

Boosting the efficiency of parametric detection with hierarchical neural networks

Jingkai Yan^{1,4}, Robert Colgan^{2,4}, John Wright^{1,4}, Zsuzsa Márka^{4,5}, Imre Bartos⁶, and Szabolcs Márka^{3,4}

¹*Department of Electrical Engineering, Columbia University in the City of New York,
500 West 120th Street, New York, New York 10027, USA*

²*Department of Computer Science, Columbia University in the City of New York,
500 West 120th Street, New York, New York 10027, USA*

³*Department of Physics, Columbia University in the City of New York,
538 West 120th Street, New York, New York 10027, USA*

⁴*Data Science Institute, Columbia University in the City of New York,
550 West 120th Street, New York, New York 10027, USA*

⁵*Columbia Astrophysics Laboratory, Columbia University in the City of New York,
538 West 120th Street, New York, New York 10027, USA*

⁶*Department of Physics, University of Florida, P.O. Box 118440, Gainesville, Florida 32611-8440, USA*



(Received 8 August 2022; accepted 29 August 2022; published 12 September 2022)

Gravitational-wave astronomy is a vibrant field that leverages both classic and modern data processing techniques for the understanding of the Universe. Various approaches have been proposed for improving the efficiency of the detection scheme, with hierarchical matched filtering being an important strategy. Meanwhile, deep learning methods have recently demonstrated both consistency with matched filtering methods and remarkable statistical performance. In this work, we propose a hierarchical detection network (HDN), a novel approach to efficient detection that combines ideas from hierarchical matching and deep learning. The network is trained using a novel loss function, which encodes simultaneously the goals of statistical accuracy and efficiency. We discuss the source of complexity reduction of the proposed model and describe a general recipe for initialization with each layer specializing in different regions. We demonstrate the performance of the HDN with experiments using open LIGO data and synthetic injections, and observe with two-layer models a 79% efficiency gain compared with matched filtering at an equal error rate of 0.2%. Furthermore, we show how training a three-layer HDN initialized using a two-layer model can further boost both accuracy and efficiency, highlighting the power of multiple simple layers in efficient detection.

DOI: [10.1103/PhysRevD.106.063008](https://doi.org/10.1103/PhysRevD.106.063008)

I. INTRODUCTION

The study of gravitational-wave (GW) signals [1–7] is a vibrant field that constantly expands our understanding of gravitational phenomena and our Universe. In the detection schemes currently employed by gravitational-wave detectors [8–10], such as KAGRA [11], GEO600 [12], Virgo [13], and LIGO [14,15], the core algorithmic methods consist of excess-energy-based burst searches [16–22] and matched filtering searches [23–42]. Matched filtering (MF) is a classic signal processing technique, which computes the correlation of the time-delayed input signal with a bank of templates. While the matched filtering method has outstanding performance and statistical rigor, it potentially has room for improvement in terms of computational complexity [43]. In fact, millions of templates are being used in the LIGO matched filtering pipeline, with the number still expanding with the scope of the search [5].

As we deepen our search for gravitational-wave signals, the issue of computational efficiency (namely, the number

of basic operations required by a computer) is becoming increasingly prominent. Detection methods that excel in both statistical performance and computational efficiency can significantly boost our capacities for exploring wider and higher-dimensional parameter spaces, and even other families of eccentric waveforms [44]. This in turn will help with uncovering more astrophysical events, potentially unveiling novel astrophysical phenomena, as well as reducing the carbon footprint associated with searching for these events.

In the literature, a promising approach to reducing the complexity of matched filtering searches has been to apply a two-step hierarchical search, which seeks to rapidly reject most negative samples [45]. Later, [46] expanded the hierarchy to involve a temporal multiscale approach. Some other meritorious extensions include using geometric template placing [47] and hierarchy based on chirp times [48]. The work of [49] applies two-step detection within the PYCBC framework, and compares the performance on

simulated data. A recent work of [50] further combines the two-step method with dimensionality reduction in the template space using principal component analysis. The above examples demonstrate improvements relative to basic matched filtering in various settings for GW detection.

Similar ideas have also been widely explored and applied in machine learning contexts. For example, [51,52] consider hierarchical matching of image features in both the spatial domain and feature domain for image classification. Reference [53] considers image classification using hierarchical matching in the spatial domain. In natural language processing, hierarchical models have also been used in sentiment classification [54]. More specific applications of this idea include medical imaging [55], human detection and segmentation [56], and crime classification [57].

In the meantime, with the growing literature on applying deep learning and neural networks to GW detection, it is tempting to leverage deep learning's power to reduce complexity. Indeed, various neural network architectures have been shown to perform tasks such as GW detection, parameter estimation, noise transients identification, and data denoising [58–93] at performance levels comparable to that of matched filtering. Furthermore, it has been shown that matched filtering is generally suboptimal for parametric signal detection [43,94], and the performance can be improved by optimizing the templates using deep learning techniques [43]. This can be achieved by setting up a neural network that is formally equivalent to matched filtering, and then training on data. Inspired by the flexibility of deep learning models, it is conceptually appealing to explicitly incorporate computational efficiency into the neural network objectives, aiming to achieve “the best of both worlds.”

In this work, we propose a novel neural network architecture named the hierarchical detection network (HDN), which takes the form of a multilayer matched filtering with trainable parameters. In order to achieve the dual goal of accuracy and efficiency, we construct a novel loss function that explicitly incorporates computational complexity. We demonstrate the efficiency gains on data with open LIGO noise data and synthetic GW signal injections. As a quick glance at the performance gains, when tested on data with synthetically injected signals at signal-to-noise ratio (SNR) 9, compared with matched filtering, a two-layer HDN can achieve false positive and false negative rates of 0.2% with 79% lower complexity, and reduce error rates by 88% when at equal complexity equivalent to 100 templates, for instance. Experimental details are described in Sec. V.

Yet, the two-layer networks do not reveal the full power of the proposed model. We further show that by training a three-layer model with careful initialization, it is capable of achieving even better accuracy at lower complexity. We also provide some intuitive insights into the mechanism behind multilayer hierarchical models and their construction.

The rest of the paper is organized as follows. Section II reviews the problem of parametric detection and some relevant models. Section III introduces hierarchical detection networks, including the setup, complexity, and training process. Section IV further discusses the complexity reduction from HDN. Section V presents the experimental results of applying HDN on real LIGO data and synthetic injections. We discuss some further implications and future steps of this work in Sec. VI.

II. PRELIMINARIES

In this section, we describe the problem setup for a single gravitational-wave detector and review some related detection algorithms, including matched filtering and a closely related model MNet [43].

A. Parametric detection and matched filtering

Consider the problem of detecting gravitational waves in a single gravitational-wave detector data stream. Formally, assume we observe detector data $\mathbf{x} \in \mathbb{R}^n$ and need to decide whether or not $\mathbf{x}$ contains any gravitational-wave signal in addition to noise. Throughout this paper, we model the inputs as having fixed dimension, where time-domain convolution can be applied for generic time-series inputs. The signals of interest are modeled as belonging to a parametric set

$$S_\Gamma = \{s_\gamma | \gamma \in \Gamma\}, \quad (1)$$

where the parameters γ can represent properties of the objects that generate the gravitational wave, such as masses, orbits, and spins. We model noise as a random vector $\mathbf{z} \in \mathbb{R}^n$ with some density ρ_0 and independent of the signal. The detection problem can thus be written as the following hypothesis test:

$$H_0: \mathbf{x} = \mathbf{z} \quad (2)$$

$$\text{or } H_1: \mathbf{x} = s_\gamma + \mathbf{z} \quad \text{for some } \gamma \in \Gamma. \quad (3)$$

For this testing problem, we need to design decision rules $\delta: \mathbb{R}^n \rightarrow \{0, 1\}$ that ideally achieve both good statistical performance and high computational efficiency.

One natural approach to this problem is matched filtering [95–97], a classical method for signal detection. A matched filter is the optimal linear filter for maximizing the SNR in stationary Gaussian white noise [95]. When detecting a single target signal $\mathbf{s}$, matched filtering takes the following form of an inner product test:

$$\delta(\mathbf{x}) = 1 \text{ iff } \langle \mathbf{s}, \mathbf{x} \rangle \geq \tau, \quad (4)$$

where τ is a cutoff threshold. Its natural extension to detecting a parametric family of signals is to take the maximum over template samples $s_{\gamma_1}, \dots, s_{\gamma_k}$ in the signal space, namely,

$$\delta(\mathbf{x}) = 1 \text{ iff } \max_{i=1,\dots,k} \langle \mathbf{s}_{\gamma_i}, \mathbf{x} \rangle \geq \tau. \quad (5)$$

As noted in the literature [43], matched filtering is sub-optimal for the generic parametric detection problem in terms of statistical performance. To address such suboptimality, one alternative approach that has been described is the MNet architectures.

B. MNet architectures

The MNet architectures, including MNet-Deep and MNet-Shallow, are two machine learning models for parametric detection based on neural networks [43]. Formally, an MNet is a neural network initialized to exactly replicate matched filtering and then trained on data for improved performance. Such an improvement is mainly due to its structural capability of handling nonconvex decision boundaries and non-Gaussian noise distributions. The shallow and deep versions differ in how the replication of matched filtering is constructed. MNet-Shallow is structured identically to matched filtering, allowing for a more direct comparison, whereas MNet-Deep uses a multilayer pairwise-max structure for higher flexibility and statistical performance.

In this work, with computational complexity in mind, we will focus on the MNet-Shallow model, which can be expressed as

$$f_{\text{MNet-Shallow}}(\mathbf{x}) = \max_i \langle \mathbf{s}_i, \mathbf{x} \rangle. \quad (6)$$

Here the weights $\mathbf{s}_i$ are initialized as templates and then trained over data, making it advantageous over classic matched filtering. If we compare its computational efficiency against matched filtering (measured in terms of the number of operations required to achieve a target error rate), the strict performance improvement with identical architecture suggests that one can expect a strict efficiency improvement as well. However, the structural similarity between MNet-Shallow and matched filtering implies that such efficiency gains may typically be very limited. In order to achieve efficiency gains on higher orders of magnitude, we may need to reconsider the parametric detection problem and innovate on the basic matched filtering rule. As we present in the next section, one solution is to arrange the templates in a multilayer hierarchy, so that significant proportions of negative-labeled data are subject to early rejections.

III. HIERARCHICAL DETECTION NETWORKS

In this section, we present the architecture of Hierarchical Detection Network (HDN). This architecture combines the strengths of both matched filtering and MNet-Shallow, to simultaneously maximize statistical performance and computational efficiency.

The main idea behind HDN is intuitive. If an input segment clearly contains no gravitational-wave signals, we may not need to subject it to millions of templates to

tell that. A small number of “gatekeeper” templates may be sufficient for confidently rejecting these “obviously wrong” instances. Once these inputs have been ruled out, we can apply a more refined test using possibly more templates and reject a larger portion of the input space. This procedure can be repeated, until in the very last step, we employ our full template bank for a full diagnosis of the remaining instances which all previous tests failed to reject. Since the overwhelming majority part of the gravitational-wave strain data contains noise only, most instances will likely be addressed by the initial simple layers of the model, saving the need for the full template bank. In addition, different layers of the HDN may be designed to specialize in different parts of the input space, such that the available parameter space of the potentially allowed waveforms is successively restricted as the hierarchical process progresses from layer to layer, allowing for further efficiency gains.

A. Architecture of HDN

We first formally define a HDN. Generally speaking, a HDN is a hierarchical template matching model trained as a neural network, as illustrated in Fig 1. Let L be the number of layers in the hierarchical structure, and let $\{\mathbf{s}_i\}_{i=1}^n$ be the entire set of n templates used by the model. For each layer $\ell = 1, \dots, L$, only the first n_ℓ of these templates are used in that layer, where $0 < n_1 < \dots < n_L = n$. Let the threshold associated with template i at layer ℓ be $t_{i,\ell}$, $i = 1, \dots, n_\ell$. Here we let layer l reuse all templates from the previous layer(s), but assign independent threshold values to the reused templates at different layers in order to reduce computation complexity.

Following the conventions of the machine learning literature on binary classification, we call an input $\mathbf{x}$ positive if it contains a GW signal, and negative if it only contains noise. For a given input $\mathbf{x}$, the model processes it using the following procedure.

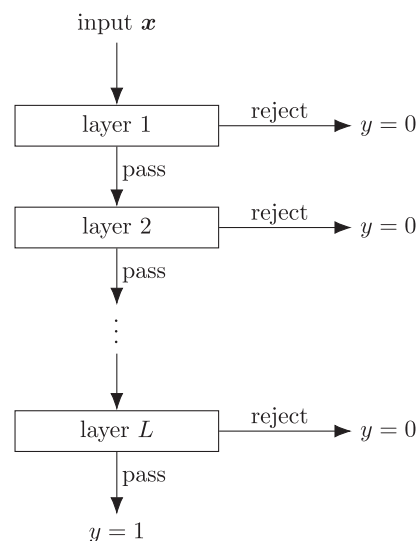


FIG. 1. Illustration of a hierarchical detection network.

Algorithm 1. The HDN algorithm.

Parameters: $L, n, \{n_\ell\}, \{s_i\}, \{t_{i,\ell}\}$
Input: $\mathbf{x}$

$\ell \leftarrow 1$
while $\ell \leq L$ **do**
 $y_\ell \leftarrow \max_{i \leq n_\ell} \langle \mathbf{x}, s_i \rangle - t_{i,\ell}$
 if $y_\ell < 0$ **then**
 return 0
 end if
 $\ell \leftarrow \ell + 1$
end while
return 1

More formally, for a given input $\mathbf{x}$, let

$$y_\ell = \max_{i \leq n_\ell} \langle \mathbf{x}, s_i \rangle - t_{i,\ell} \quad (7)$$

be the matching output at layer ℓ , $\ell = 1, \dots, L$. Let

$$I_\ell = \begin{cases} \mathbb{1}[y_\ell > 0], & \text{if } I_{\ell-1} = 1, \\ 0, & \text{if } I_{\ell-1} = 0 \end{cases} \quad (8)$$

be the indicator of whether the input passes layer ℓ of the model onto the later layer(s) $\ell = 1, \dots, L-1$, and define $I_L \equiv 0$. With these notations, the overall output of the model can be written as

$$y(\mathbf{x}) = \sum_{\ell=1}^L y_\ell (1 - I_\ell) \prod_{k=1}^{\ell-1} I_k. \quad (9)$$

Note that both matched filtering and MNet-Shallow can be unified under the framework of the HDN viewed as a model with a single layer. In the meantime, some of the existing two-step MF methods [45,47,49] can also be interpreted under this framework. Furthermore, the HDN architecture is not restricted to the typical two-layer hierarchy of coarse and fine searches, but can utilize multiple layers which specialize in different parts of the signal space. The use of multiple layers and their setup is further discussed in Sec. IV.

B. Measure of computational complexity

With matched filtering and a HDN unified under the same framework, we can provide a formal definition of computational complexity to facilitate our discussion. We are often most concerned about the execution efficiency of the model in deployment rather than in training, since it determines the real-time processing abilities. In the meantime, any computational cost of setting up the parameters of the model, including template selection for matched filtering and training for neural networks, is a one-time cost and can be conducted off-line. Therefore, it is natural to define complexity based on test time. Also, since for the vast

majority of time the input strain does not contain gravitational-wave events, we can capture the computational complexity solely by its performance on negative data. This leads to the following definition of complexity:

Definition 1 (Complexity). The (computational) complexity of a HDN model is defined as the expected number of template matching (inner product) operations conducted to evaluate a negative input.

Formally, we can write the complexity as

$$Z = \mathbb{E}_{\mathbf{x} \sim F_-} [z(\mathbf{x})], \quad (10)$$

where

$$z(\mathbf{x}) = \sum_{\ell=1}^L n_\ell (1 - I_\ell) \prod_{k=1}^{\ell-1} I_k \quad (11)$$

is the number of matching operations required for evaluating an input $\mathbf{x}$.

To illustrate this measure of complexity, note that for matched filtering and MNet-Shallow models, the complexity simply equals the number of templates used in the model. For a two-layer HDN, assuming only a proportion p of negative data enters the second layer, the complexity for the model will be $n_1 + p \cdot n_2$. Intuitively, if the initial layer contains fewer templates while being able to reject a significant portion of negative inputs, these inputs will not need to undergo the entire model, hence reducing the complexity of the model. This straightforward idea forms the basis of a HDN, upon which we further leverage the power of data through training for an additional boost in performance.

C. Training of a HDN

So far, we have described the behavior of a HDN at test/deployment time, and now we turn our attention to the training process. Conceptually, we want to set up a loss function as an appropriate combination of classification error and model complexity, so that minimizing the loss would achieve simultaneously accuracy and efficiency. However, a loss function directly based on the above expressions (9) and (11) is undesirable because of non-differentiability. Instead, we use soft surrogates for the indicators I_ℓ . Define

$$\hat{I}_\ell = \phi(y_\ell) \quad (12)$$

for $\ell = 1, \dots, L-1$ where $\phi(x) := \frac{1}{1+e^{-x}}$ is the sigmoid function serving as a soft surrogate of the step function. Also, let $\hat{I}_L \equiv 0$. Note that during training we can simply compute y_ℓ for all layers regardless of whether previous layers were passed, since this will only be a one-time off-line cost. Define the soft surrogates for $y(\mathbf{x})$ and $z(\mathbf{x})$ accordingly:

$$\hat{y}(\mathbf{x}) = \sum_{\ell=1}^L y_{\ell} (1 - \hat{t}_{\ell}) \prod_{k=1}^{\ell-1} \hat{t}_k, \quad (13)$$

$$\hat{z}(\mathbf{x}) = \sum_{\ell=1}^L n_{\ell} (1 - \hat{t}_{\ell}) \prod_{k=1}^{\ell-1} \hat{t}_k. \quad (14)$$

Assume the training dataset is $\{(\mathbf{x}_i, y_i^*)\}_{i=1}^N$, with N_+ positive entries and N_- negative entries. The loss function can be formulated as

$$\mathcal{L} = \frac{1}{N} \sum_i \ell_i^{\text{accu}} + \lambda \cdot \frac{1}{N_-} \sum_{i: y_i^*=0} \ell_i^{\text{cplx}}, \quad (15)$$

where

$$\ell_i^{\text{accu}} = y_i^* \log p_i + (1 - y_i^*) \log(1 - p_i) \quad (16)$$

with $p_i = \frac{1}{1 + e^{-\hat{y}_i}}$ equivalent to the cross-entropy loss for binary classification, and

$$\ell_i^{\text{cplx}} = \hat{z}_i \quad (17)$$

is a smooth approximation for the complexity of evaluating negative inputs. With the loss function (15) defined above, we can then train the model parameters $\{s_i\}$ and $\{t_{i,\ell}\}$ using first order optimization methods.

Experimental results of the HDN architecture will be shown in Sec. V.

IV. COMPLEXITY REDUCTION FROM MULTIPLE LAYERS

Here we provide some theoretical insights into why the hierarchical model achieves reduced complexity at similar target performance levels, particularly with more layers.

Consider as an example a two-layer hierarchical model with n_1, n_2 templates, respectively, on the layers. Let α_{ℓ} and β_{ℓ} denote, respectively, the false positive rate (FPR) and false negative rate (FNR) of layer ℓ conditioned on data that reach the corresponding layer. Recall that FPR denotes the proportion of negative samples that are falsely classified as positive, and the FNR the proportion of positive samples falsely classified as negative. The overall FPR, FNR, and the complexity z can then be represented as

$$\alpha_{\text{all}} = \alpha_1 \alpha_2, \quad (18)$$

$$\beta_{\text{all}} = \beta_1 + (1 - \beta_1) \beta_2, \quad (19)$$

$$z_{\text{all}} = n_1 + \alpha_1 (n_2 - n_1). \quad (20)$$

To understand why an improvement in complexity can be expected, we consider the following example setup of parametric detection as shown in Fig. 2. The probability

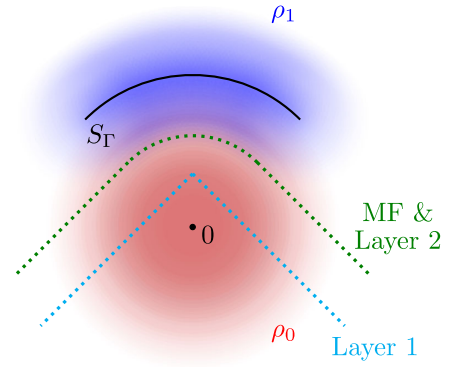


FIG. 2. An example of the complexity advantage of hierarchical detection models.

density of the two labeled classes are ρ_0 and ρ_1 , respectively. Note that the density for the negative class ρ_0 is precisely the noise density defined previously, and ρ_1 is the convolution of ρ_0 and the density of the signal s_T . Imagine a baseline MF model with a decision boundary as shown by the green curve, at the cost of n templates, where n has to be relatively large to approximate the smoothly curved boundary. Then we can construct the following hierarchical model to achieve a significantly lower complexity with identical statistical accuracy. To do this, we construct a simple two-layer hierarchical model, with the first layer decision boundary as shown by the dotted blue line and the second layer decision boundary coinciding with that of the MF model. Notice that the first layer features a very low complexity n_1 (with $n_1 = 2$ in this example), and in the meantime has a fairly high true negative rate $1 - \alpha$. Since the second layer reproduces the MF decision boundary, the overall decision rule of the hierarchical model is identical to that of the MF model, and hence, they share exactly the same ROC (receiver operating characteristic) curves. However, the complexity of the HDN model is $n_1 + \alpha_1(n_2 - n_1)$, which is significantly smaller than n_2 provided n_1 is small compared with n_2 , and α_1 is not too close to 1.

This example provides inspiration for a general recipe for designing hierarchical models with reduced complexity. For any decision rule given by a MF model, we can construct a sequence of preceding layers whose negative decision regions all lie inside the negative decision region of the MF, and finally, let the very last layer be equivalent to the original MF. The resulting hierarchical model will again have exactly the same overall decision rule and hence ROC curve, but with significantly reduced complexity. An illustrating example is shown in Fig. 3.

More generally, such constructions of hierarchical models can serve as good initializations for a HDN. One practical initialization scheme for an L -layer HDN is the following: First, train a separate $L - 1$ -layer model that

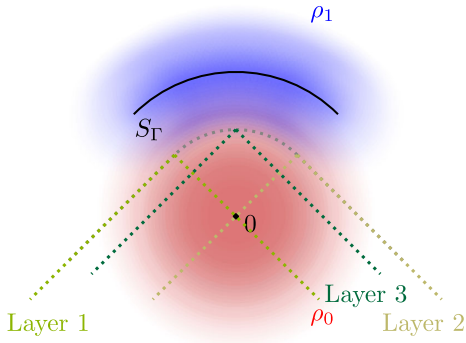


FIG. 3. A hierarchical model with more simple layers that lie inside the overall negative decision region.

only consists of the latter $L - 1$ layers of the desired model. Then we initialize the latter $L - 1$ layers of the original model with the trained network, and initialize the first layer with small $t_{i,1}$ values such that almost all inputs pass. This gives an initialization of the L -layer model which at initialization essentially replicates the $L - 1$ -layer model. From there, we train the initialized L -layer model on data, which will leverage the higher architectural capacity for further improved performance and complexity. An experiment that illustrates this approach is shown in Sec. V.

V. SIMULATION AND EXPERIMENTS

A. Data generation

In the experiments, we use open L1 strain data from LIGO Livingston's O2 run between August 1 and 25, 2017 with the ANALYSIS_READY flag [98]. The total duration of the frame files is 389.12 h. We downsample the strain data from the original 4096 to 2048 Hz for processing efficiency. The downsampled data are then divided into segments of 2 s, with each segment overlapping with 50% of its preceding segment.

To evaluate the accuracy of the detection models, we need both positive and negative datasets. For the negative datasets, the strain data themselves are used. For the positive datasets, due to the very limited number of confirmed detections of GW events, we generate positive data by injecting synthetic waveforms into the noise strains at a preset SNR value.

The entire L1 strain dataset is first divided into two sets to be used in training and testing, respectively, such that any segment in the training set does not overlap with any segment in the test set. For training and testing, respectively, a positive and a negative dataset are generated. For the positive datasets, synthetic waveforms are generated with masses m_1, m_2 uniformly drawn from [20,50] (times solar mass $M_\odot$) and three-dimensional spins drawn from an isotropic distribution and with spin dimensionless

magnitudes drawn from a uniform distribution within [0,1]. One waveform is injected to each 2-s segment of data, and the injected waveforms are aligned such that the peak is located at 0.95 s, and the injection amplitude is chosen such that the SNR after preprocessing is constant at 9. The preprocessing is applied to all data (after injection if applicable) by using a finite-impulse-response bandpass filter with cutoff frequency 30 and 400 Hz, whitening with power spectral density estimated from the L1 strain data, and truncating to only keep the center 1 s.

B. Two-layer networks

In this experiment, we limit our HDN models to two layers and $n_2 = 10n_1$. At initialization, the templates s_i are chosen as random gravitational waveforms from the same parameter space, and the thresholds $t_{i,\ell}$ are set to the same within each of the two layers. The parameter λ is the loss function and fixed at $\lambda = 10^{-4}$. For the optimization procedure of the network, we use the ADAM optimizer [99] which is common in modern deep learning, and a constant learning rate (i.e., scaling of the update at each iteration) of 10^{-4} .

Figure 4 shows a comparison of the complexity-performance trade-offs of the MF and HDN models, where the HDN models are two-layer architectures structured as described above. The horizontal axis plots the logarithm of the complexity measure defined in this paper, and the vertical axis plots the logarithm of the error rates at the point on the ROC curve where $\text{FPR} = \text{FNR}$. This choice of measure eliminates the arbitrariness of choosing the FNR at a fixed FPR level. For each architecture, ten independent runs are conducted, and the one with lowest accuracy measure is shown. The blue curve for the HDN is cut off

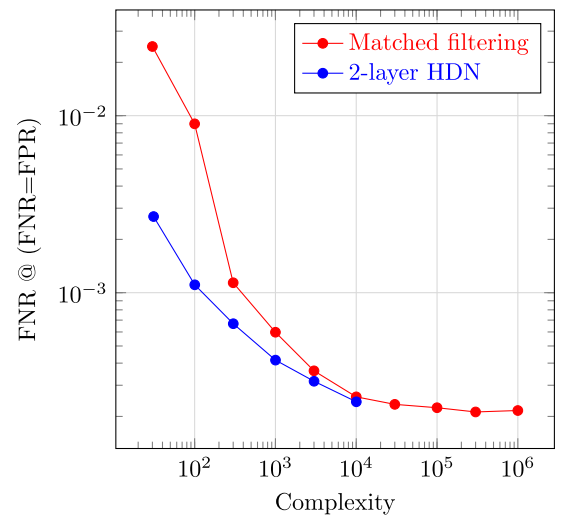


FIG. 4. Complexity-performance trade-off of matched filtering and the hierarchical neural network.

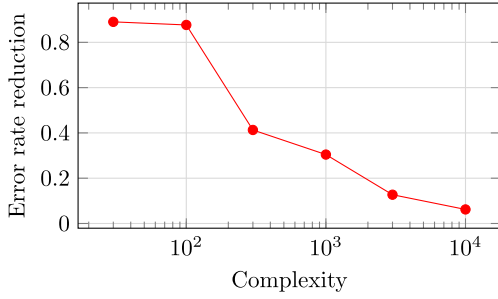


FIG. 5. Proportion of error rate reduced by using HDN over MF.

early due to memory limitations of training the model. Figure 5 shows the proportion of error rate reduced by the HDN as compared with MF at equal complexities. We see that the HDN consistently achieves a lower complexity than MF at equal accuracy, with higher advantages at lower complexities.

C. Three-layer networks

We further demonstrate the power of the proposed model with a deeper three-layer network. Conceptually, since adding more layers strictly improves model expressability, it should never hurt performance, provided that the parameters are initialized or trained appropriately.

In this experiment, we construct a three-layer HDN with layer sizes $(n_1, n_2, n_3) = (30, 100, 1000)$ in the following way. First, a shallower two-layer model with layer sizes $(100, 1000)$ is trained, and we use these trained parameters to initialize the latter two layers of the three-layer model. We then initialize the first layer of the three-layer model, setting the per-template thresholds $t_{i,1}$ as the same value for all i , such that all training data pass this first layer at initialization. This scheme ensures that at initialization, the three-layer model essentially replicates the performance of the trained two-layer model, giving it a head start before entering the training phase. The training is done in the same way as described before. Figure 6 illustrates the architecture of this three-layer model, along with the histograms of layer outputs of the test data that reach that layer, where the x axis denotes the layer output and the y axis denotes the frequency of occurrence. The histograms are divided into positive and negative according to the true class labels. Specifically, the densities of layer 1 involve all input data, and the densities of layers 2 and 3 involve only the data that pass the previous layers. We see that most of the negative data are successfully intercepted by the initial layers, with very few of them reaching the final layer, which corroborates our intuition.

Here, when evaluating the ROC curve, we adopt a slightly different approach that is more consistent with deep hierarchical models. Notice from Eq. (8) that the model uses a built-in threshold 0 to control the passing of

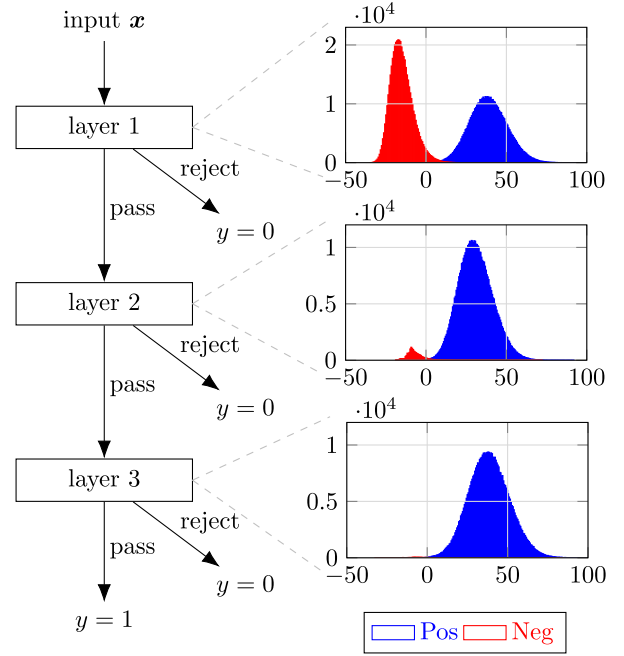


FIG. 6. Illustration of the three-layer architecture and the output densities on the test data from each layer. Only data entries that reach a given layer are shown. We see that each layer successfully rejects the vast majority of incoming negative data, and barely any negative data reach the last layer.

each layer. When generating the ROC curve using a varying threshold, such a threshold should be applied at all layers instead of only the last layer. Therefore, at test time only, we replace the threshold 0 in Eq. (8) with a variable threshold $t \in \mathbb{R}$ which is constant for all layers and compute the test outputs using (9) as before for each t value. Varying this threshold t produces the ROC curve. Also note that t determines which test entries would pass the layers; hence, it also affects the model complexity evaluated on the negative test dataset. In actual deployment, the threshold t should be fixed at some level that gives the desired trade-off between the FPR and FNR, so this is only for demonstration purposes.

Figure 7 shows the comparison of ROC curves between a matched filtering model with 100 templates, a two-layer HDN model with complexity 100.5 (used for initializing the three-layer model), and a three-layer HDN model with complexity 37.1 at the point of equal FPR and FNR. While the complexity of the three-layer model depends on the specific point chosen on the ROC curve, it does not exceed 65 for the entire segment of the ROC curve shown in the figure, and is thus always lower than the two-layer model. We see that the deeper three-layer model excels at both accuracy and efficiency compared with the two-layer model, and significantly more so if compared with the matched filtering model. This further showcases the power of depth in hierarchical models and corroborates our discussion in Sec. IV.

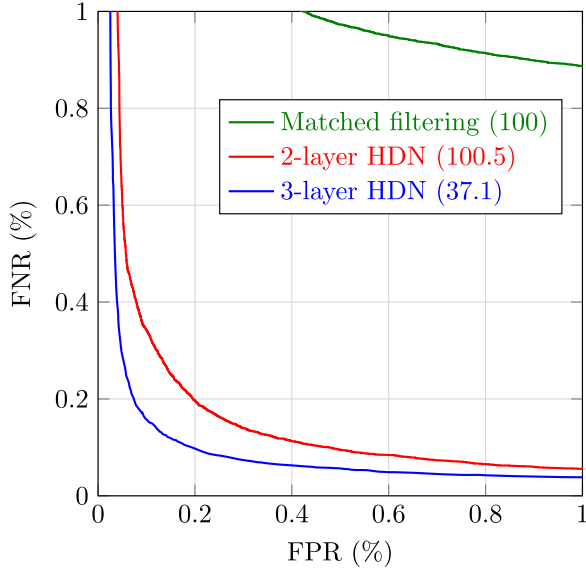


FIG. 7. Comparison of the ROC curves among the three models. The numbers in parentheses show the complexity of the model.

VI. DISCUSSION

In this paper, we showed that by leveraging the ideas from classical matched filtering and modern machine learning, we were able to design systems for GW detection that simultaneously optimize statistical accuracy and efficiency. This general conceptual idea of trainable hierarchical matched filtering can be applied to a wide range of existing proposals for efficient detection pipelines.

While the proposed HDN model conducts hierarchical rejection on the data, an alternative can be proposed to conduct hierarchical acceptance, namely, to progressively label parts of the data as positive rather than negative. This has the advantage of aligning better with the matched filtering routine, since it suffices to use one matching template to confirm a signal. In the specific problem of GW signal detection, due to the class imbalance from the scarcity of actual GW events, the majority of computational complexity hinges on the classification of negative data, and therefore, a hierarchical rejection model will have much more significant efficiency gains. In more general signal detection problems, hierarchical acceptance constructions can also be deployed in similar fashion as the HDN.

The proposed HDN can potentially have wider applications within the field of gravitational-wave science. For example, in the task of glitch detection and identification [80,100–104], one can combine existing constructions of machine-learning-based models with hierarchical models to improve both efficiency and accuracy.

One aspect to be further explored is how to select the number of layers in the hierarchy. While having more layers can boost model expressability and further leverage

the efficiency gains, excessive hierarchy may offer diminishing returns, and also make training increasingly difficult. In this paper, we demonstrated how a three-layer model excels over a two-layer one, and there may be a “sweet spot” number of layers for given signal detection setups. Another promising direction would be to incorporate prior knowledge about the signal domain such as low dimensionality and representative features into the detection model, which may be able to further outperform these current models agnostic of the signal space properties.

ACKNOWLEDGMENTS

This material is based upon work supported by NSF’s LIGO Laboratory which is a major facility fully funded by the National Science Foundation. We acknowledge computing resources from Columbia University’s Shared Research Computing Facility project, which is supported by NIH Research Facility Improvement Grant No. 1G20RR030893-01, and associated funds from the New York State Empire State Development, Division of Science Technology and Innovation Contract No. C090171, both awarded on April 15, 2010. The authors are grateful for the LIGO Scientific Collaboration for the careful review of the paper. This paper is assigned LIGO DCC No. LIGO-LIGO-P2200203. The authors acknowledge the LIGO Laboratory and Scientific Collaboration for the detectors, data, and the game-changing computing resources (National Science Foundation Grants No. PHY-0757058 and No. PHY-0823459). The authors would like to thank colleagues of the LIGO Scientific Collaboration and the Virgo Collaboration and Columbia University for their help and useful comments, in particular, the CBC group, Andrew Williamson, Stefan Countryman, William Tse, Nicolas Beltran, Asif Mallik, Sireesh Gururaja, and Thomas Dent which we hereby gratefully acknowledge. S. M. thanks David Spergel, Rainer Weiss, Rana Adhikari, and Kipp Canon for the motivating general discussions related to the role of machine learning and data analysis. This research has made use of data, software, and/or web tools obtained from the Gravitational Wave Open Science Center [105,106], a service of LIGO Laboratory, the LIGO Scientific Collaboration, and the Virgo Collaboration. This material is in part based upon work supported by the NSF’s LIGO Laboratory which is a major facility fully funded by the National Science Foundation, specifically some of the testing used LIGO detector noise publicly released by the LIGO Scientific Collaboration. LIGO Laboratory and Advanced LIGO are funded by the United States National Science Foundation as well as the Science and Technology Facilities Council of the United Kingdom, the Max-Planck-Society, 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, by the French Centre National de Recherche Scientifique, the Italian Istituto Nazionale di Fisica Nucleare, and the Dutch Nikhef, with contributions by institutions from Belgium, Germany, Greece, Hungary, Ireland, Japan, Monaco, Poland, Portugal, and Spain. The authors thank the University of Florida and Columbia University in the

City of New York for their generous support. The authors are grateful for the generous support of the National Science Foundation under Grant No. CCF-1740391. The authors thank Sharon Sputz of Columbia University for her effort in facilitating this collaboration. I.B. acknowledges the support of the National Science Foundation under Grant No. PHY-1911796 and the Alfred P. Sloan Foundation.

-
- [1] Albert Einstein, Näherungsweise integration der feldgleichungen der gravitation, *Sitz. K. Preuss. Akad. Wiss. Berlin*, **688** (1916).
 - [2] Albert Einstein, Über gravitationswellen, *Sitz. K. Preuss. Akad. Wiss. Berlin*, **154** (1918).
 - [3] LIGO and Virgo Collaborations, Observation of Gravitational Waves from a Binary Black Hole Merger, *Phys. Rev. Lett.* **116**, 061102 (2016).
 - [4] B.P. Abbott *et al.*, 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).
 - [5] 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).
 - [6] B.P. Abbott *et al.*, GW170817: Observation of Gravitational Waves from a Binary Neutron Star Inspiral, *Phys. Rev. Lett.* **119**, 161101 (2017).
 - [7] B.P. Abbott *et al.*, Multi-messenger observations of a binary neutron star merger, *Astrophys. J. Lett.* **848**, L12 (2017).
 - [8] C. Affeldt *et al.*, Advanced techniques in GEO 600, *Classical Quantum Gravity* **31**, 224002 (2014).
 - [9] F. Acernese *et al.*, Increasing the Astrophysical Reach of the Advanced Virgo Detector via the Application of Squeezed Vacuum States of Light, *Phys. Rev. Lett.* **123**, 231108 (2019).
 - [10] M. Tse *et al.*, Quantum-Enhanced Advanced LIGO Detectors in the Era of Gravitational-Wave Astronomy, *Phys. Rev. Lett.* **123**, 231107 (2019).
 - [11] T. Akutsu *et al.*, Overview of KAGRA: Detector design and construction history, *Prog. Theor. Exp. Phys.* **2021**, 05A101 (2021).
 - [12] K.L. Dooley *et al.*, GEO 600 and the GEO-HF upgrade program: Successes and challenges, *Classical Quantum Gravity* **33**, 075009 (2016).
 - [13] F. Acernese *et al.*, Advanced Virgo: A second-generation interferometric gravitational wave detector, *Classical Quantum Gravity* **32**, 024001 (2015).
 - [14] Alex Abramovici, William E. Althouse, Ronald W.P. Drever, Yekta Gursel, Seiji Kawamura, Frederick J. Raab, David Shoemaker, Lisa Sievers, Robert E. Spero, Kip S. Thorne, Rochus E. Vogt, Rainer Weiss, Stanley E. Whitcomb, and Michael E. Zucker, LIGO: The laser interferometer gravitational-wave observatory, *Science* **256**, 325 (1992).
 - [15] B.P. Abbott *et al.*, Advanced LIGO, *Classical Quantum Gravity* **32**, 074001 (2015).
 - [16] Warren G. Anderson, Patrick R. Brady, Jolien D.E. Creighton, and Éanna É. Flanagan, A power filter for the detection of burst sources of gravitational radiation in interferometric detectors, *Int. J. Mod. Phys. D* **09**, 303 (2000).
 - [17] Laura Cadonati and Szabolcs Márka, CORPOWER: A cross-correlation-based algorithm for triggered and untriggered gravitational-wave burst searches, *Classical Quantum Gravity* **22**, S1159 (2005).
 - [18] Warren G. Anderson, Patrick R. Brady, Jolien D. Creighton, and Éanna É. Flanagan, Excess power statistic for detection of burst sources of gravitational radiation, *Phys. Rev. D* **63**, 042003 (2001).
 - [19] Andrea Viceré, Optimal detection of burst events in gravitational wave interferometric observatories, *Phys. Rev. D* **66**, 062002 (2002).
 - [20] S. Klimenko and G. Mitselmakher, A wavelet method for detection of gravitational wave bursts, *Classical Quantum Gravity* **21**, S1819 (2004).
 - [21] M. Rakhmanov and S. Klimenko, A cross-correlation method for burst searches with networks of misaligned gravitational-wave detectors, *Classical Quantum Gravity* **22**, S1311 (2005).
 - [22] S. Klimenko, I. Yakushin, A. Mercer, and G. Mitselmakher, A coherent method for detection of gravitational wave bursts, *Classical Quantum Gravity* **25**, 114029 (2008).
 - [23] S.W. Hawking and W. Israel, *Three Hundred Years of Gravitation* (Cambridge University Press, Cambridge, UK, 1989).
 - [24] Benjamin J. Owen and B.S. Sathyaprakash, Matched filtering of gravitational waves from inspiraling compact binaries: Computational cost and template placement, *Phys. Rev. D* **60**, 022002 (1999).
 - [25] Curt Cutler, Theodoros A. Apostolatos, Lars Bildsten, Lee Smalley Finn, Éanna E. Flanagan, Daniel Kennefick, Dragoljub M. Markovic, Amos Ori, Eric Poisson, Gerald Jay Sussman, and Kip S. Thorne, The Last Three Minutes: Issues in Gravitational-Wave Measurements of Coalescing Compact Binaries, *Phys. Rev. Lett.* **70**, 2984 (1993).
 - [26] Curt Cutler and Éanna E. Flanagan, Gravitational waves from merging compact binaries: How accurately can one

- extract the binary's parameters from the inspiral waveform, *Phys. Rev. D* **49**, 2658 (1994).
- [27] Éanna É. Flanagan and Scott A. Hughes, Measuring gravitational waves from binary black hole coalescences. I. Signal to noise for inspiral, merger, and ringdown, *Phys. Rev. D* **57**, 4535 (1998).
- [28] Éanna É. Flanagan and Scott A. Hughes, Measuring gravitational waves from binary black hole coalescences. II. The waves' information and its extraction, with and without templates, *Phys. Rev. D* **57**, 4566 (1998).
- [29] B. Abbott *et al.*, Analysis of LIGO data for gravitational waves from binary neutron stars, *Phys. Rev. D* **69**, 122001 (2004).
- [30] PYCBC software releases, <https://github.com/gwastro/pycbc/releases>.
- [31] Bruce Allen, Warren G. Anderson, Patrick R. Brady, Duncan A. Brown, and Jolien D. E. Creighton, FINDCHIRP: An algorithm for detection of gravitational waves from inspiraling compact binaries, *Phys. Rev. D* **85**, 122006 (2012).
- [32] C. M. Biwer, Collin D. Capano, Soumi De, Miriam Cabero, Duncan A. Brown, Alexander H. Nitz, and V. Raymond, PYCBC inference: A PYTHON-based parameter estimation toolkit for compact binary coalescence signal, *Publ. Astron. Soc. Pac.* **131**, 024503 (2019).
- [33] Bruce Allen, χ^2 time-frequency discriminator for gravitational wave detection, *Phys. Rev. D* **71**, 062001 (2005).
- [34] Tito Dal Canton, Alexander H. Nitz, Andrew P. Lundgren, Alex B. Nielsen, Duncan A. Brown, Thomas Dent, Ian W. Harry, Badri Krishnan, Andrew J. Miller, Karl Wette, Karsten Wiesner, and Joshua L. Willis, Implementing a search for aligned-spin neutron star-black hole systems with advanced ground based gravitational wave detectors, *Phys. Rev. D* **90**, 082004 (2014).
- [35] Samantha A. Usman *et al.*, The PYCBC search for gravitational waves from compact binary coalescence, *Classical Quantum Gravity* **33**, 215004 (2016).
- [36] Alexander H. Nitz, Tito Dal Canton, Derek Davis, and Steven Reyes, Rapid detection of gravitational waves from compact binary mergers with PYCBC Live, *Phys. Rev. D* **98**, 024050 (2018).
- [37] Cody Messick *et al.*, Analysis framework for the prompt discovery of compact binary mergers in gravitational-wave data, *Phys. Rev. D* **95**, 042001 (2017).
- [38] Surabhi Sachdev *et al.*, The GstLAL search analysis methods for compact binary mergers in Advanced LIGO's second and Advanced Virgo's first observing runs, [arXiv:1901.08580](https://arxiv.org/abs/1901.08580).
- [39] Chad Hanna *et al.*, Fast evaluation of multidetector consistency for real-time gravitational wave searches, *Phys. Rev. D* **101**, 022003 (2020).
- [40] Qi Chu, Low-latency detection and localization of gravitational waves from compact binary coalescences, Ph.D. thesis, The University of Western Australia, 2017.
- [41] T. Adams, D. Buskulic, V. Germain, G. M. Guidi, F. Marion, M. Montani, B. Mours, F. Piergiovanni, and G. Wang, Low-latency analysis pipeline for compact binary coalescences in the advanced gravitational wave detector era, *Classical Quantum Gravity* **33**, 175012 (2016).
- [42] Alexander H. Nitz, Thomas Dent, Tito Dal Canton, Stephen Fairhurst, and Duncan A. Brown, Detecting binary compact-object mergers with gravitational waves: Understanding and improving the sensitivity of the PYCBC search, *Astrophys. J.* **849**, 118 (2017).
- [43] Jingkai Yan, Mariam Avagyan, Robert E. Colgan, Doğa Veske, Imre Bartos, John Wright, Zsuzsa Márka, and Szabolcs Márka, Generalized approach to matched filtering using neural networks, *Phys. Rev. D* **105**, 043006 (2022).
- [44] V. Gayathri, J. Healy, J. Lange, B. O'Brien, M. Szczepańczyk, Imre Bartos, M. Campanelli, S. Klimenko, C. O. Lousto, and R. O'Shaughnessy, Eccentricity estimate for black hole mergers with numerical relativity simulations, *Nat. Astron.* **6**, 344 (2022).
- [45] S. D. Mohanty and S. V. Dhurandhar, Hierarchical search strategy for the detection of gravitational waves from coalescing binaries, *Phys. Rev. D* **54**, 7108 (1996).
- [46] Anand S. Sengupta, Sanjeev Dhurandhar, and Albert Lazzarini, Faster implementation of the hierarchical search algorithm for detection of gravitational waves from inspiraling compact binaries, *Phys. Rev. D* **67**, 082004 (2003).
- [47] Benjamin J. Owen and Bangalore Suryanarayana Sathyaprakash, Matched filtering of gravitational waves from inspiraling compact binaries: Computational cost and template placement, *Phys. Rev. D* **60**, 022002 (1999).
- [48] Anand S. Sengupta, Sanjeev V. Dhurandhar, Albert Lazzarini, and Tom Prince, Extended hierarchical search (EHS) algorithm for detection of gravitational waves from inspiralling compact binaries, *Classical Quantum Gravity* **19**, 1507 (2002).
- [49] Bhooshan Gadre, Sanjit Mitra, and Sanjeev Dhurandhar, Hierarchical search strategy for the efficient detection of gravitational waves from nonprecessing coalescing compact binaries with aligned-spins, *Phys. Rev. D* **99**, 124035 (2019).
- [50] Rahul Dhurkunde, Henning Fehrmann, and Alexander H. Nitz, A hierarchical approach to matched filtering using a reduced basis, *Phys. Rev. D* **105**, 103001 (2022).
- [51] Kristen Grauman and Trevor Darrell, The pyramid match kernel: Discriminative classification with sets of image features, in *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05)* (IEEE, New York, 2005), Vol. 2, pp. 1458–1465.
- [52] Qiang Chen, Zheng Song, Yang Hua, Zhongyang Huang, and Shuicheng Yan, Hierarchical matching with side information for image classification, in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, New York, 2012), pp. 3426–3433.
- [53] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, in *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)* (IEEE, New York, 2006), Vol. 2, pp. 2169–2178.
- [54] Chenlin Shen, Changlong Sun, Jingjing Wang, Yangyang Kang, Shoushan Li, Xiaozhong Liu, Luo Si, Min Zhang, and Guodong Zhou, Sentiment classification towards

- question-answering with hierarchical matching network, in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (2018), pp. 3654–3663, <https://aclanthology.org/D18-1401/>.
- [55] Marc Vaillant and Christos Davatzikos, Hierarchical matching of cortical features for deformable brain image registration, in *Proceedings of the Biennial International Conference on Information Processing in Medical Imaging* (Springer, New York, 1999), pp. 182–195.
- [56] Zhe Lin, Larry S. Davis, David Doermann, and Daniel DeMenthon, Hierarchical part-template matching for human detection and segmentation, in *Proceedings of the 2007 IEEE 11th International Conference on Computer Vision* (IEEE, New York, 2007), pp. 1–8.
- [57] Pengfei Wang, Yu Fan, Shuzi Niu, Ze Yang, Yongfeng Zhang, and Jiafeng Guo, Hierarchical matching network for crime classification, in *Proceedings of the 42nd international ACM SIGIR Conference on Research and Development in Information Retrieval* (2019), pp. 325–334, [10.1145/3331184.3331223](https://doi.org/10.1145/3331184.3331223).
- [58] Timothy Gebhard, Niki Kilbertus, Giambattista Parascandolo, Ian Harry, and Bernhard Schölkopf, Convwave: Searching for gravitational waves with fully convolutional neural nets, in *Proceedings of the Workshop on Deep Learning for Physical Sciences (DLPS) at the 31st Conference on Neural Information Processing Systems (NIPS)* (2017), pp. 1–6, https://dl4physicsciences.github.io/files/nips_dlps_2017_13.pdf.
- [59] Daniel George and E. A. Huerta, Deep learning for real-time gravitational wave detection and parameter estimation: Results with advanced LIGO data, *Phys. Lett. B* **778**, 64 (2018).
- [60] Hunter Gabbard, Michael Williams, Fergus Hayes, and Chris Messenger, Matching Matched Filtering with Deep Networks for Gravitational-Wave Astronomy, *Phys. Rev. Lett.* **120**, 141103 (2018).
- [61] Daniel George and E. A. Huerta, Deep neural networks to enable real-time multimessenger astrophysics, *Phys. Rev. D* **97**, 044039 (2018).
- [62] XiLong Fan, Jin Li, Xin Li, YuanHong Zhong, and JunWei 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).
- [63] Filip Morawski, Michal Bejger, and Pawel Ciecielag, Convolutional neural network classifier for the output of the time-domain-statistic all-sky search for continuous gravitational waves, *Mach. Learn.* **1**, 025016 (2020).
- [64] Christoph Dreissigacker, Rahul Sharma, Chris Messenger, Ruining Zhao, and Reinhard Prix, Deep-learning continuous gravitational waves, *Phys. Rev. D* **100**, 044009 (2019).
- [65] Plamen G. Krastev, Real-time detection of gravitational waves from binary neutron stars using artificial neural networks, *Phys. Lett. B* **803**, 135330 (2020).
- [66] Bai-Jiong Lin, Xiang-Ru Li, and Wo-Liang Yu, Binary neutron stars gravitational wave detection based on wavelet packet analysis and convolutional neural networks, *Front. Phys.* **15**, 1 (2020).
- [67] Yu-Chiung Lin and Jiun-Huei Protty Wu, Detection of gravitational waves using Bayesian neural networks, *Phys. Rev. D* **103**, 063034 (2021).
- [68] Christopher Bresten and Jae-Hun Jung, Detection of gravitational waves using topological data analysis and convolutional neural network: An improved approach, [arXiv:1910.08245](https://arxiv.org/abs/1910.08245).
- [69] P. Astone, P. Cerdá-Durán, I. Di Palma, M. Drago, F. Muciaccia, C. Palomba, and F. Ricci, New method to observe gravitational waves emitted by core collapse supernovae, *Phys. Rev. D* **98**, 122002 (2018).
- [70] Takahiro S. Yamamoto and Takahiro Tanaka, Use of excess power method and convolutional neural network in all-sky search for continuous gravitational waves, *Phys. Rev. D* **103**, 084049 (2021).
- [71] Christoph Dreissigacker and Reinhard Prix, Deep-learning continuous gravitational waves: Multiple detectors and realistic noise, *Phys. Rev. D* **102**, 022005 (2020).
- [72] Roberto Corizzo, Michelangelo Ceci, Eftim Zdravetski, and Nathalie Japkowicz, Scalable auto-encoders for gravitational waves detection from time series data, *Expert Syst. Appl.* **151**, 113378 (2020).
- [73] Andrew L. Miller, Pia Astone, Sabrina D’Antonio, Sergio Frasca, Giuseppe Intini, Iuri La Rosa, Paola Leaci, Simone Mastrogiovanni, Federico Muciaccia, Andonis Mitidis *et al.*, How effective is machine learning to detect long transient gravitational waves from neutron stars in a real search?, *Phys. Rev. D* **100**, 062005 (2019).
- [74] Joe Bayley, Chris Messenger, and Graham Woan, Robust machine learning algorithm to search for continuous gravitational waves, *Phys. Rev. D* **102**, 083024 (2020).
- [75] Plamen G. Krastev, Kiranjyot Gill, V. Ashley Villar, and Edo 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).
- [76] Hua-Mei Luo, Wenbin Lin, Zu-Cheng Chen, and Qing-Guo Huang, Extraction of gravitational wave signals with optimized convolutional neural network, *Front. Phys.* **15**, 1 (2020).
- [77] Gerson R. Santos, Marcela P. Figueiredo, Antonio de Pádua Santos, Pavlos Protopapas, and Tiago A. E. Ferreira, Gravitational wave detection and information extraction via neural networks, [arXiv:2003.09995](https://arxiv.org/abs/2003.09995).
- [78] Man Leong Chan, Ik Siong Heng, and Chris Messenger, Detection and classification of supernova gravitational wave signals: A deep learning approach, *Phys. Rev. D* **102**, 043022 (2020).
- [79] Heming Xia, Lijing Shao, Junjie Zhao, and Zhoujian Cao, Improved deep learning techniques in gravitational-wave data analysis, *Phys. Rev. D* **103**, 024040 (2021).
- [80] Rahul Biswas, Lindy Blackburn, Junwei Cao, Reed Essick, Kari Alison Hodge, Erotokritos Katsavounidis, Kyungmin Kim, Young-Min Kim, Eric-Olivier Le Bigot, Chang-Hwan Lee *et al.*, Application of machine learning algorithms to the study of noise artifacts in gravitational-wave data, *Phys. Rev. D* **88**, 062003 (2013).
- [81] Daniel George, Hongyu Shen, and E. A. Huerta, Deep transfer learning: A new deep learning glitch classification

- method for advanced LIGO, *Phys. Rev. D* **97**, 101501(R) (2018).
- [82] Nikhil Mukund, Sheelu Abraham, Shivaraj Kandhasamy, Sanjit Mitra, and Ninan Sajeeth Philip, Transient classification in LIGO data using difference boosting neural network, *Phys. Rev. D* **95**, 104059 (2017).
 - [83] Massimiliano Razzano and Elena Cuoco, Image-based deep learning for classification of noise transients in gravitational wave detectors, *Classical Quantum Gravity* **35**, 095016 (2018).
 - [84] S. Coughlin, S. Bahaadini, N. Rohani, M. Zevin, O. Patane, M. Harandi, C. Jackson, V. Noroozi, S. Allen, J. Areeda *et al.*, Classifying the unknown: Discovering novel gravitational-wave detector glitches using similarity learning, *Phys. Rev. D* **99**, 082002 (2019).
 - [85] Robert E. Colgan, K. Rainer Corley, Yenson Lau, Imre Bartos, John N. Wright, Zsuzsa Márka, and Szabolcs Márka, Efficient gravitational-wave glitch identification from environmental data through machine learning, *Phys. Rev. D* **101**, 102003 (2020).
 - [86] Hiroyuki Nakano, Tatsuya Narikawa, Ken-ichi Oohara, Kazuki Sakai, Hisa-aki Shinkai, Hirotaka Takahashi, Takahiro Tanaka, Nami Uchikata, Shun Yamamoto, and Takahiro S. Yamamoto, Comparison of various methods to extract ringdown frequency from gravitational wave data, *Phys. Rev. D* **99**, 124032 (2019).
 - [87] Stephen R. Green, Christine Simpson, and Jonathan Gair, Gravitational-wave parameter estimation with autoregressive neural network flows, *Phys. Rev. D* **102**, 104057 (2020).
 - [88] Juan Pablo Marulanda, Camilo Santa, and Antonio Enea Romano, Deep learning merger masses estimation from gravitational waves signals in the frequency domain, *Phys. Lett. B* **810**, 135790 (2020).
 - [89] A. Caramete, A. I. Constantinescu, L. I. Caramete, T. Popescu, R. A. Balasov, D. Felea, M. V. Rusu, P. Stefanescu, and O. M. Tintareanu, Characterization of gravitational waves signals using neural Networks, *arXiv:2009.06109*.
 - [90] Arnaud Delaunoy, Antoine Wehenkel, Tanja Hinderer, Samaya Nissanke, Christoph Weniger, Andrew R. Williamson, and Gilles Louppe, Lightning-fast gravitational wave parameter inference through neural amortization, *arXiv:2010.12931*.
 - [91] Hongyu Shen, Daniel George, Eliu A. Huerta, and Zhizhen Zhao, Denoising gravitational waves with enhanced deep recurrent denoising auto-encoders, in *Proceedings of the ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, New York, 2019), pp. 3237–3241.
 - [92] Wei Wei and E. A. Huerta, Gravitational wave denoising of binary black hole mergers with deep learning, *Phys. Lett. B* **800**, 135081 (2020).
 - [93] Timothy D. Gebhard, Niki Kilbertus, Ian Harry, and Bernhard Schölkopf, Convolutional neural networks: A magic bullet for gravitational-wave detection?, *Phys. Rev. D* **100**, 063015 (2019).
 - [94] Thomas Dent and John Veitch, Optimizing gravitational-wave searches for a population of coalescing binaries: Intrinsic parameters, *Phys. Rev. D* **89**, 062002 (2014).
 - [95] George Turin, An introduction to matched filters, *IRE Trans. Inf. Theory* **6**, 311 (1960).
 - [96] Philip M. Woodward, *Probability and Information Theory, with Applications to Radar: International Series of Monographs on Electronics and Instrumentation* (Elsevier, New York, 2014), Vol. 3.
 - [97] Bernard F. Schutz, Gravitational wave astronomy, *Classical Quantum Gravity* **16**, A131 (1999).
 - [98] GWOSC summary, <https://ldas-jobs.ligo-la.caltech.edu/detchar/summary/day/20170801/>.
 - [99] Diederik P. Kingma and Jimmy Ba, ADAM: A method for stochastic optimization, *arXiv:1412.6980*.
 - [100] Marco Cavaglia, Kai Staats, and Teerth Gill, Finding the origin of noise transients in LIGO data with machine learning, *Commun. Comput. Phys.* **25**, 963 (2019).
 - [101] Robert E. Colgan, K. Rainer Corley, Yenson Lau, Imre Bartos, John N. Wright, Zsuzsa Márka, and Szabolcs Márka, Efficient gravitational-wave glitch identification from environmental data through machine learning, *Phys. Rev. D* **101**, 102003 (2020).
 - [102] Robert E. Colgan, Jingkai Yan, Zsuzsa Márka, Imre Bartos, Szabolcs Márka, and John N. Wright, Architectural optimization and feature learning for high-dimensional time series datasets, *arXiv:2202.13486*.
 - [103] Robert E. Colgan, Zsuzsa Márka, Jingkai Yan, Imre Bartos, John N. Wright, and Szabolcs Márka, Detecting and diagnosing terrestrial gravitational-wave mimics through feature learning, *arXiv:2203.05086*.
 - [104] Elena Cuoco *et al.*, Enhancing gravitational-wave science with machine learning, *Mach. Learn.* **2**, 011002 (2020).
 - [105] R. Abbott *et al.*, Open data from the first and second observing runs of Advanced LIGO and Advanced Virgo, *SoftwareX* **13**, 100658 (2021).
 - [106] GWOSC, Gravitational wave open science center.