

Comparative study of 1D and 2D convolutional neural network models with attribution analysis for gravitational wave detection from compact binary coalescences

Seiya Sasaoka¹, Naoki Koyama², Diego Dominguez¹, Yusuke Sakai,³
Kentaro Somiya,¹ Yuto Omae⁴, and Hiroataka Takahashi^{3,5,6}

¹*Department of Physics, Tokyo Institute of Technology,
2-12-1 Ookayama, Meguro-ku, Tokyo 152-8551, Japan*

²*Graduate School of Science and Technology, Niigata University,*

8050 Ikarashi-2-no-cho, Nishi-ku, Niigata City, Niigata 950-2181, Japan

³*Department of Design and Data Science and Research Center for Space Science,
Advanced Research Laboratories, Tokyo City University,*

3-3-1 Ushikubo-Nishi, Tsuzuki-ku, Yokohama, Kanagawa 224-8551, Japan


⁴*Artificial Intelligence Research Center, College of Industrial Technology, Nihon University,
1-2-1 Izumi-cho, Narashino, Chiba 275-8575, Japan*

⁵*Institute for Cosmic Ray Research (ICRR), The University of Tokyo,*

5-1-5 Kashiwa-no-Ha, Kashiwa City, Chiba 277-8582, Japan

⁶*Earthquake Research Institute, The University of Tokyo,*

1-1-1 Yayoi, Bunkyo-ku, Tokyo 113-0032, Japan

 (Received 8 December 2023; accepted 22 January 2024; published 7 February 2024)

Recent advancements in gravitational wave astronomy have seen the application of convolutional neural networks (CNNs) in signal detection from compact binary coalescences. This study presents a comparative analysis of two CNN architectures: one-dimensional (1D) and two-dimensional (2D) along with an ensemble model combining both. We trained these models to detect gravitational wave signals from binary black hole (BBH) mergers, neutron star-black hole (NSBH) mergers, and binary neutron star (BNS) mergers within real detector noise. Our investigation entailed a comprehensive evaluation of the detection performance of each model type across different signal classes. To understand the models' decision-making processes, we employed feature map visualization and attribution analysis. The findings revealed that while the 1D model showed superior performance in detecting BBH signals, the 2D model excelled in identifying NSBH and BNS signals. Notably, the ensemble model outperformed both individual models across all signal types, demonstrating enhanced detection capabilities. Additionally, input feature visualization indicated distinct areas of focus in the data for the 1D and 2D models, emphasizing the effectiveness of their combination.

DOI: [10.1103/PhysRevD.109.043011](https://doi.org/10.1103/PhysRevD.109.043011)

I. INTRODUCTION

The era of gravitational wave (GW) astronomy was inaugurated with the first direct detection of GWs from a binary black hole (BBH) merger by the Advanced Laser Interferometer Gravitational-wave Observatory (Advanced LIGO) [1] in 2015 [2]. This groundbreaking discovery was followed by the first joint observation of GWs and electromagnetic counterparts from a binary neutron star (BNS) merger, achieved by Advanced LIGO, Advanced Virgo [3], and other telescopes, paving the way for multimessenger astronomy [4]. Over the course of three observing runs (O1, O2, and O3), 90 GW events from compact binary coalescences (CBCs) were reported [5–8]. These events included two neutron star-black hole (NSBH) mergers [9] and two BNS mergers [10,11]. The detection of GWs, alongside electromagnetic waves and neutrinos from these

mergers, is vital for understanding the physical properties of neutron star interiors, which are reflected in their equation of state. Now, with the commencement of the fourth observing run (O4) in May 2023, which includes the participation of KAGRA [12], expectations are high for more GW detections from binary systems with neutron stars.

Traditionally, GWs from CBC sources have been analyzed using the matched-filtering technique [13] with theoretical approximants, phenomenological models, and templates derived from numerical simulations [14,15]. In this technique, the signal-to-noise ratio (SNR) is computed by correlating the detector's strain data with each template in a large bank that covers a wide parameter space, taking into account variations in source masses and/or spins. This method, however, can be computationally intensive, especially for complex GW signals that incorporate elements

such as higher-order modes, precession, or orbital eccentricity. This complexity underscores the need for more efficient algorithms to manage the growing volume of GW data.

In response to this challenge, deep learning approaches, particularly convolutional neural networks (CNNs), have been increasingly applied in the GW field. These applications range from parameter estimation of CBC sources [16,17] to sky localization [18–20] and classification of transient noises [21–23]. The effectiveness of CNNs in detecting GWs from BBH mergers was first demonstrated in 2018 by George and Huerta [24] and Gabbard *et al.* [25]. These initial studies have since been expanded to include more sophisticated models that use real detector noise and account for various signal complexities like the spin effect, precession, higher-order modes, or eccentricity [26–40]. There are also some studies targeting BNS [41–48] or NSBH signals [49–53], which are more challenging than BBH signals due to their longer duration and smaller amplitude.

Two main types of CNNs have been employed in this research: one-dimensional (1D) CNNs, which process whitened time-series data, and two-dimensional (2D) CNNs, which analyze time-frequency maps. Although most studies have favored 1D CNNs [24–28,30,31,34–36,39,41–44,46,47,50,52,53], a subset has opted for 2D CNNs [32,33,40,45,48,49,51]. 1D CNNs are preferred for their efficiency in not generating time-frequency maps, thereby reducing processing time. On the other hand, 2D CNNs excel at capturing the temporal evolution of GW frequencies in their input. For the analysis of GWs from core-collapse supernovae, Iess *et al.* [54] conducted a comparative study of 1D and 2D CNNs, alongside long short-term memory networks. Their approach involved combining these models by averaging their outputs. However, to our knowledge, a similar comprehensive comparison of various CNN architectures for CBC sources has not been extensively explored.

A common method for analyzing and interpreting CNNs involves pinpointing the segments of input data that significantly influence the model’s predictions. This analysis can be performed using class activation mapping (CAM) techniques [55] or by assessing the contribution of each input feature to the model’s output. In our prior research [56], we applied CAM techniques to a CNN classifier designed for GWs from core-collapse supernovae. This investigation revealed that the model primarily focused on specific GW modes within the input spectrogram to make predictions.

In the current study, we train both 1D and 2D CNN models to detect and classify GWs from CBC sources. We then develop an ensemble model that combines these two CNN types. Our analysis includes a detailed comparison of the detection performance of these models across each type of CBC signals. To distinguish the different aspects that 1D

and 2D models focus on within the input, we employ the integrated gradients technique [57]. This approach allows us to identify the influential regions in the input that guide the models’ predictions, revealing distinct areas of focus between the 1D and 2D models.

The paper is organized as follows. Section II details our datasets, the architecture of the CNN models, and the theoretical background of the CNN analysis methods. Section III presents the classification performance and a comprehensive analysis of our trained models. Finally, we conclude our findings in Sec. IV.

II. METHOD

Our CNN models are trained to classify strains at three detectors LIGO Hanford (H1), LIGO Livingston (L1), and Virgo (V1) into four distinct classes: BBH, NSBH, BNS, and pure noise. This section provides a detailed description of the datasets used for both training and testing our models. Following this, we describe the architecture and training procedures of the CNN models. Lastly, we address the dimensionality reduction technique implemented in our study, as well as the methodology employed for computing feature attribution, which are crucial for interpreting the models’ decision-making processes.

A. Dataset

To train and test our model, we used nonprecessing CBC signals and injected them into noise obtained from O3 real data at H1, L1, and V1, which are available at the Gravitational Wave Open Science Center [58].

1. Signal and noise generation

To construct our datasets, nonprecessing CBC signals were generated using the LIGO Algorithm Library Suite (LALSuite) [59]. Specifically, BBH signals were simulated using the `SEOBNRv4` approximant [60], based on the effective-one-body method, while NSBH and BNS signals were generated using the `SpinTaylorT4` approximant [61], a time-domain post-Newtonian model incorporating spin effects. For BBH signals, component masses were uniformly sampled in the range of 5 to $80M_{\odot}$. NSBH signals had NS masses sampled between 1 and $2M_{\odot}$, and BH masses between 5 and $35M_{\odot}$. The component masses of BNS signals ranged uniformly from 1 to $2M_{\odot}$. The individual components have spins aligned with the orbital angular momentum, uniformly distributed between 0 and 0.99. These waveforms were sampled at a rate of 4096 Hz. We used four-second data segments, with the merger event uniformly placed between 3.8 and 3.9 sec. Although NSBH and BNS signals are typically longer than 4 sec, we found this segment length is sufficient to discriminate between different classes. The use of shorter-segment signals also reduces the memory requirements for training models. The sky position of the source, defined by

declination and right ascension, was randomly selected, and GW amplitude calculations were performed considering the antenna pattern functions and time delays across detectors. These computations utilized the `PyCBC` library [62].

For noise samples and background noise for signal samples, real strain data from GPS time 1238163456 to 1238659072 was used for the training set, 1238663168 to 1239162880 was used for the validation set, and 1239166976 to 1239875584 was used for the test set. Data around the GW event time reported in the GWTC-2.1 catalog [7] were excluded.

2. Preprocessing

After the signal samples were truncated to four-second segments, they were scaled based on the computed optimal matched-filter SNR, defined as

$$\rho = \sqrt{4 \int_{f_{\min}}^{f_{\max}} \frac{|\tilde{h}(f)|^2}{S_n(f)} df}, \quad (1)$$

where $\tilde{h}(f)$ is the Fourier transform of the truncated signal and $S_n(f)$ is the one-side power spectral density of the noise, estimated using Welch's method [63]. The integration was performed from a cutoff frequency of 20 Hz up to the Nyquist frequency. The training and validation signals were scaled so that the network SNR of the three detectors, given by

$$\rho_{\text{net}} = \sqrt{\rho_{\text{H1}}^2 + \rho_{\text{L1}}^2 + \rho_{\text{V1}}^2}, \quad (2)$$

followed a uniform distribution between 8 and 24, while the SNRs of the test signals ranged from 3 to 24. After each signal was injected in noise, we whitened the sample in frequency domain using the power spectral density. For the input to the 2D CNN model, we generated time-frequency maps using the Q transform [64] of the whitened samples, defined by

$$X(\tau, \phi, Q) = \int_{-\infty}^{\infty} \tilde{x}(f + \phi) \tilde{w}^*(f, \phi, Q) e^{2\pi i f \tau} df, \quad (3)$$

where Q is the quality factor, and the Connes window functions is used as the window function [65].

The final dataset comprised 408,000 training samples, 408,000 validation samples, and 528,000 test samples. Each dataset had an equal distribution of 25% BBH, 25% NSBH, 25% BNS, and 25% pure noise samples. Representative samples from each class in the training set are displayed in Fig. 1.

B. Model

1. 1D CNN

One-dimensional CNN consists of 1D convolutional filters. Let x_i^c be the i th value of the c th channel of the input series and y_i^m be the i th value of the m th channel of the output. The output of a 1D convolutional filter is given by

$$y_i^m = \sum_{c=0}^{C-1} \sum_{k=0}^{K-1} w_k^m x_{i+k}^c + b^m, \quad (4)$$

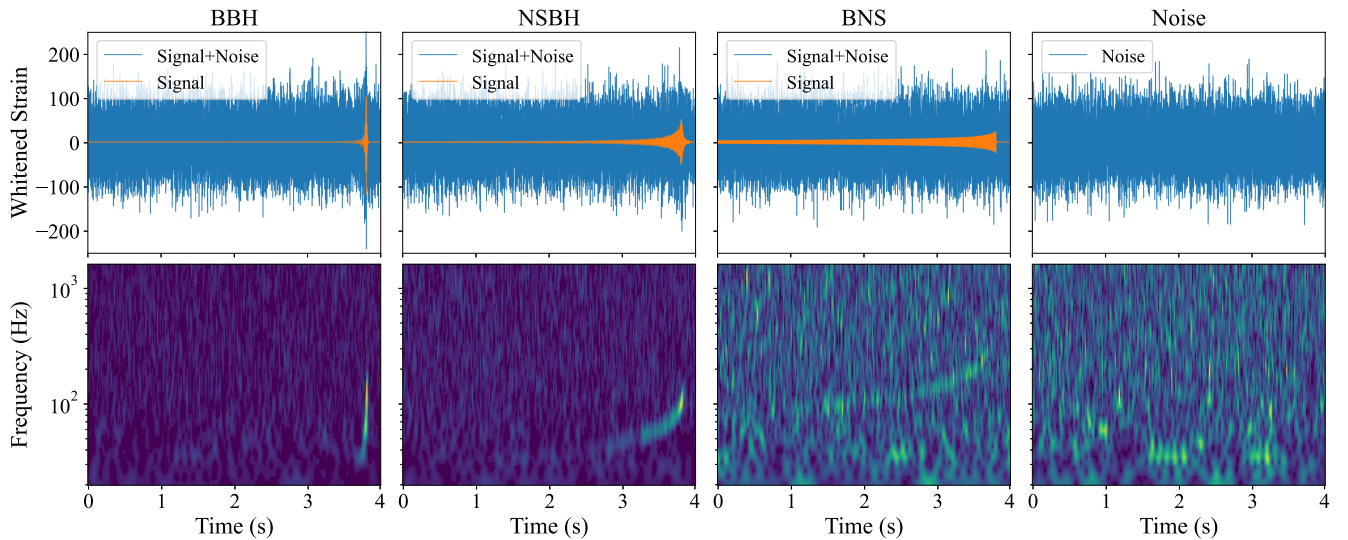


FIG. 1. Example strain data at H1 detector in the training set. The upper figures show the whitened time-series data used as input to the 1D model, and the lower figures show the time-frequency maps for the 2D model. The component masses of the BBH signal are $51.3M_{\odot}$ and $50.9M_{\odot}$, whereas in the NSBH sample, the respective masses are $33.3M_{\odot}$ and $1.83M_{\odot}$, and the masses of the BNS sample are $1.54M_{\odot}$ and $1.40M_{\odot}$. The single-detector SNR of each signal is 15, and the merger time is fixed at 3.9 s.

where K is the kernel size and C is the number of input channels. Weight parameters w and bias parameters b are learned during training processes.

Our 1D CNN model takes a three-channel whitened time series at H1, L1, and V1 as input. Our implementation uses a 54-layer deep residual network (ResNet-54) which was proposed in Ref. [39]. ResNet is a type of deep CNN architecture that uses residual blocks to address the vanishing gradient problem commonly encountered in deep networks. It achieves this by adding *skip connections* between layers, enabling the network to learn residual functions and make training deep networks more efficient [66]. Details of the ResNet-54 model architecture can be found in Ref. [39].

For input normalization, a deep adaptive input normalization layer [67] is employed in this analysis as used in Ref. [39] to address nonstationary noise that appears in real detector noise. In this layer, unlike conventional normalization, shifting and scaling parameters for normalizing the input are optimized during the training. Including the parameters in the normalization layer, the model has a total of 1,935,698 trainable parameters.

The employed architecture is one of the state-of-the-art models in BBH detection that surpassed the matched-filtering pipeline in a specific condition [39]. Alternatively, we also employed the CNN architecture used in Ref. [52] designed to detect all types of CBC signals, however, it did not show a better performance than the ResNet-54 model in our datasets.

2. 2D CNN

Two-dimensional CNN consists of 2D convolutional filters. Let $x_{i,j}^c$ be the (i, j) component of the c th channel of the input image and $y_{i,j}^m$ be the (i, j) component of the c th channel of the output. The output of a 2D convolutional filter is given by

$$y_{i,j}^m = \sum_{c=0}^{C-1} \sum_{k=0}^{K-1} \sum_{l=0}^{L-1} w_{k,l}^m x_{i+k,j+l}^c + b^m, \quad (5)$$

where (K, L) is the kernel sizes and C is the number of input channels. Weight parameters w and bias parameters b are learned during training processes.

The ResNet-50 model [66], a variant of ResNet with proven efficacy in image recognition tasks, forms the basis of our 2D CNN. This model processes three-channel images of time-frequency maps and includes 23,508,548 trainable parameters. We adopt this model because it is one of the most widely used 2D CNNs in GW signal detection and has a similar number of layers to our 1D model. Its efficiency has been validated in previous studies [45,49,51].

3. Ensemble model

In our ensemble approach, we combine the outputs of the 1D and 2D CNN models to enhance predictive performance. This is achieved by first training a fully connected neural network, which takes as input a concatenated vector of features extracted from the trained 1D and 2D models. The input vector, with a dimension of 10240, is processed through a hidden layer of 200 units, outputting a four-dimensional vector. The network incorporates a Leaky ReLU layer [68] and a dropout layer [69] with a 0.25 dropout rate for regularization. The ensemble network comprises 2,049,004 trainable parameters.

For the final model output, we employ a weighted average of the predictions from the 1D, 2D, and ensemble network. The weights, optimized for accuracy on the validation set, are set at 0.4 for each of the 1D and 2D CNNs and 0.2 for the ensemble network. The ensemble model is illustrated in Fig. 2.

4. Training process

Both the 1D and 2D CNN models, as well as the ensemble network, were developed using the PyTorch library [70] and trained on four NVIDIA Tesla V100 GPUs. All models were trained using categorical cross entropy as the loss function and Adam optimizer [71] with an initial learning rate of 10^{-3} . The learning rate was controlled by PyTorch's ReduceLROnPlateau method. During the training of the 1D and 2D CNNs, we

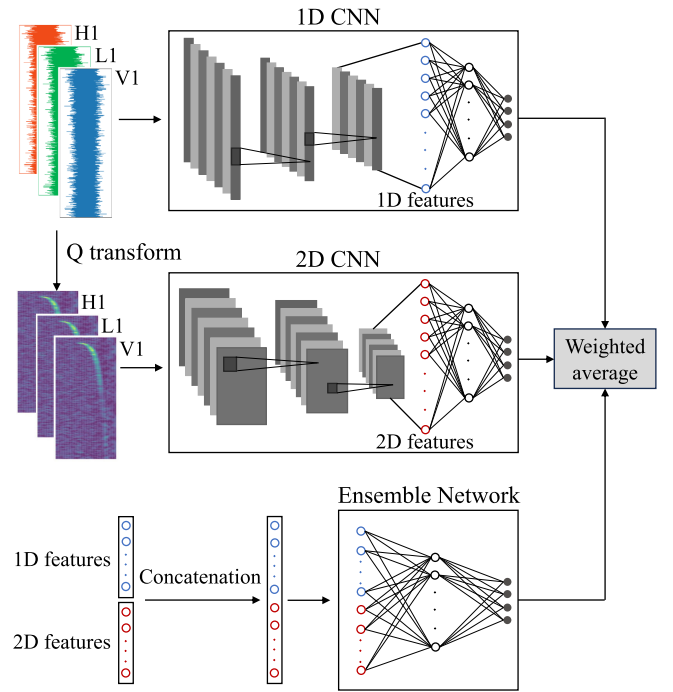


FIG. 2. Illustration of the ensemble model. The final output is the weighted average of the outputs from the 1D CNN, 2D CNN, and ensemble network.

implemented the curriculum learning technique [72]. This method involves initially training with high SNR samples and progressively incorporating lower SNR samples, thereby improving learning efficiency and model performance.

The 1D CNN underwent 300 epochs of training with a minibatch size of 1024. In contrast, the 2D CNN was trained for 45 epochs using a minibatch size of 36. The ensemble network's training lasted for 25 epochs with a minibatch size of 256.

C. t-distributed stochastic neighbor embedding

To understand the ability of 1D and 2D convolutional filters to extract meaningful features from the input for classification, we analyzed feature maps from the final convolutional layer. Given the high-dimensional nature of these feature maps, we utilized the t-distributed stochastic neighbor embedding (t-SNE) technique [73] for dimensionality reduction.

In the original space, the conditional probability $p_{j|i}$ that a data point x_i would pick a data point x_j as its neighbor is modeled as a Gaussian distribution centered at x_i , defined as

$$p_{j|i} = \frac{\exp(-\|x_i - x_j\|^2/2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2/2\sigma_i^2)}, \quad (6)$$

where σ_i is the standard deviation. We define the joint probability p_{ij} as the symmetrized conditional probabilities, which can be expressed as $p_{ij} = (p_{j|i} + p_{i|j})/2n$, where n is the number of data points. In the low-dimensional space, the student t distribution with 1 degree of freedom, defined by

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq i} (1 + \|y_k - y_l\|^2)^{-1}}, \quad (7)$$

is used to quantify the similarity of data points. The optimal low-dimensional representations are obtained by minimizing the Kullback-Leibler divergence of the distributions p_{ij} and q_{ij} , given by

$$C = \sum_{i,j} p_{ij} \log \frac{p_{ij}}{q_{ij}}. \quad (8)$$

The value of σ_i in Eq. (6) is determined by selecting the hyperparameter called perplexity, which is defined as 2 to the power of the Shannon entropy. The perplexity can be interpreted as a measure of the number of valid neighbors, and typical values are between 5 and 50 [73]. We set the perplexity at 25.

D. Integrated gradients

To discern which aspects of the inputs significantly influence the predictions in our trained 1D and 2D models, we employed the integrated gradients method [57]. While class activation mapping techniques [55,74] are commonly used for such analysis, they often yield low-resolution saliency maps, especially in deep models. To circumvent this limitation, the integrated gradients method provides high-resolution feature attribution maps, proving advantageous for our analysis.

The integrated gradients method is grounded in two axioms that attribution methods should satisfy: (i) *sensitivity*, where any difference in one feature between the input and the baseline resulting in different predictions should receive a nonzero attribution, and (ii) *implementation invariance*, where the attributions for two functionally equivalent networks should be always identical.

Consider a function F that represents a network and let x be the input and x' be the baseline input. The feature attribution map is calculated by examining the path from the baseline x' to the input x and accumulating the network's gradients along this path. A point on this path can be expressed as $x' + \alpha(x - x')$ where α varies from 0 to 1. The integrated gradients along the i th dimension for an input x are defined as

$$\text{IG}_i(x) = (x_i - x'_i) \int_0^1 \frac{\partial F(x' + \alpha(x - x'))}{\partial x_i} d\alpha. \quad (9)$$

In practice, this integration is approximated using the Riemann sum, described as

$$\text{IG}_i(x) \approx (x_i - x'_i) \sum_{k=1}^N \frac{\partial F(x' + \frac{k}{N}(x - x'))}{\partial x_i} \frac{1}{N}. \quad (10)$$

Here N represents the number of interpolation steps. For accurately approximating the integral, a step size ranging from 20 to 300 is typically effective [57]. In our implementation, we chose $N = 30$ steps. The Captum library [75] was utilized to compute the attribution maps using the integrated gradients method.

III. RESULTS AND DISCUSSION

A. Model performance

To evaluate the performance of our three models (1D, 2D, and the ensemble model), we first examined the receiver operating characteristic (ROC) curves for each signal type. The ROC curve plots the true alarm probability (TAP) against the false alarm probability (FAP) at various classification thresholds. As depicted in Fig. 3, the ROC curves for each type of GW signal at a fixed network SNR of 8 show distinctive sensitivities.

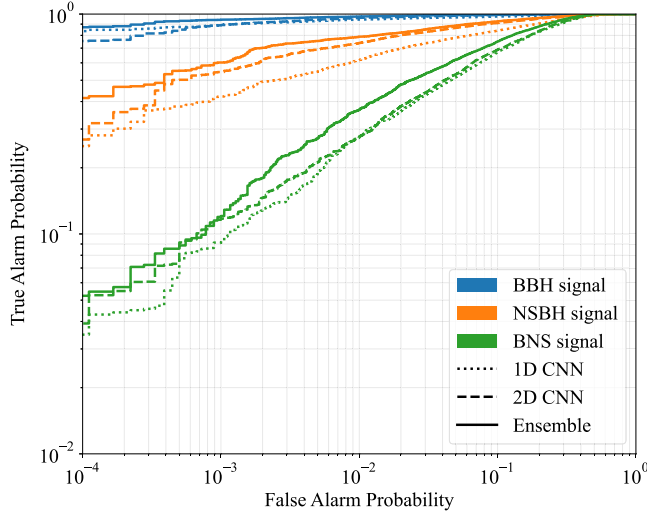


FIG. 3. ROC curves of the three models for BBH, NSBH, and BNS signals at a fixed network SNR of 8. The SNRs are computed with four-second signals.

It was observed that all models exhibited the highest sensitivity to BBH signals, followed by NSBH and BNS signals. This trend aligns with expectations considering the relative amplitude of each signal type. Notably, the ensemble model demonstrated superior performance across all signal types. For BBH signals, the performance ranking was ensemble, followed by the 1D and then the 2D model. In contrast, for NSBH and BNS signals, the 2D model outperformed the 1D model. This variation in performance can be attributed to the transient nature of BBH signals, which are more effectively captured by the 1D convolution in time-series data. Conversely, the smaller amplitudes of NSBH and BNS signals, which are more challenging to identify in time-series data, render the 2D model more effective. This difference highlights the effectiveness of combining the 1D and 2D models.

We further calculated the detection sensitivity for each signal type as a function of network SNR. Figure 4 shows the sensitivity curves for the three models at a fixed FAP of 0.001. The 1D model's sensitivity is on par with that reported in Ref. [52], where the model was trained using single-detector input. Their model's sensitivity saturates at a single-detector SNR of $\rho_{L1} \geq 8$ for BBH signals, at $\rho_{L1} \geq 10$ for NSBH signals, and at $\rho_{L1} \geq 13$ for BNS signals. Our 1D model, however, reaches saturation for BBH signals at $\rho_{\text{net}} \geq 12$, for NSBH signals at $\rho_{\text{net}} \geq 17$, and for BNS signals at $\rho_{\text{net}} \geq 22$. Given that the network SNR of three detectors is roughly $\sqrt{3}$ times that of a single-detector SNR, the performance of our 1D model is consistent with their model. The ensemble model further enhances this performance, lowering the saturation SNRs to 10 for BBH signals, 14 for NSBH signals, and 21 for BNS signals.

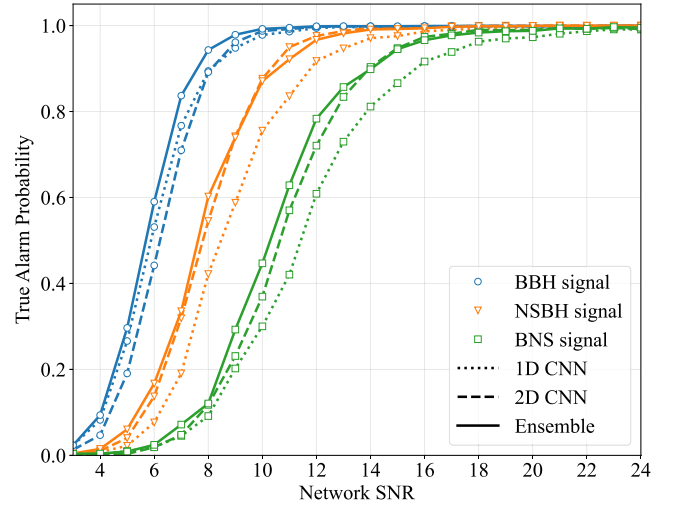


FIG. 4. Sensitivity curves of the three models for BBH, NSBH, and BNS signals at a fixed false alarm probability of 0.001. The SNRs are computed with four-second signals.

B. Feature map

We extract feature maps from the final convolutional layers. Since these vectors are fed into fully connected layers to make predictions, these feature maps represent the characteristics of each class. Figures 5 and 6 display the t-SNE projections of the feature maps for the trained 1D and 2D models, respectively. For these visualizations, we randomly selected 200 samples from the test set. In the figures, the size of each marker representing a signal sample is proportional to its SNR. Smaller markers indicate lower SNR signals, while larger markers correspond to higher SNR signals.

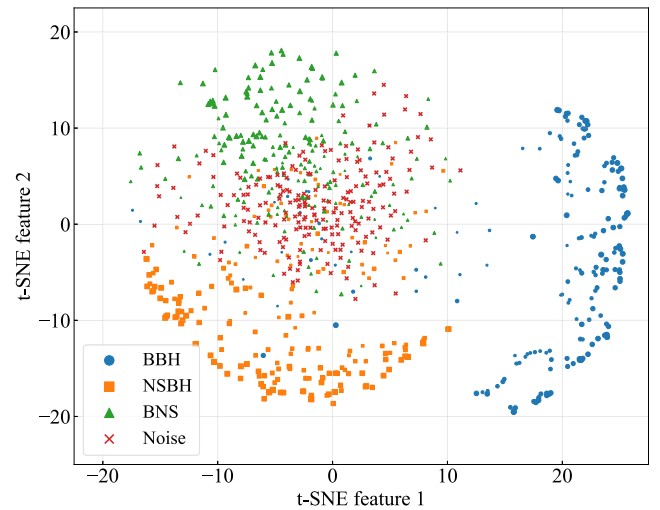


FIG. 5. Two-dimensional representations of feature maps of the 1D model by t-SNE. The marker size of signal sample corresponds to the SNR.

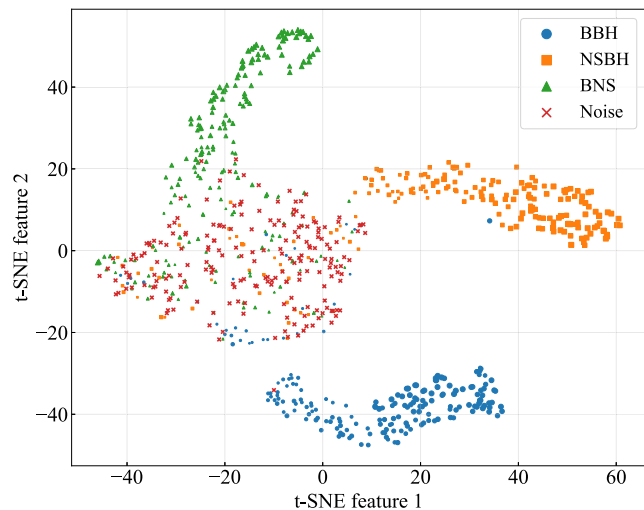


FIG. 6. Two-dimensional representations of feature maps of the 2D model by t-SNE. The marker size of signal sample corresponds to the SNR.

Figure 5 shows that the high-SNR BBH and NSBH samples are distinctly separated from the noise cluster, indicating effective classification of these signals by the 1D model. However, BNS samples are observed to be closer to the noise cluster, suggesting less clear differentiation for this signal type. Low-SNR signals across all types are more diffusely distributed within the noise cluster. In contrast, Fig. 6 indicates that the 2D model has an improved ability to separate not only the BBH and NSBH signals but also the BNS signals from the noise cluster. This indicates that the 2D model may be more proficient at identifying features of BNS signals compared to the 1D model.

Additionally, we explored embedding the feature maps into a three-dimensional space. However, this analysis revealed similar characteristics to those observed in the two-dimensional embeddings.

C. Attribution map

To understand how input features contribute to model predictions, we generated attribution maps for each type of signal using the integrated gradients method. For this analysis, a single signal was randomly selected from the test set, and ten distinct noise samples were added to create 10 different input samples. Attribution maps were produced for each input sample, and their average was computed to discern universal characteristics of the attribution.

Figures 7–9 present the attribution maps for a BBH, NSBH, and BNS signal, respectively, as identified by the 1D and 2D models. Each signal had a fixed network SNR of 20, and the attribution maps for both models were normalized to a [0, 1] range. The bottom plots in Figs. 7–9 show the values of integrated gradients summed over all frequencies for each time bin for 1D and 2D model,

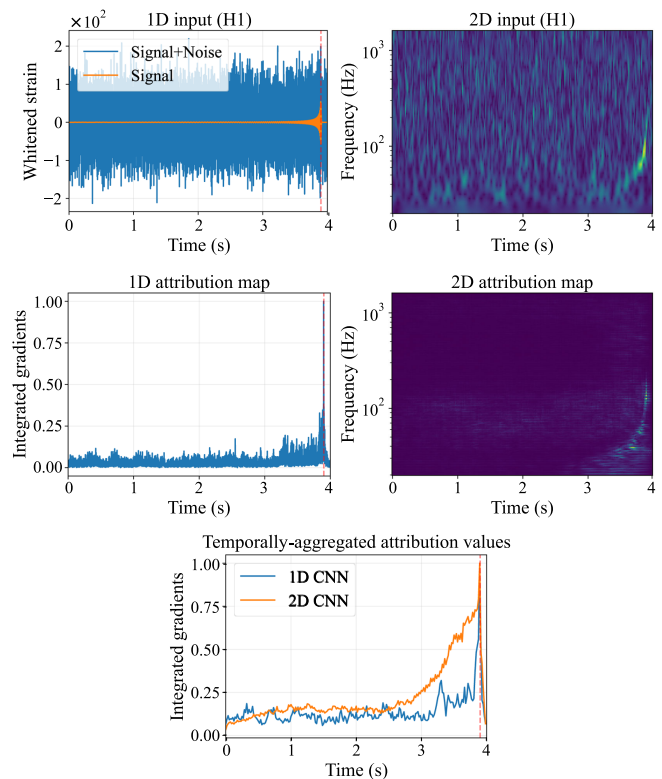


FIG. 7. Attribution maps of the 1D and 2D models for a BBH signal and corresponding input samples at H1 detector. The component masses are $37.7M_{\odot}$ and $6.94M_{\odot}$. The red dashed line shows the time of coalescence. The bottom plot shows the values of integrated gradients summed over all frequencies for each time bin.

respectively. The values are normalized to a maximum integrated gradients of one for each model.

In Fig. 7, the 1D model shows significant contributions from data at the coalescence time of the BBH signal. In contrast, the 2D model's attribution map indicates that the 2D model focuses on the entire inspiral signal. Both the 1D and 2D models exhibit similar characteristics when integrated-gradients values are temporally aggregated, however, the 2D model sees data at more broader time frame than the 1D model.

As for the NSBH signal, the 1D model exhibits multiple peaks in the integrated gradients values before the coalescence, with the peak values progressively increasing, shown in Fig. 8. Since the model not only detects the signal but also classifies it into three classes, the data prior to the time of coalescence seem to be more significant than the data at the time of coalescence for determining that the signal is NSBH, not BBH. In the 2D model, the feature contribution of the entire inspiral is large, as in the case of the BBH signal. Similar characteristics of the NSBH sample are seen in the BNS sample in Fig. 9, but in the case of 1D attribution map of the BNS sample, peaks are

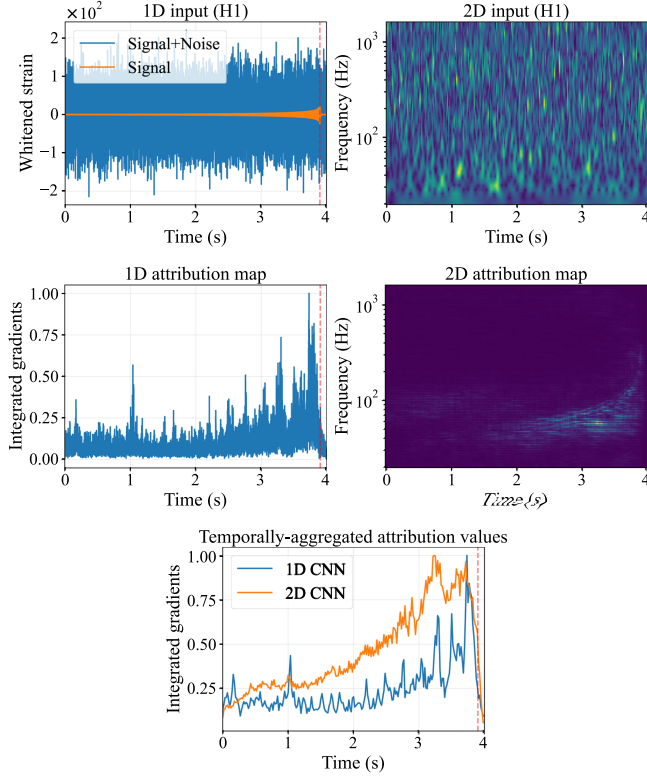


FIG. 8. Attribution maps of the 1D and 2D models for a NSBH signal and corresponding input samples at H1 detector. The component masses are $6.78M_{\odot}$ and $1.99M_{\odot}$. The red dashed line shows the time of coalescence. The bottom plot shows the values of integrated gradients summed over all frequencies for each time bin.

also seen at earlier times and the overall values of the integrated gradients are generally identical. This indicates that the model focuses on various parts of the input time series data, which is reasonable because the inspiral signal is longer than the other signals. The 2D attribution map of the BNS sample shows that the 2D model accurately captures the BNS chirp signal on the input spectrogram, and it demonstrates the consistent performance of the 2D model for BNS events. The temporally aggregated attribution maps have similar characteristics for each signal type for the 1D and 2D models, but the 2D model shows a greater emphasis on longer signal durations than the 1D model.

In summary, from the attribution maps, we observe that the 1D model places greater emphasis on the time preceding coalescence, especially as the waveform lengthens, with significant contributions from specific moments in the inspiral phase. Conversely, the 2D model assesses the entire chirp waveform in the spectrogram, classifying based on the shape of the chirp, i.e., the temporal evolution of its frequency.

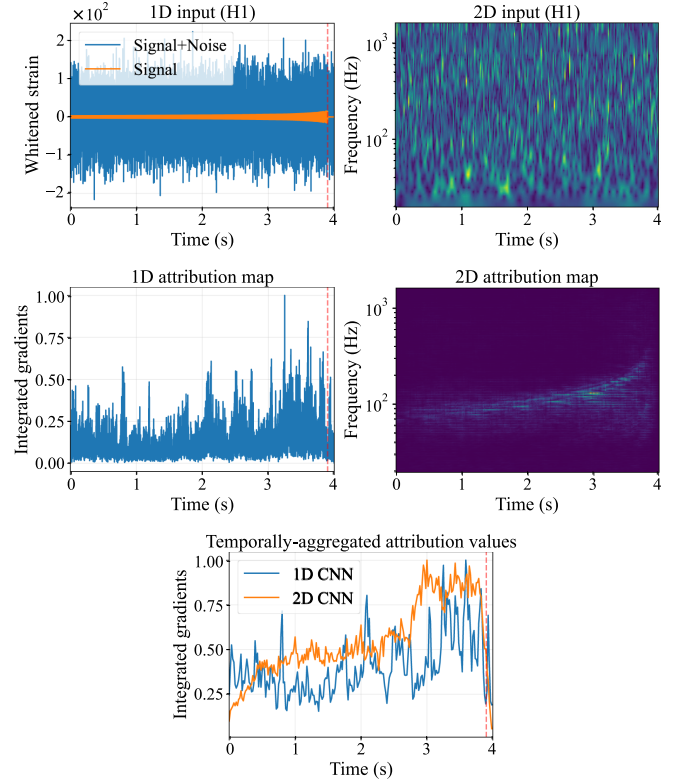


FIG. 9. Attribution maps of the 1D and 2D models for a BNS signal and corresponding input samples at H1 detector. The component masses are $1.18M_{\odot}$ and $1.15M_{\odot}$. The red dashed line shows the time of coalescence. The bottom plot shows the values of integrated gradients summed over all frequencies for each time bin.

IV. CONCLUSIONS

In this study, we explored the application of three distinct models: a 1D CNN, a 2D CNN, and an ensemble model for detecting and classifying GWs from CBC sources. The 1D model, trained on whitened time-series data, excelled in identifying BBH signals, while the 2D model, trained on Q-transformed spectrograms, showed superior performance with NSBH and BNS signals. Overall, the ensemble model demonstrated the most robust classification capability across all signal types.

The effectiveness of combining 1D and 2D models was further reinforced through feature map visualization using the t-SNE technique and attribution map analysis via the integrated gradients method. We observed that the 1D model tends to focus on data preceding the merger time, especially as signal duration increases. In contrast, the 2D model scrutinizes the entire chirp waveform, capturing the intricacies of GW signals more comprehensively. These differences in focus and performance between the models highlight the benefits of their integration.

While our study presents results based on a specific selection of architectures for 1D and 2D CNNs, it is

important to recognize that the field offers a wide variety of CNN architectures. Future research exploring additional architectures may provide a more comprehensive understanding and validation of the conclusions drawn in this study.

We discussed the performance of the models by fixing a FAP at 0.001, but given that each data sample is 4 s long, this would result in roughly one false positive every hour, making our method insufficient for real application. Efficiency could be enhanced by incorporating a subsequent model, such as a binary classifier differentiating BBH signals from noise, to further reduce the FAP. Validation is also required to address unbalanced data, considering the source population.

Our classification models hold potential for analyzing long continuous data through a sliding window approach. Although our models were trained to identify GW signals occurring between 3.8 and 3.9 sec within four-second segments, sliding input window with a step size, for example, of 0.1 seconds, allows us to detect signals at any time point in principle. However, this approach may lead to encountering multiple triggers within a single event, requiring further tuning, which will be addressed in future studies.

ACKNOWLEDGMENTS

The authors are grateful to Koustav Chandra for useful comments. This research was supported in part by the Japan Society for the Promotion of Science (JSPS) Grant-in-Aid for Scientific Research [Grants No. 22H01228

(K. Somiya), and No. 19H01901, No. 23H01176, and No. 23H04520 (H. Takahashi)]. This research was also supported by the Joint Research Program of the Institute for Cosmic Ray Research, University of Tokyo and Tokyo City University Prioritized Studies. This research has made use of data or software obtained from the Gravitational Wave Open Science Center [76], a service of the LIGO Scientific Collaboration, the Virgo Collaboration, and KAGRA. This material is based upon work supported by NSF's LIGO Laboratory which is a major facility fully funded by the National Science Foundation, as well as the Science and Technology Facilities Council (STFC) of the United Kingdom, the Max-Planck-Society (MPS), and the State of Niedersachsen/Germany for support of the construction of Advanced LIGO and construction and operation of the GEO600 detector. Additional support for Advanced LIGO was provided by the Australian Research Council. Virgo is funded, through the European Gravitational Observatory (EGO), by the French Centre National de Recherche Scientifique (CNRS), the Italian Istituto Nazionale di Fisica Nucleare (INFN), and the Dutch Nikhef, with contributions by institutions from Belgium, Germany, Greece, Hungary, Ireland, Japan, Monaco, Poland, Portugal, Spain. K. A. G. R. A. is supported by Ministry of Education, Culture, Sports, Science and Technology (MEXT), Japan Society for the Promotion of Science (JSPS) in Japan; National Research Foundation (NRF) and Ministry of Science and ICT (MSIT) in Korea; Academia Sinica (AS) and National Science and Technology Council (NSTC) in Taiwan.

-
- [1] J. Aasi *et al.* (LIGO Scientific Collaboration), Advanced LIGO, *Classical Quantum Gravity* **32**, 074001 (2015).
- [2] B. P. Abbott *et al.* (LIGO Scientific Collaboration and Virgo Collaboration), Observation of gravitational waves from a binary black hole merger, *Phys. Rev. Lett.* **116**, 061102 (2016).
- [3] F. Acernese *et al.*, Advanced Virgo: A second-generation interferometric gravitational wave detector, *Classical Quantum Gravity* **32**, 024001 (2014).
- [4] B. P. Abbott *et al.*, Multi-messenger observations of a binary neutron star merger, *Astrophys. J. Lett.* **848**, L12 (2017).
- [5] B. P. Abbott *et al.* (LIGO Scientific Collaboration and Virgo Collaboration), GWTC-1: A gravitational-wave transient catalog of compact binary mergers observed by LIGO and Virgo during the first and second observing runs, *Phys. Rev. X* **9**, 031040 (2019).
- [6] R. Abbott *et al.* (LIGO Scientific Collaboration and Virgo Collaboration), GWTC-2: Compact binary coalescences observed by LIGO and Virgo during the first half of the third observing run, *Phys. Rev. X* **11**, 021053 (2021).
- [7] R. Abbott *et al.* (LIGO Scientific Collaboration and Virgo Collaboration), GWTC-2.1: Deep extended catalog of compact binary coalescences observed by LIGO and Virgo during the first half of the third observing run, *Phys. Rev. D* **109**, 022001 (2024).
- [8] R. Abbott *et al.* (LIGO Scientific Collaboration, Virgo Collaboration, and KAGRA Collaboration), GWTC-3: Compact binary coalescences observed by LIGO and Virgo during the second part of the third observing run, *Phys. Rev. X* **13**, 041039 (2023).
- [9] R. Abbott, T. D. Abbott, S. Abraham, F. Acernese, K. Ackley, A. Adams, C. Adams, R. Adhikari, V. Adya, C. Affeldt *et al.*, Observation of gravitational waves from two neutron star–black hole coalescences, *Astrophys. J. Lett.* **915**, L5 (2021).
- [10] B. P. Abbott *et al.* (LIGO Scientific Collaboration and Virgo Collaboration), GW170817: Observation of gravitational

- waves from a binary neutron star inspiral, *Phys. Rev. Lett.* **119**, 161101 (2017).
- [11] B. Abbott, R. Abbott, T. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, R. Adhikari, V. Adya, C. Affeldt *et al.*, GW190425: Observation of a compact binary coalescence with total mass $\sim 3.4M_{\odot}$, *Astrophys. J.* **892**, L3 (2020).
- [12] T. Akutsu *et al.*, KAGRA: 2.5 generation interferometric gravitational wave detector, *Nat. Astron.* **3**, 35 (2019).
- [13] B. Allen, W. G. Anderson, P. R. Brady, D. A. Brown, and J. D. E. Creighton, FINDCHIRP: An algorithm for detection of gravitational waves from inspiraling compact binaries, *Phys. Rev. D* **85**, 122006 (2012).
- [14] J. D. E. Creighton and W. G. Anderson, *Gravitational-Wave Physics and Astronomy: An Introduction to Theory, Experiment and Data Analysis* (Wiley-VCH, New York, 2011).
- [15] M. Maggiore, *Gravitational Waves: Volume 1: Theory and Experiments*, Gravitational Waves (Oxford University Press, Oxford, 2008).
- [16] H. Gabbard, C. Messenger, I. S. Heng, F. Tonolini, and R. Murray-Smith, Bayesian parameter estimation using conditional variational autoencoders for gravitational-wave astronomy, *Nat. Phys.* **18**, 112 (2021).
- [17] M. Dax, S. R. Green, J. Gair, M. Pürrer, J. Wildberger, J. H. Macke, A. Buonanno, and B. Schölkopf, Neural importance sampling for rapid and reliable gravitational-wave inference, *Phys. Rev. Lett.* **130**, 171403 (2023).
- [18] C. Chatterjee, L. Wen, K. Vinsen, M. Kovalam, and A. Datta, Using deep learning to localize gravitational wave sources, *Phys. Rev. D* **100**, 103025 (2019).
- [19] S. Sasaoka, Y. Hou, K. Somiya, and H. Takahashi, Localization of gravitational waves using machine learning, *Phys. Rev. D* **105**, 103030 (2022).
- [20] A. Kolmus, G. Baltus, J. Janquart, T. van Laarhoven, S. Caudill, and T. Heskes, Fast sky localization of gravitational waves using deep learning seeded importance sampling, *Phys. Rev. D* **106**, 023032 (2022).
- [21] M. Zevin, S. Coughlin, S. Bahaadini, E. Besler, N. Rohani, S. Allen, M. Cabero, K. Crowston, A. K. Katsaggelos, S. L. Larson, T. K. Lee, C. Lintott, T. B. Littenberg, A. Lundgren, C. Østerlund, J. R. Smith, L. Trouille, and V. Kalogera, Gravity Spy: Integrating advanced LIGO detector characterization, machine learning, and citizen science, *Classical Quantum Gravity* **34**, 064003 (2017).
- [22] S. Bahaadini, V. Noroozi, N. Rohani, S. Coughlin, M. Zevin, J. Smith, V. Kalogera, and A. Katsaggelos, Machine learning for Gravity Spy: Glitch classification and dataset, *Inf. Sci.* **444**, 172 (2018).
- [23] Y. Sakai, Y. Itoh, P. Jung, K. Kokeyama, C. Kozakai, K. T. Nakahira, S. Oshino, Y. Shikano, H. Takahashi, T. Uchiyama, G. Ueshima, T. Washimi, T. Yamamoto, and T. Yokozawa, Unsupervised learning architecture for classifying the transient noise of interferometric gravitational-wave detectors, *Sci. Rep.* **12**, 9935 (2022).
- [24] D. George and E. A. Huerta, Deep neural networks to enable real-time multimessenger astrophysics, *Phys. Rev. D* **97**, 044039 (2018).
- [25] H. Gabbard, M. Williams, F. Hayes, and C. Messenger, Matching matched filtering with deep networks for gravitational-wave astronomy, *Phys. Rev. Lett.* **120**, 141103 (2018).
- [26] D. George and E. Huerta, Deep learning for real-time gravitational wave detection and parameter estimation: Results with Advanced LIGO data, *Phys. Lett. B* **778**, 64 (2018).
- [27] X. Fan, J. Li, X. Li, Y. Zhong, and J. Cao, Applying deep neural networks to the detection and space parameter estimation of compact binary coalescence with a network of gravitational wave detectors, *Sci. China Phys. Mech. Astron.* **62**, 969512 (2019).
- [28] T. D. Gebhard, N. Kilbertus, I. Harry, and B. Schölkopf, Convolutional neural networks: A magic bullet for gravitational-wave detection?, *Phys. Rev. D* **100**, 063015 (2019).
- [29] H. Wang, S. Wu, Z. Cao, X. Liu, and J.-Y. Zhu, Gravitational-wave signal recognition of LIGO data by deep learning, *Phys. Rev. D* **101**, 104003 (2020).
- [30] W. Wei, A. Khan, E. Huerta, X. Huang, and M. Tian, Deep learning ensemble for real-time gravitational wave detection of spinning binary black hole mergers, *Phys. Lett. B* **812**, 136029 (2021).
- [31] H. Xia, L. Shao, J. Zhao, and Z. Cao, Improved deep learning techniques in gravitational-wave data analysis, *Phys. Rev. D* **103**, 024040 (2021).
- [32] M. D. Morales, J. M. Antelis, C. Moreno, and A. I. Nesterov, Deep learning for gravitational-wave data analysis: A resampling white-box approach, *Sensors* **21**, 3174 (2021).
- [33] N. Lopac, F. Hrzić, I. P. Vukšanović, and J. Lerga, Detection of non-stationary GW signals in high noise from Cohen's class of time-frequency representations using deep learning, *IEEE Access* **10**, 2408 (2022).
- [34] M. B. Schäfer, O. c. v. Zelenka, A. H. Nitz, F. Ohme, and B. Brügmann, Training strategies for deep learning gravitational-wave searches, *Phys. Rev. D* **105**, 043002 (2022).
- [35] M. B. Schäfer and A. H. Nitz, From one to many: A deep learning coincident gravitational-wave search, *Phys. Rev. D* **105**, 043003 (2022).
- [36] C. Ma, W. Wang, H. Wang, and Z. Cao, Ensemble of deep convolutional neural networks for real-time gravitational wave signal recognition, *Phys. Rev. D* **105**, 083013 (2022).
- [37] M. B. Schäfer, O. c. v. Zelenka, A. H. Nitz, H. Wang, S. Wu, Z.-K. Guo, Z. Cao, Z. Ren, P. Nousi, N. Stergioulas, P. Iosif, A. E. Koloniari, A. Tefas, N. Passalis, F. Salemi, G. Vedovato, S. Klimenko, T. Mishra, B. Brügmann, E. Cuoco, E. A. Huerta, C. Messenger, and F. Ohme, First machine learning gravitational-wave search mock data challenge, *Phys. Rev. D* **107**, 023021 (2023).
- [38] W.-H. Ruan, H. Wang, C. Liu, and Z.-K. Guo, Rapid search for massive black hole binary coalescences using deep learning, *Phys. Lett. B* **841**, 137904 (2023).
- [39] P. Nousi, A. E. Koloniari, N. Passalis, P. Iosif, N. Stergioulas, and A. Tefas, Deep residual networks for gravitational wave detection, *Phys. Rev. D* **108**, 024022 (2023).
- [40] C. Murali and D. Lumley, Detecting and denoising gravitational wave signals from binary black holes using deep learning, *Phys. Rev. D* **108**, 043024 (2023).

- [41] P. G. Krastev, Real-time detection of gravitational waves from binary neutron stars using artificial neural networks, *Phys. Lett. B* **803**, 135330 (2020).
- [42] M. B. Schäfer, F. Ohme, and A. H. Nitz, Detection of gravitational-wave signals from binary neutron star mergers using machine learning, *Phys. Rev. D* **102**, 063015 (2020).
- [43] Y.-C. Lin and J.-H. P. Wu, Detection of gravitational waves using Bayesian neural networks, *Phys. Rev. D* **103**, 063034 (2021).
- [44] P. G. Krastev, K. Gill, V. A. Villar, and E. Berger, Detection and parameter estimation of gravitational waves from binary neutron-star mergers in real LIGO data using deep learning, *Phys. Lett. B* **815**, 136161 (2021).
- [45] W. Wei and E. Huerta, Deep learning for gravitational wave forecasting of neutron star mergers, *Phys. Lett. B* **816**, 136185 (2021).
- [46] G. Baltus, J. Janquart, M. Lopez, A. Reza, S. Caudill, and J.-R. Cudell, Convolutional neural networks for the detection of the early inspiral of a gravitational-wave signal, *Phys. Rev. D* **103**, 102003 (2021).
- [47] G. Baltus, J. Janquart, M. Lopez, H. Narola, and J.-R. Cudell, Convolutional neural network for gravitational-wave early alert: Going down in frequency, *Phys. Rev. D* **106**, 042002 (2022).
- [48] J. a. Aveiro, F. F. Freitas, M. Ferreira, A. Onofre, C. m. c. Providência, G. m. c. Gonçalves, and J. A. Font, Identification of binary neutron star mergers in gravitational-wave data using object-detection machine learning models, *Phys. Rev. D* **106**, 084059 (2022).
- [49] A. Menéndez-Vázquez, M. Kolstein, M. Martínez, and L. M. Mir, Searches for compact binary coalescence events using neural networks in the LIGO/Virgo second observation period, *Phys. Rev. D* **103**, 062004 (2021).
- [50] H. Yu, R. X. Adhikari, R. Magee, S. Sachdev, and Y. Chen, Early warning of coalescing neutron-star and neutron-star-black-hole binaries from the nonstationary noise background using neural networks, *Phys. Rev. D* **104**, 062004 (2021).
- [51] W. Wei, E. A. Huerta, M. Yun, N. Loutrel, M. A. Shaikh, P. Kumar, R. Haas, and V. Kindratenko, Deep learning with quantized neural networks for gravitational-wave forecasting of eccentric compact binary coalescence, *Astrophys. J.* **919**, 82 (2021).
- [52] R. Qiu, P. G. Krastev, K. Gill, and E. Berger, Deep learning detection and classification of gravitational waves from neutron star-black hole mergers, *Phys. Lett. B* **840**, 137850 (2023).
- [53] S. Garg, S. Sasaoka, D. S. Dominguez, Y. Hou, N. Koyama, K. Somiya, H. Takahashi, and M. Ohashi, Comparison of training methods for convolutional neural network model for gravitational-wave detection from neutron star-black hole binaries, in *Proceedings of 38th International Cosmic Ray Conference—PoS(ICRC2023)* (Sissa Medialab, Nagoya, Japan, 2023).
- [54] A. Iess, E. Cuoco, F. Morawski, C. Nicolaou, and O. Lahav, LSTM and CNN application for core-collapse supernova search in gravitational wave real data, *Astron. Astrophys.* **669**, A42 (2023).
- [55] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, Learning deep features for discriminative localization, in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE Computer Society, Los Alamitos, CA, USA, 2016), pp. 2921–2929.
- [56] S. Sasaoka, N. Koyama, D. Dominguez, Y. Sakai, K. Somiya, Y. Omae, and H. Takahashi, Visualizing convolutional neural network for classifying gravitational waves from core-collapse supernovae, *Phys. Rev. D* **108**, 123033 (2023).
- [57] M. Sundararajan, A. Taly, and Q. Yan, Axiomatic attribution for deep networks, in *Proceedings of the 34th International Conference on Machine Learning*, Proceedings of Machine Learning Research Vol. 70 (PMLR, Sydney, NSW, Australia, 2017), pp. 3319–3328, <https://proceedings.mlr.press/v70/sundararajan17a.html>.
- [58] R. Abbott *et al.* (The LIGO Scientific Collaboration, the Virgo Collaboration, and the KAGRA Collaboration), Open data from the third observing run of LIGO, Virgo, KAGRA, and GEO, *Astrophys. J. Suppl. Ser.* **267**, 29 (2023).
- [59] LIGO Scientific Collaboration, LIGO Algorithm Library—LALSuite, free software (GPL), [10.7935/GT1W-FZ16](https://www.ligo.org/lscsoft) (2018).
- [60] A. Bohé, L. Shao, A. Taracchini, A. Buonanno, S. Babak, I. W. Harry, I. Hinder, S. Ossokine, M. Pürrer, V. Raymond, T. Chu, H. Fong, P. Kumar, H. P. Pfeiffer, M. Boyle, D. A. Hemberger, L. E. Kidder, G. Lovelace, M. A. Scheel, and B. Szilágyi, Improved effective-one-body model of spinning, nonprecessing binary black holes for the era of gravitational-wave astrophysics with advanced detectors, *Phys. Rev. D* **95**, 044028 (2017).
- [61] R. Sturani, S. Fischetti, L. Cadonati, G. M. Guidi, J. Healy, D. Shoemaker, and A. Viceré, Complete phenomenological gravitational waveforms from spinning coalescing binaries, *J. Phys. Conf. Ser.* **243**, 012007 (2010).
- [62] A. Nitz *et al.*, gwastro/pycbc: v2.2.2 release of pycbc (2023), [10.5281/ZENODO.8340277](https://doi.org/10.5281/ZENODO.8340277).
- [63] P. Welch, The use of fast fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms, *IEEE Trans. Audio Electroacoust.* **15**, 70 (1967).
- [64] S. Chatterji, L. Blackburn, G. Martin, and E. Katsavounidis, Multiresolution techniques for the detection of gravitational-wave bursts, *Classical Quantum Gravity* **21**, S1809 (2004).
- [65] S. K. Chatterji, The search for gravitational wave bursts in data from the second LIGO science run, Ph.D. thesis, Massachusetts Institute of Technology, 2005.
- [66] K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Las Vegas, NV, USA, 2016), pp. 770–778.
- [67] N. Passalis, A. Tefas, J. Kannianen, M. Gabbouj, and A. Iosifidis, Deep adaptive input normalization for time series forecasting, *IEEE Trans. Neural Networks Learn. Syst.* **31**, 3760 (2020).
- [68] A. L. Maas, A. Y. Hannun, A. Y. Ng *et al.*, Rectifier nonlinearities improve neural network acoustic models, in *Proceedings of the ICML Workshop on Deep Learning for Audio, Speech and Language Processing* (ICML, Atlanta, GA, USA, 2013).

- [69] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors, [arXiv:1207.0580](https://arxiv.org/abs/1207.0580).
- [70] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, PyTorch: An imperative style, high-performance deep learning library, in *Advances in Neural Information Processing Systems 32* (Curran Associates, Vancouver, Canada, 2019), pp. 8024–8035.
- [71] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, [arXiv:1412.6980](https://arxiv.org/abs/1412.6980).
- [72] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, Curriculum learning, in *Proceedings of the 26th Annual International Conference on Machine Learning, ICML '09* (Association for Computing Machinery, New York, NY, USA, 2009), pp. 41–48.
- [73] L. van der Maaten and G. Hinton, Visualizing data using t-SNE, *J. Mach. Learn. Res.* **9**, 2579 (2008), <https://jmlr.org/papers/v9/vandermaaten08a.html>.
- [74] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, Grad-CAM: Visual explanations from deep networks via gradient-based localization, in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)* (IEEE, Venice, Italy, 2017), pp. 618–626.
- [75] N. Kokhlikyan, V. Miglani, M. Martin, E. Wang, B. Alsallakh, J. Reynolds, A. Melnikov, N. Kliushkina, C. Araya, S. Yan, and O. Reblitz-Richardson, Captum: A unified and generic model interpretability library for PyTorch, [arXiv:2009.07896](https://arxiv.org/abs/2009.07896).
- [76] <https://gwosc.org>.