parent for the model’s training. However, since our projection method assigns the same region area to pixels in the same row, we can also apply a horizontal translation of our projected skymaps without further modification to simulate a shift in GPS recording time. We employ the latter approach due to the ease of implementation. This translation can be seen in Fig 3.2 (b) where our skymap has been shifted; this new skymap corresponds to the same event recorded at a different GPS time. Since CNNs

are not translationally invariant, but can be trained to be [53], we find that by introducing image augmentation our performance is bolstered. In order to reduce the size and processing time of our training phase, we randomly translate the skymap data prior to training rather than storing additional data. However, in order to be certain no bias is introduced into the model testing data is left unaugmented.

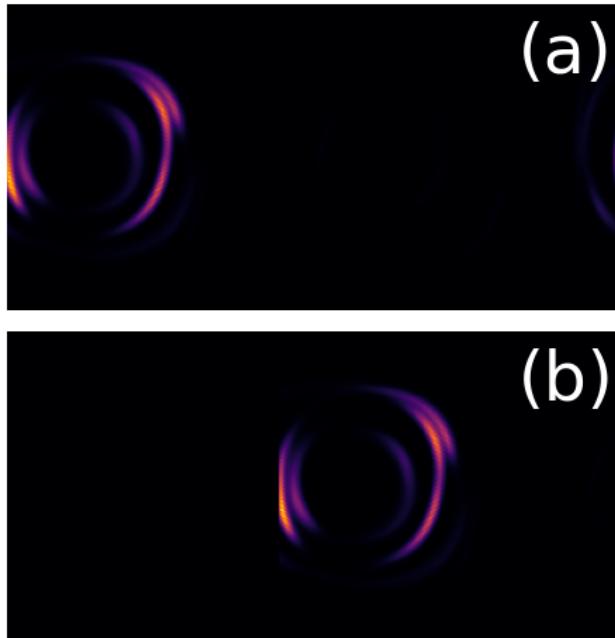


Figure 3.2: (a) shows an unaugmented skymap of gravitational wave event. (b) is the same skymap translated horizontally, corresponding to a random physical shift in GPS time. Of important note is that both events share the same “event” label in training and will share the same metadata pertaining to SNR.

3.3.2 Inception Blocks

Due to the large interest in image processing, many different approaches to CNN architecture have been developed [54–60]. The current iteration of GWSkyNet utilizes one of these architectures, residual blocks [54], which are convolutional blocks that add a convolved output to an unconvolved input. By retaining the information stored in the unconvolved image through the addition of the identity machine models are capable of achieving better performance through increased depth without the risk of overtraining.

Testing on the MDC 11 dataset suggests that we can improve the performance and through-

put of our model by utilizing an architecture incorporating InceptionNeXt blocks [61]. Taking inspiration from vision transformers, this architecture makes use of variably sized depthwise convolutions to decompose a larger convolutional kernel into separate parallel branches. Moreover, residual connections are also employed to leverage their advantages, including improved optimization and precision.

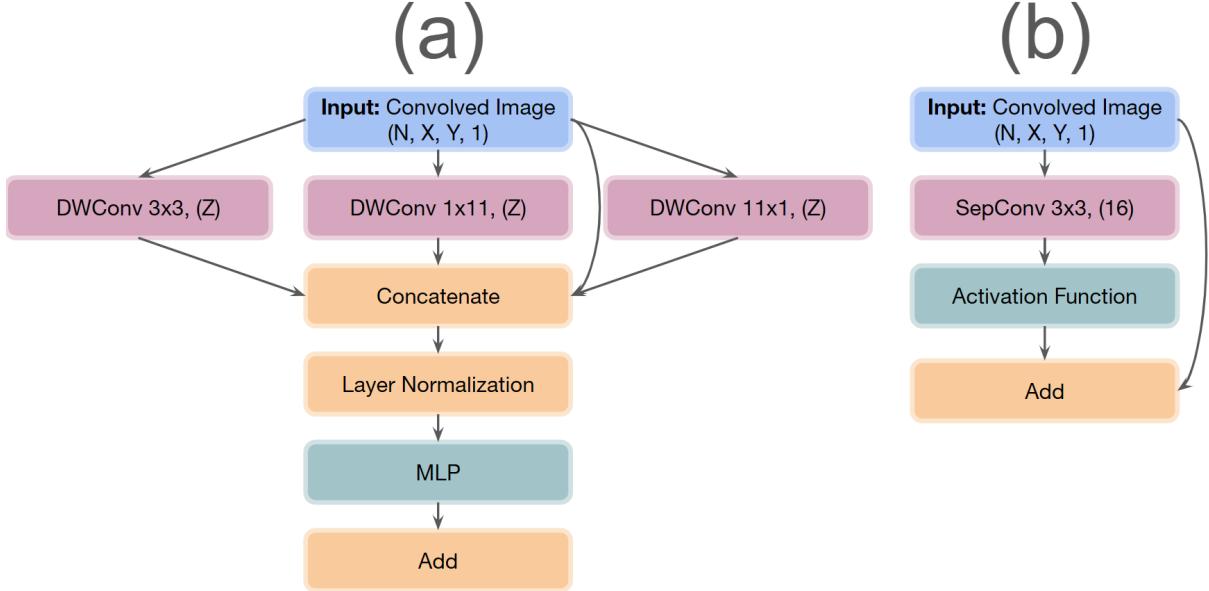


Figure 3.3: (a) is a schematic of an InceptionNeXt block. As input, the model receives 2D image data with N representing the batch size of the model and X and Y representing the dimensions of the image. The image is then processed by three depthwise convolutions of various kernel sizes with Z representing the number of filters. The output of the convolutional steps, along with a residual identity connection, are concatenated. The outputs are subsequently normalized across all features and combined through addition. (b) is a schematic of a residual block used in the baseline model and the original GWSkyNet. The same input as the InceptionNext block is used. This input is separably convolved by a 3×3 kernel with 16 filters. A non-linear activation function is then applied to the output of this convolution and the result is added back to the initial input.

3.3.3 Attention Blocks

Another architectural improvement was achieved through the use of self-attention blocks [55]. Self-attention describes the technique of assigning importance to an element of a set by comparing its similarity to all other elements. We employ multi-head attention, an extension of self-attention where the block computes multiple sets of attention weights in parallel. Each set, or head, attends to different parts of the input skymap. By allowing the model to consider var-

ious perspectives simultaneously, multi-head attention enables capturing complex relationships and dependencies within the data.

Inspired by the CoAtNet architecture [56] which achieves significant performance gains on the JFT-3B dataset [62], we incorporate these blocks after the attention blocks and before classification. Similar to the attention blocks we employ residual connections in these blocks to avoid early overtraining.

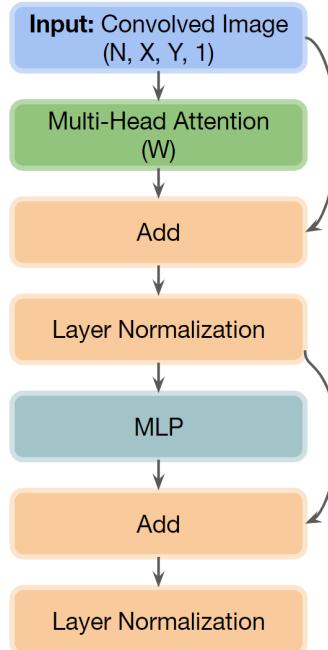


Figure 3.4: The architecture of an attention block used in the model. Again, the input to the block is $N \times Y$ pieces of 2D image data. This input is processed by a multi-head attention block with W heads. The result is then summed with the identity, and normalized with respect to the layer. In order to add additional non-linearity to the system we utilize a multilayer perceptron (MLP) with a residual connection before normalizing the data a final time.

3.3.4 Hyperparameter Tuning

After the addition of attention and inception blocks the skeleton of the model is complete. In order to optimize performance on the new dataset we optimize the hyperparameters of the model via a Bayesian sampling method, the tree-structured Parzen estimator (TPE) [63]. TPE works by modelling the objective function and searching for the optimal hyperparameters through a probabilistic model. This model is constructed through iteration where past trials provide information to update the hyperparameter space. We also employ a hyperband pruning

method [64] to discard candidates that show little potential in order to reduce the amount of computational time required to train a large number of models. It does this by treating the selection of models as a resource allocation problem, allocating resources to randomly sampled configurations. Both the sampler and pruner were implemented using the Optuna optimization framework [65].

3.3.5 Ensembling

In order to improve the generalizability of our final model [66], we ensemble the top three best-performing models selected by the hyperparameter tuning process. When the ensemble model is presented with data, each individual model returns a prediction score, which represents the average of all the scores. To increase diversity, each of the selected models has a unique architecture. We find this improves model performance significantly, most likely due to a resistance to outliers gained through having a more comprehensive set of learned features that arises from combining multiple models.

However, it should be noted that this process is computationally expensive. These predictions can be made simultaneously in a parallelized scheme, but more computational resources are demanded in comparison to a single model proportional to the number of models ensembled.

3.4 Model Interpretability

GWSkyNet and GWSkyNet-Multi [31, 32] have shown that it is possible to distinguish between true gravitational wave signals and noise using sky localization information. However, it is still largely unclear what patterns or features in skymaps are captured by these models in order to make predictions. Machine learning models are also known for their lack of transparency [67], making the outputs from such models less trustworthy. In order to increase the interpretability and trustworthiness of our results we employ various methods to discern what features the model values in order to make predictions and thereby characterize what physical features separate an event from a glitch.

3.4.1 Occlusion Maps

The fundamental concept behind elucidating machine learning predictions through occlusion involves deleting a specific feature from the input data and observing the resulting changes in the output prediction. In this case, we focus on the skymap component of the input data, while keeping the data linked to the SNR unchanged. By replacing a section of the input skymap with the map's minimum value, we effectively eliminate the spatial information at that particular location. However, in the event that the highest-valued pixel of the skymap is obscured, the data must be adjusted to the new maximum value, requiring an update to the input skymap normalization.

We pursue two different approaches to implementing occlusion maps for feature analysis. The first involves iterating over the input skymap data and setting a square of pixels to the skymap's minimum value. We then normalize the data to ensure that if we occlude the maximum value of the skymap, the resulting occluded skymap still has a maximum value of 1. The prediction is then made on the skymap with the removed information. We subsequently visualize the results by generating a heatmap where the value of each pixel corresponds to the model's prediction. This is used to highlight the geometric significance of the skymap's characteristics.

The other method involves occluding the brightest or least bright regions and gradually occluding more and more of the map. By comparing the prediction scores of the skymaps with the number of degrees squared (deg^2) that comprise 95% of the event's probability area, we can observe the change in prediction as a function of localization size.

This method allows us to examine how individual skymaps may vary in response to occlusion, as well as general trends highlighting the impact that uncertainty in localization has on the model's classifications. Analysis of individual maps may provide insight into certain preferred geometries that make a candidate event-like or glitch-like. General trends in prediction score may suggest a correlation between how localized a skymap is, and the prediction of the model. Important to note is that the skymaps generated by occlusion may not correspond to physically realizable skymaps. However, these altered maps may still provide insights into how the model classifies these candidates.

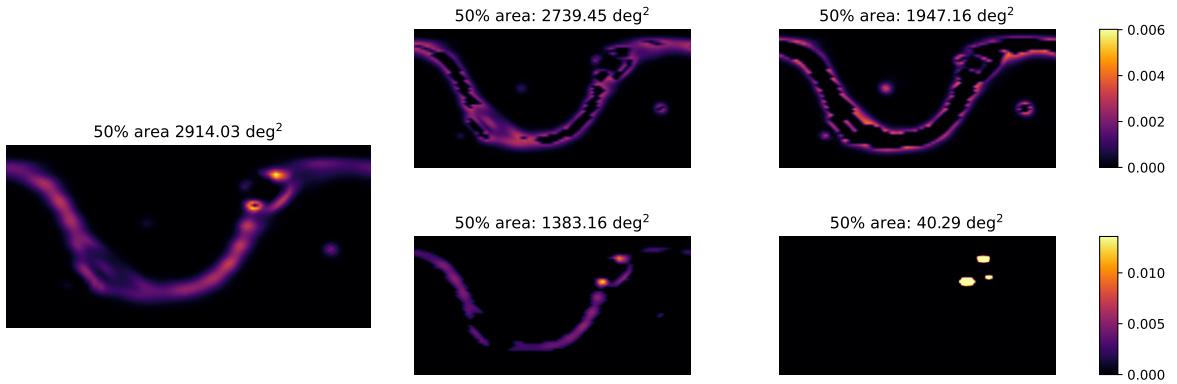


Figure 3.5: Occlusion methods. Original skymap is illustrated on the left. Two different occlusion methods for the skymap are shown on the right. On the top, we occlude areas with the highest probability first and progress towards less probable areas as we iterate. On the bottom, we occlude regions with the lowest probability area first and work towards regions with higher probabilities.

3.4.2 Saliency Maps

Another method utilized to interpret our model is saliency maps. These maps take advantage of the gradients generated by backpropagation to weight the input data [68], rather than the response of each trainable parameter. This emphasizes which aspects of the image data the model is sensitive to and can be used to understand the significance of a given feature.

$$\xi^{n+1} = \xi - \alpha \frac{\partial F(X, \xi^n)}{\partial \xi} \quad (3.2)$$

Equation 3.2 highlights the relation between our updated parameter ξ (in this case ξ is the value of a pixel in a skymap) and our function of interest, $F(X, \xi)$ where X is our input data scaled by our step size α . If we take ξ to be a subset of the input data and consider the derivative term we can effectively see the impact a small change in our input parameter has on our model’s response with respect to a particular function. This function is arbitrary, but of particular note is the prediction score given to the data by the model and the loss generated by that prediction. These gradients are calculated using TensorFlow’s GradientTape function which performs automatic differentiation to evaluate the partial derivative of a function.

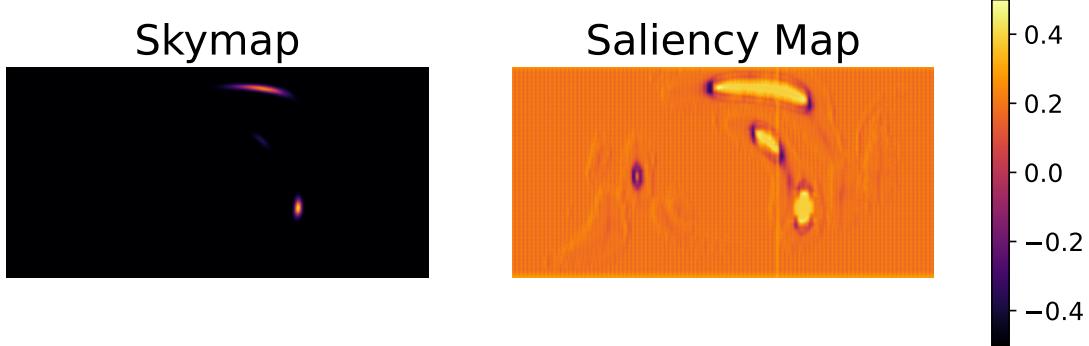


Figure 3.6: An initial skymap and its corresponding saliency map. The saliency map was generated by considering the best single model after hyperparameter tuning with respect to the prediction score.

3.4.3 Feature Visualization

Feature visualization attempts to understand what components of the image data CNNs are ‘looking at’ by maximizing the response of the network to a fabricated example [69]. This is done using a method similar to saliency maps, where the parameter ξ of Equation 3.2 is updated using the calculated gradient. By iteratively updating randomly generated input data, an example is generated that maximizes the response function.

This can be applied to a variety of components in the neural network, ranging from the overall prediction score to the output of individual neurons. However, due to the abstract nature of our input data, results generated using the latter option tend to be hard to interpret. As such, we focus primarily on the overall prediction score.

We use feature visualization in this manner on existing input skymaps in order to change the character of a skymap to be either more glitch-like or more event-like. This is done by using the model’s prediction score as the value to either be minimized in the case of a glitch, or maximized in the case of an event.

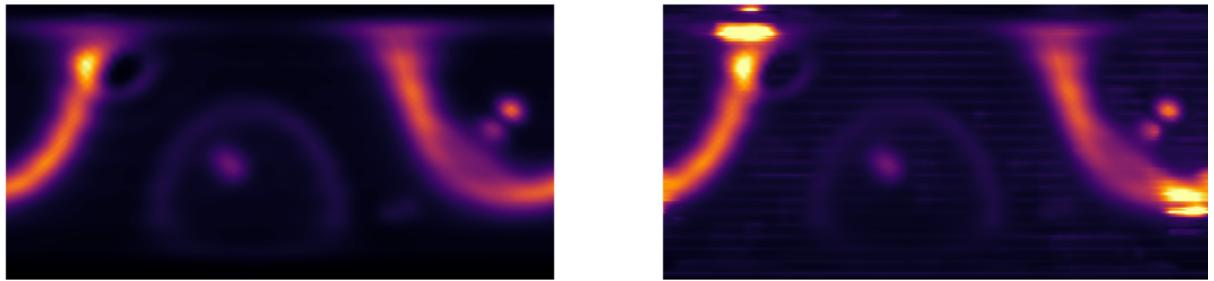


Figure 3.7: An example of feature visualization. On the left is an unaltered glitch skymap and on the right is the same skymap altered to decrease the prediction score of the skymap.

Chapter 4

Results

4.1 Model Performance

After 220 models were trained using Optuna [70] to implement Bayesian hyperparameter method detailed above, the top model was selected based on which configuration scored the highest accuracy on the MDC 11 test set. This is considered our best single model. We also selected the two second-best models and combined their predictions with those of the best single model to assess the performance of an ensemble.

To classify an event, a model assigns a prediction score to each candidate, ranging from 0 to 1. A score of 0 indicates the model is completely confident that a candidate is a glitch, while a score of 1 signifies complete confidence that the candidate is an event. We are then capable of designating a threshold where prediction scores greater than the value are classified as events, and prediction scores less than the value are classified as glitches.

Sensitivity, or true positive rate (TPR), is defined as the fraction of correctly identified events compared to the total number of events. The false positive rate (FPR) is the proportion of glitches that are incorrectly classified as events. We balance the selection of a desirable TPR for a given FPR by varying the threshold of classification. An example of this is highlighted in Fig. 4.1, with a threshold corresponding to a 20% FPR on the test set. At this FPR, the baseline model achieves a TPR of 57%, the best single model achieves 69%, and the ensemble achieves the best performance at 72%.

A range of thresholds corresponding to every possible TPR and FPR value are shown in the receiver operating characteristic (ROC) curve in Fig. 4.2 (a). This curve visually represents the trade-off between the TPR and FPR for a binary classification model. An additional metric, the area under the ROC curve (AUC), provides a summary of the classifier's performance at all possible classification thresholds. AUC can be interpreted as a way of describing the likelihood of

4.1. Model Performance

the model assigning a higher prediction score to a randomly chosen positive example compared to a randomly chosen negative example. Both the single best model and the ensemble are significantly better than baseline model’s predictions at all relevant thresholds. We calculate an AUC of 0.757 for the baseline, 0.808 for the best single model and 0.834 for the ensemble.

		Baseline Model		Best Model		Ensemble	
		Glitches	Events	Glitches	Events	Glitches	Events
Ground truth	Glitches	2500 80.03%	624 19.97%	2500 80.03%	624 19.97%	2500 80.03%	624 19.97%
	Events	8410 42.86%	11214 57.14%	6083 31.00%	13541 69.00%	5496 28.01%	14128 71.99%
		Glitches	Events	Glitches	Events	Glitches	Events
		Prediction	Prediction	Prediction	Prediction	Prediction	Prediction

Figure 4.1: Confusion matrices describing the performance of the final baseline, best and ensemble models on the MDC 11 dataset. These are created with an imposed threshold corresponding to a false positive rate of 20%.

The prediction speed of a machine learning model refers to the time it takes for the model to generate a classification given input data. For the best single model, the prediction speed for the MDC 11 candidates can be seen in Fig. 4.2 (b). With a mean prediction speed of 0.125s, the model achieves latency on the same order of magnitude as the localization software. It should be of note that due to architectural complexity GWSkyNetLite generates predictions slower than the baseline model, however, this is more than compensated for by the reduced localization time in the creation of the data.

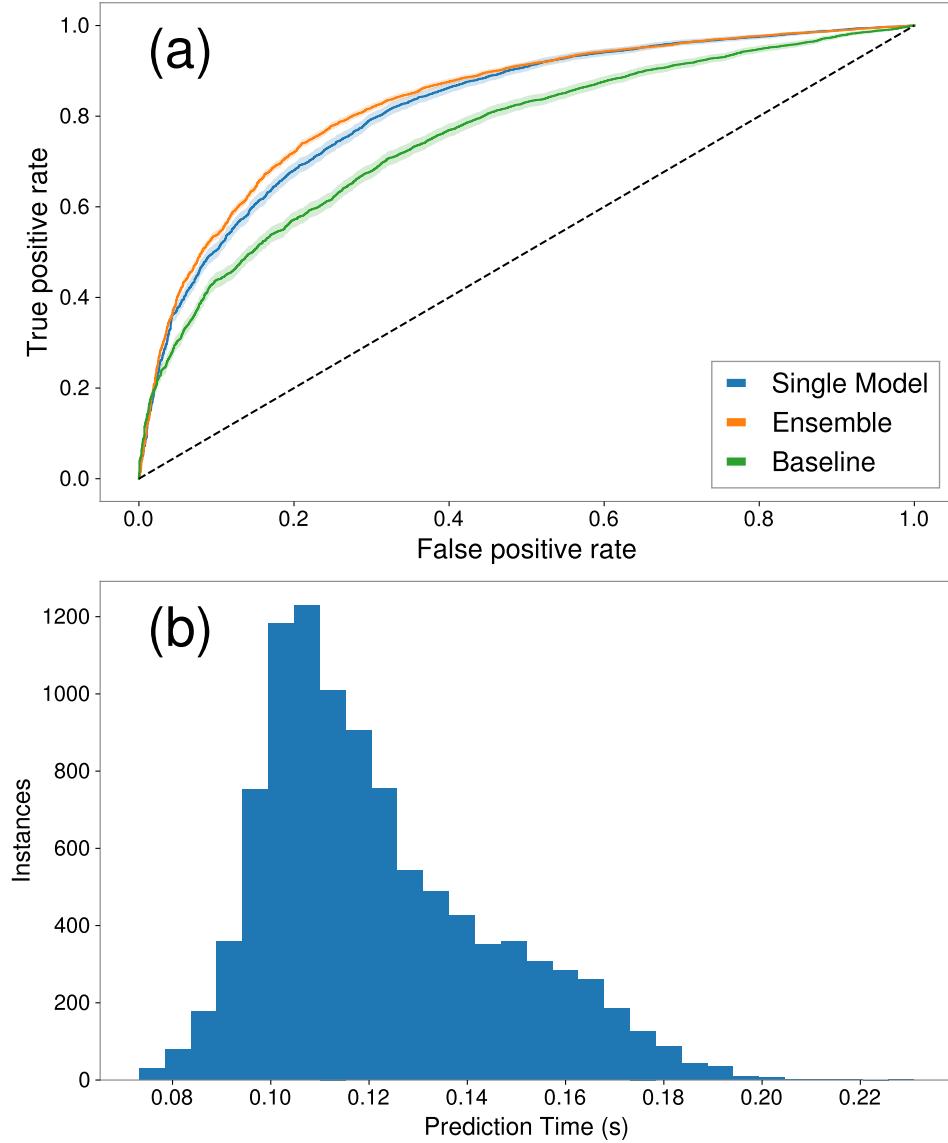


Figure 4.2: Model performance metrics. (a) An ROC curve highlighting the performance of the three models. The region highlight around the curves represents a 95% confidence interval. (b) The prediction speed of the best single model on the first 10000 MDC 11 candidates.

4.2 Interpretations

4.2.1 Saliency

In order to draw conclusions from a large population of saliency maps, we calculate a correlation score between saliency maps of all MDC 11 testing data and the skymap data fed into the model, as well as random noise. The correlations between the images are computed using the following

equation.

$$\text{cor} = \frac{\sum (\text{im1} \cdot \text{im2})}{\sqrt{\sum (\text{im1} \cdot \text{im1}) \cdot \sum (\text{im2} \cdot \text{im2})}} \quad (4.1)$$

where im1 and im2 denote the flattened vector representations of two images of interest.

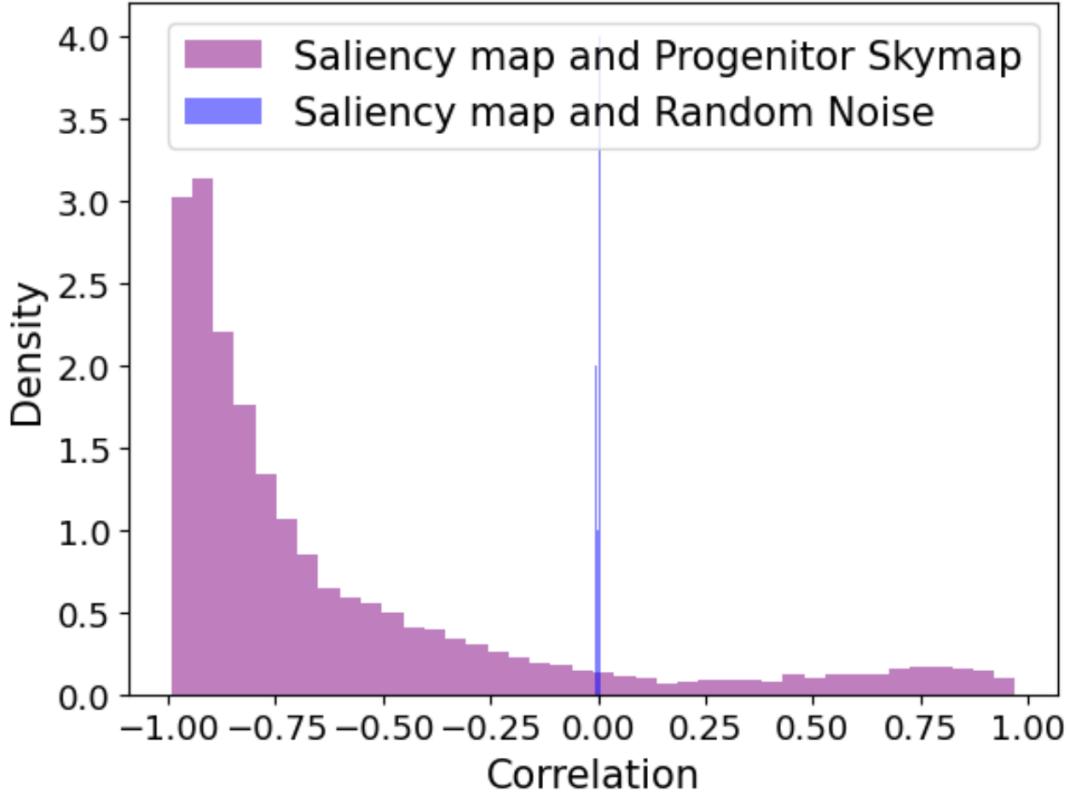


Figure 4.3: Correlation of MDC 11 saliency maps. Blue highlights correlation between saliency maps and their progenitor skymaps. Orange highlights correlation between saliency maps and random noise.

We find that the majority of these saliency maps are highly anti-correlated with their input skymaps. By visual inspection we find that this suggests that, by the definition of a gradient, a slight increase in pixel value in a skymap increases the model’s confidence that a candidate is a glitch. Moreover, by visual inspection many skymaps exhibit a ring of highly negative gradients surrounding areas of large localization probability. This seems to suggest that uncertainty in regions of high localization probability are glitch-like. This also supports the overarching idea that larger localizations tend to be associated with glitches and smaller localizations, events. These conclusions are further supported by the lack of correlation between the input data and

random noise, implying that patterns arising from the maps are highly dependent on the input data.

4.2.2 Occlusion

Our findings from analyzing the occlusion maps generated from the single best model suggests that smaller localization sizes correspond to events, and larger localization sizes correspond to glitches. Moreover, in general as a map is occluded and the probabilities become more localized the map is considered more event-like. This is illustrated in Fig. 4.4, where we can see that in the majority cases as a skymap occludes it tends to become more event-like. However, in certain examples it can be seen that this is not the case and that the localization score seems to decrease as it becomes more localized. This may arise from the interplay of geometry and the distribution of probability in the maps. It should also be noted that these maps that arise from occlusion may not correspond to physically feasible conditions.

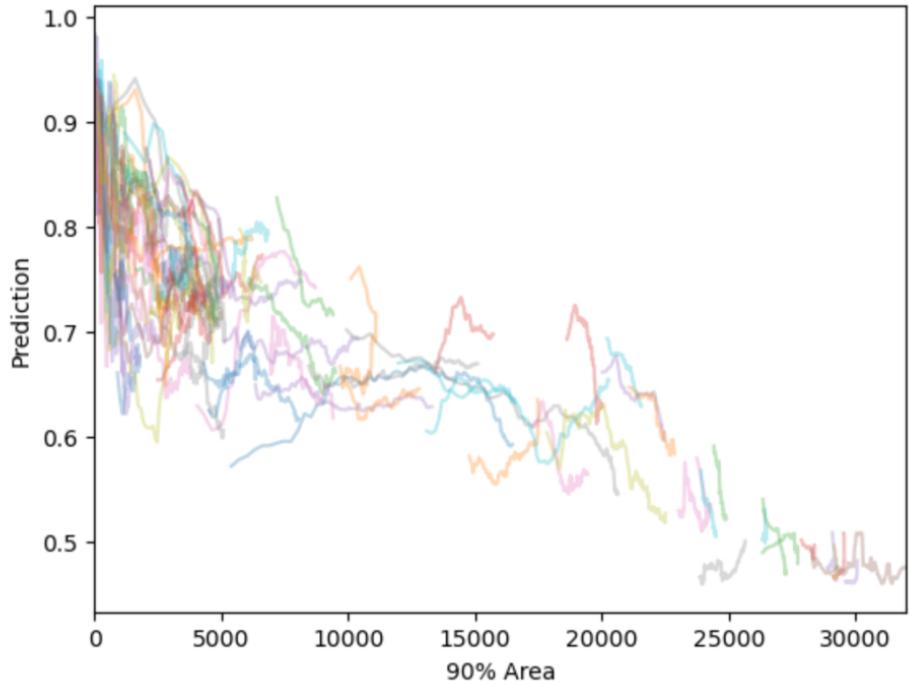


Figure 4.4: 500 occlusion runs of MDC 11 data using the best single model. Each occlusion begins at the highest valued pixel. Each run is represented by a different colored line. The start of the run and the score of the unoccluded map is furthest to the right, and as more of the map is occluded the 90% area decreases and the new prediction score is plotted.

4.2.3 Visualization

We consider two cases: a glitch evolving to become more event-like and an event evolving to become more glitch-like. In both cases, feature visualization has been shown to alter the prediction score as expected in Fig. 4.5 (a) and (e).

Our findings suggest that localizations with a smaller positional uncertainty tend to be evaluated as more “event-like” whereas larger uncertainties correspond to more “glitch-like” candidates.

In the case of a glitch evolving to become event-like, the final prediction score after visualization resembled an almost certain event classification. Confidence in event classification increased when bright regions in the skymap were brightened further, and dimmer regions were minimized.

Furthermore, in specific cases, we also observed slight changes in geometry of the localization pattern, since the probability of localization directly surrounding a bright region is increased. This indicates that regions of high contrast and highly localized probability within the skymap may indicate to the model that a candidate is more event-like and that there may be a characteristic size associated with these regions.

In the case of an event evolving to become more glitch-like, the final prediction score after visualization resembled an almost certain glitch classification. Confidence in glitch classification increased when bright regions in the skymap were dimmed, and dimmer regions were brightened. However, changes in skymap geometry did not affect the classification confidence in this case, as the iteration process seemed to increase the value of almost every pixel at each step. Additionally, when an event evolved to become more glitch-like, we did not observe changes in geometry of the localization pattern. These findings suggest that contrast between bright and dim regions in the skymap, as well as changes in geometry of the localization pattern, do not help to define glitch-like qualities to the model. Rather, as supported by the findings in the saliency map, glitch-like candidates may be primarily characterized by smeared out localization probability over the entire skymap.

This method allows precise analysis of individual skymaps, however due to the nature of skymap localization, difficulty is encountered when generalizing results. This is due to the fact

that the features of a skymap relevant to classification can be ascertained through a visual inspection of the saliency map, but since skymaps localization size, shape, and distribution can vary significantly general explainability requires

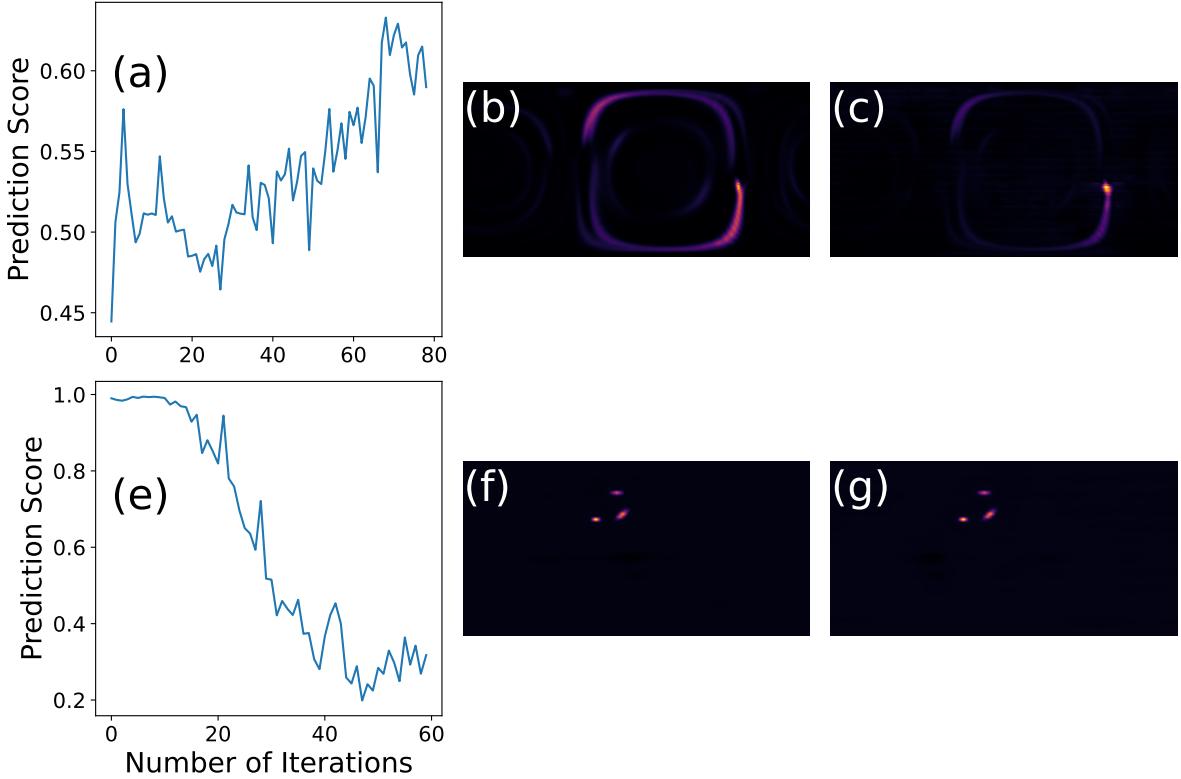


Figure 4.5: A figure showcasing the visualization of a “Scratchy” glitch, shown before in (b) and after in (c), adopting event-like characteristics through feature visualization (top) and a simulated gravitational wave event, shown before in (f) and after in (g), adopting glitch like characteristics. Plots (a) and (e) showcase the change in loss function as the skymaps are iteratively updated using backpropagation. These graphics reflect the intention to maximize our prediction score in (a) and minimize it in (e). The reduction in localization area seen in the transition from (b) to (c) seems to agree with this characteristic and plays a critical role in classification. On the other hand, figures (f) and (g) seem to imply that geometrical features in a sky map may play less of a role than overall localization area in the prediction of glitches.

4.2.4 Classifications

We can gain insight into what features our model values by looking at how its classifications vary between different candidates. More specifically we consider the events and glitches the model is most sure are events and glitches respectively. Additionally, we can look at the maximally misclassified candidates, where the model is very sure of one label but the ground truth is the

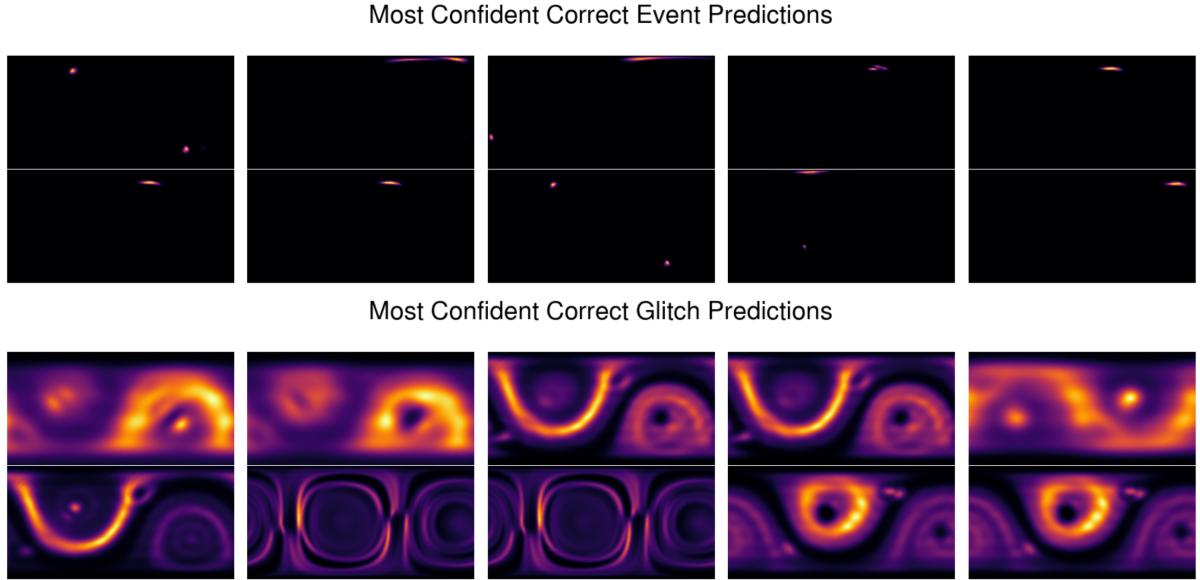


Figure 4.6: The 10 events and glitches that the model correctly classified with the highest confidence. All candidates were correctly classified.

opposite.

As seen in the most confident correct classifications of events the model seems to prefer small localization uncertainty in evaluating a candidate's "event-like" nature. Conversely, the correctly classified candidates that the model thought were most "glitch-like" all correspond to candidates with high localization uncertainty.

On the other hand, we can also gain insight into the model's classification process by observing what it gets most wrong. We observe that the trends described above also apply to these incorrectly classified predictions, mainly that low localization uncertainty seems to tell the classifier that a candidate is "event-like" and vice versa. This is a result that is consistent with our findings from the analysis of saliency maps and occlusion.

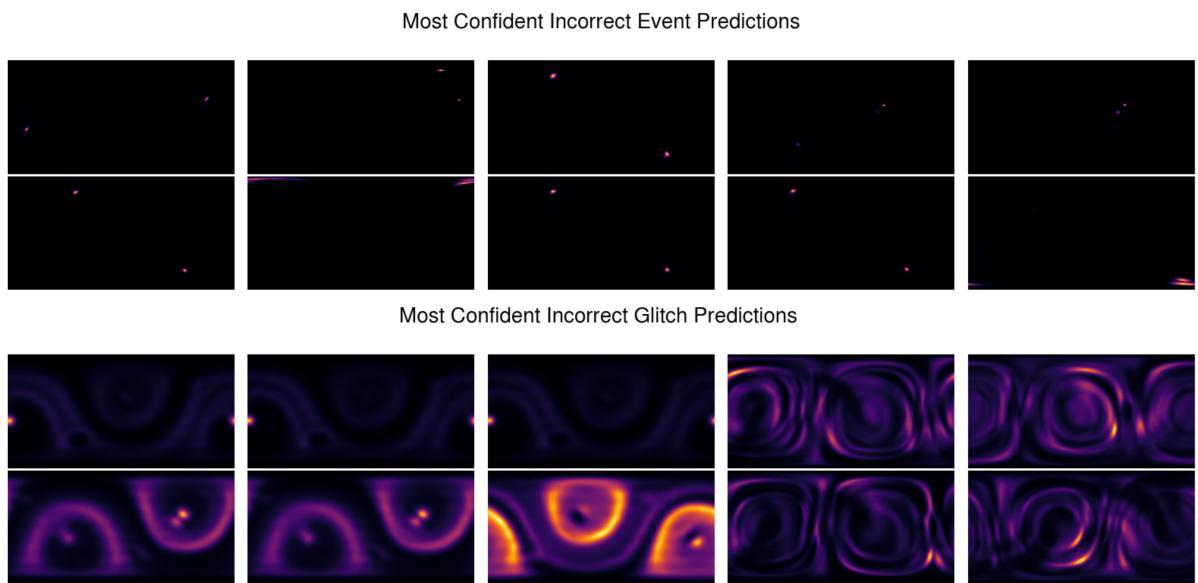


Figure 4.7: The 10 events and glitches most confidently incorrectly classified by the model. For all these candidates the model is very confident that they are described by the incorrect label. With prediction scores close to those observed in Fig 4.6.

Chapter 5

Discussion

In this study, we present GWSkyNetLite, a modified version of GWSkyNet designed to take advantage of rapid localizations of gravitational wave candidates. This model has the potential to be incorporated into a search pipeline to enhance precision, thereby guaranteeing prompt and precise responses to gravitational wave occurrences. Furthermore, by improving real-time performance, the chances of effective electromagnetic radiation (EMR) follow-up increase significantly.

The performance in both speed and accuracy of this model is promising considering the constraints imposed by the HSLM. For the ensemble model, 20% FPR, a TPR of 72% is welcomed, especially in relation to the baseline model. Moreover, if incorporated into a search pipeline the output of this model would inform the calculation of a candidate's false alarm rate, which could be included further in into background estimates and astrophysical probability calculations. As such, the new classifying power brought about by GWSkyNetLite would be valuable. To ensure that the predictions that this model makes are not degenerate with those currently employed we compare our results with the χ^2 metric used in candidate classification and find a lack of strong correlation.

Since this is the case, it suggests that GWSkyNetLite may improve performance when combined with current search pipeline models, as it provides novel classification information not presently available.

A possible extension of this work would be analyzing the trade-off between larger image sizes, which appear to improve performance, and the time taken to generate a prediction. Moreover, different ensembling methods, such as a veto process, may increase the model's robustness to outliers. It should be noted that significant performance gains could also be achieved by increasing the accuracy and metadata conferred by the localization method, as the current

GWSkyNet performs significantly better on the same data set localized with BAYESTAR.

In our investigations of interpretability, we have shed light on potential features hidden in sky localization that may be associated with glitches and events. According to our analysis of saliency maps, we find that a small rise in pixel intensity surrounding an area of high probability tends to decrease the prediction score, leading the classification towards a “glitch” designation. Given the nature of skymap localization, most of the map consists of regions where the probability is small, so an increase in pixel intensity could indicate a potential enlargement of the skymap area. This higher level of uncertainty supports the notion that the size of the skymap localization significantly influences event classification. This is further supported by the lack of correlation saliency maps demonstrated with regard to random noise.

We have also found through feature visualization, occlusion testing, and an assessment of the model’s classifications that low localization areas tends to result in an event classification, whereas larger localizations tend to result in the model thinking the candidate is a glitch. This is in line with previous work, such as the work done by Raza et al., which also suggests that the localization area plays a large role.

Bibliography

- [1] B. P. Abbott, R. Abbott, T. D. Abbott, M. R. Abernathy, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R. X. Adhikari, et al. (LIGO Scientific Collaboration and Virgo Collaboration), Phys. Rev. Lett. **116**, 061102 (2016).
- [2] R. Abbott, T. Abbott, F. Acernese, K. Ackley, C. Adams, N. Adhikari, R. Adhikari, V. Adya, C. Affeldt, D. Agarwal, et al., Physical Review X **13** (2023), ISSN 2160-3308, URL <http://dx.doi.org/10.1103/PhysRevX.13.041039>.
- [3] V. D. Favero, *Constraints on compact binary formation and effective gravitational wave likelihood approximation* (2022), [2209.03790](#).
- [4] S. Vitale, R. Lynch, R. Sturani, and P. Graff, Classical and Quantum Gravity **34**, 03LT01 (2017), ISSN 1361-6382, URL <http://dx.doi.org/10.1088/1361-6382/aa552e>.
- [5] D. Reitze, R. X. Adhikari, S. Ballmer, B. Barish, L. Barsotti, G. Billingsley, D. A. Brown, Y. Chen, D. Coyne, R. Eisenstein, et al., *Cosmic explorer: The u.s. contribution to gravitational-wave astronomy beyond ligo* (2019), [1907.04833](#).
- [6] M. Maggiore, C. V. D. Broeck, N. Bartolo, E. Belgacem, D. Bertacca, M. A. Bizouard, M. Branchesi, S. Clesse, S. Foffa, J. García-Bellido, et al., Journal of Cosmology and Astroparticle Physics **2020**, 050–050 (2020), ISSN 1475-7516, URL <http://dx.doi.org/10.1088/1475-7516/2020/03/050>.
- [7] D. V. Martynov, E. D. Hall, B. P. Abbott, R. Abbott, T. D. Abbott, C. Adams, R. X. Adhikari, R. A. Anderson, S. B. Anderson, K. Arai, et al., Phys. Rev. D **93**, 112004 (2016), URL <https://link.aps.org/doi/10.1103/PhysRevD.93.112004>.
- [8] P. Covas, R. Schofield, R. Savage, and A. Effler, *Why the gw channel detects thirsty black ravens along with colliding black holes* (2017).

- [9] B. S. Sathyaprakash and B. F. Schutz, Living Reviews in Relativity **12** (2009), ISSN 1433-8351, URL <http://dx.doi.org/10.12942/lrr-2009-2>.
- [10] D. Kasen, B. Metzger, J. Barnes, E. Quataert, and E. Ramirez-Ruiz, Nature **551**, 80–84 (2017).
- [11] B. Margalit and B. D. Metzger, The Astrophysical Journal Letters **850** (2017).
- [12] V. Gammaldi, Frontiers in Astronomy and Space Sciences **6** (2019).
- [13] R. Poggiani, Proceedings of Frontier Research in Astrophysics – III — PoS(FRAPWS2018) (2019).
- [14] B. P. Abbott, R. Abbott, T. D. Abbott, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R. X. Adhikari, V. B. Adya, et al. (LIGO Scientific Collaboration and Virgo Collaboration), Phys. Rev. Lett. **119**, 161101 (2017).
- [15] C. Kouveliotou, C. A. Meegan, G. J. Fishman, N. P. Bhat, M. S. Briggs, T. M. Koshut, W. S. Paciesas, and G. N. Pendleton, **413**, L101 (1993).
- [16] D. A. Coulter, R. J. Foley, C. D. Kilpatrick, M. R. Drout, A. L. Piro, B. J. Shappee, M. R. Siebert, J. D. Simon, N. Ulloa, D. Kasen, et al., Science **358**, 1556–1558 (2017), ISSN 1095-9203, URL <http://dx.doi.org/10.1126/science.aap9811>.
- [17] M. Nynka, J. J. Ruan, D. Haggard, and P. A. Evans, The Astrophysical Journal Letters **862**, L19 (2018), ISSN 2041-8213, URL <http://dx.doi.org/10.3847/2041-8213/aad32d>.
- [18] S. Antier, S. Agayeva, M. Almualla, S. Awiphan, A. Baransky, K. Barynova, S. Beradze, M. Blažek, M. Boér, O. Burkhonov, et al., Monthly Notices of the Royal Astronomical Society **497**, 5518 (2020), ISSN 0035-8711, <https://academic.oup.com/mnras/article-pdf/497/4/5518/33706664/staa1846.pdf>, URL <https://doi.org/10.1093/mnras/staa1846>.
- [19] J. C. Rastinejad, B. P. Gompertz, A. J. Levan, W.-f. Fong, M. Nicholl, G. P. Lamb, D. B. Malesani, A. E. Nugent, S. R. Oates, N. R. Tanvir, et al., Nature **612**, 223–227 (2022).

- [20] I. Arcavi, The Astrophysical Journal **855** (2018).
- [21] A. Einstein and N. Rosen, Journal of the Franklin Institute **223**, 43 (1937), ISSN 0016-0032.
- [22] B. S. Sathyaprakash and B. F. Schutz, Living Reviews in Relativity **12** (2009).
- [23] B. Ewing, R. Huxford, D. Singh, L. Tsukada, C. Hanna, Y.-J. Huang, P. Joshi, A. K. Y. Li, R. Magee, C. Messick, et al., *Performance of the low-latency gstellal inspiral search towards ligo, virgo, and kagra's fourth observing run* (2023), [2305.05625](#).
- [24] A. H. Nitz, T. Dal Canton, D. Davis, and S. Reyes, Physical Review D **98** (2018), ISSN 2470-0029, URL <http://dx.doi.org/10.1103/PhysRevD.98.024050>.
- [25] T. Adams, D. Buskulic, V. Germain, G. M. Guidi, F. Marion, M. Montani, B. Mouris, F. Piergiovanni, and G. Wang, Classical and Quantum Gravity **33**, 175012 (2016), ISSN 1361-6382, URL <http://dx.doi.org/10.1088/0264-9381/33/17/175012>.
- [26] Q. Chu, M. Kovalam, L. Wen, T. Slaven-Blair, J. Bosveld, Y. Chen, P. Clearwater, A. Codoreanu, Z. Du, X. Guo, et al., Phys. Rev. D **105**, 024023 (2022), URL <https://link.aps.org/doi/10.1103/PhysRevD.105.024023>.
- [27] C. M. Biwer, C. D. Capano, S. De, M. Cabero, D. A. Brown, A. H. Nitz, and V. Raymond, Publications of the Astronomical Society of the Pacific **131**, 024503 (2019), ISSN 1538-3873, URL <http://dx.doi.org/10.1088/1538-3873/aaef0b>.
- [28] B. Moe, P. Brady, B. Stephens, E. Katsavounidis, R. Williams, and F. Zhang, *Gracedb: A gravitational wave candidate event database* (2014).
- [29] T. L. S. Collaboration, the Virgo Collaboration, the KAGRA Collaboration, R. Abbott, T. D. Abbott, F. Acernese, K. Ackley, C. Adams, N. Adhikari, R. X. Adhikari, et al., *Gwtc-3: Compact binary coalescences observed by ligo and virgo during the second part of the third observing run* (2023), [2111.03606](#).
- [30] L. P. Singer and L. R. Price, Physical Review D **93** (2016).
- [31] M. Cabero, A. Mahabal, and J. McIver, The Astrophysical Journal **904**, L9 (2020).

Bibliography

- [32] T. C. Abbott, E. Buffaz, N. Vieira, M. Cabero, D. Haggard, A. Mahabal, and J. McIver, The Astrophysical Journal **927**, 232 (2022), ISSN 1538-4357, URL <http://dx.doi.org/10.3847/1538-4357/ac5019>.
- [33] B. P. Abbott, R. Abbott, T. D. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, V. B. Adya, C. Affeldt, M. Agathos, et al., Living Reviews in Relativity **23** (2020), ISSN 1433-8351, URL <http://dx.doi.org/10.1007/s41114-020-00026-9>.
- [34] C. Pankow, K. Chatzioannou, E. A. Chase, T. B. Littenberg, M. Evans, J. McIver, N. J. Cornish, C.-J. Haster, J. Kanner, V. Raymond, et al., Physical Review D **98** (2018), ISSN 2470-0029, URL <http://dx.doi.org/10.1103/PhysRevD.98.084016>.
- [35] G. Gao, Y. Yu, J. Yang, G.-J. Qi, and M. Yang, IEEE Transactions on Circuits and Systems for Video Technology **32**, 2550–2560 (2022).
- [36] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santa-maria, M. A. Fadhel, M. Al-Amidie, and L. Farhan, Journal of Big Data **8** (2021).
- [37] J. Lever, M. Krzywinski, and N. Altman, Nature Methods **13**, 703–704 (2016).
- [38] B. P. Abbott, R. Abbott, T. D. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, V. B. Adya, C. Affeldt, M. Agathos, et al., Living Reviews in Relativity **23** (2020), ISSN 1433-8351, URL <http://dx.doi.org/10.1007/s41114-020-00026-9>.
- [39] J. Aasi, B. P. Abbott, R. Abbott, T. Abbott, M. R. Abernathy, K. Ackley, C. Adams, T. Adams, P. Addesso, R. X. Adhikari, et al., Classical and Quantum Gravity **32**, 074001 (2015), ISSN 1361-6382, URL <http://dx.doi.org/10.1088/0264-9381/32/7/074001>.
- [40] F. Acernese, M. Agathos, K. Agatsuma, D. Aisa, N. Allemandou, A. Allocca, J. Amarni, P. Astone, G. Balestri, G. Ballardin, et al., Classical and Quantum Gravity **32**, 024001 (2014), ISSN 1361-6382, URL <http://dx.doi.org/10.1088/0264-9381/32/2/024001>.
- [41] J. Glanzer, S. Banagiri, S. B. Coughlin, S. Soni, M. Zevin, C. P. L. Berry, O. Patane, S. Bahaadini, N. Rohani, K. Crowston, et al., Classical and Quantum Gravity **40**, 065004 (2023), ISSN 1361-6382, URL <http://dx.doi.org/10.1088/1361-6382/acb633>.

- [42] LIGO Scientific Collaboration, Virgo Collaboration, and KAGRA Collaboration, *LVK Algorithm Library - LALSuite*, Free software (GPL) (2018).
- [43] S. S. Chaudhary, A. Toivonen, G. Waratkar, G. Mo, D. Chatterjee, S. Antier, P. Brockill, M. W. Coughlin, R. Essick, S. Ghosh, et al., *Low-latency gravitational wave alert products and their performance in anticipation of the fourth ligo-virgo-kagra observing run* (2023), [2308.04545](#).
- [44] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, *A comprehensive survey on transfer learning* (2020), [1911.02685](#).
- [45] S. Fairhurst, Classical and Quantum Gravity **28**, 105021 (2011), ISSN 1361-6382, URL <http://dx.doi.org/10.1088/0264-9381/28/10/105021>.
- [46] M. L. Chan, PhD thesis, University of Glasgow (2018).
- [47] J. Aasi, J. Abadie, B. P. Abbott, R. Abbott, T. D. Abbott, M. Abernathy, T. Accadia, F. Acernese, C. Adams, T. Adams, et al. (LIGO-Virgo Scientific Collaboration), Phys. Rev. D **88**, 062001 (2013), URL <https://link.aps.org/doi/10.1103/PhysRevD.88.062001>.
- [48] T. Tsutsui, K. Cannon, and L. Tsukada, Physical Review D **103** (2021).
- [49] K. M. Gorski, E. Hivon, A. J. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelmann, The Astrophysical Journal **622**, 759–771 (2005), ISSN 1538-4357, URL <http://dx.doi.org/10.1086/427976>.
- [50] L. B. de Amorim, G. D. Cavalcanti, and R. M. Cruz, Applied Soft Computing **133**, 109924 (2023), ISSN 1568-4946, URL <http://dx.doi.org/10.1016/j.asoc.2022.109924>.
- [51] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al., *TensorFlow: Large-scale machine learning on heterogeneous systems* (2015), software available from tensorflow.org, URL <https://www.tensorflow.org/>.
- [52] F. Chollet et al., *Keras*, <https://keras.io> (2015).

- [53] V. Biscione and J. S. Bowers, *Convolutional neural networks are not invariant to translation, but they can learn to be* (2021), [2110.05861](#).
- [54] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition* (2015), [1512.03385](#).
- [55] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, *Attention is all you need* (2023), [1706.03762](#).
- [56] Z. Dai, H. Liu, Q. V. Le, and M. Tan, *Coatnet: Marrying convolution and attention for all data sizes* (2021), [2106.04803](#).
- [57] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition* (2015), [1409.1556](#).
- [58] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, *Densely connected convolutional networks* (2018), [1608.06993](#).
- [59] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, *Aggregated residual transformations for deep neural networks* (2017), [1611.05431](#).
- [60] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, *Going deeper with convolutions* (2014), [1409.4842](#).
- [61] W. Yu, P. Zhou, S. Yan, and X. Wang, *Inceptionnext: When inception meets convnext* (2023), [2303.16900](#).
- [62] X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer, 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022).
- [63] S. Watanabe, *Tree-structured parzen estimator: Understanding its algorithm components and their roles for better empirical performance* (2023), [2304.11127](#).
- [64] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, *Hyperband: A novel bandit-based approach to hyperparameter optimization* (2018), [1603.06560](#).
- [65] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2019).

Bibliography

- [66] M. Ganaie, M. Hu, A. Malik, M. Tanveer, and P. Suganthan, Engineering Applications of Artificial Intelligence **115**, 105151 (2022), ISSN 0952-1976, URL <http://dx.doi.org/10.1016/j.engappai.2022.105151>.
- [67] M. Ntampaka and A. Vikhlinin, The Astrophysical Journal **926**, 45 (2022).
- [68] K. Simonyan, A. Vedaldi, and A. Zisserman, in *Workshop at International Conference on Learning Representations* (2014).
- [69] C. Olah, A. Mordvintsev, and L. Schubert, Distill (2017), <https://distill.pub/2017/feature-visualization>.
- [70] M. E. Akbiyik, *Data augmentation in training cnns: Injecting noise to images* (2023), [2307.06855](https://arxiv.org/abs/2307.06855).