https://phys.org/news/2018-04-machine-gravitational.html

PHYS.ORG Nanotechnology Physics Earth Astronomy & Space Technology Chemistry Biology

Home » Physics » General Physics » April 9, 2018

Machine learning could help search for gravitational waves

April 9, 2018, University of Glasgow

A visualization of a supercomputer simulation of merging black holes sending out gravitational waves. Credit: NASA/C. Henze

Featured

Matching Matched Filtering with Deep Networks for Gravitational-Wave Astronomy

(Apr 2018, Physical Review Letters 120)

Hunter Gabbard,* Michael Williams, Fergus Hayes, and Chris Messenger
from the University of Glasgow

Self-introduction

Lupin Chun-Che Lin (RAP)

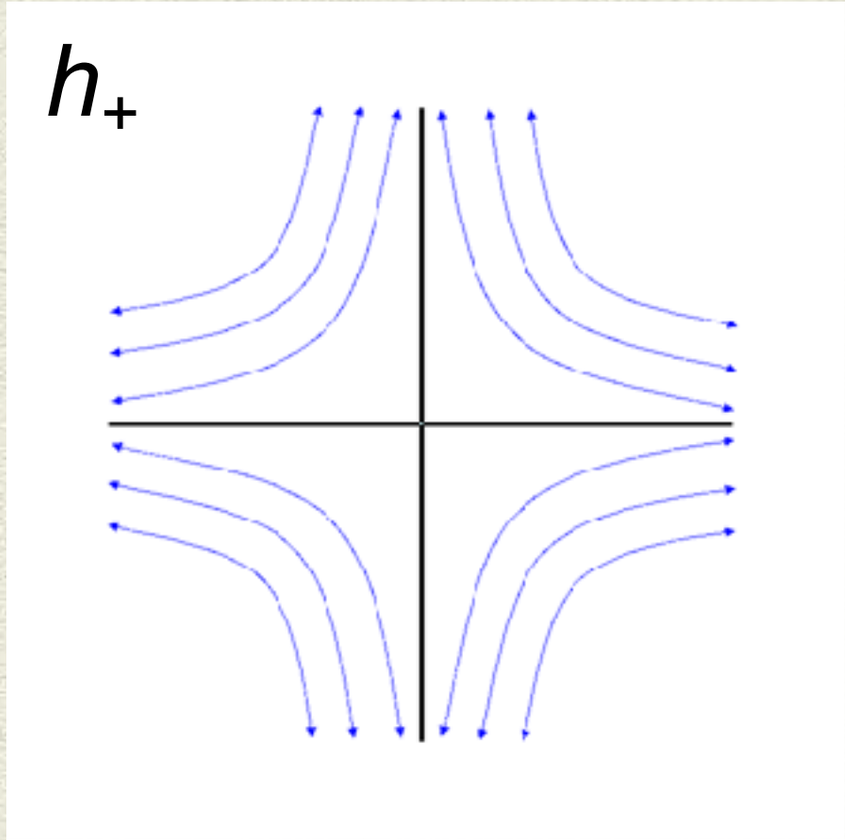
- Started to work in UNIST from mid. Dec. of 2017
- A member of CHEA (Center for High-energy Astrophysics)
- A member of FAN (Fermi Asian Network)
- A member of DiFX software corrector
- A member of KAGRA Facilities Council (KFC)
- was a member of GLT (Greenland Telescope) in ASIAA of Taiwan
- was a assistant Prof. of CMU in Taiwan
- was a PostDoc. Of NCU of Taiwan
- Expertise:
  - Timing analysis, high-energy observations, signal processing, compact source and galactic dynamics.

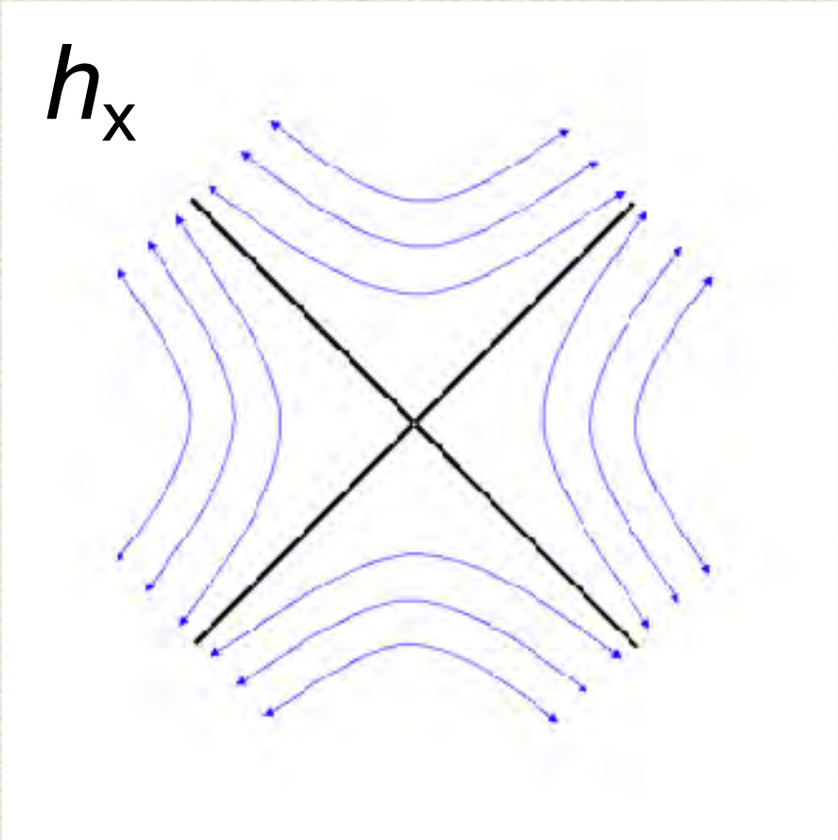
Gravitational Waves (GWs)

Gravitational waves are *propagating solutions* to the Einstein Field Equations of General Relativity ➜ solutions *h*(*r*,*t*) ➜ dynamic space-time!

$$h_{\mu\nu} \approx \frac{1}{r}\frac{G}{c^4}\ddot{I}_{\mu\nu}$$

transverse to the propagation direction of the gravitational wave





Physically, *h* is a *strain*: ΔL/L

Gravitational Waves (GWs)

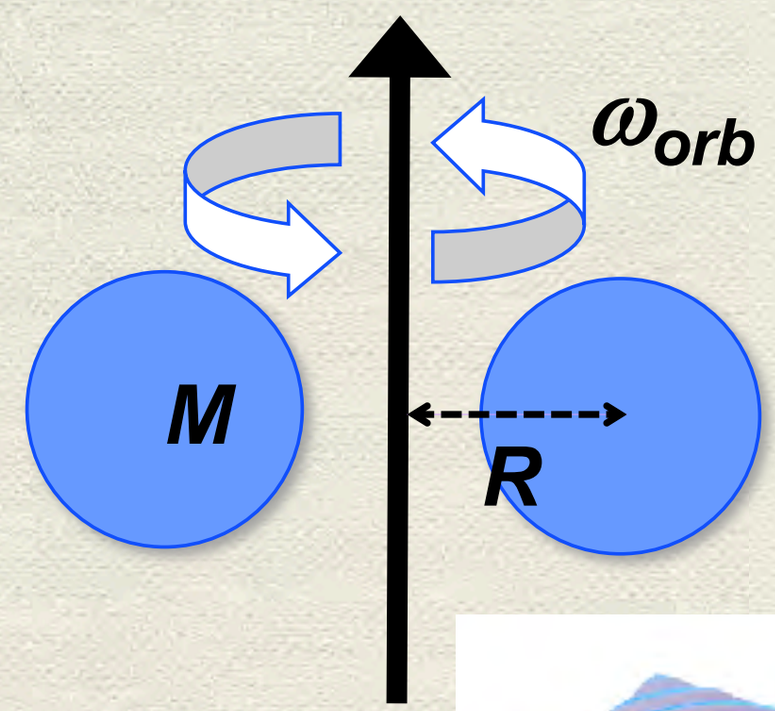
Gravitational waves are *propagating solutions* to the Einstein Field Equations of General Relativity ➜ solutions *h*(*r*,*t*) ➜ dynamic space-time!

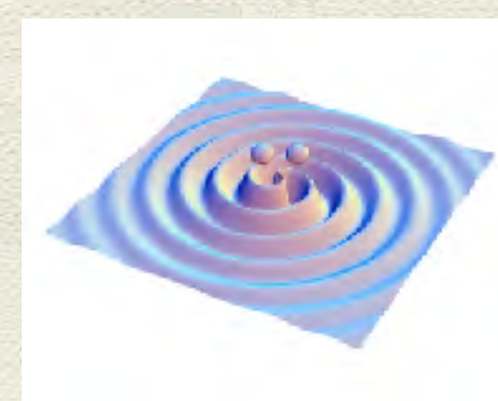
$$h_{\mu\nu} \approx \frac{1}{r}\frac{G}{c^4}\ddot{I}_{\mu\nu}$$

transverse to the propagation direction of the gravitational wave

$10M_\odot$ Binary Black Hole System

$$h \approx \frac{8GMR^2\omega_{orb}^2}{rc^4} \sim 10^{-21}$$





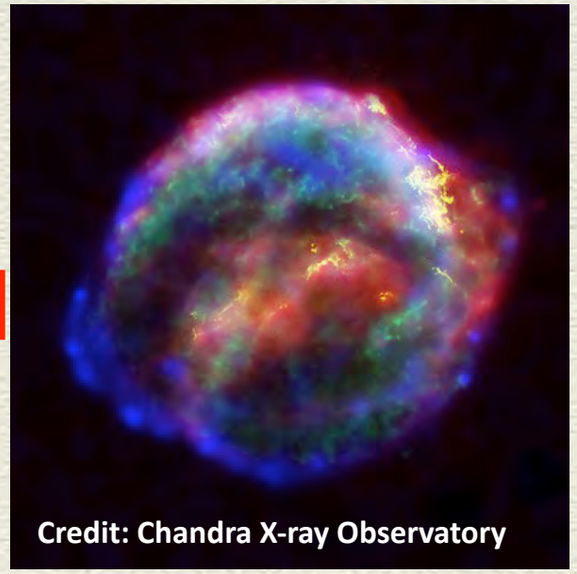
Physically, *h* is a *strain*: ΔL/L

Potential GW sources

Credit: Bohn, Hébert, Throwe, SXS

Coalescing Binary Systems

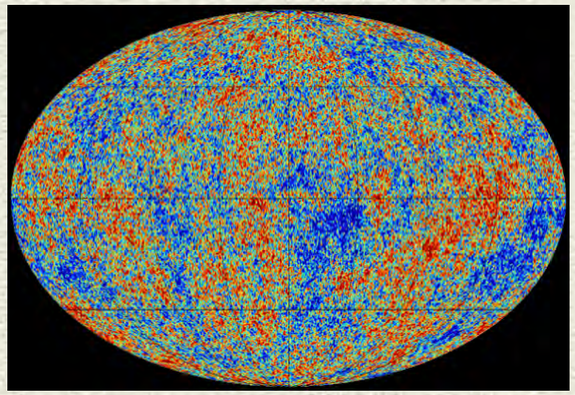
- Black hole – black hole
- Black hole – neutron star
- Neutron star – neutron star



Credit: Chandra X-ray Observatory

Transient 'Burst' Sources

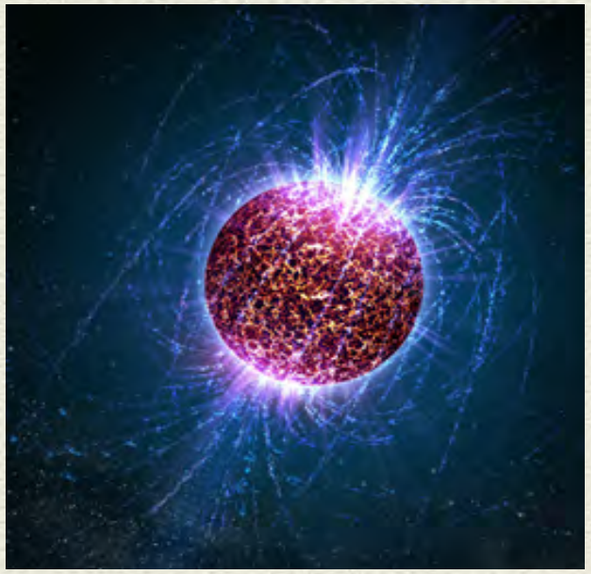
- asymmetric core collapse supernovae
- cosmic strings
- ???



Credit: Planck Collaboration

Stochastic Background

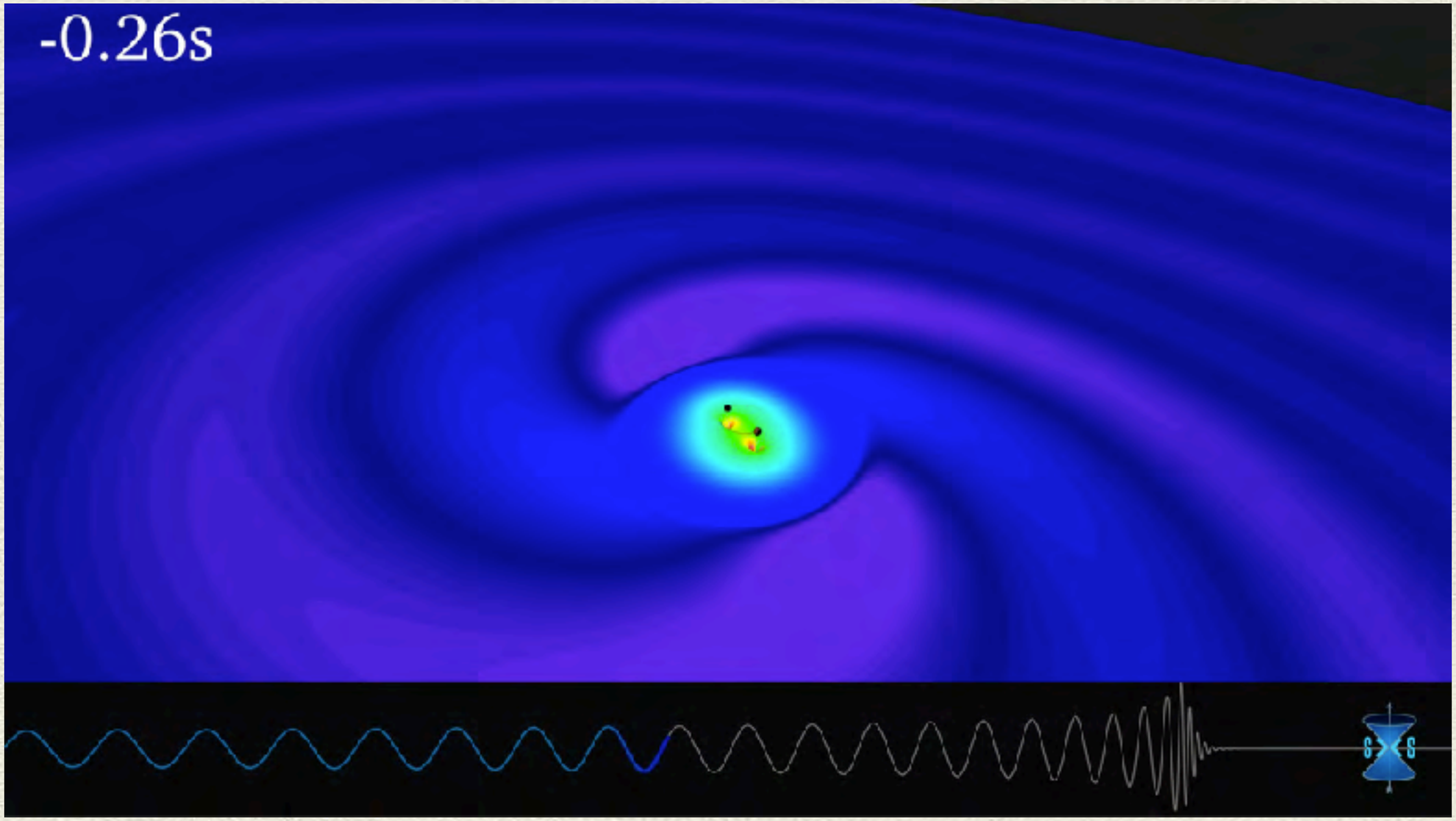
- residue of the Big Bang
- incoherent sum of unresolved 'point' sources



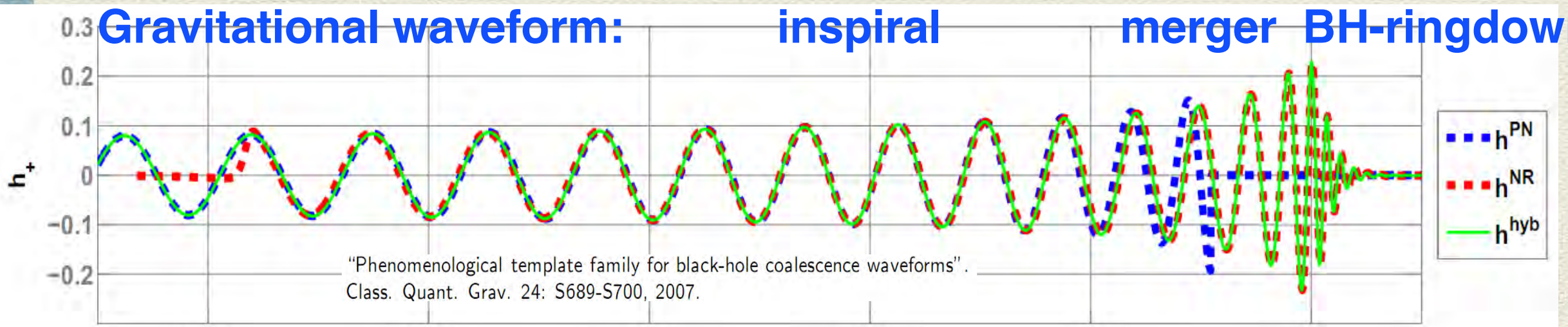
Continuous Sources

- Spinning neutron stars

Gravitational Wave from coalescing compact binaries

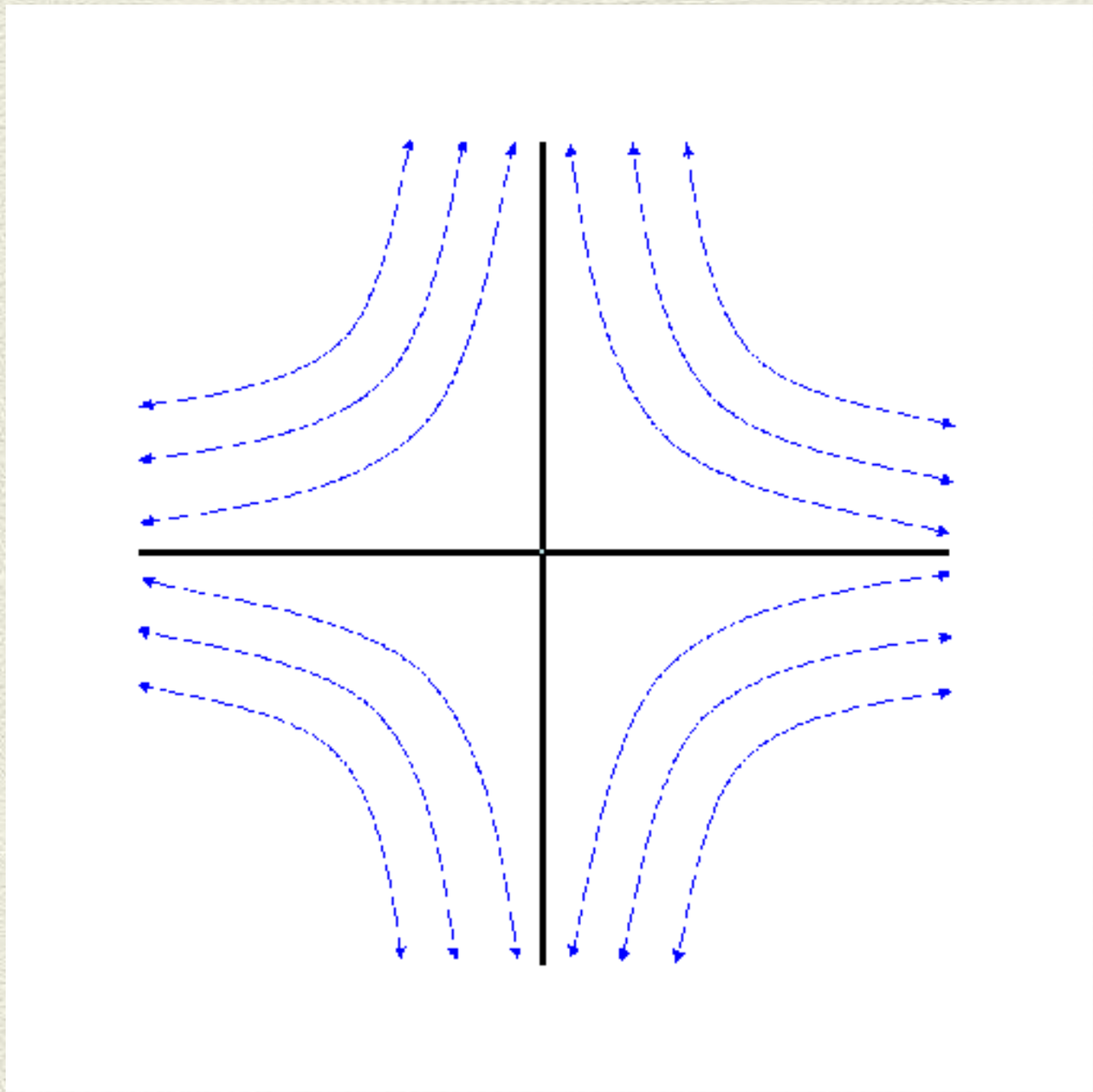


Gravitational waveform: inspiral merger BH-ringdown

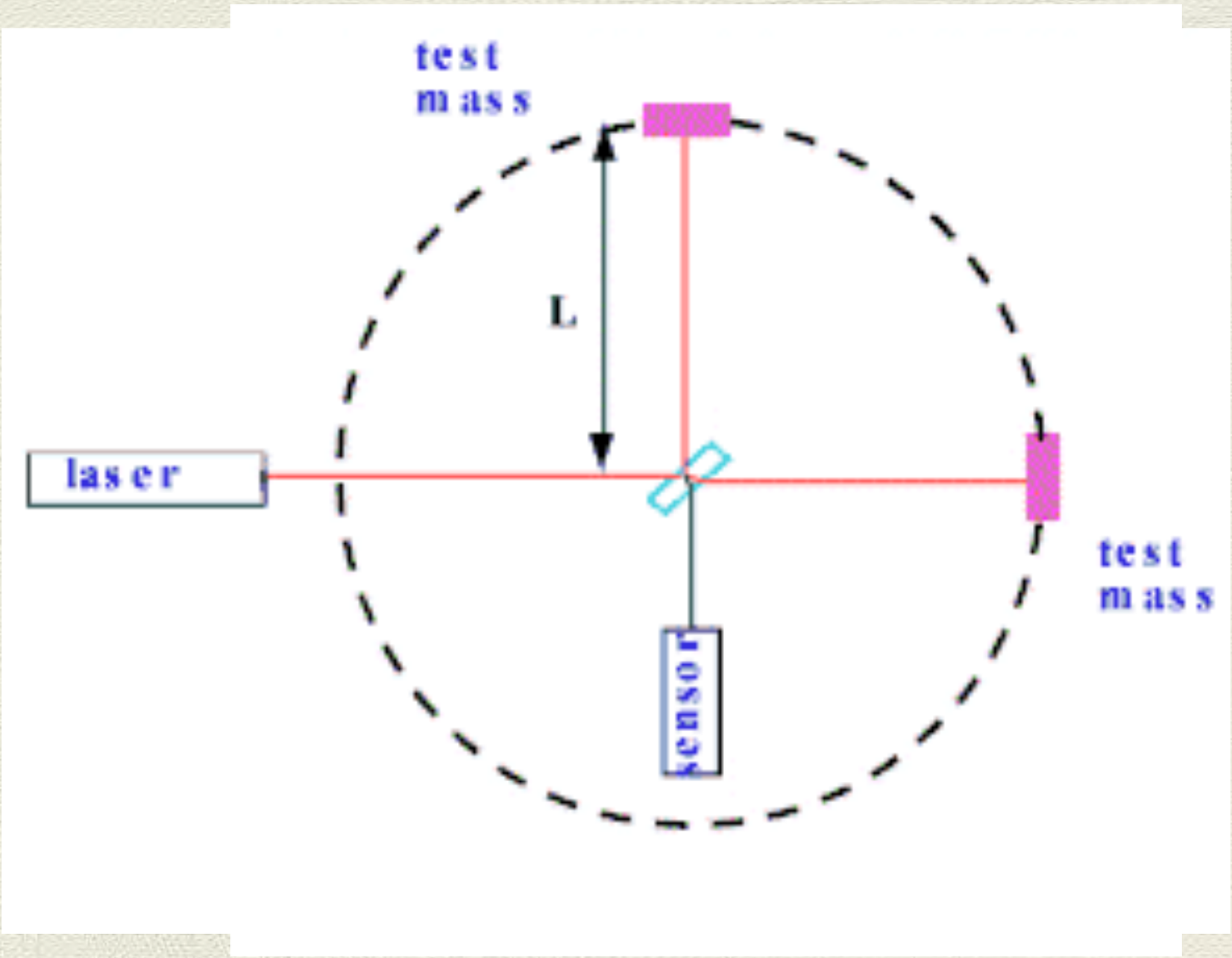


Waveform carries lots of information about binary masses, orbit, merger

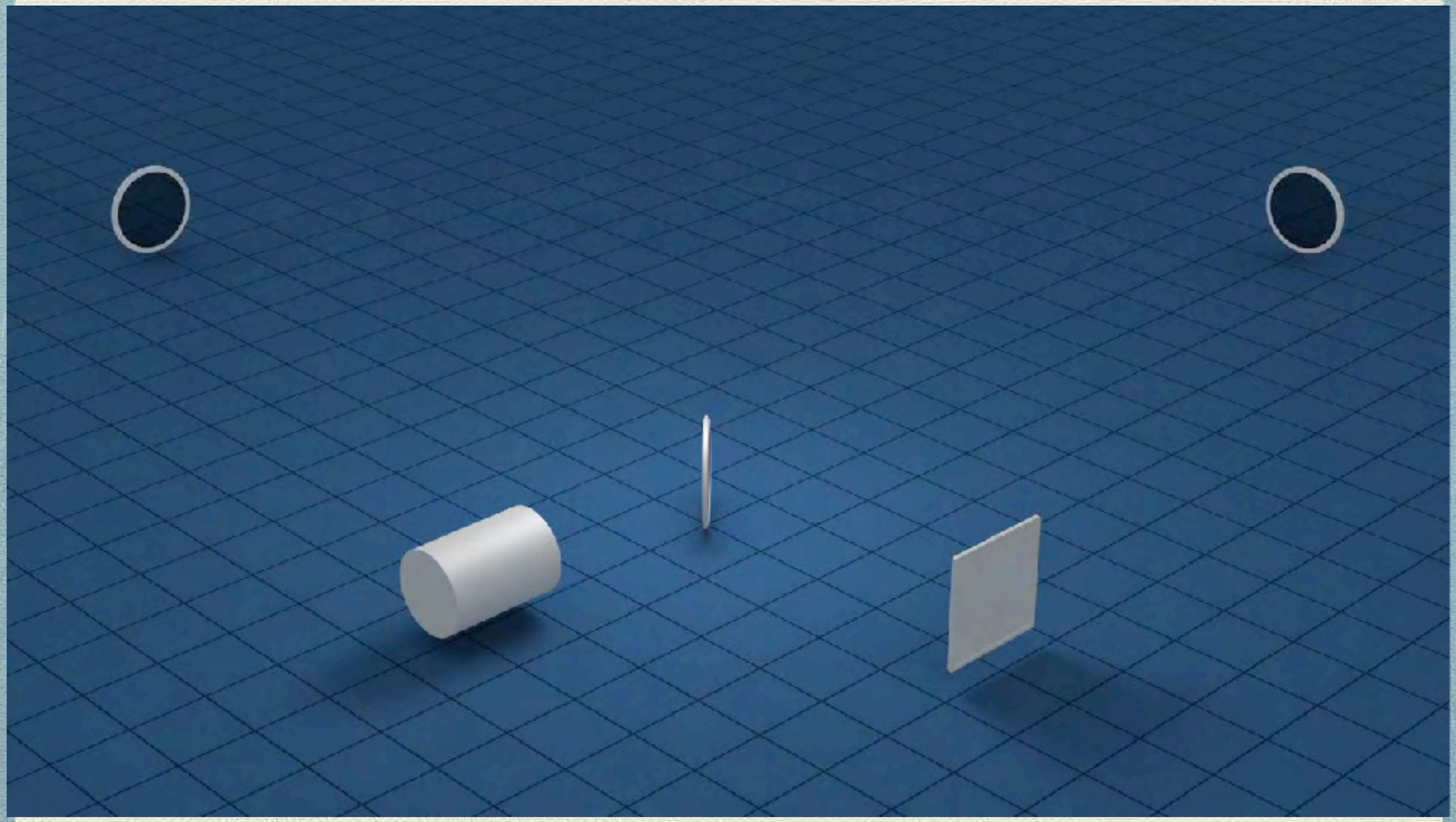
Interferometer: A Gravitational Wave Transducer



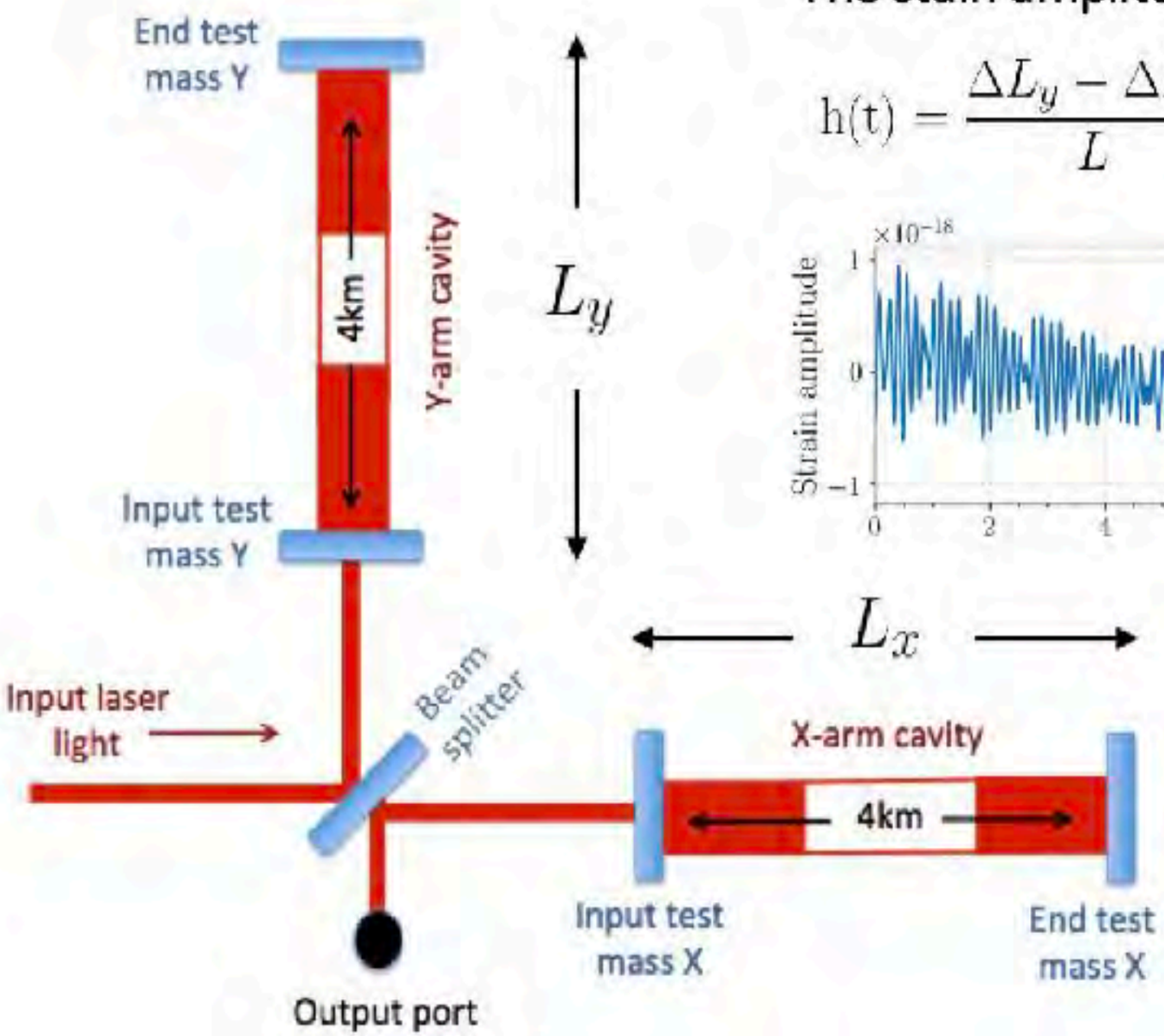
Interferometer: A Gravitational Wave Transducer



Interferometer: A Gravitational Wave Transducer



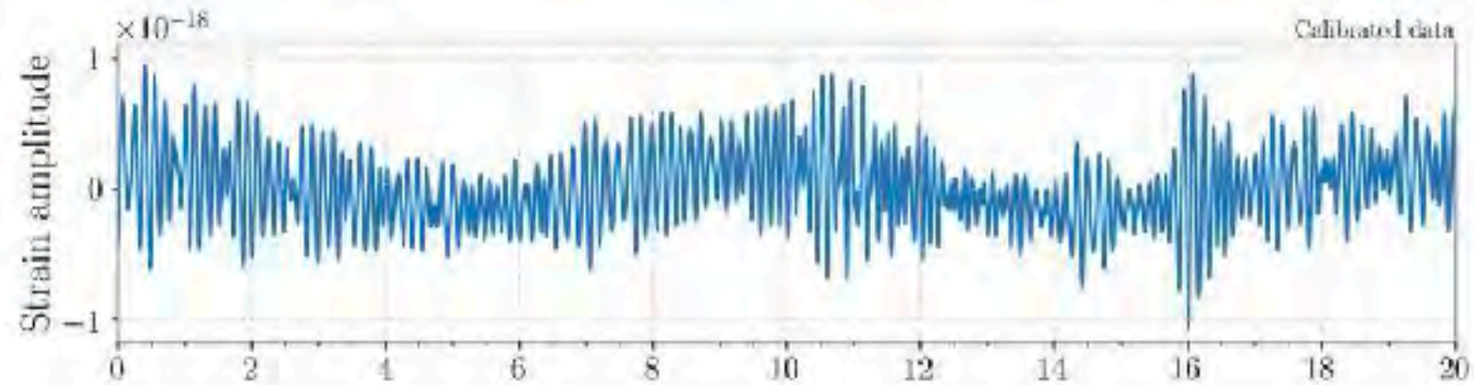
To measure the time series



The stain amplitude can be determine as:

$$h(t) = \frac{\Delta L_y - \Delta L_x}{L}$$

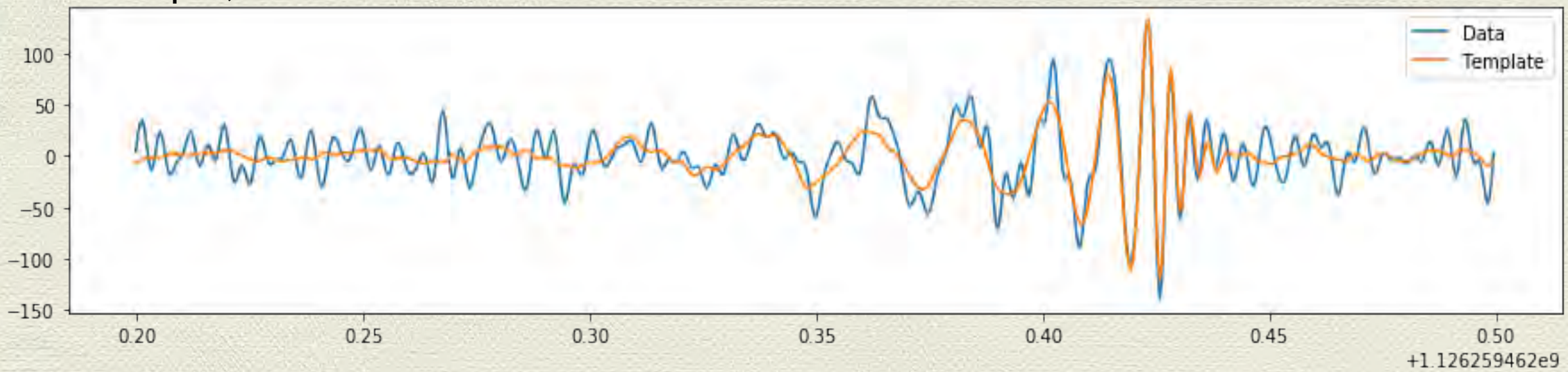
And the time series:



Matched filtering

A matched filter is obtained by correlating a known signal or template, with an unknown time series to detect the presence of the template in the unknown signal.

For example,



Matched-filter SNR:

$$\rho^2[\Delta t] = \frac{(s|h)^2[\Delta t] + i(s|h)^2[\Delta t]}{(h|h)}$$

where s is the data and h is the noise-free gravitational-wave template.

And

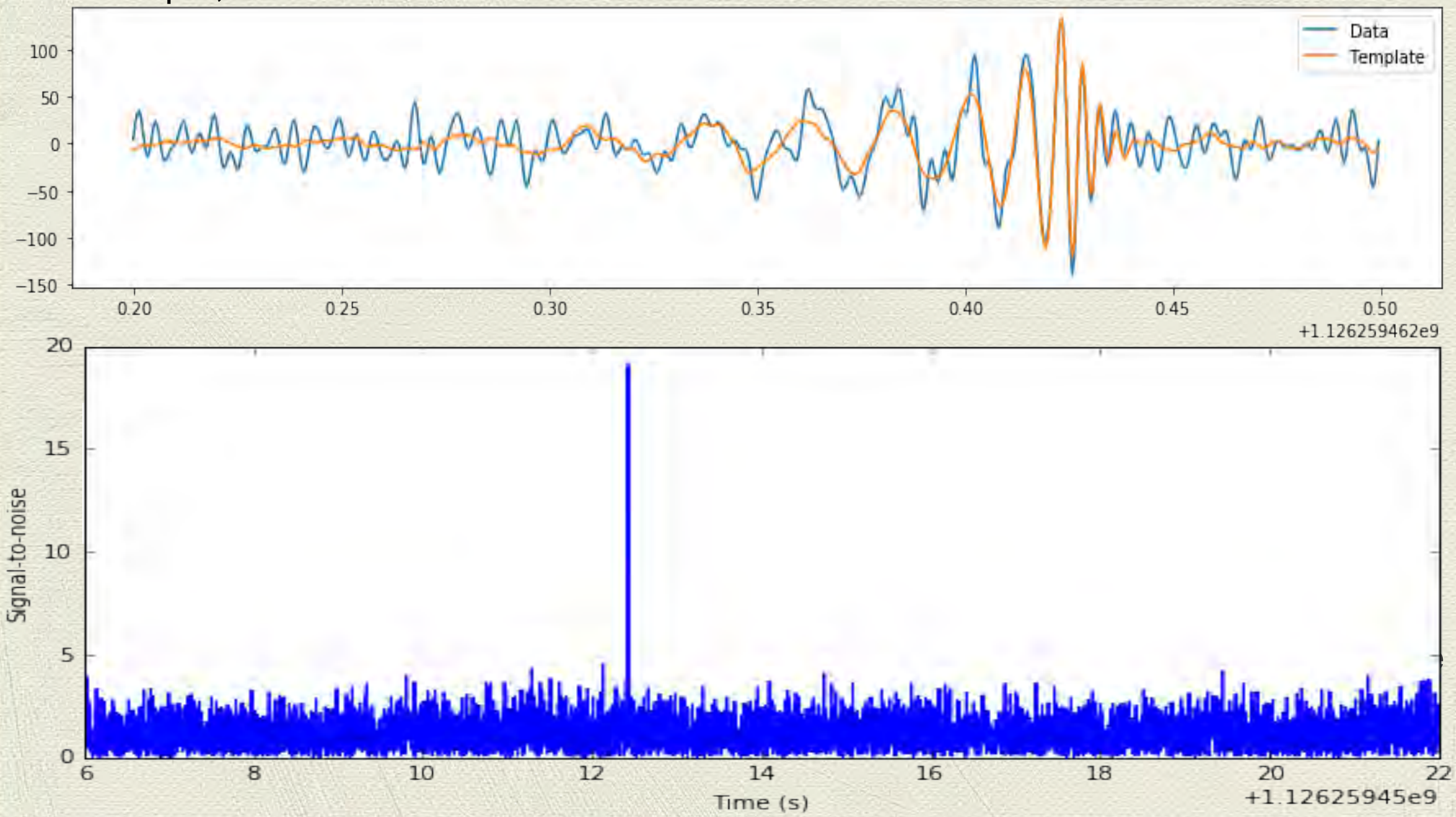
$$(a|b)[\Delta t] = 4\int_{f_{\min}}^{\infty} \frac{\tilde{a}(f)\tilde{b}^*(f)}{S_n(f)} e^{2\pi i f \Delta t} df$$

where Δt between the arrival time of the signal and the template

Matched filtering

A matched filter is obtained by correlating a known signal or template, with an unknown time series to detect the presence of the template in the unknown signal.

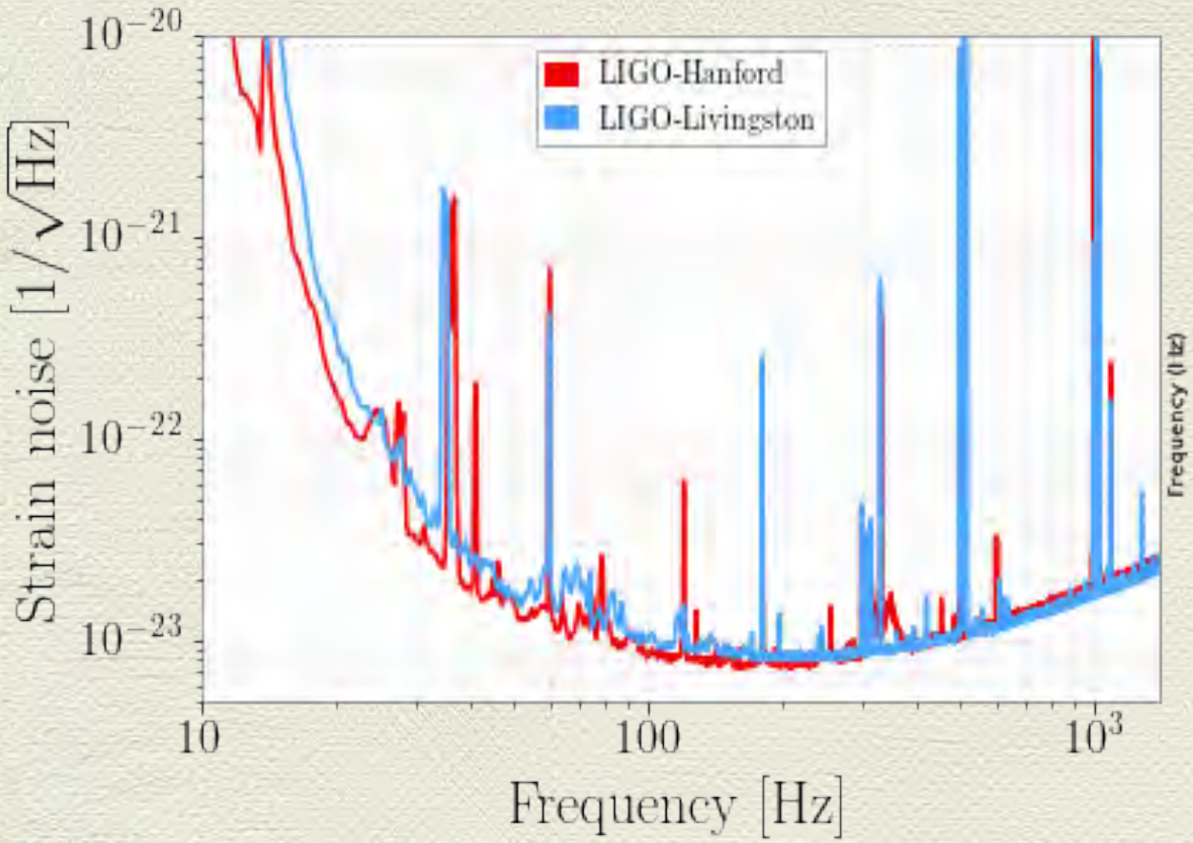
For example,



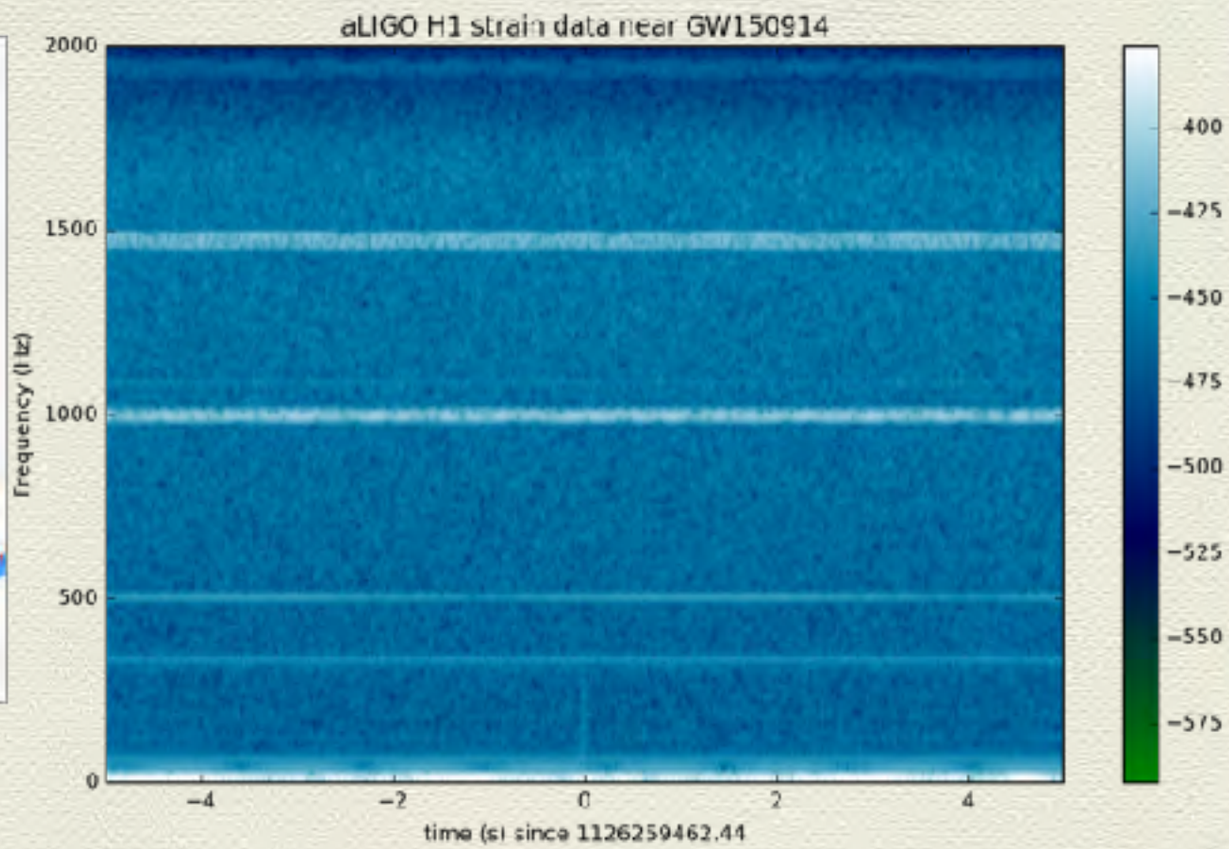
We can find a signal at 1126259462.42s with SNR 19.1247029698.

Serious contamination on the GW signal

For example, **GW150914**



Background noise to the LIGO data.

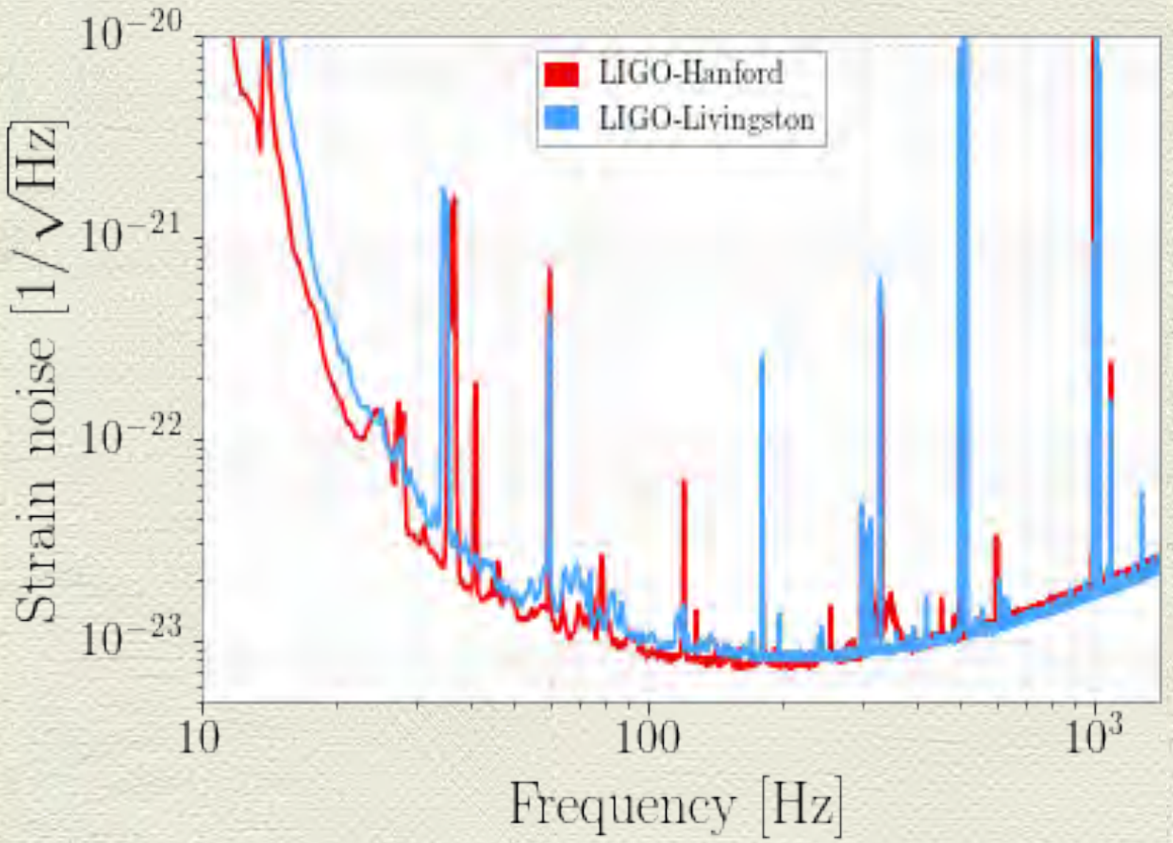


Time evolution FFT on the LIGO data.

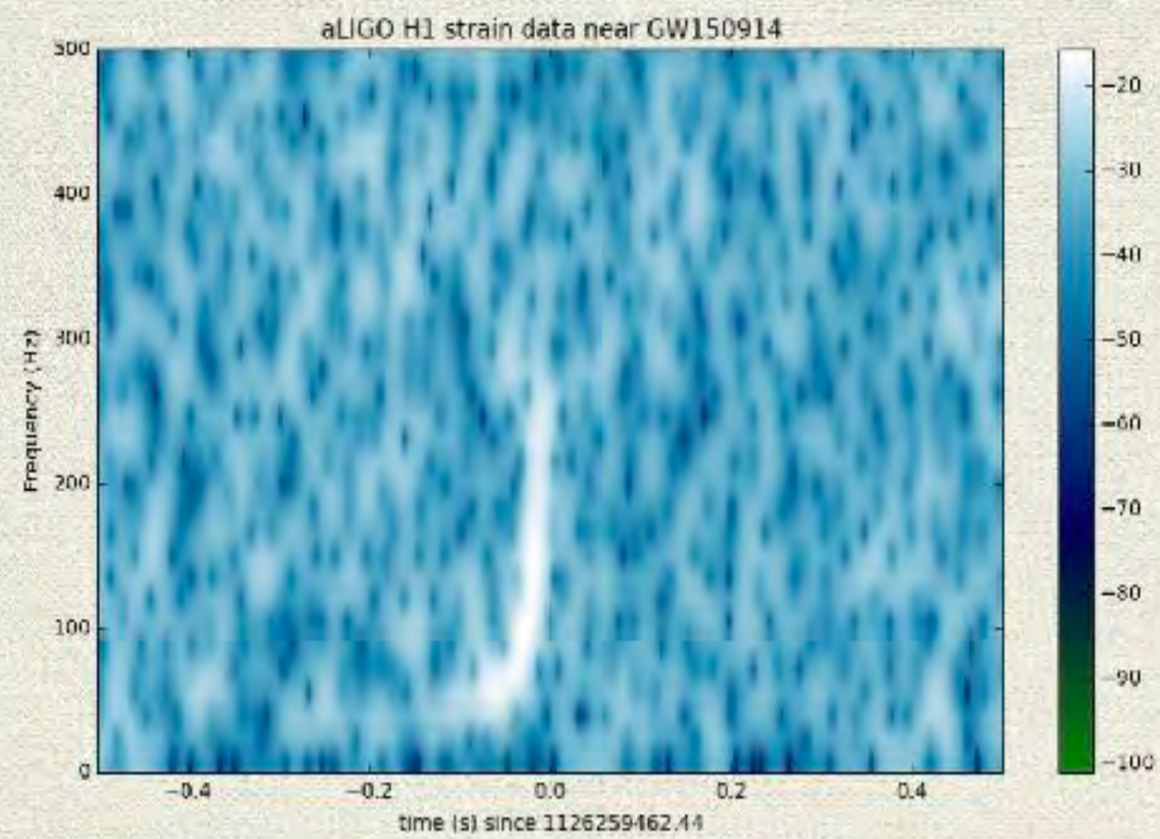
The signal can only be resolved when the strong noises were totally removed.

Serious contamination on the GW signal

For example, **GW150914**



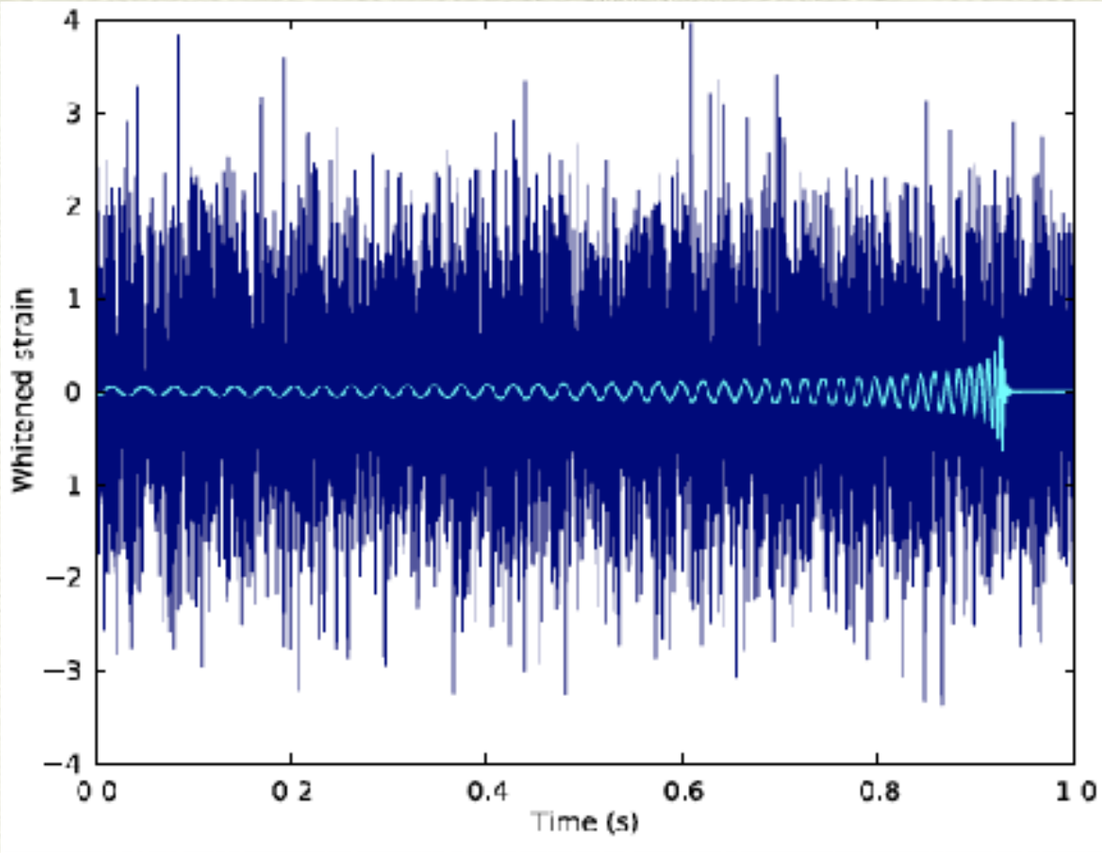
Background noise to the LIGO data.



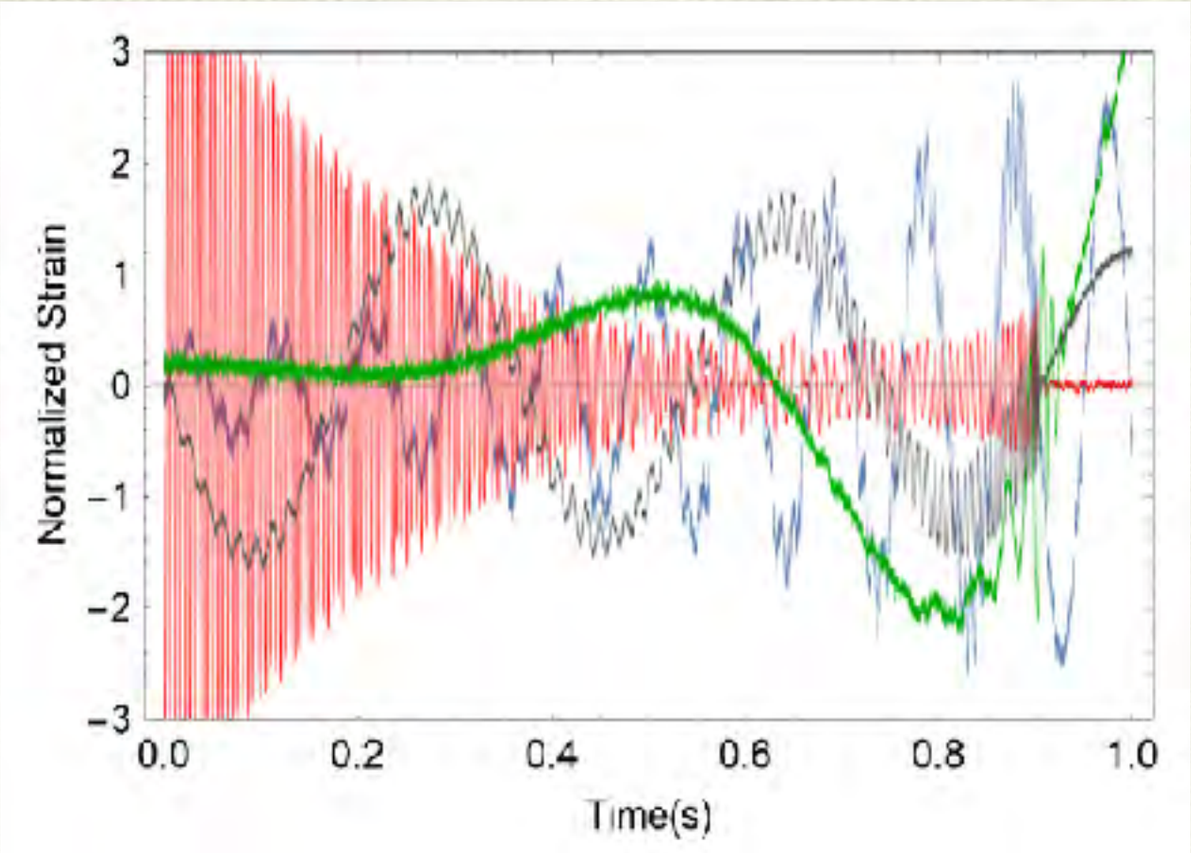
Time evolution FFT on the LIGO data.

The signal can only be resolved when the strong noises were totally removed.

The difficulty in traditional matched filtering



A whitened noise-free time series of a BBH signal with optimal SNR = 8 (cyan). The dark blue time series shows the same gravitational-wave signal with additive whitened Gaussian noise of unit variance (Gabbard et al., 2018).



Detecting signals contaminated by glitches. These are some of the signals in the test set injected into real LIGO noise and superimposed with simulated sine-Gaussian glitches from the test set. (George & Huerta, 2018).

Data sets in Deep Learning

Two cases of simulated data sets were applied in a clean comparison between the deep learning approach and matched filtering:

I. BBH merger signals in additive Gaussian noise.
II. Gaussian noise alone

Signals are simulated using **LALSuite** routines, and the **IMRPhenomD-type** waveform was used to model the inspiral, merger and the ring-down components of GW signals. The systems were simulated with BH mass in the range from 5 to 95 $M_\odot$, $m_1 > m_2$, with zero spin, and each signal was given a random R.A. and decl.

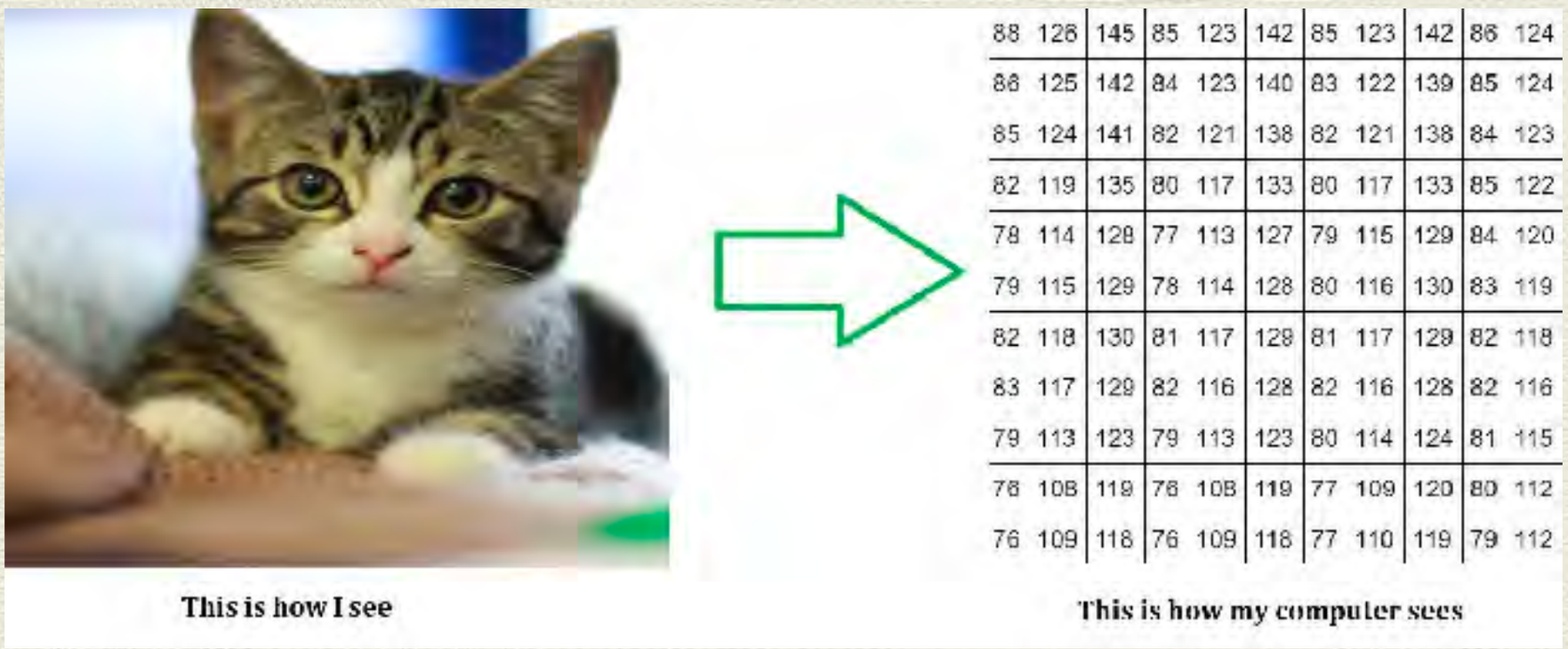
Data sets

- Training : Data samples that the network learns from
- Validation : Developer verified that the network is learning correctly
- Testing : Data use to quantify the performance of the trained network

Deep Learning Algorithm

A deep learning algorithm is composed of stacked arrays of processing units, called neurons, which can be from one to several layers deep.
A neuron acts as a filter, whereby it performs a transformation on an array of inputs. This transformation is a linear operation between the input array, and the weight and bias parameters associated with the neuron.

An ordinary neural networks:



88	126	145	85	123	142	85	123	142	86	124
86	125	142	84	123	140	83	122	139	85	124
85	124	141	82	121	138	82	121	138	84	123
82	119	135	80	117	133	80	117	133	85	122
78	114	128	77	113	127	79	115	129	84	120
79	115	129	78	114	128	80	116	130	83	119
82	118	130	81	117	128	81	117	129	82	118
83	117	129	82	116	128	82	116	128	82	116
79	113	123	79	113	123	80	114	124	81	115
78	108	119	78	108	119	77	109	120	80	112
76	109	118	76	109	118	77	110	119	79	112

What we feed is what we get.

Deep Learning Algorithm

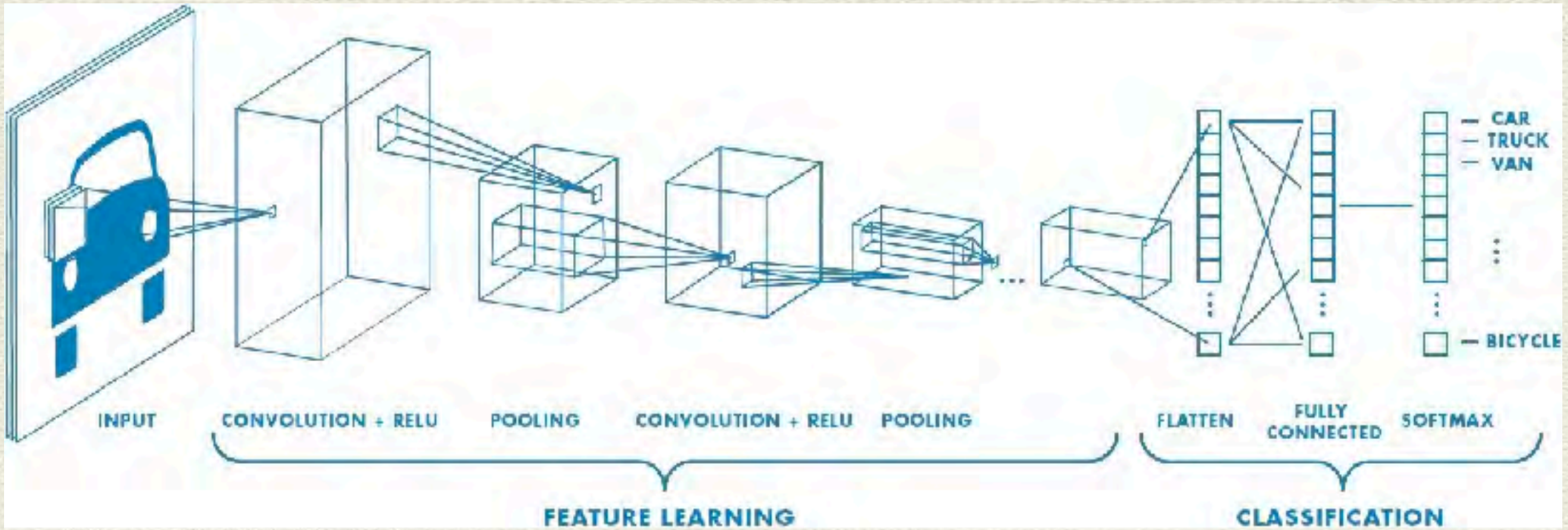
A deep learning algorithm is composed of stacked arrays of processing units, called neurons, which can be from one to several layers deep.
A neuron acts as a filter, whereby it performs a transformation on an array of inputs. This transformation is a linear operation between the input array, and the weight and bias parameters associated with the neuron.

An ordinary neural networks:

What we feed is what we get.

The network might fail to give the highest probability score as this type of features we did not train.

Convolutional Neural Network (CNN)

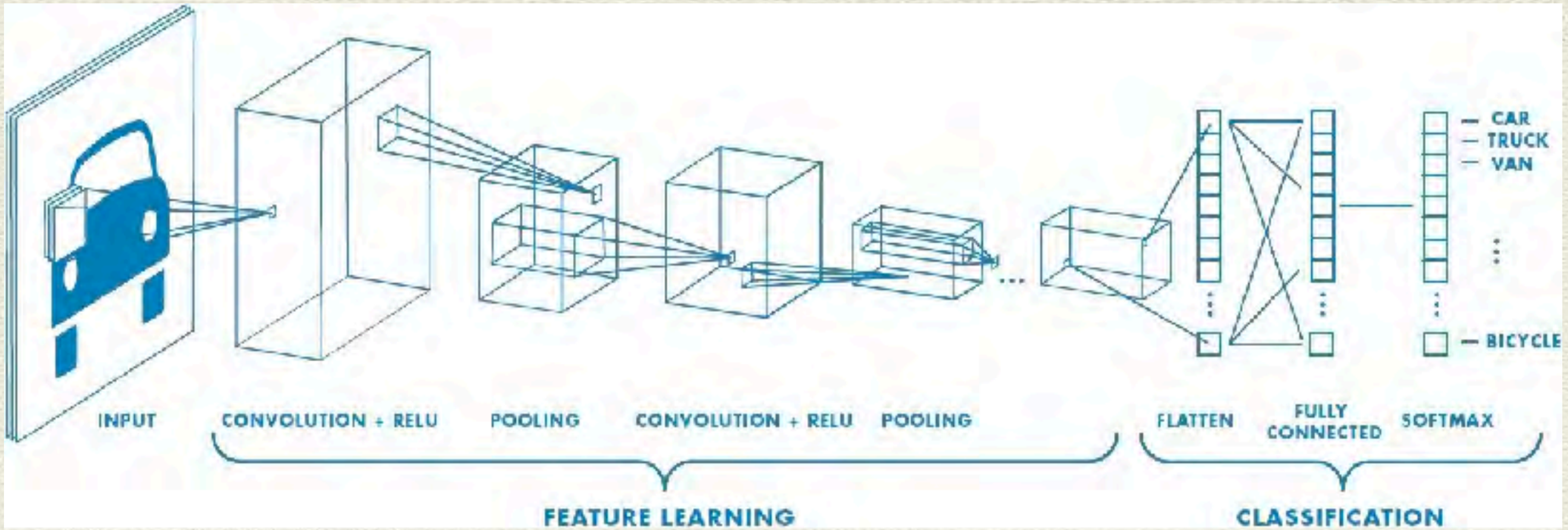


Type of layers:

1. **Convolution layer** where the convolution process happens.
2. **Normalization layer** where the activation (ReLu) process happens.
3. **Pooling layer** where the pooling process happens.
4. **Fully Connected layer**.

Process:

Convolutional Neural Network (CNN)

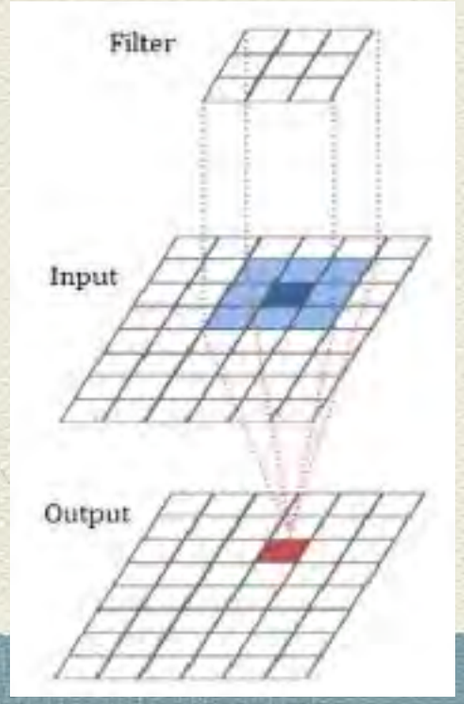


Type of layers:

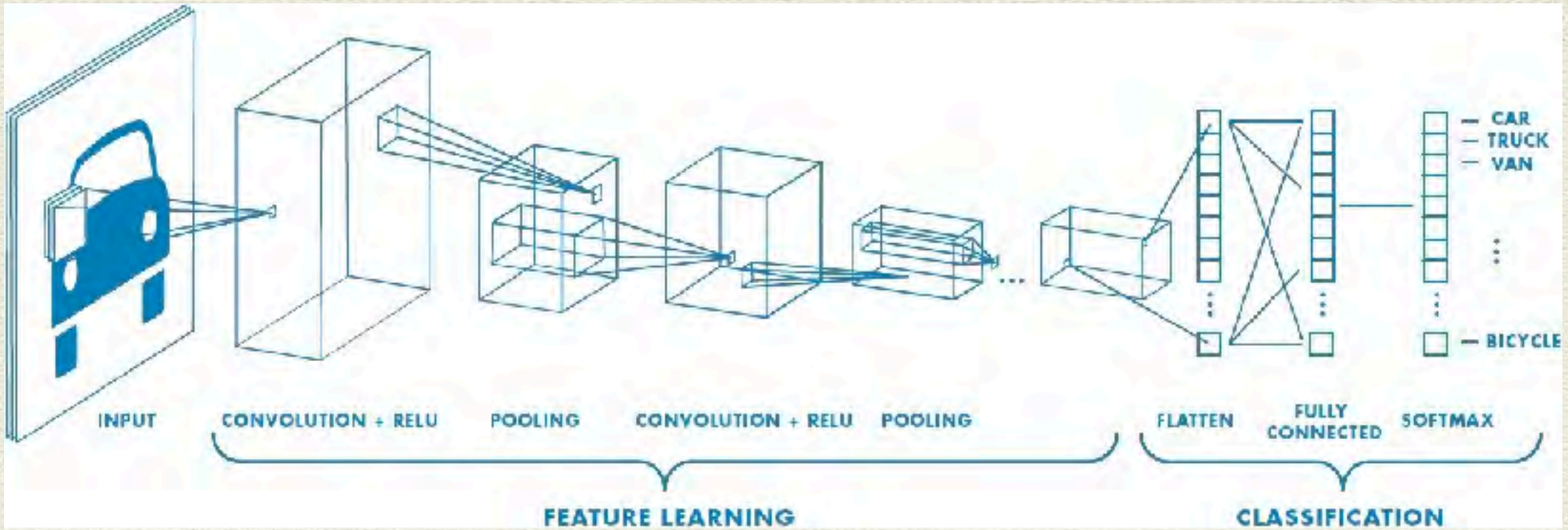
1. **Convolution layer** where the convolution process happens.
2. **Normalization layer** where the activation (ReLu) process happens.
3. **Pooling layer** where the pooling process happens.
4. **Fully Connected layer**.

Process:

1. Apply Convolution to all the filters for the input image.



Convolutional Neural Network (CNN)

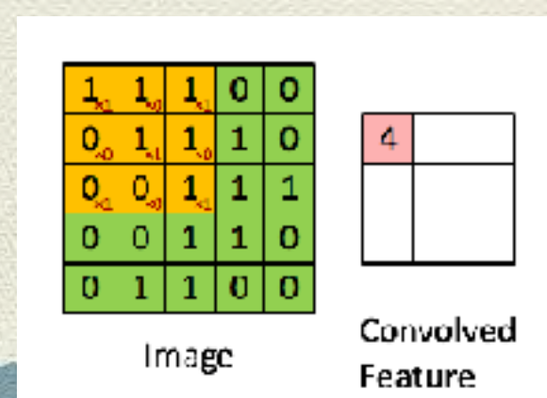


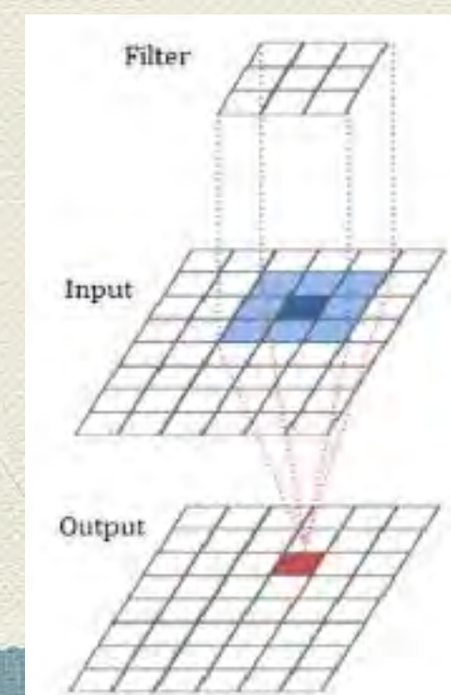
Type of layers:

1. **Convolution layer** where the convolution process happens.
2. **Normalization layer** where the activation (ReLu) process happens.
3. **Pooling layer** where the pooling process happens.
4. **Fully Connected layer**.

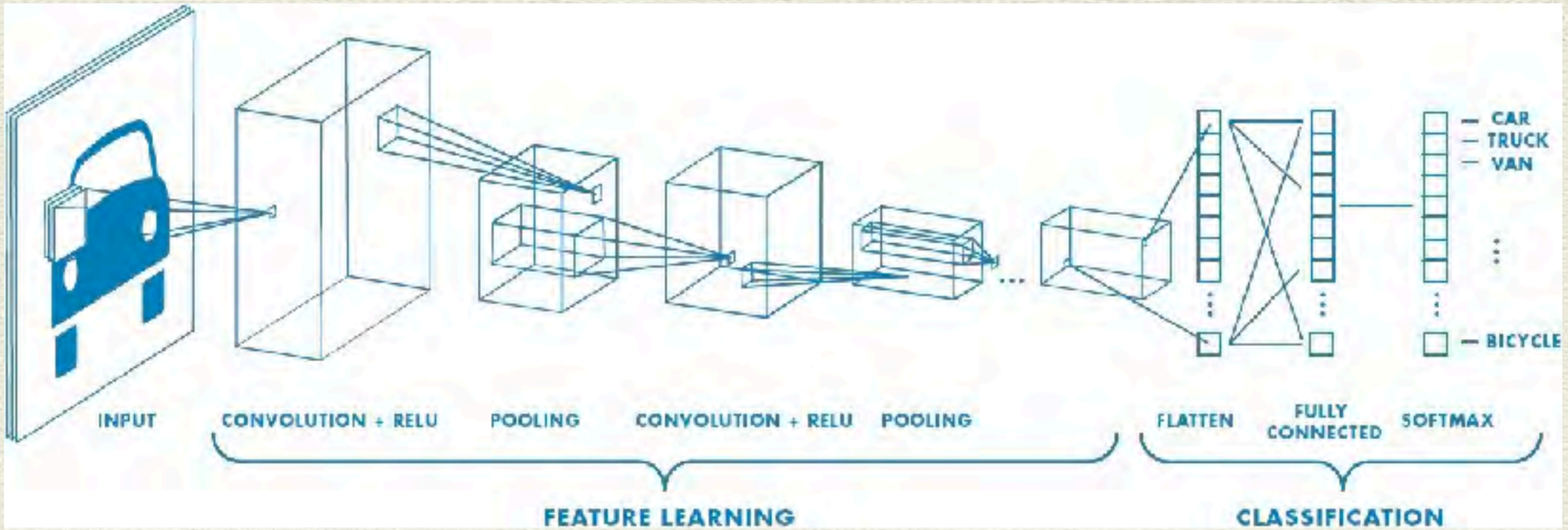
Process:

1. Apply Convolution to all the filters for the input image.





Convolutional Neural Network (CNN)



Type of layers:

1. **Convolution layer** where the convolution process happens.
2. **Normalization layer** where the activation (ReLu) process happens.
3. **Pooling layer** where the pooling process happens.
4. **Fully Connected layer**.

Process:

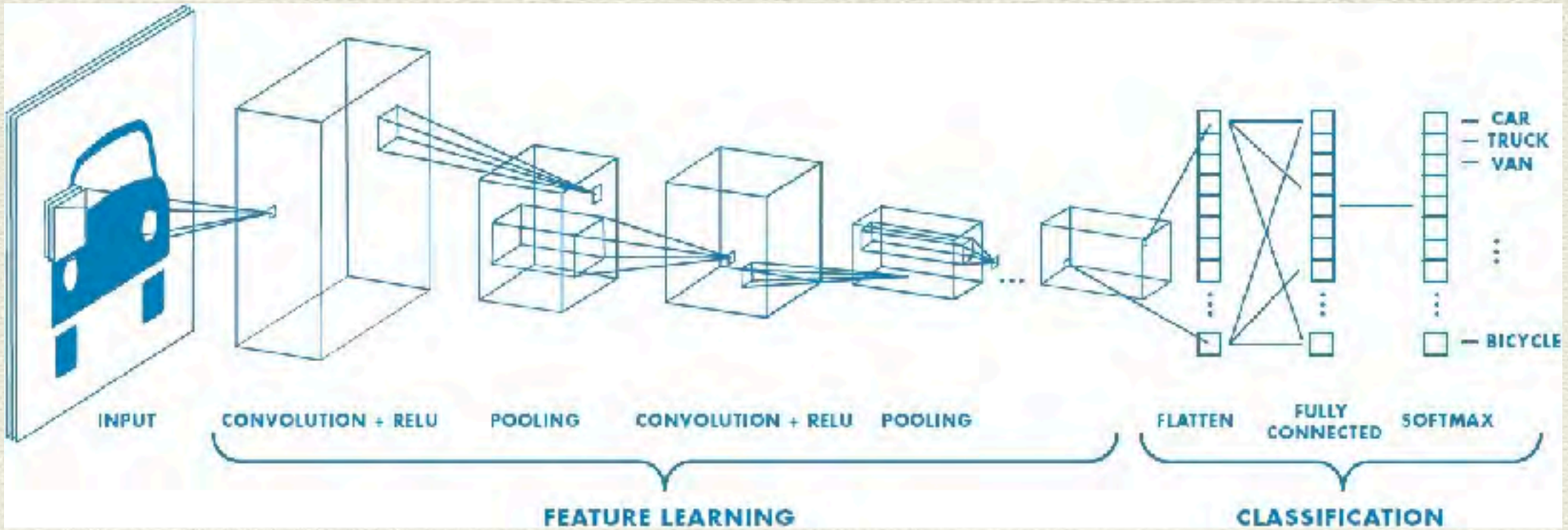
2. Normalization, this is the step when we apply an activation function, and the most used function here is ReLu (Rectified linear unit).

$$RELU(x) = \begin{cases} 0 \; if \; x < 0 \\ x \; if \; x >= 0 \end{cases}$$

-478	494
460	-477

0	494
460	0

Convolutional Neural Network (CNN)

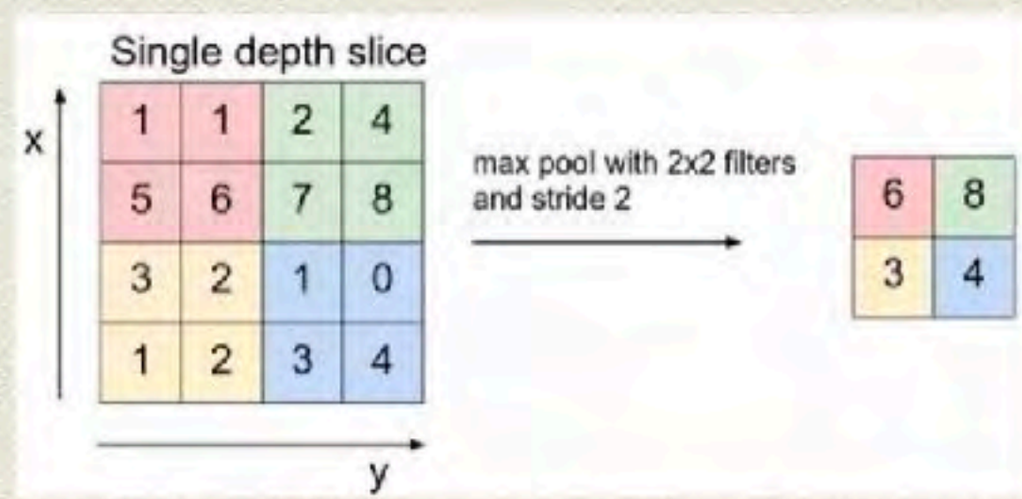


Type of layers:

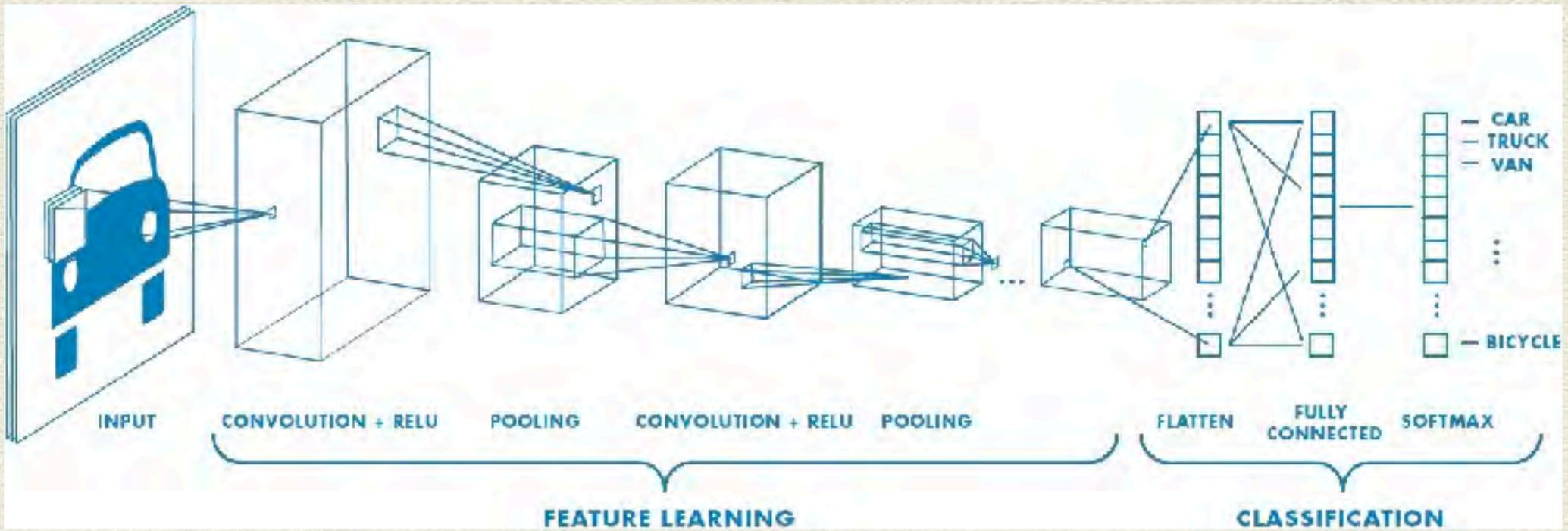
1. **Convolution layer** where the convolution process happens.
2. **Normalization layer** where the activation (ReLu) process happens.
3. **Pooling layer** where the pooling process happens.
4. **Fully Connected layer**.

Process:

3. Apply Pooling concept for the output images. For example, the max pooling:



Convolutional Neural Network (CNN)

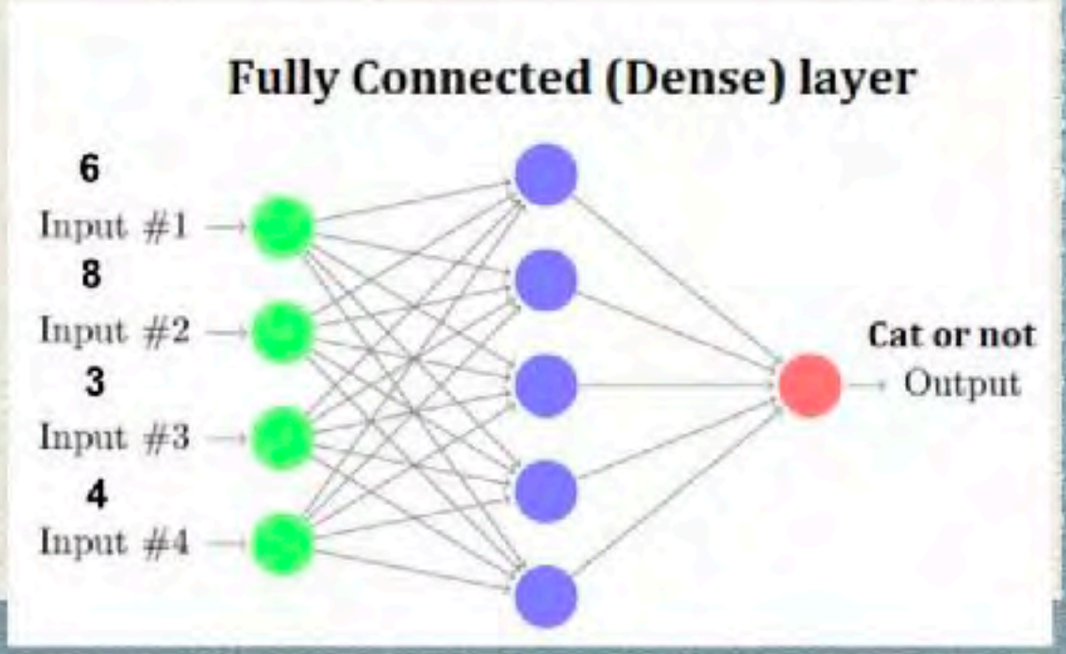


Type of layers:

1. **Convolution layer** where the convolution process happens.
2. **Normalization layer** where the activation (ReLu) process happens.
3. **Pooling layer** where the pooling process happens.
4. **Fully Connected layer**.

Process:

4. Feed these values to Fully Connected Neural network



Application of CNN

The optimized network consisting of six convolutional layers (*C*), followed by three hidden layers (*H*) (Gabbard et al., 2018).

	Layer								
Parameter (Option)	1	2	3	4	5	6	7	8	9
Type	*C*	*C*	*C*	*C*	*C*	*C*	*H*	*H*	*H*
No. Neurons	8	8	16	16	32	32	64	64	2
Filter size	64	32	32	16	16	16	Not applicable	Not applicable	Not applicable
Max pool size	Not applicable	8	Not applicable	6	Not applicable	4	Not applicable	Not applicable	Not applicable
Drop out	0	0	0	0	0	0	0.5	0.5	0
Activation function	Elu	Elu	Elu	Elu	Elu	Elu	Elu	Elu	SMax

The final ranking statistic that was extracted from the CNN analysis is taken from the last output layer, composed of two neurons, where each neuron produces a probability value between 0 and 1 with their sum being unity.

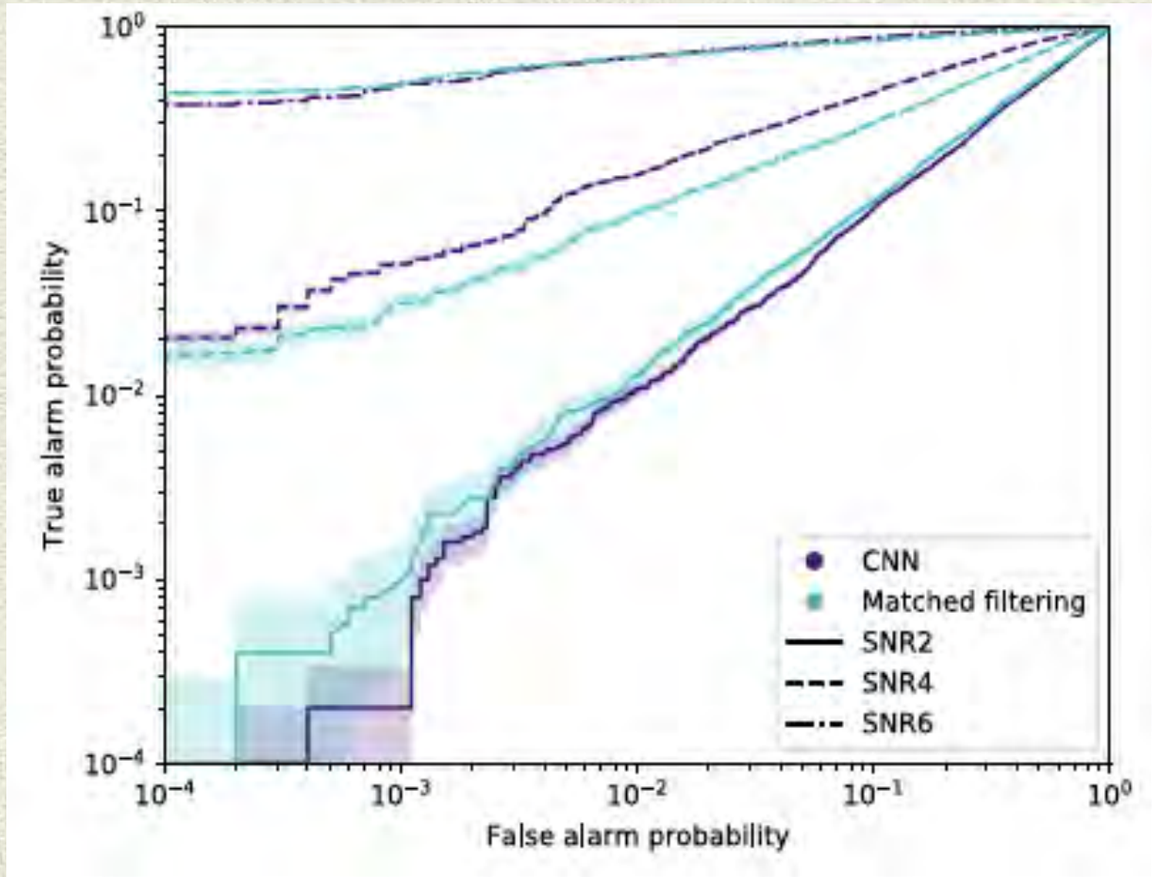
Each neuron gives the inferred probability that the input data belong to the noise or signal + noise class, respectively.

Data taking this particular analysis to the trained network can be run **10^4** times faster, but the computational time spent on training the network for each SNR case is $O(1)$ hour on a single GPU.

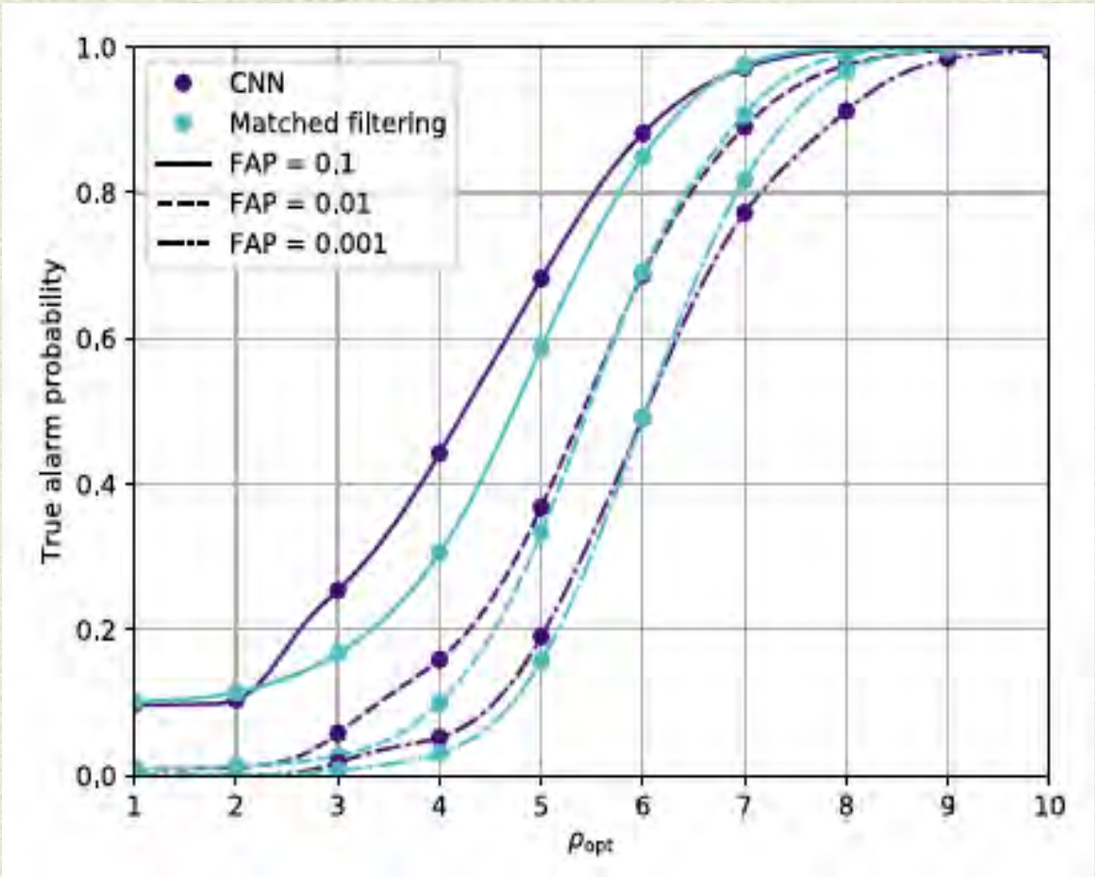
Comparison of CNN and matched filtering

To compare with templates, this paper considered a lower frequency cutoff of 20 Hz with the PyCBC geometric nonspinning template bank generation tool. This template bank contained 8056 individual templates.

Given the ranking statistic from a particular analysis and defining a parametric threshold value on that statistic, we can determine:
False alarm probability: the fraction of noise samples incorrectly identified as signals.
True alarm probability: the fraction of signal samples correctly identified.



The ROC curves for test data sets containing signals with optimal SNR.



Efficiency curves comparing the performance of the CNN and matched-filtering approaches for different false alarm probabilities.

Conclusions

1. Matched-filtering analyses are often described as the optimal approach for signal detection of GW events in Gaussian noise. However, **the deep learning algorithm (CNN), when applied to gravitational-wave time series, is able to closely reproduce the results of a matched-filtering analysis.**

2. Searches for transient signals in GW data are strongly affected by non-Gaussian noises, and the standard matched-filtering approaches need to be modified to account for this. **There exists the potential for deep networks to exceed the sensitivity of existing matched-filtering approaches in real data.**

3. The paper presented results for BBH mergers; however, **this deep learning algorithm could be applied to other merger types**, such as binary neutron star and neutron star-black hole signals. **This approach can also be extended to other well-modeled gravitational-wave targets** such as the continuous emission from rapidly rotating neutron stars. The deep learning method provide a simplest way and rapid detection confidence for parameter estimations at the output regression layer, and **it is important for GW multi-messenger astronomy**, such as the case of GW170817.

Other Similar News

https://insidehpc.com/2018/01/video-deep-learning-real-time-grav

inside HPC

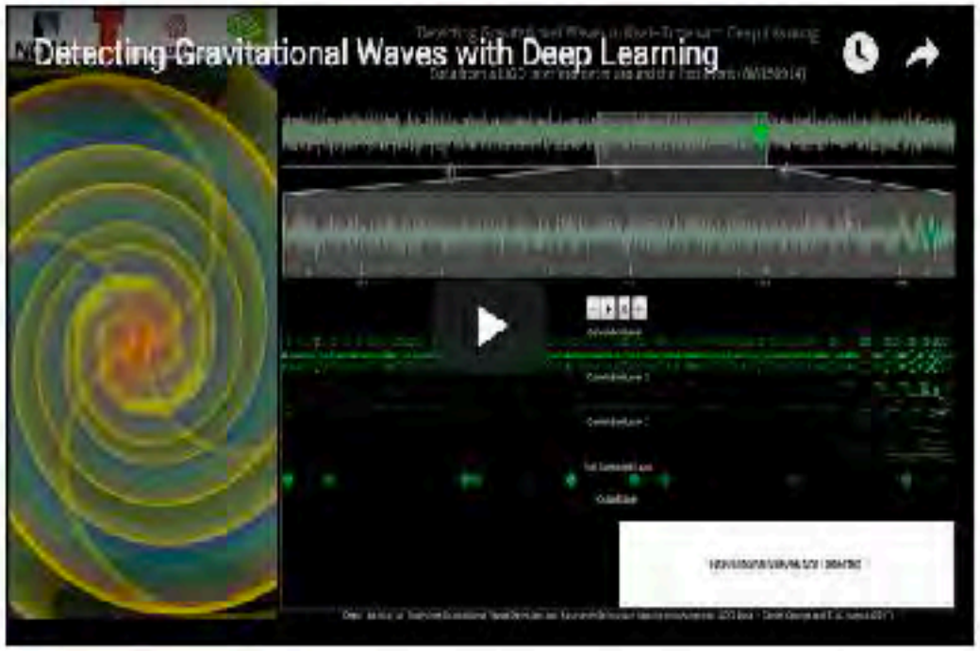
News | HPC Hardware | HPC Software | Industry Segments | White Papers | Resources | Special Reports

Sign up for our newsletter and get the latest HPC news and analysis. Email Address

Home > HPC Hardware > Video: Deep Learning for Real-Time Gravitational Wave Discovery

Video: Deep Learning for Real-Time Gravitational Wave Discovery

January 26, 2018 by staff Leave a Comment

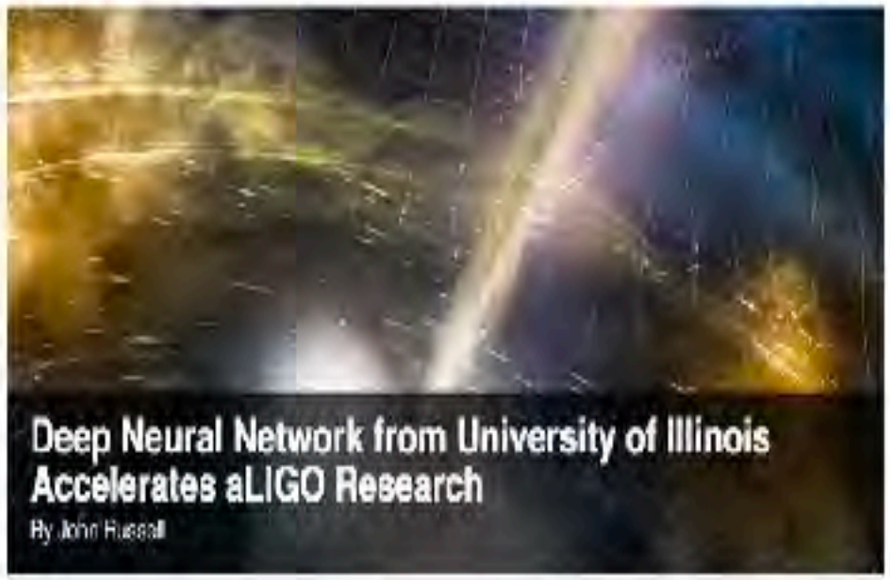


https://www.hpcwire.com/2018/03/27/deep-neural-network-from-u

TABOR NETWORK: DATANAMI | ENTERPRISETECH | HPCWIRE JAPAN | ADVANCED SCALE FORUM

HPC wire

Home
Technologies
Sectors
AI/ML/DL
Exascale
Specials
Resource Library
Events
Job Bank
About
Solution Channels
Subscribe



Deep Neural Network from University of Illinois Accelerates aLIGO Research

By John Russell

March 27, 2018

Gravitational wave astronomy burst onto the scene with the success of the original LIGO (Laser Interferometer Gravitational-Wave Observatory) effort and has since continued with the expanded Advanced LIGO (aLIGO) Project which has now identified five binary black hole mergers producing gravitational waves (GW). New deep learning tools developed at the University of Illinois Urbana-Champaign and National Center for Supercomputing Applications (NCSA) now promise to accelerate aLIGO discovery efforts.

Writing in Physical Review last month (Deep neural networks to enable real-time multimessenger astrophysics) researchers from UI and NCSA introduce Deep Filtering, new scalable machine learning method for end-to-end time-