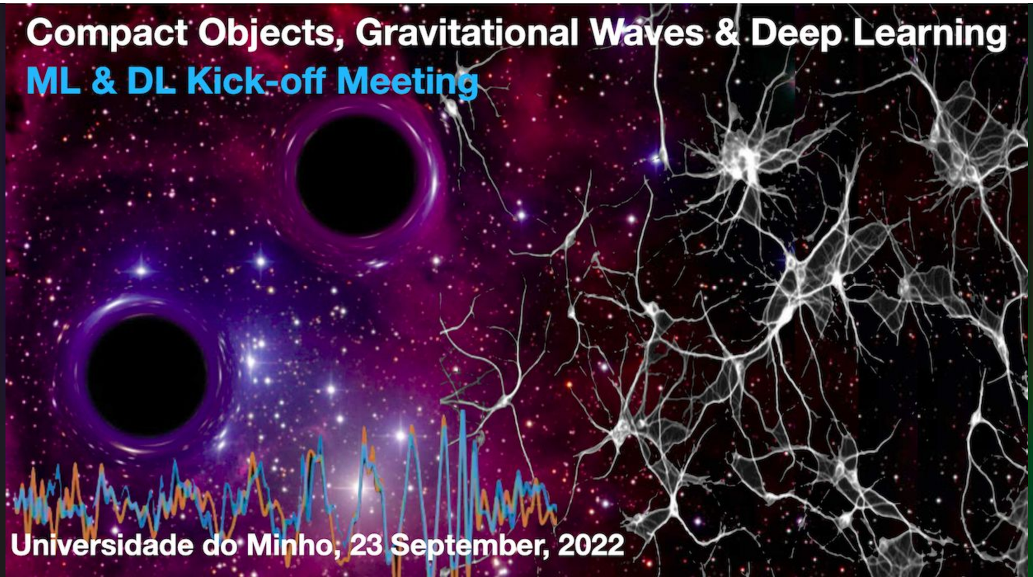
Deep Learning for the Classification of Signals and Transient Noises in the LIGO Detectors

Tiago Fernandes

Department of Physics
University of Aveiro, Portugal



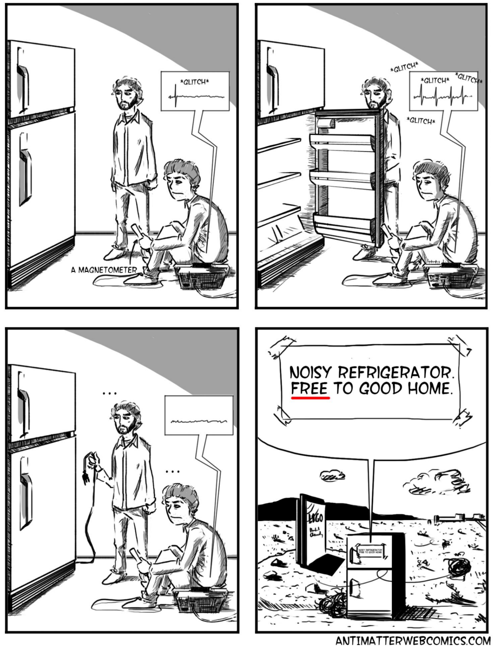
Introduction

- Measuring GWs requires very sensitive detectors. LIGO detectors are equipped with systems to minimize several noise sources.

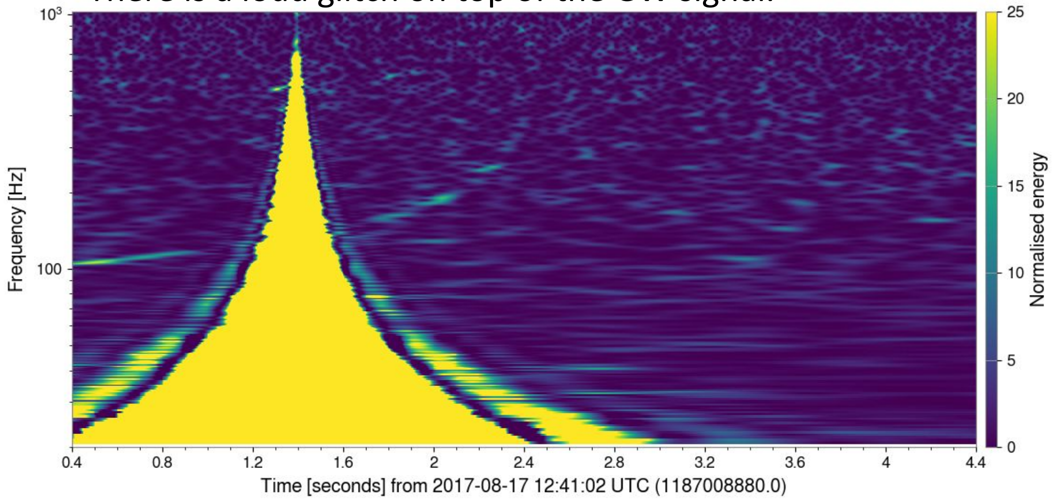
- Nevertheless, there are still noise transients, aka **glitches**, many with an unknown origin. In the last observing run, they happened at a rate of $O(1)\ \mathrm{min}^{-1}$.

- Glitches can raise false alarms or overlap with GW signals, reducing the effectiveness of the detections.
- Therefore, it is important to study the different glitches, in order to identify their causes and fix the problem.

Before the O1 run, glitches were observed at 60 Hz in LIGO-Hanford, and their rate increased as the temperature got colder.
The problem was solved when a refrigerator whose bursts of power coupled into the electronics of the interferometer was unplugged.



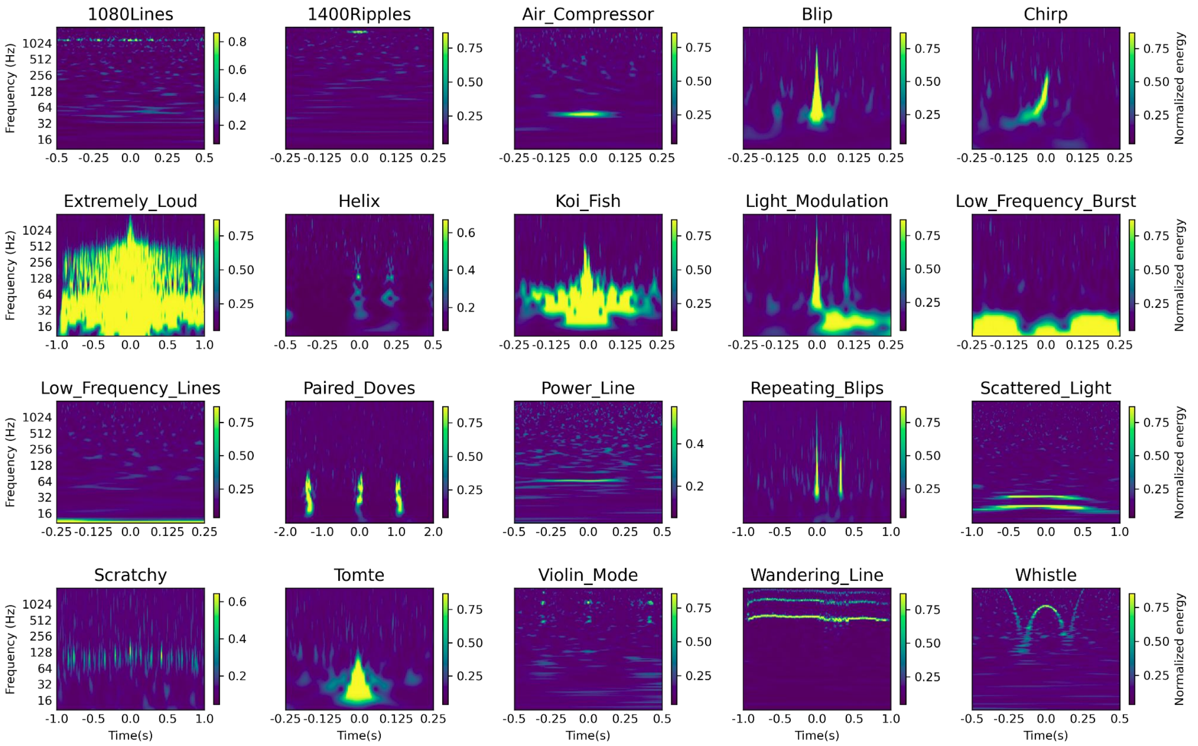
GW170817 - Livingston strain
There is a loud glitch on top of the GW signal.



Dataset

- Gravity Spy v1.0 [1, 2]:
  - 8583 samples of LIGO (O1 and O2) data;
  - each sample has 4 **spectrograms** with different durations: 0.5, 1.0, 2.0, and 4.0 seconds;
  - each sample is labelled with one of **22 classes**;
  - dataset split into train, validation and test (70/15/15).

- Almost all classes are **glitches** (noise transients), but there is also a No Glitch class and a **Chirp** class, which is made of hardware injections.
- Gravity Spy is an **imbalanced** dataset, which can be problematic for DL models.



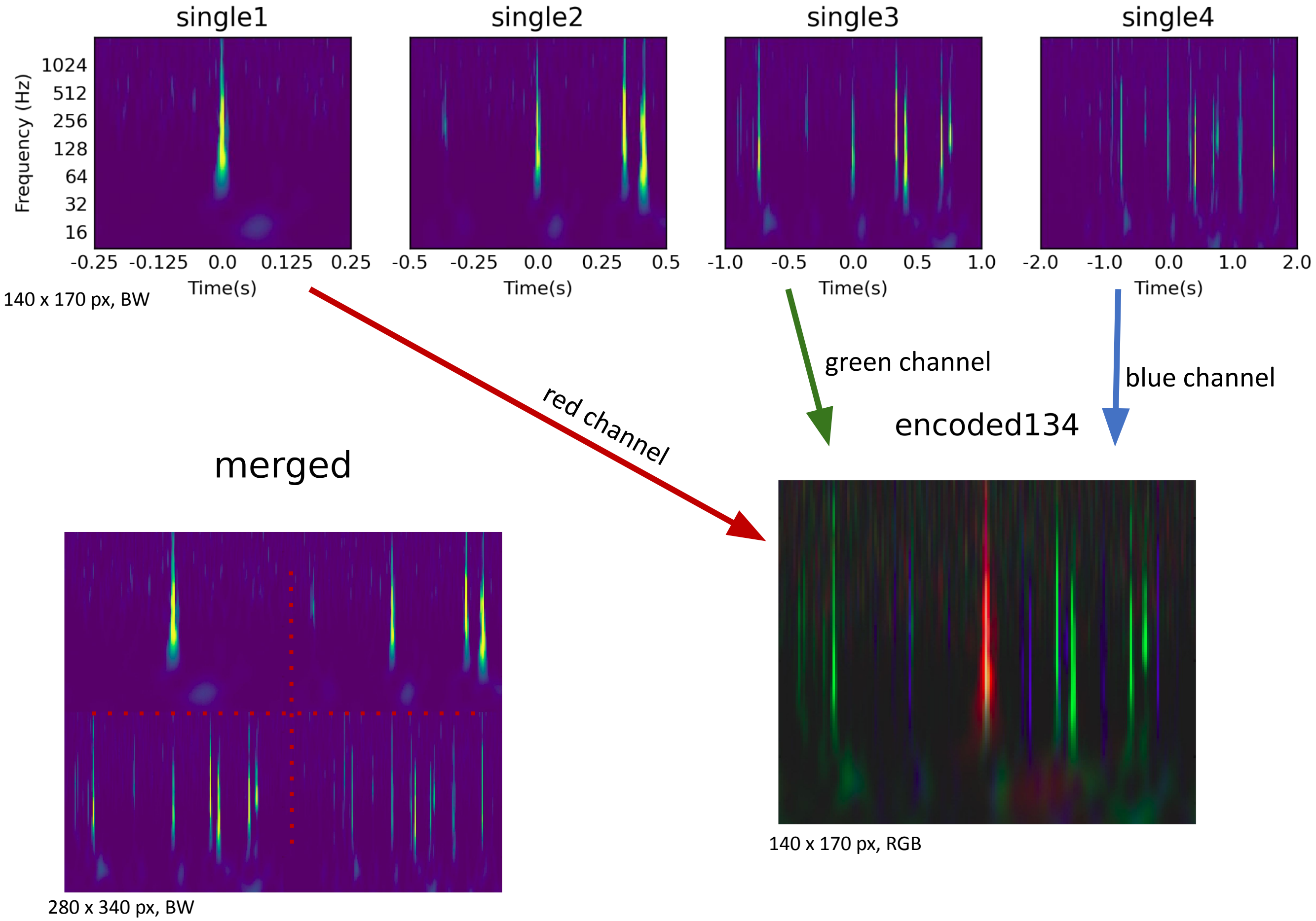
No.	Class	Total samples	No.	Class	Total samples
0	1080 Lines	328	11	No Glitch	181
1	1400 Ripples	232	12	None of the Above	88
2	Air Compressor	58	13	Paired Doves	27
3	Blip	1869	14	Power Line	453
4	Chirp	66	15	Repeating Blips	285
5	Extremely Loud	454	16	Scattered Light	459
6	Helix	279	17	Scratchy	354
7	Koi Fish	830	18	Tomte	116
8	Light Modulation	573	19	Violin Mode	472
9	Low_Frequency Burst	657	20	Wandering Line	44
10	Low Frequency Lines	453	21	Whistle	305

[1] S. Bahaadini et al., “Machine learning for Gravity Spy: Glitch classification and dataset,” Information Sciences, vol. 444, pp. 172–186, 2018. doi: 10.1016/j.ins.2018.02.068.
[2] S. Bahaadini et al., “Machine learning for Gravity Spy: Glitch classification and dataset,” Oct. 2018. url: https://zenodo.org/record/1476156

Baseline model

- Different views tried:
  - single views 1 to 4;
  - merged view [3];
  - encoded views [4] (every combination of at least 2 single views).
- Baseline models, trained from scratch:
  - ResNet18 and ResNet34 [5]

layer name	output size	18-layer	34-layer
conv1	112×112	7×7, 64, stride 2	
conv2_x	56×56	3×3 max pool, stride 2	
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax	
FLOPs		1.8×10^9	3.6×10^9



[3] S. Bahaadini et al., "Deep multi-view models for glitch classification," in 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2017, pp. 2931–2935. doi: 10.1109/ICASSP.2017.7952693.
[4] D. George, H. Shen, and E. Huerta, "Deep Transfer Learning: A new deep learning glitch classification method for advanced LIGO," 2017. arXiv preprint: 1706.07446.
[5] K. He et al., "Deep residual learning for image recognition," Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2016-Decem, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

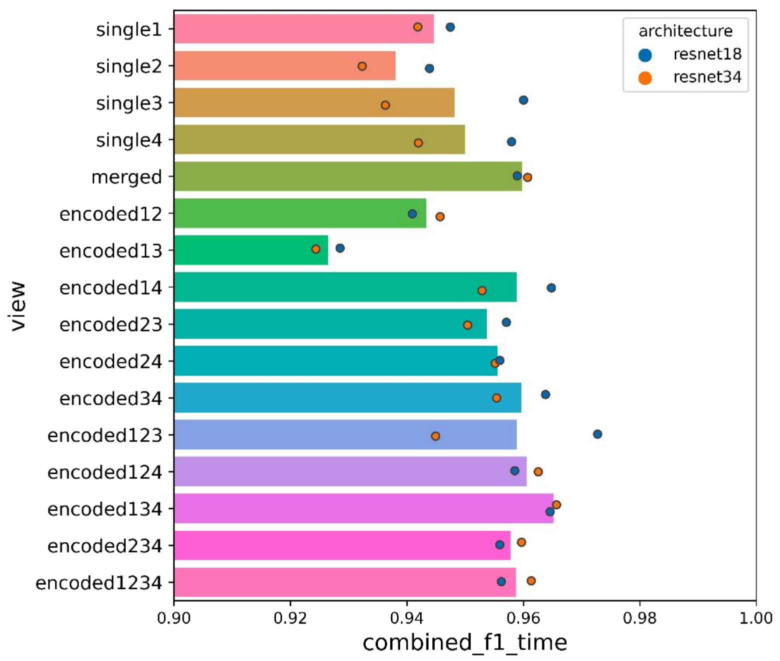
Baseline model

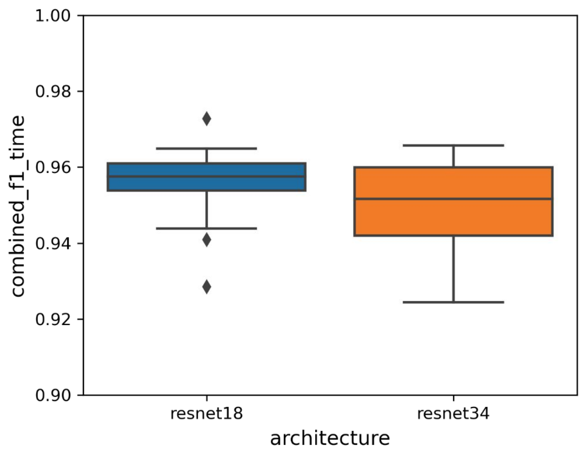
- Metrics:
  - (Macro-averaged) F1 score
  - combined_f1_time (avoid models which are too slow to train)

$$\text{combined\_f1\_time} = \text{f1\_score} - \text{total\_runtime}/30000$$

- Chosen view → **encoded134**:
  - similar F1 score as the merged view in less time (encoding information in the channel dimension is more efficient than increasing image size);
  - F1 score higher than encoded1234 (could be due to training randomness);
  - 3-channel structure is useful for transfer learning.

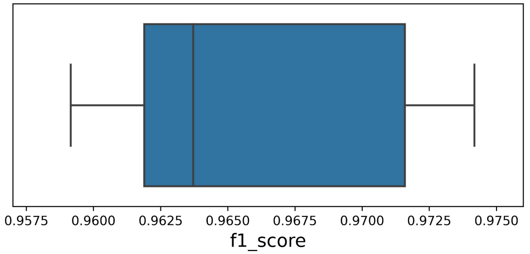
- Chosen architecture → **ResNet18**:
  - better F1 scores with less training time.

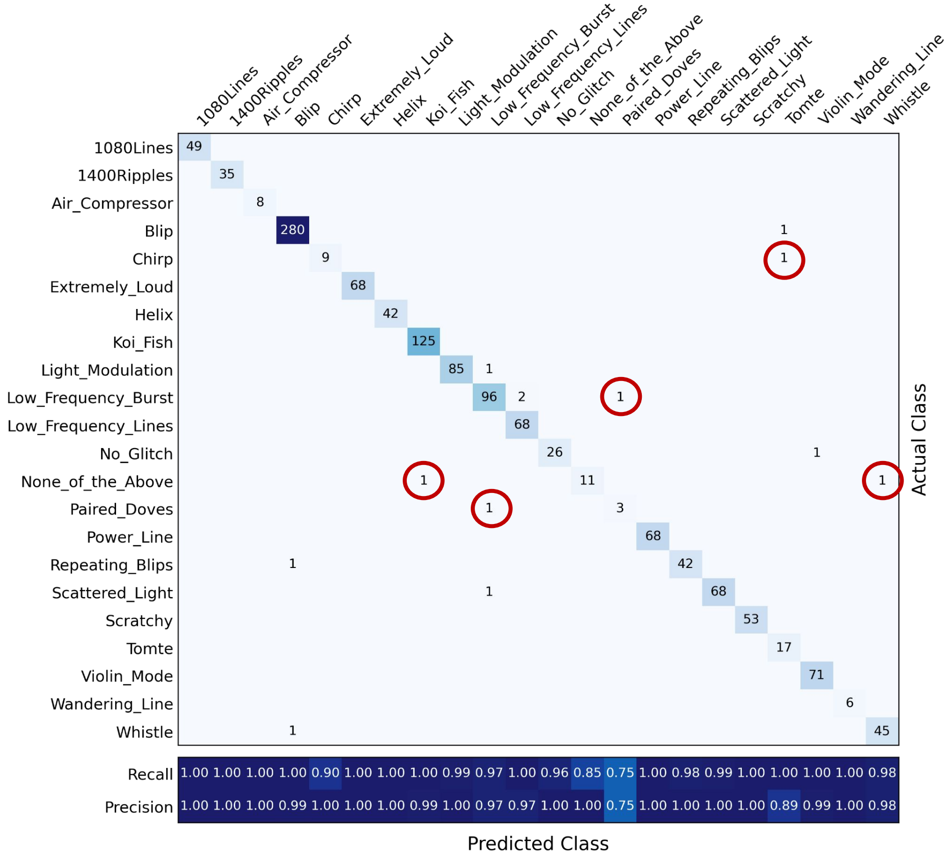


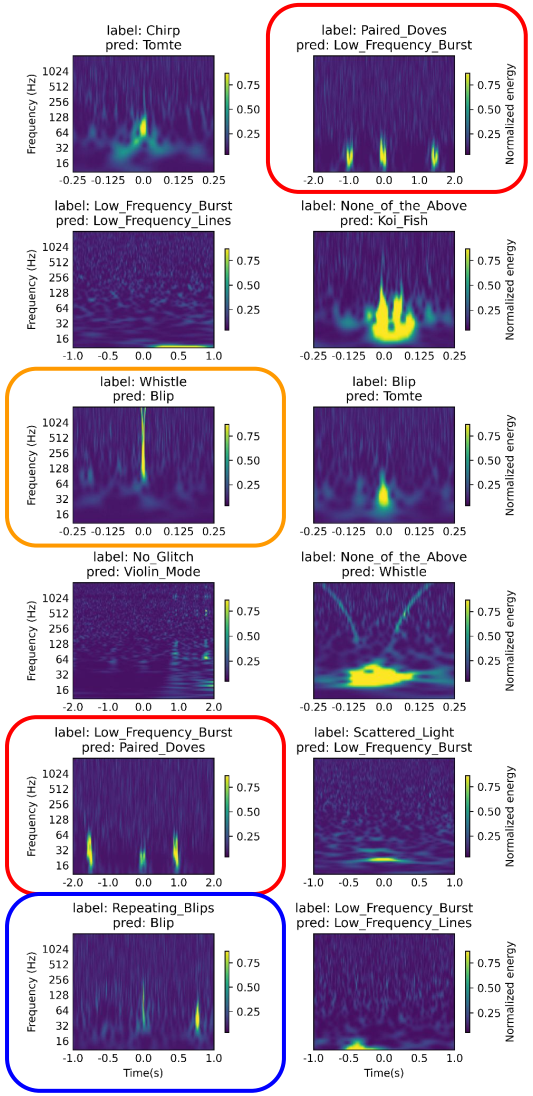


Baseline model

- Baseline configuration:
  - ResNet18 architecture
  - encoded134 view
  - 15 epochs
  - bs 64
  - steep lr function
- The baseline configuration was used to train five independent models.
- Evaluation on the best one on the validation dataset:
  - 97.4% F1 score → **98.1%** after label correction;
  - Precision and recall ≥ 95% for 18 out of 22 classes;
  - ⅓ of the errors involved the minority classes.
- Can results be improved if class imbalance is addressed?







Addressing class imbalance

- First approach → increase the importance of the less common classes.

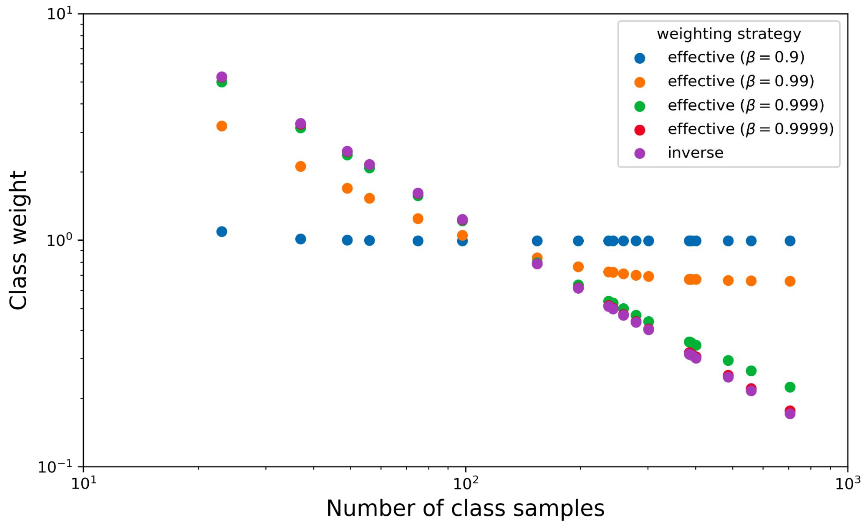
$$\mathcal{L}(\boldsymbol{\theta}) = -\sum_{k=1}^{K} w_i\, y_k \log(\hat{p}_k)$$

  - Inverse re-weighting:

$$w_i = \frac{1}{N_i}$$

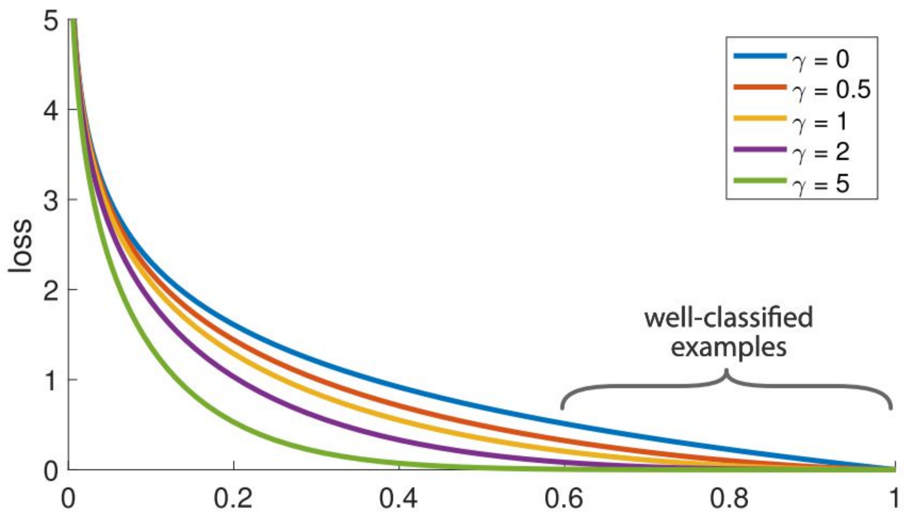
  - Effective number of samples [6]:

$$w_i = \frac{1-\beta}{1-\beta^{N_i}} \quad , \beta \in [0, 1[$$



- Second approach → use the focal loss function [7], which decreases the importance of samples were the model is very confident.

$$\mathcal{L}(\boldsymbol{\theta}) = -\sum_{k=1}^{K} w_i \boxed{(1-\hat{p}_k)^{\gamma}}\, y_k \log(\hat{p}_k) \quad , \gamma \geq 0$$

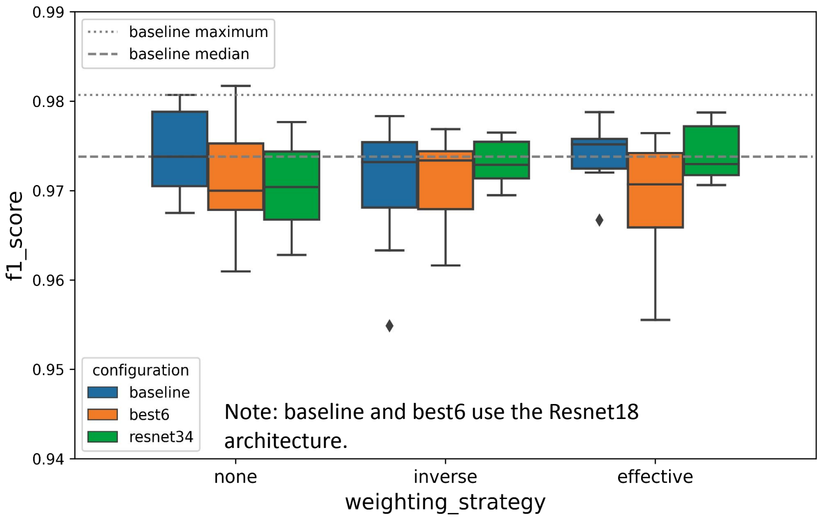


[6] Y. Cui et al., "Class-balanced loss based on effective number of samples," Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2019-June, pp. 9260–9269, 2019. doi: 10.1109/CVPR.2019.00949

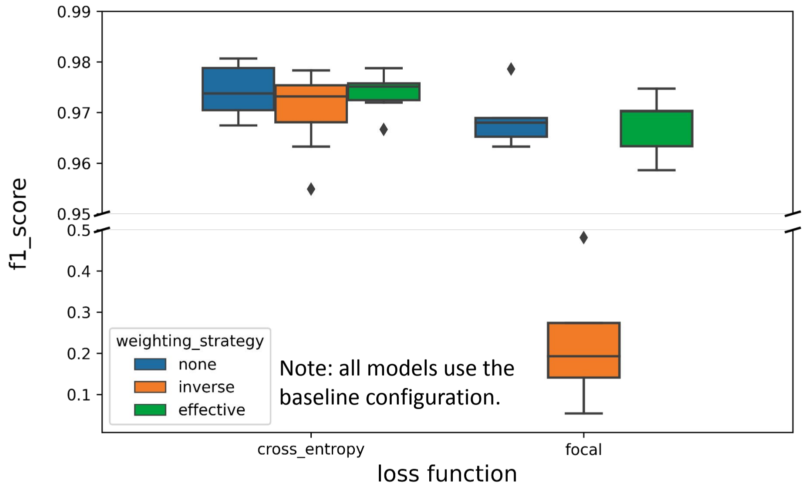
[7] T. Lin et al., "Focal Loss for Dense Object Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, no. 2, pp. 318–327, 2020. doi: 10.1109/TPAMI.2018.2858826

Addressing class imbalance

- First approach → increase the importance of the less common classes.
  - The ResNet18 models' performance does not increase, but
  - ResNet34's performance improves!



- Second approach → use the focal loss function [7], which decreases the importance of samples were the model is very confident.
  - Focal loss does not improve the performance.
  - It combines very badly with the inverse weighting strategy.

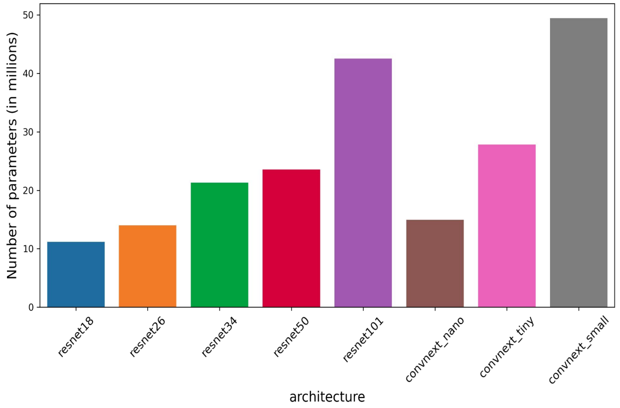


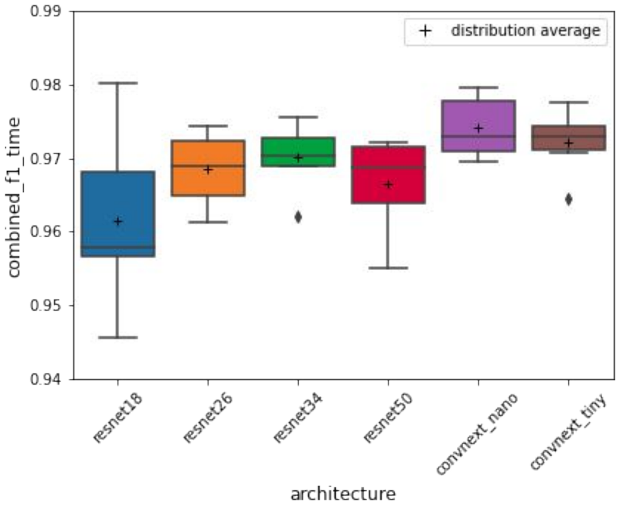
Transfer learning model

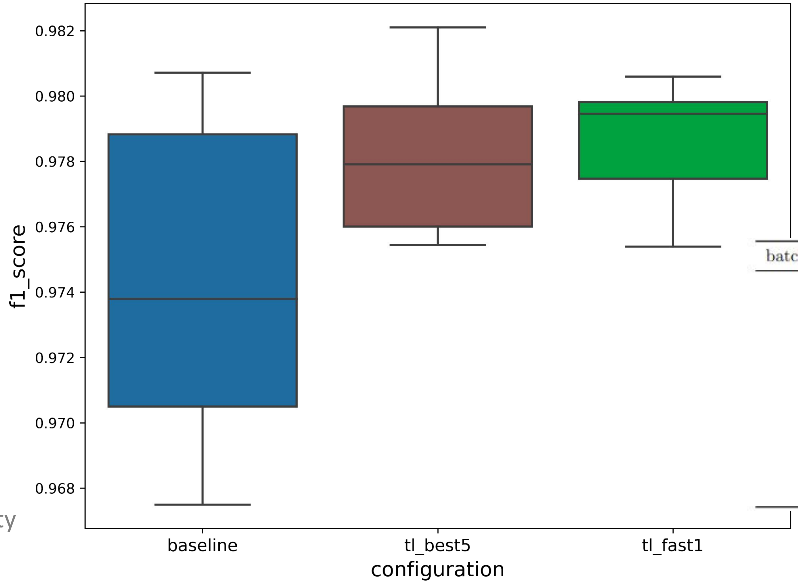
- Using pre-trained models can yield better performance and allow for faster training.

- Tested architectures:
    - Resnet18, 26, 34 and 50 [5]
    - ConvNeXt Nano and Tiny [8]
- ConvNeXt Nano outperforms the others.

- A bayesian sweep was performed to find good sets of hyperparameters for the fine-tuning of ConvNeXt Nano.
- Two of the found configurations appear to perform better than the baseline.
- The best run of tl_best5, with a 98.21% validation F1 score, was chosen as the best model.







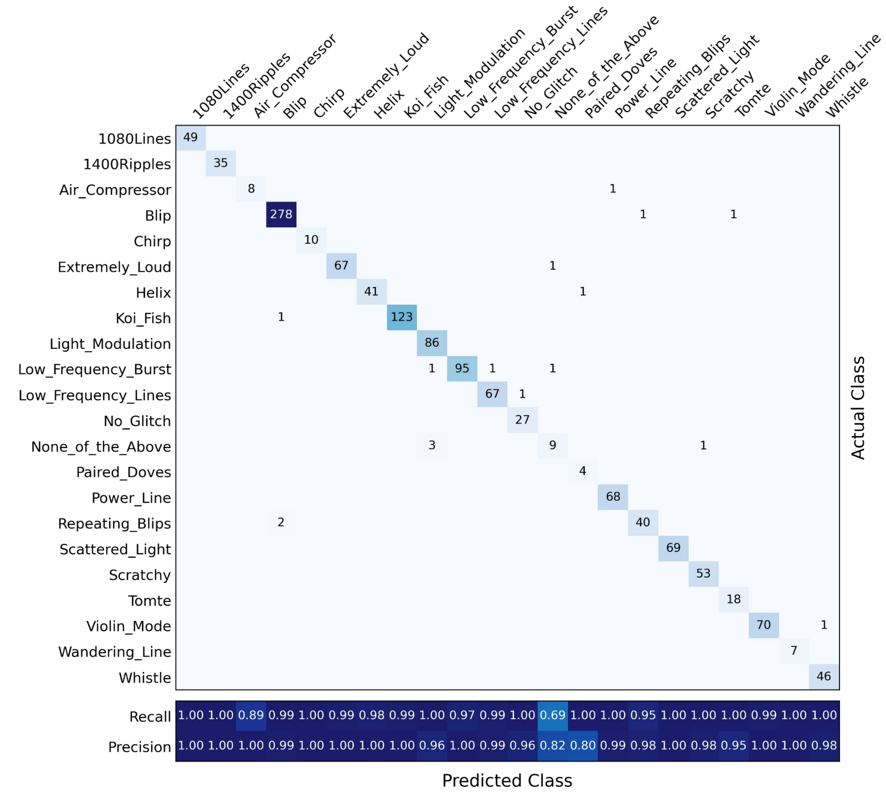
batch_size	epochs	suggest_func	loss_function	re-weighting	configuration
64	0 + 15	steep	cross_entropy	none	baseline
64	3 + 7	minimum	focal_loss	effective	tl_best1
64	2 + 6	minimum	focal_loss	none	tl_best2
32	1 + 6	steep	focal_loss	none	tl_best3
128	2 + 7	minimum	focal_loss	effective	tl_best4
64	2 + 8	minimum	focal_loss	effective	—
64	2 + 8	minimum	cross_entropy	inverse	tl_best5
[illegible]	[illegible]	steep	focal_loss	effective	tl_best6
64	1 + 5	minimum	cross_entropy	inverse	tl_fast1
64	1 + 4	steep	cross_entropy	inverse	tl_fast2
64	1 + 4	steep	cross_entropy	effective	tl_fast3

[5] K. He et al., “Deep residual learning for image recognition,” Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2016-Decem, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.
[8] Z. Liu et al., “A ConvNet for the 2020s,” 2022. arXiv preprint: 2201.03545.

Model evaluation on the test set

- The baseline and tl_best5 models were evaluated in the test dataset.
- The baseline model achieved higher performance, despite being worse than tl_best5 in the validation set. This could be due to having overfitted the validation set.

- The baseline model achieves precision and recall of at least 95% for 19 of the 22 classes.
- Results better than all previous articles other than George2017 [4].
- The chirp class has perfect F1 score, which motivates the next step: find if the model can correctly classify real GW signals, with no further training.



Model	F1 score (%)	accuracy (%)	Notes
merged view CNN [3]	not reported	96.89	different dataset version (20 classes)
merged view CNN [1]	not reported	97.67	improved version of [3]
hard fusion ensemble [1]	not reported	98.21	combines four CNNs
fine-tuned ResNet50 [4]	97.65	98.84	different split (no validation set)
tl_best5 [this work]	96.84	98.14	fine-tuned ConvNeXt_Nano
baseline [this work]	97.18	98.68	ResNet18 trained from scratch

[1] S. Bahaadini et al., "Machine learning for Gravity Spy: Glitch classification and dataset," Information Sciences, vol. 444, pp. 172–186, 2018. doi: 10.1016/j.ins.2018.02.068.

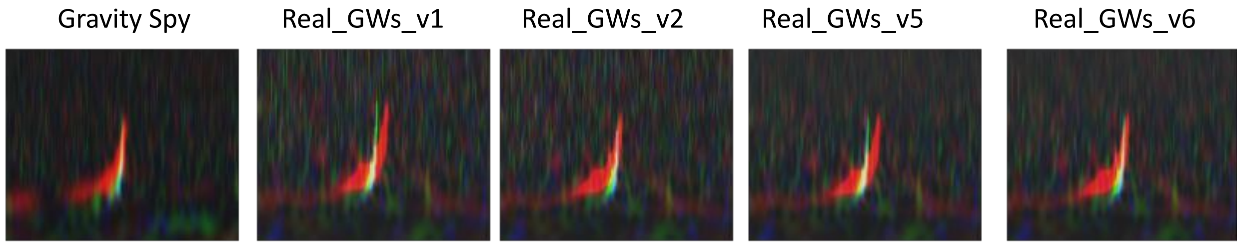
[3] S. Bahaadini et al., "Deep multi-view models for glitch classification," in 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2017, pp. 2931–2935. doi: 10.1109/ICASSP.2017.7952693.

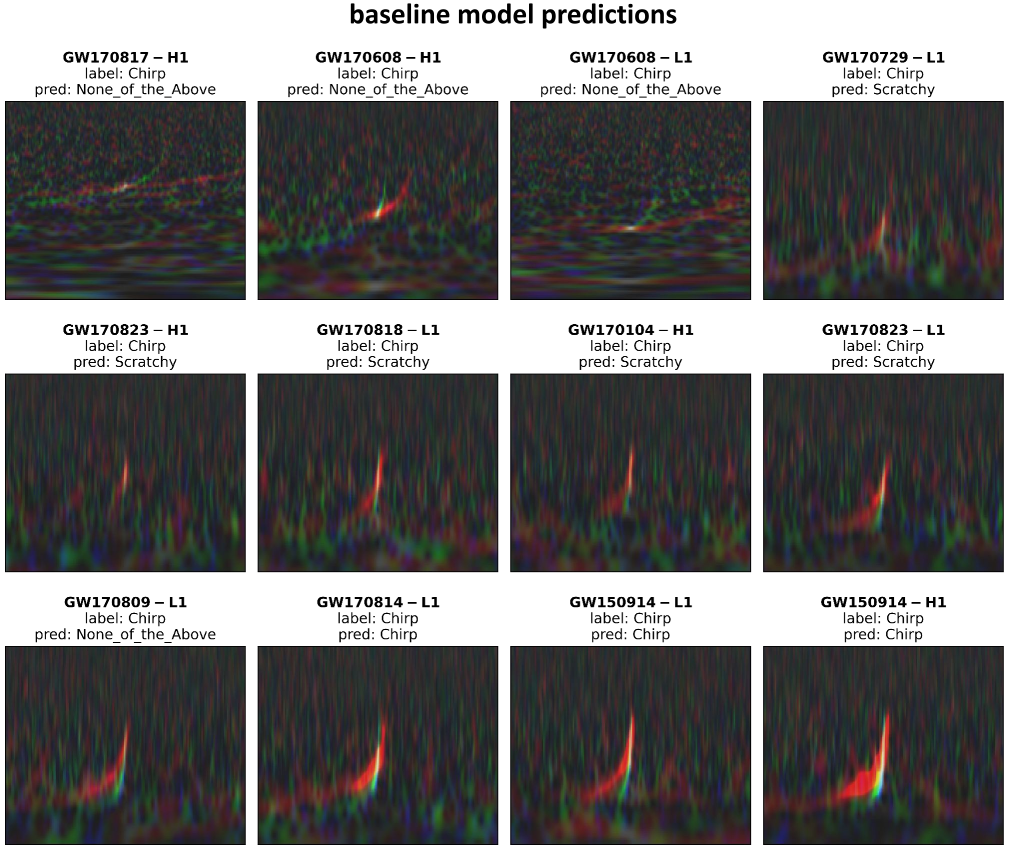
[4] D. George, H. Shen, and E. Huerta, "Deep Transfer Learning: A new deep learning glitch classification method for advanced LIGO," 2017. arXiv preprint: 1706.07446.

Testing the models with real GW signals

- The LIGO (H1 and L1) strain data from the 11 O1 and O2 confident detections were converted to a format similar to the dataset.
- 12 examples where the chirp behaviour was observable were manually selected.

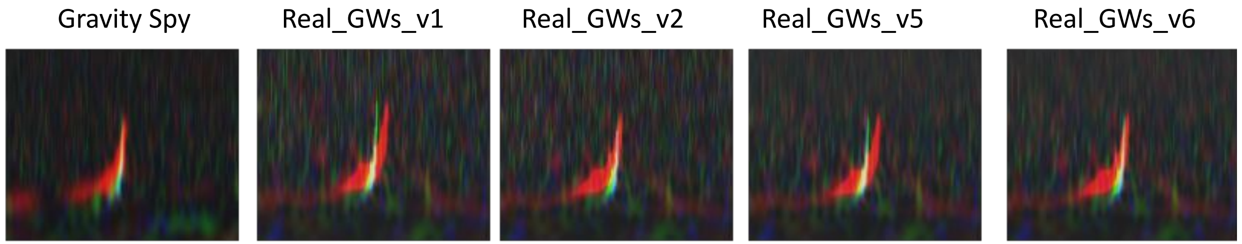
- The predictions of the baseline model were heavily influenced by the sample creation pipeline.
- For the most similar dataset, Real_GWs_v6:
    - 3 events were correctly identified as Chirp → 25% recall;
    - 4 were labelled as None of the Above (mainly due to different morphology);
    - 5 identified as Scratchy (low energy GW signal).

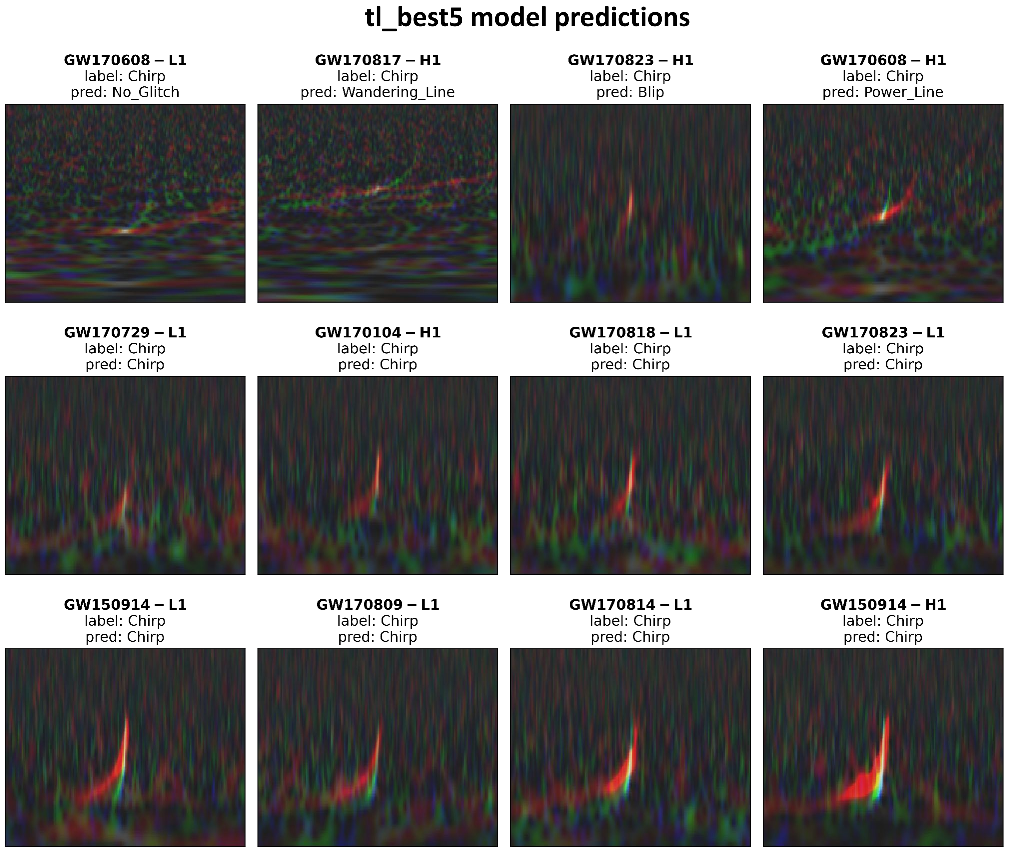




Testing the model with real GW signals

- The best model trained with transfer learning was also tested on the real GWs.
- For Real_GWs_v6 8 events were correctly identified as Chirp → 75% recall!
- For the other dataset versions, the recall was at least equal. The TL model was much more robust, even when the channels were shifted.





Conclusion

- Deep Learning is a good approach for the classification of glitches, particularly when converted to spectrograms.
- Encoded views are an effective way of presenting information to the models.
- Small models appear to be enough to separate the different glitch classes.
- Models trained with less than 50 chirp examples were capable of detecting real GWs.

- Bigger datasets, including O3 data, are needed[1].
- Synthetic data generation could help populate the less represented classes.

[1] You can help by participating in the citizen science project at https://www.zooniverse.org/projects/zooniverse/gravity-spy

**THANK YOU
FOR YOUR ATTENTION!**