Physics Letters B 778 (2018) 64–70

Contents lists available at ScienceDirect

Physics Letters B

www.elsevier.com/locate/physletb

Deep Learning for real-time gravitational wave detection and parameter estimation: Results with Advanced LIGO data

Daniel George [a,b,*], E.A. Huerta [b]

[a] Department of Astronomy, University of Illinois at Urbana-Champaign, Urbana, IL, 61801, United States
[b] NCSA, University of Illinois at Urbana-Champaign, Urbana, IL, 61801, United States

ARTICLE INFO

Article history:
Received 8 November 2017
Received in revised form 30 November 2017
Accepted 19 December 2017
Available online 27 December 2017
Editor: H. Peiris

Keywords:
Deep Learning
Convolutional neural networks
Gravitational waves
LIGO
Time-series signal processing
Classification and regression

ABSTRACT

The recent Nobel-prize-winning detections of gravitational waves from merging black holes and the subsequent detection of the collision of two neutron stars in coincidence with electromagnetic observations have inaugurated a new era of multimessenger astrophysics. To enhance the scope of this emergent field of science, we pioneered the use of deep learning with convolutional neural networks, that take time-series inputs, for rapid detection and characterization of gravitational wave signals. This approach, `Deep Filtering`, was initially demonstrated using simulated LIGO noise. In this article, we present the extension of `Deep Filtering` using real data from LIGO, for both *detection* and *parameter estimation* of gravitational waves from binary black hole mergers using continuous data streams from multiple LIGO detectors. We demonstrate for the first time that machine learning can detect and estimate the true parameters of real events observed by LIGO. Our results show that `Deep Filtering` achieves similar sensitivities and lower errors compared to matched-filtering while being far more computationally efficient and more resilient to glitches, allowing real-time processing of weak time-series signals in non-stationary non-Gaussian noise with minimal resources, and also enables the detection of new classes of gravitational wave sources that may go unnoticed with existing detection algorithms. This unified framework for data analysis is ideally suited to enable coincident detection campaigns of gravitational waves and their multimessenger counterparts in real-time.

© 2018 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/). Funded by SCOAP[3].

1. Introduction

The first detection (GW150914) of gravitational waves (GWs), from the merger of two black holes (BHs), with the advanced Laser Interferometer Gravitational-wave Observatory (LIGO) [1] has set in motion a scientific revolution [2] leading to the Nobel prize in Physics in 2017. This and subsequent groundbreaking discoveries [3–6] were brought to fruition by a trans-disciplinary research program at the interface of experimental and theoretical physics, computer science and engineering as well as the exploitation of high-performance computing (HPC) for numerical relativity simulations [7–9] and high-throughput computing facilities for data analysis [10,11].

The recent detection of the binary black hole (BBH) merger (GW170814) with a three-detector network enabled new phenomenological tests of general relativity regarding the nature of GW polarizations, while significantly improving the sky localization of this GW transient [6]. This enhanced capability to localize GW transients provided critical input for the first detection of GWs from the merger of two neutron stars (NSs) and in conjunction with follow-up observations across the electromagnetic (EM) spectrum [12]. This multimessenger event has finally confirmed that NS mergers are the central engines of short gamma ray bursts [13–18].

Matched-filtering, the most sensitive GW detection algorithm used by LIGO, currently targets a 4D parameter space (compact binary sources with spin-aligned components on quasi-circular orbits) [19–21]—a subset of the 9D parameter space available to GW detectors [22–26]. Recent studies also indicate that these searches may miss GWs generated by compact binary populations formed in dense stellar environments [25,27–29]. Extending these template-matching searches to target spin-precessing, quasi-circular or eccentric BBHs is computationally prohibitive [30].

Based on the aforementioned considerations, we need a new paradigm to overcome the limitations and computational challenges of existing GW detection algorithms. An ideal candidate would be the rapidly advancing field called “Deep Learning”, which is a highly scalable machine learning technique that can learn directly from raw data, without any manual feature engineering, by

* Corresponding author.
E-mail address: dgeorge5@illinois.edu (Daniel George).

https://doi.org/10.1016/j.physletb.2017.12.053
0370-2693/© 2018 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/). Funded by SCOAP[3].

using deep hierarchical layers of "artificial neurons", called neural networks, in combination with optimization techniques based on back-propagation and gradient descent [31,32]. Deep learning, especially with the aid of GPU computing, has recently achieved immense success in both commercial applications and artificial intelligence (AI) research [31,33–38] and has also been applied in astrophysics [39–47].

Our technique, `Deep Filtering` [48], employs a system of two deep convolution neural networks (CNNs [49]) that directly take time-series inputs for both classification and regression. In our foundational article [48], we provided a comprehensive introduction to the fundamental concepts of deep learning and CNNs along with a detailed description of this method. Our previous results showed that CNNs can outperform traditional machine learning methods, reaching sensitivities comparable to matched-filtering for directly processing highly noisy time-series data streams to detect weak GW signals and estimate the parameters of their source in real-time, using GW signals injected into *simulated* LIGO noise.

In this article, we extend `Deep Filtering` to analyze GW signals in *real* LIGO noise. We demonstrate, for the first time, that Deep Learning can be used for both signal detection and multiple-parameter estimation directly from extremely weak time-series signals embedded in highly non-Gaussian and non-stationary noise, once trained with some templates of the expected signals. Our results show that deep CNNs achieve performance comparable to matched-filtering methods, while being several orders of magnitude faster and far more resilient to transient noise artifacts such as glitches. We also illustrate how `Deep Filtering` can deal with data streams of arbitrary length from multiple detectors. Most importantly, this article shows for the very *first* time that machine learning can successfully detect and recover true parameters of *real* GW signals observed by LIGO. Furthermore, we show that after a single training process, `Deep Filtering` can automatically generalize to noise having new Power Spectral Densities (PSDs) from different LIGO events, without re-training.

Our results indicate that `Deep Filtering` can interpolate between elements in the template bank, generalize to several new classes of signals beyond the training data, and, surprisingly, detect GW signals and measure their parameters even when they are contaminated by glitches. We present experiments demonstrating the robustness of `Deep Filtering` in the presence of glitches, which indicate its applicability in the future for glitch classification and clustering efforts essential for GW detector characterization [46]. Deep learning, in principle, can learn characteristics of noise in the LIGO detectors and develop better strategies than matched-filtering, which is known to be optimal only for Gaussian noise. Since all the intensive computation is diverted to the one-time training stage of the CNNs, template banks of practically any size may be used for training after which continuous data streams can be analyzed in real-time with a single CPU, while very intensive searches can be rapidly carried out using a single GPU. The initial estimates provided by `Deep Filtering` may be used to instantly narrow down the parameter space of GW detections, which can then be followed up with existing pipelines using a few templates around the predicted parameters to obtain more informative parameter estimates and confidence measurements, thus accelerating GW analysis with minimal computational resources across the full parameter space of signals.

2. Methods

`Deep Filtering` consists of two steps, involving a classifier CNN and a predictor CNN, with similar architectures, as described in our previous article [48]. The classifier has an additional softmax layer which returns probabilities for True or False depending on whether a signal is present. The classifier is first applied to the continuous data stream via a sliding window. If the classifier returns higher probability for True, the predictor is then applied to the same input to determine the parameters of the source. In a multi-detector scenario, the `Deep Filtering` CNNs may be applied separately to each data stream and the coincidence of detections with similar parameters would strengthen the confidence of a true detection, which can then be verified quickly by matched-filtering with the predicted templates.

In this work, we have used injections of GW templates originating from quasi-circular, non-spinning, stellar-mass BBH systems, which LIGO/Virgo is expected to detect with the highest rate [50]. We assumed the source is optimally oriented with respect to the detectors which reduces our parameter-space to the two individual masses of the BBH system, which we restricted in the range 5 $M_\odot$ to 75 $M_\odot$ such that their mass-ratios were between 1 and 10. In the same manner as before, we fixed the input duration to 1 second, and a sampling rate of 8192 Hz, which is more than sufficient for the events we are considering. These are arbitrary choices, as the input size of the CNNs can be easily modified to take inputs with any duration or sampling rate from any number of detectors.

The datasets of waveform templates used to train and test our CNNs were obtained using the open-source, effective-one-body (EOB) code [51]. Our training set contained about 2500 templates, with BBHs component masses sampled in the range 5 $M_\odot$ to 75 $M_\odot$ in steps of 1 $M_\odot$. The testing dataset also contained approximately 2500 templates with intermediate component masses separated from the training set by 0.5 $M_\odot$ each. Subsequently, we produced copies of each signal by shifting the location of their peaks randomly within the final 0.2 seconds to make the CNNs more resilient to time translations. This means that in practice, our algorithm will be applied to the continuous data stream using a 1-second sliding window with offsets of 0.2 seconds.

We obtained real LIGO data from the LIGO Open Science Center (LOSC) around the first 3 GW events, namely, GW150914, LVT151012, and GW151226. Each event contained 4096 seconds of real data from each detector. We used noise sampled from GW151226 and LVT151012 for training and validation of our model and noise from GW150914 was used for testing. These tests ensure that our method is able to generalize to different noise distributions, also in the presence of transient glitches, since it is well known that the PSD of LIGO is highly non-stationary, varying widely with time. Therefore, if `Deep Filtering` performs well on these test sets, it would also perform well on data from future time periods, without being re-trained.

Different realizations of noise was randomly sampled from the real LIGO data around the two events GW151226 and LVT151012 and superimposed with signals from the training set over multiple iterations, thus amplifying the size of the training datasets. The power of the noise was adjusted according to the desired optimal matched-filter Signal-to-Noise Ratio (SNR [52]) for each training round. The inputs were then whitened with the average PSD of the real noise measured at that time-period. We also scaled and mixed different samples of LIGO noise together to artificially produce more training data and various levels of Gaussian noise was also added to augment the training process. However, the testing results were measured using only pure LIGO noise not used in training with true GW signals and with signals injected from the unaltered test sets (see Fig. 1).

We used hyperparameters similar to our original CNNs in [48]. There were 4 convolution layers with the filter sizes to 64, 128, 256, and 512 respectively and 2 fully connected layers with sizes 128 and 64. The standard ReLU activation function, $\max(0, x)$, was used throughout as the non-linearity between layers. We used kernel sizes of 16, 16, 16, and 32 for the convolutional layers and 4 for

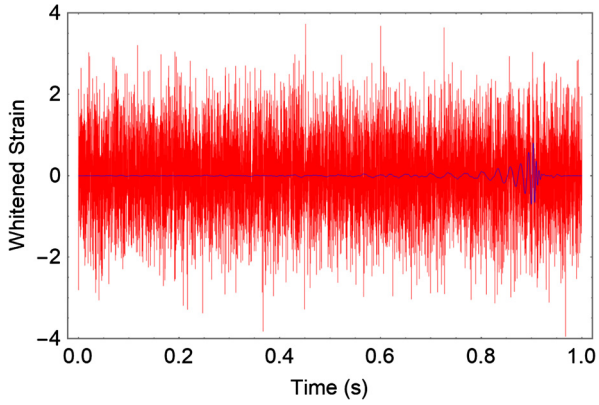

Fig. 1. Sample signal injected into real LIGO noise. The red time-series is an example of the input to our `Deep Filtering` algorithm. It contains a hidden BBH GW signal (blue) from our test set which was superimposed in real LIGO noise from the test set and whitened. For this injection, the optimal matched-filter SNR = 7.5 (peak power of this signal is 0.65 times the power of background noise). The component masses of the merging BHs are 57 $M_\odot$ and 33 $M_\odot$. The presence of this signal was detected directly from the (red) time-series input with over 99% sensitivity and the source's parameters were estimated with a mean relative error less than 10%. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

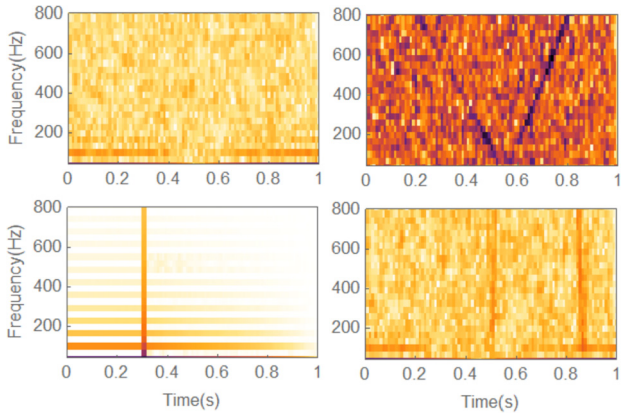

Fig. 2. Spectrograms of real LIGO noise test samples. We used signals injected into real data from the LIGO detectors in this article, ensuring that the training and testing sets did not contain noise from the same events. These are some random examples of real glitches that were present in our test set of LIGO noise. The `Deep Filtering` method takes the 1D strain directly as input and is able to correctly classify glitches as noise and detect true GW signals as well as simulated GW signals injected into these highly non-stationary non-Gaussian data streams, with similar sensitivity compared to matched-filtering.

all the (max) pooling layers. Stride was chosen to be 1 for all the convolution layers and 4 for all the pooling layers. We observed that using dilations [53] of 1, 2, 2, and 2 in the corresponding convolution layers improved the performance. The final layout of our predictor CNN is shown in Fig. 3.

We had originally optimized our CNN architectures to deal with only Gaussian noise having a flat PSD. However, we later found that this model also obtained the best performance with noise having the colored PSD of LIGO, among all the models we tested. This indicates that our CNNs will be robust to a wide range of noise distributions. Furthermore, pre-training the CNNs on Gaussian noise (transfer learning) before fine-tuning on the limited amount of real noise prevented over-fitting, i.e., memorizing only the training data without generalizing to new inputs. We used the Wolfram Language (*Mathematica* [54]) neural network functionality, built using the open-source MXNet framework [55], that uses the cuDNN library [56] for accelerating the training on NVIDIA

	Input	vector (size: 8192)
1	Reshape	matrix (size: 1 × 8192)
2	Convolution	matrix (size: 64 × 8177)
3	Pooling	matrix (size: 64 × 2044)
4	ReLU	matrix (size: 64 × 2044)
5	Convolution	matrix (size: 128 × 2014)
6	Pooling	matrix (size: 128 × 503)
7	ReLU	matrix (size: 128 × 503)
8	Convolution	matrix (size: 256 × 473)
9	Pooling	matrix (size: 256 × 118)
10	ReLU	matrix (size: 256 × 118)
11	Convolution	matrix (size: 512 × 56)
12	Pooling	matrix (size: 512 × 14)
13	ReLU	matrix (size: 512 × 14)
14	Flatten	vector (size: 7168)
15	Linear Layer	vector (size: 128)
16	ReLU	vector (size: 128)
17	Linear Layer	vector (size: 64)
18	ReLU	vector (size: 64)
19	Linear Layer	vector (size: 2)
	Output	vector (size: 2)

Fig. 3. Architecture of deep convolutional neural network. This is the dilated 1D CNN used as the predictor which outputs the component masses of the BBH system. The classifier has the same architecture, except for a softmax layer added at the end which outputs the probability for the presence of a GW signal. The input is a time-series vector of length 8192 corresponding to 1 s of data sampled at 8192 Hz. The classifier is applied separately to continuous data streams from each detector using a sliding window. If the classifier detects a signal in coincidence across multiple detectors, then the inputs are fed to the predictor which estimates the parameters of the GW source. The classifier is separated from the predictor and softmax regression is used so that neurons can be added to the final layer of the classifier for representing different types of signals and glitches, and specialized predictors may be designed to be applied depending on the class of the signal.

GPUs. The learning algorithm was ADAM [57] and other details were the same as before [48].

While training, we used the curriculum learning strategy in our first article [48] to improve the performance and reduce training times of the CNNs while retaining performance at very high SNR. By starting off training inputs having high SNR (≥ 100) and then gradually increasing the noise in each subsequent training session until a final SNR distributed in the range 4 to 15, we found that the performance of prediction can be quickly maximized for low SNR while retaining performance at high SNR. We first trained the predictor on the datasets labeled with the BBH masses and then copied the weights of this network to initialize the classifier and then trained it on datasets having 90% pure random noise inputs, after adding a softmax layer. This transfer learning procedure, similar to multi-task learning, decreases the training time for the classifier and improves its sensitivity.

3. Results

The sensitivity (probability of detecting a true signal) of the classifier as a function of SNR is shown in Fig. 4. We achieved 100% sensitivity when SNR is greater than 10. The false alarm rate was tuned to be less than 1%, i.e., 1 per 100 seconds of noise in our test set was classified as signals. Given independent noise from multiple detectors, this implies our 2-detector false alarm rate would be less than 0.01%, when the classifier is applied independently to each detector and coincidence is enforced. Although the false alarm rate can be further decreased by tuning the fraction of noise used for training or by checking that the predicted parameters are consistent, this may not be necessary since run-

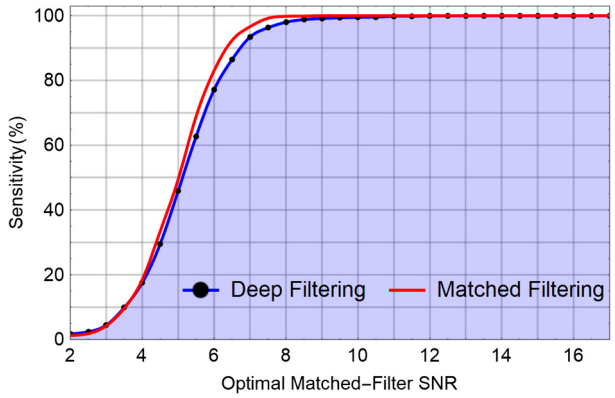

Fig. 4. Sensitivity of detection with real LIGO noise. The figure shows the sensitivity of detecting GW signals injected in real LIGO noise (from LOSC) from our test set using `Deep Filtering` compared with matched filtering with the same template bank used for training the CNN. Note that the SNR is on average proportional to 10 ± 1.5 times the ratio of the amplitude of the signal to the standard deviation of the noise for our test set. This implies that we are capable of detecting signals significantly weaker than the background noise. While matched-filtering has the advantage of being optimized with the PSD of the LIGO noise in the test set, `Deep Filtering` was only trained on noise from other events, therefore our results demonstrate the ability of the CNNs to automatically generalize to non-stationary LIGO noise having different PSDs without retraining.

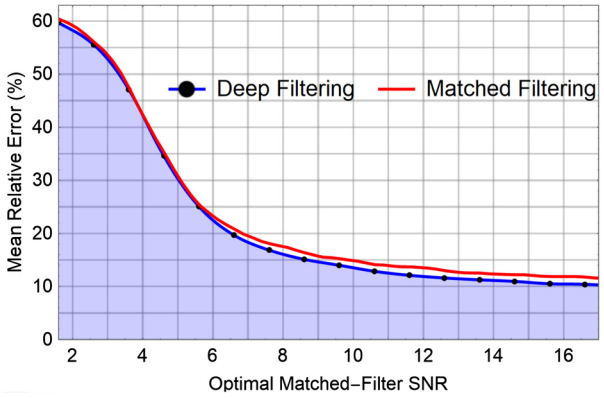

Fig. 5. Error in parameter estimation with real LIGO noise. This shows the mean percentage absolute error of estimating masses on our testing signals at each SNR, injected in real LIGO noise from events not used for training, compared to matched filtering using the same template bank that was used for training the CNN. While the mean error of matched-filtering, with the same template bank used for training, is always greater than 11% at every SNR we can see that the `Deep Filtering` method is able to interpolate to test set signals with intermediate parameter values.

ning matched-filtering pipelines with a few templates close to our predicted parameters can quickly eliminate these false alarms.

Our predictor was able to successfully measure the component masses given noisy GW signals, that were not used for training, with an error lower than the spacing between templates for optimal matched-filter SNR ≥ 15.0. The variation in relative error against SNR is shown in Fig. 5. We observed that the errors follow a Gaussian distribution for each region of the parameter space for SNR greater than 10. For high SNR, our predictor achieved mean relative error less than 10%, whereas matched-filtering with the same template bank always has error greater than 10%. This implies that `Deep Filtering` is capable of interpolating between templates seen in the training data.

Although, we trained only on simulated quasi-circular non-spinning GW injections, we applied `Deep Filtering` to the LIGO data streams containing a true GW signal, GW150914, using a sliding window of 1 s width with offsets of 0.2 s through the data around each event from each detector. This signal was

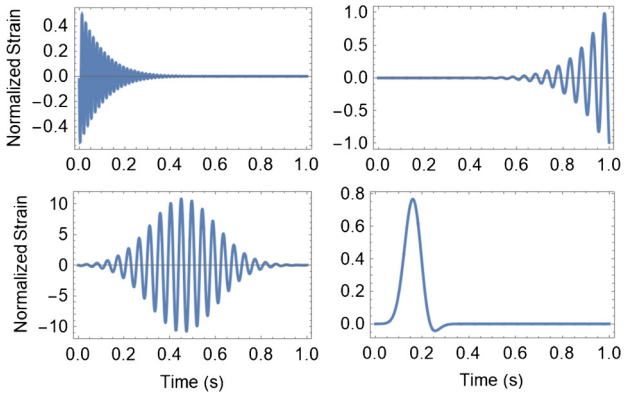

Fig. 6. Examples of sine-Gaussian glitches. These are some samples of simulated sine-Gaussian glitches from our test set. We found that our classifier was able to correctly differentiate GW signals from these glitches and classify them as noise when they were injected into real LIGO data streams. This suggests that `Deep Filtering` can be extended to create a unified pipeline for glitch classification along with signal detection and parameter estimation.

correctly identified by the classifier at the true position in time and each of the predicted component masses was within the published error bars [2]. There were zero false alarms after enforcing the constraint that the detection should be made simultaneously in multiple detectors. This shows that deep learning is able to generalize to real GW signals after being trained only with simulated GW templates injected into LIGO noise from other events with different PSDs. A demo showing the application of `Deep Filtering` to GW150914 can be found here: tiny.cc/CNN.

The data from the first LIGO event, that was used for testing, contained a large number of non-Gaussian transient noise called glitches. Some of these can be seen in Fig. 2. Therefore, our results demonstrate that the `Deep Filtering` method can automatically recognize these glitches and classify them as noise. This suggests that by adding additional neurons for each "glitch" class, `Deep Filtering` could serve as an alternative to glitch classification algorithms based on two-dimensional CNNs applied to spectrograms of LIGO [44,46] or machine learning methods based on manually engineered features [58–60].

Furthermore, we conducted some experiments to show the resilience of `Deep Filtering` to transient disturbances, with a simulated set of sine-Gaussian glitches, which cover a broad range of morphologies found in real LIGO glitches, following [59] (see Fig. 6 for some examples). We ensured that a different set of frequencies, amplitudes, peak positions, and widths were used for training and testing. We then injected some of these glitches into the training process and found that the classifier CNN was able to easily distinguish new glitches from true signals, with a false alarm rate less than 1%, using only single detector inputs. When we applied the standard naive matched-filtering algorithm to the same test set of glitches, approximately 30% of glitches were classified as signals due to their high SNR. This is because matched-filtering is unable to distinguish signals from loud glitches having similar frequencies. Note that additional signal consistency tests and coherence across detectors can be enforced to decrease this false alarm rate for both methods.

We then tested the performance of `Deep Filtering`, when a signal happens to occur in coincidence with a glitch, i.e., the signal is superimposed with both a glitch and real LIGO noise (see Fig. 7). We trained the CNNs by injecting glitches from the training set and measured the sensitivity of the classifier on the test set signals superimposed with glitches sampled from the test set of glitches. We found that over 80% of the signals with SNR of 10 were detected, and their parameters estimated with less than 30% relative error,

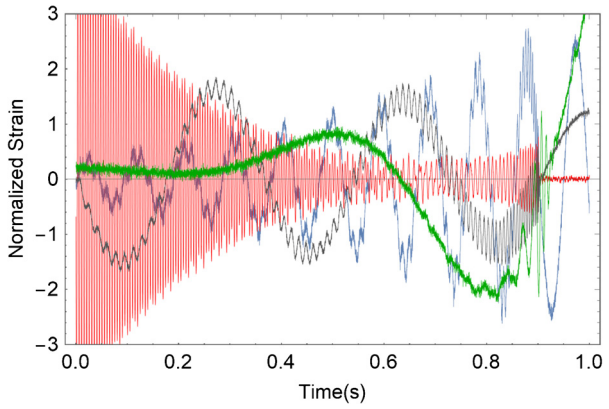

Fig. 7. Detecting signals contaminated by glitches. These are some of the signals in our test set injected into real LIGO noise and superimposed with simulated sine-Gaussian glitches from the test set. Each of these inputs was correctly detected as a signal by our classifier. This indicates that `Deep Filtering` can be used as an automatic trigger generator for GW signals that occur in coincidence with glitches.

even after they were superimposed with glitches. These results are very promising, motivating an in-depth investigation, since we may be able to automatically detect GW signals that occur during periods of bad data quality in the detectors using `Deep Filtering`, whereas currently such periods are often vetoed and left out of the analysis by LIGO pipelines.

Another important experiment that we carried out was to inject waveforms obtained from simulations of eccentric BBH systems, with eccentricities between 0.1 and 0.2 when entering the LIGO band that we performed using the open-source Einstein Toolkit [61,62] on Blue Waters, as well as waveforms from spin-precessing binaries from the public SXS catalog [63] as described in [48]. We found that, after injecting these signals into real LIGO noise from the test set events, they were detected by our classifier with the same sensitivity as the original test set of quasi-circular BBH waveforms, with 100% sensitivity of detection and less than 35% error in estimating masses for SNR > 10, thus demonstrating its ability to automatically generalize to GW signals from new classes of BBH mergers beyond the training data to some extent. This is particularly promising since recent studies indicate that moderately eccentric BBH populations may be missed by quasi-circular GW searches [25,28,64,65]. Parameter estimation of these new classes of signals could be more challenging, without training, as spin and eccentricity may have the effect of mimicking features of systems of different masses. We expect that by training on the full parameter space of signals, we will be able to decrease this error and also predict more parameters.

Both our CNNs are only 23 MB in size each, yet encodes all the relevant information from about 2500 GW templates ($\sim$ 300 MB) of templates and several GB of noise used to generate the training data. The time-domain matched-filtering algorithm used for comparison required over 2 s to analyze 1 s inputs on our CPU. The average time taken for evaluating each of our CNNs per second of data is approximately 85 milliseconds and 540 microseconds using a single CPU core and GPU respectively, thus enabling analysis even faster than real-time. While the computational cost of matched-filtering grows exponentially with the number of parameters, the `Deep Filtering` algorithm requires only a one-time training process, after which the analysis can be performed in constant time. Therefore, we expect the speed-up compared to matched-filtering to further increase by several orders of magnitude when the analysis is extended to a larger number of parameters. When considering the full range of signals that span a very high-dimensional parameter space which cannot be sampled densely due to computational costs, we expect `Deep Filtering` may have higher sensitivity due to its ability to interpolate and also because the one-time training process can be carried out with template banks much larger than what is feasible to use with matched-filtering.

4. Conclusion

In this article, we have shown for the very first time that CNNs can be used for both *detection* and *parameter estimation* of GW signals in LIGO data. This new paradigm for real-time GW analysis may overcome outstanding challenges regarding the extension of established GW detection algorithms to higher dimensions for targeting a deeper parameter space of astrophysically motivated GW sources. The results of `Deep Filtering` can be quickly verified, and the time of coalescence computed, via matched-filtering with a small set of templates in the region of parameter space predicted by the CNN. Therefore, by combining `Deep Filtering` with well-established GW detection algorithms, we may be able to accelerate multimessenger campaigns, pushing the frontiers of GW astrophysics and fully realize its potential for scientific discovery.

The intrinsic scalability of deep learning can overcome the curse of dimensionality [32] and take advantage of massive datasets [66]. This ability could enable fast automated GW searches covering millions or billions of templates over the full range of parameter-space that is beyond the reach of existing algorithms. Extending `Deep Filtering` to predict any number of parameters such as spins, eccentricities, etc., or additional classes of signals or noise, is as simple as adding an additional neuron for each new parameter, or class, to the final layer and training with noisy waveforms with the corresponding labels. Furthermore, the input dimensions of the CNNs can be enlarged to take simultaneous time-series inputs from multiple detectors, thus allowing coherent searches and measurements of parameters such as sky locations.

The results presented in this article provide a strong incentive to extend `Deep Filtering` to cover the parameter space of spin-aligned BBHs on quasi-circular orbits and beyond. This study is underway, and will be described in a subsequent article. In addition to our primary results, we have also presented experiments exhibiting the remarkable resilience of this method for detection in periods of bad data quality, even when GW signals are contaminated with non-Gaussian transients. This motivates including additional classes of real glitches, e.g., from the Gravity Spy project [46], to the training process to automatically classify and cluster glitches directly from the time-series inputs. It may be possible that, by investing more effort in hyperparameter tuning and by experimenting with deeper architectures, we may be able to outperform the sensitivity of matched-filtering searches in the future (since matched-filtering is proven to be the optimal algorithm only for Gaussian noise). These considerations provide sufficient motivation to develop and deploy—before the beginning of the upcoming LIGO-Virgo GW discovery campaign—a single, robust, and computationally efficient `Deep Filtering` pipeline that unifies real-time GW detection and parameter estimation, along with glitch classification and clustering.

`Deep Filtering` can be extended to provide instant alerts with accurate parameter estimates for EM follow-up campaigns. Furthermore, the initial point estimates provided by the `Deep Filtering` predictor may be used to inform and accelerate computationally intensive offline Bayesian parameter estimation methods [67,68] to estimate the full posterior distributions for each parameter by constraining the parameter space of new GW detections. It is also possible to obtain uncertainty distributions for the parameters predicted by the CNN [69], which will be explored in a following article. As deep CNNs excel at image processing, applying

the same approach to analyze raw telescope data may also accelerate the subsequent search for transient EM counterparts. Our results also suggest that, given templates of expected signals, `Deep Filtering` can be used as a generic tool for efficiently detecting and estimating properties of highly noisy time-domain signals embedded in Gaussian noise or non-stationary non-Gaussian noise, even in the presence of transient disturbances. Therefore, we anticipate that the techniques we have developed for analyzing weak signals hidden in complex noise backgrounds may also be useful in other domains of science and technology.

Acknowledgements

This research is part of the Blue Waters sustained-petascale computing project, supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) and the state of Illinois. Blue Waters is a joint effort of the University of Illinois at Urbana-Champaign and its National Center for Supercomputing Applications. The eccentric numerical relativity simulations used in this article were generated with the open source, community software, the Einstein Toolkit on the Blue Waters petascale supercomputer and XSEDE (TG-PHY160053). We express our gratitude to Gabrielle Allen, Ed Seidel, Roland Haas, Miguel Holgado, Haris Markakis, Zhizhen Zhao and other members of the NCSA Gravity Group along with Prannoy Mupparaju for comments and interactions and to the many others who reviewed our manuscript, including Jin Li and Kai Staats. We acknowledge the LIGO/Virgo collaboration, especially the CBC and MLA groups, for their feedback and for use of computational resources including the NVIDIA DGX-1 machine. We thank Vlad Kindratenko for granting us access to GPUs and HPC resources in the Innovative Systems Lab at NCSA. We are grateful to Wolfram Research for providing technical assistance and *Mathematica* licenses and to NVIDIA for donating several Tesla P100 GPUs, which we used for this analysis.

References

[1] The LIGO Scientific Collaboration, J. Aasi, et al., Class. Quantum Gravity 32 (2015) 074001, arXiv:1411.4547 [gr-qc].
[2] B.P. Abbott, R. Abbott, T.D. Abbott, M.R. Abernathy, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R.X. Adhikari, et al., Phys. Rev. Lett. 116 (2016) 061102, arXiv:1602.03837 [gr-qc].
[3] B.P. Abbott, R. Abbott, T.D. Abbott, M.R. Abernathy, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R.X. Adhikari, et al., Phys. Rev. Lett. 116 (2016) 241103, arXiv:1606.04855 [gr-qc].
[4] B.P. Abbott, R. Abbott, T.D. Abbott, M.R. Abernathy, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R.X. Adhikari, et al., Phys. Rev. X 6 (2016) 041015, arXiv:1606.04856 [gr-qc].
[5] B.P. Abbott, R. Abbott, T.D. Abbott, M.R. Abernathy, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R.X. Adhikari, et al., Phys. Rev. Lett. 118 (2017) 221101.
[6] The LIGO Scientific Collaboration, The Virgo Collaboration, B.P. Abbott, R. Abbott, T.D. Abbott, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, et al., ArXiv e-prints, arXiv:1709.09660 [gr-qc], 2017.
[7] A. Einstein, K. Preuss. Akad. Wiss. Berl. Sitz.ber. 1915 (1915) 844.
[8] B.P. Abbott, R. Abbott, T.D. Abbott, M.R. Abernathy, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R.X. Adhikari, et al., Phys. Rev. D 94 (2016) 064035, arXiv:1606.01262 [gr-qc].
[9] S.A. Usman, A.H. Nitz, I.W. Harry, C.M. Biwer, D.A. Brown, M. Cabero, C.D. Capano, T. Dal Canton, T. Dent, S. Fairhurst, M.S. Kehl, D. Keppel, B. Krishnan, A. Lenon, A. Lundgren, A.B. Nielsen, L.P. Pekowsky, H.P. Pfeiffer, P.R. Saulson, M. West, J.L. Willis, Class. Quantum Gravity 33 (2016) 215004, arXiv:1508.02357 [gr-qc].
[10] E.A. Huerta, R. Haas, E. Fajardo, D.S. Katz, S. Anderson, P. Couvares, J. Willis, T. Bouvet, J. Enos, W.T.C. Kramer, H.W. Leong, D. Wheeler, ArXiv e-prints, arXiv:1709.08767 [cs.DC], 2017.
[11] D. Weitzel, B. Bockelman, D.A. Brown, P. Couvares, F. Würthwein, E. Fajardo Hernandez, ArXiv e-prints, arXiv:1705.06202 [cs.DC], 2017.
[12] The LIGO Scientific Collaboration, The Virgo Collaboration, ArXiv e-prints, arXiv:1710.05832 [gr-qc], 2017.
[13] D. Eichler, M. Livio, T. Piran, D.N. Schramm, Nature 340 (1989) 126.
[14] B. Paczynski, Astrophys. J. Lett. 308 (1986) L43.
[15] R. Narayan, B. Paczynski, T. Piran, Astrophys. J. Lett. 395 (1992) L83, arXiv:astro-ph/9204001.
[16] C.S. Kochanek, T. Piran, Astrophys. J. 417 (1993) L17, arXiv:astro-ph/9305015.
[17] C.D. Ott, Class. Quantum Gravity 26 (2009) 063001, arXiv:0809.0695.
[18] E.S. Phinney, The Astronomy and Astrophysics Decadal Survey, Astronomy, vol. 2010, 2009, arXiv:0903.0098 [astro-ph.CO].
[19] D. Gerosa, M. Kesden, E. Berti, R. O'Shaughnessy, U. Sperhake, Phys. Rev. D 87 (2013) 104028, arXiv:1302.4442 [gr-qc].
[20] C.L. Rodriguez, S. Chatterjee, F.A. Rasio, Phys. Rev. D 93 (2016) 084029, arXiv:1602.02444 [astro-ph.HE].
[21] C.L. Rodriguez, M. Morscher, B. Pattabiraman, S. Chatterjee, C.-J. Haster, F.A. Rasio, Phys. Rev. Lett. 115 (2015) 051101, arXiv:1505.00792 [astro-ph.HE].
[22] F. Antonini, S. Chatterjee, C.L. Rodriguez, M. Morscher, B. Pattabiraman, V. Kalogera, F.A. Rasio, Astrophys. J. 816 (2016) 65, arXiv:1509.05080.
[23] S. Naoz, B. Kocsis, A. Loeb, N. Yunes, Astrophys. J. 773 (2013) 187, arXiv:1206.4316 [astro-ph.SR].
[24] J. Samsing, M. MacLeod, E. Ramirez-Ruiz, Astrophys. J. 784 (2014) 71, arXiv:1308.2964 [astro-ph.HE].
[25] E.A. Huerta, P. Kumar, B. Agarwal, D. George, H.-Y. Schive, H.P. Pfeiffer, R. Haas, W. Ren, T. Chu, M. Boyle, D.A. Hemberger, L.E. Kidder, M.A. Scheel, B. Szilagyi, Phys. Rev. D 95 (2017) 024038.
[26] L. Lehner, F. Pretorius, Annu. Rev. Astron. Astrophys. 52 (2014) 661, arXiv:1405.4840 [astro-ph.HE].
[27] S. Klimenko, G. Vedovato, M. Drago, F. Salemi, V. Tiwari, G.A. Prodi, C. Lazzaro, K. Ackley, S. Tiwari, C.F. Da Silva, G. Mitselmakher, Phys. Rev. D 93 (2016) 042004, arXiv:1511.05999 [gr-qc].
[28] E.A. Huerta, P. Kumar, S.T. McWilliams, R. O'Shaughnessy, N. Yunes, Phys. Rev. D 90 (2014) 084016, arXiv:1408.3406 [gr-qc].
[29] E.A. Huerta, D.A. Brown, Phys. Rev. D 87 (2013) 127501, arXiv:1301.1895 [gr-qc].
[30] I. Harry, S. Privitera, A. Bohé, A. Buonanno, Phys. Rev. D 94 (2016) 024012, arXiv:1603.02444 [gr-qc].
[31] Y. Lecun, Y. Bengio, G. Hinton, Nature 521 (2015) 436.
[32] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016.
[33] J. Schmidhuber, Neural Netw. 61 (2015) 85.
[34] A. Esteva, B. Kuprel, R.A. Novoa, J. Ko, S.M. Swetter, H.M. Blau, S. Thrun, Nature 542 (2017) 115.
[35] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, et al., Nature 550 (2017) 354.
[36] M. Moravčík, M. Schmid, N. Burch, V. Lisý, D. Morrill, N. Bard, T. Davis, K. Waugh, M. Johanson, M. Bowling, Science (2017), https://doi.org/10.1126/science.aam6960.
[37] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, K. Kavukcuoglu, ArXiv e-prints, arXiv:1609.03499 [cs.SD], 2016.
[38] T.J. O'Shea, J. Corgan, T.C. Clancy, Convolutional radio modulation recognition networks, in: C. Jayne, L. Iliadis (Eds.), Proceedings of the Engineering Applications of Neural Networks: 17th International Conference, EANN 2016, Aberdeen, UK, September 2–5, 2016, Springer International Publishing, Cham, 2016, pp. 213–226.
[39] Y.D. Hezaveh, L.P. Levasseur, P.J. Marshall, Nature 548 (2017) 555.
[40] T. Hinners, K. Tat, R. Thorp, ArXiv e-prints, arXiv:1710.06804 [astro-ph.IM], 2017.
[41] N. Sedaghat, A. Mahabal, ArXiv e-prints, arXiv:1710.01422 [astro-ph.IM], 2017.
[42] K.A. Pearson, L. Palafox, C.A. Griffith, ArXiv e-prints, arXiv:1706.04319 [astro-ph.IM], 2017.
[43] S. Caron, G.A. Gómez-Vargas, L. Hendriks, R. Ruiz de Austri, ArXiv e-prints, arXiv:1708.06706 [astro-ph.HE], 2017.
[44] D. George, H. Shen, E.A. Huerta, ArXiv e-prints, arXiv:1706.07446 [gr-qc], 2017.
[45] D. George, H. Shen, E.A. Huerta, ArXiv e-prints, arXiv:1711.07468 [astro-ph.IM], 2017.
[46] M. Zevin, S. Coughlin, S. Bahaadini, E. Besler, N. Rohani, S. Allen, M. Cabero, K. Crowston, A. Katsaggelos, S. Larson, T.K. Lee, C. Lintott, T. Littenberg, A. Lundgren, C. Oesterlund, J. Smith, L. Trouille, V. Kalogera, ArXiv e-prints, arXiv:1611.04596 [gr-qc], 2016.
[47] S. Bahaadini, N. Rohani, S. Coughlin, M. Zevin, V. Kalogera, A.K. Katsaggelos, ArXiv e-prints, arXiv:1705.00034 [cs.LG], 2017.
[48] D. George, E.A. Huerta, ArXiv e-prints, arXiv:1701.00008 [astro-ph.IM], 2017.
[49] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Proc. IEEE 86 (1998) 2278.
[50] K. Belczynski, D.E. Holz, T. Bulik, R. O'Shaughnessy, Nature 534 (2016) 512, arXiv:1602.04531 [astro-ph.HE].
[51] A. Taracchini, A. Buonanno, Y. Pan, T. Hinderer, M. Boyle, D.A. Hemberger, L.E. Kidder, G. Lovelace, A.H. Mroué, H.P. Pfeiffer, M.A. Scheel, B. Szilágyi, N.W. Taylor, A. Zenginoglu, Phys. Rev. D 89 (2014) 061502, arXiv:1311.2544 [gr-qc].
[52] B.J. Owen, B.S. Sathyaprakash, Phys. Rev. D 60 (1999) 022002.
[53] F. Yu, V. Koltun, in: ICLR, 2016.
[54] Wolfram Language System & Documentation Center, https://reference.wolfram.com/language/.

[55] T. Chen, M. Li, Y. Li, M. Lin, N. Wang, M. Wang, T. Xiao, B. Xu, C. Zhang, Z. Zhang, CoRR, arXiv:1512.01274, 2015.
[56] S. Chetlur, C. Woolley, P. Vandermersch, J. Cohen, J. Tran, B. Catanzaro, E. Shelhamer, CoRR, arXiv:1410.0759, 2014.
[57] D.P. Kingma, J. Ba, CoRR, arXiv:1412.6980, 2014.
[58] N. Mukund, S. Abraham, S. Kandhasamy, S. Mitra, N.S. Philip, Phys. Rev. D 95 (2017) 104059.
[59] J. Powell, et al., Class. Quantum Gravity 34 (2017) 034002, arXiv:1609.06262 [astro-ph.IM].
[60] J. Powell, D. Trifirò, E. Cuoco, I.S. Heng, M. Cavaglià, Class. Quantum Gravity 32 (2015) 215012, arXiv:1505.01299 [astro-ph.IM].
[61] F. Löffler, J. Faber, E. Bentivegna, T. Bode, P. Diener, R. Haas, I. Hinder, B.C. Mundim, C.D. Ott, E. Schnetter, G. Allen, M. Campanelli, P. Laguna, Class. Quantum Gravity 29 (2012) 115001, arXiv:1111.3344 [gr-qc].
[62] http://einsteintoolkit.org.
[63] T. Chu, H. Fong, P. Kumar, H.P. Pfeiffer, M. Boyle, D.A. Hemberger, L.E. Kidder, M.A. Scheel, B. Szilagyi, Class. Quantum Gravity 33 (2016) 165001, arXiv:1512.06800 [gr-qc].
[64] V. Tiwari, S. Klimenko, N. Christensen, E.A. Huerta, S.R.P. Mohapatra, A. Gopakumar, M. Haney, P. Ajith, S.T. McWilliams, G. Vedovato, M. Drago, F. Salemi, G.A. Prodi, C. Lazzaro, S. Tiwari, G. Mitselmakher, F. Da Silva, Phys. Rev. D 93 (2016) 043007, arXiv:1511.09240 [gr-qc].
[65] E.A. Huerta, C.J. Moore, P. Kumar, D. George, A.J.K. Chua, R. Haas, E. Wessel, D. Johnson, D. Glennon, A. Rebei, A.M. Holgado, J.R. Gair, H.P. Pfeiffer, ArXiv e-prints, arXiv:1711.06276 [gr-qc], 2017.
[66] M.M. Najafabadi, F. Villanustre, T.M. Khoshgoftaar, N. Seliya, R. Wald, E. Muharemagic, J. Big Data 2 (2015) 1.
[67] J. Veitch, V. Raymond, B. Farr, W. Farr, P. Graff, S. Vitale, B. Aylott, K. Blackburn, N. Christensen, M. Coughlin, W. Del Pozzo, F. Feroz, J. Gair, C.-J. Haster, V. Kalogera, T. Littenberg, I. Mandel, R. O'Shaughnessy, M. Pitkin, C. Rodriguez, C. Röver, T. Sidery, R. Smith, M. Van Der Sluys, A. Vecchio, W. Vousden, L. Wade, Phys. Rev. D 91 (2015) 042003, arXiv:1409.7215 [gr-qc].
[68] C. Pankow, P. Brady, E. Ochsner, R. O'Shaughnessy, Phys. Rev. D 92 (2015) 023002, arXiv:1502.04370 [gr-qc].
[69] L. Perreault Levasseur, Y.D. Hezaveh, R.H. Wechsler, Astrophys. J. Lett. 850 (2017) L7, arXiv:1708.08843.