Monthly Notices of the ROYAL ASTRONOMICAL SOCIETY

MNRAS **510,** 4504–4524 (2022) https://doi.org/10.1093/mnras/stab3553
Advance Access publication 2021 December 10

# Radio Galaxy Zoo: giant radio galaxy classification using multidomain deep learning

H. Tang [1,2]★ A. M. M. Scaife,[1,3] O. I. Wong [4,5,6] and S. S. Shabala [7]

[1] *Jodrell Bank Centre for Astrophysics, School of Physics and Astronomy, The University of Manchester, England, Manchester M13 9PL, UK*
[2] *Department of Astronomy, Tsinghua University, Beijing 100084, China*
[3] *The Alan Turing Institute, Euston Road, London NW1 2DB, UK*
[4] *CSIRO Astronomy and Space Science, PO Box 1130, Bentley, WA 6102, Australia*
[5] *ICRAR-M468, University of Western Australia, Crawley, WA 6009, Australia*
[6] *ARC Centre of Excellence for All Sky Astrophysics in 3 Dimensions (ASTRO 3D), Australia*
[7] *School of Natural Sciences, Private Bag 37, University of Tasmania, Hobart, TAS 7001, Australia*

Accepted 2021 December 2. Received 2021 December 2; in original form 2021 May 6

## ABSTRACT

In this work we explore the potential of multidomain multibranch convolutional neural networks (CNNs) for identifying comparatively rare giant radio galaxies from large volumes of survey data, such as those expected for new generation radio telescopes like the SKA and its precursors. The approach presented here allows models to learn jointly from multiple survey inputs, in this case NVSS and FIRST, as well as incorporating numerical redshift information. We find that the inclusion of multiresolution survey data results in correction of 39 per cent of the misclassifications seen from equivalent single domain networks for the classification problem considered in this work. We also show that the inclusion of redshift information can moderately improve the classification of giant radio galaxies.

**Key words:** methods: statistical – radio continuum: galaxies – software: development.

## 1 INTRODUCTION

Radio galaxies are active galaxies with structures that typically consist of two radio lobes straddling a central host galaxy. These sources, together with radio loud quasars and some Seyfert galaxies, are often referred to as Double Radio sources associated with active galactic nuclei (DRAGNs; Leahy 1993). The radio emission from DRAGNs is dominated by the synchrotron radiation process (Shklovskii 1955; Burbidge 1956), which allows one to calculate constraints on the strength of the local magnetic fields and the cosmic ray energy spectrum of the emitting plasma (Shu & Kranakis 1982). They are believed to be powered by collimated, relativistic jets (Blandford & Rees 1974; Scheuer 1974), and magnetic fields of a certain geometry from their active nuclei (Peng, Chen & Strom 2015). Such active nuclei, or active forms of SuperMassive BlackHoles (SMBHs), have been found to reside in the most massive galaxies (Soltan 1982; Begelman, Blandford & Rees 1984; Rees 1984; Magorrian et al. 1998; Kormendy & Ho 2013; Dabhade et al. 2020b). The radio lobes of DRAGNs then in turn can be used to probe the SMBHs in their host galaxy centres. To date, thanks to the availability of large scale radio sky surveys such as the Revised 3C catalogue (3CR; Edge et al. 1959), the Faint Images of the Radio Sky at Twenty-Centimeters (FIRST; Becker, White & Helfand 1995), the NRAO VLA Sky Survey (NVSS; Condon et al. 1998), the Sydney University Molonglo Sky Survey (SUMSS; Bock, Large & Sadler 1999; Mauch et al. 2003), and the Westerbork Northern Sky Survey at 325 MHz (WENSS; Rengelink et al. 1997), tens of thousands of radio galaxies have been identified.

The large statistical samples from these sky surveys have motivated research into the maximum size to which a radio source might evolve, resulting in the discovery of the first two Giant Radio Galaxies (GRGs; Willis, Strom & Wilson 1974). GRGs are defined as those radio galaxies that have a projected linear size greater than 700 kpc (Dabhade et al. 2020a) under a ΛCDM cosmology with $\Omega_\mathrm{m} = 0.31$ and a Hubble constant of $H_0 = 67.8$ km s$^{-1}$ Mpc$^{-1}$ (Planck Collaboration XIII 2016). For example, Fig. 1 shows a log-scale radio map of 3C 236 from NVSS at 1.4 GHz, one of the first two GRGs discovered. The source has a host spectroscopic redshift of 0.099 358, measured from the Sloan Digital Sky Survey (SDSS; Albareti et al. 2017), and therefore an angular size distance of 1.892 kpc arcsec$^{-1}$ (Wright 2006). Since the end-to-end angular extent of the source as estimated from the NVSS map is 2505 arcsec, the estimated physical linear size of 3C 236 is 4.7 Mpc, significantly larger than the 700 kpc limit in the GRG definition.

The primary motivation for finding such giant radio sources is to investigate the possible modes of energy replenishment that allow for the existence of such a population (Longair, Ryle & Scheuer 1973), as energy losses over the physical scales associated with these gigantic radio components are unavoidable. Furthermore, such objects can also be used as probes of the local intergalactic medium (IGM), as their structures are likely to be influenced by their environment, and it has been shown in particular that GRGs can be used to investigate the missing baryon problem in intergalactic filaments, as probes of the Warm Hot Intergalactic Medium (WHIM; Peng et al. 2015).

★ E-mail: hongming.tang@manchester.ac.uk, abo060313@gmail.com

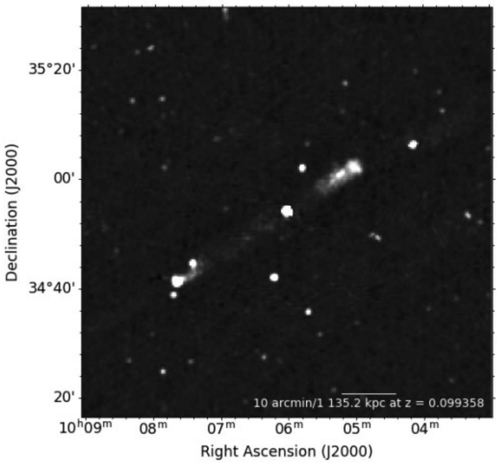


**Figure 1.** The NVSS radio map of 3C 236, a GRG example. The map is shown in logarithmic scale, with the scale bar on the diagram represents 10 arcmin in angular size. This is consistent to the linear scale of 1135.2 kpc at $z = 0.099\,358$ (Wright 2006).

Willis et al. (1974) have also pointed out that the large angular extent of these radio sources allows detailed imaging of such structures, and can therefore assist in studies of the physical processes occurring within the galaxies themselves. For instance, Krause et al. (2019) investigated well-resolved radio maps of 33 3CR radio sources. They found that 24 objects in their sample showed strong jet precession evidence, which is consistent with the hypothesis of black hole binary mergers (e.g. Babul, Sharma & Reynolds 2013; Cielo et al. 2018). This idea has also motivated an interest in the discovery of GRGs with unusual morphologies. For instance, by discovering several giant Double-Double Radio Galaxies, (DDRG; Schoenmakers et al. 2000b; Saikia, Konar & Kulkarni 2006; Bagchi et al. 2014; Banfield et al. 2016), it was found that jet-interruption might have taken place within radio sources with such unique radio morphology. In another example, Solovyov & Verkhodanov (2011, 2014) reviewed a list of GRG candidates, and found 8 radio sources with signatures of galaxy interaction. These objects have X-shape radio morphologies, and are believed to be in the final stage of mergers. To date, there are over 800 GRGs identified in the literature (Kuźmicz et al. 2018; Dabhade et al. 2020a,b; Tang et al. 2020; Delhaize et al. 2021), and the work of hunting for new GRGs is still ongoing.

In this paper we introduce a novel automated method for classifying radio galaxies as giants or non-giants based on a convolutional neural network approach. The structure of the paper is as follows: in Section 2, we review existing approaches to the classification of giant radio galaxies; in Section 3, we explain the theoretical background to the algorithms used in this work; in Section 4, we describe how we selected and built two machine learning data sets with different object class ratios and sample constitutions; in Sections 5 and 6 we compare the model performances resulting from different training strategies and data samples, including an interpretation of different model behaviour and the connection to data selection; we finally draw our conclusions in Section 7.

In this work we assume a ΛCDM cosmology with $\Omega_\mathrm{m} = 0.31$ and a Hubble constant of $H_0 = 67.8$ km s$^{-1}$ Mpc$^{-1}$ (Planck Collaboration XIII 2016). The model training, validation, and testing are performed using the Google Colaboratory (Bisong 2019), which is equipped with a NVIDIA Tesla T4 (14.7 GB memory).

## 2 GRG CLASSIFICATION APPROACHES

Regardless of exact methodology, classifying a radio galaxy as a GRG requires astronomers to (i) identify radio components belonging to the same DRAGN; (ii) measure the source's Largest Angular Size (LAS) from radio maps; (iii) find the corresponding host galaxy of the DRAGN; (iv) measure the host galaxy redshift, and (v) derive the source's projected physical linear size based on the source LAS and host galaxy redshift. In the following section, we recap previous GRG classification methodologies, and highlight the advantages and disadvantages that have motivated the algorithm development presented in this work.

### 2.1 Visual inspection

The majority of historic GRG studies use 'by eye' classification, also known as visual inspection. In these studies, new GRGs were confirmed by following-up sample candidates pointed out by previous studies, or by searching large-scale radio survey catalogues by eye (Willis et al. 1974; Bridle et al. 1976; Waggett, Warner & Baldwin 1977; Laing, Riley & Longair 1983; Kronberg, Wielebinski & Graham 1986; de Bruyn 1989; Ekers et al. 1989; Jones 1989; Lacy et al. 1993; Law-Green et al. 1995; Cotter, Rawlings & Saunders 1996; McCarthy et al. 1996; Subrahmanyan, Saripalli & Hunstead 1996; Ishwara-Chandra & Saikia 1999; Schoenmakers et al. 2000a; Lara et al. 2001; Machalski, Jamrozy & Zola 2001; Sadler et al. 2002; Letawe et al. 2004; Saripalli et al. 2005; Saikia et al. 2006; Huynh, Jackson & Norris 2007; Machalski et al. 2008; Hota et al. 2011; Kozieł-Wierzbowska & Stasińska 2011; Solovyov & Verkhodanov 2011; Bagchi et al. 2014; Molina et al. 2014; Solovyov & Verkhodanov 2014; Amirkhanyan, Afanasiev & Moiseev 2015; Tamhane et al. 2015; Amirkhanyan 2016; Clarke et al. 2017; Dabhade et al. 2017; Kapińska et al. 2017; Prescott et al. 2018; Sebastian et al. 2018; Dabhade et al. 2020a,b; Koziel-Wierzbowska, Goyal & Zywucka 2020; Tang et al. 2020; Delhaize et al. 2021).

Before large-scale radio surveys such as NVSS or FIRST were available, most studies in this field would examine the validity of a GRG candidate by making a deep observation of a particular source or set of sources in specific radio and optical wavebands and would use the optical spectrum of the galaxy host in order to measure source redshift (e.g. Bagchi et al. 2014; Amirkhanyan et al. 2015; Tamhane et al. 2015). When large-scale radio surveys (e.g. NVSS, SUMSS, and FIRST), and optical surveys such as SDSS became available, later studies tended to select source candidates from a particular survey and to perform early cross validation using other survey image data if available. With the availability of photometric redshifts for large numbers of objects from surveys such as SDSS, a number of more recent discoveries have also used photometric redshift when estimating object distances (e.g. Dabhade et al. 2020a,b; Tang et al. 2020). Such approaches have enabled researchers to measure source host redshift and object LAS of these GRGs with excellent reliability, and investigated their spectral properties (e.g. Dabhade et al. 2020a).

In addition to experts, citizen scientists have also recently joined the hunt for GRGs. Radio Galaxy Zoo (RGZ; Banfield et al. 2015), an online citizen science project, aims to cross match radio sources at 1.4 GHz with their infrared host galaxies. Although the project was initially launched to create a large scale radio galaxy catalogue with the help of citizen scientists, its online forum *RadioTalk* allowed citizen scientists to collaborate with project team scientists and find radio galaxies of special types. Four out of the six radio galaxies confirmed as GRGs by the RGZ team were pointed out on the forum as GRG candidates by several project citizen scientists in advance of their confirmation (Banfield et al. 2016; Tang et al. 2020).

### 2.2 Automated searches

Each of the GRG identification processes described in the previous section relies largely on visual inspection and manual analysis. These methods are widely accepted as they allow for cross validation with diverse complementary radio/optical survey images, deep source imaging, and/or spectral confirmation. The consensus level of such approaches is therefore extremely high. However, while such approaches work well when dealing with source catalogues such as NVSS, SUMSS, and FIRST, with modest sample sizes, such methods become impractical when faced with millions rather than thousands of candidate galaxies. For example, the Evolutionary Map of the Universe (EMU; Norris et al. 2011) survey, one of the Australia SKA Pathfinder (ASKAP; Johnston et al. 2008) early science projects, is expected to provide a catalogue containing approximately 70 million sources (Norris et al. 2011, 2021), of which ~7 million will require visual inspection. Reliable automated GRG search methods therefore become necessary in the era of astronomical big data.

One recent GRG search method (Proctor 2016) attempted to challenge the traditional visual inspection methods by using a decision tree based machine learning approach. The study focused on sources with only two radio components and aimed to identify GRGs among them, regardless of component morphology (lobes, jets, and core). The training set for this work consisted of 51 195 source pairs from the NVSS catalogue, of which 48 had previously been confirmed to be GRGs by Lara et al. (2001). When selecting sample features, the study mostly used component major axis, minor axis, and peak flux as input features. The best classifier in this study achieved a training accuracy of 97.8 ± 1.5 per cent. Proctor (2016) then used the best classifier to find GRGs from 870 000 candidate pairs in the full NVSS catalogue, extracting those objects with high GRG probabilities. Such a semi-automated procedure produced a list of 1 616 GRG candidates with LAS $\geq$ 4 arcmin.

Although this pioneering study predicted a large number of GRG candidates, it did not consider source host galaxy redshift as this was unavailable in the NVSS catalogue. Consequently, only 16 of the selected candidates were included in the later updated catalogue of GRGs by Kuźmicz et al. (2018). This catalogue assumes $H_0 = 71\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$, $\Omega_M = 0.27$, $\Omega_{\mathrm{vac}} = 0.73$, and lists 349 confirmed GRGs as of the end of 2018. The catalogue uses host galaxy redshifts either from the literature or from SDSS.

Limited by the lack of host galaxy redshift information, validation of the sample from Proctor (2016) was not comprehensively addressed until Dabhade et al. (2020b) performed a follow-up study. Thanks to the availability of source coordinates in the Proctor (2016) candidate catalogue, the team was able to track positions for each candidate, manually visualize their NVSS, FIRST, The GMRT 150 MHz All-sky Radio Survey (TGSS; Intema et al. 2017), and the Karl G. Jansky Very Large Array Sky Survey (VLASS; Lacy et al. 2020) images (if available), and also check their host galaxy redshift from publicly available optical surveys and data bases (Dabhade et al. 2020b). Source LAS were measured from NVSS images for uniformity, where only emission above a $3\sigma$ level was considered. The team found that there were 165 known and 151 newly discovered GRGs among the candidates. In other words, around 20.8 per cent of the candidates in the list were finally confirmed to be GRGs.

Although the algorithm of Proctor (2016) did not predict a fully reliable set of GRGs from the test sample, it had successfully produced a good candidate pool. The availability of traceable source coordinates allowed follow up work to perform traditional visual inspection and cross validation. Such availability further allowed them to check sample host galaxy redshifts, as source LAS and redshift are the two key factors to determine whether a source is a GRG. As a result, the Proctor (2016) candidate list has contributed more than 35 per cent of all sources to the total confirmed GRG popularization.

In spite of its success, the algorithm presented in Proctor (2016) also raises questions about selection biases: for example, in this instance only sources with two radio components and large LAS ($\geq$4 arcmin) are considered. In other words, the Proctor (2016) algorithm considered only those radio sources with a host galaxy redshift larger than $z = 0.17$. Such selection biases would have excluded at least 108 GRGs of smaller LAS in the Kuźmicz et al. (2018) catalogue. Also, it is problematic that GRGs with more complicated morphologies could not be recognized by the Proctor (2016) algorithm as these are considered important for particular types of investigation, as described in Section 1. In light of these considerations, in this work we present a GRG classifier capable of identifying GRGs of smaller LAS and with diverse radio morphologies using an approach based on Convolutional Neural Networks (CNN; Krizhevsky, Sutskever & Hinton 2012). Considering the traditional approaches to GRG candidate validation using multifrequency radio survey data, we also explore the possibility of using multisurvey image data and host galaxy redshifts as algorithm inputs.

## 3 CONVOLUTIONAL NEURAL NETWORKS

In recent years, CNNs have become widely used for astronomical image pattern recognition related problems, such as galaxy cluster and filament detection (Gheller, Vazza & Bonafede 2018), supernovae classification/detection (e.g. Kimura et al. 2017; Chan, Heng & Messenger 2020), compact and extended radio galaxy distinction (Lukic et al. 2018), and radio galaxy localization (Wu et al. 2019). CNNs are popular as they can decompose 1D or 2D inputs into partly overlapped patches, where each cortex neuron only captures features from a specified patch (Matsugu et al. 2003; Kiranyaz et al. 2019). Also, CNNs are weight sharing, allowing them to be translation equivariant and therefore robust to offsets in radio images.

In the context of radio galaxy morphology classification, most state-of-the-art CNN applications have concentrated on single image inputs. Specifically, both training and test samples typically come from only one radio sky survey (Aniyan & Thorat 2017; Alhassan, Taylor & Vaccari 2018; Lukic et al. 2018; Tang, Scaife & Leahy 2019; Bowles et al. 2021; Scaife & Porter 2021), or have radio source contours overlaid on infrared sky survey maps (Alger et al. 2018; Wu et al. 2019). Notably, Alger et al. (2018) trained their single image input CNN along with 10 derived features to cross-identify radio galaxies and their corresponding host galaxies. Although they themselves did not highlight it in their text, this algorithm could be regarded as the first multidomain CNN in the field of radio galaxy classification.

More broadly in astronomy, AstroNet (i.e. Shallue & Vanderburg 2018) is the earliest and the most well-known multibranch CNN (Li et al. 2017) application. AstroNet is multibranch neural network which is used to find exoplanet candidates from light curves observed by NASA's `Kepler Space Telescope` (Koch et al. 2010). The authors imported both 'global view' (fixed-length complete light curve) and 'local view' (fixed-length window over the detected transit) data of the observed exoplanet light curve into their algorithm, and successfully identified two new exoplanet candidates. Following AstroNet, several other deep-learning applications also used the multibranch strategy have been made to detect exoplanets (i.e. Ansdell et al. 2018; Osborn et al. 2020) or fast radio bursts (FRB; i.e. Connor & van Leeuwen 2018).

Considering the conventional procedure of GRG identification, the proposed algorithm needs to be able to learn from multiple radio survey images, which potentially could be achieved by the aforementioned multibranch or multidomain networks. These algorithms could ideally concatenate the image features learned from the convolutional layers with parametrized features such as host galaxy redshift and provide a combined input to a subsequent model layer (e.g. a fully connected layer). Such CNN-based algorithms that are able to concatenate features learned from different inputs are referred to as multidomain multibranch CNNs.

### 3.1 Multibranch CNNs

Concepts similar in nature to that of multibranch neural networks can be found from 2016, when Cheng et al. (2016) announced an algorithm for recommender systems based on feed-forward neural networks. They referred to their approach as 'Wide and Deep Learning', where 'wide' refers to wide linear models and 'deep' represents deep neural networks. Similar approaches using CNNs as a backend soon appeared, including Amerini et al. (2017), which attempts to merge two independent top-down CNNs. That work initially developed a network architecture that they referred to as a multidomain CNN to localize JPEG double compression. In their model, a spatial domain CNN (2D input) and a frequency domain CNN (1D input) had their outputs passed to two fully connected layers, the outputs from which were then concatenated into a larger fully connected layer. The outputs from this layer then served as an input to subsequent layers. Such an architecture allowed the model to learn all features jointly through end-to-end backpropagation and it was shown that such approach could provide superior model performance for specific problems compared to a traditional spatial domain CNN (Amerini et al. 2017).

Inspired by the Amerini et al. (2017) architecture, several multibranch CNN algorithms have subsequently been developed and are being used in computer vision and medical research (Li et al. 2017; Schilling et al. 2018; Aslani et al. 2019; Cao et al. 2020). Thanks to the availability of feature concatenation, these studies were able to do bulk model training on different image inputs, different channels of the same image data cube, or have each branch trained on the same image data using different kernel sizes. Multibranch networks that use multiple inputs from different sources, such as 2D image data and 1D spectral data, are also referred to as *multidomain* CNNs as they perform joint learning on data from different domains.

In the context of GRG classification, it is essential for an algorithm to be able to estimate source LAS from 2D image inputs and to also use the source host redshift as a scalar feature in order to link the LAS to the physical size of the object. Thanks to the concatenation feature of multibranch CNNs, such multidomain inputs can be joined together in a bulk training process either from the top of their corresponding branches or simply from the concatenation layer. This makes multibranch CNNs an ideal candidate algorithm for developing automated GRG object classifiers. In the following section, we describe the multidomain data set constructed for use in this work and in Section 5 we describe the multibranch model architectures considered.

## 4 DATA SAMPLE CONSTRUCTION

Our models should not only be able to distinguish differences in source angular extent, but furthermore be able to find differences in physical linear sizes. Therefore, the data set sample selection in this work needs to build a data set that includes confirmed GRGs and radio galaxies with smaller sizes. The data sample selection should also follow the rules of data set foundation for training Multi-branched CNNs. Consequently, the data sets used in this work were selected in accordance to the following criteria:

**Data availability**: All data samples should include (i) image survey data from both the NVSS and FIRST surveys, (ii) host galaxy redshifts, (iii) largest angular size (LAS) measurements, and (iv) have a physical linear size calculated.

**Source-image relationship**: Image data from each radio survey should have the same image size in terms of angular size.

**Image pre-processing**: After pre-processing images should only contain positive-valued pixels and the source should be visible in the image.

**Traceability**: Users should be able to trace the source coordinates, catalogued object ID, and original source catalogue for each sample.

With respect to data availability, in this work we only include samples that have both NVSS and FIRST data available. We do not require the data availability of LOFAR Two-metre Sky Survey Data Release 1 (LoTSS DR1; Williams et al. 2019) conducted by the LOw Frequency ARray (LOFAR; van Haarlem et al. 2013). This is because LoTSS DR1 covers only 424 square degrees and this would significantly limit data availability for the samples selected in Section 4.1. Data traceabilty is implemented not only for the convenience of model training and testing, but allows both users and developers to evaluate and explain model outcomes based on their scientific understanding of the data. For example, in this work, we use data traceability in Section 6 to explain misclassifications with respect to different models.

### 4.1 Source sample selection

#### *4.1.1 Radio galaxies of smaller sizes*

Radio galaxies with non-giant dimensions were selected from Data Release 1 of Radio Galaxy Zoo (RGZ DR1; Wong et al., in preparation). RGZ DR1 is a radio galaxy catalogue created by over 12 000 volunteers through the RGZ citizen science project. Project users are asked to cross-match radio source lobes with a corresponding infrared host galaxy. The radio images come mainly from the FIRST survey, and the infrared images are largely 3.4 μm *WISE* images. These radio and infrared images share a 3 × 3 arcmin field of view. RGZ DR1 includes 75 641 cross-matched identifications (Wong et al., in preparation).

A previous investigation of the DR1 catalogue has shown that the uniform 3 × 3 arcmin image size constrains its ability to identify GRGs (Tang et al. 2020). However, the catalogue does provide a large source sample with full data availability and traceability, as required for the construction of a machine learning data set. Previous analysis of the full RGZ DR1 catalogue found that at least 11 237 non-duplicated samples fulfill the requirements for training data set selection outlined at the start of Section 4.1. These samples have LAS from 2 to 195 arcsec, and are all within the 3 × 3 arcmin field of view.

In addition to the source-image relationship required for each data set sample, we further require that the radio centroid and host galaxy position should be consistent for each galaxy, as asymmetries in sources positions are known to exist among the RGZ DR1 catalogued samples (Wong et al., in preparation). Consequently, we only retain sources with an angular separation between the host galaxy and estimated radio centre smaller than 1.8 arcsec, which is the pixel size of FIRST survey images (Becker et al. 1995). This criterion is to ensure sample symmetry and reduces the 11 237 RGZ DR1 samples to 6021 samples. On inspection, we also removed the known GRG source GRG J1402 + 2442 from the sample, as we only consider objects with smaller sizes ($\leq$700 kpc) in the RGZ DR1 sample. Finally, we examined the survey image data availability of NVSS and FIRST using the SkyView API query `astroquery.skyview.get_image_list`, to confirm that all sources had the required image data.

### *4.1.2 Giant Radio Galaxies*

The GRG sample for this work comes from the Kuźmicz et al. (2018) and Dabhade et al. (2020a) catalogues. Kuźmicz et al. (2018) performed a detailed review of the literature identifying 349 GRGs, of which 89.7 per cent are FR II objects. The catalogue has its sources validated using the NVSS at 1.4 GHz and measures their flux densities using those data. However, we note that primary image data used for GRG identification in the literature covers a wide range of image angular resolutions, from arcsec level to 45 arcsec (e.g. LoTSS, NVSS, and SUMSS).

Dabhade et al. (2020a), on the other hand, performed an independent GRG search using the Value Added Catalogue (VAC; Williams et al. 2019) of LOFAR. The team identified 239 GRGs, including 225 which were previously unknown. The newly discovered GRGs in the catalogue of Dabhade et al. (2020a) have their candidates identified from LoTSS survey images at 151 MHz with 6 arcsec resolution. They cross-validated their sources using the FIRST, WENSS, and TGSS surveys (Dabhade et al. 2020a). The difference in GRG sample selection and cross-validation between these samples will allow us to compare model behaviour when using samples selected with different class definitions. We discuss this further in Section 6.2.

From these catalogues we found that 310 GRGs in the Kuźmicz et al. (2018) catalogue have NVSS images available, and 186 also have FIRST images. All newly discovered GRGs in the Dabhade et al. (2020a) sample have both NVSS and FIRST images available. Considering that the maximum source LAS in the RGZ DR1 entries is 195 arcsec, we further require the GRG samples to have LAS equal or smaller than the DR1 LAS limit. This reduces the samples to 58 GRGs from the Kuźmicz et al. (2018) catalogue and 167 GRGs from the Dabhade et al. (2020a) catalogue.

## 4.2 Image pre-processing and further sample selection

In this work all image data are obtained using the Skyview Virtual Observatory.[1] Consistent with the RGZ DR1, we define our FIRST image data to have a field of view with a uniform 3 × 3 arcmin size. This is equivalent to 100 × 100 pixels for the FIRST images where 1 pixel = 1.8 arcsec. Since NVSS has an angular resolution of 45 arcsec, we define 18 × 18 pixels dimensions for the NVSS images, where 1 pixel = 15 arcsec, in order to avoid truncating objects with large angular extents that have emission close to the image boundaries. We acquire both the NVSS and FIRST postage stamp images in FITS format, defining the image centres using the host galaxy position from RGZ DR1.

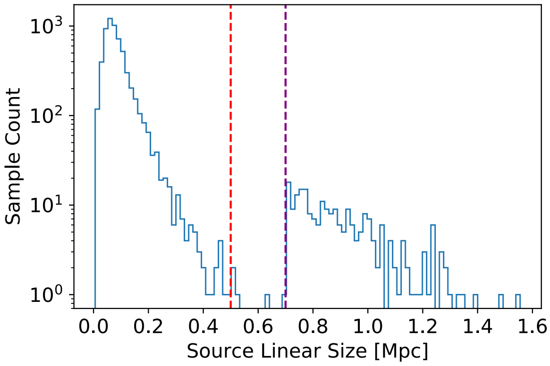


**Figure 2.** Bulk sample projected linear size distribution after we performed sample section procedure in Section 4.1 and Section 4.2. The red and blue dashed lines on the diagram indicate projected linear sizes of 500 and 700 kpc, respectively.

The original FITS images are linearly scaled, and have units of Jy beam$^{-1}$. Following the literature, we then subject each FITS image to a series of pre-processing steps before use. Aniyan & Thorat (2017) highlighted the importance of image noise reduction, and replaced all image pixels with values lower than a specified noise threshold with zeros, giving the sample images cleaner backgrounds and enabling neural networks to train with high sparsity. Specifically, they proposed this sigma-clipping for each sample image to be at the $3\sigma_{\rm rms}$ level. Later studies showed that applying sigma-clipping at a 3, 4, or $5\sigma_{\rm rms}$ level did not significantly change the model outcome, and the approach of sigma-clipping generally has been applied successfully by a number of applications (e.g. Aniyan & Thorat 2017; Tang et al. 2019).

In this work, we also follow the method of Aniyan & Thorat (2017) and sigma-clip the images individually at a level of $3\sigma_{\rm rms}$. We then apply the following image normalization to each of the images:

$$\mathrm{Output} = \frac{\mathrm{Input} - \mathrm{Min}}{\mathrm{Max} - \mathrm{Min}} \times (255.0 - 0.0). \qquad (1)$$

Following this pre-processing, we found that some radio sources with very low signal to noise ratios were eliminated. We found that 15, 7, and 15 objects from the DR1, Kuzmicz and Dabhade source catalogues, respectively, had at least one of the FIRST or NVSS images result in an empty field of view and consequently these objects were removed from our sample.

Fig. 2 shows the size distribution of the remaining objects, where the red dashed line indicates a source size of 500 kpc and the blue line represents 700 kpc, the GRG size cut-off. Only four objects have linear sizes intermediate to these two values and, for clarity, we exclude these four intermediate sources from the data set and define the two target classes of radio galaxy in this work to be:

(i) NOM: Radio galaxies with linear sizes smaller than 500 kpc.
(ii) GRG: Radio galaxies with linear sizes larger than 700 kpc, consistent with the standard definition from the literature.

Following the pre-processing described above, the sample contains 6001 radio galaxies of class NOM, and 205 of class GRG. Since the two classes have a clear difference in linear size, in principle a good classifier should be able to distinguish them well. All of these objects are centred in their images. Data for each sample also includes source object ID, qualified host galaxy redshift, LAS measurement, and computed source linear size. However, as described in Section 5,

[1] https://skyview.gsfc.nasa.gov

**Table 1.** A summary of the sample division of the GRGNOM-A and GRGNOM-B. 'Count' refers to the total source sample number of a class, and 'Total' represents the sample number of each column.

| GRGNOM-A | NOM | GRG | | Total |
|---|---|---|---|---|
| | RGZ DR1 | Kuzmicz | Dabhade | |
| **Training** | 561 | 39 | 0 | 600 |
| **Testing** | 187 | 13 | 0 | 200 |
| Count | 748 | 52 | 0 | 800 |
| **GRGNOM-B** | **NOM** | **GRG** | | Total |
| | **RGZ DR1** | **Kuzmicz** | **Dabhade** | |
| **Training** | 447 | 52 | 101 | 600 |
| **Testing** | 149 | 0 | 51 | 200 |
| Count | 596 | 52 | 152 | 800 |

only image data and redshift information are used as model inputs for the classifiers evaluated in this work.

### 4.3 Data formatting, division, and summary

Although there are ∼6000 sample sources of class NOM, the data set would be extremely imbalanced if we use all of them when building our training/testing set. In consideration of the observationally imbalanced nature of GRGs and sources with smaller sizes, we use these data to build a modestly imbalanced data set of 600 training samples and 200 testing samples (3 to 1 ratio), with GRG samples extracted from the Kuźmicz et al. (2018) catalogue only, and samples of class NOM taken from RGZ DR1. The resulting data set is named GRGNOM-A, and has a NOM:GRG class balance of ∼14:1. Given the availability of the Dabhade et al. (2020a) samples, we also build a second dataset named GRGNOM-B using 204 out of 205 GRGs in the selected data samples. The 204 objects would be separated into 153/51 GRG samples in the resulting GRGNOM-B training/testing set. Sources in both class GRG and NOM in the GRGNOM-B data set then would follow the 3 to 1 data sample ratio. For either GRGNOM-B training set or testing set, source NOM:GRG sample ratio in each subset is around 2.9: 1. Given that the test set of GRGNOM-A only contains Kuźmicz et al. (2018) samples, we only include Dabhade et al. (2020a) samples in the GRGNOM-B test set in order to allow models trained using either data set to test their generalization ability. The resulting data samples are summarized in Table 1 and Fig. 3.

It can be seen that the GRGNOM-B training sample is dominated by Dabhade et al. (2020a) samples primarily identified at 151 MHz, while the GRGNOM-A data set only includes Kuźmicz et al. (2018) samples identified at 1.4 GHz. In particular, the test set of GRGNOM-B only contains samples from the Dabhade et al. (2020a) catalogue, in order to see if the features learned from Kuźmicz et al. (2018) samples can facilitate the identification of Dabhade et al. (2020a) samples when looking at their 1.4 GHz radio image data. The data sample construction also gives those models trained with GRGNOM-B a chance to test their generalization ability upon samples with their identification, source LAS, and host galaxy redshifts measured in a uniform manner.

In terms of data format, both GRGNOM-A and GRGNOM-B are split into two parts: (i) text-format tables of numerical source information, e.g. Table 2 405.0pt, and (ii) a group of machine readable documents containing feature data in various formats. This second component of the data set is comprised of:

(i) **FIRST images**: The pre-processed FIRST survey grey-scale images with a universal size of 100 × 100 pixels. Three versions of these image files are generated:

(a) **Image batch files**: Images are saved in 4 batched files (3 training batches and 1 testing batch), along with a metadata file containing corresponding image header information. These files are in a format that is understandable for our `Pytorch` models. These files are saved in a folder named FIRST.

(b) **Encoded compressed file**: This is the encoded compressed file of **a**. The compressed file is named as FIRST.tar.gz. Creating such a data file allows future developers to download and re-use the image data samples for machine learning training.

(c) **Individual images**: These images are saved in another image folder named `FIRST_IMG`. Image names follow the format: 'Catalogue Name_Catalogue Source No_Right Ascension_Declination.png'.

(ii) **NVSS images**: The pre-processed NVSS survey grey-scale images with a universal size of 18 × 18 pixels are saved in the same manner as the FIRST images.

(iii) **Source host galaxy redshift**: Consistent with the image data, numerical source host galaxy redshifts are separated into 4 batches and saved as `numpy` arrays.

(iv) **Source LAS**: Numerical source LAS are organized and saved in the same way as (iii).

(v) **Source linear size**: Numerical source linear size are organized and saved in the same way as (iii).

(vi) **Source object ID**: Primary object ID of each source sample in their original catalogues. These data strings have been encoded in the `uint8` format and saved in the same way as (iii). As necessary, they can be decoded in the `utf-8` format.

(vii) **Class label**: Numerical class label of each data sample: 0 and 1 represent source classes NOM and GRG, respectively. They are organized and saved in the same way as (iii).

By using the PYTHON `pickle` package, we built our training/testing data sets with these components, which allow users to call any of the source data samples in the data set. Hash values were generated separately for (i) a, (i) b, (ii) a, and (ii) b to protect their integrity and avoid manipulation.

We note that although source LAS and physical size are included in the data set for completeness, this information is not used to train any of the models in this work.

### 4.4 Data normalization and augmentation

In order to improve the convergence of model training, it is recommended to define a data normalization and augmentation strategy (LeCun et al. 2012). Normalization constrains data values within a given range, reduces skew and therefore speeds up the training process of a model. In this work, we require the image data to be normalized to have both a mean and standard deviation of 0.5 before importing to a model, which constrains image pixel values largely within the range from 0 to 1.

In the case of data augmentation, we apply horizontal flipping and image rotation in this work (i.e. Fig. 4). Specifically, we perform horizontal flipping of every image sample with a random probability of 50 per cent. We then further rotate each image in a clockwise manner using a randomly selected angle from $-180^{\circ}$ to $180^{\circ}$, where the rotated angle should be an integer in units of degrees.

Data augmentation is performed dynamically during training as the data are imported to the model. This strategy both ensures the model

**Table 2.** The first 10 rows of the GRGNOM-A training sample catalogue. Source object ID, RA/Dec., host galaxy redshift ($z$) and LAS are extracted from RGZ DR1, while the source linear size is derived from the $z$ and LAS of each sample based on the cosmological parameters defined in Planck Collaboration XIII (2016). Class labels are defined as described in Section 4.2.

| Object ID | RA (J2000.0) | Dec. (J2000.0) | $z$ | LAS | Size | Label |
|---|---|---|---|---|---|---|
| | (h:m:s) | (d:m:s) | | (arcsec) | (Mpc) | |
| RGZJ000606.0+013125 | 00:06:06.07 | 01:31:25.20 | 0.23372 | 21 | 0.079 | NOM |
| RGZJ000626.4+081838 | 00:06:26.41 | 08:18:38.49 | 0.41540 | 18 | 0.102 | NOM |
| RGZJ000627.2+060407 | 00:06:27.21 | 06:04:07.29 | 0.30091 | 14 | 0.064 | NOM |
| RGZJ000746.4+031938 | 00:07:46.46 | 03:19:38.99 | 0.29194 | 44 | 0.200 | NOM |
| RGZJ000851.0+045243 | 00:08:51.01 | 04:52:43.92 | 0.34255 | 19 | 0.095 | NOM |
| RGZJ000911.0+145105 | 00:09:11.05 | 14:51:05.07 | 0.36832 | 20 | 0.105 | NOM |
| RGZJ001042.9+091917 | 00:10:42.92 | 09:19:17.44 | 0.15308 | 19 | 0.052 | NOM |
| RGZJ001051.0+141655 | 00:10:51.09 | 14:16:55.86 | 0.31507 | 17 | 0.079 | NOM |
| RGZJ001146.7+101528 | 00:11:46.75 | 10:15:28.45 | 0.22175 | 17 | 0.064 | NOM |
| RGZJ001524.2+143038 | 00:15:24.23 | 14:30:38.83 | 0.22668 | 33 | 0.122 | NOM |

**Table 3.** A summary of the modified LeNet-5 architecture used in this work as a base architecture. IC refers to the independent component (Chen et al. 2019) and BN represents batch normalization.

| Layer No. | Layer type | Input channels | Output channels | Kernel size | Stride | Activation | Regularization |
|---|---|---|---|---|---|---|---|
| 1 | Convolutional | 1 | 6 | 5 | 1 | ReLU | **IC/BN** |
| 2 | Max pooling | 6 | 6 | 2 | 2 | | |
| 3 | Convolutional | 6 | 16 | 5 | 1 | ReLU | **IC/BN** |
| 4 | Max pooling | 16 | 16 | 2 | 2 | | |
| 5 | Convolutional | 16 | 120 | 5 | 1 | ReLU | **IC/BN** |
| $5'$ | Squeeze layer 5 outputs | | | | | | |
| 6 | Fully connected | 120 × Down-sampled neuron number | 120 | | | ReLU | Dropout |
| 7 | Fully connected | 120 | 84 | | | ReLU | Dropout |
| 8 | Fully connected | 84 | 2 | | | Softmax | |

has a large enough number of data samples to learn and minimizes memory usage. Using this approach, the augmented training data set has a size of 600 × 2 × 360 = 432 000 samples when the model is trained for 720 epochs.

## 5 NETWORK ARCHITECTURE

In this work we consider five different network architectures to create seven different models. These are summarized in Table 4. The first network architecture is a traditional, or classical, CNN approach that takes a single source of image data as an input, this forms the basis for Architecture A (NVSS) and Architecture B (FIRST) in Table 4. The second form of network is a multidomain architecture that takes a single source of image data plus redshift information as its inputs, this forms the basis for Architecture C (NVSS + $z$) and Architecture D (FIRST + $z$) in Table 4. The third form of network is a multidomain network that takes multiple sources of image data as inputs, this forms the basis for Architecture E (NVSS + FIRST), and the fourth form of network is the expansion of this architecture to include redshift as an input, Architecture F (NVSS + FIRST + $z$). The final form of network is Architecture G in Table 4, which has the same inputs as Architecture F but replaces all convolutional layers with Inception Modules, see Section 5.3.

### 5.1 Classical CNN

Zhu et al. (2014) were able to train their pulsar identification algorithms with 3756 labelled 48 × 48 images samples using a slightly modified LeNet-5 CNN architecture. LeNet-5 is one of the earliest Convolutional Neural Networks, demonstrating high success in digit/character recognition tasks (Lecun et al. 1998). The network has a simple 7-layer architecture, with 2 convolutional layers, 2 pooling layers, and 3 fully connected layers. In this work, we start with a modified version of LeNet-5, including one extra convolutional layer, see Table 3. This extra convolutional layer is followed by a down-sampling that differs between the FIRST and NVSS survey images: FIRST images are downsampled to 25 × 25 = 625, while NVSS images are downsampled to 9 × 9 = 81. Rather than using the mean squared error (MSE) originally proposed for LeNet-5, we use the now more common cross-entropy loss function to train our logistic regression algorithms. The layers shown in Table 3 form the base architecture of all networks used in this work and we will refer to specific layer numbers from Table 3 whenever we manipulate or replace any functionality in the following sections.

Although such a network is sufficient to train a model, previous deep learning attempts at classifying radio galaxy morphology have applied additional regularization methods in order to improve their model generalization (test) error and avoid model overfitting (Goodfellow, Bengio & Courville 2016). In this work, we apply the independent component (IC) layer regularization strategy of Chen et al. (2019) to all convolutional layers in our network and this is described in more detail in the following section. In addition, we include a dropout layer before each fully connected layer, with the exception of the output layer. We note that AlexNet (Krizhevsky et al. 2012), a well-known CNN architecture, has previously been used to classify radio galaxy morphologies (Krizhevsky et al. 2012; Aniyan & Thorat 2017). Following hyperparameter optimization, we found that AlexNet requires 3–4 times the training time per epoch compared to the modified LeNet-5 used in this work, and does not provide comparable or improved model performance.

**Table 4.** The summary of architectures we adopted in this work. *Convolution Branches* refers to the number of independent top-down architectures from Layer 1 to Layer 5 arcmin in Table 3.

| Architecture | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| Input data | **NVSS** | **FIRST** | **NVSS** and $z$ | **FIRST** and $z$ | **NVSS** and **FIRST** | **NVSS**, **FIRST**, and $z$ | **NVSS**, **FIRST**, and $z$ |
| Convolution Branches | 1 | 1 | 1 | 1 | 2 | 2 | 2 |
| Layer 6 input (**NVSS**) | 120 × 9 × 9 | | 120 × 9 × 9 | | 120 × 9 × 9 | 120 × 9 × 9 | 128 × 9 × 9 |
| Layer 6 input (**FIRST**) | | 120 × 25 × 25 | | 120 × 25 × 25 | 120 × 25 × 25 | 120 × 25 × 25 | 128 × 25 × 25 |
| Layer 7 input | 120 | 120 | 120 + 1 | 120 + 1 | 120 + 120 | 120 + 120 + 1 | 120 + 120 + 1 |
| Extra FC | No | No | No | No | Yes | Yes | Yes |
| Branch Module | Nil | Nil | Nil | Nil | Nil | Nil | Inception modules |

### 5.2 Independent component layer

As deep learning develops, one important issue is how to train complex networks with higher efficiency (Ioffe & Szegedy 2015; Chen et al. 2019). Among all the techniques available, Batch Normalization (BN; Ioffe & Szegedy 2015) and Dropout (Srivastava et al. 2014) are perhaps most frequently used by radio galaxy related deep learning approaches (e.g. Aniyan & Thorat 2017; Ma et al. 2019).

Batch normalization is able to normalize the net activations and have their mean and unit variance become zero (Chen et al. 2019). The purpose of applying such an approach is to reduce the internal covariate shift, in other words the change in the distribution of network activations due to the change in the network parameters during training (Ioffe & Szegedy 2015). The technique therefore is able to speed up network training, regularize model performance, and further induce a stable predictable behaviour during gradient descent (Santurkar et al. 2018).

Dropout, on the other hand, performs regularization in a different way. It introduces random gates for all inputs to a given layer, where each neuron has a probability, $p$, to be set to zero. Such a measure is able to remove weakly connected neurons, and has been demonstrated to regularize network performance and prevent neuron co-adaptation (Srivastava et al. 2014).

The Independent Component layer (IC) is a recently developed technique incorporating both of these techniques that has been proposed to boost model training efficiency and improve model stability (Chen et al. 2019; Li et al. 2019). Each IC layer contains a stacked combination of BN and Dropout layers, see Fig. 5, which has been proven to be able to reduce the mutual information and the degree of correlation between any pair of neurons. Such techniques achieve more stable training behaviour and IC networks typically have their generalization ability improved (Chen et al. 2019). A recent approach put IC layers before the activation layers (Li et al. 2019), and found that such method could boost model performance compared with those which only insert a BN layer between a convolutional layer and an activation function.

Inspired by Li et al. (2019), in this work we replace each convolutional layer with the combination of a convolutional layer, an IC layer and an activation function and refer to this as a *Conv module*. This is illustrated in Fig. 5. We did not use the Chen et al. (2019) strategy as to maintain image input completeness. We also implemented the popular Conv->BN->ReLU regularization strategy (e.g. He et al. 2016; Tang et al. 2019) for further comparison.

### 5.3 Inception module

The *Inception module* was created by Szegedy et al. (2014) and is named following a previous approach referred to as Network in Network (NIN; Lin, Chen & Yan 2013). Unlike classical CNN approaches that have all of their convolutional layers stacked sequentially, the Inception module has 4 ‘branches’: a convolutional layer with limited kernel sizes of 1 × 1, 3 × 3, 5 × 5 and a 3 × 3 max pooling layer. The four branches in each module operate in parallel on the same input feature map. Outputs from each ‘branch’ then are concatenated together and serve as the input to the next layer, see Fig. 5.

In order to give the network improved representation power, Szegedy et al. (2014) further introduced a 1 × 1 convolutional layer both before the 3 × 3 and 5 × 5 convolutional layers, and after the max pooling layer. An Inception module with these additional layers is known as an *Inception module with dimension reduction* (Szegedy et al. 2014). The addition of these 1 × 1 convolutional layers can reduce the number of input channels, hence lowering the computational complexity by serving as dimension reduction module, whilst increasing network depth at the same time.

The first architecture equipped with the Inception module with dimension reduction was GoogLeNet, the winner of the ImageNet Large-Scale Visual Recognition Challenge 2014 (ILSVRC14; Russakovsky et al. 2015). After the success of GoogLeNet, this module was also used in astronomical studies such as supernovae classification (Brunel et al. 2019) and Faraday spectra classification (Brown et al. 2019), and the mapping between simulation based galaxy cluster distributions and the underlying dark matter distribution (Zhang et al. 2019). In this work, we explore its potential when creating Architecture G, see Table 4 and Section 5.5.

### 5.4 Multidomain CNNs

Although sources with class NOM and GRG are clearly separated in physical linear size, see Fig. 2, their LAS and host galaxy redshift distributions are not as distinct, see Fig. 3. Typically, the source LAS of the two classes are generally separated, while it can be seen that host galaxy redshift distribution of the two classes are significantly overlapped. Moreover, a source might exhibit different emission structure between its NVSS and FIRST images due to structure appearing on a range of scales. To account for these possibilities, we modify the original network architecture, allowing multiple inputs to train together. Architectures with multiple inputs are also known as *multidomain* neural networks, and were originally proposed by Amerini et al. (2017). They introduced both spatial domain data (2D) and frequency domain data (1D) as network inputs, with each domain training on an independent branch with their outputs concatenated into a single fully connected layer. For such a multidomain network, model inputs can be 2D images, 1D arrays, or scalars. Networks with multiple inputs are sometimes considered to be a variant of multibranch networks, and we discuss this further in Section 5.5.

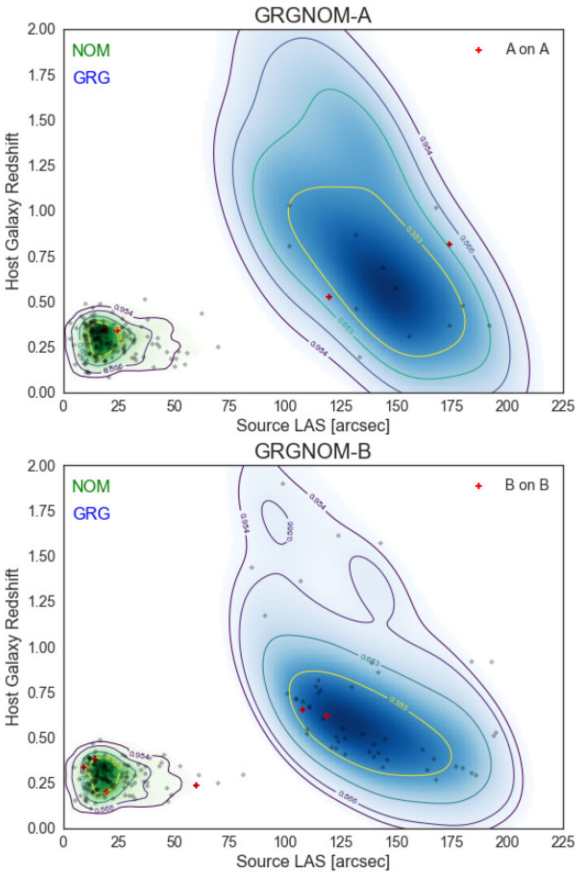


**Figure 3.** Upper: the LAS versus host galaxy redshift density map of the GRGNOM-A data set. Training samples of class NOM and GRG are represented in green and blue, respectively. Contours on the diagram refer to the iso-proportion of the density, i.e. 95.4 per cent of the probability mass lies within the 0.954 contour. Contours are shown at 0.383, 0.683, 0.866, and 0.954 in the figure, which correspond to $0.5\sigma$, $1\sigma$, $1.5\sigma$, and $2\sigma$. Grey data points are the GRGNOM-A test sample data points and red data points on the diagram indicate samples that are frequently misclassified by models of Architecture G trained and tested on the GRGNOM-A data set. Lower: equivalent distributions for the GRGNOM-B data set. The red data points indicate samples that are frequently misclassified using Architecture G trained and tested on the GRGNOM-B data set. This diagram was plotted using the `pyrolite` (Williams et al. 2020) package.

### 5.4.1 *Including source redshift*

Source redshift, $z$, as a numerical parameter, is one of the two key parameters used when identifying GRGs, as it determines the distance to the radio galaxy and hence the conversion of projected angular size to projected physical size. It is therefore intuitive to include source redshift into our models. However, given that numerical data cannot be passed through layers expecting 2D inputs, we specify for this parameter to join the training at Layer 7, see Table 3. Specifically, we concatenate the down-sampled outputs from Layer 6 of the network with the source redshift, an example of which is shown in Fig. 6. Source redshifts are normalized to lie in the range 0–1, consistent with the normalization of image feature data. This normalization is discussed further in Section 6.2.4.

### 5.4.2 *Multiple Image inputs*

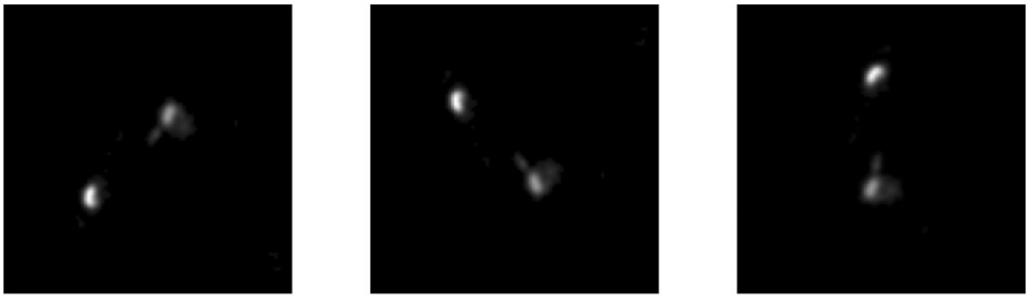
In addition to combining image data with numerical features such as redshift, we also expand the multidomain approach further to include additional image inputs. Using this approach, images of radio galaxies observed by multiple surveys with different angular resolutions are able to be learned together. Such an approach has previously been considered with in the field of neuroscience (Aslani et al. 2019).

In order to implement this strategy, we combine elements of the two CNNs described in the previous section together, see Fig. 6. The depth of linear neuron concatenation remains the same as in the *Image* + $z$ methods. By using the resulting architecture, both NVSS and FIRST images of a single source will have 120 features extracted from each survey that are then concatenated and passed into the final two fully connected layers. In this scenario we add an additional fully connected layer with size 120 after Layer 6, in order to increase the learning ability of the network given the larger volume of input data as well as to regularize the algorithm further.

With this architecture it is trivial to also include source redshift, $z$, see Fig. 6. By adding the source redshift in the same way as in other network architectures, Layer 6 of the network has 241 neurons, while other layers remain unchanged.

## 5.5 Multibranched CNN

The model architectures we have described so far are top-down architectures, where the input to each layer comes from the previous layer's outputs. Such architectures require their convolutional layers to have a customized kernel and stride size, which can be restrictive



(1) Sigma-clip (2) Horizontal flip (3) Rotation

**Figure 4.** An illustration of image pre-processing and data augmentation using an example FIRST survey image (object id: Dabhade201), which is a radio source of class GRG with LAS of 108 arcsec.

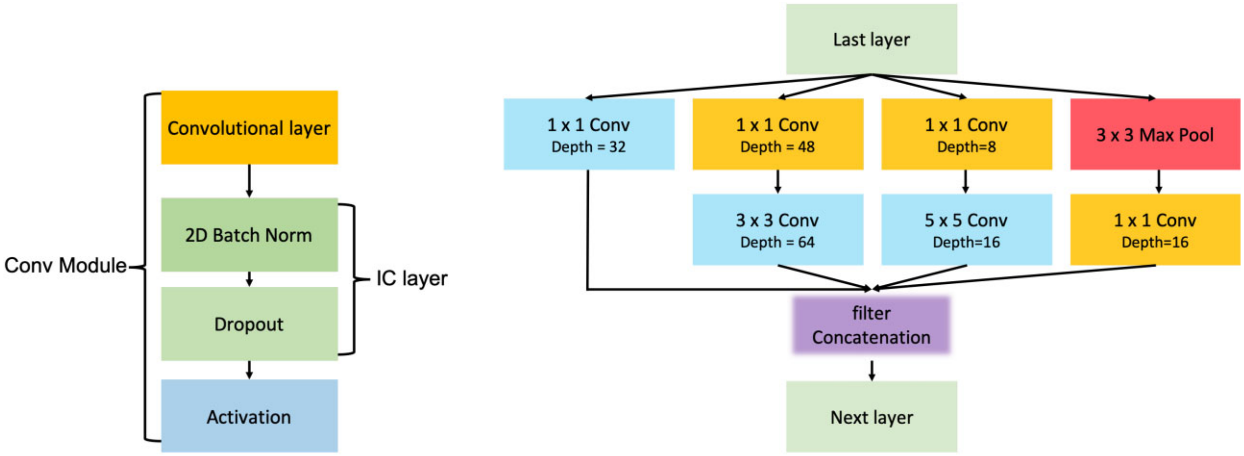


**Figure 5.** Left: The Conv module we used in this work, which is inspired by Li et al. (2019). The *IC layer* refers to the Independent Component layer (Chen et al. 2019); Right: The Inception module with dimension reduction Szegedy et al. (2014).

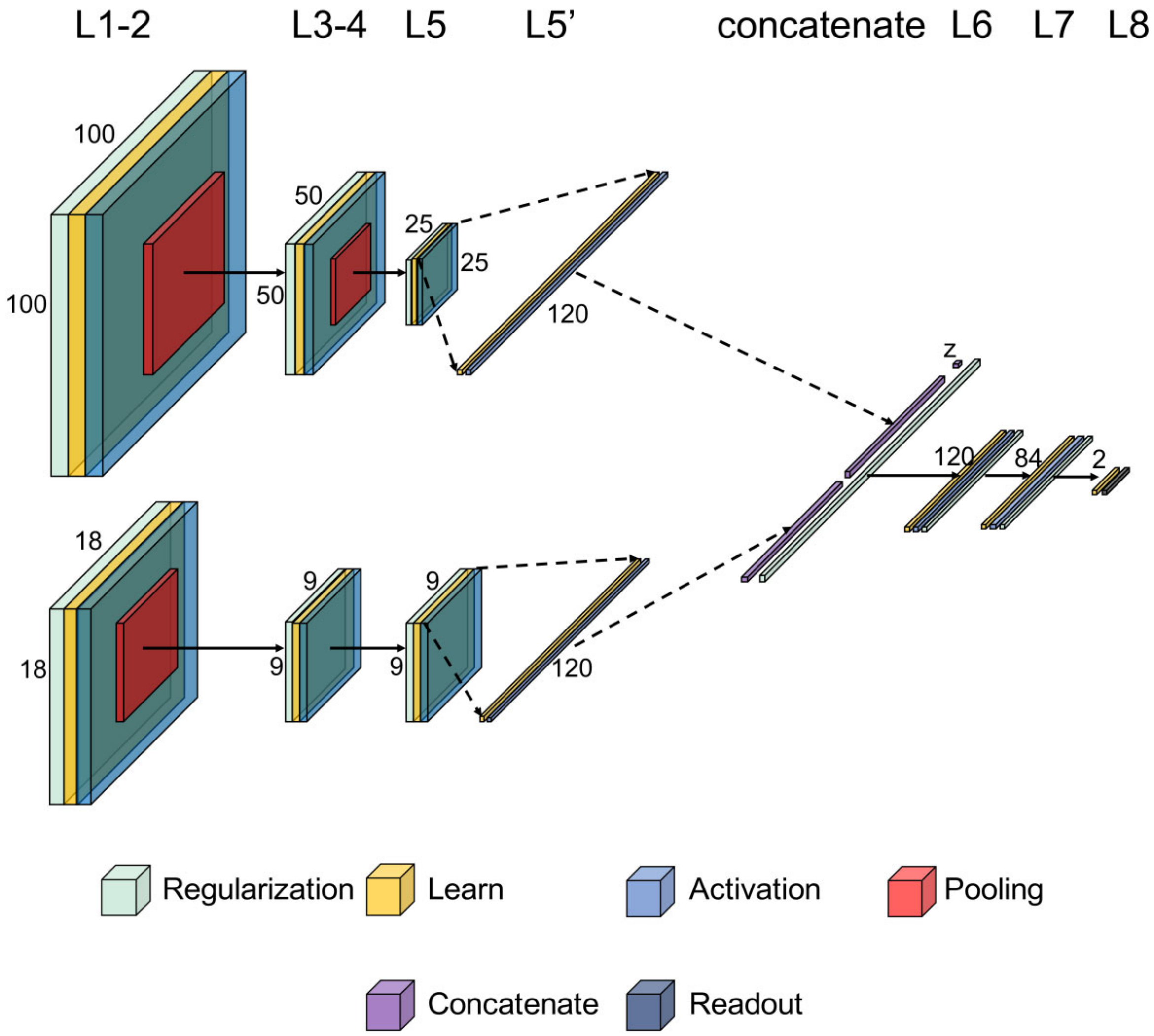


**Figure 6.** The network illustration of the architecture **F** in our work. In this diagram, `Regularization` refers to the use of IC/BN layer for the convolutional layers (L1, L3, and L5), and dropout layer for fully connected layers. `Learn` layers include convolutional layers (L1, L3, and L5) and fully connected layers (L6-8). `Activation` layers are all ReLU, while `Pooling` layers in the diagram are max-pooling layers. `Concatenate` operation implies that outputs from the last layer and the extra imported parameter (host galaxy redshift) would be concatenated as an 1D vector input for the next fully-connected layer (L6). Finally, the `Readout` layer is where softmax fucntion is operated, which provides the model class probability prediction. For full parametric details of this architecture, see Tables 3 and 4.

when one wants to simultaneously learn general features with different kernel sizes. It is constraints such as these that motivated the invention of CNNs with branched modules and later multibranch CNNs (e.g. Li et al. 2017; Georgakilas et al. 2020).

The very early and well-known branched CNNs include the GoogLeNet and later Inception networks (e.g. Lin et al. 2013; Szegedy et al. 2014). These networks implemented the Inception module structure described in Section 5.3.

Although the Inception module can be beneficial in terms of model performance, such an architecture can lead to large scale outputs, requiring heavy computation power. In this work, in order to minimize training costs, we adopt the modified Inception Module and use it to replace the final convolutional layers in Architecture F, denoted L5 in Fig. 6, and thus enable those layers to learn features with diverse kernel sizes. The filter dimensions for each layer of the Inception module in this work are chosen to be half of the equivalent value for the ‘inception (3a)’ model of GoogLeNet (e.g. Szegedy et al. 2014), making the output parameter number comparable to that of Layer 6 in Architecture F. We refer to this modified architecture Architecture G.

### 5.6 Model training

In order to simplify model performance comparisons, all model training performed in this work uses the Stochastic Gradient Descent optimizer (SGD; Robbins & Monro 1951) for optimization. Training sample data is imported to each model using a batch size of 20, and training data are shuffled in every training epoch. For model hyperparameter selection, we perform hyperparameter grid searches for model architectures A, B, and E using the Independent Component layer regularization strategy (see Section 5.2), along with different initial learning rates (1e−3, 1e−4, and 1e−5), dropout rates (0.4, 0.5, and 0.6) and training epoch numbers (180, 360, 720, and 1080) separately for both the GRGNOM-A and GRGNOM-B data sets.

In order to both prevent the resulting models from overfitting, and to achieve optimal model performances, we compared the learning curves, model AUC values and GRG class recall of the models with different hyperparameters. By requiring the GRG class recall of a model to be larger than 0.5 and that the model training and validation losses drop together till the end of model training regardless of architecture (A, B, and E), we found that models trained and tested with the GRGNOM-A data set matched the requirements only when their initial learning rate was 1e−3, the dropout rate was 0.4 and the number of training epoch was equal to 1080.

We then applied the same requirements to those models trained and tested with the GRGNOM-B data set, and found that the models matched the criteria when their dropout rate was 0.4, the initial learning rate was 1e−3 and the number of training epochs was no larger than 360. The validation loss comparison between the models trained for 180 and 360 epochs, shows that models generally perform better when they are trained for 360 epochs, with an average validation loss lower compared to those trained for 180 epochs. We therefore decided to train the models of all architectures following the aforementioned result, see Table 5. In the rest of this section, we describe the components of these networks in more detail.

## 6 RESULTS AND DISCUSSION

### 6.1 Model evaluation metrics

In the context of deep learning classification algorithms, popular model evaluation metrics include Accuracy, Recall, Precision, $F_1$ score and AUC score, see e.g. definitions in Appendix A of Bowles et al. (2021). These metrics have been widely used previously in the literature to evaluate CNN based radio galaxy classifiers (e.g. Aniyan & Thorat 2017; Ma et al. 2019; Tang et al. 2019; Bowles et al. 2021).

**Table 5.** A summary of hyperparameters we used in model training and evaluation. These hyperparameters are applied to all architectures we used in this work.

| Hyper-parameters | GRGNOM-A | GRGNOM-B |
|---|---|---|
| **Initialization** | From scratch | From scratch |
| **Batch size** | 20 | 20 |
| **Epoch** | 1080 | 360 |
| **Learning rate** | 1e−3 | 1e−3 |
| **Dropout rate** | 0.4 | 0.4 |

Performance metrics for the data set as a whole (Accuracy, AUC) and class-specific performance metrics for the GRG class (Precision, Recall) for each of the models considered in this work evaluated against the GRGNOM-A data set are listed in Table 6 and against the GRGNOM-B data set in Table 7.

### 6.2 Model performance

#### 6.2.1 Models trained with GRGNOM-A

Given that the GRGNOM-A data set has a severe class imbalance, class predictions are expected to be biased in the early phases of model training. This can be seen in Fig. 7, where the models used in this work tend to predict almost all validation samples as class NOM in the first 200–400 training epochs, although the models do gradually overcome this issue as the training continues.

Looking at the model loss curves that used the testing set as validation set, it can be seen from Fig. 7 that models trained with architectures A and C have their validation loss saturated quickly, while their ability to classify GRG objects increases gradually as training continues. On the other hand, the NOM recall for these models drops from 100 to around 98 per cent and then becomes stable. In other words, the mild improvement in performance seen from architectures A and C in terms of validation accuracy is partly contributed by the improvement of GRG recall but at the expense of NOM recall.

We also note that the architectures that use only NVSS images as their input tend to exhibit more stable training and result in a higher rate of correctly classified GRG samples. This can be seen in both Fig. 7 and Table 6. Compared to Architectures B and D, which have only FIRST data as an image input, models trained with only NVSS data as image inputs (Architectures A and C) have higher GRG recall by 18–27 per cent on average. This is likely to be caused by the GRG sample selection of GRGNOM-A, as shown in Fig. 9, where GRG objects are cross-validated using NVSS images, and thus their radio components are more clearly visible compared to those of their FIRST image counterparts.

The inclusion of host galaxy redshift as an input feature boosts model performance regardless of architecture. For architectures with or without redshift (A and C, B and D, and E and F), it can be observed that the inclusion of redshift information causes a marginal improvement in model accuracy, AUC value and GRG class metrics, as seen in Table 6.

Although the selection of image inputs gives different performances when using single image input models, using both image inputs generally contribute to better classification results when they

**Table 6.** Summary of model performance metrics for all architectures trained and tested with the GRGNOM-A data set. Each model architecture listed are trained 10 times on the GRGNOM-A training sets using independent Xavier initializations, and perform 10 independent model evaluations on each trained model.

| Architecture (IC regularization) | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| Input data | NVSS | FIRST | NVSS and $z$ | FIRST and $z$ | NVSS and FIRST | NVSS, FIRST, and $z$ | NVSS, FIRST, and $z$ |
| Accuracy (per cent) | 96.7 ± 0.5 | 95.7 ± 1.4 | 97.1 ± 0.6 | 96.4 ± 1.3 | 97.2 ± 0.9 | 97.4 ± 0.7 | 96.9 ± 0.9 |
| AUC | 0.948 ± 0.018 | 0.944 ± 0.027 | 0.953 ± 0.016 | 0.949 ± 0.027 | 0.974 ± 0.012 | 0.971 ± 0.014 | 0.969 ± 0.014 |
| Train loss | 0.080 ± 0.008 | 0.071 ± 0.014 | 0.067 ± 0.008 | 0.056 ± 0.012 | 0.070 ± 0.007 | 0.070 ± 0.008 | 0.074 ± 0.010 |
| Test loss | 0.107 ± 0.017 | 0.156 ± 0.022 | 0.105 ± 0.011 | 0.143 ± 0.032 | 0.103 ± 0.016 | 0.103 ± 0.027 | 0.100 ± 0.012 |
| Generalization gap | 0.027 ± 0.018 | 0.085 ± 0.026 | 0.038 ± 0.013 | 0.088 ± 0.034 | 0.033 ± 0.017 | 0.033 ± 0.028 | 0.026 ± 0.016 |
| Precision (GRG) | 0.727 ± 0.045 | 0.738 ± 0.130 | 0.777 ± 0.056 | 0.793 ± 0.117 | 0.821 ± 0.071 | 0.848 ± 0.060 | 0.800 ± 0.080 |
| Recall (GRG) | 0.796 ± 0.049 | 0.521 ± 0.170 | 0.783 ± 0.067 | 0.603 ± 0.143 | 0.723 ± 0.108 | 0.732 ± 0.089 | 0.698 ± 0.103 |
| F1 score (GRG) | 0.759 ± 0.036 | 0.603 ± 0.155 | 0.778 ± 0.048 | 0.679 ± 0.129 | 0.765 ± 0.080 | 0.783 ± 0.065 | 0.741 ± 0.080 |
| Architecture (BN regularization) | **A** | **B** | **C** | **D** | **E** | **F** | **G** |
| Input data | NVSS | FIRST | NVSS and $z$ | FIRST and $z$ | NVSS and FIRST | NVSS, FIRST, and $z$ | NVSS, FIRST, and $z$ |
| Accuracy (per cent) | 96.9 ± 0.5 | 97.0 ± 0.8 | 97.2 ± 0.5 | 96.7 ± 0.9 | 97.7 ± 0.6 | 97.9 ± 0.6 | 97.7 ± 0.6 |
| AUC | 0.957 ± 0.015 | 0.955 ± 0.025 | 0.967 ± 0.011 | 0.953 ± 0.021 | 0.987 ± 0.008 | 0.985 ± 0.010 | 0.989 ± 0.005 |
| Train loss | 0.057 ± 0.004 | 0.026 ± 0.011 | 0.040 ± 0.005 | 0.027 ± 0.012 | 0.033 ± 0.005 | 0.031 ± 0.005 | 0.0253 ± 0.006 |
| Test loss | 0.117 ± 0.010 | 0.149 ± 0.042 | 0.107 ± 0.012 | 0.172 ± 0.033 | 0.080 ± 0.014 | 0.073 ± 0.016 | 0.088 ± 0.022 |
| Generalization gap | 0.061 ± 0.010 | 0.123 ± 0.044 | 0.067 ± 0.013 | 0.146 ± 0.035 | 0.047 ± 0.015 | 0.042 ± 0.016 | 0.063 ± 0.022 |
| Precision (GRG) | 0.748 ± 0.042 | 0.837 ± 0.070 | 0.775 ± 0.049 | 0.812 ± 0.082 | 0.843 ± 0.056 | 0.858 ± 0.054 | 0.839 ± 0.059 |
| Recall (GRG) | 0.789 ± 0.066 | 0.670 ± 0.113 | 0.801 ± 0.053 | 0.641 ± 0.103 | 0.791 ± 0.079 | 0.808 ± 0.071 | 0.798 ± 0.087 |
| F1 score (GRG) | 0.766 ± 0.042 | 0.740 ± 0.088 | 0.786 ± 0.036 | 0.712 ± 0.085 | 0.814 ± 0.055 | 0.830 ± 0.051 | 0.814 ± 0.055 |

**Table 7.** Summary of model performance metrics for all architectures trained and tested with the GRGNOM-B data set. Each model architecture listed are trained 10 times on the GRGNOM-B training sets using independent Xavier initializations, and perform 10 independent model evaluations on each trained model.

| Architecture (IC regularization) | A | B | C | D | E | F | **G** |
|---|---|---|---|---|---|---|---|
| Input data | NVSS | FIRST | NVSS and $z$ | FIRST and $z$ | NVSS and FIRST | NVSS, FIRST, and $z$ | NVSS, FIRST, and $z$ |
| Accuracy (per cent) | 84.6 ± 1.0 | 89.9 ± 1.2 | 85.1 ± 1.2 | 89.9 ± 1.3 | 88.9 ± 1.3 | 89.1 ± 1.4 | 89.3 ± 1.3 |
| AUC | 0.873 ± 0.019 | 0.928 ± 0.011 | 0.891 ± 0.017 | 0.925 ± 0.011 | 0.926 ± 0.010 | 0.928 ± 0.010 | 0.931 ± 0.011 |
| Train loss | 0.285 ± 0.008 | 0.212 ± 0.015 | 0.254 ± 0.006 | 0.205 ± 0.010 | 0.223 ± 0.008 | 0.218 ± 0.013 | 0.222 ± 0.012 |
| Test loss | 0.388 ± 0.028 | 0.287 ± 0.016 | 0.367 ± 0.028 | 0.279 ± 0.016 | 0.278 ± 0.015 | 0.292 ± 0.017 | 0.288 ± 0.021 |
| Generalization gap | 0.104 ± 0.029 | 0.075 ± 0.022 | 0.113 ± 0.029 | 0.0741 ± 0.019 | 0.055 ± 0.017 | 0.075 ± 0.022 | 0.066 ± 0.024 |
| Precision (GRG) | 0.766 ± 0.033 | 0.826 ± 0.037 | 0.778 ± 0.034 | 0.823 ± 0.035 | 0.792 ± 0.036 | 0.806 ± 0.038 | 0.809 ± 0.037 |
| Recall (GRG) | 0.570 ± 0.027 | 0.765 ± 0.035 | 0.586 ± 0.037 | 0.771 ± 0.033 | 0.766 ± 0.033 | 0.756 ± 0.033 | 0.762 ± 0.030 |
| F1 score (GRG) | 0.653 ± 0.023 | 0.794 ± 0.024 | 0.668 ± 0.030 | 0.796 ± 0.026 | 0.778 ± 0.026 | 0.780 ± 0.027 | 0.784 ± 0.025 |
| Architecture (BN regularization) | **A** | **B** | **C** | **D** | **E** | **F** | **G** |
| Input data | NVSS | FIRST | NVSS and $z$ | FIRST and $z$ | NVSS and FIRST | NVSS, FIRST, and $z$ | NVSS, FIRST, and $z$ |
| Accuracy (per cent) | 86.2 ± 1.2 | 90.2 ± 1.5 | 86.5 ± 1.0 | 90.5 ± 1.9 | 91.1 ± 1.0 | 90.7 ± 1.1 | 91.4 ± 1.2 |
| AUC | 0.9 ± 0.011 | 0.935 ± 0.010 | 0.915 ± 0.010 | 0.934 ± 0.011 | 0.941 ± 0.010 | 0.941 ± 0.009 | 0.942 ± 0.009 |
| Train loss | 0.229 ± 0.008 | 0.156 ± 0.008 | 0.197 ± 0.007 | 0.158 ± 0.012 | 0.139 ± 0.011 | 0.140 ± 0.013 | 0.126 ± 0.016 |
| Test loss | 0.346 ± 0.014 | 0.286 ± 0.023 | 0.330 ± 0.017 | 0.280 ± 0.026 | 0.265 ± 0.020 | 0.275 ± 0.020 | 0.283 ± 0.019 |
| Generalization gap | 0.118 ± 0.017 | 0.131 ± 0.024 | 0.133 ± 0.019 | 0.122 ± 0.028 | 0.126 ± 0.022 | 0.136 ± 0.024 | 0.157 ± 0.025 |
| Precision (GRG) | 0.812 ± 0.030 | 0.825 ± 0.061 | 0.818 ± 0.024 | 0.836 ± 0.065 | 0.862 ± 0.026 | 0.863 ± 0.043 | 0.874 ± 0.033 |
| Recall (GRG) | 0.597 ± 0.032 | 0.794 ± 0.054 | 0.604 ± 0.042 | 0.787 ± 0.034 | 0.774 ± 0.038 | 0.761 ± 0.042 | 0.777 ± 0.035 |
| F1 score (GRG) | 0.688 ± 0.029 | 0.806 ± 0.027 | 0.694 ± 0.028 | 0.809 ± 0.031 | 0.815 ± 0.023 | 0.807 ± 0.024 | 0.822 ± 0.025 |

are imported together. Both with and without host galaxy redshift information or the presence of Inception modules, we found that architectures E, F, and G outperform the single image input models across model AUC values and GRG class precision, along with similar or better model test accuracies as compared to those single image inputs. This is consistent with what has been found from other non-astronomical applications, that multibranched approaches can boost model performance compared to the classical single input approach (e.g. Li et al. 2017; Georgakilas et al. 2020). However, it is also noteworthy that the inclusion of multiple image inputs on average decreases GRG class recall by 7 per cent, implying that such multibranch approaches should be treated with caution when the objective of the classifier is to identify as many GRGs as possible.

When we introduce the Inception module with dimension reduction to the model, we do not find a significant improvement in model performance when testing against the GRGNOM-A test sample. However, we find that this architecture performs differently when trained and test on GRGNOM-B, and this is discussed in more detail in the following section.

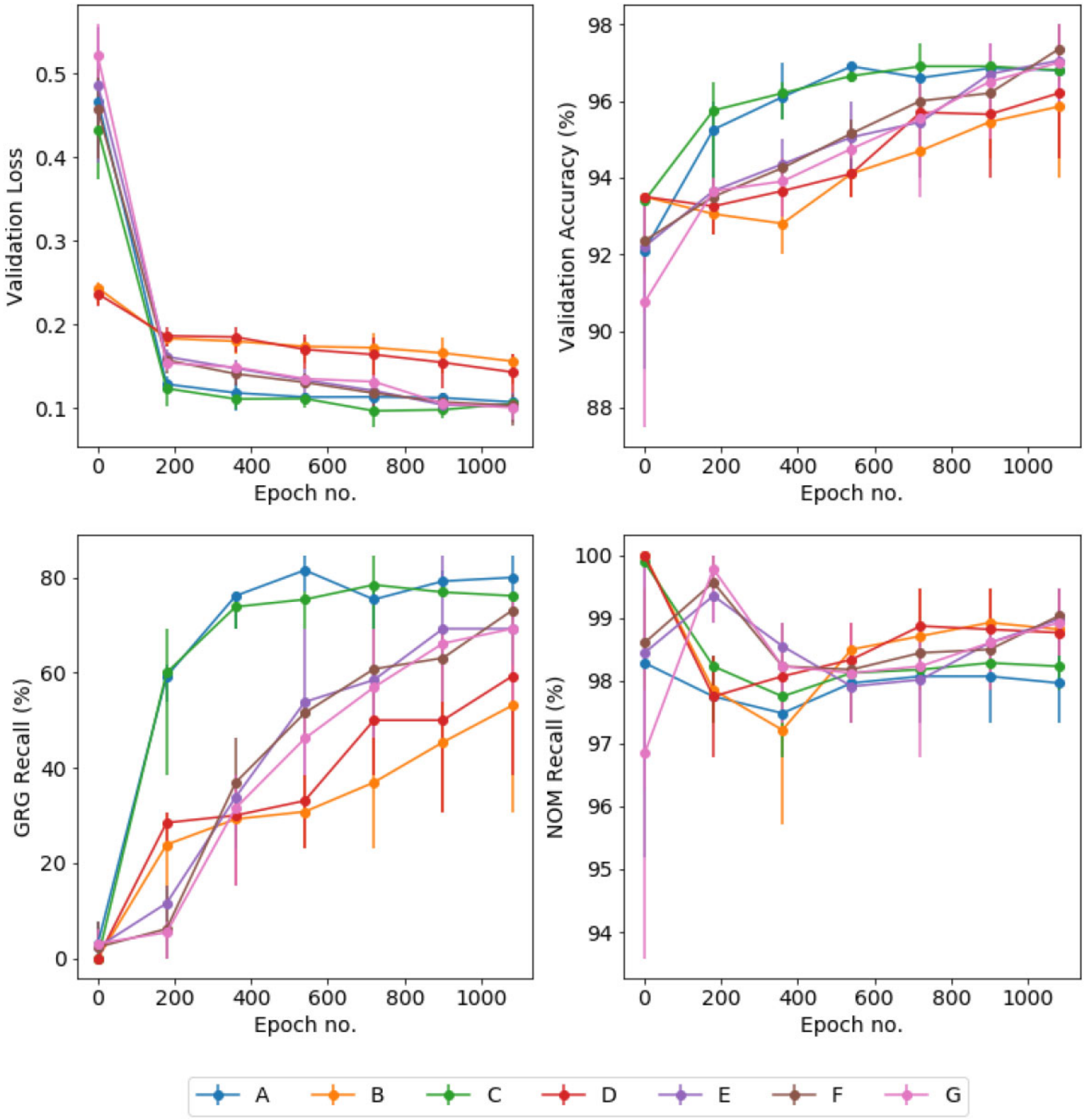


**Figure 7.** The averaged learning curve of the model architectures trained and validated with GRGNOM-A. Assuming normal distribution, the asymmetric errors of each data point on the diagram has covered 60 per cent of data distribution.

### 6.2.2 *Models trained with GRGNOM-B*

By comparing Tables 6 and 7, it can be seen that models trained using the GRGNOM-B data set are able to make more stable predictions. The inclusion of the Dabhade et al. (2020a) data in the training set lowers the class imbalance ratio from 14:1 in GRGNOM-A to around 3:1 in this data set. With more GRG examples in the training set, models are able to learn more quickly and make more stable predictions, see Fig. 8.

The biggest difference between Figs 7 and 8 is the reversal of model performance differences between single image input models trained with NVSS images and FIRST images. Compared with Architectures A and C, Architectures B and D have lower test losses and higher test accuracies after 180 epochs of training. The largest contribution to this difference can be attributed to data sample selection. The 101 Dabhade et al. (2020a) samples in the GRGNOM-B training set are identified from LoTSS survey maps with an angular resolution of 6 arcsec, and consequently the source morphology of these objects is found to be much closer to that of the FIRST survey with an angular resolution of 5.4 arcsec, rather than the lower resolution NVSS survey. Such similarity in angular resolution will contribute to model performance: once FIRST image inputs are imported, models trained with GRGNOM-B data samples receive $F1$ scores higher than 0.76 on average, see Table 7. Moreover, models trained using only FIRST images as inputs have GRG Recall/Precision values ≥19.7/6.8 per cent higher than those trained with equivalent NVSS images (A and B).

The inclusion of host galaxy redshift results in similar (B and D) or improved (A and C, E and F) model performance when trained and tested using the GRGNOM-B data set. The influence of the multibranch network approach, however, appears to behave differently. Compared to Architectures A and B, Architecture E was found to have similar or poorer model performance.

Interestingly, the inclusion of the Inception modules also seems to mildly improve model performance when testing with the GRGNOM-B data set. This is perhaps also due to the higher

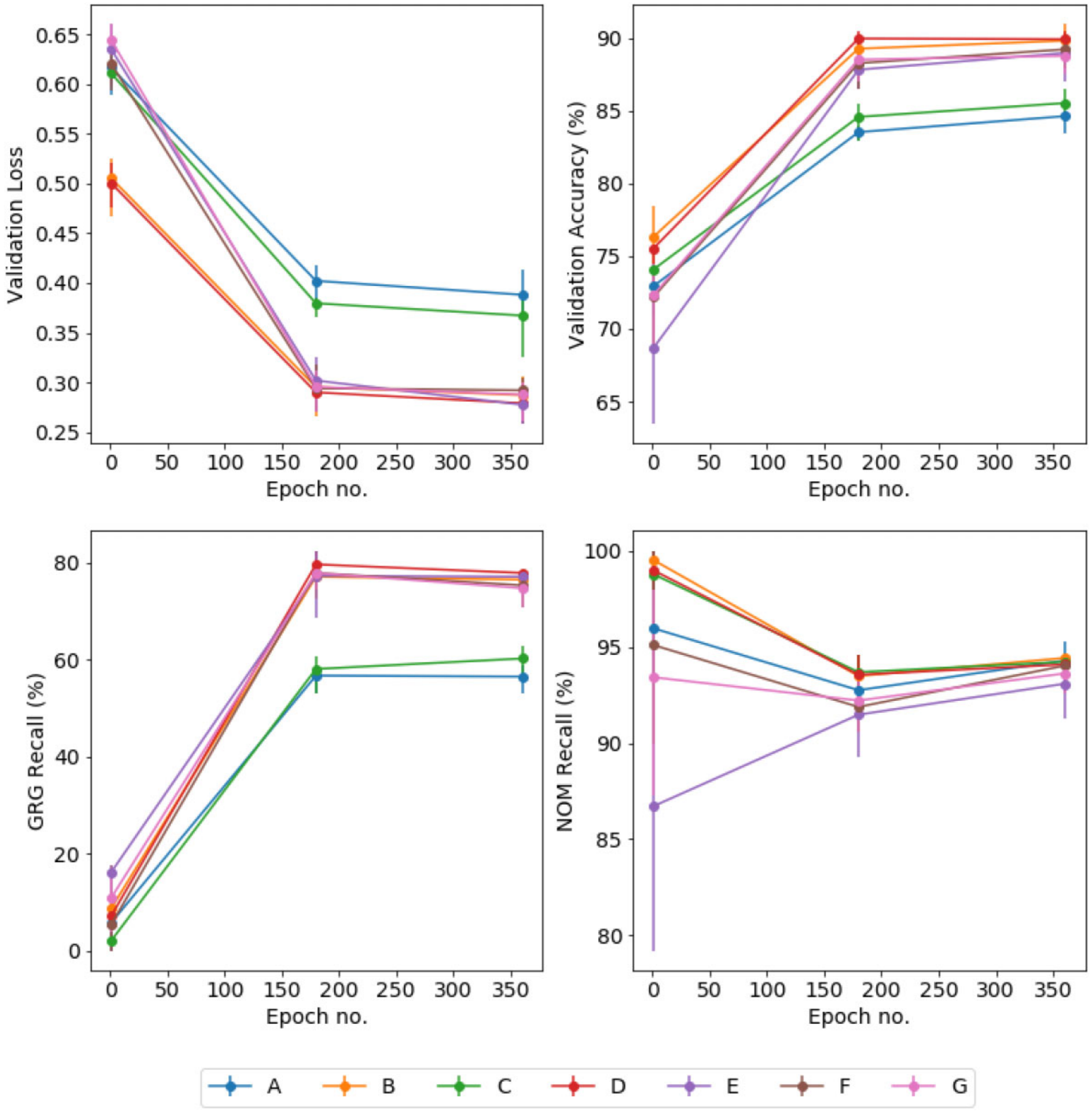


**Figure 8.** The averaged learning curve of the model architectures trained and validated with GRGNOM-B. Assuming normal distribution, the asymmetric errors of each data point on the diagram has covered 60 per cent of data distribution.

resolution sample selection for this data set. The extra network parameters are able to learn more complex source morphology features from these samples, and thus have slightly boosted model performance relative to other architectures.

Other than model performance evaluation, we also measure the model computational complexity of each architecture. This is achieved by measuring the number of floating point operations for a single instance of a forward pass through a given model (Becker et al. 2021). It can be seen from Fig. 10 that the Architecture G in our work has reduced model complexity by 0.01 Giga-FLOating Point operations (G-FLOPs) for models trained on both GRGNOM-A and GRGNOM-B compared to Architecture F.

### *6.2.3 Dataset Shift*

To this point we have evaluated the models in this work using test data taken from the same underlying data set as the training data, either GRGNOM-A or GRGNOM-B. However, it is also important to evaluate the generalization ability of a model when there exist small differences between underlying data distributions, a phenomenon referred to as data set shift.

A model with good generalization ability is able to perform well on new inputs unseen during the model training phase (Goodfellow et al. 2016). However, the use of this term should be treated with caution when the data set foundations do not follow i.i.d assumptions: i.e. that samples in the training and test sets are independent from each other, and share an identical underlying distribution (Goodfellow et al. 2016). One would then need to clarify that the model's generalization ability is evaluated under data set shift.

Almost all quantitative metrics used to evaluate a model's generalization ability are usually based on an assumption that the training set and the testing set share an identical joint generating distribution of inputs and outputs (Quionero-Candela et al. 2009). However, it is easy for this assumption to be violated. For instance, such an assumption will not hold when only the distribution of inputs is

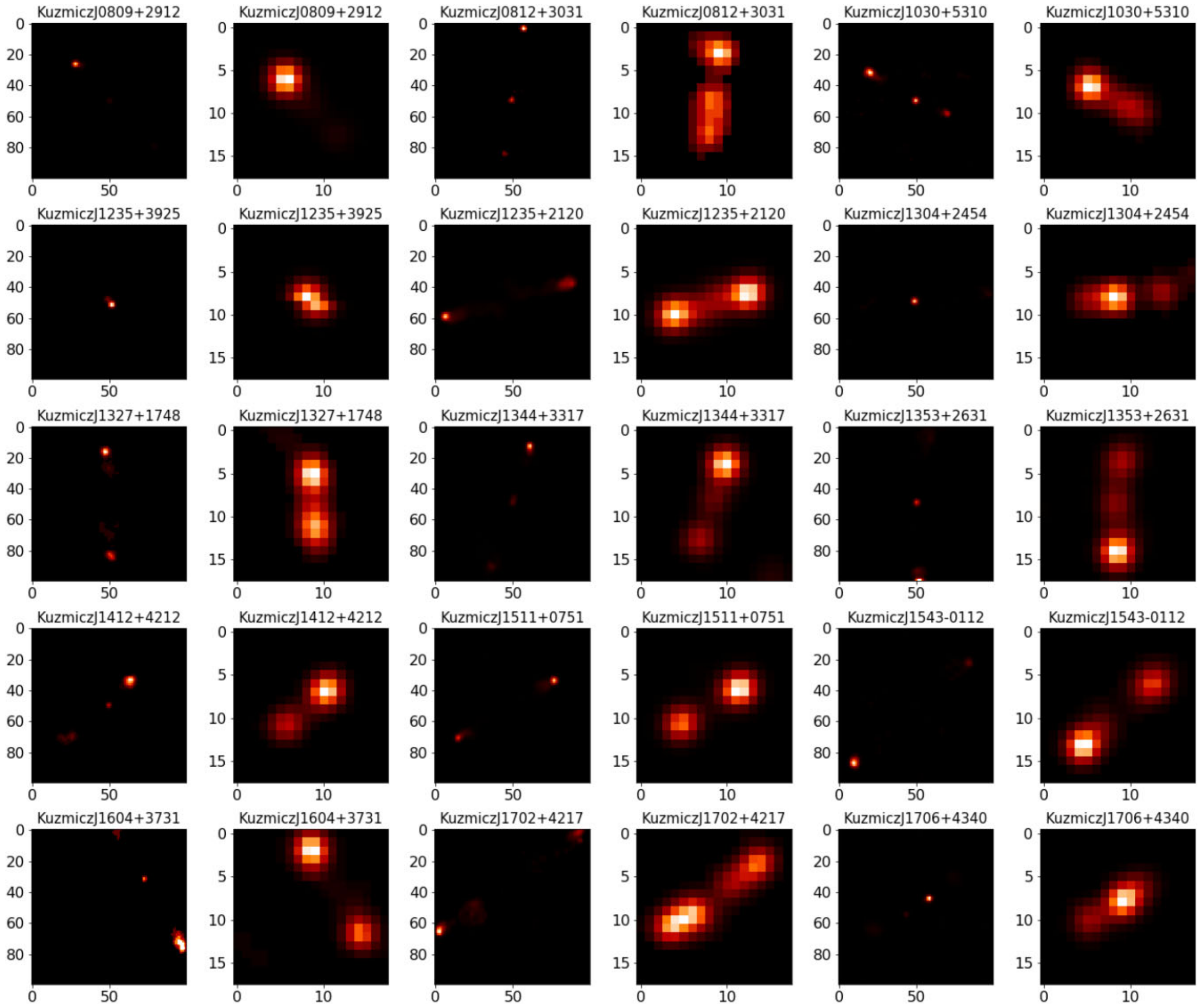


**Figure 9.** Example **GRG** images extracted from the Kuźmicz et al. (2018) catalogue. Under the same object ID, the left image refers to its pre-processed FIRST image, while the right image is the pre-processed NVSS image of the object.

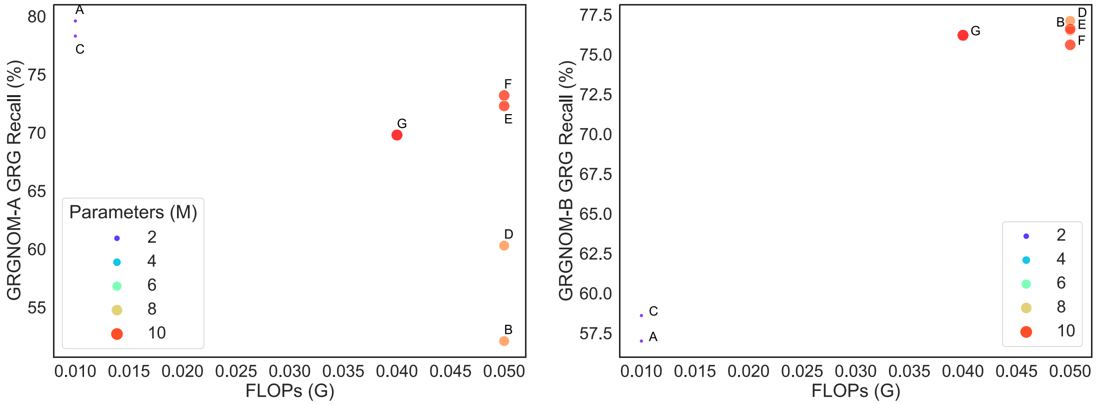


**Figure 10.** Left: Model computational complexity versus GRG Class Recall for an architecture regularized via IC, trained and tested on the GRGNOM-A data set. Model computational complexity is evaluated by computing the Giga floating point operations per second (G-FLOPs) shown on the $x$-axis. Model GRG Recall is shown on the $y$-axis as a percentage (per cent) given that the metric is of most importance if one aims to generate a comprehensive GRG catalogue from the GRGNOM-A test set using the models in our work. The size of the circles on the diagram indicates the number of trainable model parameters for each architecture (A–G) considered in this work in units of $10^6$. Right: The same diagram for each architecture trained and tested on the GRGNOM-B data set.

**Table 8.** Summary of model performance metrics for all architectures trained with the GRGNOM-A dataset and tested with the model generalization test set described in Section 6.2.3. Each model architecture listed are trained 10 times on the GRGNOM-A training sets using independent Xavier initializations, and perform 10 independent model evaluations on each trained model.

| Architecture (IC regularization) | A | B | C | D | E | F | **G** |
|---|---|---|---|---|---|---|---|
| Input data | NVSS | FIRST | NVSS and $z$ | FIRST and $z$ | NVSS and FIRST | NVSS, FIRST, and $z$ | NVSS, FIRST, and $z$ |
| Accuracy (per cent) | 80.9 ± 0.8 | 83.6 ± 1.3 | 81.3 ± 0.9 | 83.8 ± 1.3 | 81.6 ± 1.1 | 81.7 ± 1.0 | 81.4 ± 0.9 |
| AUC | 0.787 ± 0.021 | 0.828 ± 0.018 | 0.805 ± 0.016 | 0.833 ± 0.020 | 0.804 ± 0.023 | 0.805 ± 0.023 | 0.808 ± 0.021 |
| Precision (GRG) | 0.848 ± 0.047 | 0.883 ± 0.047 | 0.882 ± 0.044 | 0.887 ± 0.044 | 0.878 ± 0.050 | 0.884 ± 0.051 | 0.866 ± 0.046 |
| Recall (GRG) | 0.305 ± 0.026 | 0.415 ± 0.047 | 0.308 ± 0.031 | 0.419 ± 0.043 | 0.324 ± 0.038 | 0.327 ± 0.031 | 0.320 ± 0.033 |
| $F1$ score (GRG) | 0.448 ± 0.031 | 0.562 ± 0.045 | 0.456 ± 0.037 | 0.568 ± 0.044 | 0.472 ± 0.042 | 0.477 ± 0.036 | 0.466 ± 0.036 |
| Architecture (BN regularization) | **A** | **B** | **C** | **D** | **E** | **F** | G |
| Input data | NVSS | FIRST | NVSS and $z$ | FIRST and $z$ | NVSS and FIRST | NVSS, FIRST, and $z$ | NVSS, FIRST, and $z$ |
| Accuracy (per cent) | 81.3 ± 0.6 | 84.1 ± 1.1 | 81.4 ± 0.6 | 84.1 ± 1.2 | 82.7 ± 0.8 | 82.5 ± 0.7 | 82.6 ± 0.8 |
| AUC | 0.767 ± 0.027 | 0.841 ± 0.028 | 0.774 ± 0.019 | 0.834 ± 0.037 | 0.808 ± 0.021 | 0.812 ± 0.023 | 0.823 ± 0.016 |
| Precision (GRG) | 0.858 ± 0.032 | 0.869 ± 0.048 | 0.866 ± 0.027 | 0.873 ± 0.044 | 0.884 ± 0.028 | 0.885 ± 0.028 | 0.882 ± 0.026 |
| Recall (GRG) | 0.320 ± 0.022 | 0.446 ± 0.045 | 0.318 ± 0.023 | 0.441 ± 0.047 | 0.370 ± 0.027 | 0.360 ± 0.029 | 0.365 ± 0.034 |
| $F1$ score (GRG) | 0.465 ± 0.024 | 0.587 ± 0.039 | 0.465 ± 0.025 | 0.584 ± 0.042 | 0.521 ± 0.029 | 0.511 ± 0.030 | 0.515 ± 0.035 |

changed between the training and test set, a situation known as simple covariate shift, or alternatively when only the data class distribution changes (prior probability shift), or the training data samples do not accurately represent the full data distribution of the test samples (sample selection bias). A difference in the data source between the training and test set could also violate this assumption (source component shift). These scenarios can all be summarized as examples of data set shift (Quionero-Candela et al. 2009), which is almost inevitable when applying a model to make predictions on unseen data in a real world scenario.

In order to evaluate model generalization ability when data set shift exists, we now consider models trained using GRGNOM-A and tested using data from the 51 GRG samples in the GRGNOM-B test set. We also test these models with another 149 RGZ DR1 samples of class NOM that are not found in either training set. We refer to the resulting test set as GRGNOM-Gen. When comparing the GRGNOM-Gen test set with the GRGNOM-A train set, it is clear that both sample selection bias and source component shift are likely to be present, and so we expect to see these examples of data set shift manifest in differing model performance. We do not consider the opposite approach, as the GRG samples in the GRGNOM-A test set have been included in the GRGNOM-B training set. The evaluation metrics from these tests can be seen in Table 8.

From Table 8 it can be seen that the previously observed advantage of including multidomain data does not hold when testing on GRGNOM-Gen, resulting in a decrease of 2.2 per cent in model test accuracy, 0.029 in model AUC value, and 9.5 per cent decrease in GRG recall (e.g. D and E). More generally, it can be seen that when making predictions on these test samples, models trained with GRGNOM-A perform comparatively less well in terms of general model metrics than those directly trained on the GRGNOM-B data set. A similar situation also occurs when looking at GRG recall and $F_1$ score: even the best-performing Architecture D can only provide a GRG recall of 0.419 ± 0.043. On the other hand, the same architecture has a GRG precision of 88.7 per cent on average, implying that the model is able to identify NOM class objects well. In other words, although these models achieve higher GRG classification precision when compared to those trained using the GRGNOM-B data set, they are unable to reach a comparably high classification completeness. In order to find the majority of the GRGs in the Dabhade et al. (2020a) sample, it is essential to have some of the Dabhade et al. (2020a) samples included in the model training set.

### 6.2.4 *Angular size distance versus host galaxy redshift*

A consideration when introducing host galaxy redshift as an input feature is that the relationship between host galaxy redshift and angular size distance, $D_A$, is not linear. As an experiment, we used the equivalent $D_A$ in Gpc to replace host galaxy redshift when training Architecture F using the GRGNOM-A data set. The resulting models have an average AUC of 0.970 ± 0.016, slightly lower than, but not significantly different from, that found when using $z$ directly. When looking at GRG classification performance, the $D_A$ alternative returns a GRG Precision of 0.814 ± 0.069 and a GRG Recall of 0.707 ± 0.092. Comparing these metrics with those in Table 6, which use $z$ directly, it can be seen that the results are slightly poorer. This suggests that the network architecture already has sufficient capacity in its trainable parameters to learn the $z$–$D_A$ relationship, or an approximation of it.

## 6.3 BN versus IC

In order to compare the utility of the IC and BN layers, we performed a comprehensive model performance evaluation on all architectures that used either IC or BN for regularization, see Table 6–8). In general, both approaches are able to prevent our models from overfitting. The benefit of using BN for regularization is that it boosted model test accuracy, AUC, GRG recall in most cases by 1–3 per cent compared with those models that used IC. On the other hand, models using IC share smaller generalization gaps, i.e. the gap between the training and test loss values. The decrease in training loss for models regularized by BN is the major factor causing this difference, while these models typically have comparable test loss to those that used IC as a regularization method.

When we dive into specific architectures or data sets, we find several model performance differences between the two approaches. When we looked at those models which adopted BN regularization, trained and tested on GRGNOM-B, we found that they experienced a performance advantage (both for test accuracy and AUC value) from having multiple image inputs. It is true though that the model GRG recall behaves more poorly than for those trained with single image inputs (B/D and E), regardless of regularization method.

Moreover, in our previous discussion of different architectures regularized by IC, we noted that including redshift information provided similar or improved model performance for those trained on GRGNOM-A and tested on the GRGNOM-Gen data set, depending on architecture. However, we did not observe such a comprehensive improvement when we tested the models using BN for regularization on GRGNOM-Gen: models trained with redshift gained a 0.1–0.8 per cent improvement in GRG Precision, at the expense of a 0.2–1.0 per cent decrease in GRG Recall. Finally, when comparing Architectures F and G, we note that the inclusion of inception modules leads to slightly better performance (0.5 per cent improvement in GRG recall) when testing on the GRGNOM-Gen data set. Consequently, we conclude that IC is a non-harmful regularization method, but does not show significant benefit in this specific application. Users would need to compare its regularization ability with other methods when deciding model regularization strategies.

### 6.4 Common features shared by the misidentified samples

The model evaluation we have presented so far is based on a simple assumption: that the GRGNOM-A/B data sets are fully understood, reliable, and confidently labelled. Yet it is unclear whether frequently misclassified objects in our models share common features. In this section and in Section 6.5, we consider the models using IC as a regularization method and trained with the GRGNOM-B data set as examples. By applying each of these models to all samples in the GRGNOM-B test set, we are able to identify the GRGNOM-B test samples that have a misclassification rate of ≥50 per cent for all of the 7 architectures used in this work. These samples are summarized in Table 9. In general, our models mistakenly identify GRGs as class NOM more frequently than the reverse. This is unsurprising since both the GRGNOM-A and GRGNOM-B training samples contain a much higher number of NOM-type objects.

A potential data 'trap' in the GRGNOM-B data sets comes from the Dabhade et al. (2020a) samples. These objects were identified using the 151 MHz LoTSS survey (Shimwell et al. 2019). Considering that $S \propto \nu^{\alpha}$, where $\alpha = -0.7$ for optically thin synchrotron emission, sources will be brighter at 151 MHz compared to their NVSS or FIRST counterparts at 1.4 GHz. In addition, the median rms of LoTSS is 71 μJy beam$^{-1}$, lower than half that of the FIRST survey and around 16 per cent of the NVSS sensitivity of 0.45 mJy beam$^{-1}$. This means that LoTSS will be more sensitive to faint radio emission, and that some radio structures present in LoTSS images might be missed or resolved out in the equivalent NVSS and/or FIRST images. Besides the 'trap', the aforementioned image specifications, pre-processing choices, selection of input domains (NVSS images, FIRST images, host galaxy redshifts), and selection of architectures could also result in differences in model performance.

In order to investigate these frequent misclassifications we use the data traceability built into our data set, see Section 4. Similarly traceable data sets have been built and implemented for a number of recent deep learning studies (e.g. Wu et al. 2019; Walmsley et al. 2020) and their data traceability used to explain why some samples are mistakenly identified in a frequent manner (e.g. Wu et al. 2019). In this case, the sources listed in Table 9 were traced using their unique object IDs, see Section 4.3. We then analysed these cases of object misclassification by looking at their NVSS and FIRST pre-processed images as well as the host galaxy redshift in each case.

**Table 9.** A summary of frequent mistakenly identified GRGNOM-B testing samples in this work. A sample would be included in this table if it has over 50 per cent rate to be mistakenly identified in at least one architecture adopted in this work.

| Object ID | ≥50 per cent Mistakenly Identified Architectures |
|---|---|
| Dabhade230 | All |
| Dabhade217 | All |
| Dabhade198 | All |
| Dabhade173 | All |
| Dabhade237 | All |
| Dabhade186 | All |
| Dabhade216 | All |
| Dabhade193 | All |
| RGZJ080417.6+320250 | A, B, C, E, F, G |
| RGZJ075855.6+360246 | A, B, C, D, E, F |
| Dabhade204 | A, C, E, F, G |
| RGZJ075030.7+525022 | A, C, E, F, G |
| RGZJ080448.0+081254 | A, C, E, F, G |
| RGZJ075539.6+160158 | A, C, E, F, G |
| RGZJ075157.7+212049 | B, D, E, F, G |
| RGZJ080427.8+132930 | A, D, E, F |
| RGZJ075812.7+190043 | B, D, E, G |
| Dabhade185 | A, B, C, D |
| Dabhade221 | B, D, F |
| Dabhade201 | A, B, D |
| Dabhade197 | A, C, G |
| Dabhade214 | B, F, G |
| Dabhade199 | A, C |
| Dabhade227 | A, C |
| Dabhade206 | A, C |
| Dabhade226 | A, C |
| RGZJ074627.1+174337 | A, C |
| RGZJ074720.7+335008 | A, C |
| Dabhade229 | A, C |
| Dabhade231 | A, C |
| Dabhade220 | A, C |
| Dabhade209 | A, C |
| Dabhade163 | A, C |
| RGZJ080404.5+153334 | B, D |
| RGZJ075306.1+121504 | B, D |
| Dabhade210 | E, G |
| RGZJ080402.5+452258 | E |
| RGZJ075620.0+301630 | E |

#### 6.4.1 Low surface brightness

As described in Section 4, the sigma-clipping performed during image pre-processing will replace all pixels with values lower than the local $3\sigma_{\rm rms}$ level with zeros, resulting in faint objects that have only a mild luminosity difference with the noise background becoming even fainter in the normalized image when a secondary source is present in the field of view. For example, the source Dabhade 237 was faint but visible at 151 MHz in the original LOFAR data (fig. A.8; Dabhade et al. 2020a). However, the both the radio core and the nearby radio lobes seem to be too faint to be clearly identified in the pre-processed FIRST and NVSS maps used in this work. The models are therefore unable to find a GRG like object in the image, but instead identify a secondary source at the edge of the field to be class NOM. This suggests that using data samples selected at different frequencies for model training/testing should be treated with caution.

### *6.4.2 Input domain selection*

In general, multidomain approaches are expected to be able to help a network to cross-validate its prediction results. In this work, 39 per cent of the frequently misidentified objects listed in Table 9 have their identification corrected when both image domains are imported. The sources RGZ J075306.1+121504 and Dabhade 199 are two representative examples: RGZ J075306.1+121504 is frequently misidentified without the inclusion of its NVSS image. In spite of the slightly extended structure along with scattered faint background emissions in its FIRST image, its NVSS image retains clear compact structure, helping the networks to correctly identify the object class. On the other hand, Dabhade 199, whilst still having segments of its lobes resolved out in the FIRST image, has a clearly extended central source. In this case, the lower resolution NVSS image leads to confusion between two objects: Dabhade 199 as a GRG and another compact source, which makes it difficult for the network to identify Dabhade 199 as a GRG with confidence; however, with the help of the additional information from the FIRST image, the networks are able to classify this source correctly in most cases.

Conversely, the use of multiple image domains may also have a negative impact on model performance. Two examples of this are RGZ J075030.7+525022 and RGZ J075157.7+212049. While RGZ J075030.7+525022 looks compact in its FIRST image, the object shows a well-extended morphology in its NVSS map. Such differences in sample morphology lead to architectures that include the NVSS image input being unable to make a correct classification. On the other hand, the source RGZ J075157.7+212049 shows an extended morphology along with significant scattered emission in its FIRST map, while its NVSS map is dominated by a single compact object. Architectures including FIRST image data as an input make incorrect identifications in this case. Considering these examples, it appears that sources with different numbers of radio components in different surveys are more likely to be misclassified.

### *6.4.3 Architecture bias*

As well as the selection of input data, differences in network architecture will also contribute to model performance. For examples, RGZ J075855.6+360246 ($z = 0.336$; see Fig. 11) was correctly identified only when the Inception module was introduced. Both of the NVSS and FIRST images for this source include a nearby well-extended object, while the target itself only appears in its FIRST image, where it appears compact and faint. Considering the uniform kernel size of 5 for the convolutional layers in the models of Architectures A–F, these models may have found it difficult to capture the morphological features of the small and faint object. However, by introducing an Inception module to a model with Architecture G, the network was able to capture features down to a 1-pixel scale. This perhaps explain why the object is correctly identified only when we use Architecture G. Given that 78.9 per cent of the misclassified objects in Table 9 have at least one architecture producing a different result from the others, it appears that those sources where architectures disagree with each other are more likely to be misclassified.

## 6.5 Cases requiring further explanation

Apart from the aforementioned examples, there are a number of frequently misclassified objects in Table 9 that we found difficult to explain by simply looking at object sample data. For instance, Dabhade 185 can be seen as a unique example of multidomain cross validation. Its radio lobes in the FIRST image have been resolved out, but are unresolved in the NVSS image map. This object was classified as GRG when both NVSS or FIRST are used, which we find difficult to interpret.

Another issue we encounter when trying to explain model behaviour occurs when there is more than one well-extended object in an image map. From Table 9 it can be seen that the source Dabhade 221 was consistently misclassified by three of the models in this work. In this case, via visual inspection we found that apart from the GRG, there is another well-extended radio source visible in the FIRST image, see Fig. 11. The translational equivariance of the convolutional layers does not require a source to be located at the centre of the image in order to be correctly classified and this is an example of such a circumstance. Which source emission finally contributes to the prediction remains uncertain.

Analysis of such complex cases could perhaps benefit from the use of eXplainable Artificial Intelligence (XAI) tools (e.g. SHAP; Lundberg & Lee 2017). Rather than visualizing feature maps for each specific layer, these tools allow users to directly visualize which features the network as a whole has recognized from each sample image, and in some cases can evaluate their contributions to target class identification. Another possibility might be to use attention-gating, which produces attention maps that similarly facilitate the interpretation of a classification choice as made by the model (see e.g. Bowles et al. 2021).

## 6.6 Comparison to other automated search methods

As described in Section 2.2, a previous approach to automated GRG identification was made by Proctor (2016) using a decision tree based machine learning approach. Using source pair separations from the NVSS catalogue, Proctor (2016) produced a list of GRG candidates with $LAS \geq 240$ arcsec. From Fig. 3 it can be seen that all of the GRG class objects used in this work are smaller than the limit of Proctor (2016); however, it can also be seen that there is still a clear separation in *LAS* between NOM class galaxies and GRGs and we suggest that it is for this reason that the inclusion of redshift information did not strongly affect the results of the models presented in this work. This *LAS* separation is likely to be a consequence of the historic sample selection governing current catalogues of known GRGs. The development of labelled training sets with larger numbers of intermediate size radio galaxies will enable future studies to investigate this cross-over region of parameter space in more detail.

# 7 CONCLUSIONS

Previous GRG searches have largely depended on visual inspection, and, while successful given the size of historic observational data bases, this human-powered methodology is unlikely to scale well to the new generation of radio sky survey data, which will require the investigation of millions of extended radio sources.

In this work, we have explored the possibility of automated GRG identification through deep-learning by using seven different CNN-based model architectures, including a multibranched CNN algorithm that incorporates information from multiple surveys with different resolution. The best-performing models in this study achieve 97.9 and 91.4 per cent test accuracy using the GRGNOM-A and GRGNOM-B test samples described in this work. This result follows from an extensive investigation of model performance

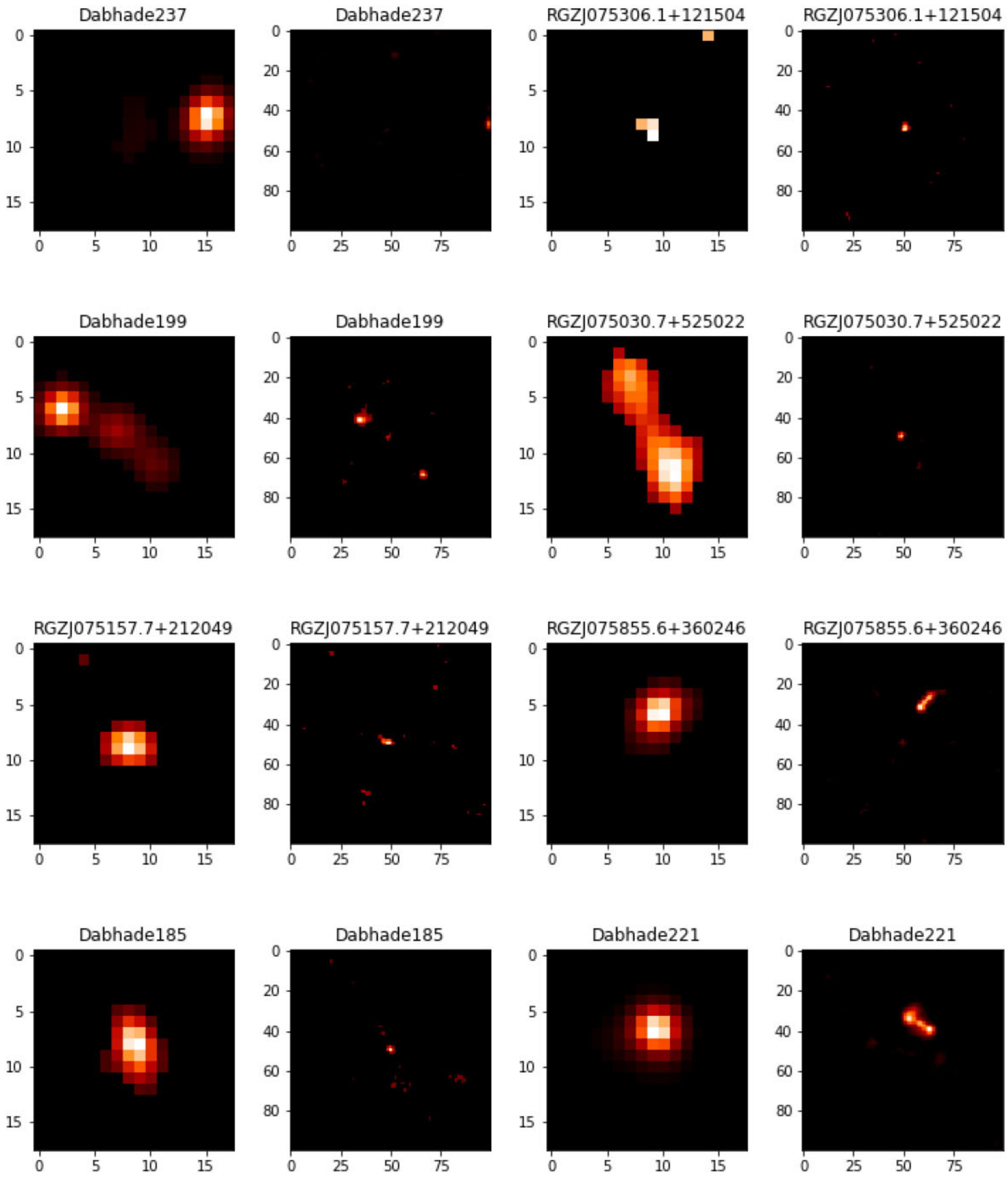


**Figure 11.** A summary of misidentified GRGNOM-B testing samples, which represent the typical types of misclassification described in Section 6.4. Under the same object ID, the left image refers to its pre-processed NVSS image, while the right image is the pre-processed FIRST image of the object.

under different architectural choices, data set shift and data set composition.

A key result from this work is the introduction of multibranched networks in order to boost model performance compared with the classical CNN architectures that use a single type of image input. By including host galaxy redshift information, model performance was also found to be improved. We find that importing both NVSS and FIRST images as dual inputs corrected 39 per cent of objects that were misclassfied by a single domain network.

Architecturally, IC as a regularization method was found to be non-harmful while need to be compared with other methods when deciding regularization strategies. Furthermore, it was shown that the use of the inception module could lower the model computational cost, affect model performance under certain circumstances, and was found to be able to correct misclassifications in the case when the target object is small and compact.

Finally, we investigated the cause of frequent misclassifications by inspecting individual samples, and found that other than the aforementioned factors, a sample might be frequently misclassified if (a) its sample image contains multiple sources, (b) the standard pre-processing procedures have eliminated part of its extended morphology, or (c) its radio component was partly resolved out comparing with the survey map images used to identify the GRG (e.g. LoTSS at 151 MHz).

## ACKNOWLEDGEMENTS

The authors are grateful for the assistance of over 12 000 volunteers in the Radio Galaxy Zoo project, whose contributions are acknowledged at http://rgzauthors.galaxyzoo.org. The authors are also very grateful for discussions from the machine learning group at Jodrell Bank Centre for Astrophysics, JBCA. This research was supported by JBCA, University of Manchester. The corresponding author is also grateful for the effort of project team members, participating school teachers and students of RGZ_CN: Scientific exploration course of Radio Galaxy Zoo, a teaching side project of the Radio Galaxy Zoo. AMS gratefully acknowledges support from an Alan Turing Institute AI Fellowship EP/V030302/1. This research made extensive use of Astropy,[2] a community-developed core PYTHON package for Astronomy (Astropy Collaboration 2013, 2018). HT acknowledges the support from the Shuimu Tsinghua Scholar Program of Tsinghua University.

## DATA AVAILABILITY

Exemplar model weights for all architectures presented in this work using IC as the regularization method are available from Zenodo (DOI: 10.5281/zenodo.5749316). The RGZ DR1 catalogue will be made publicly available through Wong et al. (2022, in preparation), at which time the GRGNOM-A/B data sets will also become available on Zenodo.

## REFERENCES

Albareti F. D. et al., 2017, ApJS, 233, 25
Alger M. J. et al., 2018, MNRAS, 478, 5547
Alhassan W., Taylor A. R., Vaccari M., 2018, MNRAS, 480, 2085
Amerini I., Uricchio T., Ballan L., Caldelli R., 2017, IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE, p. 1865
Amirkhanyan V. R., 2016, Astrophys. Bull., 71, 384
Amirkhanyan V. R., Afanasiev V. L., Moiseev A. V., 2015, Astrophys. Bull., 70, 45
Aniyan A. K., Thorat K., 2017, ApJS, 230, 20
Ansdell M. et al., 2018, ApJ, 869, L7
Aslani S., Dayan M., Storelli L., Filippi M., Murino V., Rocca M. A., Sona D., 2019, NeuroImage, 196, 1
Astropy Collaboration, 2013, A&A, 558, A33
Astropy Collaboration, 2018, AJ, 156, 123
Babul A., Sharma P., Reynolds C. S., 2013, ApJ, 768, 11
Bagchi J. et al., 2014, ApJ, 788, 174
Banfield J. K. et al., 2015, MNRAS, 453, 2326
Banfield J. K. et al., 2016, MNRAS, 460, 2376
Becker R. H., White R. L., Helfand D. J., 1995, ApJ, 450, 559
Becker B., Vaccari M., Prescott M., Grobler T., 2021, MNRAS, 503, 1828
Begelman M. C., Blandford R. D., Rees M. J., 1984, Rev. Mod. Phys., 56, 255
Bisong E., 2019, Google Colaboratory. Apress, Berkeley, CA, p. 59
Blandford R. D., Rees M. J., 1974, MNRAS, 169, 395
Bock D. C.-J., Large M. I., Sadler E. M., 1999, AJ, 117, 1578
Bowles M., Scaife A. M. M., Porter F., Tang H., Bastien D. J., 2021, MNRAS, 501, 4579
Bridle A. H., Davis M. M., Meloy D. A., Fomalont E. B., Strom R. G., Willis A. G., 1976, Nature, 262, 179
Brown S. et al., 2019, MNRAS, 483, 964
Brunel A., Pasquet J., Pasquet J., Rodriguez N., Comby F., Fouchez D., Chaumont M., 2019, preprint (arXiv:1901.00461)
Burbidge G. R., 1956, ApJ, 124, 416
Cao H., Liu H., Song E., Hung C.-C., Ma G., Xu X., Jin R., Lu J., 2020, Applied Soft. Computing, 86, 105934
Chan M. L., Heng I. S., Messenger C., 2020, Phys. Rev. D, 102, 043022
Chen G., Chen P., Shi Y., Hsieh C.-Y., Liao B., Zhang S., 2019, preprint (arXiv:1905.05928)
Cheng H.-T. et al., 2016, DLRS@RecSys, 7
Cielo S., Babul A., Antonuccio-Delogu V., Silk J., Volonteri M., 2018, A&A, 617, A58
Clarke A. O. et al., 2017, A&A, 601, A25
Condon J. J., Cotton W. D., Greisen E. W., Yin Q. F., Perley R. A., Taylor G. B., Broderick J. J., 1998, AJ, 115, 1693
Connor L., van Leeuwen J., 2018, AJ, 156, 256
Cotter G., Rawlings S., Saunders R., 1996, MNRAS, 281, 1081
Dabhade P., Gaikwad M., Bagchi J., Pandey-Pommier M., Sankhyayan S., Raychaudhury S., 2017, MNRAS, 469, 2886
Dabhade P. et al., 2020a, A&A, 635, A5
Dabhade P. et al., 2020b, A&A, 642, A153
de Bruyn A. G., 1989, A&A, 226, L13
Delhaize J. et al., 2021, MNRAS, 501, 3833
Edge D. O., Shakeshaft J. R., McAdam W. B., Baldwin J. E., Archer S., 1959, MNRAS, 68, 37
Ekers R. D. et al., 1989, MNRAS, 236, 737
Georgakilas G. K., Grioni A., Liakos K. G., Chalupova E., Plessas F. C., Alexiou P., 2020, Sci. Rep., 10, 9486
Gheller C., Vazza F., Bonafede A., 2018, MNRAS, 480, 3749
Goodfellow I. J., Bengio Y., Courville A., 2016, Deep Learning. MIT Press, Cambridge, MA
He K., Zhang X., Ren S., Sun J., 2016, IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, p. 770
Hota A. et al., 2011, MNRAS, 417, L36
Huynh M. T., Jackson C. A., Norris R. P., 2007, AJ, 133, 1331
Intema H. T., Jagannathan P., Mooley K. P., Frail D. A., 2017, A&A, 598, A78
Ioffe S., Szegedy C., 2015, JMLR, 37, 448
Ishwara-Chandra C. H., Saikia D. J., 1999, MNRAS, 309, 100
Johnston S. et al., 2008, Exp. Astron., 22, 151
Jones P. A., 1989, Proc. Astron. Soc. Aust., 8, 81
Kapińska A. D. et al., 2017, AJ, 154, 253
Kimura A., Takahashi I., Tanaka M., Yasuda N., Ueda N., Yoshida N., 2017, IEEE 37th International Conference on Distributed Computing Systems Workshops (ICDCSW). IEEE, p. 354
Kiranyaz S., Avci O., Abdeljaber O., Ince T., Gabbouj M., Inman D. J., 2021, MSSP, 151, 107398
Koch D. G. et al., 2010, ApJ, 713, L79
Kormendy J., Ho L. C., 2013, ARA&A, 51, 511
Kozieł-Wierzbowska D., Stasińska G., 2011, MNRAS, 415, 1013
Koziel-Wierzbowska D., Goyal A., Zywucka N., 2020, ApJS, 247, 53
Krause M. G. H. et al., 2019, MNRAS, 482, 240
Krizhevsky A., Sutskever I., Hinton G. E., 2012, in Pereira F., Burges C. J. C., Bottou L., Weinberger K. Q., eds, Advances in Neural Information Processing Systems 25. Curran Associates, Inc., p. 1097
Kronberg P. P., Wielebinski R., Graham D. A., 1986, A&A, 169, 63
Kuźmicz A., Jamrozy M., Bronarska K., Janda-Boczar K., Saikia D. J., 2018, ApJS, 238, 9
Lacy M., Rawlings S., Saunders R., Warner P. J., 1993, MNRAS, 264, 721
Lacy M. et al., 2020, PASP, 132, 035001
Laing R. A., Riley J. M., Longair M. S., 1983, MNRAS, 204, 151
Lara L., Márquez I., Cotton W. D., Feretti L., Giovannini G., Marcaide J. M., Venturi T., 2001, A&A, 378, 826
Law-Green J. D. B., Eales S. A., Leahy J. P., Rawlings S., Lacy M., 1995, MNRAS, 277, 995
Leahy J. P., 1993, in Röser H.-J., Meisenheimer K., eds, Jets in Extragalactic Radio Sources. Springer, Berlin, Heidelberg, p. 1
Lecun Y., Bottou L., Bengio Y., Haffner P., 1998, Proc. IEEE, 86, 2278
LeCun Y., Bottou L., Orr G., Müller K., 2012, in Neural Networks: Tricks of the Trade, Lecture Notes in Computer Science. Springer, Berlin, Heidelberg

[2] http://www.astropy.org

Letawe G., Courbin F., Magain P., Hilker M., Jablonka P., Jahnke K., Wisotzki L., 2004, A&A, 424, 455

Li B., Luo H., Zhang H., Tan S., Ji Z., 2017, preprint (arXiv:1710.05477)

Li X., Chen S., Hu X., Yang J., 2019, Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, p. 2682

Lin M., Chen Q., Yan S., 2013, preprint (arXiv:1312.4400)

Longair M. S., Ryle M., Scheuer P. A. G., 1973, MNRAS, 164, 243

Lukic V., Brüggen M., Banfield J. K., Wong O. I., Rudnick L., Norris R. P., Simmons B., 2018, MNRAS, 476, 246

Lundberg S., Lee S.-I., 2017, in Guyon I.  et al., eds, Advances in Neural Information Processing Systems (NIPS), Vol. 30. Curran Associates, Inc., p. 4765

Ma Z. et al., 2019, ApJS, 240, 34

Machalski J., Jamrozy M., Zola S., 2001, A&A, 371, 445

Machalski J., Kozieł-Wierzbowska D., Jamrozy M., Saikia D. J., 2008, ApJ, 679, 149

Magorrian J. et al., 1998, AJ, 115, 2285

Matsugu M., Mori K., Mitari Y., Kaneda Y., 2003, Neural Netw., 16, 555

Mauch T., Murphy T., Buttery H. J., Curran J., Hunstead R. W., Piestrzynski B., Robertson J. G., Sadler E. M., 2003, MNRAS, 342, 1117

McCarthy P. J., Kapahi V. K., van Breugel W., Persson S. E., Athreya R., Subrahmanya C. R., 1996, ApJS, 107, 19

Molina M., Bassani L., Malizia A., Bird A. J., Bazzano A., Ubertini P., Venturi T., 2014, A&A, 565, A2

Norris R. P. et al., 2011, Publ. Astron. Soc. Aust., 28, 215

Norris R. P. et al., 2021, Publ. Astron. Soc. Aust., 38, e046

Osborn H. P. et al., 2020, A&A, 633, A53

Peng B., Chen R. R., Strom R., 2015, in Proc. Sci., Giant radio galaxies as probes of the ambient WHIM in the era of the SKA. SISSA, Trieste, p. PoS#109

Planck Collaboratio XIII, 2016, A&A, 594, A13

Prescott M. et al., 2018, MNRAS, 480, 707

Proctor D. D., 2016, ApJS, 224, 18

Quionero-Candela J., Sugiyama M., Schwaighofer A., Lawrence N. D., 2009, Dataset Shift in Machine Learning. The MIT Press, Cambridge, MA

Rees M. J., 1984, ARA&A, 22, 471

Rengelink R. B., Tang Y., de Bruyn A. G., Miley G. K., Bremer M. N., Roettgering H. J. A., Bremer M. A. R., 1997, A&AS, 124, 259

Robbins H., Monro S., 1951, Ann. Math. Statist., 22, 400

Russakovsky O. et al., 2015, Int. J. Comput. Vis., 115, 211

Sadler E. M. et al., 2002, MNRAS, 329, 227

Saikia D. J., Konar C., Kulkarni V. K., 2006, MNRAS, 366, 1391

Santurkar S., Tsipras D., Ilyas A., Madry A., 2018, Advances in Neural Information Processing Systems 31: NeurIPS 2018. p. 2488

Saripalli L., Hunstead R. W., Subrahmanyan R., Boyce E., 2005, AJ, 130, 896

Scaife A. M. M., Porter F., 2021, MNRAS, 503, 2369

Scheuer P. A. G., 1974, MNRAS, 166, 513

Schilling H., Bulatov D., Niessner R., Middelmann W., Soergel U., 2018, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., 11, 4299

Schoenmakers A. P., Mack K. H., de Bruyn A. G., Röttgering H. J. A., Klein U., van der Laan H., 2000a, A&AS, 146, 293

Schoenmakers A. P., de Bruyn A. G., Röttgering H. J. A., van der Laan H., Kaiser C. R., 2000b, MNRAS, 315, 371

Sebastian B., Ishwara-Chandra C. H., Joshi R., Wadadekar Y., 2018, MNRAS, 473, 4926

Shallue C. J., Vanderburg A., 2018, AJ, 155, 94

Shimwell T. W. et al., 2019, A&A, 622, A1

Shklovskii I. S., 1955, AZh, 32, 215

Shu F., Kranakis E., 1982, The Physical Universe: An Introduction to Astronomy, Series of Books in Astronomy. University Science Books, New York

Solovyov D. I., Verkhodanov O. V., 2011, Astrophys. Bull., 66, 416

Solovyov D. I., Verkhodanov O. V., 2014, Astron. Lett., 40, 606

Soltan A., 1982, MNRAS, 200, 115

Srivastava N., Hinton G., Krizhevsky A., Sutskever I., Salakhutdinov R., 2014, J. Mach. Learn. Res., 15, 1929

Subrahmanyan R., Saripalli L., Hunstead R. W., 1996, MNRAS, 279, 257

Szegedy C. et al., 2014, IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE Computer Society, p. 1

Tamhane P., Wadadekar Y., Basu A., Singh V., Ishwara-Chandra C. H., Beelen A., Sirothia S., 2015, MNRAS, 453, 2438

Tang H., Scaife A. M. M., Leahy J. P., 2019, MNRAS, 488, 3358

Tang H., Scaife A. M. M., Wong O. I., Kapińska A. D., Rudnick L., Shabala S. S., Seymour N., Norris R. P., 2020, MNRAS, 499, 68

van Haarlem M. P. et al., 2013, A&A, 556, A2

Waggett P. C., Warner P. J., Baldwin J. E., 1977, MNRAS, 181, 465

Walmsley M. et al., 2020, MNRAS, 491, 1554

Williams W. L. et al., 2019, A&A, 622, A2

Williams M. J., Schoneveld L., Mao Y., Klump J., Gosses J., Dalton H., Bath A., Barnes S., 2020, J. Open Source Softw., 5, 2314

Willis A. G., Strom R. G., Wilson A. S., 1974, Nature, 250, 625

Wright E. L., 2006, PASP, 118, 1711

Wu C. et al., 2019, MNRAS, 482, 1211

Zhang X., Wang Y., Zhang W., Sun Y., He S., Contardo G., Villaescusa-Navarro F., Ho S., 2019, preprint (arXiv:1902.05965)

Zhu W. W. et al., 2014, ApJ, 781, 117

This paper has been typeset from a TEX/LATEX file prepared by the author.