PROCEEDINGS OF SCIENCE

Trialling the use of generative adversarial networks for efficient identification of radio frequency interference

J. E. Brooks[*] **and M. K. Argo**

Jeremiah Horrocks Instiitute,
University of Central Lancashire,
Preston, UK, PR1 2HE

E-mail: jebrooks@uclan.ac.uk

A new method of detecting RFI in time-frequency visibility images is presented, using the novel Generative Adversarial Network machine learning architecture. The network is trained entirely on manually flagged data produced by the Multi-Element Radio Linked Interferometer Network (e-MERLIN). The network shows an excellent ability to identify and flag RFI features that are common in the training set, but is not as proficient at identifying less common features such as wide-band RFI. The network will be tested against current methods of automatic flagging to quantify and compare its effectiveness. More advanced methods of excising RFI are critical to the future of radio astronomy as the radio spectrum becomes more polluted and even the most remote stations are disrupted by modern satellite networks.

15th European VLBI Network Mini-Symposium and Users' Meeting (EVN2022)
11-15 July 2022
University College Cork, Ireland

[*] Speaker

PoS(EVN2022)051

1. Introduction

All observations in the radio range of frequencies will, to varying degree, suffer from contamination by Radio Frequency Interference (RFI). The scale of contamination is generally determined by the location of the antenna, or antennas. An antenna near an airport, for example, will pick up more RFI than an antenna situated in the middle of the desert. Interferometers are less affected by RFI than single-dish observatories as it is less likely an RFI signal will be correlated at both ends of a baseline, though the problem still persists. Furthermore, the location of the antenna is becoming less of an advantage, as even the most remote antenna will still pick up interference from satellite networks which are becoming more populated and complex year by year [1].

Improving methods of RFI detection is naturally an important area of research as RFI contamination becomes more problematic. Currently, most post-correlation RFI detection algorithms will operate on a time-frequency image in a statistical manner [2]. These methods are, unquestionably, effective at removing RFI, but can be overly aggressive without significant human intervention. This intervention is often the responsibility of the researcher, who may not be familiar with the method implemented in a particular observatory's pipeline. This creates a potential problem, where large amounts of uncontaminated data produced by an observatory may never be used. This problem is compounded as RFI becomes more prevalent across the spectrum, so progressively larger chunks of a given observation are lost to RFI.

Finding alternative methods for flagging RFI is an ongoing area of research and experimentation (e.g. [3–5]). Algorithms that flag RFI must conform to a specific standard that the technical challenges of interferometry impose. First and foremost, the algorithm must remove as close to 100% of the RFI as possible. After this, the order of importance is much more specific to an observatory, but generally speaking the algorithm should: be quick, the overall post-processing of an observation (where flagging normally takes place) should run for less than the observation's duration; it should also be precise, and leave untouched data that is representative of the desired signal; ideally it should also require as little human intervention as possible. It should also be adaptive enough so that any observatory can implement it, with minimal configuration. This simplifies the job of an astronomer that is using data from different observatories as the flagging method would be common amongst them, and simplifies collaboration between the observatories themselves for the same reason. The algorithm should also allow for modification, to skip over signals that are astronomical but may appear as RFI. Examples of this include spectral lines and Fast Radio Bursts (FRBs). Finally, it should ideally be transparent enough to allow any problems to be diagnosed and fixed. Fulfilling all these needs is certainly a challenging technical problem, but theoretically possible.

The most promising candidates in the literature use some form of machine learning [3, 6–8], a methodology that has become popular in recent years as even the most complex algorithms have been made accessible to non-experts. One such algorithm that shows promise is the Generative Adversarial Network (GAN), a novel technique that uses two networks working against each other to produce a network that is adept at generating accurate predictions, while maintaining the appearance of the training set [9, 10]. Machine learning methods desire perfect knowledge of RFI to be effective, which presents a problem as there is often a fairly large grey area between an RFI-contaminated signal and an RFI-free signal. A solution is to use simulated visibilities instead, where perfect knowledge of RFI is achievable. However this also introduces an assumption that the simulated signal is perfectly representative of a 'real' signal the observatory might receive during an observation.

This document will outline the work completed so far on using a GAN, trained only on manually flagged empirical data, to flag RFI. No simulated data is used to remove the assumption that it introduces, and instead 'perfect' knowledge of the RFI in the training set is approximated by manually flagging every baseline. This work builds on the conclusions of Li et al. [8] who used the pix2pix [10] model to detect RFI. They use simulations from the HIDE package [11] as training data to detect RFI in single-dish radio observations. In this experiment a similar GAN model is used, instead trained on real, manually flagged data from an interferometer array.

2. Network Architecture

The defining feature of a GAN is the presence of two opposing networks. In most architectures only a single network is present, and it scores its performance using a pre-defined loss function. In a GAN, one of the networks replaces the role of the loss function so that there are two networks, and one scores the other in addition to their own internal optimisations. One of the networks in a GAN is known as the discriminator, a binary classifier tasked with classifying images into 'real' and 'fake' images by producing a probability that the input it receives is real. Real images, by definition, come from the training set and fake images are generated by the other network in the GAN known as the generator. This generator, in turn, is tasked with generating images that will trick the discriminator into classifying its output as real. The flavour of GAN used in this work, known as a Conditional GAN (or cGAN) is a slight modification of the original GAN that allows an image input to the generator that informs its output.

The performance of the GAN is encapsulated by the value function:

$$V(G, D) = \mathbb{E}_{x,y}[\log D(x, y)] + \mathbb{E}_{x}[\log(1 - D(x, G(x)))]$$

where G represents the generator, D represents the discriminator, x is the input image containing the visibilities, and y is the ground truth RFI mask associated with x. The generators attempt at mimicking y is $G(x)$. The tension between the two networks arises because during training G is tasked with minimising V, whereas D is tasked with maximising it. Each network also has its own method of optimising itself, these loss functions are detailed in the sections following.

2.1 The Discriminator

For a discriminator, a Convolutional Neural Network (CNN) is used to classify images into two categories: 'real' and 'fake'. During training, the network has access to both the real and fake image, which is how it measures its own performance. The loss function that the discriminator network will attempt to minimise is:

$$L_D = CE(G(x), O) + CE(y, J)$$

where CE represents a binary cross entropy operation, O is a matrix of zeros, and J is a matrix of ones. By minimising this loss function, the discriminator becomes adept at separating real images from generated images.

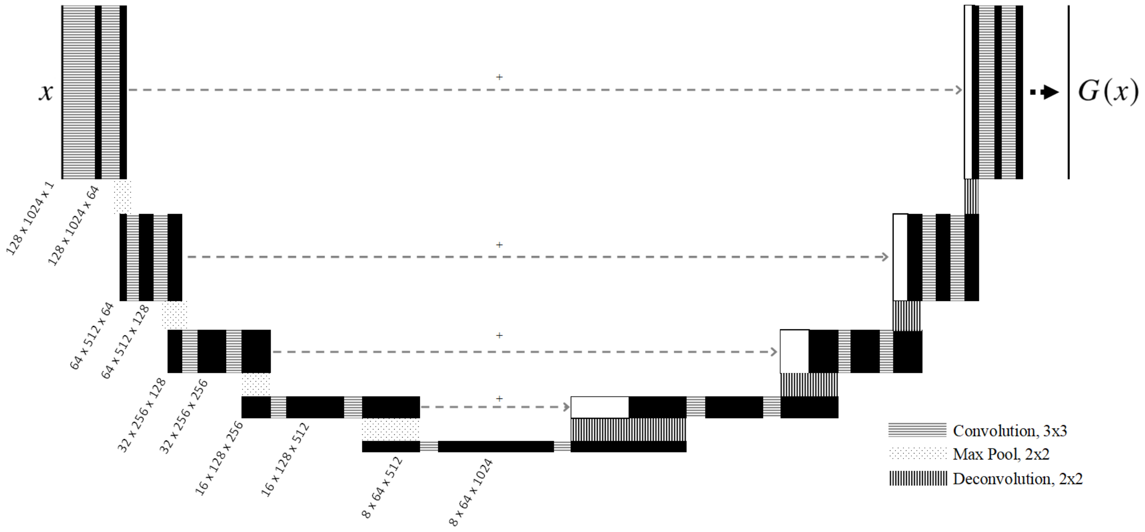

Figure 1: An illustration of how an input image, x, is operated on by the U-net used in this work. The output, $G(x)$, is a probability mask derived from x that indicates the likelihood of RFI at each pixel. Image dimensions are given to the left of each node, and are mirrored as the image expands in size. Operations are indicated by varying shaded regions in the space between the image nodes. Skip connections, indicated by the horizontal arrows marked with an addition symbol, give further detail to the deconvolved images. The format of this image is similar to that presented in [12].

2.2 The Generator

For a generator, the U-net architecture is used [12]. A U-net uses several layers of convolution operations that progressively sample higher order features. For example, in a U-net tasked with generating an image of a car, the surface layers will detect simple features like edges and corners. Deeper layers will develop a sensitivity to more complex features, like a door handle or a headlight. This contracting series of operations is followed by an expansive series of deconvolution operations that unravel the learned features into an output image. In the case of RFI detection, the U-net is trained to produce a probability mask that sits on top of the input image, and indicates the likelihood of RFI in each pixel. As a final step, a threshold is imposed on the output to convert it into the widely used boolean flag mask of RFI. Figure 1 shows how a particular input progresses through the network to be converted into a probability mask.

To encourage the U-net to produce images that appear like those in the training set, while factoring in the output of the discriminator, a two part loss function is used:

$$L_G = V(G, D) + \lambda L_{L1}(y, G(x))$$

where V is the value function defined above, L_{L1} is the L1 loss, and λ is a constant with a value $\lambda = 100$ [9]. The inclusion of V within the generator loss is how the discriminator encourages the generator into producing images that appear like those in the training set; if the discriminator performs well then the value of L_G increases proportional to V, and the generator will make greater changes to decrease L_G.

During training, the U-net is given flag masks consisting of ones and zeros, representing the probability of RFI and not RFI. This level of precision is too difficult for the U-net to effectively reproduce and causes problems during training. Therefore, following the technique of [8], a very small amount of gaussian noise is added to both the generated flag mask and the real flag mask, which gives the U-net some room for error in its output.

3. Training Data

Two observations conducted by the Multi-Element Radio Linked Interferometer Network (e-MERLIN), under project code CY5209, are used to create flag masks to train the GAN. These observations are of a BL Lac type AGN located at $\alpha = 10^{\mathrm{h}}58^{\mathrm{m}}37.73^{\mathrm{s}}$ and $\delta = +56°28'11.18''$, which functions as the phase calibrator source for both observations on 16th June 2017, and 1st January 2018. A total of approximately 6 hours of observing time on this source is included in the training set, divided into two chunks of 3 hours for each observation. The data was manually flagged, baseline by baseline, using a purpose built GUI, so that the flags do not unnecessarily cover pixels that are not RFI and as much data as possible is preserved.

Before the visibilities can be given to the GAN, some operations must be performed on them so that the RFI is easier to identify by the network. The network itself is biased by this process as it learns from the pre-processed training data, so any images that are passed to the GAN must also have been through the same process to achieve a consistent result. Firstly, each time-frequency image, which corresponds to a particular polarisation and baseline, is winsorized to the 99th percentile. This is performed on the absolute values of the visibilities, and significantly reduces the impact of extreme outliers in a given image. Secondly, a two-dimensional surface is removed from each image, using the method described in section 2.2 of [2]. The intent is to separate the majority of the underlying astronomical signal that we are attempting to observe, and leave behind any RFI to be removed. This is and should be done for each field in the observation, as each field will naturally observe a slightly different signal. Finally, features that are common across the observation are removed from each time-frequency image, namely the visibilities normally removed due to antennas not yet being on-source, and the visibilites at the extreme edges of each spectral window that are heavily attenuated by instrumentation. These pre-processing steps produce a time-frequency image like that in Figure 2a, which can then be passed to the GAN to identify RFI in the image.

4. Results

At the time of writing, the GAN is functional and appears to produce reasonable results, but has not yet been robustly tested against existing methods of RFI detection. An example of the output of the GAN is given in Figure 2b. This figure illustrates the excellent ability of the GAN to detect very narrow-band RFI. However, it is not as proficient at detecting less well-defined RFI such as the wideband RFI at approximately 23 minutes, and the persistent RFI at approximately 1.305 GHz, both in Figure 2a. The ability of the GAN to handle different types of RFI will be based on the representation of that RFI within the training set. As persistent RFI with a very narrow frequency range is, by far, the most common type in the training set, it is logical that the GAN is better at detecting it compared to other types. Therefore, we plan to include observations that suffer more from these other types of RFI within the training set in future work.

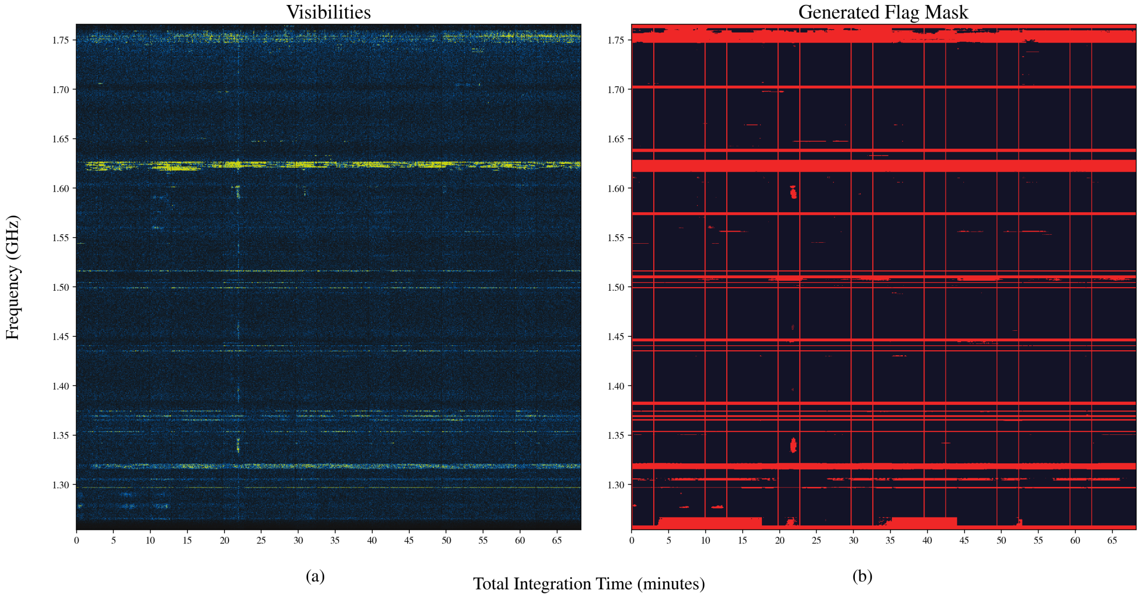

Figure 2: (a): A sample of visibilities not included in the training set. They have been pre-processed according to the method described in section 3, so this is a direct representation of the input to the GAN. (b): the output of the GAN derived from the input displayed in (a). Red areas indicate flags generated by the GAN.

One of the incentives of using the GAN architecture is its ability to reproduce the 'look' of images in the training set. In this case, the images in the training set are flagged to minimise the occurrence of false positives. One reason the method shows promise, is that the GAN seems able to reproduce this in its predictions, and not identify RFI where there is none simply because it is in proximity to other contaminated pixels. For short observations, this may not make a huge difference, but for the total output of an observatory or any long-exposure observation, this behaviour is beneficial. Furthermore, any observation that prioritises sensitivity, such as deep-field investigations, will also benefit from a low false-positive flagging rate as more uncontaminated visibilities will be utilised in the final imaging.

Work is underway to robustly test the performance of the network against a variety of methods that will flag the same observation. Two manually flagged versions of the test observation, flagged by two different people, will allow a comparison against human intervention. The popular flagging software AOFlagger [13] will be tested, along with some automatic flagging methods included in the Common Astronomy Software Applications (CASA) [14] software package. The metric used to rank these methods will likely be a measurement of the rms noise in the final image, as this is what determines the precision of measurements such as flux density. As long as the process used to produce each image is the same in each case, the rms noise for a particular observation is a viable ranking metric.

5. Conclusion

A new method of automatic identification of RFI in radio visibilities is presented. The method uses a Generative Adversarial Network architecture, consisting of a U-net for image translation and a Convolutional Neural Network for image classification. The network shows an excellent ability to identify persistent RFI with a very narrow bandwidth, but is less able to identify less common types. This is easily explained by the relative representation of these types in the training set, thus a solution is to expose the network to a more diverse range of RFI at training time. Work is being conducted on an experiment that will test how use of the network in place of other automatic flagging methods changes the rms noise in a given observation, which dictates the accuracy of measurements from a radio image. Researching new methods of RFI detection is a critical area of development as RFI at radio observatories becomes more and more pervasive across the world.

References

[1] A.A. Deshpande and B.M. Lewis, *Iridium Satellite Signals: A Case Study in Interference Characterization and Mitigation for Radio Astronomy Observations*, *Journal of Astronomical Instrumentation* **8** (2019) 1940009 [1904.00502].

[2] A.R. Offringa, A.G. de Bruyn, M. Biehl, S. Zaroubi, G. Bernardi and V.N. Pandey, *Post-correlation radio frequency interference classification methods*, *MNRAS* **405** (2010) 155 [1002.1957].

[3] D. Agarwal, K. Aggarwal, S. Burke-Spolaor, D.R. Lorimer and N. Garver-Daniels, *FETCH: A deep-learning based classifier for fast transient classification*, *MNRAS* **497** (2020) 1661 [1902.06343].

[4] Y. Maan, J. van Leeuwen and D. Vohl, *Fourier domain excision of periodic radio frequency interference*, *A&A* **650** (2021) A80 [2012.11630].

[5] Q. Zeng, X. Chen, X. Li, J.L. Han, C. Wang, D.J. Zhou et al., *Radio frequency interference mitigation based on the asymmetrically reweighted penalized least squares and SumThreshold method*, *MNRAS* **500** (2021) 2969.

[6] J. Akeret, C. Chang, A. Lucchi and A. Refregier, *Radio frequency interference mitigation using deep convolutional neural networks*, *Astronomy and Computing* **18** (2017) 35 [1609.09077].

[7] J. Kerrigan, P. La Plante, S. Kohn, J.C. Pober, J. Aguirre, Z. Abdurashidova et al., *Optimizing sparse RFI prediction using deep learning*, *MNRAS* **488** (2019) 2605 [1902.08244].

[8] Z. Li, C. Yu, J. Xiao, M. Long and C. Cui, *Detection of radio frequency interference using an improved generative adversarial network*, *Astronomy and Computing* **36** (2021) 100482.

[9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair et al., *Generative adversarial nets*, in *Advances in Neural Information Processing Systems*,

Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence and K. Weinberger, eds., vol. 27, Curran Associates, Inc., 2014, https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf.

[10] P. Isola, J.-Y. Zhu, T. Zhou and A.A. Efros, *Image-to-image translation with conditional adversarial networks*, in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, 2017, DOI.

[11] J. Akeret, S. Seehars, C. Chang, C. Monstein, A. Amara and A. Refregier, *HIDE & SEEK: End-to-end packages to simulate and process radio survey data*, *Astronomy and Computing* **18** (2017) 8 [1607.07443].

[12] O. Ronneberger, P. Fischer and T. Brox, *U-net: Convolutional networks for biomedical image segmentation*, in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W.M. Wells and A.F. Frangi, eds., (Cham), pp. 234–241, Springer International Publishing, 2015.

[13] A.R. Offringa, J.J. van de Gronde and J.B.T.M. Roerdink, *A morphological algorithm for improving radio-frequency interference detection*, *Astronomy & Astrophysics* **539** (2012) A95 [1201.3364].

[14] J.P. McMullin, B. Waters, D. Schiebel, W. Young and K. Golap, *CASA Architecture and Applications*, in *Astronomical Data Analysis Software and Systems XVI*, R.A. Shaw, F. Hill and D.J. Bell, eds., vol. 376 of *Astronomical Society of the Pacific Conference Series*, p. 127, Oct., 2007.