Deep Learning in Radio Astronomy

Dissertation

zur Erlangung des Doktorgrades
an der Fakultät für Mathematik, Informatik und Naturwissenschaften
Fachereich Physik
der Universität Hamburg

vorgelegt von

Vesna Lukic

Hamburg, Wintersemester 2019

Gutachter/innen der Dissertation:	Prof. Dr. Marcus Brüggen
Zusammensetzung der Prüfungskommission:	Prof. Dr. Marcus Brüggen Prof. Dr. Gregor Kasiescka Prof. Dr. Francesco De Gasperin Prof. Dr. Jochen Liske Prof. Dr. Robi Banerjee
Vorsitzende/r der Prüfungskommission:	Prof. Dr. Jochen Liske
Datum der Disputtation:	26.11.2019
Vorsitzender Fach-Promotionsausschusses PHYSIK:	Prof. Dr. Günter H. W. Sigl
Leiter des Fachbereichs PHYSIK:	Prof. Dr. Michael Pothoff
Dekan der Fakultät MIN:	Prof. Dr. Heinrich Graener

Zusammenfassung

Aktuelle und kommende radioastronomische Himmelsvermessungen liefern weiterhin neue Erkenntnisse über die Entstehung und Entwicklung von Galaxien, unseres kosmologischen Modells und seiner Parameter. Die vorliegende Arbeit fasst unsere Arbeiten zu Tiefenlerntechniken für die Radioastronomie zusammen. Das Datenvolumen, das bei radioastronomische Durchmusterungen anfällt, ist enorm und nimmt aufgrund von technologischen Verbesserungen ständig zu. Dies führt zu einer steigenden Nachfrage nach der Entwicklung komplexerer Tools zur Analyse der Daten, da manuelle Analysen nicht mehr möglich sind. Um diesen Prozess zu vereinfachen, wurden maschinelle Lerntechniken entwickelt, die sich auf die Voraussetzung stützen, dass sie zum Identifizieren von Mustern und Merkmalen in Daten verwendet werden können. Der Schwerpunkt der vorliegenden Arbeit liegt auf der Analyse von Radiodaten auf der Basis von Bildern mit Hilfe von Deep-Learning-Techniken, ein Ansatz, der sich bei hochdimensionalen Daten bewährt hat. Mit Hilfe von Radio-Galaxien-Bildern aus dem Radio Galaxy Zoo Citizen Science Project demonstrieren wir, dass es möglich ist, drei verschiedene Klassen von Quellen zu klassifizieren. Anschließend testen unser Datennetzwerk mit Daten Release 1. Wir vergleichen die Leistung traditioneller, tief neuronaler Faltungsnetzwerke mit der Leistung von Kapseln Netzwerke. Letztere sind eine in jüngerer Zeit entwickelte Technik, bei der Gruppen von Neuronen verwendet werden, die Eigenschaften eines Bildes einschließlich der relativen räumlichen Positionen von Merkmalen beschreiben. Anhand von Bildern aus der LOFAR-Zwei-Meter-Himmelsvermessung (LoTSS) zeigen wir, dass die herkömmlichen neuronalen Faltungsnetze für die Art der vorliegenden Funkgalaxiendaten eine bessere Leistung erbringen. Schließlich entwickeln wir einen Quellensucher, der auf einem Faltungsautocodierer basiert ist, und vergleichen ihn mit einem hochmodernen Quellensucher unter Verwendung simulierter Quadratkilometer-Array-Daten. Wir stellen fest, dass die Leistung zwischen den Quellenfindern je nach Belichtungszeit, Frequenz und Signal-Rausch-Verhältnis sich variiert.

This thesis is based on the following publications:

- V. Lukic, M. Brüggen, J K Banfield, O. I. Wong, L. Rudnick, R. P. Norris, B. Simmons 'Radio Galaxy Zoo: Compact and Extended Source Classification with Deep Learning', Mon. R. Astro. Soc. 466,1, pp. 246-260 (2018)
- V. Lukic, M. Brüggen, B. Mingo, J.H. Croston, G. Kasieczka, P. N. Best 'Morphological classification of radio galaxies: capsule networks versus convolutional neural networks', Mon. R. Astro. Soc. 487, 2, pp. 1729-1744 (2019)
- V. Lukic, F. De Gasperin, M. Brüggen 'AutoSource: Radio-astronomical source-finding with convolutional autoencoders', submitted to Galaxies, arXiv:1910.03631 (2019)

Abstract

Current and forthcoming radio surveys continue to provide new insights in understanding the formation and evolution of galaxies, our cosmological model and its parameters. The current thesis summarises our work on deep learning techniques applied to radio astronomy. The volume of data produced from radio surveys is vast and constantly growing due to improvements to technology. This results in increasing demand to develop more sophisticated tools to analyse the data, as manual analyses will become unfeasible. Machine learning techniques have been developed to facilitate this process, and rely on the premise that they can be trained to recognise patterns and features in data. The focus of the current thesis is analysing radio data based on images using deep learning techniques, an approach which has proven successful on high-dimensional data. Using radio galaxy images from the citizen science project Radio Galaxy Zoo, we show that it is possible to classify between compact sources and three classes of extended sources, and test our trained network on Data Release 1. We compare the performance of traditional convolutional deep neural networks against Capsule networks. The latter are a more recently developed technique using groups of neurons that describe properties of an image including the relative spatial locations of features. Using images from the LOFAR Two-metre Sky Survey (LoTSS), we show that for the type of radio galaxy data at hand, the traditional convolutional neural networks perform better. Finally, we develop a source finder based on a convolutional autoencoder, and compare the performance against a state-of-the-art source-finder, using simulated Square Kilometre Array data. We find the performance varies between the source-finders based on exposure time, frequency and signal-to-noise ratio.

Eidesstattliche Versicherung

Hiermit versichere ich an Eides statt, die vorliegende Dissertationsschrift selbst verfasst und keine anderen als die angegebenen Hilfsmittel und Quellen benutzt zu haben.

Die eingereichte schriftliche Fassung entspricht der auf dem elektronischen Speichermedium.

Die Dissertation wurde in der vorgelegten oder einer ähnlichen Form nicht schon einmal in einem früheren Promotionsverfahren angenommen oder als ungenügend beurteilt.

(Vesna Lukic)
Brüssel, den 10.10.2019

Contents

1 Introduction **13**
1.1 Sources in the Radio spectrum . . . 13
1.1.1 Introduction . . . 13
1.1.2 Spectral line emission . . . 13
1.1.3 Thermal emission . . . 15
1.1.4 Non-thermal emission . . . 15
1.1.5 Galaxies with significant radio emission . . . 19
1.1.6 Supernova remnants . . . 27
1.2 Radio astronomical telescopes and surveys . . . 27
1.2.1 Main parameters in surveys . . . 27
1.2.2 Science goals and purpose of surveys . . . 30
1.2.3 Past and present (large-scale) surveys . . . 37
1.3 Machine learning and applications to astronomy . . . 41
1.3.1 Supervised learning and common techniques . . . 42
1.3.2 Unsupervised learning and common techniques . . . 47
1.3.3 Machine learning theory . . . 48
1.3.4 Machine learning techniques applied to astronomy . . . 53

2 Compact and Extended Radio Source Classification **63**
2.1 Introduction . . . 63
2.2 Deep neural networks . . . 66
2.3 Methods . . . 67
2.3.1 Pre-processing . . . 67
2.3.2 PyBDSF . . . 68
2.3.3 Image augmentation . . . 69
2.3.4 Deep learning algorithms . . . 70
2.3.5 Selection of sources for two-class classification . . . 72
2.3.6 Selection of sources for four-class classification . . . 73
2.4 Results for two classes . . . 77
2.5 Results for four classes . . . 86
2.5.1 Comparing results with Data Release 1 of the Radio Galaxy Zoo . . . 88

2.6 Conclusions 92

3 Convolutional vs Capsule networks 97
3.1 Introduction 97
3.2 LOFAR HETDEX v1.0 dataset 99
3.2.1 Source cutouts 99
3.2.2 Classifications 100
3.3 Methods 104
3.3.1 Pre-processing 104
3.3.2 Image augmentation 105
3.4 Deep Learning algorithms 105
3.4.1 Convolutional Neural Networks 106
3.4.2 Capsule networks 106
3.4.3 Deep learning parameters 109
3.5 Results 111
3.5.1 LOFAR original images 113
3.5.2 LOFAR original and augmented images 122
3.5.3 Sigma-clipped images 125
3.5.4 Additional results 128
3.6 Conclusions 131

4 Source-finding with convolutional autoencoders 135
4.1 Introduction 135
4.1.1 Source-finding at radio frequencies 135
4.1.2 Types of radio sources 137
4.1.3 Deep learning 137
4.1.4 Simulated SKA data 138
4.2 Methods 141
4.2.1 Convolutional autoencoders 141
4.2.2 Pre-processing 144
4.2.3 Dataset generation 144
4.2.4 Image augmentation 146
4.3 Results 149
4.3.1 Very low significance source metrics at SNR=1 151
4.3.2 Low significance source metrics at SNR=2 152
4.3.3 High significance source metrics at SNR=5 154
4.3.4 Execution times 158
4.4 Discussion and Conclusions 159
4.5 Appendix 162

5 Conclusions and Outlook **167**

6 Bibliography **1**

1 Introduction

The introductory section is composed of three parts; the physics of sources in the radio spectrum, radio surveys and finally machine learning and applications in astronomy, with particular emphasis on its use in radio images.

1.1 Sources in the Radio spectrum

1.1.1 Introduction

Radio astronomy is the branch of astronomy concerned with the study of celestial bodies at radio frequencies, which range from 15 MHz to 300 GHz (20m to 1mm in wavelength). The low frequency limit is set by the opacity of the ionosphere and the high frequency limit is due to the strong absorption from oxygen and water bands in the lower atmosphere. Observing in radio enables us to see objects that are otherwise invisible at other wavelengths (Field & Chaisson, 1985).

In the radio regime, there are two types of emission: spectral line emission and continuum emission. Line emission refers to the radiation emitted at very narrow (discrete) frequency bands, whereas continuum emission covers radio emission from a broad range spectrum of radio wavelengths. Two types of continuum emission are possible; thermal and non-thermal emission. The distinguishing feature is the shape of the spectrum: thermal emission follows a black-body law whereas non-thermal emission shows a power-law spectrum.

Figure 1.1 shows an impression of what we could observe if we looked up at the sky, and had the ability to see radio emission. The small white spots are radio galaxies and the larger structures are supernova remnants.

1.1.2 Spectral line emission

In the radio regime, the 21 cm line is the most important line emission.

The 21 cm line was predicted by H.C. Van de Hulst in 1945, after being asked by his supervisor Jan Hendrik Oort to identify and determine the frequencies of different types of spectral lines

Figure 1.1: An impression of what the sky would look like at radio wavelengths. The small individual white spots are radio galaxies, the larger distortions are due to Supernova remnants. Source: National Radio Astronomy Observatory, Associated Universities, Inc. National Science Foundation

that might exist at radio wavelengths. de Hulst chose to study hydrogen, given its high abundance in the interstellar medium, after which he predicted that transitions in the energy levels in hydrogen should produce radiation at 21 cm wavelengths (de Hulst, 1945). The radiation was observed for the first time in 1951 using a microwave radiometer, appearing as emission with a width of $\sim$ 80 kc. The detected source appeared to be extended and approximately centred about the galactic plane (Ewen & Purcell, 1951).

The observation of the 21 cm line provided much knowledge about the structure and interaction of galaxies, both within our Galaxy and other galaxies in the Universe. For example, observations of the 21 cm line of neutral hydrogen in the Milky way disk showed that the disk extends to at least two to three times the radius of the solar circle, and that the HI disk in the outer Galaxy is warped (see Dickey et al. (2009) and references therein). Detectable HI is an exceptionally sensitive tracer of tidal interactions between galaxies (e.g. Lelli et al. (2015) and references therein).

In terms of extragalactic observations, the manifestation of dark matter through galaxy rotation curves has been observed using the 21cm line, one example being with the use of the Westerbork Synthesis Radio Telescope (WSRT) (Rogstad & Shostak, 1972; Bosma, 1978). The accurate detection of the 21cm line from the epoch of reionisation is a powerful tool to investigate the neutral Intergalactic medium (IGM), which can be used to probe information at high redshift such as matter density fluctuations and the thermal history of the IGM, therefore informing about the evolution of the early universe (Colafrancesco, S. et al., 2016; Pritchard & Loeb, 2012).

1.1.3 Thermal emission

Thermal emission, also known as black-body radiation, is emission due to the temperature of the body. Radio emission from planets in our solar system was found to be thermal in origin. One example of a study involved the use of the CSIRO 210-ft radio telescope in measuring the thermal radio emission from several planets in the solar system between wavelengths of 6 and 48 cm (Kellermann & Pauliny-Toth, 1966).

1.1.4 Non-thermal emission

Non-thermal emission, which follows a power law, is independent of the temperature of the source. Extragalactic non-thermal radio sources are the source type of focus in the current work.

Two of the main mechanisms behind non-thermal emission are synchrotron radiation and Compton/Inverse Compton radiation.

Synchrotron radiation

Synchrotron radiation is a result of relativistic electrons spiraling around magnetic field lines.

A single electron will emit radiation over a range of frequencies that peaks at some frequency ν_{max}, also known as the critical frequency. In realistic cases, the distribution of the energy spectrum is composed of an ensemble of electron energies which all have their own individual peak in frequency, as shown in Figure 1.2.

Since the energy spectrum of electrons emitting synchrotron radiation does not follow a Maxwellian distribution, the emission is non-thermal and the distribution of electron energies follows a power-law. If the relativistic plasma is transparent (optically thin) to its own radiation, which occurs in the extended regions of radio sources, the power-law takes on the form $N(E) = KE^{-p}$ (Kellermann & Owen, 1988), where $N(E)$ is the relativistic electron energy distribution, p is the index of the electron energy distribution, $E = \gamma mc^2$, where the Lorentz factor $\gamma = \frac{1}{\sqrt{1-\frac{v^2}{c^2}}}$. The radiation spectrum is a power law with

$$S(\nu) \propto \nu^{\alpha}, \tag{1.1}$$

where $\alpha \sim \frac{1-p}{2}$ is the radio spectral index. The slope of the spectrum is determined by the electron energy distribution. Figure 1.3 shows the variation of radio spectral index (flux density with respect to frequency) across four radio sources. This variation was first observed in Cygnus A (Mitton & Ryle, 1969), where the radio spectral index flattens towards the hotspots of the source, and is the second flattest compared to the compact core (Hargrave & Ryle, 1976). The effect has been explained as the aging of the relativistic electrons due to synchrotron emission. Electrons at higher frequency have a higher energy and will be depleted first, resulting in a steeper slope over time (Scheuer & Williams, 1968). The age of the electrons that generated the emission can therefore be determined by the slope of the spectrum.

Synchrotron emission accounts for most of the radio emission from Active Galactic Nuclei (AGN) thought to be powered by supermassive black holes in galaxies and quasars (Burbidge, 1956) and appears to be responsible for generating the morphologies observed in radio sources (Fanaroff & Riley, 1974).

Re-absorption of the synchrotron electron radiation becomes important if the intensity of synchrotron radiation within a source rises above a certain threshold, an effect referred to as synchrotron self-absorption (Kellermann & Verschuur, 1988). It results in a drastic modification of the spectrum at low frequencies.

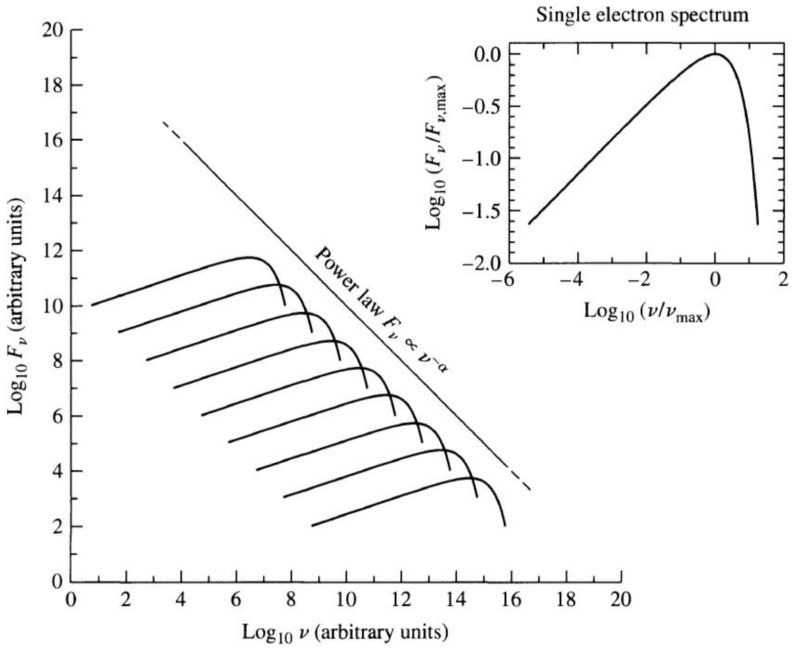

Figure 1.2: Showing the spectrum of an individual electron on the top right, as well as how the spectrum from an ensemble of electrons is obtained. Figure taken from Carroll & Ostlie (2006).

Figure 1.3 shows the spectra of flux density against frequency of four radio sources.

The magnetic fields responsible for synchrotron radiation have an effect on the interstellar medium. Magnetic fields are an important factor driving star formation within star forming clouds. Models generated on a supercomputer showed that stellar winds interacting with the magnetic field of the cloud generated energy and influenced gas at far greater distances than previously thought (S. R. Offner & Liu, 2018).

Magnetic fields in interstellar and intergalactic space have traditionally been measured in the four following ways (Beck & Wielebinski, 2013): (i) Observing starlight polarization and polarized dust emission (Davis & Greenstein, 1951; Hoang & Lazarian, 2008). Interstellar space contains tiny dust grains, generally aspherical in shape and rotated by the ambient radiation, preferring to align their short axes with the local magnetic field, thereby making it possible to measure a component of the interstellar magnetic field. (ii) Zeeman splitting of emission or absorption lines (Troland & Heiles, 1982). The strength and direction of the magnetic field is provided by the circular polarisation driven by Zeeman Splitting. (iii) Through detecting synchrotron radiation as it requires both relativistic electrons and a magnetic field (Webber et al., 1980), by invoking the equipartition argument (Beck & Krause, 2005), where it is assumed that the total energy is at a minimum when the magnetic field strengths and energies of the relativitistic particles are approximately equal. (iv) Through Faraday rotation as a result of a magnetic field amongst relativistic electrons (Burn, 1966). Measurements of

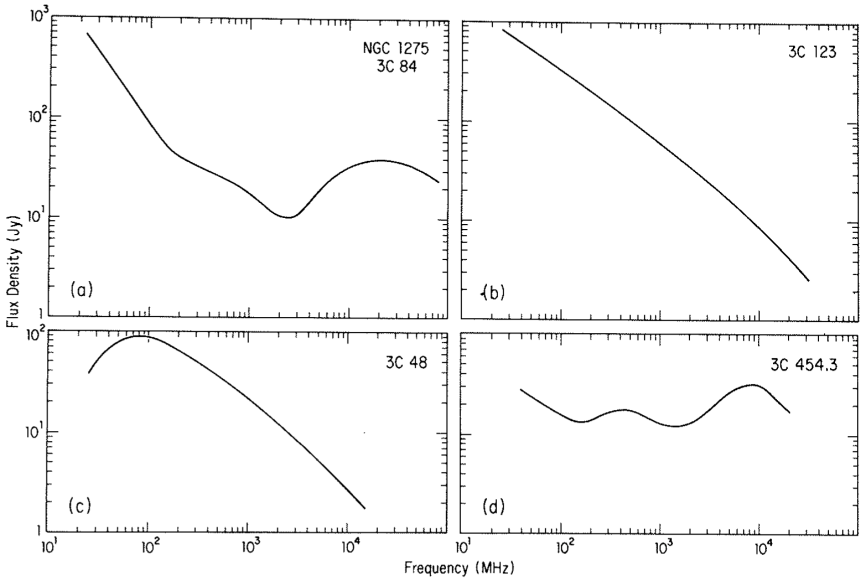

Figure 1.3: Showing the spectra of four radio sources. The top left panel shows the radio source 3C 84 that has a very compact component, 3C 123 in the top right panel is transparent across all frequencies, 3C 48 in the bottom left shows self-absorption below 100 MHz and is transparent above this value, and the bottom right panel shows opaqueness at different frequencies. Figure taken from Kellermann & Verschuur (1988).

Faradays rotation measure on increasing numbers of extragalactic radio sources has led to a more refined modeling of the large scale magnetic field structure.

Compton and Inverse Compton radiation

Compton scattering refers to the energy transferred from a photon to a charged particle, usually an electron. Inverse Compton (IC) scattering occurs when the electron has significant kinetic energy compared to the energy of the photon, which causes the scattered photon to have a higher frequency, and energy is transferred from the electron to the photon. If photons propagate through a distribution of energetic electrons, they can gain or lose energy depending on their energy relative to the electron temperature.

Despite the low radiation density of ultra-relativistic electrons in intergalactic space, the electrons might lose significant amounts of energy by the IC process (Feenberg & Primakoff, 1948). The IC process has been under consideration in regard to whether it can account for the observed X-ray background, as well as the role it plays in the synchotron radiation in compact and intense radio sources, such as quasars (see Rees (1967) and references therein.)

It is inferred that IC scattering must occur in the lobes of radio sources, as the cosmic microwave background permeates all of space, so every synchrotron radio structure should have a corresponding X-ray structure. Feigelson et al. (1995) was the first to detect the IC

radiation from X-rays in the lobes of nearby radio galaxy Fornax A.

1.1.5 Galaxies with significant radio emission

The early universe contained many small lumps of matter which coalesced to form galaxies. This forms the basis of hierarchical cosmological models. A traditional approach to understanding the physics underlying galaxy formation is to use physical observables of galaxies and their relations to each other, in order to test models (Conselice, 2012). The ability to perform simulations has played a major role in furthering our progress in understanding galaxy formation.

A few established facts about galaxy formation are as follows. Galaxy formation is strongly influenced by the Interstellar Medium (ISM), black holes also play a major part, and galaxies form through the cooling of gas at the centers of dark matter halos, where the gas condenses into stars (White & Rees, 1978).

Despite the existing knowledge, there are several remaining key questions regarding galaxy formation. For example, how did the first stars and galaxies evolve to produce the galactic structures that exist today? What are the underlying physical processes regulating the structure of the ISM? The processes driving star formation and galactic outflows are not yet understood. Another major theoretical challenge is understanding stellar and AGN Feedback in detail and to identify physically correct sub-resolution models taking into account all relevant physical processes.

Radio astronomy has made important contributions to the study of galaxy formation and evolution. Through production of the CMB map, it has provided information on the origins of large-scale structure, as well as refined the values of cosmological parameters (Partridge, 2011). The following subsections describe the extragalactic sources detectable at radio wavelengths, namely star-forming galaxies (SFGs) and Active Galactic Nuclei (AGN), in regards to the knowledge gained about them from radio surveys. The study and separation of these two galaxy populations is important because they are fundamentally different classes of objects with different properties.

Star-forming galaxies

There are two fundamental types of galaxies: red sequence and blue cloud galaxies, both of which occupy different regions on a color-magnitude diagram that plots the relationship between luminosity and mass of the galaxy. Galaxies in the red sequence category tend to be elliptical and contain relatively little gas and dust compared to the galaxies found in the blue cloud category, which tend to be spiral and have higher star formation rates (SFR) (Strateva

et al., 2001). Therefore, galaxy populations can be traced through star formation. SFR is key in characterising galaxies, and observing how the average SFR changes is an important factor influencing the evolution of the universe.

Star-forming galaxies (SFGs) are defined as those having a high SFR that is strongly correlated with total stellar mass (Maragkoudakis et al., 2016). They can be used to help measure the star formation rate in the Cosmos. There are three types of SFGs: normal late-type, starburst and proto-spheroidal galaxies. SFGs show both thermal and non-thermal emission, emitting at radio wavelengths due to both synchrotron and free-free radiation processes; they are characterised by steep GHz radio spectra, but also have a flat free-free component (Padovani, 2016). The thermal emission can be due to the dust grains in these galaxies.

A fundamental ingredient for star formation is neutral atom hydrogen (HI), making it an ideal candidate to probe the rate of star formation and therefore whether or not a galaxy can be classed as being a SFG (Rhee et al., 2018). The HI content can also be used to separate them from other types of galaxies that would contain less HI.

Although past radio surveys have detected a greater number of AGN, the increased sensitivity of surveys over time has enabled the detection of an increasing number of SFGs, found to constitute a significant fraction of the faint radio sky (Norris et al., 2006; Padovani et al., 2014). This makes SFGs promising in being able to chart the cosmic history of star formation. The study of SFGs in the radio regime has led to the discovery that their emission is tightly correlated with the SFR (Kennicutt & Evans, 2012). As such, radio observations have been used to measure the rate at which galaxies form new stars (Tabatabaei et al., 2017).

There also exists a correlation between the Far infrared (FIR) and radio emission of ordinary and SFGs (Helou et al., 1985), which can be expressed as $S_{1.4} = 10^{-q} S_{FIR}$, where $S_{1.4}$ is the radio frequency flux density at 1.4 GHz and S_{FIR} is the FIR flux. The exponent q appears to be independent of the source luminosity as well as redshift, which has interesting consequences for star and galaxy formation. The correlation holds over a very wide luminosity range, although care should be taken to exclude potential AGN components of the radio emission, as it would violate this relation. It is presumed that star formation produces both the dust re-emission that dominates the FIR luminosity, as well as the supernovae that produce relativistic electrons and hence the synchrotron radiation (Partridge, 2011).

Radio observations are able to probe very recent star formation activity and to some extent trace its location (Padovani, 2016), due to the synchrotron emission from SFGs being a result of relativistic plasma accelerated in supernova remnants that are associated with massive star formation.

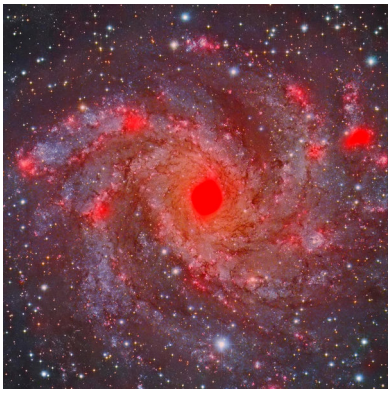

Figure 1.4: Showing the optical image superimposed with the radio emission in the 3cm band (red) in galaxy NGC 6946. The radio emission indicates regions of star formation. Image taken from (Tabatabaei, 2017).

Active Galactic Nuclei (AGN)

AGN are the most luminous objects in the Universe, up to $L = 10^{48}$ ergs s^{-1}. They are powered by a super massive black hole (SMBH), which accretes gas and dust from its surroundings. AGN come in different types, such as Quasars, Seyfert-galaxies and radio galaxies. The two main categories that AGN fall into are radio-loud, displaying powerful jets, and radio-quiet, with very weak jets. There are many other ways of classifying AGN; the different ways constitute an entire AGN 'zoo' (Padovani, 2017). The majority of the current thesis is concerned with classifying the morphologies arising from radio-loud AGN, which in the broadest sense can be compact or extended in morphology.

In the past, AGN were mainly studied as laboratories in which to probe exotic high-energy processes. Despite attempts to understand the role that the environment might play in triggering or fueling the AGN, there was virtually no concept that AGN played any significant role in the evolution of typical galaxies. The current status is very different: there is very strong evidence in support of the idea that the evolution between galaxies and AGN is strongly linked (Heckman & Best, 2014).

There are two modes of AGN Feedback; radiative-mode and jet-mode. The radiative mode occurs when most of the energy is released by radiation or strong winds originating from the black holes accretion disk, usually associated with high luminosity AGN. The jet mode is most likely the dominant mode in low-power AGN and is due to the presence of jets of relativistic plasma depositing kinetic energy into the surrounding medium. AGN Feedback mechanisms are used in both semianalytic models and numerical simulations to successfully produce the properties observed in massive galaxies (Cattaneo et al., 2009; Fabian, 2012).

Due to their large luminosity, AGN are observable to very high redshifts (Mortlock et al., 2011); therefore they serve as a cosmological probe to the early universe. For example, the

absorption lines of some AGN types show hints of cosmic reionisation (Keel, 2007). AGN provided evidence for the existence of supermassive black holes, such as through the study of emission-line variability data in Seyfert 1 galaxies (Peterson & Wandel, 2000). AGN are key players in the process of galaxy formation and evolution, as evidenced by the discovery of the tight correlation between galaxies and properties of the central nucleus (Kormendy & Ho, 2013), and similar evolutionary trends between the growth histories of supermassive black holes and galaxies (e.g. Boyle & Terlevich (1998), Marconi et al. (2004)). It is also possible to probe galaxy evolution by investigating the role of supermassive black holes by measuring the host galaxy stellar mass function (Bongiorno et al., 2016). AGN also have the ability to heat, relocate and in some cases remove the surrounding gas from the host galaxy (Brienza, 2018).

The study of AGN has enabled the derivation of the SMBH mass function, defined as the comoving number density of black holes per bin in log mass at a given redshift. Several previous analyses reviewed in Heckman & Best (2014) indicate that there is a population of more massive black holes produced at higher redshifts. Since a redshift of $z \sim 1$, only the population of black holes with masses below approximately 10^8 solar masses has been growing significantly, and that the population of black holes with masses $> 10^9$ solar masses has been largely in place by approximately $z \sim 2$.

The use of AGN to study SMBH, together with using star formation as a tracer for galaxy populations revealed that their evolution is very similar: a steep rise in both the star formation rate (SFR) and SMBH growth rate by about a factor of 10 from redshift $z = 0 - 1$, a broad maximum in both rates at $z \sim 2 - 3$, and then a relatively steep decline at higher redshifts (see Shankar et al. (2009) and references therein).

Radio galaxies

Radio galaxies fall into the category of radio-loud AGN that are some of the most unusual and powerful objects in the Universe, having extents varying between the order of 1 pc or less, up to the Mpc scale. They are powered mainly through the synchrotron emission mechanism and tend to be elliptical galaxies (Rogstad & Ekers, 1969).

The main components of radio galaxies tend to be a core, jets and lobes. Energy is generated in the core of the AGN and expelled from it in the form of two opposing relativistic beams (Longair & Riley, 1979), which can travel vast distances before spreading out into giant, radio-emitting lobes. The components (core, jet and lobes) have different spectral shapes as their relative strength depends on frequency. The structure and spectra of radio emission from radio galaxies contains information on the history of AGN activity in the source (Saikia & Jamrozy, 2009).

The jets are very well collimated structures (Blandford & Rees, 1974), with opening angles of no more than a few degrees. They can be one-sided (Cohen & Unwin, 1984), meaning that only one jet can be observed. There have been a couple of long-unresolved questions regarding the origin of radio jets, such as from what energy reservoir the large radio luminosities (up to 10^{38} W between 10 MHz and 100 GHz) are drawn, and how does the AGN supply such high luminosity relativistic particles and fields to the radio lobes (Bridle & Perley, 1984). There have been mainly two theories regarding the origin of jets (Rees, 1982): (i) an ion-supported torus (accreting disk) anchors magnetic fields which extract rotational energy from the hole in the form of two collimated beams of relativistic particles and fields e.g. (Rees et al., 1982) and (ii) a radiation-supported torus, where due to centrifugal effects greatly enhancing the effective gravity along the cylindrical walls surrounding the axis of rotation, the radiation might cause the ejection of jets e.g. (Jaroszynski et al., 1980). Almost all the information available in regard to jets relates to their morphology and luminosity.

Fanaroff & Riley (1974) were the first to notice the correlation between radio luminosity and the relative positions of high and low surface brightness in the lobes of extended extragalactic radio sources. The pattern of the emission can be characterised into two main morphological types; (1) FRI, where the brightest part of the emission is closest to the core of the source, having typical luminosities of $L_{\nu=1.4\mathrm{GHz}} \leqslant 10^{32}$ ergs s^{-1} Hz^{-1}, and (2) FRII, where the lobes are the components having the brightest emission, with typical luminosities of $L_{\nu=1.4\mathrm{GHz}} \geqslant 10^{32}$ ergs s^{-1} Hz^{-1}. However, there is not a clear separation between the two classes.

The jets of FRI radio sources tend to be less collimated than those of FRIIs, indicating they are weaker and show a deceleration (Laing & Bridle, 2002). The regions of high brightness tend to be located closer to the host galaxy, and the sources become fainter towards the outer parts of the lobes where the spectra are the steepest, indicating that the emitting particles have aged the most (Kembhavi & Narlikar, 1999). Figure 1.5 shows the different appearances that FRI radio galaxies can have.

FRII radio sources often have jets between the compact core and the radio lobes, which often show some internal structure, as well as hotspots at the outer edges of the radio lobes. The jets tend to be more powerful compared to those from FRIs, and therefore have smaller opening angles (Bridle & Perley, 1984). The FRIIs are likely to have supersonic jets, and the hot spot might be where the jets meet the ambient medium and decelerate through a shock transition. The emission pattern of the FRIIs suggests that energy is carried away from the core to the lobes. Figure 1.6 shows a variety of morphologies available for FRIIs.

There are also smaller subgroups of morphologies such as Hybrids (Gopal-Krishna & Wiita, 2000), which display a mixture of FRI and FRII morphologies. Further classes include the Wide-Angle Tail (WAT) (Begelman et al., 1979), Narrow-Angle Tail (NAT), double-double (Schoenmakers et al., 2000) and HyMORs (Kapińska et al., 2017) morphologies. Additionally,

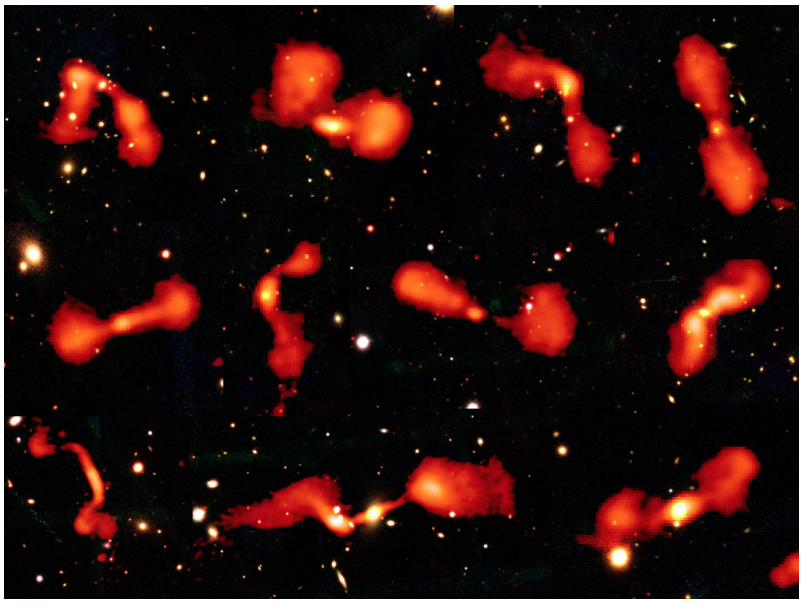

Figure 1.5: Common appearances of FRI radio galaxies. Source: Judith Croston and the LOFAR Surveys team (private communication).

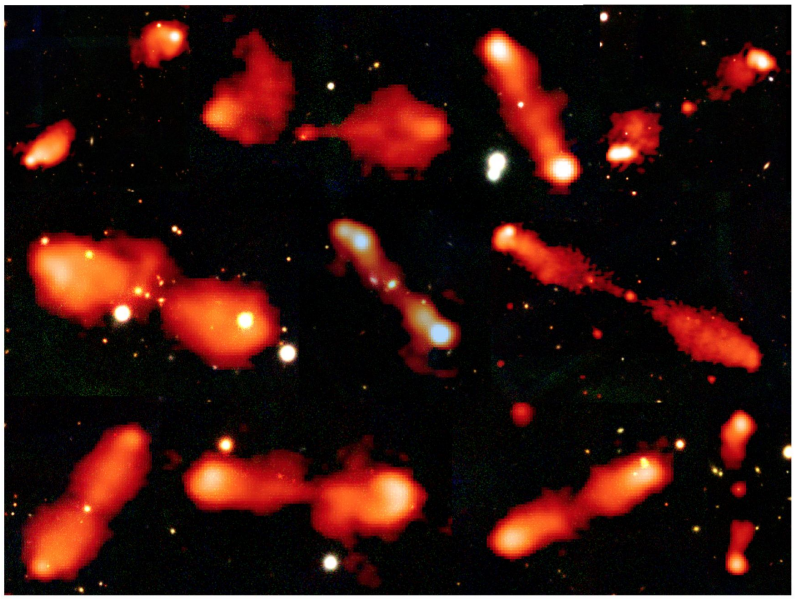

Figure 1.6: Common appearances of FRII radio galaxies. Source: Judith Croston and the LOFAR Surveys team (private communication).

some extended sources may have a compact morphology, and there has been some debate as to whether they should form a class in their own right, or if they are FRI/FRII radio galaxies in early stages of evolution, or whether they are short-lived sources (Miraghaei & Best, 2017).

One unresolved question is the origin of the FRI and FRII sources, also known as the FRI/FRII dichotomy, and its resolution will help us to understand the processes which initiated nuclear activity (Ferrarese & Celotti, 2002). The differences between the FRI/FRII jets may be associated with the central engine and/or external medium. There are two competing theories for explaining the dichotomy. The first theory is that there could be intrinsic differences in the jet's kinetic power between the sources (Baum et al., 1995) or fundamental AGN parameters. The second is that the deceleration process of the jets causes the differences in the sources (Kawakatu et al., 2009; Meliani & Keppens, 2009), attributed to the different physical conditions in the environment in which the radio source propagates. Gopal-Krishna & Wiita (2000) presents a more detailed summary of the dichotomy.

The classification of radio galaxies, which can be done according to their morphology, is important as the morphologies can provide information about the surrounding environment of the radio galaxy, at different redshifts. Some examples include the discovery that tailed radio galaxies occur preferentially in high-density regions of the intracluster medium (Burns et al., 1994), and can also trace its weather. At nearby redshifts ($z < 0.5$), FRIIs are often found to be hosted by field galaxies while the FRIs are found to be located in galaxy clusters (Saripalli, 2012). At higher redshifts, both FRI and FRII type radio sources are found in rich environments (Hill & Lilly, 1991).

The broadest categories that radio galaxies can be divided into are compact or extended sources. Sources may be compact due to being unresolved by the telescope, however it is also possible to have a class of resolved compact sources (Baldi et al., 2016b). In terms of spectral indices, compact sources tend to have flat spectra, whereas extended sources have steep spectra. The way the radio galaxies are orientated must account for some observed differences in their appearance. For example, the jets may radiate anisotropically giving the appearance that one jet may be brighter than the other, whereas if the radio galaxy was orientated differently the jets may appear equally bright.

A recent work by Hardcastle (2018) notes that the difference between the two morphological classes is not the same as the difference between the two types of jets seen in the FRI and FRII sources. Therefore, the distinction between the classes is much less clear than previously thought.

There are other radio galaxy types that display different morphologies. One such example is remnant radio galaxies, where the remnant phase refers to the end of the radio galaxies life cycle when the jets switch off, usually displaying curved steep spectra and a relaxed

morphology without compact components, however overall there is a range of morphologies available to them (see Brienza et al. (2016) and references within). Another source type is restarted radio galaxies which display remnant lobes associated with active jets. They usually have a 'double-double' morphology, where the centre of the radio galaxy is in line with the two radio lobes (Schoenmakers et al., 2000).

The duty cycle of radio galaxies refers to phases of high and low jet activity. Different duty cycles are likely to have resulted from the two AGN modes (radiative mode and jet mode), as they are also related to two accretion modes acting on different timescales. Radiative-mode AGN produces highly energetic but short-lived AGN activity, whereas it is believed that jet-mode AGN go through a self-regulated feeding and feedback loop, where the same gas fueling the black hole gets regularly heated by it and stops it being accreted, resulting in cyclically active behaviour for most of the galaxies life (Best et al., 2005).

Flux-limited radio surveys detect powerful radio sources such as FRIIs at relatively higher redshifts, whereas the low-power radio sources such as FRIs are identified at lower redshifts. The slope of the radio luminosity function leads to the prevalence of low power sources at lower redshifts, resulting in a bias and therefore limited knowledge of the relative abundance of low-power sources at higher redshifts (Saripalli, 2012). Radio luminosity functions suggest that FRIs, which are located mainly in galaxy clusters, are frequently triggered and spend over a quarter of their time in an active state whereas FRIIs are more rarely triggered, remaining active for short periods of time (Best et al., 2005; Shabala et al., 2008).

The probability of a galaxy being radio–loud is a strong function of its optical luminosity, a result shown using the bivariate radio–optical luminosity function (Auriemma et al., 1977; Sadler et al., 1989; Ledlow & Owen, 1996). Other statistical studies of luminosity functions such as Best et al. (2005) show that the fraction of galaxies which host radio–loud AGN is very strongly related to both stellar and black hole mass. The distribution of radio luminosities does not generally depend on black hole mass, and that within the range of radio luminosities studied, radio and emission-line AGN activity are independent of each other. Shabala et al. (2008) constructed a bivariate luminosity function and used it to constrain the time a radio source is inactive. They found that radio and emission line AGN activity are independent phenomena, and that the radio source lifetime and duration of the quiet phase of AGN activity depends strongly on mass, where massive hosts possess longer-lived sources that are triggered more frequently. Sabater et al. (2019) show that stellar mass is a more important driver of radio-AGN activity than black hole mass, which suggests a possible connection between the fuelling gas and the surrounding halo.

In regard to optical spectra, almost all FRI radio galaxies are Low-Excitation AGN, while optical hosts of FRIIs usually have strong emission lines and are thus classified as High-Excitation AGN. However, there is not a direct correspondence between the FR and emission

line classes as many FRII radio galaxies are also Low-Excitation AGN (Evans et al., 2006).

1.1.6 Supernova remnants

Supernova remnants (SNRs) are the remains of a supernova explosion, and emit synchrotron radiation. They are made up of material expanding from the explosion, bounded by a shock wave. There are two ways that SNRs could form; either due to a massive star exhausting its fuel and collapsing under its own gravity, or from the accretion of material in a likely dwarf star binary system and undergoing a thermonuclear explosion upon reaching a critical mass. Approximately 10^{51} ergs of mechanical energy are ejected in the interstellar medium as well as several tens of solar masses of stellar material, regardless of the origin of the explosion. The first SNRs were detected in our galaxy from radio observations.

1.2 Radio astronomical telescopes and surveys

The aim of performing astronomical surveys in general is to produce a catalogue of astronomical objects using measurements from telescopes that describe properties of interest, such as total flux, size and position. The subsequent analysis of individual sources in the catalogue and/or source populations with the use of statistics, serves to improve our current understanding of the Universe.

In contrast to the use of optical telescopes, there is more freedom in choosing the locations where radio telescopes will be built; usually the constraint is that they should be located in uninhabited places, away from radio transmission and television broadcasting, to reduce interference effects.

The subsequent subsections discuss radio all-sky surveys, in particular the parameters, instruments and science goals, as well as discoveries from past and present radio surveys. We also discuss surveys planned in the future and what they might find.

In addition to continuum surveys, three other types of surveys are possible at radio wavelengths, namely spectral line, polarisation and galactic plane surveys. The current thesis is concerned with radio sources detected using continuum surveys.

1.2.1 Main parameters in surveys

Prior to exploring the different radio surveys that have been done, are currently in progress or surveys planned in the future, it is important to know the parameters that define them, as differences in these help highlight the survey's strengths and weaknesses as well as differences

in properties of the detected sources. The key parameters involved in continuum surveys are the frequency, sensitivity (depth), angular resolution, noise, as well as the survey area, speed and field of view.

Frequency

At radio frequencies, the non-thermal synchrotron emission is enhanced whereas the thermal (black-body) emission is dampened. The use of a frequency or range of frequencies in radio surveys affects the type of radio sources that are selected (Simpson, 2017), as well as their measured properties. For example, low frequencies reveal extended structures in greater detail compared to when higher frequencies are used, therefore the same source across the two frequencies will have a difference in size.

Early radio continuum surveys utilized low frequencies owing to technical limitations, but also because most radio sources are stronger at low frequencies. However, they also tend to have lower resolution and are affected by radio-frequency interference. Higher frequencies offer a higher resolution, positional accuracy and dynamic range, and are sensitive to free-free emission, at the expense of being less sensitive to extended structures.

More recently, there has been renewed interest in utilising low frequencies for purposes such as detecting neutral hydrogen at cosmological distances and investigating ultra-steep spectrum radio galaxies at high redshifts (Brienza, 2018).

Sensitivity

Radio signals are relatively weak and typically measured in Janskys ($10^{-26}\mathrm{W/m^2/Hz}$). Sensitivity is a measure of the weakest possible detectable source of radio emission (Wrobel & Walker, 1999), and is synonymous with depth.

The radio sensitivity is proportional to the signal-to-noise ratio, which depends on the effective area of the telescope and the system temperature, all of which are described in more detail in the subsections below.

Angular resolution

The resolution determines the limit by which two objects that are known to be separate, can also be seen as such. This depends on the aperture of a telescope. The radio waves diffract around the telescope aperture and produce a diffraction pattern.

Resolution is defined in the same way as it is for optical telescope instruments: $\theta = \lambda/D$, where λ is the wavelength and D is the diameter of the radio telescope or the baseline length, if radio interferometry is used. The order of magnitude difference in wavelengths between optical and radio causes a large difference in the resolution. Due to this difference, in the early days of radio astronomy the angular resolution of a radio telescope was much weaker compared to that achieved in optical astronomy.

In single radio telescopes, antennas with diameters of several kilometres are needed to achieve the same optical resolution as that of an optical telescope, which is unfeasible. This is what brought about the development of radio interferometers, which are radio antenna arrays used simultaneously in astronomical observations, to mimic a single telescope with a very large aperture.

Noise

The noise can be broadly defined as the uncertainty in the output signal that should be dominated by statistical fluctuations in the photons that produce the signal and ideally free of systematic effects.

The noise is expressed in terms of temperatures, where the noise terms add linearly:

$$T_{\rm sys} = T_{\rm cmb} + T_{\rm rsb} + \Delta T_{\rm source} + [1 - e^{-\tau_A}]T_{\rm atm} + T_{\rm spill} + T_{\rm r} + ..., \tag{1.2}$$

where $T_{\rm cmb}$ is the cosmic microwave background, $T_{\rm rsb}$ is the radio source background, $\Delta T_{\rm source}$ is due to the astrophysical source, $\tau_{\rm A}$ is the optical depth, $T_{\rm atm}$ is the opacity of the atmosphere due to the absorption of the signal and extra thermal emission, $T_{\rm spill}$ is due to the imperfect illumination of the dish or subreflector, and $T_{\rm r}$ is the receiver background.

It is expected that longer integration times should reduce the noise levels. The radiometer equation can be used to estimate the noise with respect to integration time (Johnson, 1928):

$$\sigma_T \approx \frac{T_{sys}}{\sqrt{\tau \Delta\nu}}, \tag{1.3}$$

where $\sigma_{\rm T}$ is the temperature fluctuation, $T_{\rm sys}$ is the noise, τ is the integration time and $\Delta\nu$ is the bandwidth.

The source confusion is also considered to belong under the category of noise. Confusion is defined as the inability to measure faint sources due to the presence of other sources. Confusion usually limits the sensitivity of single-dish continuum observations at frequency below approximately 10 GHz.

Field of view

For a single dish antenna of diameter D, the width of the field of view of the antenna is equal to its angular resolution. The field of view is also referred to as the beam, and determines whether or not a source will be a point source, or an extended one. For point sources, the flux density measured by a telescope is the brightness integrated over the entire source. For extended sources, some part of the source will fall beyond the field of view, in which case the brightness will be calculated over the beam size.

Survey speed, depth, area

The survey speed is the amount of time required to complete a survey, and is proportional to the number of sub-arrays in the collecting area, the field of view and their number, and inversely proportional to the integration time per sky position.

Different radio surveys detect sources in different areas of the sky. The area of a survey is usually measured in deg^2. The smaller the survey area, the greater the cosmic variance and increased likelihood that they would miss intrinsically small objects. On the other hand, surveys that have covered wider areas have been relatively shallow, and so may have missed the most active epochs of galaxy formation (Norris et al., 2006).

The depth of a survey is synonymous with the sensitivity. The number of sources detected in a survey is affected by the survey area and sensitivity. There is usually a trade-off between the two; the number of sources detected can be increased by focusing on a smaller area but at greater sensitivity, or over a larger area at decreased sensitivity. Technological limitations, cost and time required are obstacles in maximising both the sensitivity and area.

Surveys can cover parts of the two hemispheres, for example FIRST and NVSS examined the northern hemisphere, which is currently undertaken with LOFAR. The PARKES telescopes examined the southern hemisphere and will be covered by the SKA in the future. There have been several all-sky surveys such as TIFR GMRT Sky Survey (TGSS), The GaLactic and Extragalactic All-Sky MWA Surve (GLEAM), and VLASS at present.

Figure 1.7 is a plot of the logarithm of area in units of degrees versus the logarithm of sensitivity across a selection of radio telescopes, in units of mJy. It shows the trend of their increasing sensitivity with respect to area.

1.2.2 Science goals and purpose of surveys

Undertaking continuum surveys in the radio regime helps us to understand the formation and evolution of galaxies in the early Universe. In particular, continuum surveys shed light on

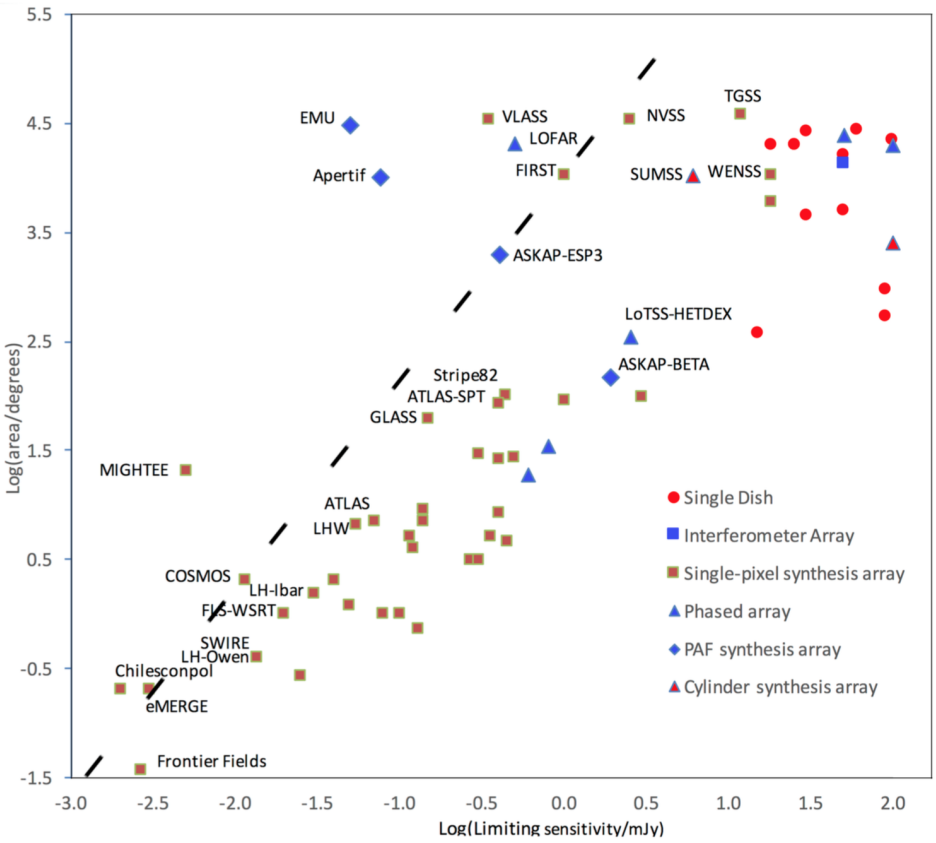

Figure 1.7: The logarithm of area versus limiting sensitivity across the large-scale surveys. Figure taken from (Norris, 2017a).

the evolution of star formation and AGN, galaxy clusters and the detection of dark matter, cosmology and the determination of its parameters, as well as making unexpected discoveries, all of which are discussed in further detail in the following subsections.

The physics and evolution of star formation

There are several scientific aims that radio surveys will attempt to address in regard to learning about SFGs, such as a better understanding of the cosmic history of star formation and star formation rate density (SFRD) curve, uncovering the physical basis of the FIR correlation, being better able to distinguish between radio-loud AGN and SFGs using the FIR correlation in SFGs, measuring star formation rate (SFR) accurately (Norris, 2017a), as well as modeling the evolution of the luminosity function of SFGs (Bonato et al., 2017).

The cosmic history of star formation is one of the most fundamental observables in astrophysical cosmology. Its study can help astronomers to map the transformation of gas into stars, investigate the production of heavy elements, as well as the reionisation of the Universe from the dark ages to the present epoch. Determining the early history of star formation is required to help establish whether massive stars in young SFGs are responsible for cosmic reionisation (Madau & Dickinson, 2014).

Different methods of measuring the SFR have given different values, largely due to the inaccurate determinations of extinction corrections (Madau et al., 1998). Future surveys will enable a better characterisation of the SFRD curve, as many more SFGs will be discovered compared to in previous surveys.

Future radio surveys will detect more SFGs compared to using FIR surveys, therefore giving larger sample sizes, including up to high redshifts. Radio waves provide a more reliable way to measure SFR as they are not affected by dust extinction compared to using FIR emission (Calzetti, 2001; Mancuso et al., 2015). As such, radio surveys offer a good way to measure the cosmic star formation history of the universe. However, special care needs to be taken to disregard radio emission from a potential AGN component. By discovering the physical basis of the FIR correlation, it will shed light on how the many physical processes such as the propagation of relativistic electrons, strength and structure of the magnetic field, size and composition of the dust grains, must work together to produce the observed relation (see Mancuso et al. (2015) and references within.)

Assessing the relationship between radio continuum luminosity and SFR is of crucial importance to make reliable predictions for upcoming ultra-deep radio surveys and in using their results to measure the cosmic star formation history (Bonato et al., 2017).

The physics and evolution of AGN

One of the biggest questions yet to be resolved in astronomy is understanding the difference between radio-loud and radio-quiet AGN. For example, investigating why the host properties of radio-loud AGN depend on redshift is of great interest. Many papers have discussed the potential mechanisms responsible for the radio emission observed in radio-quiet AGN (see Padovani et al. (2014) and references therein). Despite evidence that SMBHs and their host galaxies have co-evolved, the mechanisms driving this co-evolution remains uncertain (Shankar et al., 2009; Kormendy & Ho, 2013; Heckman & Best, 2014).

The brightness of a SMBH in an AGN can be influenced by the host galaxy's environment, making AGN important tools for understanding the evolution and formation of structure in the Universe (see Bieri (2016) and references within).

Future radio surveys will hopefully help uncover the physics driving the origin of the FRI/FRII dichotomy and the mechanism driving jets.

The study of the duty cycle (phases of high and low jet activity) in AGN has recently gained new relevance due to the role of AGN Feedback in galaxy evolution (Brienza et al., 2018), as jets can have a major impact on the interstellar and intergalactic medium. Future radio surveys should help to reveal a more detailed picture of the duty cycle.

Radio galaxies appear to follow a life cycle, where they pass through different evolutionary states, ending when the nuclear activity drops or ceases and the jet fuel supply is exhausted. At this stage they are referred to as remnant radio galaxies. Understanding the evolution of remnants is important to see whether or not they can have a long-term effect on the cluster (Basson & Alexander, 2003). There are relatively few continuum observations of remnants (Giovannini et al. (1988), Brienza (2018) and references within), therefore limiting the possibility of studying them in a statistical sense (Mahatma et al., 2018). Future surveys will detect a larger sample of remnant radio galaxies, enabling better quality population studies and to help develop more accurate models that describe how the radio galaxies evolve after the jets switch off. Larger samples will also reveal higher numbers of unusual remnant sources such as those displaying a steep spectrum at low radio frequencies, which is important in understanding their rarity and the role they play in feedback processes (Brüggen et al., 2018).

Restarted radio galaxies, which display a 'double-double' radio morphology composed of remnant lobes associated with active jets, are one of the most indicative sources of episodic activity in radio galaxies and provide a way to investigate their duty cycle. It is important to identify more examples of such sources to understand the episodic activity of the jets and constrain the time scales over which it occurs, and for studying the propagation of jets in different media (Saikia & Jamrozy, 2009).

Giant Radio Galaxies (GRGs) are defined as those having sizes greater than or equal to a Mpc (Willis et al., 1974; Ishwara-Chandra & Saikia, 1999). Despite the discovery of hundreds of thousands of radio galaxies to date, only a few hundred GRGs have been found. The mechanism that explains their enormous size is still unknown (Dabhade et al., 2019). Although several hypotheses have been proposed, larger samples of GRGs are required to provide further evidence for any hypothesis, which will be available from future surveys. Analyses of smaller samples have led to contradictory results. For example Subrahmanyan et al. (1996) find that GRGs are very old radio galaxies and have had sufficient time to expand over large distances, whereas Mack et al. (1998) found evidence that the ages of GRGs in their sample are similar to that of normal sized radio galaxies.

Other open questions are the connection between the accretion disk and the jet, and the origin and launching mechanism of the ionised winds in AGN (Mehdipour & Costantini, 2019). Hopefully, upcoming radio surveys will shed some light on the mechanisms driving these phenomena.

Future surveys will help to understand the relationship between AGN and star formation evolution and their role in AGN Feedback. At high redshift, the dominant mode of accretion onto a SMBH is cold mode accretion, whereas the dominant mode at low redshift is hot mode accretion (Kereš et al., 2005). These differing mechanisms cause a change in the space density and luminosity functions of radio sources (Norris, 2017a). AGN Feedback may explain the close correlation between the high evolution rate of cosmic star formation and the peak of AGN activity at redshifts between $z = 1 - 2$ (Silk, 2011).

Many more millions of AGN sources will be observed in future surveys and they will make up an important part of multiwavelength studies of galaxy evolution (Padovani et al., 2014).

Clusters of galaxies

Galaxy clusters are the largest organised structures in the Universe that appear bounded by gravity. They are made up of galaxies in a cloud of intergalactic gas, located at the intersections of filaments and sheets of the cosmic web. Synchrotron radiation in the intracluster magnetic field causes the diffuse radio emission observed in galaxy clusters (Feretti et al. 2012). Depending on its morphology, location, and size, a galaxy cluster can be classified as a diffuse elongated object (radio relic), a large diffuse halo, or a mini halo (Brunetti & Jones, 2014; van Weeren et al., 2019). Galaxy clusters should contain very large populations of relativistic electrons and ions as a result of particle acceleration in shocks (Sarazin, 2002; Brüggen et al., 2012; Kang, 2018).

Diffuse synchrotron radiation from the ICM is detected in a variety of radio observations (e.g. Brown & Rudnick (2011)), providing evidence for non-thermal particles, which introduced

fundamental questions about their origins and impact on the physics of the ICM and evolution of galaxy clusters (Kaastra et al., 2008). Upcoming radio telescopes will explore new parameter space and reach unrivaled sensitivities to cluster scale emission over a wide frequency range, enabling more detailed studies of non-thermal components in galaxy clusters. These telescopes will also probe cluster scale magnetic fields, through polarization and Faraday Rotation studies of background and cluster radio sources with unprecedented statistics, frequency and dynamic range (Brunetti & Jones, 2014).

Narrow-Angle Tail (NAT) and Wide-Angle Tail (WAT) galaxies, which are a subset of extended (radio-loud AGN) structures, are tracers of weather in the ICM (Clarke et al., 2014). Therefore, radio surveys can be used to investigate the AGN population in galaxy clusters. Observing the radio emission originating from dominant galaxies in galaxy clusters enables study of the feedback between the ICM and AGN (Gitti et al., 2012; Blanton et al., 2010).

The radio data currently available contains only a handful of galaxy clusters, which may be a biased sample as they were initially discovered at X-ray wavelengths. Large samples of clusters are important in the study of cosmology and large-scale structure formation; and to this effect, upcoming radio surveys will discover hundreds more clusters (Norris, 2017a).

Dark matter annihilation can result in the production of stable Standard Model particles that lose energy through synchrotron radiation due to the presence of magnetic fields, which can be observed through radio emission. Galaxy clusters are excellent targets to search for or constrain the rate of dark matter annihilation, due to their large structure and dark matter composition (Storm et al., 2013). The results of future radio surveys should achieve better constraints on the rate of dark matter annihilation.

Future surveys may help to address the open questions surrounding the origin of cosmic magnetism, such as when and how the first fields were generated, whether present-day magnetic fields are a result of dynamo action or whether they represent persistent primordial magnetism, and the role that magnetic fields play in turbulence, cosmic ray acceleration and galaxy formation (Gaensler et al., 2004). However, the particle acceleration mechanism in galaxy clusters is uncertain, which surveys in the future are hoped to reveal. Several potential mechanisms are reviewed in Petrosian & Bykov (2008).

Cosmology and cosmological parameters

A couple of the first successes of radio surveys with respect to cosmology are ruling out the steady-state theory of the Universe (Shakeshaft et al., 1955; Schmidt, 1963) and the measurement of the cosmic dipole (Blake & Wall, 2002).

Radio continuum surveys in the future are expected to reveal more about cosmology, such

as the source autocorrelation function, the cross-correlation of radio sources with foreground objects resulting from cosmic magnification, and a joint analysis involving the CMB power spectrum and supernovae. The cross-correlation can be used to test and constrain cosmological issues, such as the evolution and clustering of structures, models of dark energy and alternative models for the gravitational potential (see Raccanelli et al. (2012) and references therein). Future radio surveys are expected to produce the most significant measurement of the integrated Sachs-Wolfe effect (the cross-correlation between radio sources and cosmic microwave background (CMB) maps), as well as bring complementary measurements to other experiments (Raccanelli et al., 2012).

Cosmological reionisation is key in understanding the early phases of structure formation and evolution. Radio surveys, together with CMB surveys, will offer unique and specific ways of studying the evolution and energetics of the reionisation process simultaneously, which will provide both global and cross-sectional structure information (Trombetti & Burigana, 2018). The WEAVE/LOFAR survey is expected to find tens of radio galaxies at $z > 6$ and together with using the 21cm forest, it will enable examination of the IGM structure during the Epoch of Reionisation (Simpson, 2017).

A long-standing problem of cosmology is that of the missing baryons, where there are $< 60\%$ of confirmed baryons compared to the total amount predicted (see Nicastro (2016) and references therein). The missing baryons are expected to be located in the hot and tenuous filamentary gas connecting galaxies, also known as the warm-hot intergalactic medium (WHIM). To date, they have been observed in very few absorption lines, for example there have been X-ray observations of filamentary structures of gas at 10^7 K associated with the galaxy cluster Abell 2744 (Eckert et al., 2015), and Nicastro et al. (2018) used observations of two highly ionized oxygen (OVII) intervening absorbers in the exceptionally high signal to noise X-ray spectrum of a quasar at $z > 0.4$. It should be possible to use upcoming radio polarisation observations such as that of the SKA, whose increased sensitivity will enable improved detections of the cosmic web and therefore increase our knowledge about its constituents. As such, Locatelli et al. (2018) use numerical simulations of galaxy clusters to address the possible detection of the terminal part of the magnetised cosmic web, by focusing on observations of intracluster filaments connected to massive galaxy clusters via the Faraday Rotation effect.

Unexpected discoveries

When telescopes enter previously unchartered areas of observational space, they make unexpected discoveries, which often outshine the original goals for which they were built (Norris, 2017b). In fact, out of 18 major astronomical discoveries in the past 60 years, only 7 were planned (Ekers, 2010). One example is pulsars, which were discovered when the radio sky

was studied for the first time with high time resolution, to study interstellar scintillation. The use of high time resolution represented new paramater space at the time. Intimate knowledge of the instrument enabled recognising that the observed noise was not due to terrestrial interference (Bell Burnell, 1969).

In light of future surveys, unexpected discoveries require the development of algorithms that can mine the data for the unexpected. One such example is searching for unusual spectra to enable 'weird' galaxies to be identified in Sloan Digital Sky Survey (SDSS) data (Baron & Poznanski, 2016).

1.2.3 Past and present (large-scale) surveys

We describe a selection of large-scale radio surveys performed over time, which achieved major results in radio astronomy.

2C and 3C surveys

The first large-scale astronomical survey performed was the 2C Survey (Shakeshaft et al., 1955), the first of 10 done by the Cambridge Surveys. The survey detected O(1000) sources at a frequency of 81.5 MHz. Despite the detection of spurious sources in the 2C survey, which were subsequently corrected for, it was the first survey to give evidence for the Big Bang model of the Universe as opposed to the steady state theory.

The subsequent survey (3C) published in 1959, was performed at 159 MHz. The revised 3C survey used observations at 178 MHz, which listed the brighter radio sources in the Northern Hemisphere, detecting several hundred sources in total. The updated version included more recent information about the sources already detected by the original 2C survey, as well as newer sources (Bennett & Simth, 1962). The 3C surveys were also the first without major errors that showed evidence of the strong evolution of radio sources in favour of the Big Bang model.

FIRST/NVSS

Approximately 4 decades after the 3C survey, there was a monumental increase in the number of sources detected, from the order of tens of thousands up to a few million. Two examples of such surveys were FIRST and NVSS, both operating in parallel, using the Very Large Array (VLA).

The NRAO VLA Sky Survey (NVSS) covered the northern sky and was conducted at a frequency of 1.4 GHz and resulted in the catalogue of $\sim$ 1.8 million sources (Condon et al.,

1998). Although it is the largest radio survey to date, it is relatively shallow.

A survey performed in parallel with the NVSS survey is the VLA Faint Images of the Radio Sky at Twenty-cm (FIRST) survey (Becker et al., 1995). FIRST covers a smaller area of the Northern sky and contains over 800,000 unique sources.

The FIRST survey has a higher angular resolution compared to NVSS, as well as a greater depth. The resolution of the survey was chosen to enable identification of optical counterparts to the radio sources as well as their radio morphology. Additionally it should enable the multiple components of a radio source to be resolved (Becker et al., 1994).

In light of current surveys, one drawback of the FIRST survey was the relatively poor angular resolution and the low sensitivity to extended low surface brightness structures, which limited the ability to determine both source sizes and peak locations, particularly for smaller sources (Miraghaei & Best, 2017).

LOFAR

Whereas the FIRST/NVSS radio surveys were conducted at GHz frequencies, LOFAR (van Haarlem, M. P. et al., 2013) is conducted in the MHz regime. Due to the compact core and long baselines of LOFAR, the images provide excellent sensitivity to both highly extended and compact emission (W. Shimwell et al., 2016).

The LOFAR Two-metre Sky Survey (LoTSS; Shimwell, T. W. et al. (2019)) continuum survey consists of three tiers. The survey has detected more than 325,000 sources to date, achieving a signal of 5 times the noise. The source density is a factor of approximately 10 times higher than the most sensitive existing very wide-area radio-continuum surveys.

VLASS

The VLA Sky Survey (VLASS[1]) is currently underway, with preliminary results available. The survey is performed using the Jansky Very Large Array (VLA) at a frequency of 2-4 GHz. Over the last few years, half of the sky north of -40 degrees declination was observed. By completion, it is expected to detect approximately 5 million sources.

[1] https://science.nrao.edu/science/surveys/vlass

Future Surveys

A selection of upcoming surveys include the SKA[2], Evolutionary Map of the Universe (EMU) (Norris et al., 2011) and MeerKAT[3].

The SKA will survey part of the Southern hemisphere in great detail, as it is expected that the angular resolution and survey speed will exceed those of current surveys by thousands of times. It will make extremely deep observations over small areas of sky (Norris, 2017a), where the sensitivity is expected to be two orders of magnitude greater than that of current surveys. The SKA will be divided into two phases, with the first one planned to be completed around 2023, with the second one being a decade later.

EMU, the continuum survey of ASKAP, is considered to be an 'all sky' survey and expected to find around 70 million radio sources including most of the SFGs in the Universe and the first black holes (Norris et al., 2011). It will have O(10) more sensitivity and 5 times the angular resolution compared to NVSS (Norris & the Emu team, 2009). Although the reliability of identifying the optical/infrared counterparts to radio sources does not suffer significantly at the resolution used, the depth of the complementary data has a major effect, which will be the main limitation in trying to extract the maximum from all-sky surveys such as EMU (Simpson, 2017).

MeerKAT is the South African SKA pathfinder telescope, with 64 antennas spanning an area 8 km in diameter. It is a precursor to the SKA. Continuum surveys with MeerKAT will probably include the MIGHTEE survey (Jarvis et al., 2016) which will be conducted at about 1.4 GHz and detect about 200,000 sources.

Cross-matching surveys

One of the main advantages of modern large radio surveys is the ability to perform the essential task of cross-matching with surveys at other wavelengths in order to identify the multiwavelength counterparts of radio sources, to enable statistical studies of their source populations and host galaxy properties (Williams, W. L. et al., 2019). The majority of radio sources are compact, which allows for relative ease of cross-matching with the optical counterpart using automated statistical methods, such as the Likelihood Ratio test (Sutherland & Saunders, 1992).

However, there remains a smaller fraction of radio sources usually belonging to the radio-loud AGN class, that can be difficult to classify. In particular, some of these radio-loud AGN display large and complicated source structures. One example that illustrates obstacles to the

[2]https://www.skatelescope.org
[3]http://public.ska.ac.za/meerkat/meerkat-large-survey-projects

cross-identification of radio sources is when there are two unresolved nearby radio components, where the optical host lies halfway in between. The two components may be the lobes of an FRII, or they may be two independent sources (Norris, 2017a). One way to resolve such a case would be to visually inspect the radio emission pattern and classify it accordingly.

For surveys detecting relatively few O(1000) sources, visual inspection is commonly used to perform the cross-identifications of such sources. This is where citizen scientists have been proven to be useful, where non-expert astronomers are trained to classify objects based on a few simple rules, such as in the Radio Galaxy Zoo (Banfield et al., 2015). However, there is an increased need to automate the cross-matching procedure as many surveys are detecting on the order of many thousands, up to several million sources. The classification of citizen scientists can be used to train a machine learning algorithm. In particular, a convolutional neural network has been used on Radio Galaxy Zoo and was found to produce comparable results to the cross-identifications performed by experts. However, the performance was limited by sample size (Alger et al., 2018).

Another major difficulty in performing cross-matching is being able to distinguish between radio-loud AGN and SFGs. One way to address this is to use the FIR correlation in SFGs.

Summary of radio surveys

The current section has included a description of the parameters on which surveys depend and how they affect the properties of the sources detected. A small selection of large surveys that represented significant milestones in radio astronomy have been reviewed. These surveys have included providing increasing evidence for the big bang model of the Universe and detecting larger populations of faint radio sources including at higher redshifts, which has given a clearer picture about the composition of the Universe and how it has evolved.

In time, radio surveys have achieved substantial increases in the number of sources detected, as a result of significant improvements in sensitivity, resolution and/or area. Figure 1.8 shows the logarithm of the number of sources versus the date a particular survey was first published, since the near-beginning of radio astronomy up to the year 2020. The details of a few future surveys, as well as what is expected to be found from them, has also been discussed.

The ability to cross-match the results of surveys across different wavelengths enables more accurate studies of individual source properties and populations.

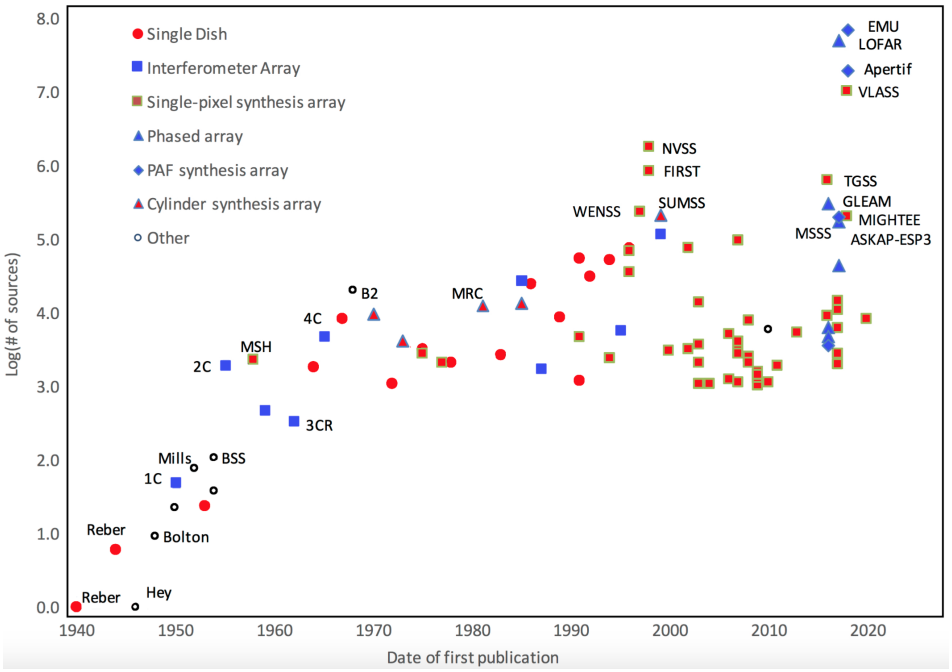

Figure 1.8: Showing the log of the number of sources detected/expected to be detected vs year, from 1940 up to 2020 across the large-scale radio surveys. There is an exponential growth in the number of sources versus time, since radio astronomy began. Figure taken from (Norris, 2017a).

1.3 Machine learning and applications to astronomy

Machine learning involves the use of algorithms and statistical models which aim to learn, infer structure and recognise patterns in data. It belongs under the category of Artificial Intelligence (AI), which can be broadly defined as the intelligence demonstrated by machines, where they have the capacity to perform functions associated with the human mind, such as learning or problem solving (Russell & Norvig, 2009).

A major goal of machine learning is to construct a model, or hypothesis, that can make predictions based on the data provided. The learning process usually takes place on a sample set of (training) data, and after the models are trained they are applied to a sample of test data to evaluate their performance. In some cases, the progress of learning can be evaluated on a separate part of the data called a validation set, which in some circumstances can also be used as the test data.

Machine learning is different to data mining, although both stem from the field of data science. Data mining aims to use intelligent methods to extract information from a dataset and transform it into a usable form for future purposes (Hastie et al., 2009), whereas machine learning aims to learn from data.

There are many applications of machine learning, especially as the development of technology yields increasing amounts of rich and complex data, requiring sophisticated algorithms to extract information from and analyse. It is used in most industries that work with data, including across all areas of science, for example in genomics, where it can be used to identify patterns in genetic sequences e.g. Libbrecht (2015). Machine learning is heavily utilised in social media platforms such as Facebook and Youtube, where the algorithms can for example predict what advertisements to show based on the items that people click and their reactions to them. Another example of its use is in financial services, for example in detecting fraud (Ngai et al., 2011). Machine learning is becoming increasingly important in astronomical applications, as a result of continuously developing telescope and survey technologies (Ball & Brunner, 2010). The application of machine learning to astronomy is the main focus of the current section.

The main categories that machine learning techniques fall into are Supervised and Unsupervised learning. In Supervised learning, labels (outputs) are provided with the training data (inputs). Unsupervised learning does not involve the use of labels, so the algorithm must infer structure using the input data only. A third smaller category, semi-supervised learning, is when labels are provided for some of the inputs.

1.3.1 Supervised learning and common techniques

Examples of tasks involved in supervised learning are classification and regression. In classification, discrete labels are given and the algorithms learn to separate the data into at least two distinct classes, based on their features. A common example is detecting spam in emails, where one can use a set of emails labeled as being 'spam' or 'not spam' as training data for a machine learning algorithm to learn the features that differentiate between the two classes.

In regression problems, continuous values such as measurements of properties are provided, which the algorithm learns from and predicts continuous values for unseen samples. An example is the prediction of house prices, based on properties such as location, size and number of rooms.

Several common supervised machine learning methods that have also had numerous astronomical applications are decision trees (Belson, 1959), random forests (Breiman, 2001) and support vector machines (SVMs; Boser et al. (1992)). Comprehensive reviews include Kotsiantis (2013) for decision trees, Parmar et al. (2019) for random forests and Nalepa & Kawulok (2019) for SVMs. These methods are not described in further detail here since they are not utilised in the current thesis, however we discuss their use in astronomy in a later section.

The following two subsections describe the methods that form the basis of this thesis, namely deep learning techniques and the neural networks from which they stem.

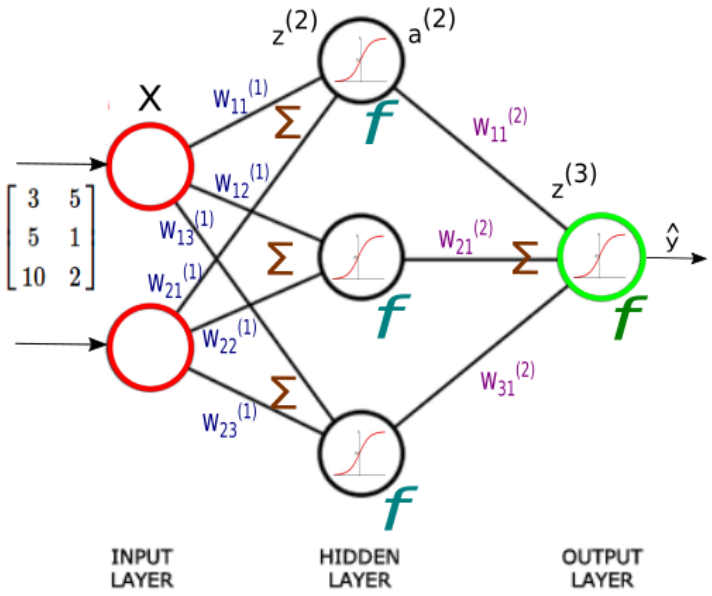

Figure 1.9: Showing the basic architecture of a neural network with an input layer, hidden layer and output layer. Figure taken from Hong (2019).

Artificial Neural networks

Artificial Neural networks (ANNs), in the context of machine learning or artificial intelligence, are inspired by the neural networks in the brains of animals (McCulloch & Pitts, 1943). They can be used in a supervised or unsupervised setting. The terms that underlie the functioning of neural networks are in boldface.

Neural networks consist of an input layer, an output layer and at least one intermediate (hidden) layer. The layers consist of nodes, which in traditional neural networks are completely interconnected between adjacent layers. A **weight**, and generally a **bias**, governs the connections between nodes. The output of nodes is mediated through the use of an **activation function**, which produces an output signal based on some usually nonlinear combination of weights, inputs and biases. The weights are the strength of the connections between the nodes in subsequent layers, usually initialised from a random distribution. A simple neural network architecture is shown in Figure 1.9, containing an input layer having input matrix X, a hidden layer and an output layer, which outputs predictions $\hat{y}$. The weights, which connect the nodes between consecutive layers are shown with subscripts. The inputs to a particular node are a sum (referred to as Σ) of the inputs from the previous layer multiplied by the weights. The activation function of a particular node is denoted using f. The activity of the third layer $z^{(3)}$ is expressed as the activity of the second layer $a^{(2)}$ multiplied by the weights in the second layer $W^{(2)}$.

The data, and corresponding labels in the case of supervised learning, are fed into the input layer and propagated through the network, where a prediction is given at the output layer. The difference between the prediction and true label is computed, which constitutes the error. This error is sent back through the network, and the weights and biases are adjusted in order to minimise the error, using the **gradient descent** algorithm (Cauchy, 1847), discussed in more detail in a later subsection. The process of sending the data back through the network and adjusting the weights to minimise the cost function is called **backpropagation** (Rumelhart et al., 1986). The procedure is repeated for a certain number of samples at a time, as specified by the **batch size** until the entire training set is cycled through, after which the weights are applied to the validation set. The network is said to be trained for a certain number of **epochs**, referring to the number of times the entire training set is cycled through. Ideally, the network should keep being trained until the training and validation losses both reach a (global) minimum. If the training loss continues to decrease and the validation losses start to increase, this signals that the network has begun to **overfit**, which can be remedied with the use of **early stopping** (Caruana et al., 2000), where one specifies a number of epochs to wait and see whether or not the validation loss will decrease once more. If not, training is stopped.

The more complex the data is, the larger the neural networks need to be (in terms of number of layers and nodes per layer), in order to capture the patterns better. At some point the network will reach a level of complexity such that the signals propagated through the layers continue to decrease and can eventually approach zero. This is known as the **vanishing gradient problem** (Hochreiter, 1998). It is remedied by reducing the number of parameters in the network, through techniques such as deep learning, discussed in the following subsection.

Deep learning

Deep neural networks (DNNs) are neural networks containing many layers. Traditionally, all the nodes in neural networks are interconnected between the subsequent layers. When the data becomes high-dimensional it can lead to the vanishing gradient problem, as discussed in the previous subsection. The problem can be remedied with the use of convolutional neural networks (CNNs), which use convolutional layers containing a number of user-defined **filters**, that are smaller 2D windows of a certain size as specified by the user, that scan across the image and detect features. The use of filters greatly reduces the number of parameters in the network, while simultaneously enforcing translational invariance. The number of pixels by which the filters are moved across the image are also specified by the **stride**. The number of parameters can also be reduced by using pooling layers, which summarise the pixel intensities over a small region of the image, as specified by the user. However, pooling results in the loss of some information.

An example architecture of a CNN as shown in Figure 1.10 has stacked layers of convolutional and pooling layers, which achieve a hierarchical extraction of features, followed by up to a few dense layers at the end, which serve to tie together the global features in the image, before outputting the classification, or regression scores, at the final layer.

CNNs are very useful in high-dimensional problems such as image classification. One advantage is that the original data can be input into the network, which performs the feature extraction, rather than using features extracted from another method. The data usually has some form of pre-processing applied to ensure that it is all on the same scale, for example.

A powerful method to boost the performance of CNNs is to artificially increase the training set size using **image augmentation** (Krizhevsky et al., 2012), where many more training samples can be generated through the use of label-preserving affine transformations, such as translations, rotations, flipping, adding random noise and whitening.

One perceived drawback of using CNNs is that despite being translationally invariant, they are not rotationally invariant, as the pooling operation causes the loss of relative feature information within the image. The local features are preserved, however their relationship on a global scale tends to degrade. This problem motivated the introduction of Capsule networks (Sabour et al., 2017), which consist of groups of neurons that aim to preserve all relative feature locations in the data.

There have been many CNN architectures that have been developed for certain applications, with some examples as follows. Recurrent CNNs incorporate recurrent connections into each convolutional layer (Liang & Hu, 2015), ResNets utilise skip connections between convolutional layers if the classification accuracy does not improve with the addition of layers (He et al., 2015b), Generative Adversarial Networks (GANs) are made up of an image generator and discriminator, which has the function of differentiating between real and generated images (Isola et al., 2016), Conditional Adversarial Networks are based on GANs, which condition the generative model on additional information and have been investigated as practical solutions for image-to-image translation problems. Regional proposal networks extract regions of interest within images prior to computing the CNN features (Girshick et al., 2013), and localise objects.

Instead of building a new neural network for every specific task, it is possible to initialise a network using the weights from networks that have been trained on large image databases such as ImageNet (Russakovsky et al., 2014), a method known as **transfer learning** (Pratt et al., 1991). Transfer learning is popular given the computing power and time required to train a deep learning model from scratch, and usually increases the rate of error convergence and achieves improved classification metrics. The application of a transfer learning model must take into account that the earlier layers will detect common features between different

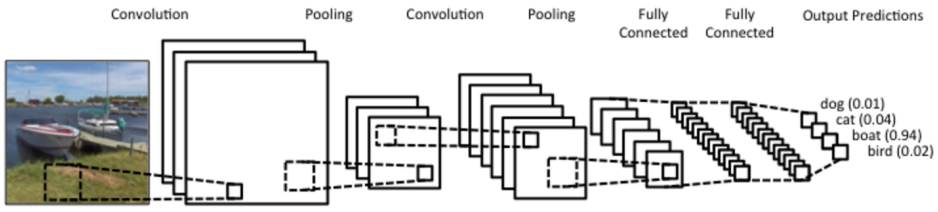

Figure 1.10: An example of a convolutional network architecture. Convolutional layers are generally stacked with pooling layers in between to achieve a hierarchical extraction of features. Figure taken from (Britz, 2019).

datasets, whereas the last few layers are problem specific (Yosinski et al., 2014). As such, it has traditionally been used between datasets that are similar in nature (Tan et al., 2018). Transfer learning can boost the classification performance for small training sets (Quattoni et al., 2008).

A criticism for machine learning techniques, in particular DNNs is that they are considered to be a 'black box' approach in that one is uncertain how the network came to a particular decision given the data provided. As the use of machine learning becomes more integral in real-world data problems, it is becoming increasingly important to attempt to unravel the inner workings of the network. For example, not understanding the mechanism of the algorithm behind self-driving cars can have fatal consequences.

Several approaches have been developed to address these issues, which work on the two different levels based on interpretability and explanation of a deep learning model (Montavon et al., 2017). Direct interpretability models incorporate interpretability into the structure of the model. Post-hoc interpretability models (which seek to understand what the model predicts based on the input) include those based on activation maximisation, which searches for an input pattern that produces a maximum response for a quantity of interest and extends to the use of GANs. Techniques to explain the decisions made by a machine learning model include those based on backpropagation, such as deconvolution (a visualization technique that gives insight into the function of intermediate feature layers and the operation of the classifier) (Zeiler & Fergus, 2013) and layer-wise propagation (LRP) (Bach et al., 2015). LRP allows one to visualise the contributions of single pixels to predictions.

Another criticism specific to DNNs has been regarding their generalisation ability. In some cases, DNNs can misclassify samples when small pertubations are applied to the inputs (Szegedy et al., 2013; Goodfellow et al., 2014). Zahavy et al. (2016) find the use of a technique called ensemble robustness, which centers around the robustness of a collection of hypotheses, improves their ability to generalise.

1.3.2 Unsupervised learning and common techniques

Unsupervised learning techniques seek to infer structure by forming groups from the data, when labels are not provided. They can be used to reduce the dimensionality of the input data and extract the discriminating features between the groups.

Common unsupervised learning techniques include Principal Component Analysis (PCA) (Pearson, 1901), hierarchical clustering (Rokach & Maimon, 2005), k-means clustering (Lloyd, 1982) and self-organising maps (Kohonen, 1989). Neural network architectures can also be used in the context of unsupervised learning, with autoencoders being one example, which are discussed next in further detail as they are used in the current thesis.

For more details on the afore-mentioned methods, useful reviews include Jolliffe & Cadima (2016) for PCA, Murtagh & Contreras (2012) for hierarchical clustering, Ankita Dubey (2017) for k-means and Miljkovic (2017) for SOMs.

Autoencoders

Autoencoders are neural networks that are made up of an encoder, whose purpose is to compress the input into a hidden representation having a lower dimensionality (bottleneck) (Tishby et al., 1999), and a decoder which aims to reconstruct the input using the hidden representation. In mathematical terms, if the encoder is expressed using function $h = f(x)$ and the decoder is $r = g(h)$, the autoencoder function is $r = g(f(x))$. The autoencoder aims to use the hidden representation to achieve as close a reconstruction r as possible, to the original data x (Vincent et al., 2008). When the most accurate possible reconstruction is achieved, the loss (error), which is some function of the difference between the reconstruction and the input, will be minimised. It is common to use the mean squared error (MSE) or the sum of squares error (SSE).

Autoencoders were used for the first time to perform unsupervised pre-training of neural networks (Ballard, 1987), to initialise the weights in a neural network, a purpose for which they continue to be used, rather than initialising from a random distribution. Other common applications of autoencoders are non-linear dimensionality reduction, denoising and visualisation. A simple autoencoder architecture is shown in 1.11. To help avoid the vanishing gradient problem for high complexity datasets, one can construct an autoencoder using a combination of convolutional layers and dense layers, instead of using only dense layers.

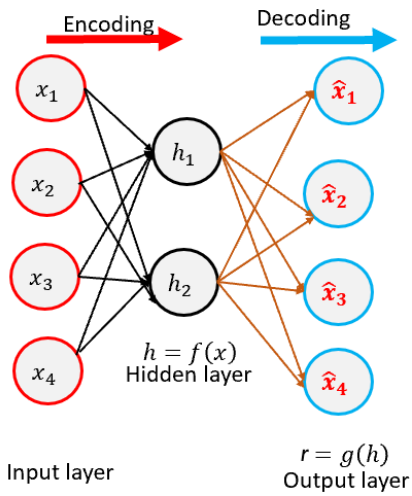

Figure 1.11: A simple autoencoder structure having an input layer, one hidden layer and an output layer. Figure taken from Khandelwal (2018).

1.3.3 Machine learning theory

Machine learning techniques have several aspects in common. As mentioned previously, the aim is to construct a model that can make predictions based on the data provided, or map inputs to outputs. The success of many machine learning algorithms largely depends on the training set size, and the choice of a particular algorithm depends on the task at hand, the data available and its complexity. From hereon in, the underlying concepts concerning many machine learning techniques are highlighted in boldface.

Prior to the application of any machine learning technique, it is necessary to **pre-process** the data to convert it into a form amenable for analysis. A common method is to use normalisation, to ensure all the data is on the same scale. The type or sequence of steps involved in pre-processing usually depends on the data at hand and the problem to be solved (Ball & Brunner, 2010).

In constructing a machine learning model, we can assume that in the simplest case, our prediction $\hat{y}$ has a linear dependence on the input data x, and that our data is continuous, which can be expressed as the following:

$$\hat{y} = \beta_0 + \beta_1 x \tag{1.4}$$

where $\hat{y}$ is the prediction vector, β_0 and β_1 represent the coefficients of the estimates, and x is the input data vector.

One important quantity is the difference between the true value y (given data x) and the predicted value $\hat{y}$, which forms the basis of any **cost function** (Wald, 1949), where the error

can be expressed as $\hat{y} - y$. The cost function is some function of the difference between $\hat{y}$ and y, and there are many available for use. A common cost function used in machine learning is the mean squared error (MSE), which is the average of the sum of differences squared:

$$MSE = \frac{1}{2m} \sum_{i=1}^{m} (\hat{y}^{(i)} - y^{(i)})^2, \tag{1.5}$$

where m is the number of samples in the dataset, i is a particular sample in the dataset, $\hat{y}^{(i)}$ is the prediction for sample i, and $y^{(i)}$ is the true value for sample i.

Other common cost functions are the mean absolute error, cross-entropy and log-loss. The choice of method depends on the problem at hand and the type of data available. One usually wants to minimise the cost function such that the errors approach 0. Consequently, the prediction $\hat{y}$ will be as close to the true y.

Substituting $\hat{y}^{(i)} = \beta_0 + \beta_1 x^{(i)}$ and denoting the cost function, with parameters β_0 and β_1 as $J(\beta_0, \beta_1)$:

$$J(\beta_0, \beta_1) = \frac{1}{2m} \sum_{i=1}^{m} ((\beta_0 + \beta_1 x^{(i)}) - y^{(i)})^2 \tag{1.6}$$

To find the minimum of the cost function we can differentiate with respect to parameters β_0 and β_1 and set the cost function to 0. This method is known as **gradient descent**. It is also possible to see how much the cost function changes with small modifications to each parameter $\frac{\partial J(\beta_0,\beta_1)}{\partial \beta_0}$ and $\frac{\partial J(\beta_0,\beta_1)}{\partial \beta_1}$.

Seeing what modifications to β_0 and β_1 will result in a reduction in cost function $J(\beta_0, \beta_1)$, the updated parameters β_0^* and β_1^* are calculated as such:

$$\beta_0^* = \beta_0 - \alpha \frac{\partial J(\beta_0, \beta_1)}{\partial \beta_0} \tag{1.7}$$

and

$$\beta_1^* = \beta_1 - \alpha \frac{\partial J(\beta_0, \beta_1)}{\partial \beta_1} \tag{1.8}$$

where α is the **learning rate**, which is important to tune correctly as choosing values that are too small make the cost function take a longer time to converge to a minimum, whereas values that are too large cause erratic changes to the cost function and may cause it to overshoot the minimum.

Depending on the number of parameters, the cost function can be thought of as being in a high-dimensional space consisting of many minima, such as shown in Figure 1.12. It is

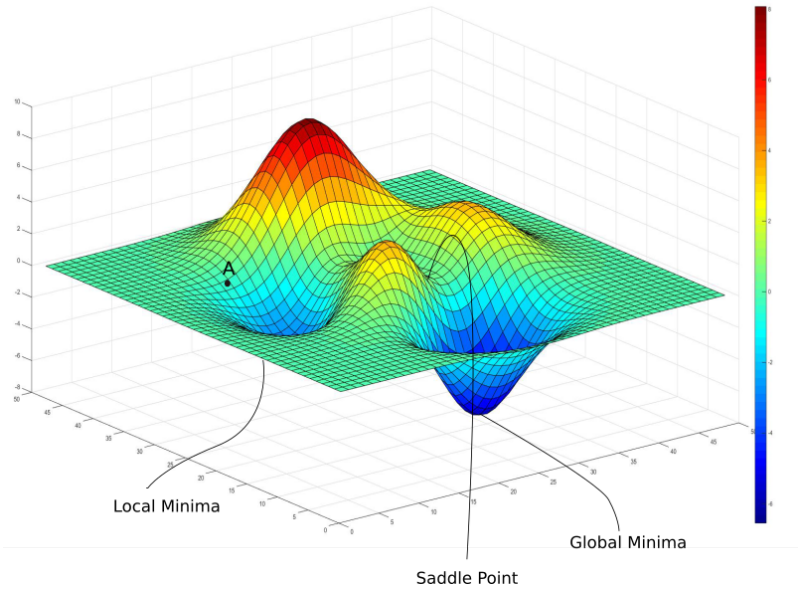

Figure 1.12: The cost function can be viewed as a higher-dimensional space that depends on the number of parameters. Figure taken from (Daniel, 2019)

possible for the cost function to converge on a local minimum rather than a global one.

It can be difficult to manually tune the various parameters to find the global minimum to the cost function. One way to make the search easier is to use a grid search (Bergstra & Bengio, 2012).

One fundamental function of machine learning algorithms is the detection and extraction of **features**, which are individual measurable properties or observed characteristics in the data (Deng & Yu, 2014). Choosing discriminative and independent features is important in being able to distinguish between classes of interest. Features can be expertly chosen (extracted), based on ones expert domain knowledge, or this could be done using a machine learning algorithm. However, given that the process reduces the dimensionality of the data, feature extraction inevitably results in some information loss.

The **curse of dimensionality** is a problematic phenomenon that occurs in machine learning as well as other fields, when analysing and organising data in high-dimensional, rather than low-dimensional spaces such as that of the everyday world (Bellman, 2003). To combat this effect, one needs to use a sufficient amount of training data, to ensure that there are several data examples for each dimension in the representation.

Another undesired effect which can result from training a machine learning algorithm is **overfitting**, which is where the model memorises the training samples on which it achieves good results but is not able to achieve equivalent performance on a set of test/validation data; the model fails to generalise to an independent data set (Baum & Haussler, 1989). This

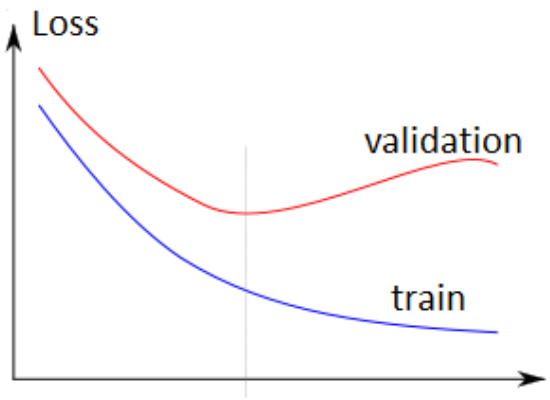

Figure 1.13: Showing an example of overfitting when training a neural network. Training should continue if the training and validation losses both continue to decrease, however if the validation losses start to increase while the training loss is still decreasing, this indicates that overfitting has started to occur. The dashed line represents the ideal time when training should be stopped. Figure taken from Avendi (2018).

generally occurs when there is an excess of parameters compared to the complexity and/or size of the dataset. There are different methods available that can help reduce the effect of overfitting, but some are common only to certain machine learning approaches. For example there are ways to reduce overfitting in neural networks which do not apply to other machine learning methods. A graphical example of overfitting is shown in Figure 1.13.

However, one common way among many machine learning methods to reduce overfitting is to use **regularisation** (Buehlmann & van de Geer, 2011), explained in further detail below.

Considering the learned relation Y, and β represents the coefficients estimates for different predictors X,

$$Y \approx \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_n X_n \approx \beta_0 + \sum_{p=1}^{p=n} \beta_p X_p \tag{1.9}$$

To fit such a model requires the introduction of a loss function, known as a residual sum of squares or RSS, which will minimise the coefficients β.

$$RSS = (Y - (\beta_0 + \sum_{p=1}^{p=n} \beta_p X_p))^2 \tag{1.10}$$

Real life data contains noise, which means Equation 1.10 may not generalise well. To account for this, an extra (regularisation) term is added to the RSS loss function;

$$RSS = (Y - (\beta_0 + \sum_{p=1}^{p=n} \beta_p X_p))^2 + \lambda \sum_{i=1}^{i=n} \beta_i^2, \tag{1.11}$$

where λ represents the shrinkage quantity, which determines the extent to which flexibility in the model is penalised. Making the model more flexible requires the β coefficients to have higher values, however at the same time the cost function should be minimised. Therefore the added term prevents these coefficients from becoming too high. As $\lambda \to 0$, the equation becomes just the simple sum of squares, whereas as $\lambda \to$ inf the coefficients β_i will be very small, therefore the model will not be as flexible. Formula 1.11 is referred to as **weight decay** in machine learning, or ridge regression in statistics (Tikhonov & Arsenin, 1977).

The regularisation method helps to reduce overfitting by ensuring the predictor coefficients do not become too high.

Evaluating performance

There are several methods that can be used to evaluate the performance of machine learning techniques designed for classification. Some popular metrics are precision, recall, F1 score and accuracy. These metrics are some function of at least two of the following: true positives, true negatives, false positives and false negatives.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{1.12}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{1.13}$$

$$\text{F1 score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{1.14}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}, \tag{1.15}$$

where TP refers to the true positives (when the positive class is predicted and matches the label), TN refers to true negatives (when the negative class is predicted and matches the label), FP refers to the false positives (when the positive class is incorrectly predicted) and FN refers to false negatives (when the positive class is predicted to be in another class).

Precision refers to the fraction of true positives returned among all returned positive instances, also commonly called 'reliability'. Recall is the fraction of true positives that are identified correctly, which also gives an indication of the sensitivity of the classifier, also commonly called 'completeness'. The F1 score can be interpreted as the average of the precision and

recall values. The accuracy is the total proportion of correct predictions.

Many classifiers use the classification accuracy only to evaluate their performance. However, its downfalls include less distinctiveness, discriminability, informativeness, and bias towards the class having most data (Hossin & M.N, 2015). A more complete picture of the classifier performance is given by additionally considering the precision, recall and F1 score metrics, as they take the minority class into account.

Another way to evaluate classifier performance is to plot a Receiver Operating Characteristic (ROC) curve, which is the true positive rate versus the false positive rate (Fawcett, 2006). A larger area under the ROC curve indicates better classifier performance. However, evaluating the classifier performance using ROC plots may be problematic when datasets are imbalanced (Saito & Rehmsmeier, 2015). Precision-Recall (PR) curves have been proposed as an alternative to ROC curves for tasks with a large class imbalance (see Davis & Goadrich (2006) and references therein).

To handle the issue of imbalanced datasets, it can be possible to oversample items in the minority class. For example, in the context of image classification, image augmentation can be used to generate supplementary images if there are relatively few in a given class. It is also possible to use a larger weight in the cost function for incorrect classifications of positive instances (Vluymans, 2018).

Confusion matrices can also be used to evaluate the performance of a classifier. It is a 2D matrix of the number of matches between the true label and predicted label across all categories (Stehman, 1997).

1.3.4 Machine learning techniques applied to astronomy

A selection of both supervised and unsupervised machine learning techniques have been applied to astronomical datasets, where a few of the most popular applications include the estimation of photometric redshifts, classifying objects such as stars and galaxies (including morphology), as well as different types of AGN (Du Buisson, 2015). The following subsections describe the application of a particular machine learning technique in the astronomical data setting.

Decision trees

Decision trees have been useful in interpreting astronomical data, owing to the simplicity in the way their results are presented.

In astronomy, decision trees have been applied to the star/galaxy separation problem, with a few examples as follows. Weir et al. (1995) determine a set of highly informative object attributes on which decision tree induction was applied, to infer the rules for distinguishing between objects in the high-dimensional space, using images. Another study (Ball et al., 2006) involves the classification of more than a million photometric objects in SDSS-DR3, where they demonstrate that the star/galaxy classifications performed using decision trees are expected to be reliable for a substantial fraction of objects, over a range of magnitudes. Comparable or improved metrics were observed over the same range of object magnitudes by Vasconcellos et al. (2011), who use a number of different decision tree algorithms applied to hundreds of thousands of photometric objects in SDSS-DR7, and find that the Functional Tree algorithm (which combines a standard univariate decision tree with linear functions of the attributes using linear regressions) works best on the star/galaxy classification problem.

Another application of decision trees has been in selecting quasars for surveys. One such example is White et al. (2000), who obtained the first optically bright radio-selected sample of quasars that was competitive in size with optically selected quasar surveys at the time.

Random forest

Random forests use a collection of decision trees and as a result, tend to be more robust. Several applications of random forests to astronomical data are as follows.

Random forests have been used to identify quasars from the FIRST survey (Breiman et al., 2003). A related application has been in studying the separation of quasars from stars, as well as the classification of quasars, stars and galaxies, using multiwavelength data (Gao et al., 2009).

They have also been used to estimate photometric redshifts, for example Carliles et al. (2008) employ a random forest trained on color features and spectroscopic redshifts from tens of thousands of randomly chosen primary galaxies from SDSS-DR6, to produce a mapping from color to redshift. They achieve redshift estimates with a tight RMS scatter, which is consistent with the results from previous studies using similar datasets. In Carliles et al. (2010), they use random forest regression to provide independent constraints on the redshift of each galaxy, and find that it improves the utility of redshift estimates by giving good measurements of the estimation error.

Several other uses of random forests include the construction of a large, deep catalog of point sources utilizing Pan-STARRS1 (PS1) 3π survey data, where they are found to outperform several alternative models (Tachibana & Miller, 2018). In the context of galaxy mergers, Snyder et al. (2019) presents their image-based evolution from the Illustris cosmological simulation at different time-steps for redshifts between 0.5 and 5. They train redshift-dependent

random forests, yielding improved measurements of the late-stage merger fraction compared to conventional approaches. An outlier detection technique has also been developed using an unsupervised random forest algorithm (Baron & Poznanski, 2016) and found to be successful in being able to detect unusual objects, such as those having extreme emission line ratios, abnormally strong absorption lines, unusual continua, as well as extremely reddened galaxies.

SVMs

Due to their ability to generalise well, excellent performance and ease of adjusting the model parameters (Zhang & Zhao, 2014), SVMs have proven to be useful in numerous astronomical contexts, both in classification and regression tasks.

SVMs have been used to estimate photometric redshifts. For example, Wadadekar (2005) use SVMs on a combination of many thousands of galaxy samples from the SDSS-DR2 and the hybrid galaxy formation code GalICS. Another use is in the identification of potential supernovae using photometric and geometric features obtained from astronomical imagery (Romano et al., 2006). SVMs have also been used to quantify morphologies of seeing-limited galaxies from a simulated sample built from a local visually classified sample from the SDSS, and show that a qualitative separation between late-type and early-type galaxies, which have a different morphology, can be obtained (Huertas-Company, M. et al., 2008).

Across broader astrophysical scales, Hartley et al. (2017) apply SVMs in the context of gravitational lensing, where they are used to provide learning criteria for separation of lenses and non-lenses in simulated data. Systems with small Einstein radii are found to constitute most of the lensing objects in the sky, however they are difficult to detect by eye without very careful subtraction of the potential lensing galaxy, whereas the SVM is able to find these objects successfully. A SVM approach has also been taken to study imprints of environmental effects on the mass assembly of haloes. As such, Hui et al. (2018) discover strong connections between the cosmic environment and the shape and depth of the merger tree, formation time and galaxy density.

ANNs

ANNs have been very useful in astronomical applications as their flexible structure allows them to perform many different tasks such as clustering and dimensionality reduction, in addition to the usual classification and regression (Baron, 2019). Their flexibility also offers greater control over the parameters, which tend to be more fixed in other machine learning methods. Shallow ANNs, having relatively few layers, are generally applied to problems

having dimensions on the order that would also be analysed using lower-level machine learning algorithms such as decision trees and random forests.

Several applications of shallow ANN architectures are as follows. In addition to estimating photometric redshifts with other machine learning methods, neural networks have also been used, for example on SDSS data (Firth et al., 2003; Tagliaferri et al., 2003). The mapping from photometry to spectroscopic redshifts is derived, and the best-fit solution is applied to the photometry data only. However, this requires well-controlled and representative training sets. To this effect, Bilicki, M. et al. (2018) applies two neural-network based methods on survey data which is a precursor to the data from future surveys that will have limited spectroscopic data available. They show that at the bright, low-redshift end, the two neural network methods perform better than the Bayesian Photometric Redshift in most statistics.

Storrie-Lombardi et al. (1992) use an artificial neural network (ANN) to classify galaxies morphologically into several categories, using a set of galaxy parameters. The trained network outperformed another popular method used at the time, using photometric and structural parameters (Lauberts & Valentijn, 1989).

Neural networks have also been used in the star/galaxy separation problem (Andreon et al., 1999) and found to be competitive with traditional methodologies.

Another application has been in the classification of stellar spectra. As such, Weaver & Torres-Dodgen (1997) use ANNs to classify stellar spectra across all temperature and luminosity classes with the same accuracy as human experts.

CNNs

CNNs, belonging under the category of deep learning, are a type of artificial neural network having many layers. CNNs have been used across varying astronomical scales, from stars up to grand cosmological scales, at different wavelengths, on mainly image data.

Applications to stars

The application of CNNs in the stellar regime includes the implementation of a DNN architecture to identify signatures of stellar feedback in simulated molecular clouds (Van Oort et al., 2019). The network is applied to the two tasks of dense regression and segmentation, on simulated density and synthetic ^{12}CO observations, and found to perform well on two segmentation tasks and related regression tasks. A related study is the use of a CNN for the recognition of the predefined magnetic types in sunspot groups, using three different models that take magnetograms, continuum images, and two-channel pictures as input (Fang et al., 2019). They show that the CNN is able to identify the magnetic types in solar active regions.

Recurrent convolutional neural network (RCNN) have been useful for time-varying data. As

such, Carrasco-Davis et al. (2019) propose the use of such a sequential classification model for time-varying astronomical objects such as variable stars and supernovae, using sequences of images as inputs, instead of light curves. They also generate synthetic image sequences, aiming to mimic real data, which they then use to train the RCNN classifier and evaluated the model on real images. The RCNN model showed good classification performance on the simulated test set as well as the real dataset after fine tuning. This work was the first time that a sequential classifier was used in time-domain astronomy.

Applications to optical galaxy classification

CNNs have been used in regard to optical galaxy classification, with a few examples as follows. The use of a CNN architecture was the winning solution to the Kaggle Galaxy Zoo challenge, where the competitors were tasked with implementing a regression technique to predict how citizen scientists would respond to questions about optical galaxy morphologies, such as whether it is round, smooth or had a bulge (Dieleman et al., 2015b). Following this, Domínguez Sánchez et al. (2018) uses CNNs to obtain classifications for a morphological catalogue for galaxies in SDSS based on two classification schemes: T-type based on the Hubble sequence, and the Galaxy Zoo 2 (GZ2) classification scheme. On T-type galaxies, the results show smaller offset and scatter compared to previous models using SVMs. High performance metrics are achieved using the GZ2 classification scheme. Using similar galaxy morphology classifications, Zhu et al. (2019) proposes a variant of residual networks (ResNets) on images from the GZ2 dataset. Various performance metrics show that the proposed network achieves state-of-the-art classification performance among networks such as Dieleman and other ResNets.

Application to merger classification

Pearson et al. (2019) develop a CNN architecture in the context of galaxy mergers, to test whether it can reproduce visual classification of observations and physical classification of simulations, and highlight any differences between the two. Their results overall suggest that most of the simulated mergers do not have obvious merger features, and visually identified merger catalogues from observations are incomplete and biased towards certain types of mergers.

Applications to radio galaxy classification

Applications of CNNs to radio astronomy are the focus of the current thesis. The first work to use CNNs in classifying radio galaxy morphologies was Aniyan & Thorat (2017) where they classify radio galaxies into FRI, FRII and Bent-tailed classes, however despite the use of aggressive augmentation there were problems in regards to overfitting, due to having an insufficient number of original training samples. Following this, in Lukic et al. (2018) we show it is possible to classify RGZ data into three classes of extended objects and

one class of compact objects, applying the trained network on the RGZ DR1 dataset and find it can reproduce most of the citizen scientists classifications. In a similar application, Alhassan et al. (2018) train a CNN to classify radio sources from FIRST, into the classes of Compact, FRI, FRII and Bent-tailed morphologies. A more specialised neural network approach trained to pay attention to relevant regions in radio galaxy images is by Wu et al. (2018), who use a region-based convolutional network (ClaRaN) to recognize, detect and classify radio galaxies, achieving optimal results when using radio contour data overlaid on infrared data, and masking out unassociated emission. In Lukic et al. (2019) we take a different specialised approach using Capsule networks (a type of network designed to preserve the relative location of features) and compare their performance against CNNs, and find that CNNs always outperform the Capsule networks for the given dataset. This may be due to the pooling operation in CNNs offering increased robustness to the noise present in the images, and more freedom for the variations in morphology within the classes. It could also be the case that Capsule networks require a larger original sample size compared to traditional convolutional networks.

Applications across extragalactic scales

Additional uses of CNNs have been across wider astronomical scales. For example, Gheller et al. (2018) develops a CNN called COSMODEEP to detect extended extragalactic radio sources in existing and upcoming surveys, which proves to be accurate and fast in detecting very faint sources in the simulated radio images, and comparable in performance to that of a standard source-finding technique such as PyBDSF.

CNNs have also been applied to gravitational lensing. As such, they have been used to quantify image distortions caused by strong gravitational lensing and estimate the lensing parameters (Hezaveh et al., 2017). In contrast, George & Huerta (2018) uses time-series inputs to a CNN to rapidly detect and characterise gravitational wave signals. A different problem is the consideration of noise in weak gravitational lensing maps. As such, Shirasaki et al. (2019) explores the use of an image-to-image translation technique with conditional adversarial networks (CANs) in the reduction of noise in such maps, where a successful reduction of 30-40% is achieved when using observational data from ongoing and upcoming galaxy imaging surveys.

Ntampaka et al. (2019) estimates galaxy cluster masses from Chandra mock images using a CNN trained and tested on a sample of Chandra X-ray mock observations, based on 329 massive clusters from the IllustrisTNG simulation. The CNN learns from a low resolution spatial distribution of photon counts, instead of spectral information. They find that the CNN has learned to ignore the central regions of clusters, which are known to have high scatter with mass.

Application to noise removal

CNNs have also been investigated in the context of noise removal. For example, Kerrigan et al. (2019) explore the use of deep learning methods in regard to the identification and removal of Radio Frequency Interference (RFI). They apply a Deep Fully Convolutional Neural Network on interferometric data, using both amplitude and phase information simultaneously to identify RFI.

Transfer learning

A very useful approach in training neural networks, that speeds up the rate of convergence and improves metrics, is transfer learning. This method has not been utilised as much in astronomy compared to building and training a neural network from scratch.

In the context of optical galaxy classification, Domínguez Sánchez et al. (2019) tests the performance of deep learning models, trained with SDSS data, on the Dark Energy survey (DES) using images of several thousand galaxies with a similar redshift distribution to SDSS. The use of pre-loaded weights trained on SDSS data and fine-tuning them by training the models with a small DES sample of several hundred galaxies outperforms the results obtained using a direct application of the models to DES data. In a very similar application, Khan et al. (2019) demonstrate that DNNs pre-trained with real-object images can be transferred to classify galaxies that overlap both SDSS and DES, achieving state-of-the-art accuracy. The network is used to label DES galaxies that are not present in previous surveys, and also use the network as a feature extractor for unsupervised clustering, finding that unlabelled DES images can be grouped together in two distinct galaxy morphology classes.

Tang et al. (2019) employ a transfer learning approach in regard to the radio galaxy classification problem. Their machine learning models trained using a random initialisation achieve similar accuracies to that of other radio galaxy classification algorithms. When applying transfer learning, they find that using weights pre-trained on FIRST images can improve the model performance applied to lower resolution NVSS data, however the use of a pre-trained weights from NVSS applied to FIRST data impairs the performance of the classifier.

Ackermann et al. (2018) investigate the use of deep CNNs and transfer learning for automatic visual detection of galaxy mergers, and find them to perform significantly better than current state-of-the-art merger detection methods. The resulting metrics are also slightly better compared to the results obtained by using a normal CNN architecture without transfer learning (Pearson et al., 2019), although the authors note differences between the studies.

Autoencoders

Autoencoders are useful in astronomical applications given their ability to achieve a non-linear dimensionality reduction and form a generalization of linear methods such as principal component analysis (PCA) (Graff et al., 2014).

A neural network training algorithm for astronomy called SKYNET makes use of autoencoders for the purposes of compressing and denoising galaxy images, and unsupervised pretraining (Graff et al., 2014). Lucas et al. (2018) describe the use of an autoencoder in a purely denoising application where it is trained on a selection of noiseless and noisy astronomical images and applied to previously unseen data in order to reconstruct a corrected bispectra. Another denoising application has been in gravitational wave data, where Shen et al. (2017) find their autoencoder model achieves superior recovery performance for gravitational wave signals embedded in real non-Gaussian LIGO noise.

Regier et al. (2015) describe the first use of an autoencoder as a generative model for optical galaxy images. In addition to their capacity to generate images, autoencoders can also be used for feature extraction. For example, Iwasaki et al. (2019) apply an autoencoder architecture to X-ray data from Tycho's supernova remnant. They use a variational autoencoder, which reduces the observed dimensions in the observed spectral data, with a Gaussian mixture model, used to perform clustering in feature space, to the spatio-spectral analysis of the X-ray data. They find that the use of spectral properties only is enough for the method to automatically recognise characteristic spatial structures.

In Lukic et al (2019); submitted to Galaxies, we use a convolutional autoencoder (AutoSource) as a novel approach to source-finding in radio astronomical data at different signal-to-noise ratios, and find the performance competitive to the state of the art Gaussian-fitting technique, PyBDSF.

Principal Components Analysis (PCA)

Given the ability of PCA to perform dimension reduction of relatively high-dimensional data, they are able to extract meaningful information from astronomical datasets.

A popular application of PCA in astronomical tasks has been in spectral classification. Connolly et al. (1995) finds a strong correlation in the mean between the spectral classifications obtained from applying PCA, and those from morphological classifications in the literature. A similar spectral classification application is in Ronen et al. (1999), who identify principal components of variance in the synthetic spectra of galaxies and find that with a high enough signal-to-noise ratio, the age, star formation history and metallicity can be derived. Yip et al.

(2004) perform classification of galaxy spectra from SDSS and find the galaxy populations can be divided into three classes corresponding to early late to intermediate late types.

A couple of applications of PCA have been in characterising aspects of quasars, with a couple of examples as follows. Bailey (2012) uses PCA on quasars and quasi-stellar source spectra from the SDSS, a comparatively noisy dataset containing missing values. The measurement error is estimated and used to weight the input data such that the resulting eigenvectors are more sensitive to the underlying signal variations. Delchambre (2016) determines the redshift of quasars from the twelfth SDSS quasar catalog and derive the proper spectral reduction and redshift selection methods using PCA. The redshift uncertainty and associated confidence is derived and it is found that the results of this application are similar to the performance of the SDSS pipeline.

Self-Organising Maps (SOMs)

SOMs are another dimensional reduction technique that groups similar objects together on a map, which makes them useful in organising astronomical data.

They have been applied to the visualisation, exploration and mining of catalogues in large astronomical surveys, for example COSMOS (Geach, 2012).

Another application of SOMs has been in automatically classifying light curves to identify variable stars. To this effect, Brett et al. (2004) find their maps successfully distinguish between light curve types in both synthetic and real datasets and are robust to the chosen learning parameters. Sarro et al. (2006) presents a refined SOM map for classifying light curves of eclipsing binaries.

SOMS have also been used to characterise AGN types. For example, Torniainen et al. (2008) applies an SOM to gigahertz peaked spectrum sources, and find that the sources form distinctive clusters on the map, indicating the presence of different subpopulations, besides the expected galaxy-quasar dualism. In regard to investigating AGN of a different nature, Faisst et al. (2019) find that SOMs are successfully able to find galaxies with brightness-variable AGN.

Particular applications to radio astronomy include Parallelized rotation/flipping Invariant Kohonen-maps (PINK; Polsterer et al. (2016)) which uses a rotation and flipping invariant similarity measure on self-organizing maps (SOM) to obtain a visual representation of the input data. The method is applied to Radio Galaxy Zoo image data, where the SOM is trained on hundreds of thousands of images. PINK software is also used to produce a multi-channel SOM using images based on RGZ DR1 (Galvin et al., 2019). The resulting SOM exhibits a range of morphologies that are representative prototypes of the training objects

used, and they are able to achieve a visible clustering of labels given by RGZ volunteers, on the surface of the SOM map.

Comparisons/combinations of machine learning techniques

There are many examples in the astronomical literature that use several machine learning techniques and compare the results to see which one performs the best for the dataset in question. Also, it is possible to combine a number of approaches into a single classifier optimised for a particular dataset. Some such examples are outlined below.

Wright et al. (2015) apply artificial neural networks, support vector machines and random forests to the identification of astronomical transients, and find that the random forest classifier outperforms the others.

In regard to galaxy applications, Hocking et al. (2018) presents an unsupervised machine learning technique, composed of a combination of three unsupervised algorithms, to automatically segment and label galaxies in astronomical imaging surveys using only pixel data. The algorithm is able to clearly separate early and late type galaxies by training it on galaxies from one field and applying the result to another field. Liu et al. (2019) show another example applied to galaxies, that instead compares the performance of different machine learning techniques on multiwavelength images. A training set is constructed that consists of a combination of magnitudes and other derived features, and used to determine how to identify submillimetre galaxy counterparts. They find that a DNN performs the best, compared to other machine learning approaches such as SVMs, decision trees, random forests and normal neural networks.

With respect to studying galaxies in the radio, Ralph et al. (2019) use a combination of a self-organising map and convolutional autoencoder to perform unsupervised clustering on radio astronomical images from the Radio Galaxy Zoo. The method is capable of separating outliers accurately on a SOM with neighbourhood similarity, and achieves a K-means clustering with a distinct class of a small number of highly complex sources.

A study that utilises galaxy properties is in Wu & Boada (2019), who train a deep residual CNN to predict the gas-phase metallicity of galaxies derived from spectroscopic information using images from SDSS. The CNN outperforms a trained random forest algorithm. They were able to use predicted metallicity from the CNN and independently measured stellar masses to recover a mass-metallicity relation. Their results suggest that by utilising optical imaging, the CNN has learned a representation of the gas-phase metallicity, which would not be available from using oxygen spectral lines.

2 Compact and Extended Radio Source Classification

The following chapter presents work as it is published by Lukic et al. (2018).

2.1 Introduction

Extragalactic radio sources are among the most unusual and powerful objects in the universe. With sizes sometimes larger than a megaparsec, they have radio luminosities that are typically 100 times those of star-forming galaxies for example (van Velzen, Sjoert et al., 2012), and display a wide range of morphologies. A new generation of wide-field radio interferometers are undertaking efforts to survey the entire radio sky to unprecedented depths making manual classification of sources an impossible task. Among the current and upcoming radio surveys that will detect such high numbers of radio sources are the LOw Frequency ARray (LOFAR[1]) surveys, Evolutionary Map of the Universe, the largest of such surveys in the foreseeable future (Norris et al., 2011), VLA Sky Survey (VLASS[2]) and surveys planned with the Square Kilometre Array (SKA[3]). The SKA alone will discover up to 500 million sources to a sensitivity of 2 μJy/beam rms (Prandoni & Seymour, 2015). Radio interferometry data often display high levels of noise and artefacts (Yatawatta, 2008), which presents additional challenges to any method of obtaining information from the data, such as extracting sources, detecting extended emission or detecting features through deep learning.

Machine learning techniques have been increasingly employed in data-rich areas of science. They have been used in high-energy physics, for example in inferring whether the substructure of an observed jet produced as a result of a high-energy collision is due to a low-mass single particle or due to multiple decay objects (Baldi et al., 2016a). Some examples in astronomy are the detection of 'weird' galaxies using Random Forests on Sloan data (Baron & Poznanski, 2016), Gravity Spy (Zevin et al., 2017) for LIGO detections, optimizing the performance and probability distribution function of photo-z estimation (Sadeh et al., 2016), differentiating

[1] http://www.lofar.org
[2] https://science.nrao.edu/science/surveys/vlass
[3] https://www.skatelescope.org

between real vs fake transients in difference imaging using artificial neural networks, random forests and boosted decision trees (Wright et al., 2015) and using convolutional neural networks in identifying strong lenses in imaging data (Jacobs et al., 2017).

Traditional machine learning approaches require features to be extracted from the data before being input into the classifier. Convolutional neural networks, a more recent machine learning method falling within the realm of deep learning, is able to perform automatic feature extraction. These suffer less information loss compared to the traditional machine learning approaches, and are more suited to high-dimensional datasets(LeCun et al., 2015). These are based on neural networks that contain more than one hidden layer (Nielsen, 2015). Each layer extracts increasingly complex features in the data before performing a classification or regression task. The raw data can be input into the network, therefore minimal to no feature engineering is required (LeCun et al., 1989), and the network learns to extract the features through training. However, it should still be noted that convolutional neural networks do not always capture the data features.

The classification of optical galaxy morphologies is based on a few simple rules that makes it suitable for machine learning. It also lends itself to citizen science, where these rules can be taught to non-experts. The Kaggle Galaxy Zoo (Willett et al., 2013) was a competition where the aim was to predict the probability distribution of the responses of citizen scientists about a galaxy's morphology using optical galaxy image data, and the winning solution used convolutional neural networks (Dieleman et al., 2015b).

The convolutional neural network approach has only very recently started to be applied to radio galaxy images. One example has been in using convolutional neural networks to infer the presence of a black hole in a radio galaxy (Alger, 2016). Another example is in a recently published paper by Aniyan & Thorat (2017), where the authors present their results on classifying radio galaxy images using convolutional neural networks into the classes of Fanaroff & Riley Type 1 or 2 (FRI/ FRII) (Fanaroff & Riley, 1974) and bent-tailed radio galaxies using a few hundred original images in each class and producing a highly augmented dataset. They use a fusion classifier to combine the results of the three groups because poor results were achieved when attempting to do the three all together. Despite obtaining classification accuracies of above 90% on the FRI and FRII classes, the authors have commented on issues with regards to overfitting due to having few representative samples in each class prior to augmentation, resulting in a small feature space and the fact that the network was highly sensitive to the preprocessing done to the images.

In the case that outputs or labels are not provided alongside the input data to train on, one can use unsupervised machine learning techniques. In regards to machine learning with radio galaxy images, one method uses an unsupervised learning approach involving Kohonen maps (Parallelized rotation/flipping INvariant Kohonen maps, abbreviated to PINK) to construct

prototypes of radio galaxy morphologies (Polsterer et al., 2016).

There are also automated methods that can help to generate labels, therefore the task becomes a supervised machine learning problem. In the astronomical context for example, there are source finding tools that can provide structure to data, and one such tool is PyBDSF (Rafferty, 2016). This is the approach taken in the current work to provide the training labels.

The current work initially aims to classify radio galaxy morphologies into two very distinct classes, consisting of compact sources in one class and multiple-component extended sources in another class using convolutional neural networks. This setup we call the two-class problem. Once an optimal setup of parameters is found, we will test how it will work for the four-class problem of classifying into compact, single-component extended, two-component extended and multiple-component extended sources.

A compact source is an unresolved single component or point source, and an extended source is a resolved source, having at least one component. The detection of point sources is important as they are used for calibration purposes and they are also easier to match to their host galaxy. Making a proper census of unresolved sources is important for mapping out phase calibrators for radio interferometry (Jackson, N. et al., 2016). Although there are more conventional techniques to detect point sources, deep learning provides an alternative approach.

The Lasagne 0.2.dev1 library[4] is used to build a deep neural network to differentiate between different classes of images of radio galaxy data. We compare the classifier metrics obtained on test samples, between the different models.

This paper is outlined as follows: Section 3.4.1 covers some basic theory about neural networks, and the advantages of using deep neural networks. In Section 3.3 we discuss the data provided from Radio Galaxy Zoo, the minor pre-processing steps, and the use of algorithms to help select an image dataset consisting of compact and extended sources. Section 2.4 explores the two-class problem of distinguishing between compact and multiple-component extended sources. It documents the parameters and classifier metrics used. Section 2.5 applies the optimal setup and parameters that were identified in Section 2.4 to the four class problem of classifying between compact and three classes of extended sources. The best-performing setup is also tested to see how well it replicates the findings in Data Release 1 (DR1; Wong et al. in preparation) of the citizen science project Radio Galaxy Zoo.

[4]https://lasagne.readthedocs.io/en/latest/

2.2 Deep neural networks

Neural networks can be used to perform classifications of data. If the input data is in the form of pixels of an image, along with corresponding labels for the image, this information is fed into the input layer of the network (Nielsen, 2015). Neural networks are initialised with a set of weights and biases in the hidden layers (Bishop, 1995). The data is propagated through the network and the output layer computes a prediction. An error is calculated at the output layer using a cost or loss function, which is based on the difference between the true output and the predicted output (LeCun et al., 2012). This error is back-propagated through the network, and the network adjusts the weights and biases to reduce the error (Rumelhart et al., 1986). These steps are iterated a number of times until the cost function is minimised. This is known as training a neural network.

In feed forward neural networks, the nodes in the hidden layers are fully connected to the nodes in the adjacent layers. Therefore, the deeper the network becomes, the more computationally intensive and time consuming it is to train, and often leads to the vanishing gradient problem (Hochreiter, 1991). Convolutional neural networks have been shown to work much more efficiently in high-dimensional data such as image data (Krizhevsky et al., 2012) and although they still suffer from the vanishing gradient problem, one can lessen the impact by proper initialisation of the weights and biases, choosing an appropriate activation function and by doing layer wise pre-training. Such networks employ a number of filters of a certain size, as specified by the user. The receptive field is also referred to as the filter size. The filters are initialised with weights and biases from some distribution, and are connected to a small spatial portion of the input data. Features of the input data are learned through training. In image data, one can achieve a dramatic reduction in the number of parameters through parameter sharing, under the assumption of translational invariance. For example, if one feature is useful to compute at a particular spatial position, it should also be useful to compute at a different spatial position. Parameter sharing is achieved through the use of filters (Karpathy, 2016). One can reduce the computational complexity through data reduction with the use of pooling, in essence a subsampling method. There are several methods of implementing pooling such as max pooling and average pooling (Lee et al., 2016). The current work uses max pooling, where the maximum value within a window of the input feature map is chosen. The convolutional and pooling layers are stacked with the end result being a hierarchical extraction of features. These layers are usually followed by one or more fully-connected layers, before finishing at the output layer, where a prediction is given (Karpathy, 2016).

One problem that occurs with neural networks is overfitting, which is when the architecture and parameters in the neural network fail to generalise to a separate dataset extracted from the same source, that has not been trained on. In this case, the model captures the noise in the data rather than the underlying signal, or there are real features in the training set that

may be peculiar to individual sources but not common to the class as a whole. Overfitting is evident if the validation error is higher than the training error. To reduce the effect of overfitting, one can use image augmentation to artificially generate more images from the original data (Krizhevsky et al., 2012). Another method is to use dropout in the dense or fully-connected layers, where a certain proportion of connections to nodes in adjacent layers are dropped to stop the network relying on the presence of particular neurons, hence it is made to be more robust (Srivastava et al., 2014). Although early stopping is recommended to address the behaviour exhibited by deep neural networks trained on noise, defined as the memorization effect by Arpit et al. (2017), they find that such networks trained with Stochastic Gradient Descent learn patterns before memorizing, even in the presence of noise examples.

2.3 Methods

We utilise the radio galaxy images from the Radio Galaxy Zoo project (Banfield et al., 2015), which uses 1.4 GHz radio galaxy images from the Faint Images of the Radio Sky at Twenty Centimeters (FIRST). The original FIRST data reached a 1σ noise level of 150μJy beam^{-1} at $5''$ resolution (Becker et al., 1995). There are 206399 FITS files in total that contain single-channel image data. The size of the images is mainly (132,132) pixels resampled to a pixel size of $1.37''$.

2.3.1 Pre-processing

The pixel values, representing brightness in mJy/beam were normalised by dividing by 255 such that the values are contained within the [0,1] range. Any 'NaN' pixel value was converted to 0. The images were cropped to (110,110) pixels in order to slightly reduce the amount of data fed into the neural network. We were reluctant to do any further cropping because some of the extended sources tended to be very close to the image boundaries, which is information we did not want to remove. These were the only pre-processing steps taken to the original data. Later on we explore the effect of sigma clipping[5] using a standard deviation of 3 to remove the background noise. This involves calculating the median (m) and standard deviation (σ) of the pixel values, and removing any value above $m + 3\sigma$ and below $m - 3\sigma$. However, deep neural networks should be able to account for the noise in the data without performing additional background noise removal. No procedure has been performed to remove artefacts in the data. As strong sidelobe emission is observed more often in the synthesis imaging of compact radio sources, sidelobe artefacts are expected to be minimal in

[5]http://docs.astropy.org/en/stable/api/astropy.stats.sigma_clip.html

RGZ and similarly so, for the purposes of this paper. Banfield et al. (2015) added 5% of the total sources as compact radio sources thus resulting in a smaller number where the sidelobe pattern could pose an issue. Therefore, we do not expect large numbers of artefacts in the images to be misidentified as radio sources or components to cause an issue with our method. RGZ has a biased selection towards extended sources from the FIRST catalogue.

In order to provide an estimate of the presence of artefacts, we considered the sources in the two-component extended class from the four-class problem and found that 18 out of 11939 sources (0.15%) contained one component having a total flux that was at least 50 times that of the other component.

2.3.2 PyBDSF

PyBDSF (the Python Blob Detector and Source Finder, formerly PyBDSM) by Mohan & Rafferty (2015a) is a tool designed to decompose radio interferometry images into sources composed of a set of Gaussians, shapelets, or wavelets. For the purposes of the current work, we assume that each image is of a single source or radio galaxy. Therefore, PyBDSF will detect the components belonging to the source.

In order to provide some initial structure to the data, we used the default settings of the PyBDSF version 1.8.11 'process_image' task to help count the number of components in each image. The default settings include using 5-sigma for the pixel threshold and 3-sigma for island boundaries. The number of output lines in the resulting .srl file from running PyBDSF provides the user with the number of components that were able to be fit, using Gaussian fitting. The images were all initially run through PyBDSF. Out of the original 206399 images, 30945 produced an error, either due to the image having all blanked pixel values, presenting as NaNs (94.6%), or there were no components detected in the image (5.4%). 175454 images were successfully able to be processed by PyBDSF, and produced source list (srl) files that contained information about each detected source. In the successfully processed images, 99.7% contained an NaN pixel percentage in the range between 0 and 10%. The highest percentage of NaN pixel values was 93.2% and the median was 1.9%. The NaN values occur only along the edges of the images and are due to observations at the edges of fields. Table 2.1 lists the number of components detected in each image by PyBDSF, showing the results up to eleven components.

There are sometimes discrepancies between the number of components that PyBDSF had detected and how many there visually appeared to be in the image, therefore PyBDSF does not always perform as a human would in counting the number of components in the image. These inconsistencies remained even if the grouping parameters were altered. It was found that the number of inconsistencies detected increased with the number of components in the

Table 2.1: The number of components detected by PyBDSF including how many of these sources there are, for up to 11 components.

PyBDSF number of components	Number of sources
1	63051
2	66589
3	29482
4	10437
5	3517
6	1136
7	510
8	264
9	163
10	79
11	48

image.

2.3.3 Image augmentation

The classification accuracy of deep neural networks increases with the size of the training set. It is possible to generate more images through label-preserving transformations such as horizontal, vertical translation and rotations (Krizhevsky et al., 2012). This method is called augmentation and reduces the amount of overfitting to the data. It can also improve performance in imbalanced class problems (Wong et al., 2016).

We augmented our images using translation, rotation and flips but not skewing or shearing the data since such transformations applied to compact sources can make them appear as having extended emission, which would render the label incorrect. The amount by which the images are translated is within the range of 0 to 22 pixels of the image width and height. Since no boundary conditions have been applied to the images, it is likely that 2.9% of images in the two-class problem and 1.0% in the four-class problem are likely to have components that have been shifted out of the image. The images are rotated by any random angle between 0 and 360 degrees. The Keras[6] package 2.0.3 was used to produce the augmented images. Keras is a high-level neural networks API, developed with the aim of enabling fast experimentation. It is written in the Python language and able to be run on top of either TensorFlow [7] or Theano[8].

Fig. 2.1 shows examples of rotation, shifting and flipping on a source with extended emission. The image is an example of how some extended sources that have a small amount of extended

[6]https://keras.io/preprocessing/image/
[7]https://www.tensorflow.org
[8]http://deeplearning.net/software/theano/

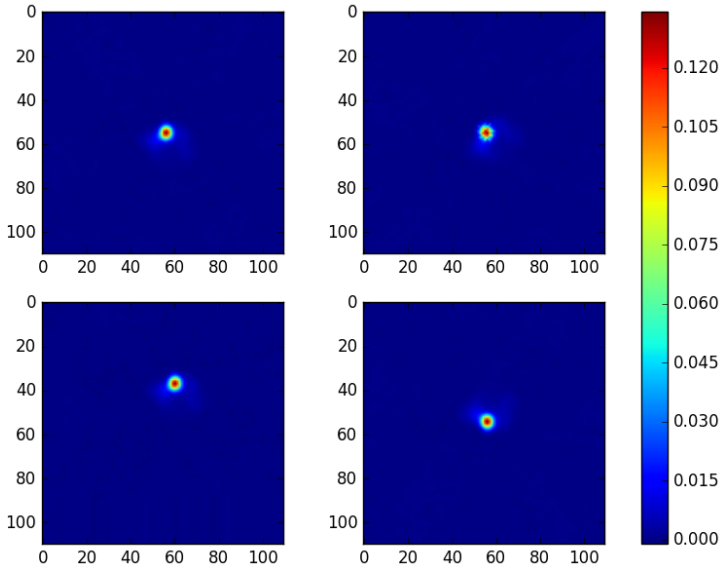

Figure 2.1: Examples of image augmentations with an extended source. The original image is shown on the top left. The transformations, from left to right, top to bottom are a random rotation, shift and flip. The size of the images is (110,110) pixels, with an angular resolution of 1.14". The colour bar represents the normalised flux densities.

emission can look similar to compact sources, therefore presenting challenges for deep learning methods or other programs used to extract information from images.

2.3.4 Deep learning algorithms

There are several deep learning implementations currently available for use. The present work uses Lasagne 0.2.dev1 (Dieleman et al., 2015a), a lightweight library to build and train neural networks in Theano using the Python language. Python version 2.7.12 is used and the Theano version is 0.9.0dev2. Theano is a Python library that allows the user to define, optimize, and evaluate mathematical expressions involving multi-dimensional arrays efficiently. Some features of Theano include the ability to use Numpy arrays in Theano-compiled functions, the transparent use of a graphics processing unit (GPU), which enables data intensive computations to be accomplished much faster than on a CPU, and the ability to compute derivatives for functions with one or many inputs. The python library Lasagne is built on top of Theano, but leaves the Theano symbolic variables transparent, so they can be easily altered to make changes to the model or learning procedure, as desired. This provides the user with more flexibility compared to that of other libraries.

Several network models built using the Lasagne library have been trained, in order to see the setup of parameters that results in the optimal test classification accuracy. The learning rate was set to 0.001 at the beginning and reduced by a factor of 10 at four points during

training. 1000 training epochs in total were used for all the models shown. The network parameters at the 1000^{th} epoch were chosen for the final validation of the results. Training was stopped at this time because the training and validation losses appeared to reach their minimum and only fluctuated around this value, without the validation loss becoming higher than the training loss in the attempt to avoid overfitting, unless otherwise stated.

A simple manual tuning strategy was used to optimise the hyper-parameters, that involved experimenting with batch sizes of 8, 16 and 32 against different chunk sizes and learning rates. A batch size of 8 was found to give optimal results. The batch Stochastic Gradient Descent method (Bottou, 1998) was used, where the gradient is computed using the input dataset with a specified batch size, rather than using a single sample. The momentum update method used was Nesterov, with a momentum of 0.9 and a weight decay of 0. The Nesterov momentum update evaluates the gradient at the future rather than current position, resulting in better informed momentum updates and hence improved performance (Sutskever, 2013). The validation step is done every 10 training epochs. The networks were trained on a single NVIDIA Tesla K20m GPU, with CUDA version 8.0.61. The categorical cross-entropy[9] cost function was used, which has the following form:

$$L_i = -\sum_j t_{i,j} \log(p_{i,j}) \,, \tag{2.1}$$

where i, j denotes the classes and observations respectively, $t_{i,j}$ represents the targets and $p_{i,j}$ represents the predictions. Equation (2.1) is used for predictions falling in the range (0,1) such as the softmax output of a neural network. The outputs of the softmax function represent the probabilities that the images belong to the given classes, and add up to 1. The predictions are clipped to be between 10^{-7} and $1 - 10^{-7}$ in order to make sure that they fall within the (0,1) range. There are over 1.6M parameters to train in total.

At the conclusion of training, the predictions at the final layer of the network are rounded to 0 or 1. In the two-class problem, the output [1,0] represents a compact source and [0,1] represents a multiple-component extended source. Training, validation and test classification accuracies are calculated using the proportion of rounded predictions that matched the labels. The image and label data have had the rows shuffled at two stages to make sure that there was no sampling bias when choosing the training, validation and test sets. A dropout of 50% has been applied to the dense layers. The ReLU activation function (Glorot et al., 2011) was used in the convolutional layers. The ReLU function is max(0,x), therefore only positive inputs are sent forward, and the negative ones are set to 0. This makes the network more sparse, therefore more efficient. Since the output is linear only in parts of the network,

[9] http://lasagne.readthedocs.io/en/latest/modules/objectives.html#lasagne.objectives.categorical_crossentropy

this ensures that the gradients flow across the active neurons, hence avoiding the vanishing gradient problem. The PReLU activation function (He et al., 2015a), which uses a negative linear function with a coefficient to control the negative part of the function was also tried, however it resulted in slightly worse accuracies compared to using the ReLU. In the dense layers, the identity activation function was used. The weights were initialised with the Uniform Glorot distribution (Glorot & Bengio, 2010), which has the following form when the ReLU activation function is used:

$$\sigma = \sqrt{\frac{2}{(n_1 + n_2) \cdot f}} \quad , \tag{2.2}$$

where n_1 and n_2 is the number of connections coming in and out of the layer respectively, and f is the receptive field size. The biases were initialised with the constant 0.

In section 2.4, we explore the effect of varying the number of convolutional layers. Section 2.4 investigates the effect of adding augmented data for varying chunk sizes, and section 2.4 explores the effect of using only a subset of the original provided images. The chunk size refers to the number of data examples per iteration and should be divisible by the batch size for optimal performance.

2.3.5 Selection of sources for two-class classification

In a first step, we applied a deep learning approach to two very distinct classes of radio sources: compact sources and multiple-component extended sources. Once this setup is optimised, we consider classification involving four classes.

In the current work, we define our sample of compact sources from the images where PyBDSF detected a single component, and additionally using Equation (2.3) from Kimball & Ivezić (2008) as follows:

$$\theta = \Big(\frac{F_{\rm int}}{F_{\rm peak}}\Big)^{\frac{1}{2}}, \tag{2.3}$$

where $F_{\rm int}$ and $F_{\rm peak}$ are the integrated and peak flux intensities, respectively. According to this definition, values of $\theta \sim 1$ are highly concentrated (unresolved), while components with larger θ are extended (resolved). Kimball & Ivezić (2008) adopt $\theta \approx 1.06$ as the value separating resolved and unresolved components, where components above $\theta \approx 1.06$ are resolved. We therefore define our compact components as those having values $\theta < 1.06$. If there was only one compact component in the image, then it was classified as a 'compact' source; there were 2682 such cases. The $F_{\rm int}$ and $F_{\rm peak}$ values were extracted from the provided FITS files,

Table 2.2: Summarising the number of images used for the two-class problem

Source/Image type	# Original	# Augmented
Compact	2682	15558
Multiple-extended	18000	144633
Total	20682	160191

using the 'imfit' function from CASA Version 4.7.2-REL. Several batches of samples assigned to the 'compact' class were additionally examined visually to verify that they truly appeared to be compact sources.

The choice of multiple-component extended sources was taken from a random sample of 18000 images where PyBDSF had detected at least 3 components. This sample can include images of multi-component compact sources.

Taking this sample of compact and extended sources, there are 20682 images all together that can be divided into a training, validation and test data set for the initial deep learning approach. The number of images used for the two-class problem is summarised in Table 2.2. Fig. 2.2 shows some typical examples of compact and multiple-component extended sources. It should be noted that there are many more examples of multiple extended sources compared to compact sources, however the compact sources display a very well defined morphology compared to the multiple extended sources, which can assume an almost infinite number of unique morphologies.

In examining the images where PyBDSF had detected at least three components, it appears that some of the images contain superpositions, or have fewer than three components in the image. Upon closer inspection of a random sample of 250 images where PyBDSF has detected at least three sources, there were roughly 44% that appeared to be superpositions or that visually had fewer than three components in the image. This means that a substantial number of images assigned to the multi-component class do not truly belong, however there is still a stark contrast in morphology compared to the sources chosen for the compact class, therefore it should not have an overly detrimental effect on the classification accuracies. We attempt to eliminate these contaminated images when choosing sources for the four-class problem.

2.3.6 Selection of sources for four-class classification

Assuming there is an optimal choice in hyper-parameters that results in a high classification accuracy for the two-class problem of distinguishing between compact and multiple-component extended sources, we also wanted to see how such a setup would be able to distinguish between sources belonging to four classes. We chose the images belonging to cate-

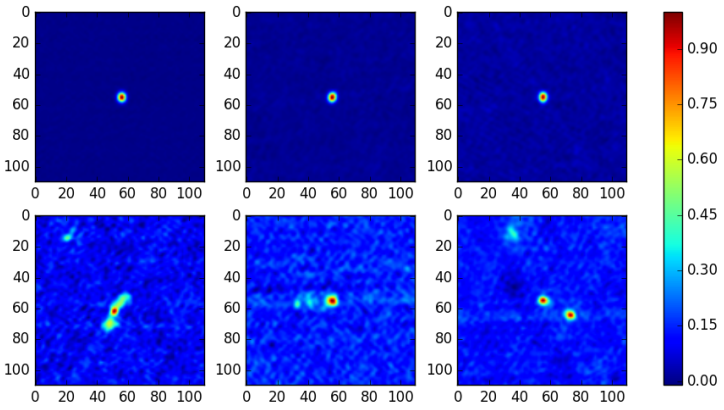

Figure 2.2: Examples of compact and multiple component extended source classifications that we initially aim to make our deep neural networks differentiate between. The top row of images represents compact sources, whereas the bottom row represents multiple component extended sources. The colour bar represents the normalised flux densities.

gories of compact sources, single component extended, two component extended and multiple component extended sources. The compact sources are the same ones as were used for the two-class problem, and the multiple-component extended sources are a subset of the ones used for the two-class problem. The single-component extended and two-component extended classes are the new classes, and the images belonging to them have not previously been used for the deep learning approach.

The labels for the images were able to be generated with the help of the 'S_code' output of PyBDSF. The S_code quantity defines the component structure (Mohan & Rafferty, 2015a) and the output values are defined as such:

- 'S' = a single-Gaussian component that is the only component in the island
- 'C' = a single-Gaussian component in an island with other components
- 'M' = a multi-Gaussian component

The four classes are described below:

- Compact source: Sources where PyBDSF has detected one component and choosing sources as defined by Equation (2.3) from Kimball & Ivezić (2008). The same set of compact sources were used for the two-class problem.
- Single component extended source: Sources where PyBDSF has detected one component, and the S_code quantity contains an 'M' (multi-Gaussian component).
- Two component extended sources: Sources where PyBDSF has detected two components, and the S_code quantity contains an 'M' (multi-Gaussian component) for both components.

- At least three component extended sources: Sources where PyBDSF has detected at least three components. We started with the set of 18000 images as for the two-class problem, required that the S_code quantity contains at least two 'M's, and any number of 'C's. Additionally, two blob-detection algorithms (logarithm of gaussian and difference of gaussian) were run using the scikit-image 0.17.1 package in Python[10]. The images were also all inspected visually in an attempt to ensure that each image contained at least three extended components that appeared to be part of the same source, rather than being superpositions of sources. After this, there were 577 images remaining. However upon cross-checking with several optical/IR images, more than 40% of this subset of images still appeared to contain superpositions of components associated with more than one AGN. Therefore, although the classification successfully identifies multiple-component structures, they are contaminated by such superpositions in comparison with Radio Galaxy Zoo classifications.

The condition 'S_code=S' was not found to be useful in characterising components. Occasionally there was a source that appeared as though it should belong to another class, so a small level of label contamination must be accepted. The number of images used for the four-class problem is summarised in Table 2.3 and Fig. 2.3 shows some example images for each of the four classes. The four-class classification scenario also contains an imbalance in the number of original images for each class, however this can be alleviated by augmenting the classes of data displaying richer morphologies more (single, two extended and multiple component extended sources), compared to the compact sources.

The existence of the remaining superpositions of sources in the multiple-component extended class in the training set means that the deep learning algorithm will not be able to make the distinction between images that contain superpositions, and images with components that are likely to be part of the same source. Even radio galaxy experts cannot always reach a consensus about these differences.

The fact that the compact and single-component extended sources all come from the set of images where PyBDSF has detected a single source mean that the deep learning algorithm is doing more than just learning the method by which PyBDSF counts components. It is also performing the source structure functions of PyBDSF, with the advantage that it uses solely image data to learn about the differences in morphology between compact sources and single component extended sources.

[10]http://scikit-image.org/docs/dev/auto_examples/features_detection/plot_blob.html

Table 2.3: Summarising the number of images used for the four-class problem.

Source/Image type	# Original	# Augmented
Compact	2682	15558
Single-extended	6735	43099
Two-extended	11939	35994
Multiple-extended	577	46381
Total	21933	141032

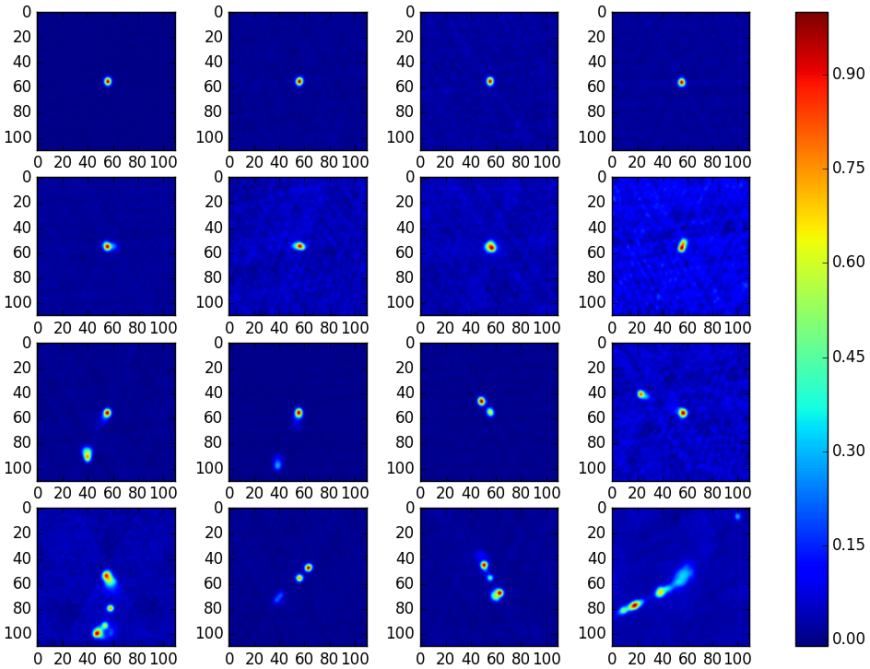

Figure 2.3: Examples of compact, single-extended, two-component extended and multiple-component extended sources, for a deep neural network to differentiate between. Top row: Compact sources. Second row: Single-component extended sources. Third row: Two-component extended sources. Fourth row: Multiple-component extended sources.

2.4 Results for two classes

Our first aim is to see how well a deep neural network is able to distinguish between two classes of data that are very morphologically distinct: compact sources and multiple component sources. There were 2682 compact and 18000 multiple-component extended sources, giving a total of 20682 images provided as input data for classification by the convolutional neural network designed in Lasagne. When the augmentation data is used as well, there are a total of 180873 images. The number of sources and augmented images used is summarised in Table 2.2.

The results shown are the classifier metrics on the validation and test data sets. The extended source class is used as the positive class for the metrics, therefore a true positive (TP) is defined as when an extended source is predicted that is also labelled as an extended source. A false positive (FP) is defined when an extended source is predicted, but is labelled as a compact source. A false negative (FN) is defined when a point source is predicted, but is labelled as an extended source. The following four metrics are used to evaluate the performance of the classifier:

- Precision = TP/(TP+FP)
- Recall = TP/(TP+FN)
- F1 score =(2×Precision×Recall)/(Precision + Recall)
- Accuracy = (TP+TN)/(TP+FP+TN+FN)

where FP, FN and TN denotes false positives, false negatives and true negatives respectively. For the current task of classifying between extended and point sources, precision represents the classifier's ability to not classify point sources as extended sources. Recall evaluates the classifiers ability to not classify extended sources as point sources, hence provides an estimate of the sensitivity of the classifier, in whether it can correctly predict the labeled extended sources. It is worth noting that in the literature, precision is often called “reliability" and recall is often called “completeness" (e.g. Hopkins et al. (2015)). The F1 score can be interpreted as the weighted average of precision and recall. The accuracy is the overall classification accuracy across the classes, how many correct predictions for the labeled point and extended sources were made overall. The F1 and accuracy scores tend to correlate highly. The precision, recall, F1 score and accuracy metrics were calculated for both the validation and test data sets to assess the performance of each deep neural network model. It should be noted that in machine learning theory, the precision scores are a better assessor of performance compared to accuracy in imbalanced dataset problems. However, we address the imbalance in our dataset through augmentation, therefore use the classification accuracy to assess the performance of the classifiers. The training and validation losses are also plotted as a function of epochs for

several chosen models.

In order to assess which models are significantly better than others, rather than arising as a result of random fluctuations, we use the root mean square error (RMSE) measure to quantify the scatter in the overall accuracies, for the original data. The RMSE is defined according to Equation 2.4.

$$RMSE = \sqrt{\frac{1}{n}\sum_{j}^{n}(p_{i,j} - t_{i,j})^2}\,, \tag{2.4}$$

where i, j denotes the classes and observations respectively, n is the total number of observations, $t_{i,j}$ represents the targets and $p_{i,j}$ represents the predictions. We consider any value beyond two times the RMSE value to be significant in terms of metrics. This error estimate is conservative in that it is a measure of the actual scatter, as opposed to the derived error in the mean accuracies.

Effect of increasing convolutional layers

We first explore the effect of increasing the number of convolutional layers, in order to see the effect the model complexity has on obtaining better classification accuracies, without excessive overfitting.

The results in Tab. 2.5 and Fig. 2.4 show the effect of adding an increasing number of layers to the network. Simply using two dense layers results in precision, recall and F1 scores above 0.95 and a test accuracy above 93%. The addition of two adjacent convolutional layers and the use of sigma clipping produces a classification accuracy of 97.0%. Taking into account the RMSE values to establish random fluctuations in accuracy, the model that is significantly better than all others is the three convolutional and two dense layer model with sigma clipping (model F), achieving the optimal accuracy of 97.5%. However, this setup results in overfitting as shown in Fig. 2.5, and hence we exclude this model. The next best-performing models that are significantly better than the others, without causing overfitting, are models D and E.

Using two adjacent convolutional layers followed by a pooling layer as opposed to using a single convolutional layer followed by a pooling layer reduces the number of parameters, given that the two filter sizes of the adjacent convolutional layers are smaller compared to using a single larger one (Simonyan & Zisserman, 2014). When putting a max pooling layer in between the first and second convolutional layer, it had a detrimental effect on the test accuracy, reducing it by almost 1% which is significant given the RMSE values, and it took more training epochs to attain a smaller training loss (results not shown). The radio galaxy images with extended emission generally have structure that span across large portions of the image, yet it would

Table 2.4: The deep learning models that were explored.

Code	Model	# Pooling layers
A	2 dense	0
B	1 conv + 2 dense	1
C	2 conv + 2 dense	1
D	2 conv + 2 dense sigma clip	1
E	3 conv + 2 dense	2
F	3 conv + 2 dense sigma clip	2

increase the number of parameters by too much of a factor if a single convolutional layer with a very large receptive field, or filter size was used. Therefore, it is better to combine two adjacent convolutional layers that have smaller receptive fields.

The deep learning algorithm appears to be robust to the classes being imbalanced; there are approximately 9 times more examples of the multiple extended class images compared to the compact source images. However, the compact sources have a much more stable morphology, largely consisting of a source in the centre of the image, compared to the multiple component extended class images, which can be spread out all over the image.

Considering the results for the test data set and taking into account the RMSE values, the precision (reliability) values are on average significantly higher compared to the recall (completeness) values. This implies that the classifier is better at not classifying the multiple-component extended sources as point sources but is not as sensitive in identifying all the labeled multiple-component extended sources. The training losses begin at a low value of around 0.27 and quickly settle to their minimal value for a particular model by 200 epochs. A likely reason why the losses begin and remain low during training is because the classes contain images that are morphologically very different; one containing a single concentrated source in the centre of the image and the other generally containing multiple sources that are spread throughout the image.

The fact that a very substantial number of images belonging to the multiple-component extended class appear to contain superpositions or visually appear as though they contain fewer than three components probably does not hinder the classification accuracies significantly, since the contents of the images are very different between the two classes.

The memory requirements for a typical run using the three convolutional layer architecture is 1.87 GB, with a computational time of 192 minutes using a single NVIDIA Tesla K20m GPU, with CUDA version 8.0.61.

Table 2.5: Effect of increasing the number of convolutional layers for the original images. The precision, recall, F1 score and accuracy values are shown for both the validation and test data sets, calculated over 1000 training epochs. The validation set is used every 10 epochs, and the final trained parameters are used on the test data set after training is complete. 20682 images were used in total, with a chunk size of 6000, and the training samples make up 60% of the total data.

	Valid. Prec.	Recall	F1 score	Accuracy	RMSE
A	96.6%	95.6%	96.1%	93.3%	0.27
B	97.9%	97.0%	97.4%	95.6%	0.22
C	97.4%	97.5%	97.4%	95.6%	0.20
D	98.2%	96.9%	97.5%	95.7%	0.21
E	98.6%	97.5%	98.0%	96.6%	0.19
F	98.4%	97.5%	97.9%	96.4%	0.19
	Test. Prec.	Recall	F1 score	Accuracy	RMSE
A	97.4%	95.7%	96.6%	94.0%	0.26
B	98.2%	96.3%	97.3%	95.3%	0.22
C	97.7%	96.7%	97.2%	95.1%	0.21
D	98.1%	98.4%	98.3%	97.0%	0.19
E	98.2%	97.8%	98.0%	96.5%	0.21
F	98.7%	98.3%	98.5%	97.5%	0.18

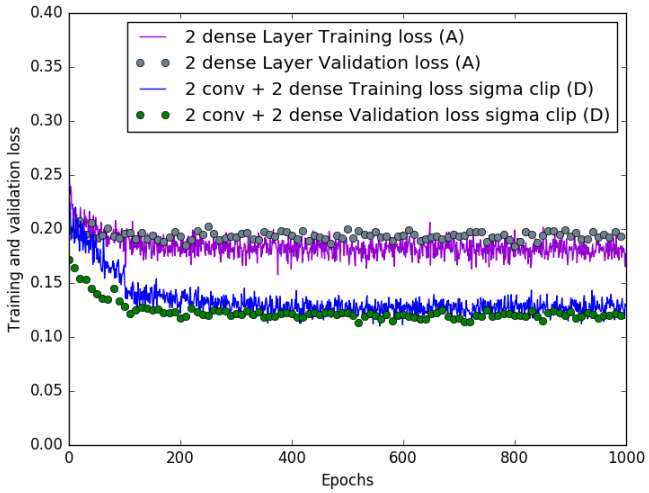

Figure 2.4: Plot of training and validation losses as a function of training epochs, for models A and D in Table 2.4. The higher training and validation losses are from using only 2 dense layers and no convolutional layers, which are the highest losses amongst the six models and consequently produced the lowest classification accuracies. Adding 2 convolutional layers produces lower training and validation losses, and therefore improved classification accuracies. The 2 convolutional and 2 dense layer architecture with sigma clipping was one of two models that performed the best out of all the models considered for this set of images.

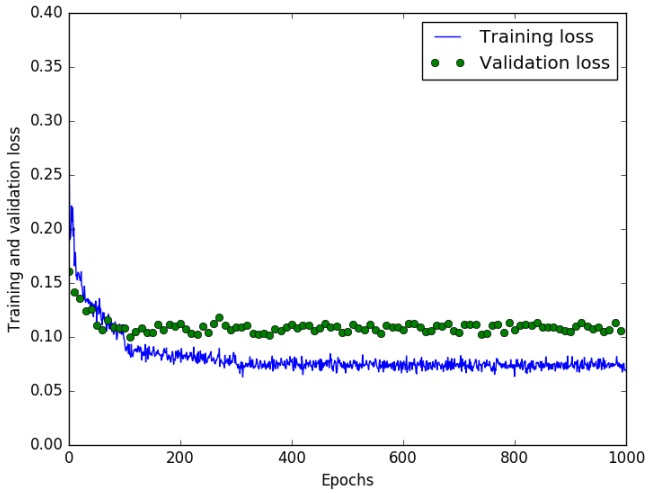

Figure 2.5: Plot of training and validation losses as a function of training epochs, for a three convolutional and two dense layer model with sigma clipping (model F). Despite this model achieving the highest test accuracy for the original set of images, overfitting is evident as the validation losses are higher than the training losses.

Effect of including augmented data

Next we studied the effect of image augmentation on the classification accuracies. Table 2.6 and Fig. 2.6 shows that when the full set of augmented data is used in addition to the original images, it results in overall significantly improved F1 scores, validation and test accuracies, compared to when the original data is used. The use of the augmented images enables the choice of the larger chunk size leading to improved accuracy without causing the network to overfit, hence a chunk size of 20000 is used, compared to the previous size of 6000. The chunk size should be made as large as possible for a given set of data, since the more training examples are seen simultaneously, the more accurately the weights can be adjusted to produce a lower training loss. The best performing architecture with the original and augmented images is the three convolutional and two dense layer architecture, with no sigma clipping (model E). This setup achieves the highest observed test accuracy for the two-class problem of 97.4%. Model D performs equally well in terms of overall accuracy when taking into account the RMSE values, however there is a greater difference in the training loss compared to the validation loss, as is evident in Fig. 2.6. Therefore, model E is the overall best-performing model, since the training and validation losses are closer together.

The most likely reason why a higher accuracy is unable to be achieved is that there is a small amount of label contamination, for example a few of the images in the multiple component extended class may look more like compact sources. This is due to PyBDSF detecting multiple components in an image, even though visually the image appears to only contain a compact

Table 2.6: Effect of using all augmented images in addition to original data. The precision, recall, F1 score and accuracy values are shown for both the validation and test data sets, calculated over 1000 training epochs. The validation set is used every 10 epochs, and the final trained parameters are used on the test data set after training is complete. 180873 images were used in total, with a chunk size of 20000, and the training samples make up 60% of the total data

	Valid. Precision	Recall	F1	Accuracy
C	97.0%	98.2%	97.6%	95.7%
D	98.3%	98.3%	98.3%	96.9%
E	98.8%	97.9%	98.4%	97.0%
F	98.6%	97.8%	98.2%	96.8%
	Test Precision	Recall	F1	Accuracy
C	96.6%	98.5%	97.6%	95.6%
D	98.7%	98.4%	98.5%	97.4%
E	99.2%	97.9%	98.6%	97.4%
F	98.7%	97.8%	98.2%	96.9%

Table 2.7: Details of the layer parameters used for the best-performing model. The # of parameters gives a cumulative sum at each layer. There are 1676914 trainable parameters in total.

Layer	Depth	Filter Size	Stride length	#Parameters
Conv2D	16	8	3	1040
Conv2D	32	7	2	26160
MaxPool2D	32	3	-	26160
Conv2D	64	2	1	34416
MaxPool2D	64	2	-	34416
Dense	1024	-	-	625264
Dense	1024	-	-	1674864
Softmax	2	-	-	1676914

source, as shown in Fig. 2.7. The three convolutional and two-dense layer architecture is shown in Fig. 2.8, and the details of the layers with the number of parameters used are shown in Tab. 2.7.

Fig. 2.10 shows the features that are learnt in the first and third convolutional layers for the three convolutional layer architecture, halfway through training at 500 epochs.

Effect of using a subset of images

Next we explored the effect of using only a subset of images. Using a subset of the available images (1000 original and 1000 augmented) tended to significantly reduce the validation and test scores compared to when using the full set of original images, as shown in Tab. 2.8, caused greater fluctuations during training, and introduced a higher level of overfitting as shown in Fig. 2.9. The larger fluctuations during training are most likely due to the algorithm not

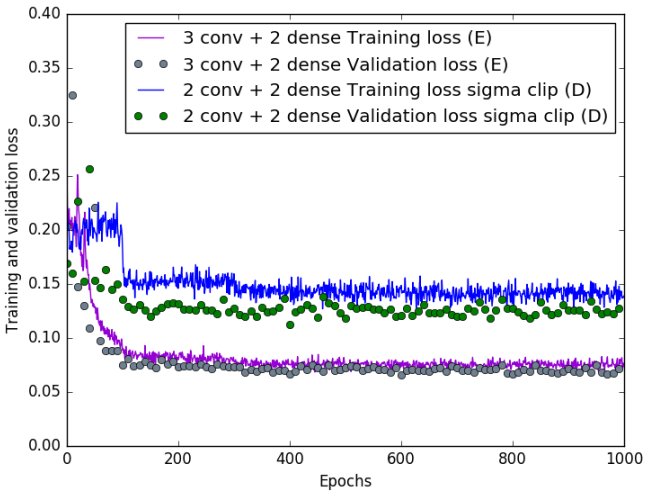

Figure 2.6: Plot of training and validation losses for the 2 conv + 2 dense layer with sigma clipping (model D) and 3 conv + 2 dense layer (model E), when using the original and augmented data. The training and validation losses are higher and fluctuate more for model D, and there is a greater difference between the training and validation losses compared to model E, despite achieving a similar test classification accuracy. Taking these factors into account, model E performs better overall.

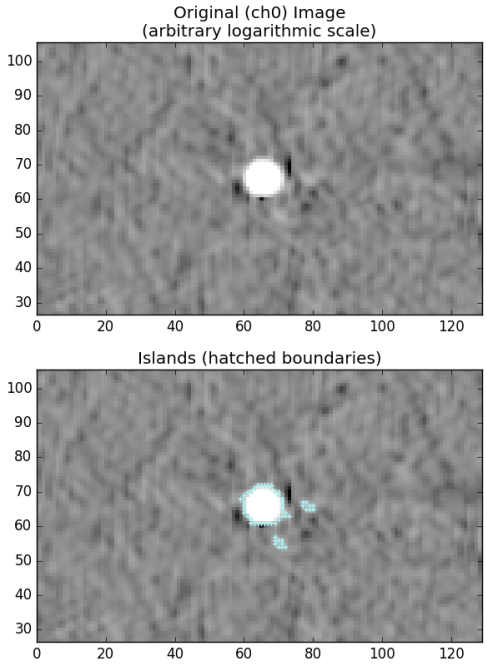

Figure 2.7: Example of an image where PyBDSF has detected 3 components, even though the image appears to be that of a point source.

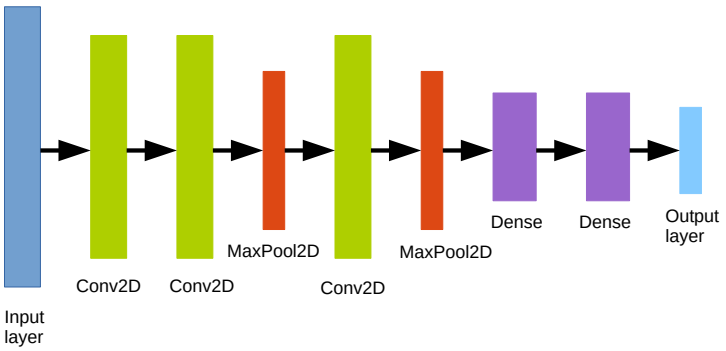

Figure 2.8: The 3 conv + 2 dense architecture, which constituted the best performing model. The colours are arbitrarily chosen to represent the different layers used.

Table 2.8: Effect of using a subset of the original and augmented images. The precision, recall, F1 score and accuracy values are shown for both the validation and test data sets, calculated over 1000 training epochs. The validation set is used every 10 epochs, and the final trained parameters are used on the test data set after training is complete. 1000 original and 1000 augmented images were used (2000 in total) with a chunk size of 400, and the training samples make up 60% of the total data.

	Valid. Precision	Recall	F1	Accuracy
C	94.1%	97.8%	95.9%	92.7%
D	95.2%	96.4%	95.8%	92.6%
E	95.3%	96.2%	95.7%	92.4%
F	95.3%	96.3%	95.8%	92.5%
	Test Precision	Recall	F1	Accuracy
C	96.5%	94.0%	95.2%	91.4%
D	93.8%	98.1%	95.9%	93.0%
E	95.3%	95.3%	95.3%	92.2%
F	94.4%	91.8%	93.1%	88.3%

seeing as large a number of samples at a time compared to when the full set of images is used, hence the weights cannot be estimated as accurately for each subsequent training epoch. The validation and test accuracies however still remained above 90%, with the exception of model F (three convolutional and two dense layer setup with sigma clipping.)

Tensorflow for Poets

'Tensorflow for Poets' uses the 'Inception v3' network, a pre-trained deep neural network that is trained for ImageNet Large Visual Recognition Challenge. It is able to differentiate between 1000 different classes. We used this approach to perform classifications and compare the results to the custom-designed networks using the Lasagne library, however we found the results to be inferior. This poorer performance can be explained by the fact that the class types trained on are mainly examples of every-day objects and animals rather than scientific

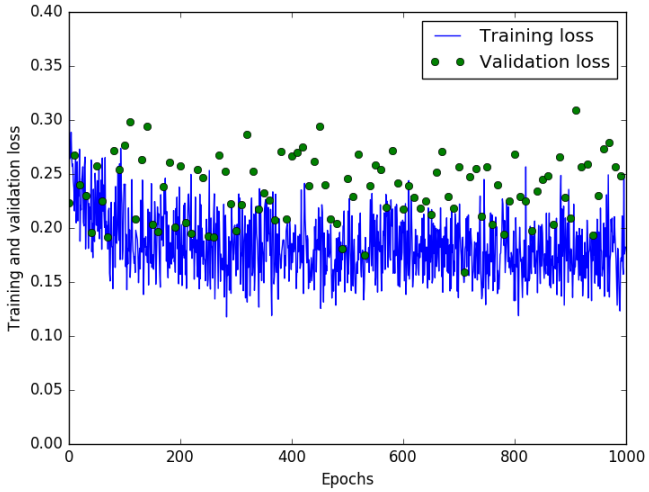

Figure 2.9: Training and validation losses when using only 1000 original and 1000 augmented images, when using the 2 convolutional and 2 dense layer setup with sigma clipping (model D). The training losses are around the same compared to when using the full set of 20682 images, and the fluctuations are greater. There is also some amount of overfitting.

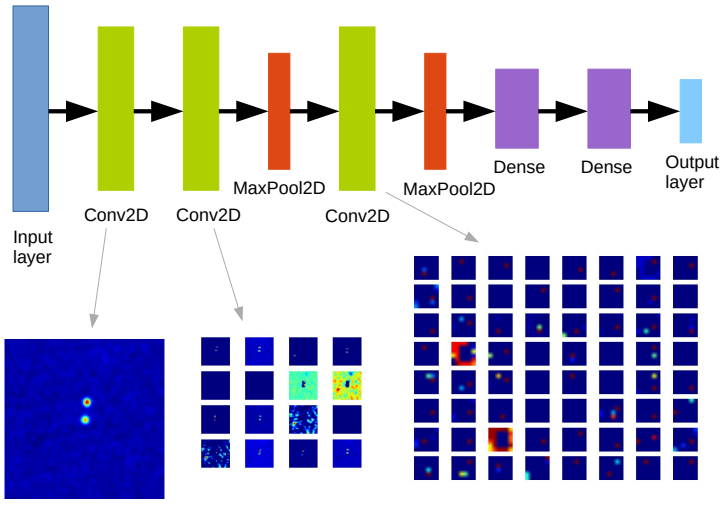

Figure 2.10: Showing the input image, first and third convolutional feature map activations at 100 epochs into training using the three convolutional and two dense layer architecture. The colours in the architecture are arbitrarily chosen to represent the different layers used.

Table 2.9: Results for four-class model. The difference between using sigma clipping or not is very minor, and can be attributed to random fluctuations for each subsequent run.

	Valid. Precision	Recall	F1	Accuracy
E	92.6%	92.7%	92.7%	92.0%
F	93.2%	93.3%	93.2%	92.7%
	Test Precision	Recall	F1	Accuracy
E	94.0%	94.1%	94.0%	93.5%
F	94.0%	93.9%	93.9%	93.5%

images. Another reason is that using a custom-designed network has much more freedom in adjusting parameters compared to using a 'black-box' approach, where more parameters are fixed.

2.5 Results for four classes

In the previous section we have explored varying several parameters using the custom designed network in Lasagne and found the optimal one that results in the highest test classification accuracy for the two-class problem of distinguishing between compact and multiple component extended sources, which was the 3 convolutional and 2 dense layer architecture without sigma clipping (model E), using both original and augmented images. Given these results, we wanted to see how well such a deep neural network setup could distinguish between two additional classes of data, consisting of single component extended and two component extended sources.

The same parameters were used as was described in Section 2.3.4. Two models were explored for the task of four-class classification; the 3 convolutional and 2 dense layer architecture with and without sigma clipping, using both original and augmented images. This architecture and set of images performed best on the two-class problem, which is why it was chosen for the four-class problem. The numbers of images used are summarised in Table 2.3. The issue with class imbalance was addressed by augmenting the single, two and multiple-component images to achieve roughly the same number of images for these extended sources. We used the same set of original and augmented images for the compact sources as for the two-class problem. The results shown in Table 2.9 are the classifier metrics on the validation and test data sets, as was done similarly for the two-class problem, however applying a 'macro' average over the four classes to obtain an overall summary of the number of true and false positives and negatives across the confusion matrix.

The inclusion of an additional two classes of data results in a significantly reduced performance compared to when only two classes are used. This is likely due to the low-level amount of label

Table 2.10: Individual precision and recall values computed from the confusion matrix for the 4-class test set, using the original and augmented images, for model E.

	Precision	Recall
Compact	96.9%	97.4%
Single-extended	93.4%	95.3%
Two-component extended	91.1%	87.6%
Multiple-component extended	94.6%	96.1%

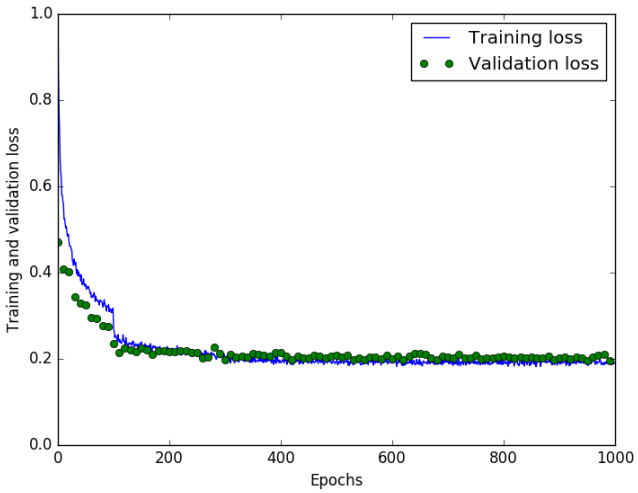

Figure 2.11: Training and validation losses shown when using a chunk size of 20000 with a 3 conv + 2 dense model with no sigma clipping (model E), for the four-class problem. The training losses are much higher at the start compared to what was observed in the 2-class problem, and settle to a loss of around 0.2 by 400 training epochs.

contamination in the new classes, in addition to the level already present in the previous two classes. The manually chosen images for the multiple-component extended source class still contain a substantial number of images that are superpositions. Despite this, the accuracies remain above 93%.

Note that our machine learning algorithms are still making the correct decision in determining membership in each of the four classes, in that they were trained to simply recognise the number of extended components in close proximity. The information required to identify some of these images as superpositions of more than one physical radio galaxy requires more detailed information about both the radio morphology and the location of possible optical/IR counterparts. This is a future task for machine learning algorithms to make use of labels with higher-level information, for example from Radio Galaxy Zoo.

The individual precision and recall values were computed for each of the four classes in Table 2.10. The results show that the precision and recall values are the highest for the compact sources, so the deep learning algorithm is able to identify all the compact sources and not confuse them with any other source, with most accuracy. This is to be expected since they have a very well-defined morphology with the least variability amongst the classes. The deep learning algorithm however produces the lowest scores for the two-component extended sources, and this is likely because there is the most overlap between these sources and the two classes on either side; the single-extended and multiple-extended sources.

These higher-level classes of data can be used as an initial step to facilitate the generation of more specific radio morphology classes of scientific interest.

2.5.1 Comparing results with Data Release 1 of the Radio Galaxy Zoo

Radio Galaxy Zoo classification is a two-step process. For a single classification, users firstly select all radio emission they consider to be originating from a single radio galaxy. After selecting the radio components, users will try to match it with a galaxy in the near-IR data. If there are multiple radio sources in the image, users can repeat both steps to identify other radio galaxies. Individual classifications are aggregated to provide a consensus classification of the image based on the majority vote (Willett, 2015).

Data Release 1 of the Radio Galaxy Zoo (DR1; Wong et al. in preparation) was made with the purpose of obtaining citizen-scientists input in identifying which components belonged together in a source. The ‘Number of components’ is defined as the number of discrete radio components that a source encompasses, that also depends on the lowest radio contour level chosen. The ‘Number of peaks’ that examines the components identified by RGZ participants, refers to how many peaks are present in the radio source as determined by an automatic pipeline processor. DR1 consists of 74627 sources where user weightings have been applied to

the consensus levels, retaining the sources which have a consensus level of 0.65 or higher. The minimum reliability of DR1 is 75% for a minimum weighted consensus level of 0.65 for the classifications of the FIRST survey. Using the dataset for the four-class problem consisting of 21933 images, there are 10722 (14.4%) images in common with the DR1 dataset, where the matching is done based on the source name. After removing the sources that contained invalid entries in the 'matchComponents' and 'WISECATmismatch' columns, there were 9537 remaining (12.8%).

Using the 'Number of components' and 'Number of peaks' information provided that originated from the citizen scientists' and the post-processing pipeline, along with the images in the DR1 dataset, we were able to generate labels for the overlapping dataset of 9537 images. Since there is no way of distinguishing between compact sources and extended sources based on this information alone, we decided to make a single class composed of compact and single-component extended images. The labels for the classes were generated using the rules as shown in Table 2.11. These sources make up the test set, to assess how well the custom designed network in Lasagne is able to reproduce the labels generated based on the citizen-scientists input. The sources where no class could be assigned were removed, leaving 6966 sources. The remaining images that were not part of the test set of intersected images formed the training and validation set. These numbers are summarised in Table 2.12. It is worth noting that the original set of images again contains an imbalance in the number of sources belonging to each class, where there are fewer compact/single-extended sources and the fewest multiple-component extended sources. This imbalance is compensated by augmenting these classes more.

The architecture used is the three convolutional and two dense layer architecture since this is the overall best-performing architecture. Two datasets are used; the first one using just the original images that contain imbalanced classes, as well as the original and augmented images that contain much more balanced numbers of images in the classes. The parameters used for these two datasets are summarised in Table 2.13.

The results show that when using just the original images, the precision and recall metrics are quite low overall, as shown in the first row of Table 2.14. Upon exploring the individual metrics for the three classes in Table 2.15, the deep learning algorithm is able to identify the compact/single-extended sources effectively, however it struggles more with identifying the two-component extended sources, despite there being more examples of this class to train on. The most likely reason is that the DR1 data contains more information for each source compared to what the deep learning algorithm is trained on. Based on the input from the citizen scientists, it will take a 2 component extended source and divide it into two 1 component sources, depending on the WISE ID status. The deep learning algorithm performs exceptionally poorly with the multiple-component extended images, which is not surprising

given that there are only several hundred examples of this class of images to train and validate on.

In using the augmented images that have been generated to even out the class imbalance, in addition to the original images, all the average metrics are improved, as can be seen in the second row of Table 2.14. Upon examining the individual metrics for each class in Table 2.16, the precision values are improved across all the classes. The recall values are improved for the compact/single-extended class, and are substantially higher for the multiple-component extended class compared to when only the original images are used, however they still remain quite low for this class. The deep learning algorithm is therefore much less precise and sensitive in identifying the images belonging to the multiple-component extended class, when the labels are generated according to citizen scientists input, compared to the other two classes. It does not perform as well in detecting the images that are labelled as multiple-component extended sources, and it also predicts images as being in this class when they are labelled as belonging to another class. A couple of reasons are as follows. There were only on the order of a few hundred (475) original images to train on for images in the multiple-component extended source class, and although they are augmented to generate a set of images that has a roughly the same number compared to the other classes, there are perhaps not enough original examples of the different morphologies that can exist, therefore making the feature space smaller for this class. Additionally, although the multiple-component extended sources in the training and validation set were inspected in an attempt to ensure that the images contain at least three components that are part of the same source, which was the classification scheme used by RGZ users, there were still found to be a substantial number of images that contained source superpositions, upon cross-checking with several optical/IR images. However it is important to keep in mind that the deep learning algorithm was trained to recognise the number of extended components in close proximity, using radio galaxy images only. It should be noted that all multi-component sources, whether they are superpositions or not, belong in the multi-component class.

Presumably, the higher the number of components an image appears to contain, the more likely it is that the images are superpositions of sources. This would explain why the compact/single-extended and two-component extended sources are not affected as much in terms of the precision and recall metrics as the multiple-component extended class. It should further be noted that 77.6% of images belong to the compact/single-component extended class, which explains the overall high classification accuracies in Table 2.14.

The generation of augmented images to even out the imbalance in classes in the original data overall improves the metrics in predicting the labels that are generated using citizen-scientists input.

Table 2.11: Rules by which labels were generated for the DR1 dataset, based on citizen scientists input, to test the best-performing Lasagne convolutional neural network architecture. The number given refers to both the number of components and number of peaks in a given image. For example, the Compact/Single-extended class is defined as having 1 component and 1 peak.

# components and # peaks	Label
1	Compact/Single-extended
2	Two-component extended
≥ 3	Multiple-component extended

Table 2.12: Summary of the numbers of sources used for training, validation and testing of the labels generated from the DR1 data, for both the original (Orig.) and augmented (Aug.) images.

Data	# Orig.	# Orig. + Aug.
DR1	74627	
Final intersected dataset (Test)	6966	
Compact/Single-extended (Train)	4147	14588
Two-component extended (")	10306	14306
Multiple-component extended (")	475	15177

Table 2.13: Chunk sizes and percentage of data used for training, validation and testing for the Lasagne deep learning network in the DR1 cross-check analysis.

	Chunk size	% Train.	% Valid.	% Test
Orig.	1000	59%	9%	32%
Orig.+Aug.	3000	78%	8%	14%

Table 2.14: Validation and Test metrics for the DR1 cross-check analysis.

	Valid. Precision	Recall	F1	Accuracy
Orig.	89.7%	58.7%	58.2%	86.4%
Orig.+Aug.	90.6%	90.6%	90.5%	90.7%
	Test Precision	Recall	F1	Accuracy
Orig.	75.6%	62.6%	61.6%	92.8%
Orig.+Aug.	79.6%	81.6%	80.6%	94.8%

Table 2.15: Individual precision and recall values computed from the confusion matrix for the DR1 test set of 6966 images, when training on just the original images.

	Precision	Recall
Compact/Single-extended	97.2%	95.0%
Two-component extended	79.5%	90.7%
Multiple-component extended	50.0%	2.1%

Table 2.16: Individual precision and recall values computed from the confusion matrix for the DR1 test set of 6966 images, when training on both the original and augmented images.

	Precision	Recall
Compact/Single-extended	97.5%	96.9%
Two-component extended	88.0%	89.5%
Multiple-component extended	53.4%	58.5%

2.6 Conclusions

This is a methods paper that explored the use of deep neural networks for classifying compact and various classes of extended sources in radio astronomical data. We have found an optimal set of parameters obtained from examining the two-class problem of distinguishing between two well-defined classes of data composed of compact and multiple-component extended sources, and applied this to a classification scenario involving more classes, and have shown that the classification accuracies remain high without excessive overfitting. The results were cross-checked on the Radio Galaxy Zoo DR1 dataset, where the generation of augmented images in order to address the class imbalance highly influenced the accuracies to predict the labels generated based on the citizen scientists input. However, the predictions for the multiple-component extended class remained poor, most likely because this dataset contained the fewest number of original images to train on, and did not have the additional information of which components made up a radio source and how many peaks were contained in the source, which was the additional information provided in the DR1 dataset.

The first part of the results explored various architectures and identified the optimal parameters for distinguishing between the two morphological extremes of compact and multiple-component extended sources. We found that the three convolutional and two dense layer architecture using the original and augmented images with no sigma clipping produced the maximal accuracy of 97.4% for the two-class problem, which is significantly better compared to using just the original images with the same architecture. Although the equivalent architecture with sigma-clipping produced an accuracy in the same range, the difference between the training and validation loss was greater. A better model is ensured if the training and validation losses are closer together. The largest influence of performance other than the model architecture was to use a relatively large chunk size, since the more examples that are seen simultaneously, the better the estimate can be for adjusting the weights to achieve a lowered cost function. This is where the use of augmented data is useful, as it allows one to use a larger chunk size. Another important impact on the performance of the deep neural network is to use quite a small learning rate at the start and make it smaller by a factor of 10 at certain points during training, and using a small batch size of 8 samples.

When training deep neural networks with a large enough number of images, removing noise

through the use of sigma clipping appears to offer no significant benefit. Given there is an adequate number of images belonging to the available classes in question, with varying levels of noise, the deep learning network can learn these properties and become robust to them.

Using the knowledge gained from the factors that influence the performance of the classifier in the two-class problem, we assumed that the setup would perform similarly for distinguishing between an additional two classes of images. It is unclear what the effect would have been, had two classes been chosen that were not extreme examples of morphologies. For the four-class problem of distinguishing between compact, single, two-component and multiple-component extended sources, and using the three convolutional and two dense layer setup with original and augmented images, we were able to achieve a classification accuracy of 93.5%. The fact that the compact and single-component extended sources are both chosen from where PyBDSF has detected one component, and that the deep learning algorithm is able to achieve high precision and recall values for these two classes, means that the deep learning algorithm is doing more that just counting the number of components in the images.

Both the two-class and four-class problems contain different numbers of original images in each class. This did not appear to dramatically affect the performance of the classifier when using the original set of images in the two-class problem, most likely because the minority set of images was comprised of compact sources that have a very specific morphology, and the sources are almost always found the in the centre of the image.

It is worth noting that at least 44% of images in the multiple-component extended class in the two-class problem appeared to contain superpositions, or fewer than three components. Although we attempted to remove these images in the four-class problem by manually selecting the sources, a substantial number of images with superpositions remained, upon cross-checking with several optical/IR images. However, the deep learning algorithm was trained to identify extended components in close proximity in a radio galaxy image, so it is still making the correct decisions in determining class membership based on using the image data alone.

The other classes explored apart from compact sources display a much richer variety of morphologies, which is why it is important to augment those images much more in comparison to the compact sources. Roughly equal augmented datasets were generated for the extended source classes in the four-class problem, to make up for the class imbalance present in the original images. This was especially important for the DR1 analysis, where the deep learning algorithm was much better able to predict the labels generated based on citizen scientists input when the augmented data was used in addition to the original data, to compensate for uneven classes. Although the precision and recall values for the compact/single-component extended sources is quite high, it is possible to use linear regression and simple positional matches to identify such sources. The metrics were moderately high for the two-component extended sources. The deep learning algorithm however struggles more to identify the multiple-

component sources when the labels are generated using input from the citizen-scientists, as is evidenced from the poorer precision and recall values for this class of images. This indicates the need for both more original images and labels with higher-level information from citizen scientists to make up the training and validation set, in order to predict these sources more accurately. The value in using both data from the RGZ as well as the help of computer algorithms is the ability to connect discrete individual components that may be associated with a source.

The first example of using convolutional neural networks to classify radio morphologies was in Aniyan & Thorat (2017), where they choose a couple of hundred examples of FRI, FRII and Bent-tailed galaxy morphologies, perform sigma clipping, apply a high amount of augmentation, and build a fusion classifier to combine the results for the three classes. However, the authors run into problems of substantial overfitting, due to not using enough examples of different varieties of FRI, FRII and Bent-tailed classes. An earlier study using an unsupervised learning approach consisting of Kohonen maps has shown that when categorising radio galaxies into FRI and FRII type sources, sigma clipping and other pre-processing may be necessary (Polsterer et al., 2016). In contrast, the current work has shown that with enough examples of broad classes of radio galaxy morphologies, it appears that pre-processing and noise removal through sigma clipping does not offer a significant advantage and that it is possible to classify radio galaxy morphologies into more than two classes using only convolutional networks, without a high level of overfitting.

The use of deep learning networks appears to be very well suited to source classification in radio surveys. However one must keep in mind that the deep learning algorithm will only able to make predictions that are as good as the level or complexity of information that is input into it. When there are a limited number of people to make the classifications, one option to sift through the vast amount of data is to use automated techniques such as PyBDSF or blob-detection algorithms, to assist in providing structure. However these techniques do not always reflect how humans would classify images; they are poorer at making the distinction between images containing superpositions, and images containing sources that have multiple components associated with each other. They can also detect components that a human would identify as noise, as shown in Fig. 2.7. Therefore it is more likely that there will be contaminations in the training set. However, given access to the classifications from an increasing consensus of people that are trained to identify which components belong together in a particular image, the training labels will be more accurate, as will the predictions. Citizen Science projects like RGZ are an excellent way of generating training sets, and appear to have a reliability similar to that of trained astronomers.

When there are few people available to make classifications, there are limitations in the extent of human intervention that can be applied to reduce contamination in the data. In this case,

the results shown indicate that it is better to devote more time in further classifying the images where PyBDSF has detected only up to a few components, as they are less likely to contain superpositions.

The labels generated with the help of algorithms such as PyBDSF are able to attain a certain level of concordance when compared to labels used from citizen-scientists. However, they appear unable yet to replace input from humans, who are able to detect finer-scale structures and subtle aspects of morphologies such as the amount and direction in which the bulges in the edges of radio components are pointing and how far apart they are, that influences whether the components are associated with each other, for a source in question. With the availability of higher-level training labels provided by humans as opposed to the lower-level ones provided by automated techniques such as PyBDSF, deep-learning techniques should exceed the performance of PyBDSF in the future.

Another consideration is the identification of rare sources such as radio relics that make up a small fraction of the overall observed morphologies. Although they are more likely to be found in those images where PyBDSF has detected a multitude of components, these images contain an increasing number of source superpositions, so it is still necessary to have humans to visually inspect the source to see whether they are true relics or not, since PyBDSF has certain ways of grouping the gaussians that are fit to the sources, that may not match how a person would associate them, even when changing parameters that control how the components are grouped.

In future work, we aim to optimise deep neural network setups for more complex morphological classifications and will use them on LOFAR survey data (LOTSS, W. Shimwell et al. (2016)). We will also explore neural networks that perform cross-identification with optical/IR surveys (Norris, 2016).

3 Convolutional vs Capsule networks

The following chapter presents work as it is published by Lukic et al. (2019).

3.1 Introduction

Active Galactic Nuclei (AGN) are energetic, astrophysical sources powered by accretion onto super-massive black holes in galaxies (Padovani, 2017; Fabian, 1999). There are many classes of AGN, where one subset is radio-loud AGN, also known as radio galaxies. The two main ways of classifying radio galaxies is by the properties of optical emission lines (Hine & Longair, 1979) or by the radio morphology of the jets (Bicknell, 1995). The classification of radio galaxy morphology is of research interest in wide-field radio surveys as it correlates with physical properties of the galaxy such as the total power, dust distribution, surrounding environment, and galaxy and cluster evolution (Saripalli, 2012). Radio galaxies can present compact or extended radio morphologies (Miraghaei & Best, 2017) and are often classified into either the FRI (core-bright) or FRII (edge-bright) galaxies (Fanaroff & Riley, 1974). Rarer are hybrid galaxies, which fall in between FRI and FRII galaxies (Gopal-Krishna & Wiita, 2000). There are physical differences between the two classes. The jets of FRIs are less powerful, and are disrupted quite close to the core of the radio galaxy, while the jets of FRII are more powerful and stay relativistic for much larger distances, terminating in a shock (Contopoulos et al., 2015). The transition from FRII to FRI radio galaxies is thought to occur as the jet becomes sub-relativistic (Bicknell, 1994). As the environment plays a large role in the morphology of radio galaxies, it is not unusual for both lobes to have different appearances, especially the FRIs. The dynamics of the ambient gas and the motion of the host galaxy can create tails or distort the jets through ram pressure stripping (Feretti, 2003). Compact radio sources may be either scaled-down (young) versions of the FRI or FRII sources, or may represent a physically distinct population (Baldi et al., 2015).

Radio surveys map ever-increasing numbers of radio sources. The visual classification of such sources becomes increasingly time-consuming and will be completely unfeasible with the rapidly increasing data volumes. Recent and upcoming surveys, such as the LOFAR Two-Metre Sky Survey (LoTSS; Shimwell, T. W. et al., 2017), the Evolutionary Map of the Universe (EMU; Norris et al., 2011) and surveys with the Square Kilometre Array (SKA;

Prandoni & Seymour, 2015) will detect many millions of galaxies. Citizen science projects have been used for classifying astronomical sources, for example in Galaxy Zoo 2 (Willett et al., 2013) and Radio Galaxy Zoo (Banfield et al., 2015). It is also possible to use automated techniques to classify images. Ultimately, these approaches can be used as a training set for machine learning algorithms, in particular deep learning algorithms, when the data is high-dimensional (Wu et al., 2018).

The most prominent wide-area radio surveys, such as the Faint Images of the Radio Sky at Twenty centimetres (FIRST; Becker et al., 1995) and the NRAO VLA Sky Survey (NVSS; Condon et al., 1998), have mostly been conducted at GHz frequencies. In contrast, the LoTSS survey, which is the focus of the current work, has been carried out at 150 MHz with the Low Frequency Array (LOFAR). As such, LOFAR can detect synchrotron emission from older populations of relativistic electrons (which have steeper spectra) found in the extended regions of sources. Furthermore, with its combination of long and short baselines, LoTSS offers both a high angular resolution ($\approx 6''$) for detailed mapping, and a high sensitivity to extended emission.

The cross-identification of radio sources with their optical or infrared hosts helps to associate radio components to sources and to determine properties, such as host galaxy redshift and mass. Previously, cross-identification has been done using visual input from citizen scientists input in Radio Galaxy Zoo (Banfield et al., 2015), and automated methods in cross-identifying radio emission with infrared counterparts have been explored (Alger et al., 2018). In the LoTSS survey (Shimwell, T. W. et al., 2019) the radio sources have been cross-matched with their optical counterparts. For the majority of sources a maximum-likelihood ratio test was adequate because the sources are small and unresolved. For sources that are too large or complex, a visual host identification has been applied (Williams, W. L. et al., 2019).

The first published work on the automated image classification of radio sources using deep learning algorithms was Aniyan & Thorat (2017) where they use a limited number of original radio galaxy images and apply aggressive augmentation to classify sources into FRI, FRII and bent-tailed classes. In previous work, we have shown that it is possible to classify radio sources into four categories based on the number of components belonging to the radio source and produced a classification accuracy of 94.8 % (Lukic et al., 2018) on the Radio Galaxy Zoo (RGZ) DR1 catalogue (Wong et al, in prep). Alhassan et al. (2018) developed a convolutional neural network model to classify FIRST sources into four classes including compact, FRI, FRII and bent-tail sources, achieving overall accuracies >90%. Wu et al. (2018) use regional convolutional networks to localise, recognise and classify sources, the best model obtaining a final mean average precision of 83.4%, using the number of peaks and number of components of a particular radio source. This approach, however, does not always lend itself easily to clear morphological classifications in the FRI or FRII cases because the relative orientations

of components are not taken into account.

The aim of the current work is to compare the performance of two setups of deep learning networks (capsule networks and convolutional networks) in the classification of radio sources. As a data set, we used the first data release of the LoTSS survey (Shimwell, T. W. et al., 2019). Capsule networks are a more recently developed deep learning technique, invented to help preserve the local feature information within an image, which can be degraded in traditional convolutional networks, owing to the pooling operation. In the context of radio galaxies, the orientation and pattern of the emission is important as it determines the morphological classification. The data from the LOFAR LoTSS survey reveals sources in unprecedented detail, therefore one source that had a particular morphology in an earlier survey may be revealed to have a different one when imaged with LOFAR.

This paper is outlined as follows: Section 3.2 describes the LOFAR dataset, including catalogue information and image data as well as how the classifications are generated. Section 3.3 discusses the pre-processing and augmentation applied to the original images. Section 3.4 describes the theory behind the two deep learning approaches explored, namely convolutional neural networks and capsule networks. Section 3.5 explores the performance of different capsule network models against standard convolutional neural network setups, including transfer learning on the LOFAR data, when training on different sets of images. The results are also discussed in Section 3.5. Section 3.6 summarises our overall findings.

3.2 LOFAR HETDEX v1.0 dataset

3.2.1 Source cutouts

The sources in our dataset originate from a 424 square degree region of the HETDEX Spring Field, mapped from the LOFAR Two-metre Sky Survey (LoTSS), and release as Data Release 1 (Shimwell, T. W. et al., 2019). The LoTSS survey detects a total of 325,694 sources where the signal is five times that of the noise and the density of sources is a factor of approximately 10 times higher than the most sensitive existing very wide-area radio-continuum surveys. We use v1.0 of the value-added catalogue for the HETDEX-area data release of LoTSS. The first step in creating the value-added catalogue involved using PyBDSF[1] to produce a radio source catalogue for the field, after which a decision tree was used to further categorise the sources, with details provided in (Williams, W. L. et al., 2019). After filtering the 325,694 sources to only include those classified as resolved leaves 24,096 sources (Shimwell, T. W. et al., 2019). The catalogue also contains 180 columns describing the properties, such as redshift, position etc, of the sources. In order to exclude star-forming galaxies and sources

[1] http://www.astron.nl/citt/pybdsf/

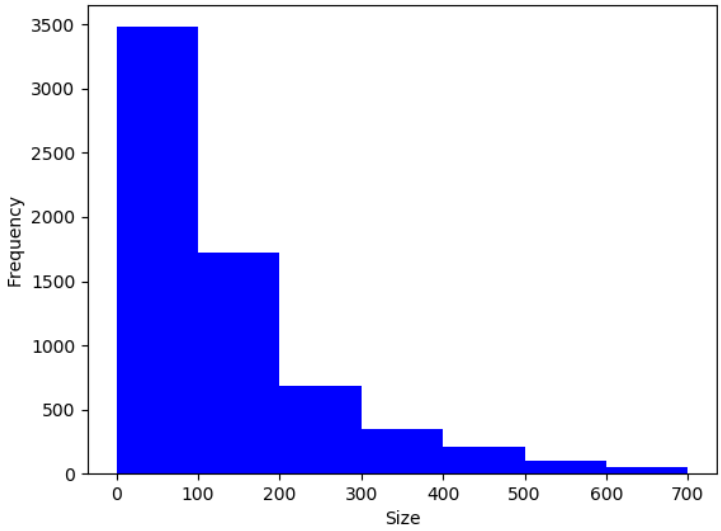

Figure 3.1: Histogram of sizes (in pixels per side) of the filtered cutout images. The total number of images is 6708.

with less certain redshift values, we made use of the AGN subsample of the LoTSS catalogue, derived by Hardcastle et al. (2019) leaving 6708 sources. We note that this is a substantial limitation of the machine learning approach when using radio galaxy image data only, as it is generally not always possible to filter out the star-forming galaxies without the use of additional data at other wavelengths. The source classifications were only available for those 6708 sources classified as AGN and with known redshifts, therefore the analysis is restricted to this set. However, the accurate knowledge of redshift is not strictly required for morphological classification.

Finally, we assume that there is one source per image. Square cutouts of each source are produced from the FITS images, where the cutout size is determined by the catalogued size of the radio source. These range from size (66,66) pixels up to (2342,2342) pixels. The size of the pixels is roughly 1.5x1.5″. Figure 3.1 shows the histogram of the side length in pixels of the images for these 6708 samples.

3.2.2 Classifications

The LoTSS association and cross-identification effort (Williams, W. L. et al., 2019) was a project in which expert astronomers were tasked with characterising the radio emission for sources larger than 15″. Indicated were the locations of the peaks and extents of the emission, and whether there was one or more sources present.

The 6708 source sample (see Section 3.2.1) were classified into 6 classes using an automated technique (Mingo et al. in prep). The 6 classes are Unresolved-1, FRI, FRII, Hybrid-1,

Hybrid-2 and Unresolved-2, all of which are described in further detail as follows. After the host galaxy location had been identified through the LoTSS identification effort (Williams, W. L. et al., 2019), the distances, d1 and d2, were determined as the distances in pixels from the host galaxy to the brightest peaks of emission on both sides of the source (shown with points marked with Y/inverted Y in Figure 3.2). Similarly, Maxd1 and Maxd2 were determined as the maximum extents of the source in each direction (marked with triangles on the plots), out to the masked 4rms limit. A 120 degree aperture cone is used to find those along the direction of d1, d2. The comparison of d1/Maxd1 and d2/Maxd2 is then used to classify the sources. If, on both sides, the peak is less than half of the distance between the position of the host galaxy and the maximum extent of the emission (ie. d1/Maxd1 < 0.5 and d2/Maxd2 < 0.5) then the source is classified as an *FRI*, making up 15% of the total sources. Likewise, if it is more than half of the distance (d1/Maxd1 > 0.5 and d2/Maxd2 > 0.5) then the source is classed as an *FRII*. The *FRIIs* make up 7% of the total sources.

In addition to the FRI and FRII labels, four further labels were defined. *Hybrid-1* and *Hybrid-2* classes refer to sources which show FRI morphology on one side of the source and FRII in the other, with the '1' or '2' reflecting the classification of the brighter of the two sides. The *Hybrid* classes together make up 6% of the sources. *Unresolved-1* sources correspond to those images that have less than 5 pixels of signal above 4rms, making up 22% of the sources. This class is useful as it indicates which images are too noisy to be characterised into a particular class (note that it is different from the Unresolved sources previously discussed, which were based on the extent of the overall radio emission). Finally, the *Unresolved-2* class contains a collection of mostly FRI and FRII sources that were unable to be classified accurately by the automated algorithm as they were too small, which makes up 50% of the sources. Figure 3.2 shows an example image source, demonstrating how the classification labels were generated.

In the current work, we have chosen the Unresolved-1 (henceforth called Unresolved), FRI and FRII classes to evaluate the performance of our deep learning algorithms, as these had the most confident classifications. There are 2901 original images in total, as shown in Table 3.1.

The automated classification technique (Mingo et al. in prep) involved using masked 4rms arrays (where emission below 4rms is removed and potential unassociated emission is masked), rather than the raw FITS data. We define unassociated emission as radio emission which does not appear to belong to the radio source in question. A flood-filling algorithm[2] and masking techniques have additionally been applied in order to identify and use associated structures and consequently remove unassociated emission from the image (Mingo et al. in prep). On the other hand, the current work emphasises using the raw FITS images as the input to the deep learning algorithms, to see if they could be trained to cope with unassociated emission

[2]http://scikit-image.org/docs/dev/api/skimage.measure.html#skimage.measure.label

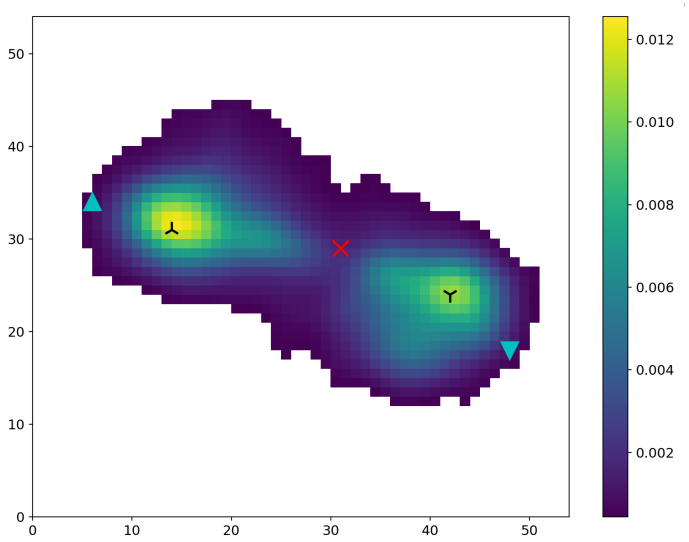

Figure 3.2: The masked array from which classifications are generated. The red cross indicates the position of the optical source, the black Y's indicate the peaks of the emission and the blue triangles indicate the maximum extents of emission. The optical position is calculated from the user's clicks on the LOFAR Two-metre Sky Survey images, or from the maximum likelihood method. The Y's and blue triangles are outputs from the automated classification code.

and unfiltered noise. After visual inspection we found there were approximately 1% of images containing potentially unassociated emission, whereas the majority of the images contain varying levels of noise.

In cases where the calibration did not perform as expected, the source will not be de-convolved accurately, causing flux leakage. This could result in the source being misclassified, leading to label errors. After inspecting several batches of images, we estimated the amount of labels containing errors to be less than 6%, when considering both FRIs and FRIIs. Since larger sources are easier to classify, there is a decreased likelihood that they will be mislabeled, therefore the size of the source affects the presence of noisy labels. However, pre-filtering is applied to ensure the effect is not very large.

Figure 3.3 shows typical examples of source types across the three classes. It is evident that there are varying levels of noise present in the images, presenting the largest hindrance to the deep learning algorithms' ability to classify the sources accurately. One of the aims of the current work is to see how well the algorithms can classify the sources in the presence of such undesirable features, present in the original radio images (FITS files). We also compare the results obtained when using the masked 4rms clipped arrays (see Section 3.5.3), where emission below 4rms is removed and potential unassociated emission is masked.

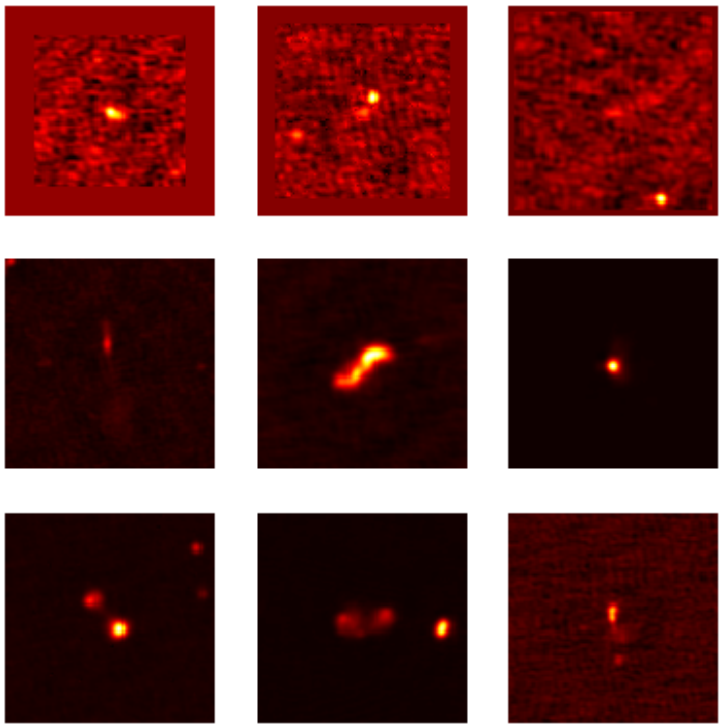

Figure 3.3: Showing morphology samples of the FITS cutouts when converted to png images using the 'hot' colormap. The top row shows the 'Unresolved class', middle row shows the FRI class, bottom row shows FRII. There are varying levels of noise and the occasional potentially unassociated emission present in the images.

3.3 Methods

We use the radio galaxy image FITS cutouts from version 1.0 of the LoTSS DR1 value-added catalogue (Williams, W. L. et al., 2019). The extended source identifications do not differ from the final version to a large extent.

3.3.1 Pre-processing

Since the size of each cutout varies, they first need to be made the same size. The FITS images have been resized to (200,200) pixels, where the smaller images have been padded with zeros around the edges, and the larger images have been downsampled, using bicubic interpolation. The sizes of the arrays varies across all three classes. Following this, the images are centred on the position of the optical source, ensuring its position is at (100,100). We crop to the inner (100,100) pixel part of the image as the source is likely to be contained in this interval and to reduce the amount of data input into the network. The pixel values, representing brightness in mJy/beam were normalised by dividing by the maximum value in each image, therefore the values are contained within the [0,1] range. The images are taken at 150MHz. We apply the 'hot' colormap from the python matplotlib library, which converts the images from a single channel numpy array to a RGB png image. This is done by assigning a color (RGB vector) according to the value in the single channel array. For example, values close to 1 are bright yellow in the 'hot' colormap scheme, therefore (r,g,b) $\approx$ (1,1,0.99). The conversions to the RGB vector are provided[3]. The conversion is done to make the arrays more amenable to deep learning analysis and has no bearing on the flux values. The number of sources in each class is given in Table 3.1.

Cropping the images to (100,100) pixels, instead of using the originally resized images of (200,200) pixels, reduces the impact of radio emission that is potentially unassociated with the main source in the centre. We have also experimented with using central sizes other than (100,100) pixels, however they resulted in worsened performance metrics. Smaller images tended to have some associated emission truncated, whereas larger images encapsulated more unassociated emission. The cropping still preserved the general noise characteristics surrounding the source.

The upsizing of images should not have any detrimental effects on image quality, however the downsizing may cause effects such as as slight distortion of the radio emission due to the interpolation.

[3] y =(0,0.36): (r,g,b) $\approx$ (x=y/0.36,0,0)
y = (0.36,0.74): (r,g,b) $\approx$ (1,x=(y-0.37)/0.37,0)
y = (0.74,1): (r,g,b) $\approx$ (1,1,x=(y-0.75)/0.25)

Table 3.1: The number of original and augmented sources, divided into training and testing sets. The percentage of samples in each class is also given for the test set. Since only original images should be used in the test set, the augmented images are used for training only.

Class	# Orig.(Train)	# Orig.(Test)	# Aug.	# Total
Unres.	1156	301 (50.2%)	4371	5828
FRI	765	219 (36.5%)	5904	6888
FRII	380	80 (13.3%)	2760	3220
Total	2301	600	13035	15936

3.3.2 Image augmentation

Deep learning algorithms generally require large numbers of labeled images in order to make predictions more successfully and to reduce the effect of overfitting, in which the algorithm memorises the training samples and therefore the model fails to generalise on an independent dataset. More images can be generated artificially, by performing simple transformations to the original data (Krizhevsky et al., 2012). As such, we apply translation, rotation and flipping to generate more images. In using translation, we initially use a random number that shifts the image between 0 and 20 pixels in any of the four directions, using the condition that if such a translation moves the brightest pixel out of the image, the translation is reduced to 10% of the original value. This is to reduce the possibility that part of a radio component will be shifted out of the image. The images have been rotated randomly in multiples of 90 degrees only in order to avoid interpolation artefacts. We note that since there is a limited range of rotation applied, it is not enough to ensure complete rotational invariance in our models. Both horizontal and vertical flipping has been applied at random. The augmentation of the FRI and FRII sources has been done keeping their overall proportions similar in number to the original dataset as this resulted in improved performance. The number of original and augmented images used in the current work is given in Table 3.1. Image augmentation is applied on both the original LOFAR images, as well as the masked 4rms arrays.

3.4 Deep Learning algorithms

The most successful class of machine learning methods in the context of extracting information from high-dimensional data is deep learning, which has achieved unprecedented performance in a variety of domains such as image recognition, sentiment analysis and genomics (LeCun et al., 2015). Their ability to learn multiple representations of data lies in their stacked layer architecture. The most commonly used implementation of deep learning has to date been convolutional neural networks. However, more recent advances were made in addressing the lack of rotational invariance in convolutional neural networks through the development of

capsule networks.

3.4.1 Convolutional Neural Networks

Neural networks and deep learning algorithms are generally trained using the backpropagation algorithm, where a gradient descent optimisation algorithm is used to minimise the error between the predictions of the network and the input labels by calculating the gradients and adjusting the weights accordingly (Rumelhart et al., 1986). A deep fully connected neural network becomes time-consuming and computationally intensive to train. Convolutional neural networks employ smaller sized filters that scan across the image and extract features, which greatly reduces the dimensionality compared to using adjacent layers of fully connected neurons and enforces parameter sharing and therefore translational invariance (Karpathy, 2016). Spatial pooling layers are typically inserted between at least one convolutional layer which further reduces the dimensionality of features propagated through the network. In max pooling, the maximum value of a certain region of the image is output into the next layer. However, since the pooling operation summarises the information in a local part of the image, the global feature information within the image tends to degrade.

3.4.2 Capsule networks

Capsule networks (Sabour et al., 2017) have been developed to preserve the relative locations of features within images and thus model the hierarchical relationships better. Whereas traditional neural networks output a single activation value, capsule networks are higher dimensional and output a vector representing a group of parameters such as orientation, skew, thickness etc., depending on the input. The overall length of these vectors give the probability that the entity exists. Capsule networks have achieved state of the art performance on the MNIST dataset (Lecun et al., 1998) without data augmentation (Xi et al., 2017).

In the context of radio galaxy classification, capsule networks should be able to preserve the emission pattern features over a large spatial extent, given an adequate training set size.

Below we summarize the theory behind capsule networks but see Sabour et al. (2017) for a detailed description. For all capsules above the first layer of capsules, the input to a capsule s_j is a weighted sum over all prediction vectors from the capsules in the layer below, given by multiplying the coupling coefficients c_{ij} by the output u_i of a capsule in the layer below by a weight matrix W_{ij}, as shown in Equation 3.1

$$s_j = \sum_i c_{ij} W_{ij} u_i \tag{3.1}$$

The coupling coefficients c_{ij} are determined by a routing softmax function given by Equation 3.2

$$c_{ij} = \frac{e^{b_{ij}}}{\sum_k e^{b_{ik}}} \tag{3.2}$$

The coupling coefficient c_{ij} is the level of agreement between the predicted output of capsules in a layer, to their parent capsules in the layer above. b_{ij} gives the log prior probabilities that capsule i should be coupled to capsule j.

The vector length is calculated as shown in Equation (3.3)

$$v_j = \frac{||s_j||^2}{1 + ||s_j||^2} \frac{s_j}{||s_j||}, \tag{3.3}$$

where v_j is the vector output of capsule j and s_j is its total input. This output gives the probability that a specific property exists in the input to the capsule, that is represented by the capsule. The vector output v_j is an activation function, that is also referred to as a squashing function as it shrinks short vectors to near zero if a property is not present in the capsule, and long vectors to lengths close to 1 if the property exists.

The agreement a_{ij} for updating log probabilities and coupling coefficients is given by Equation (3.4)

$$a_{ij} = v_j . W_{ij} u_i \tag{3.4}$$

A margin loss function is used in order to determine whether a radio galaxy of a particular class is present, which has the form given by Equation (3.5):

$$L_k = T_k \, max(0, m^+ - ||v_k||)^2 + \lambda(1 - T_k) \, max(0, ||v_k|| - m^-)^2, \tag{3.5}$$

where $T_k = 1$ if a radio galaxy of class k is present and $m^+ = 0.9$ and $m^- = 0.1$, to ensure that the vector length remains within reasonable bounds. The λ down-weighting function is introduced for numerical stability and suggested to be set at 0.5.

The mean squared error difference between the reconstructed image from the decoder (the part of the Capsule network after LabelCaps) and the input image acts as a regulariser for the capsule network, such that near-perfect reconstructions will produce a near-zero error and poor reconstructions will produce a large error. The reconstruction loss is scaled down by 0.0005 so it does not dominate the margin loss during training, and the coefficient for the default model is designed for the MNIST digits which have an image size of 28x28, thus the coefficient is worked out to be $0.0005 \times 28 \times 28 = 0.392$.

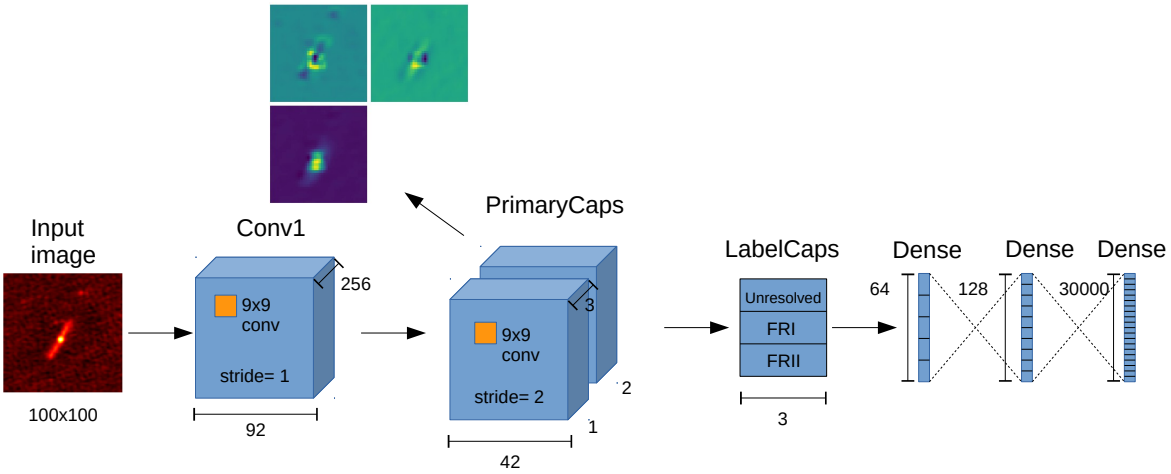

Figure 3.4: The default architecture for CapsNet, using three classes. The input to the network is a 100x100x3 image. The encoder is the part of the network that encapsulates the convolutional layer up to and including the LabelCaps layer. The decoder refers to the final three dense layers. An example of features detected by the PrimaryCaps layer prior to reshaping and squashing is shown, for the given input image. There is a small amount of extended emission to the top right of the image that appears to be unassociated with the main source in the centre, which the capsule network preserves, suggesting that it is not robust to potential unassociated sources. Additionally, the feature maps appear to show extra distortion in the core of the source.

Table 3.2: Showing architecture for the default capsule network model

Layer	Output shape	# Params
Input_1	(None, 100, 100, 3)	0
conv2d	(None, 92, 92, 256)	62464
PrimaryCap_conv2d	(None, 42, 42, 6)	124422
PrimaryCap_reshape	(None, 3528, 3)	-
PrimaryCap_squash	(None, 3528, 3)	-
LabelCaps	(None, 3, 3)	95256
Input_2	(None, 3)	-
mask	(None, 9)	-
capsnet	(None, 3)	-
decoder	(None, 100, 100, 3)	3878960
Total		4,161,102

3.4.3 Deep learning parameters

There are several deep learning implementations currently available for use. The present work uses Keras[4] with the TensorFlow[5] backend and Python version 2.7.14.

We use the Adam optimiser (Kingma & Ba, 2014) with the default learning rate of 0.001. In order to keep more parameters the same between the models, both the convolutional and capsule network models are trained using a batch size of 100, for 50 epochs.

The deep-learning task is a multi-classification problem, where the models output a 3-dimensional vector representing the probability that the object belongs to each class. The predicted class is chosen as the one with the largest probability value. As the probabilities are independent, there is no constraint that they need to add to unity.

The models are trained using CPUs from 27 available Intel XEON CPU nodes with six available cores per node on a computing cluster at the University of Hamburg.

ConvNet-4 parameters

We use an architecture of two pairs of stacked convolutional layers with pooling layers in between, as shown in Figure 3.5, with parameters given in Table 3.3. This model is referred to as ConvNet-4. Using two adjacent convolutional layers with smaller filter sizes obtained improved results compared to using a single larger convolutional layer, and also reduced the number of parameters (Simonyan & Zisserman, 2014). We use the categorical cross-entropy cost function[6] and 16 filters of size 5x5 across all layers, as well as the default learning rate decay of 0. In order to reduce the effect of overfitting, dropout layers are used. A dropout value of 0.25 is used after each pair of convolutional layers, and a value of 0.5 in between the dense layers. A penalty term is added to the cost function using L2 regularisation (Ng, 2004) in the first dense layer. All the convolutional layers use the ReLU activation function (Nair & Hinton, 2010), and the softmax activation function at the final layer where classifications are made. There are 5,022,467 trainable parameters in total.

ConvNet-8 parameters

In order to investigate the performance for more complex convolutional networks, we can add additional layers. The ConvNet-8 model uses an architecture of four pairs of stacked convolutional layers with pooling layers in between. There are also an increasing number

[4]https://keras.io/preprocessing/image/
[5]https://www.tensorflow.org
[6]https://keras.io/losses/#categorical_crossentropy

of feature maps with each subsequent double stacking of convolutional layers, as shown in Table 3.4. The architecture also uses smaller feature maps of size 3x3. There are 7,446,259 trainable parameters in total.

CapsNet parameters

Finally, we explore several variations of capsule network models. We downloaded the original CapsuleNet[7] code implemented in Keras that was built for the MNIST dataset (Sabour et al., 2017), and modified the code to use our datasets, vary the models from the original architecture and to calculate the metrics. The original architecture contains approximately 58M parameters, which is more than 14x the number of parameters as for the ConvNet-4 model. We therefore simplified the architecture to one having just over 4M parameters, and refer to this as the default model. The original CapsuleNet model is simplified in order to have the same order of magnitude as the parameters in the ConvNets and to help prevent overfitting.

The default architecture of CapsNet and decoder is illustrated in Figure 3.4 and the number of parameters is given in Table 3.2. In essence it is comprised of an encoder and decoder. The encoder consists of a convolutional layer, which extracts features in the image, which are then input into the first capsule layer (PrimaryCaps), whose function is to take the 256x9x9 output of the convolutional layer and produce combinations of the detected features. The output of the PrimaryCaps layer is then sent to the LabelCaps layer, which produces one 3D capsule for each of the three radio galaxy classes. Routing is used between the PrimaryCaps layer and the LabelCaps layer such that the level of agreement of feature existence can be quantified and contribute to the vector length of the capsule. The decoder refers to the part of the network after the LabelCaps layer (the three dense layers at the end). There are 4,161,102 free parameters in the default CapsNet model.

We use 256 filters in the first convolutional layer, a filter size of 9 in both the first Convolutional layer and PrimaryCaps layer, 3 capsules in the PrimaryCaps and LabelCaps layers, 2 channels in the PrimaryCaps and the decoder contains (64,128) nodes. We use the default setup of three routings and a learning rate of 0.001 with a decay of 0.9. The first convolutional layer uses the ReLU activation function. CapsNet has image augmentation built into the training of the model, which we disable in order to use our augmentation technique, that allows more control over which classes get augmented and the type of transformations that are used. For the default CapsNet model, there are 4,161,102 parameters, which is a very similar number of parameters that was used for ConvNet-4.

In addition to the default CapsNet model, we experiment with two other CapsNet models.

[7] https://github.com/XifengGuo/CapsNet-Keras/blob/master/capsulenet.py

In the first of these models (Inc. filtersize), we set the filter size to 24 and 18 in the first Convolutional layer and PrimaryCaps layer respectively and slide the filters across using a stride of 4 in the convolutional layer. The inc. filtersize model has 4,819,470 parameters. In the second model (Inc. decoder), we increase the complexity of the decoder to (128,256) nodes in the dense layers and the loss function of the decoder weight is increased from 0.392 to 5 respectively. The weight is calculated by taking the scaled-down reconstruction loss and multiplying it by the size of the images $0.0005 \times 100 \times 100 = 5$. There are 8,026,446 parameters in the inc. decoder model.

We chose to increase the filters from a size of 9 pixels in the inc. filtersize model because the original filter sizes that were designed for the MNIST image sizes of (28,28) pixels are likely too small compared to what would be needed for our (100,100) pixel images. We also experimented with increasing the number of nodes and weight loss of the decoder in the inc. decoder model to better account for the noise and potential unassociated emission in the dataset, as well as more variability in and between classes.

3.5 Results

Due to the inherent stochasticity of training deep learning models, each run can produce slightly different results. We therefore train each model five times. The training data is also shuffled for each run to ensure there is no correlation between subsequent samples. There are several classification metrics that can help evaluate the performance of a classifier. In imbalanced class problems, the classification accuracy alone has several weaknesses in distinguishing between the performance of models (Hossin & M.N, 2015). The precision, recall and F1 scores are more informative measures of performance compared to using the classification accuracy. Precision refers to the fraction of true positives returned among all returned positive instances, recall is the fraction of true positives that are identified correctly, which also gives an indication of the sensitivity of the classifier. The F1 score is the harmonic mean of precision and recall, and can be interpreted as the average of the precision and recall values. The accuracy is the total proportion of correct predictions. Precision, recall, F1 score and accuracy are defined in Eqs. (3.6)-(3.9).

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3.6}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3.7}$$

$$\text{F1\_score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.8}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}, \tag{3.9}$$

where TP refers to the true positives, FP refers to the false positives and FN refers to false negatives. A true positive is when the prediction matches the label. A false positive is when the positive class is incorrectly predicted. A false negative is when the positive class is predicted to be in another class.

We also calculate the 95% confidence interval using the mean and standard deviation of the metrics to account for the variability in performance across the runs. We declare a model to be statistically significantly better than another model if the mean of its metrics is higher than the 95% confidence interval of the other models metrics. In order to ensure a fair comparison, the same training and testing sets were used for the ConvNet and CapsNet architectures.

The same set of data is used for both validation and testing when running the models, with the exception of the application of early stopping (results shown in Section 3.5.4). When early stopping is used, the validation data is used to determine when to stop the training. Otherwise, the use of the same dataset for validation and testing is of no consequence, as the weights that are modified using the training set are applied to the validation/test set to calculate the loss. No adjustment is made to the weights using the validation set. At the conclusion of training, the final weights are applied to the validation/test set and the metrics are calculated.

Section 3.5.1 of the results shows the classification metrics across the two deep learning techniques when using the original data only, with 2301 (79%) samples for training, and 600 (21%) samples for both validation and testing. The fraction of samples in each class is given in Table 3.1 for the test set. Section 3.5.2 makes use of augmented images in addition to the original images and Section 3.5.3 explores the effects when the 4rms sigma-clipped data is used.

Table 3.3: ConvNet-4 architecture. A filter size of 5 is used in the convolutional layers.

Layer	Output shape	# Params
Input	(None, 100, 100, 3)	0
conv2d	(None, 100, 100, 16)	1216
conv2d	(None, 100, 100, 16)	6416
maxpool2d	(None, 50, 50, 16)	-
dropout	(None, 50, 50, 16)	-
conv2d	(None, 50, 50, 16)	6416
conv2d	(None, 50, 50, 16)	6416
maxpool2d	(None, 25, 25, 16)	-
dropout	(None, 25, 25, 16)	-
flatten	(None, 10000)	-
dense	(None, 500)	5000500
dropout	(None, 500)	-
dense	(None, 3)	1503
Total		5,022,467

3.5.1 LOFAR original images

ConvNet-4 and ConvNet-8 models

We use the ConvNet-4 and ConvNet-8 models on the original 2901 images from LOFAR, which have been classified into Unresolved, FRI and FRII sources. The results are shown in Table 3.5 and Table 3.6. Each epoch consisting of 2301 training samples takes approximately 32 and 66 seconds to train for ConvNet-4 and ConvNet-8 respectively.

The models perform the best in recovering the images in the Unresolved class, which could be due to the images being generally noisier and the sources smaller, compared to the other images. The recovery of FRIIs is poorer however compared to the FRIs. This may be because there are fewer examples of images in this class (460 FRIIs compared to 984 FRIs). Although it can be argued that the morphological diversity is greater for the FRI class as they can be straight, bent, or one-sided with a peak at one end, FRIIs contain lobes that may or may not be connected, therefore the source can contain either one or two components. We have experimented with using different weights for the classes, giving proportionally greater weights for the FRIIs such that wrong predictions are penalised more, however the performance remained the same as before, across all classes. The recall (accuracy) tends to be higher compared to precision for the FRIs, whereas it is lower compared to precision for the FRIIs. This is likely due to it being easier to recover sources containing emission that is more concentrated in one place (in the case of the FRIs), compared to emission that is further apart.

Examples of detected features in the ConvNet-4 model at the output of the second and fourth convolutional layers, after max pooling are shown in Figure 3.5. The training and validation

Table 3.4: ConvNet-8 architecture. A filter size of 3 is used in the convolutional layers.

Layer	Output shape	# Params
Input	(None, 100, 100, 3)	0
conv2d	(None, 100, 100, 32)	896
conv2d	(None, 100, 100, 32)	9248
maxpool2d	(None, 50, 50, 32)	-
dropout	(None, 50, 50, 32)	-
conv2d	(None, 50, 50, 64)	18496
conv2d	(None, 50, 50, 64)	36928
maxpool2d	(None, 25, 25, 64)	-
dropout	(None, 25, 25, 64)	-
conv2d	(None, 25, 25, 128)	73856
conv2d	(None, 25, 25, 128)	147584
maxpool2d	(None, 13, 13, 128)	-
dropout	(None, 13, 13, 128)	-
conv2d	(None, 13, 13, 256)	295168
conv2d	(None, 13, 13, 256)	590080
maxpool2d	(None, 7, 7, 256)	-
dropout	(None, 7, 7, 256)	-
flatten	(None, 12544)	-
dense	(None, 500)	6272500
dropout	(None, 500)	-
dense	(None, 3)	1503
Total		7,446,259

Table 3.5: The average metrics (in percentages) across each of the classes in (1) the original LOFAR dataset, (2) the original and augmented dataset, (3) the original 4rms clipped dataset, and (4) the original and augmented 4rms clipped dataset for the ConvNet-4 model. Five runs were done in total, using 600 samples in the test set.

Class	Precision	Recall	F1 score	Accuracy
(1)				
Unres.	95.7 ± 0.9	96.7 ± 1.4	96.2 ± 0.9	95.9 ± 0.9
FRI	86.2 ± 2.4	86.8 ± 1.1	86.5 ± 1.0	89.9 ± 0.9
FRII	68.0 ± 1.1	63.5 ± 2.1	65.6 ± 1.0	90.9 ± 0.2
Avg.	88.5 ± 0.8	88.7 ± 0.8	88.6 ± 0.9	93.1 ± 0.8
(2)				
Unres.	98.1 ± 0.4	98.2 ± 0.5	98.1 ± 0.4	98.0 ± 0.4
FRI	92.3 ± 0.9	93.3 ± 1.3	92.3 ± 0.2	94.2 ± 0.1
FRII	80.9 ± 2.0	75.2 ± 4.9	77.8 ± 1.9	94.2 ± 0.2
Avg.	93.3 ± 0.2	93.4 ± 0.2	93.3 ± 0.2	96.2 ± 0.2
(3)				
Unres.	97.9 ± 0.3	98.1 ± 0.5	98.0 ± 0.2	97.9 ± 0.2
FRI	90.4 ± 0.7	90.0 ± 0.6	90.2 ± 0.4	92.8 ± 0.3
FRII	72.1 ± 0.6	72.2 ± 1.6	72.1 ± 0.8	92.5 ± 0.2
Avg.	91.8 ± 0.2	91.9 ± 0.3	91.8 ± 0.3	95.5 ± 0.2
(4)				
Unres.	98.7 ± 0.6	99.7 ± 0.2	99.2 ± 0.2	99.2 ± 0.2
FRI	91.5 ± 0.9	94.9 ± 0.6	93.1 ± 0.4	94.9 ± 0.3
FRII	88.1 ± 1.3	75.5 ± 2.3	81.3 ± 1.4	95.3 ± 0.3
Avg.	94.9 ± 0.2	94.7 ± 0.3	94.7 ± 0.2	97.3 ± 0.1

Table 3.6: The average metrics (in percentages) across each of the classes in (1) the original LOFAR dataset, (2) the original and augmented dataset, and (3) the original 4rms clipped dataset, and (4) the original and augmented 4rms clipped dataset, for the ConvNet-8 model. Five runs were done in total, using 600 samples in the test set.

Class	Precision	Recall	F1 score	Accuracy
(1)				
Unres.	96.6 ± 1.1	98.8 ± 0.3	97.7 ± 0.5	97.5 ± 0.6
FRI	88.7 ± 1.0	90.4 ± 0.7	89.6 ± 0.5	92.2 ± 0.4
FRII	75.2 ± 4.1	64.5 ± 2.8	69.3 ± 1.1	92.3 ± 0.4
Avg.	90.9 ± 0.4	91.2 ± 0.4	90.9 ± 0.5	94.9 ± 0.4
(2)				
Unres.	98.2 ± 0.7	98.4 ± 0.2	98.3 ± 0.3	98.2 ± 0.3
FRI	92.5 ± 0.6	94.0 ± 0.5	93.2 ± 0.4	95.0 ± 0.3
FRII	84.5 ± 1.9	80.0 ± 1.0	82.2 ± 1.1	95.3 ± 0.3
Avg.	94.3 ± 0.2	94.3 ± 0.2	94.3 ± 0.2	96.7 ± 0.1
(3)				
Unres.	99.6 ± 0.3	98.8 ± 1.0	99.2 ± 0.5	99.1 ± 0.5
FRI	92.7 ± 1.0	93.4 ± 3.3	93.0 ± 2.1	95.2 ± 1.7
FRII	83.4 ± 9.3	83.4 ± 2.8	83.1 ± 5.8	95.2 ± 1.8
Avg.	95.0 ± 1.6	94.9 ± 1.8	94.9 ± 1.7	97.3 ± 1.0
(4)				
Unres.	99.6 ± 0.1	99.1 ± 0.4	99.3 ± 0.2	99.3 ± 0.2
FRI	94.4 ± 0.4	95.2 ± 0.7	94.8 ± 0.4	96.2 ± 0.3
FRII	86.0 ± 1.0	85.8 ± 1.5	85.9 ± 0.6	96.2 ± 0.1
Avg.	96.0 ± 0.2	95.9 ± 0.2	95.9 ± 0.2	97.9 ± 0.2

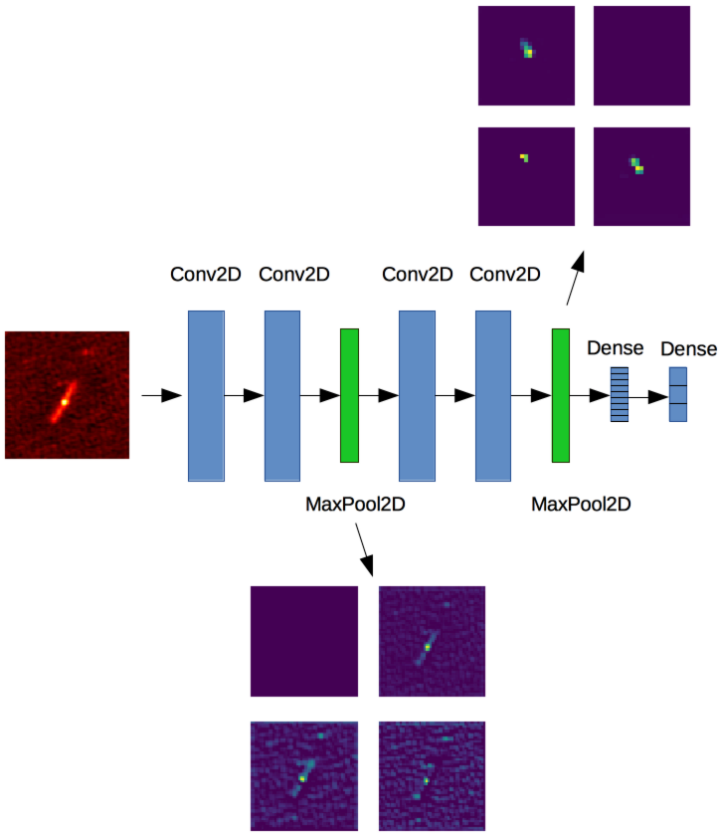

Figure 3.5: The ConvNet-4 architecture. The input to the network is a 100x100x3 image. Showing an example input image with features detected at the second and fourth convolutional layers, after pooling, at the end of training (50 epochs). We show 4 feature maps for each of the two outputs.

losses for a single run with the ConvNet-4 architecture are shown in Figure 3.6.

The use of a more complex architecture (ConvNet-8 compared to ConvNet-4) appears to improve the classification metrics (Avg. Recall = 91.2 compared to 88.7 respectively).

CapsNet model

Each epoch consisting of 2301 training samples takes approximately 3.4 minutes for the default model, 14 seconds for the inc. filtersize model and 3.5 minutes for the inc. decoder model. The faster time for the inc. filtersize model is due to the fact that the feature maps are moved across the image by 4 pixels (stride of 4) in the first convolutional layer as opposed to using a stride of 1, therefore the feature maps are able to scan through the image faster.

Examples of detected features at the PrimaryCaps layer, prior to the reshape and squashing functions are shown in Figure 3.4 for the default model. Figure 3.7 shows the training and validation loss curve for the default model. Table 3.7 shows that the default model attains

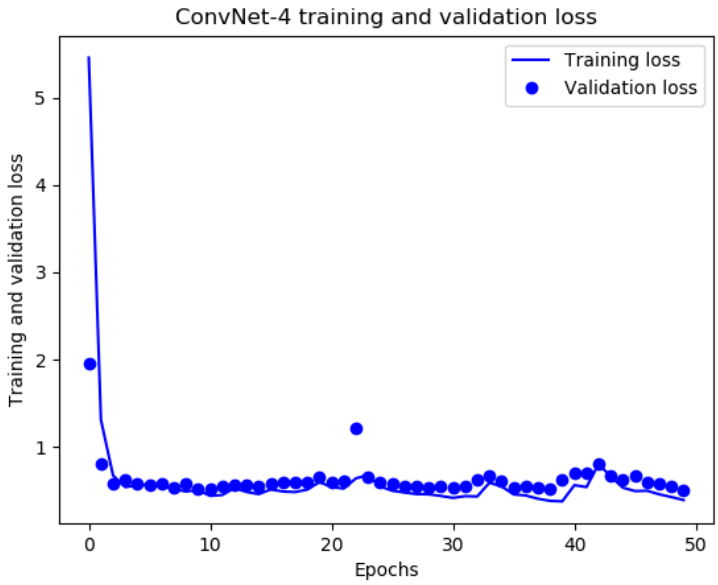

Figure 3.6: The training and validation losses for a single run with the ConvNet-4 architecture using the cross-entropy loss, with 2301 (79%) samples for training and 600 (21%) samples for testing.

Table 3.7: The average metrics (in percentages) across each of the classes in the original LOFAR dataset, for the default CapsNet model. Five runs were done in total, using 600 samples in the test set.

Class	Precision	Recall	F1 score	Accuracy
(1)				
Unres.	92.7 ± 1.4	95.7 ± 0.7	94.2 ± 1.0	93.4 ± 1.2
FRI	78.3 ± 3.1	87.7 ± 1.6	82.7 ± 1.1	86.3 ± 1.3
FRII	66.6 ± 5.1	35.0 ± 13.0	43.1 ± 12.7	88.2 ± 0.8
Avg.	84.0 ± 1.3	84.7 ± 1.5	83.2 ± 2.6	90.1 ± 1.2
(2)				
Unres.	96.4 ± 0.6	96.4 ± 0.9	96.4 ± 0.2	96.1 ± 0.2
FRI	85.5 ± 1.4	90.2 ± 0.2	87.8 ± 0.7	90.7 ± 0.6
FRII	75.8 ± 1.8	64.2 ± 0.5	69.6 ± 1.4	92.3 ± 0.4
Avg.	89.7 ± 0.5	89.9 ± 0.5	89.7 ± 0.5	93.7 ± 0.3
(3)				
Unres.	97.3 ± 0.5	98.1 ± 0.1	97.7 ± 0.3	97.5 ± 0.3
FRI	90.9 ± 0.7	88.4 ± 0.8	89.6 ± 0.6	92.5 ± 0.5
FRII	72.0 ± 2.6	75.2 ± 3.3	73.6 ± 2.8	92.7 ± 0.8
Avg.	91.6 ± 0.7	91.5 ± 0.7	91.5 ± 0.7	95.0 ± 0.4
(4)				
Unres.	98.4 ± 0.1	98.3 ± 0.1	98.3 ± 0.0	98.3 ± 0.0
FRI	92.0 ± 0.6	91.3 ± 1.2	91.7 ± 0.5	93.9 ± 0.4
FRII	80.4 ± 2.4	82.3 ± 1.8	81.2 ± 1.1	94.9 ± 0.4
Avg.	93.7 ± 0.3	93.6 ± 0.4	93.6 ± 0.3	96.2 ± 0.2

Table 3.8: The average metrics (in percentages) across each of the classes in the original LOFAR dataset, for the inc. filtersize CapsNet model. Five runs were done in total, using 600 samples in the test set.

Class	Precision	Recall	F1 score	Accuracy
Orig.				
Unres.	89.6 ± 0.7	94.2 ± 0.3	91.8 ± 0.5	90.8 ± 0.5
FRI	80.4 ± 2.5	79.6 ± 2.9	79.9 ± 0.1	85.0 ± 0.5
FRII	63.2 ± 6.4	50.5 ± 10.8	54.2 ± 6.7	88.4 ± 0.2
Avg.	82.7 ± 0.5	83.0 ± 0.5	82.5 ± 1.1	88.4 ± 0.4

Table 3.9: The average metrics (in percentages) across each of the classes in the original LOFAR dataset, for the inc. decoder CapsNet model. Five runs were done in total, using 600 samples in the test set.

Class	Precision	Recall	F1 score	Accuracy
Orig.				
Unres.	90.6 ± 2.7	95.0 ± 0.8	92.7 ± 1.8	91.6 ± 2.2
FRI	75.1 ± 2.5	87.8 ± 1.9	80.9 ± 2.2	84.5 ± 1.9
FRII	65.8 ± 2.9	22.7 ± 9.0	32.3 ± 10.7	87.4 ± 0.8
Avg.	81.6 ± 2.0	82.7 ± 2.1	80.3 ± 3.0	88.5 ± 1.9

higher overall metrics compared to the other two CapsNet models (although this is not always significant).

The inc. filtersize model, that was designed with larger filters to capture more extended emission, for the most part performs as well as the default model and the metrics for the FRIIs are improved. However, they tend to be lower for the Unresolved and FRI classes, which make up the majority of samples. The results are shown in Table 3.8.

The inc. decoder model, which uses a more complex decoder, performs as well as the default model in the metrics for the Unresolved and FRI classes. However, it performs worse overall for the FRIIs, as shown in Table 3.9. This may be due to the more complex decoder confusing radio emission from the FRIIs with noise.

As the default CapsNet model performs better overall compared to the other two CapsNet models, it is chosen as the basis of comparison against the two ConvNet models across the original FITS and masked 4rms sigma-clipped datasets.

The default CapsNet model still performs significantly worse compared to the two ConvNets, as it is beyond both their 95% confidence intervals, across all metrics. The variability in metrics is higher for the original dataset compared to that of the two ConvNets, as is evident in the generally increased confidence intervals of the CapsNet model, in Table 3.7, particularly for the FRIIs.

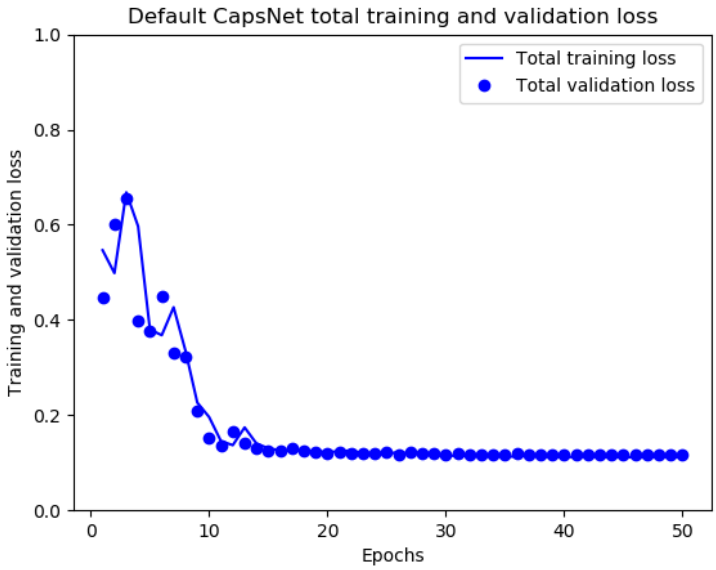

Figure 3.7: The training and validation losses for a single run with the default capsule network architecture, using the margin loss as defined in Equation 3.5, with 2301 (79%) samples for training and 600 (21%) samples for testing. The total loss is obtained by adding the capsule network loss to the decoder weight multiplied by the decoder loss.

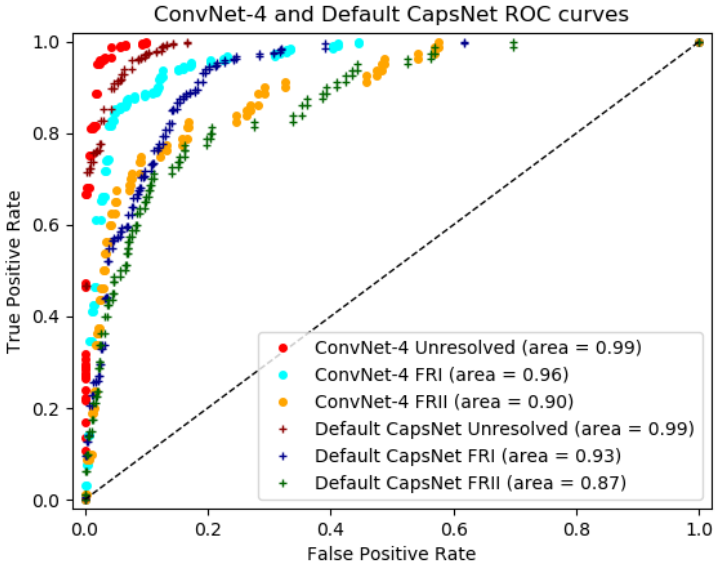

Figure 3.8: ROC curves for both a single run with the default CapsNet model and the ConvNet-4 model. The curves show that ConvNet-4 outperforms the default CapsNet across all the classes.

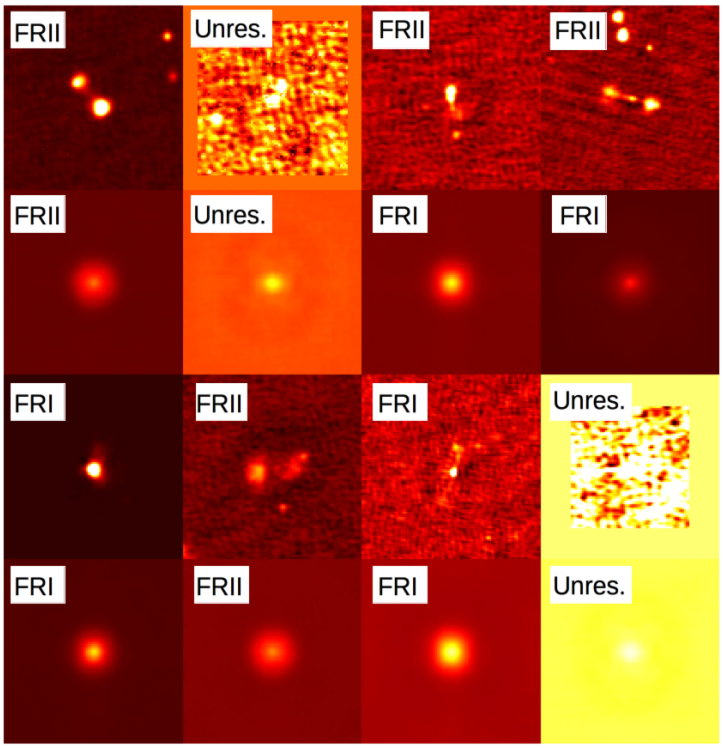

Figure 3.9: The real and reconstructed images using the default capsule network setup when training on the original and augmented images, annotated with the corresponding labels. The top row shows the real images, the second row shows the corresponding reconstructions. The third row shows the real images and the final row shows the reconstructions. The decoder always detects that there is an object in the centre of the image, however it is unable to reconstruct the object accurately. Based on the reconstruction, we see that CapsNet is determining class membership based on the characteristics of the sphere in the centre.

Table 3.10: The labels and corresponding probability vector of the default CapsNet network predictions, using four examples of sources shown in Figure 3.10, having probabilities greater than 50% across two classes.

Source	Label	CapsNet prediction
1	FRI	41% Unres., 50% FRI, 51% FRII
2	Unres.	51% Unres., 36% FRI, 62% FRII
3	FRII	34% Unres., 59% FRI, 57% FRII
4	FRII	16% Unres., 72% FRI, 70% FRII

Figure 3.8 shows the Receiver Operating Characteristic (ROC) curves across the default capsule network and ConvNet-4. ROC curves plot the true positive rate (recall) against the false positive rate.

In a first attempt to use the default CapsNet model (containing 58M free parameters), we observed a clear overfitting, owing to the large number of free parameters compared to the number of training images. Despite this, the model still achieved very similar results to the models using many fewer parameters quoted in the current work.

Figure 3.10 shows four examples of radio galaxies in which the probabilites are greater than 50% across 2 classes, that the CapsNet could therefore not reliably classify. There are a total of 55 out of 600 (9.2%) such cases. Table 3.10 shows the CapsNet probability vector across the four examples. In Source 1, CapsNet gives similar probabilities between the FRI and FRII classes, which could be because the source is quite faint, therefore it is having trouble extracting the morphology. Source 2 is predicted more confidently as an FRII compared to an Unresolved source, perhaps because it appears as though it has two lobes close together. Sources 3 and 4 are labeled as an FRII, however the CapsNet predicts them more confidently as an FRI compared to an FRII, as it may not detect the lobes.

3.5.2 LOFAR original and augmented images

We augmented the images with translation, rotation and flipping as outlined in Section 3.3.2, keeping the distribution of FRI and FRII sources the same as in the original dataset. Table 3.1 gives the number of original and augmented images. There are again 79% and 21% of the original samples used in training and testing respectively.

ConvNet-4 and ConvNet-8

We applied both ConvNet-4 and ConvNet-8 models to the original and augmented dataset, with the results shown in Table 3.5 and Table 3.6. The overall metrics are significantly better (Avg. Recall = 93.4 and 94.3) than was observed when the same model was used on the

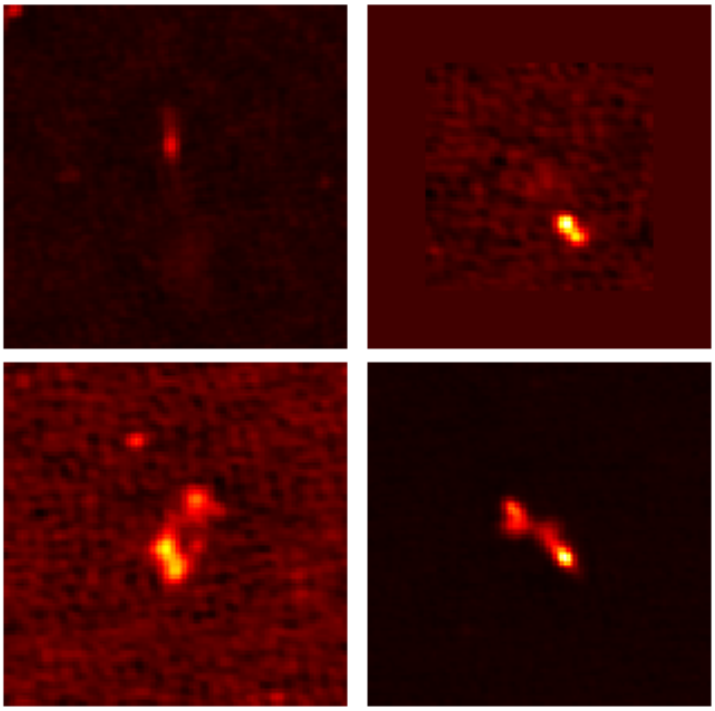

Figure 3.10: Examples of radio galaxies having probabilities greater than 0.5 in more than two classes in the default CapsNet architecture, that are also incorrectly predicted. The labels and predictions from left to right, top to bottom are [FRI,Unres.,FRII,FRII] and [FRII,FRII,FRI,FRI] respectively. These sources are labeled as (1,2,3,4) in Table 3.10.

Table 3.11: Confusion matrix for a single run with the ConvNet-4 architecture, after training on the original and augmented images. The predictions are along the columns and the labels are along the rows.

	Unres.	FRI	FRII	Total
Unres.	294	6	1	301
FRI	3	202	14	219
FRII	5	12	63	80
Total	302	220	78	600

original images (Avg. Recall = 88.7 and 91.2 for ConvNet-4 and ConvNet-8 respectively), therefore both models benefit from data augmentation. The confidence intervals are also usually reduced.

Although the classification metrics remain the poorest for the FRII class, they improved the most when using the augmented data, despite the fact that there were more examples of FRIs.

A confusion matrix is provided in Table 3.11 for the ConvNet-4 model, to see the numbers of samples that are both correctly and incorrectly predicted.

CapsNet

The best-performing capsule network (the default model) was used to see whether an improvement in overall metrics could be obtained when using augmented images in addition to the original images. The results are shown in Table 3.7. The confusion matrix for a single run with the default CapsNet architecture, after training on the original and augmented images, is given in Table 3.12.

The classification metrics are significantly improved when using the augmented images (Avg. Recall = 89.3 with augmentation, compared to Avg. Recall = 84.2 ± 0.2 without), therefore the capsule network also benefits from training on additional images. Despite the fact that capsule networks output a vector describing the properties of images across the classes and aim to extract the underlying patterns, they still benefit from the use of additional augmented images, for the FITS file dataset. The noise in the images could be preventing the network from seeing the underlying morphology in the signal, and there is an insufficient number of images available across the classes, hence improved results are observed when more examples are provided. Despite CapsNet benefiting from augmentation, the classification metrics are still significantly lower compared to when augmentation is applied to the two ConvNets.

Figure 3.9 shows the real and reconstructed images for a single run of the default CapsNet model when training on the original and augmented images. The labels match the predictions

Table 3.12: Confusion matrix for a single run with the default CapsNet architecture, after training on the original and augmented images. The predictions are along the columns and the labels are along the rows.

	Unres.	FRI	FRII	Total
Unres.	289	12	0	301
FRI	4	198	17	219
FRII	6	24	50	80
Total	299	234	67	600

with the exception of the third and fourth images in the top two rows, where the true labels are FRIIs but the predictions are FRIs. The reconstructions of the images are innaccurate, giving the appearance that CapsNet is determining class membership based on the blurriness of the reconstructed spheres. The images in the 'Unresolved' class are represented as concentrated spheres, FRIs are less concentrated, blurrier spheres, and FRIIs are the most diffuse. The inaccuracy of the reconstructions is most likely due to the fact that CapsNet appears to have trouble distinguishing signal from noise. Despite this, the average metrics are still above 89% when training on the original and augmented images, as it does not appear to be necessary to have accurate reconstructions to determine class membership. Overall, the FRII source predictions appear to be the most affected by the noise level and/or potential unassociated emission in the images; since the reconstructions tend to be blurrier spheres with only one component, they become confused with FRIs and FRIIs, as FRIIs can have either both lobes being connected, as well as disconnected.

Similar to what was observed in the ConvNet architectures, the metrics across the FRII class are the poorest. However, after training with the original and augmented images, the FRII metrics improved the most. The FRII class has the fewest examples of images compared to the other two classes.

Despite the use of image augmentation, it is likely that the number of original training samples available is insufficient to train a capsule network.

3.5.3 Sigma-clipped images

In order to test whether the CapsNet performance could be improved by removing noise and the occasionally unassociated emission, we used the sigma clipped images that mask out pixels below 4rms. A flood-filling algorithm and masking techniques have additionally been applied to the dataset to identify and connect associated emission (Mingo et al. in prep). We analyse the results obtained from using the original sigma-clipped images, as well as both the original and augmented images.

The performance of both ConvNets is significantly improved as shown in Tables 3.5 and 3.6

(Avg. Recall = 91.9% compared to 88.7% for ConvNet-4, 94.9% compared to 91.2% for ConvNet-8) when using the original sigma clipped images, compared to using the original FITS files that includes noise and potential unassociated sources. The use of the original sigma clipped images is significantly worse compared to using the original and augmented FITS images for the ConvNet-4 model (Avg. Recall = 91.9% compared to 93.4%), and is not significantly better for the ConvNet-8 model. The inclusion of augmented images on the sigma-clipped dataset appears to benefit the ConvNet-4 model more compared to the ConvNet-8 model.

The performance of CapsNet is significantly improved as shown in Table 3.7 when using the sigma-clipped original images (Avg. Recall = 91.5% compared to 84.7% with the original FITS images, and compared to 89.9% with the original and augmented FITS images). However, CapsNet still performs worse compared to both ConvNet-4 and ConvNet-8. The use of image augmentation on the sigma-clipped images appears to improve the performance (Avg. Recall = 93.6% compared to 91.5%) The confidence intervals are also generally smaller compared to when the FITS images are used, therefore the performance is slightly more stable.

The use of the sigma clipped and masked arrays is also significantly better than using the FITS images, when comparing the performance within the original, and the original and augmented datasets, across both ConvNet models and CapsNet models. Therefore, none of the deep learning models can be trained to be completely robust to noise and potentially unassociated emission.

In considering the results of one particular run with the ConvNet-8 model, out of 600 test samples, there are 20 where the predictions do not match the labels. Figure 3.11 shows four such examples of images from the 4rms sigma-clipped dataset. Upon inspection of all the incorrectly predicted radio galaxies using the ConvNet-8 model, all 12 images that have been labelled as an FRII are predicted to be an FRI. Out of 3 images labeled as 'Unresolved', two are predicted to be an FRI and one is predicted to be an FRII. The remaining 5 images labelled as FRI are predicted to be FRIIs. The wrongly classified galaxies mostly appear to have an ambiguous morphology and therefore it could be argued that they are mis-classified by the automated algorithm used to label them (see Section 3.2.2 and Mingo et al. (in prep.)). For example, the top right and bottom left panels in Figure 3.11 do not appear to be a representative examples of an FRII, and the bottom right panel appears more as an FRII, whereas it is labeled as an FRI.

We note that the larger the proportion of sources that are mis-classified by the automated algorithm, the more difficult it will be for the models to learn.

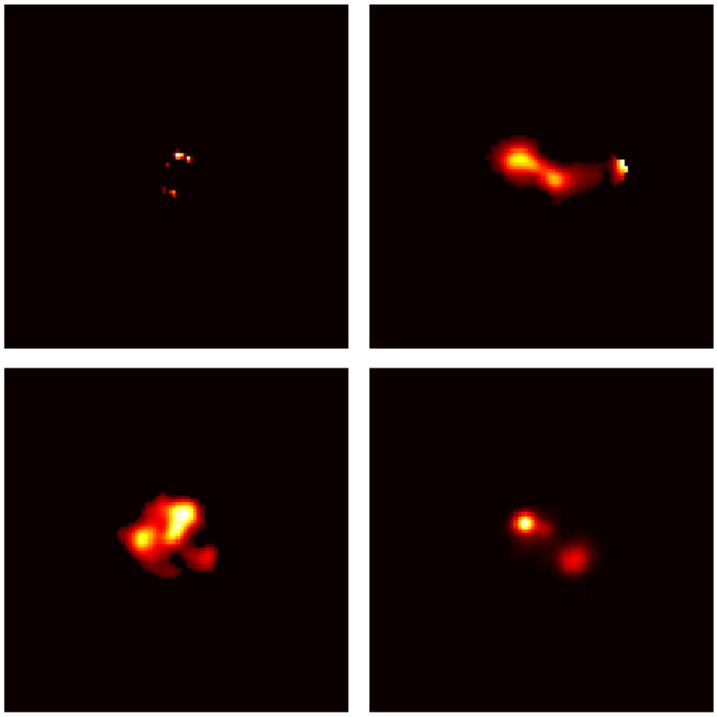

Figure 3.11: Examples of incorrectly classified radio galaxies from the 4rms sigma-clipped dataset using the ConvNet-8 layer architecture. The labels and predictions from left to right, top to bottom are [Unres.,FRII,FRII,FRI] and [FRI,FRI,FRI,FRII] respectively. The top left image appears to have too few pixels to be reliably classified, thus belonging to the 'unresolved' class, however the remaining three may have been misclassified by the automated algorithm.

Table 3.13: The average metrics (in percentages) across each of the classes in the (2) original and augmented LOFAR dataset using a 5 convolutional layer model with no intermediate dense layers. Five runs were done in total, using 600 samples in the test set.

Class	Precision	Recall	F1 score	Accuracy
(2)				
Unres.	97.7 ± 0.6	96.9 ± 0.9	97.3 ± 0.3	97.2 ± 0.3
FRI	88.4 ± 1.7	92.8 ± 1.9	90.5 ± 0.1	92.8 ± 0.1
FRII	77.8 ± 2.9	69.0 ± 2.5	73.0 ± 1.3	93.1 ± 0.4
Avg.	91.6 ± 0.2	91.7 ± 0.1	91.6 ± 0.1	95.0 ± 0.1

3.5.4 Additional results

This section summarises other convolutional and capsule network architectures as well as parameters, that were tried. These include transfer learning, the application of early stopping and comparison of results with similar work.

ConvNet models

We also wanted to test the performance of a simple purely convolutional architecture using 5 layers (with no intermediate dense layer following the convolutions). The purpose of these dense layers is to help model complex global patterns in the data. The metrics were significantly lower compared to those of both ConvNet models, as shown in Table 3.13. Therefore, at least one intermediate dense layer could be necessary for optimal performance in convolutional networks. We also tested an architecture using 4 convolutional with no pooling layers, and found the results to be inferior compared to using the ConvNet-4 model. Therefore, the use of pooling is appears to be advantageous in the current dataset, perhaps because it allows more degrees of freedom for the morphology within classes.

CapsNet models

Other variations on capsule network models included stacking two convolutional layers instead of one, using 90% training data and 10% testing data, using an ensemble of capsule network models, increasing the number of routing iterations, decreasing the filter size, changing the batch size, adjusting the learning rate, using different activation functions, applying dropout, pooling and using a combination of increased filter sizes together with a more complex decoder, all which resulted in similar or worsened performance metrics. The only possible improvement could be the use of a larger sample of original training images.

Transfer learning

Transfer learning (Pratt et al., 1991) involves applying the knowledge from one trained neural network to help another learn a related task. In the deep learning context, weights are typically pre-loaded from a network trained on a large dataset with many classes to another unseen dataset.

We used the Inception ResNet model v2 (Szegedy et al., 2016), which combines Inception and Residual network architectures. An inception network consists of a convolutional network using filters of various sizes and pooling within the same layer, and a residual network utilises skip connections between convolutional layers if the classification accuracy becomes saturated with the subsequent stacking of layers. The Inception ResNet model is trained on the ImageNet dataset (Deng et al., 2009), to classify over 14M images into 1000 categories. Although the nature of the ImageNet dataset is different to the radio galaxy images, pre-loading weights from a network trained with such a dataset is better than initialising the weights from a random distribution.

To use the pre-trained ResNet model in Keras requires images of size of at least 139x139 pixels. As such we padded our images with zeros for 20 pixels along the horizontal and vertical directions, resulting in images of 140x140 pixels.

The pre-trained ResNet model is applied to the LOFAR original and augmented FITS images, to verify whether the classification metrics could be improved from those of our other models. The results in Table 3.14 show that the classification metrics are not significantly better (Avg. Recall = 94.5%) compared to when training on the same set of images from randomly initialised weights with the ConvNet-8 architecture (Avg. Recall = 94.3%). The metrics are significantly better than for the ConvNet-4 architecture (Avg. Recall = 93.4%). Optimal results are still obtained when using the sigma clipped dataset, where noise and potentially unassociated sources are removed.

We note that the results obtained with transfer learning may be improved if there is a neural network trained on a similar astronomical classification task from which pre-trained weights can be loaded. A successful implementation of transfer learning in classifying optical galaxy morphology is in Domínguez Sánchez et al. (2019), and most recently in radio galaxy morphology classification (Tang et al., 2019).

The pre-trained network converges faster; ConvNet-4 required 40 epochs of training to reach the optimal validation accuracy as opposed to 30 epochs for the transfer learning model, when averaged over five runs.

Table 3.14: The average metrics (in percentages) across each of the classes in the (2) original and augmented LOFAR dataset, for the transfer learning model. Five runs were done in total, using 600 samples in the test set.

Class	Precision	Recall	F1 score	Accuracy
(2)				
Unres.	98.7 ± 0.2	98.3 ± 0.4	98.5 ± 0.2	98.4 ± 0.2
FRI	91.8 ± 0.5	95.0 ± 0.4	93.4 ± 0.2	95.0 ± 0.2
FRII	85.4 ± 1.2	78.7 ± 2.7	81.9 ± 1.2	95.3 ± 0.2
Avg.	94.4 ± 0.2	94.5 ± 0.2	94.4 ± 0.2	96.8 ± 0.1

Early stopping

We also experimented with applying early stopping in the training of both the Capsnet and ConvNet models. The implementation was such that if the validation accuracy did not improve for 10 subsequent epochs, training was stopped and the metrics on the test set were calculated. However, we found the performance to be the same for the ConvNet model, and worse for the CapsNet model, compared to when training for a pre-defined number of 50 epochs (results not shown). In a work focused on the usage of early stopping, Montavon et al. (2012) used a mix of more than 1000 training runs across 12 different problems and 24 different architectures and concluded that slower stopping criteria allow for $\approx 4\%$ average improvement in generalisation, at a cost of around a factor of four longer in training time.

Recent similar work

Recently, Katebi et al. (2019) applied a capsule network to classify optical galaxies based on morphology, using the classes of spiral, elliptical and star/artefact. They find that their capsule network classification accuracy surpasses that of their baseline convolutional network (98.77% versus 96.96% respectively). The capsule network architecture has over 124M parameters, for a total of 61,578 images. In contrast, our best-performing capsule network uses just over 4M parameters with up to 15,936 images using the original and augmented dataset.

We note that the difference in morphology between their classes is starker than in our case. Additionally, the optical images show a much better contrast between object and background, where noise is less prominent. The optical galaxy classifications were crowd-sourced, whereas our labels originated from an automated algorithm which comes with some limitations, as outlined in Section 3.2.2. The radio emission also produces sparser images compared to the optical galaxy images.

It is difficult to compare their work to ours as the number of images in each of their 3 classes is unknown. Hence, it is uncertain whether the classification accuracy is the best

discriminator to use between the models (Hossin & M.N, 2015). Other classification metrics are not provided, such as precision and recall, which may be more powerful in discriminating models. There is also no indication of variability between runs, as well as the degree of overfitting in the networks during training.

3.6 Conclusions

This paper explored two deep learning approaches in the classification of radio data from the LoTSS HETDEX field across three classes of radio galaxies: Unresolved sources, FRI and FRII galaxies. The labels were generated using an automated algorithm, which used a catalogue of sources from the LoTSS DR1 source catalogue with optical IDs and associations (Williams, W. L. et al., 2019). The radio galaxies belonging to the FRI and FRII classes were additionally cross-checked to eliminate galaxies in which the radio emission is likely to be dominated by star formation (Hardcastle et al., 2019). Despite the classifications being generated using masked images that remove potentially unassociated sources and emission below 4rms from the images, one of our aims was to test how robust our deep learning algorithms could be when such effects were present.

We tested the performance of a four and eight layer convolutional neural network (ConvNet-4 and ConvNet-8) against various architectures of capsule networks (CapsNet), using the precision, recall, F1 score and accuracy, to evaluate the performance of the models. Python code implementing v1.0 of the algorithms can be obtained from github[8]. Automated classifications of LoTSS sources obtained with the algorithms will be presented in a future paper (Mingo et al., in preparation).

The first CapsNet model explored was the default model, a simplified architecture of the original model designed for the MNIST dataset, the second used larger filter sizes in the first convolutional layer and Primary capsule layer, and a larger stride in the convolutional layer. The third model used a more complex decoder and a higher loss for the decoder weight. The second and third models were designed to better account for the increased complexity of the data. Four different sets of data were used to train and test the two ConvNets and the variations on CapsNet architectures: (i) using the original FITS images only, (ii) original and augmented FITS images, (iii) the original masked arrays that remove emission below 4rms and potential unassociated sources and (iv) original and augmented masked 4rms arrays.

We found that the optimal CapsNet performance was obtained when using the default model, in terms of the overall classification metrics.

The results showed that the ConvNet architectures always exceeded the performance of the

[8]https://github.com/vlukic973/RadioGalaxy_Conv_Caps

chosen CapsNet model, and ConvNet-8 always performed better compared to ConvNet-4, most likely because the ConvNet-8 model has twice the number of convolutional layers and parameters as ConvNet-4, therefore it is able to extract higher-dimensional features that are particular to each class.

The use of transfer learning on the original and augmented images achieved the same results as ConvNet-8. The performance of all deep learning models was optimised when using the 4rms sigma clipped numpy array, which is expected as the noise and potential unassociated emission is removed. Some observations of differences in results between using ConvNet and CapsNet architectures and the likely reasons are as follows:

- As CapsNet tends to capture and preserve the relative location of features in the images, it is not as successful in distinguishing signal from noise, or dealing with the presence of potentially unassociated emission, as the ConvNet architectures
- The use of pooling in the ConvNet architectures generally appears to be advantageous in two respects: (i) increased likelihood that noise and potential unassociated sources will be filtered out, (ii) allowing more degrees of freedom for variability in morphology within the classes, when the undesirable effects have been removed through use of the 4rms dataset
- The removal of noise and potentially unassociated emission through the use of sigma-clipped and masked arrays improves the performance of both deep learning approaches, when considering the metrics within the original, and original and augmented datasets
- The use of image augmentation appears to benefit both ConvNets and CapsNet, when using the FITS files, which contain the original radio emission.

The LoTSS survey is the first wide-area survey to contain such faint sources. It is sensitive to a larger range of source evolutionary states, and can also see structure on a wider range of spatial scales due to the combination of well-sampled UV coverage and long baselines. These features result in images having richer, more varied and sometimes ambiguous morphologies that are more difficult to categorise into distinct classes.

Across both deep learning algorithms, the 'Unresolved' class is recovered most successfully, followed by the FRI class. The FRIIs tend to be the least well recovered. Although FRIs display morphological diversity as they can be straight or bent, FRIIs have two peaks of varying distances that may or may not be connected by extended emission with the host galaxy. Therefore, FRIs are more likely to contain a single connected component whereas FRII can contain either a single or two connected components. There are also fewer examples of FRIIs in the dataset compared to FRIs. When we inspected some incorrectly predicted galaxies using the sigma-clipped dataset, we found the morphologies to be ambiguous in most cases, as shown in Figure 3.11.

Traditional convolutional neural networks generally contain pooling layers in their architecture in order to reduce the number of parameters. However, this can cause the relative locations of features within the image to degrade, which capsule networks are designed to preserve. Our results indicate that for the radio galaxy data in the current work, the performance of capsule networks is inferior to that of convolutional neural networks. This could be due to the number of original samples being insufficient to train the capsule network. Another reason may be that since they attempt to preserve the relative location of features, capsule networks appear to interpret noise as signal and introduce extra distortion into the image, as shown in Figure 3.4. This aspect has proven to be most detrimental in the recovery of FRII sources, as they are more susceptible to the mingling of signal with noise due to the fact that they are comprised of either one or two components. Additionally, the FRII class contains the fewest examples of images.

In comparison with previous works that use convolutional neural networks to classify radio galaxy morphologies (Aniyan & Thorat (2017), Lukic et al. (2018), Wu et al. (2018) and Alhassan et al. (2018)), the current work explored the use of capsule networks, which are designed to preserve the hierarchical feature information in an image, and finds their performance to be inferior to that of standard convolutional network architectures. The data from the LOFAR LoTSS survey reveals fainter and more detailed emission compared to the data from the surveys which the previous works analysed, providing additional challenges for classification. As such, our findings hold for surveys having a comparable setup, provided they produce images with similar morphologies and noise profiles.

Based on the current results obtained, it appears that convolutional neural networks still hold as the deep learning technique that should be used for future surveys. They are also faster to train as they use fewer parameters. Capsule networks, in their present form, are generally slower and require further development to be made more robust to noisy real data, however the current performance may be improved by explicitly training them on cleaned data with various examples of morphologies present within each class.

There are several limitations that would need to be overcome to apply these methods to large samples, such as the need for ancillary data to separate star-forming galaxies. The exclusion cannot be performed based purely on the radio galaxy morphology. The classes should also be extended to encompass the hybrid sources, as well as other rare sources such as bent-tailed and double-double sources.

4 Source-finding with convolutional autoencoders

The following chapter presents work as it is submitted by Lukic et al. (2019) to Galaxies.

4.1 Introduction

An ongoing task in astronomy is the ability to find astronomical sources. This is of importance because it forms the basis by which a radio astronomical catalogue can be built. Modern radio telescopes can observe many millions of radio sources and this number will only increase in time owing to rapidly developing technologies (Norris, 2017a). It is therefore important that the methods developed to find sources can keep up with the capabilities of the technology, with respect to the quality of sources that are detected by the telescope.

In this section, we give a brief summary of the main factors affecting the ability to find sources in radio data, the different types of radio sources (star-forming galaxy or type of AGN), how a machine learning approach can work, details about the simulated Square Kilometre Array (SKA; Prandoni & Seymour, 2015) data used, as well as a brief review of the previous work in this area.

4.1.1 Source-finding at radio frequencies

Radio telescopes measure the surface brightness of the radio sky across some frequency or range of frequencies and produce a map of the surface brightness. What constitutes a source is a collection of pixels above some value, which is determined by estimating the background, or noise. The noise is usually composed of a combination of instrumental noise, observed background emission and leftover system uncertainties (Savage & Oliver, 2007).

The first step involved in source-finding is usually pre-processing the image containing the radio sources. This involves some transformations to the image, such as scaling the pixel intensities, to facilitate the source-finding method by suppressing undesired distortions or enhancing features (Miljković, 2009), while preserving the physics of radio sources in the

image. The second step is to estimate the background, after which a threshold can be chosen, that defines where the sources are. Contiguous pixels above a certain threshold are considered to form part of an object (Hopkins et al., 2002), after which a local peak search is performed where maximum-value pixels are isolated.

In the presence of low SNR, which occurs when there is a relatively high background compared to surface brightness (signal) from the source, it can be difficult to group the pixels belonging to a particular source. Additionally, the sizes and intensities of the astronomical bodies can vary significantly (Zheng et al., 2015). As the SNR is increased, finding and extracting the sources becomes easier as the pixels belonging to the source show a greater contrast compared to the background. However, it is more frequently the case that shorter integration times are used, which results in noisier data and it is not always easy to capture the background signal, which may also vary across regions in the image. Another problem to consider is that of source confusion, which is the inability to measure faint sources due to the presence of other sources nearby. Also, at radio frequencies, the noise tends to be more correlated compared to other frequencies (Radhakrishnan, 1999; Ellingson, 2011; Hale et al., 2019), posing further challenges for source-finding and extraction.

Many algorithms have been developed to perform source-finding across different wavelengths such as optical, radio, infrared or x-ray, some of which use a combination of techniques. Masias et al. (2012) presents the largest overview of the most common techniques, although there have been more recent developments. For example, a source extractor originally developed for source-finding in optical images (ProFound; Robotham et al. (2018)) can also successfully be used at radio wavelengths (Hale et al., 2019).

One state-of the art source-finding algorithm is the Python Blob Detector and Source-Finder[1] (PyBDSF;Mohan & Rafferty (2015b)), which works as follows: After reading in the image, it performs some pre-processing, for example computing the image statistics. Using a constant threshold for separating the source and noise pixels, the local background rms and mean images are computed. Adjacent islands of source emission are identified, after which each island is fit with multiple Gaussians, or Cartesian shapelets. The fitted Gaussians or shapelets are flagged to indicate whether they are acceptable or not. The residual FITS images are computed for both Gaussians and shapelets. Gaussians within a given island are then grouped into discrete sources.

There have been a couple of recent works on using deep learning methods to perform source-finding. ClaRAN (Wu et al., 2018) trained a source-finder on Radio Galaxy Zoo data (Banfield et al., 2015) to learn two separate tasks; localisation and recognition, after which the source is classified according to the number of peaks and components, with accuracies >90%. More recently, DeepSource (Vafaei et al., 2019) presents a deep-learning algorithm to find point

[1] https://www.astron.nl/citt/pybdsf/

sources in simulated images and compares the results against PyBDSF, using different signal-to-noise ratios. In contrast, the current work examines the recovery of SFGs and two classes of AGN as well as all sources combined, at different SNRs using a convolutional autoencoder (AutoSource) and compares the results against PyBDSF, and shows in which circumstances one performs better than the other and the likely reasons why. DeepSource requires the tuning of more hyperparameters, which are variables that need to be defined prior to applying a machine learning algorithm. Autosource requires only the usual deep learning parameters such as number and type of layers, batch size, cost function and gradient-descent method.

4.1.2 Types of radio sources

Galaxies exhibiting significant radio emission usually fall into one of two groups; star-forming galaxies (SFGs) or Active Galactic Nuclei (AGN). Radio-loud AGN can be grouped based on their appearance; they can be either 'compact' or 'extended'. The two most influential factors that govern whether a source will appear point-like, elongated or very resolved are the distance of the source and the resolution. Different radio source types can be characterised by a different spectral index α, which is related to the frequency ν and flux density S through $S(\nu) \propto \nu^{\alpha}$. The slope of the spectrum is determined by the electron energy distribution. Extended radio sources generally have a steep radio spectrum (typical values are $\alpha \lesssim -0.8$ (de Gasperin et al., 2018)) and can be referred to as steep-spectrum AGN (AGN-SS), where the majority of sources can be divided into two distinct classes depending on the morphology of the radio lobes; FRI (core-dominated) and FRII (lobe-dominated) (Fanaroff & Riley, 1974). Compact radio sources tend to exhibit a flat radio spectrum (typical values are $\alpha \leq -0.5$) and denoted as flat-spectrum AGN (AGN-FS) (Peterson, 1997). It should also be noted that some steep-spectrum sources can be compact.

Since the relative strength of the emission from radio sources depends on frequency, different components of a radio source can have different spectral shapes.

4.1.3 Deep learning

Deep learning methods have been successful in extracting information from high-dimensional data such as images (Krizhevsky et al., 2012; Farabet et al., 2013; Zeiler & Fergus, 2013). The use of filters in convolutional layers, which serve to scan across the images and detect features, typically have sizes of a few pixels across and therefore greatly reduce the number of parameters compared to the fully connected layers in traditional neural networks. The stacking of convolutional layers results in a hierarchical extraction of features. Convolutional networks are more successful in avoiding the vanishing gradient problem compared to fully connected

neural networks while simultaneously enforcing translational invariance (Goodfellow et al., 2016).

The current work explores a novel approach to source-finding by training an autoencoder on a solution map derived from knowledge of the source locations. Autoencoders are generally an unsupervised learning technique (Liou et al., 2008), and are made up of an encoder, that aims to compress the input into a lower dimensional representation, and a decoder, whose original function is to reconstruct the original image from the compressed representation (Rumelhart et al., 1986). For our purpose of source-finding, the output images we aim to produce are those of the locations of the sources, rather than the original input source maps. Given that the source locations can be transformed into image data, the source location map, along with the original source map, can be segmented into smaller square images (having size 50x50 pixels in the current work), which are then used as the inputs to train the autoencoder to predict the source locations.

We note that we only focus on the source-finding aspect in this work, rather than the source characterisation and classification.

4.1.4 Simulated SKA data

The SKA aims to be the largest radio telescope built to date. It will eventually have a collecting area of more than one square kilometre and operate over a wide range of frequencies (50 MHz - 14 GHz in the first two phases of construction) and will be 50 times more sensitive than any other radio instrument to date. In the meantime it is possible to use simulated data products to generate data similar to what would be expected to be observed by the SKA. The SKA Data Challenge 1[2] (SKA SDC1; (Bonaldi & Braun, 2018)) was a recent challenge set for the community to develop or use existing source-finders to perform source-finding, characterisation of the sources and source population identification (either SFG, AGN-steep or AGN-flat).

Catalogues of objects to be included in the simulated maps were generated using the Tiered Radio Extragalactic Continuum Simulation (T-RECS) simulation code (Bonaldi et al., 2018). The radio sky was modelled in continuum, over the 150 MHz-20 GHz range, with two main populations of radio galaxies: Active Galactic Nuclei (AGN) and Star-Forming Galaxies (SFGs) and their corresponding sub-populations. The wide-ranging frequency has been enabled by allowing specific conditions for the spectral modeling. Across the AGN, the sources are allowed to have a different spectral index below and above $\sim$ 5GHz, constrained by the modelled counts from Massardi et al. (2010) for the lower frequency range and de Zotti et al. (2005) for the higher frequency range. In the SFG population, the spectral modelling in-

[2]https://astronomers.skatelescope.org/ska-science-data-challenge-1/

cludes synchrotron, free-free and thermal dust emission, all expressed as a function of the star-forming rate. The redshift range of the simulation is $z = 0 - 8$. The T-RECS simulation output used for SDC1 contains all the sources in a 3x3 Field of View (FoV) with integrated flux density at 1.4 GHz $>$ 100 nJy (Bonaldi et al., 2018).

The data used in the current work is based on the simulated data products generated for SDC1. There are three available frequencies (B1: 560MHz, B2: 1400MHz and B5: 9200MHz) at 3 integration times (8 h, 100 h and 1000 h) for each frequency. There are 9 maps altogether in the form of FITS files. The size of the maps is 32768×32768 pixels. The FoV was chosen for each frequency to contain the primary beam for a single telescope pointing out to the first null, giving a map size of 5.5, 2.2 and 0.33 degrees on a side for B1, B2 and B5 respectively, with corresponding pixel sizes of 0.60, 0.24, and 0.037 arcsec. Properties of sources in a training set area are also provided, across the three frequencies, to see how the sources are characterised in a particular area so the source-finders can be calibrated or trained. We use the generated data and focus on the source-finding aspect only.

In constructing the SDC1 image corresponding to the T-RECS source catalogue, sources have been injected with a different procedure depending on whether they were extended or compact (major axis greater or smaller than 3 pixels respectively) with respect to the adopted frequency-dependent pixel size. The SFGs have been modeled using an exponential Sersic profile (Sérsic, 1963), projected into an ellipsoid using a given axis ratio and position angle. The AGN populations (steep-spectrum and flat-spectrum) are treated as the same object type viewed from a different angle. Steep-spectrum AGN will assume FRI/FRII morphologies, and flat-spectrum AGN are composed of a compact core with a single lobe, but pointing in the direction of view. The steep-spectrum sources have been generated as postage stamps (that includes affine transformations) from a library of scaled real high-resolution images. They have also had a correction applied to the flux of the core in order to give it a flat spectral index, thus the same AGN can have a different core to lobe fraction when viewed at different frequencies. The flat-spectrum AGN are added as a pair of circular Gaussian components: a compact core with a more extended end-on lobe.

A mild Gaussian convolution has been applied to the extended source images, using a FWHM of two pixels. The three catalogues (SFGs, steep spectrum AGN and flat spectrum AGN) of compact objects were added to the image as elliptical Gaussian components.

All the compact sources that belong to the classes of SFGs, steep and flat spectrum AGN are described by an integrated flux density and a major and minor axis size. The compact flat spectrum AGN are additionally described with a core fraction that indicates the proportion of emission belonging to the core of the source compared to the source extent.

Visibility data files were generated using the SKA1-Mid configuration. There were two cases

explored: (1) When the 64 Meerkat dishes were included there were 197 antenna locations specified at B2, and (2) when the Meerkat dishes were not included, 133 antenna locations were used at B1 and B5. Both cases are frequency-dependent and reflect the fact that Meerkat will most likely not be equipped with feeds for B1 and B5.

The visibility sampling is based on 91 spectral channels that span a 30% fractional bandwidth, using a time sampling which spanned -4h to 4h of Local Sidereal Time with an increment of 30 s integration time at an assumed Declination of -30. The visibility files were used to generate the noise images and the point spread functions. The gridding weights for the visibility data were determined by firstly accumulating the visibility samples in the visibility grid with their natural weights. After this, a FFT-based convolution was applied to the visibility density grid using a Gaussian convolving function with FWHM of 178 m. The convolving function width was manually tuned to match as closely as possible to the sampling provided by the array configuration. Uniform weights for the visibilities were formed by using the inverse of the local smoothed data density. After this, a Gaussian taper was used such that it resulted in the most Gaussian possible dirty beam with a target FWHM of (1.5, 0.60 and 0.0913) arcsec at (560, 1400 and 9200) MHz. The actual dirty beam dimensions were closely matched to the target specification. There was a degradation of image noise compared to the naturally weighted image noise, therefore they were rescaled in amplitude to represent realistic variations in RMS for the different integration times. Adding the various noise images to the convolved sky model resulted in the final data products.

Additional files provided include the Primary Beam images, which are used to correct the flux values in the original maps, the synthesised beam images and the training set files, which include the properties of the sources such as flux, size, and class, for a particular area in the entire map. There are three training set files, for the three frequencies. Therefore, the same training set file is used across the 3 different integration times within one frequency.

For more specific details on the generation of the simulated SKA data, please refer to Bonaldi & Braun (2018).

The paper is outlined as follows: In section 4.2 we discuss the specifics about the the SKA simulated dataset, the pre-processing steps on the raw data, the parameters by which PyBDSF is run, how the dataset has been generated prior to undertaking source-finding with AutoSource and PyBDSF, the background of autoencoders as well as how the images have been augmented for AutoSource. Section 4.3 describes the major results summarised in F1 scores that combine precision and recall. We also provide confusion matrices for some data subsets. Section 4.4 summarises our overall findings. Appendix 4.5 contains the precision and recall classification metrics.

4.2 Methods

4.2.1 Convolutional autoencoders

In the context of neural networks, an autoencoder is made up of an encoder and decoder. The encoder compresses the input into a latent space representation that usually has a smaller dimensionality (referred to as a bottleneck), compared to the input data (Tishby et al., 1999). The encoder can be represented by the function $h = f(x)$. The aim of the decoder is to reconstruct the input from the latent space representation, which can be represented by $r = g(h)$. The complete autoencoder function, which can be expressed as $r = g(f(x))$, aims to achieve a reconstruction, r that is as close as possible to the original input data, x (Vincent et al., 2008).

Autoencoders are generally an unsupervised learning technique (Liou et al., 2008) as they are usually trained without labels. One of the aims is to detect determining features in the data, that could help to characterise similar types of images and therefore infer properties that are common to groups.

Some applications of autoencoders include denoising and dimensionality reduction, as well as unsupervised pre-training to better initialise the weights of the neural network compared to using a random distribution (Schmidhuber, 2015).

Autoencoders can be made up of fully connected layers, however this can be impractical for image data as it is generally high-dimensional, and could cause learning to stall owing to the saturating/declining gradients (Mao et al., 2016). Using convolutional layers reduces the number of parameters, as they employ filters of a smaller size that scan across the network and detect features.

In the most basic case, one possible application of an autoencoder on radio astronomy data of the type explored in the current work, is to reconstruct the original input maps that contain the sources as well as the background noise, which is an undesired feature. A better application could be to investigate whether it is possible to derive maps similar to the 1000 h maps using the 8 h emission maps, because the shorter integration time maps can be viewed as noisier versions of the longer integration time maps. This can be the subject of a future work.

The key idea behind AutoSource is to train an autoencoder $r = g(f(x))$, where x is the input map data and r is the reconstruction of the solution map, where the sources of varying sizes and emission patterns are collapsed to individual pixel locations, and the remainder of the image is blank. The autoencoder is trained to do source-finding using segmented real maps and the corresponding solution map, both having sizes of 50x50 pixels, across three SNR

ratios of 1,2 and 5, and the results are compared with the sources found using PyBDSF.

It should be noted that this method of source-finding can be posed as an image-to-image translation problem, as are many problems in the computer vision field (Isola et al., 2016), which are generally solved using Generative Adversarial Networks (GANs). GANs consist of a generator of images (usually an autoencoder), as well as a discriminator, whose job it is to differentiate between real and generated images. The aim of using GANs is to generate images in which the discriminator fails to distinguish between them and real images. In contrast, our method requires only the use of an autoencoder as we are generating images having a much lower level of complexity compared to the input maps; where the radio emission of sources is collapsed to between one and a few pixels.

The present work uses Keras[3] with the TensorFlow[4] backend and Python version 2.7.15. We use a convolutional network architecture of three consecutive convolutional layers and one dense layer, having a total of 32,193 parameters.

Early stopping was used with a patience of 5 training epochs. A single training epoch is when all training samples are passed through the network. 80% of the data is used for training, and the remaining 20% is used for testing.

The AutoSource architecture, as shown in Table 4.1 and Figure 4.1, is made up of 3 convolutional layers and one dense (fully-connected) layer. There are 16, 32 and 64 filters, with a filter size of 7, 5 and 3 in the first, second and third convolutional layers respectively. A dropout layer using a dropout fraction of 0.25 is inserted between the first and second convolutional layers to make the network more robust. We slide the filters along by one pixel in each layer to ensure maximal information extraction. The batch size is set to 128. We use the Adadelta optimiser (Zeiler, 2012) with a default learning rate of 1.0, decay of 0 and a rho of 0.99. Adadelta is based on Adagrad (Duchi et al., 2011) (an optimizer with parameter-specific learning rates), however Adadelta adapts the learning rates based on a moving window of gradient updates. We also use the binary cross-entropy cost function (Mannor et al., 2005). The architecture shown in Figure 4.1 also contains an example of a real input map and solution map, the features detected, and corresponding reconstructed map.

We note the absence of a bottleneck (a compressed representation of the inputs) in the architecture. A bottleneck is not used because we are not attempting to see whether the original map can be reconstructed from the input data using a lower-dimensional projection, but to train the network to predict the location of the sources. The stacked convolutional layers extract the signal from the noise, where each layer produces an output having the same dimensions as the input, in order to directly see the detected signals that are propagated through the network.

[3]https://keras.io/preprocessing/image/
[4]https://keras.io/losses/#categorical crossentropy

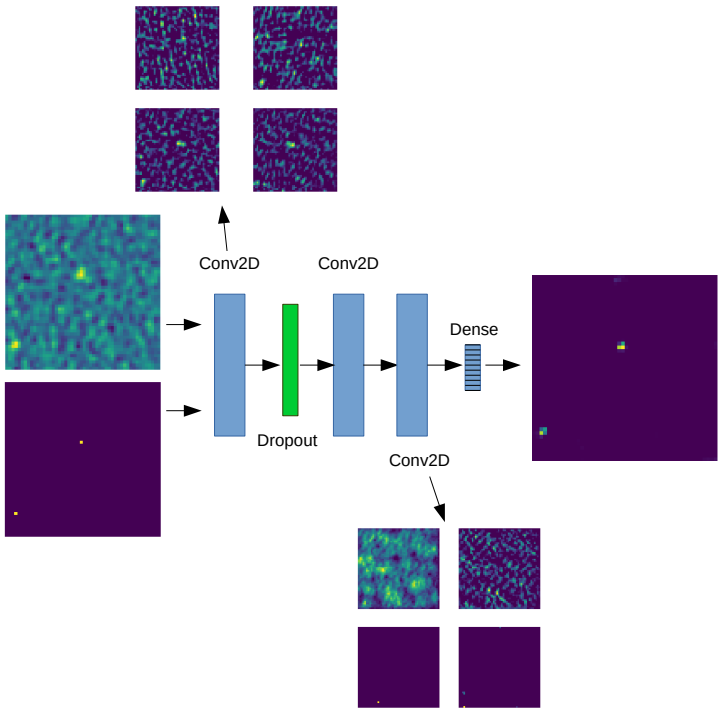

Figure 4.1: AutoSource architecture and examples of inputs (real maps and solution maps), features detected at the output of the first and third convolutional layer, as well as the resulting reconstructed image of the solution map.

Table 4.1: Architecture of the AutoSource model

Layer	Output shape	# Params
Input_1	(None, 50, 50, 1)	0
conv2d_1	(None, 50, 50, 16)	800
dropout_1	(None, 50, 50, 16)	0
conv2d_2	(None, 50, 50, 32)	12,832
conv2d_3	(None, 50, 50, 64)	18,496
dense_1	(None, 50, 50, 1)	65
Total		32,193

4.2.2 Pre-processing

In order to ensure accurate flux values, we used the Primary Beam image and raw FITS files provided, and ran CASA (McMullin et al., 2007) to regrid the image and correct for the Primary Beam. The resulting FITS file was the one used to perform source-finding in AutoSource and PyBDSF. Some tests were done using the non primary-beam corrected FITS files and we observed no change in performance regarding source-finding ability.

To determine the background noise level in the image, we output the background rms maps when we ran PyBDSF, by specifying RMS_MAP=TRUE using the PROCESS_IMAGE command.

To perform source-finding in PyBDSF, we ran the PROCESS_IMAGE command using the default parameters of THRESH_ISL=3.0 and THRESH_PIX=5.0.

4.2.3 Dataset generation

The solution maps have been generated using the training set files across each frequency. Since we are only interested in the source-finding, we took note of the corresponding (x,y) positions of each source in the training set. We focused only on the sources which could be found given the noise. The source locations have been inserted into the solution maps as single pixels, under the condition that the sources have a flux above a certain threshold (when the sources have flux greater than one, two, and five times the mean noise level, referred to as SNR=1,2 and 5.)

The SS, FS and SFG sources are encoded using the integers 1,2 and 3 respectively in the original solution map. This is so we can calculate the recovery of each of these classes of sources when testing AutoSource and PyBDSF. However, the solution map used for training has all the sources encoded with 1, irrespective of class. Tables 4.2 and 4.3 show the number of each class of sources across SNR=2 and SNR=5 respectively. It should be noted that there are many more SFG sources compared to SS and FS sources, which is why we focus on augmenting those source types to see if it improves the performance of AutoSource. There are fewer sources available at higher SNRs compared to at lower SNRs, since the threshold for inserting sources into the solution map is a lot higher.

Given there are very few sources available in the B5 dataset as shown in Tables 4.2 and 4.3, we focus our attention on the B1 and B2 datasets only.

We verified that the noise level in the maps is uniform. Table 4.4 shows there are only small proportional differences in the number of solutions obtained when taking the individual quartile cut-offs versus using the cut-offs derived from the whole training set area.

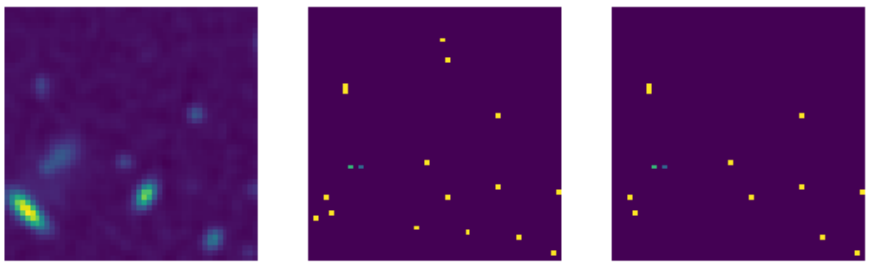

Figure 4.2: Left panel: Real map of a panel containing a combination of SFGs, SS and FS sources at B1 at 1000 h. Middle panel: Solutions at SNR= 2. Right panel: Solutions at SNR=5. The yellow, blue and green pixels indicate SFGs, SS and FS sources respectively. In this particular case both the SS and FS sources are very close together and very faint, which presents a challenge for both source-finders. The panels have a side length of 50x50 pixels.

Figure 4.3: Left panel: Real map of a panel containing a combination of SFGs, SS and FS sources at B2 at 8 h. Middle panel: Solutions at SNR= 2. There are two SFGs and one each ofSS and FS galaxies. Right panel: Solutions at SNR=5. At this SNR, only one SFG and one SS source remain; the other SFG and FS sources had a total flux that was lower than the cut-off threshold at that SNR ratio. The yellow, blue and green pixels indicate SFGs, SS and FS sources respectively. The panels have a side-length of 50x50 pixels.

The left panels of Figures 4.2 and 4.3 show a section from a real map of B1 at 1000 h and B2 at 8 h respectively, containing SFGs, SS and FS sources, along with the solutions injected at an SNR of 2 and 5. The smaller the SNR, the more sources will appear in the solution map, that look increasingly less obvious as they would be getting mixed with the noise background. Conversely, the larger the SNR, the fewer sources in the solution map, and only increasingly large and/or bright sources will appear.

To generate input image data for AutoSource, we divided the training set area (4000x4000 pixels) into 50x50 pixel blocks and moved these blocks across by increments of 50 pixels (resulting in 6,400 images), ensuring that the segmented blocks cover the entire area. The blocks may contain sources located on their boundaries, however all parts of the sources are accounted for. We have also experimented with using 20 pixel increments (resulting in 39,204 images) instead of 50 pixel increments, such that the same part of a source is seen across at least one other block and therefore sources on the boundary in one block will not be on the boundary in a neighbouring one, and noted no significant improvement in results.

In the 4000x4000 pixel area, there are 6400 images for training and testing altogether, with 5120 (80%) for training and 1280 (20%) for testing. We also investigate how much the results can be improved when using image augmentation, so 5120 is the minimum number of images with which we train. Similarly, we generated the input solution data for AutoSource by inserting individual pixels to represent the true location of the source, with the position obtained from the training set.

The image data has been multiplied by 10e6 because the pixel values are the surface brightness, which can be very small with O(10e-6) in magnitude, and they can also be negative. Applying a scaling to the original values ensures there is sufficient contrast between them, which facilitates detection by the autoencoder. The scaling is also done to match the order of magnitude of the values in the solution maps, which were generated by inserting '1' against a background of '0'. We note that we also experimented with multiplying the data by 10e9 and found no noticeable difference.

We use only the training set region out of the whole map, which consists of a 4000x4000 pixel area across the B1 and B5 datasets, and 4200 pixel area across B2, as shown in Table 4.5. It should be noted that the same area is not covered between the three frequencies - however it is the same within the same frequency between the three integration times.

The solution maps have been generated in the same way as the input image maps, using 50x50 pixel blocks with increments of 50 pixels, where a '1' has been inserted at the location of the centroid position of the source. The blocks that contain no solutions are empty 50x50 blocks. The source selection has been subject to a flux threshold, where only sources having a flux greater than 1,2 or 5 times the background for each map have been selected. The background maps have been determined using PyBDSF. Figure 4.4 shows the segmentation of part of the training area into 50x50 pixel blocks, for both the original primary-beam corrected FITS file as well as the corresponding solution map generated.

We ran PyBDSF with the default parameters in order to perform source finding across the whole map, which was later subset to only include the training set area in the images.

4.2.4 Image augmentation

Deep learning techniques are able to take advantage of image augmentation as it generates more training samples, which should improve the performance up to some threshold (Krizhevsky et al., 2012). Since there are many fewer steep-spectrum (SS) and flat-spectrum (FS) AGN sources compared to star-forming galaxies (SFGs), we wanted to see whether we could improve on the metrics for these types of sources if we augmented the images that contained them. We employed vertical and horizontal flipping, and rotation by 90, 180 and 270 degrees. The results show the metrics when applying no augmentation, augmenting the

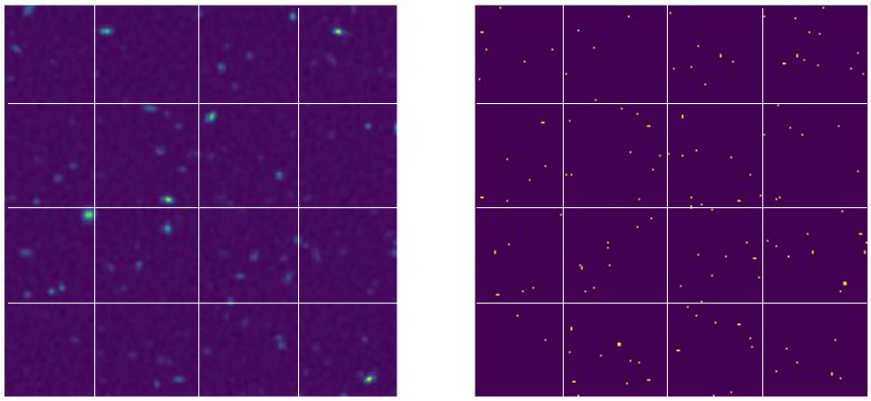

Figure 4.4: Left: Segmentation of a portion of the primary-beam corrected images in the training set area. Right: Segmentation of the solution map in the same area. These images are generated from the B1 1000 h dataset, using a SNR=5 to determine threshold of flux for injecting the solutions. Each block forms a single 50x50 pixel image that is input into the AutoSource algorithm; the blocks on the left make up the training set images (train_X), and the blocks on the right make up the solution set images (train_Y).

Table 4.2: The total number of steep-spectrum AGN, flat-spectrum AGN and star-forming galaxies across each integration time across all frequencies, when using SNR=2.

Dataset	#SS-AGN	# FS-AGN	# SFG
B1			
8 h	342	117	13920
100 h	644	386	34158
1000 h	957	682	57797
B2			
8 h	91	64	4028
100 h	166	151	9423
1000 h	278	294	17283
B5			
8 h	3	1	26
100 h	4	2	103
1000 h	6	6	223

Table 4.3: The total number of steep-spectrum AGN, flat-spectrum AGN and star-forming galaxies across each integration time across all frequencies, when using SNR=5.

Dataset	#SS-AGN	# FS-AGN	# SFG
B1			
8 h	213	94	5717
100 h	395	208	16885
1000 h	605	366	31597
B2			
8 h	59	25	1877
100 h	101	73	5096
1000 h	178	155	10251
B5			
8 h	3	1	7
100 h	4	1	43
1000 h	4	3	114

Table 4.4: Percentage difference in the number of sources depending on whether the quartile threshold from the training set are taken versus using the threshold obtained from the training set as a whole, at an SNR=5.

Frequency	8 h	100 h	1000 h
B1	4.4	3.7	2.6
B2	4.5	3.6	2.8

Table 4.5: The x and y ranges of the training area, according to the locations within the whole map.

Frequency	x range	y range	Area
B1	16300 - 20300	16300 - 20300	4000 pixels sq.
B2	16300 - 20500	16300 - 20500	4200 pixels sq.

SS and FS sources, as well as augmenting all sources. There would be little merit in explicitly augmenting the SFGs because they tend to appear more point-like.

4.3 Results

The results presented are the summary metrics of the sources across the different classes; SS, FS and SFG sources, and all sources as a whole.

The original reconstructed image output of AutoSource is composed of continuous pixel values which are mainly close to zero. To determine the output predictions for the source locations, we define a reconstruction threshold that ranges between 0 and 1. Then we choose the value across all metrics depending on which reconstruction threshold produces the highest F1 score. PyBDSF outputs only 0's for sources not found and 1's for sources found.

We allow a leniency of 3 pixels for the positions of sources found. To calculate the metrics, we make the following definitions:

- TP: sum of pixels with values greater than the reconstruction threshold in the reconstructed solution map that is **less than** 3 pixels of a source in true solution map
- FP: sum of pixels with values greater than the reconstruction threshold in the reconstructed solution map that is **equal to or greater than** 3 pixels of a source in true solution map
- TN: sum of pixels with values lower than the reconstruction threshold in the reconstructed solution map that are **also** empty in the true solution map
- FN: sum of pixels with values lower than the reconstruction threshold in the reconstructed solution map that are **not** empty in the true solution map

where TP refers to the true positives, TN refers to true negatives, FP refers to the false positives and FN refers to false negatives.

Given that source-finding in the current work is defined as being directly related to the sum of pixels output by the source-finders, the sum of the sources detected between the source-finders is not expected to be constant.

Since the true catalogue is quite richly populated with sources, and given the 3 pixel leniency, there could be some sources that are found across both algorithms by chance. Ideally there should be zero chance findings, but in reality there will be some small fraction. We use the SNR=1 dataset to test this effect, as this dataset has the highest population of sources in the solution map (and also in the reconstructed map). Therefore, the SNR=1 dataset can be considered to be the worst case scenario for chance matches. The effect of chance matches

is tested by randomly rotating the reconstructed solution maps, comparing it with the real solution map and calculating the metrics, as for the rest of the results.

The precision, recall and F1 score metrics, in the form of bar-plots are provided in the current work. We do not include the accuracy because of the way the true negatives are defined; the value is always very high leading to accuracies greater than 99% across both source-finders. The metrics are defined in Equations 4.1 - 4.4.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4.1}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4.2}$$

$$\text{F1 score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.3}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}, \tag{4.4}$$

The bar plots of the F1 score, defined as the harmonic mean across precision and recall, are provided in the main text, as well as a subset of the confusion matrices. The precision and recall bar-plots are given in Appendix 4.5.

We experimented with using pooling layers, by applying it to the B1 8 h dataset with no augmentation. Pooling reduces the dimensionality of the layer by outputting the average pixel value across some area whose size is defined by the user. Two different architectures were considered: placing a pooling layer after the first, or after the second convolutional layer respectively. Due to the halved dimensions in the architecture as a result of pooling, an upsampling layer had to be inserted prior to obtaining the output. In both cases, the resulting metrics were all inferior to the equivalent model without pooling. The source locations tended to be less precise and generally spanned an area of 4 square pixels, most likely because the pooling operation lost the precise source location.

The use of pooling resulted in AutoSource identifying no true positives, and it generated a few false positives due to the reconstruction of the source positions along the edges of the image only. The likely explanation is that the true signal from the source only occupies a small area, therefore when pooling is used it can ‘wash out’ these pixels, in some cases causing the source to become lost among the background.

We also omit the results across the B5 dataset because both source-finders failed to recover any sources across any integration time. This is most likely because of the dataset being the noisiest, as well as having very few sources in the catalogue.

4.3.1 Very low significance source metrics at SNR=1

Figure 4.5 shows the F1 score metrics across the different classes of sources, as well as when all are considered together. AutoSource almost always performs better across the SFGs and all sources in the B1 dataset, for all integration times, whereas PyBDSF performs better for the remaining datasets (SS and FS sources across B1 and B2, and SFGs and all sources at B2.)

The better performance of AutoSource across the SFGs and all sources at B1 is most likely due to the effect of chance matches, as shown in Figure 4.6, which shows the source-finding metrics when the predicted source locations are randomised, to see how many sources are found due to chance. On average, AutoSource is more affected by chance matches across the SFGs and all sources. Some possible causes of the increased chance matches in AutoSource are that the SFGs are highest in number, and that many sources found tend to be spread over several pixels rather than confined to one. On the other hand, PyBDSF is more affected on average by chance findings across the SS and FS sources. In AutoSource, at worst the chance matches reach up to $\sim 26\%$ compared to real findings, whereas in PyBDSF the effect is more pronounced in the datasets where fewer sources are found overall. The worst case for PyBDSF is across the SS sources in the B2 100 h dataset, where there are barely more real matches compared to chance matches. It should also be noted that the SNR=1 dataset is the noisiest one that also has the most densely populated solution and reconstruction maps, which maximises the risk of chance findings, therefore represents the worst case scenario in terms of datasets. We further note that the sources have very low significance at this SNR.

An improvement in the F1 score across the SFGs can be observed due to augmenting the images containing all sources, since the vast majority of all sources are SFGs. However, augmenting the SS and FS sources does not improve the SFG scores by much since we are not giving the network more examples of SFGs to train on. Image augmentation does not have the same effect on the randomised data, as shown in Figure 4.6.

Figure 4.7 may indicate possible reasons why the AutoSource performance is poorer overall compared to PyBDSF, at an SNR=1. Since the solutions are injected into the map at the threshold of the mean background noise level, there appear to be solutions that are not obvious by eye, and can become confused with the background noise. It is therefore possible that AutoSource has not successfully learnt to extract the sources at this SNR. For the examples given, there is one SS source in each map, while the rest are SFGs. Both PyBDSF and AutoSource recover the SS source in the top row, whereas AutoSource finds a false positive and misses other SFGs. PyBDSF recovers one of the SFGs successfully but misses the others. Neither PyBDSF nor AutoSource recover the SS source in the bottom row, and AutoSource partially recovers one of the SFGs however the location is spread out over several pixels. It

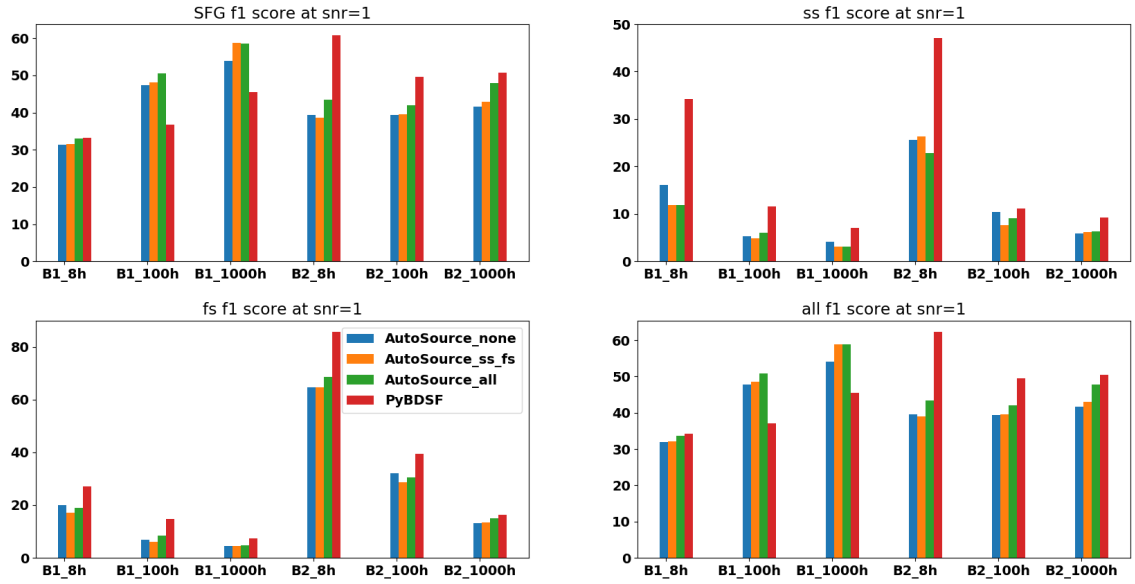

Figure 4.5: F1 scores at SNR=1, across the two frequencies B1 (560MHz) and B2 (1400MHz) and the three integration times. There are three results given from AutoSource, depending on the augmentation used when training. The blue bar represents no augmentation, orange represents augmenting the SS and FS sources, and the green bar represents augmenting all sources. The graphs show that PyBDSF usually performs better compared to AutoSource at this SNR. Although it appears that AutoSource performs better across the SFGs and all sources in the B1 dataset, for all integration times, the better performance appears to be explained by the increased proportion of chance matches at this SNR, as shown in Figure 4.6. However, it should be noted that these sources have very low significance given the SNR.

misses the other SFGs and gets a couple of false positives. PyBDSF recovers only one SFG in this example, and misses the others which result in a number of false negatives.

4.3.2 Low significance source metrics at SNR=2

Figure 4.8 shows the F1 score metrics across the different classes of sources, as well as when we consider them all together.

Across the SFGs/all sources at SNR=2, AutoSource performs better on average, where now it recovers these sources better in the B2 8 h dataset, where one example is shown in Figure 4.9. However, PyBDSF generally performs better across the SS and FS sources.

Considering the F1 scores of the SS sources in the 8 h datasets, the augmentation of either the SS/FS or all sources worsens the score, most likely because this dataset is the noisiest of the three, therefore some signal would become lost in the noise. There are generally slight

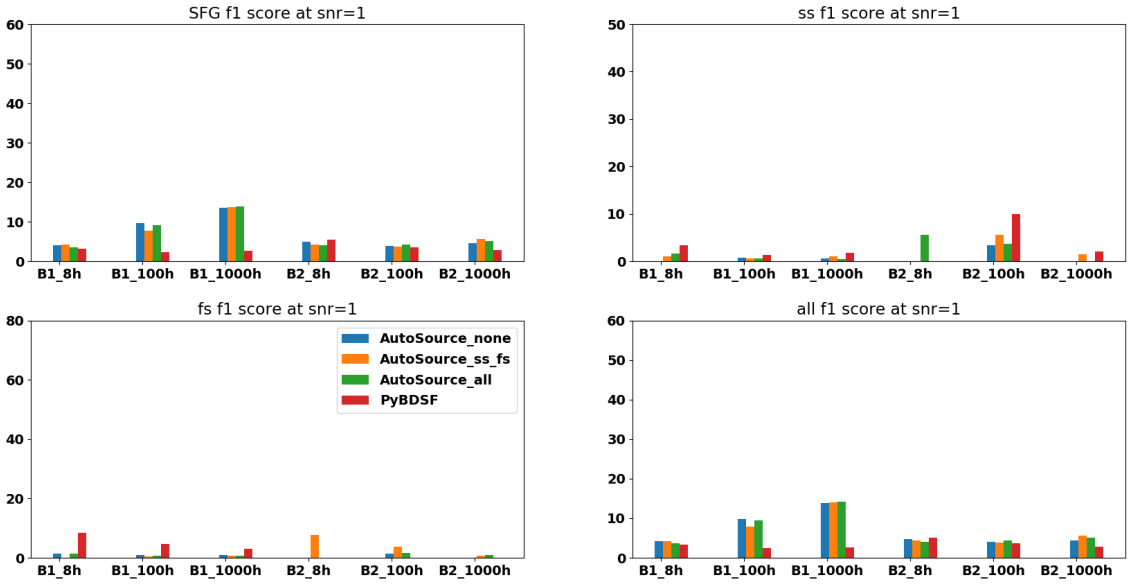

Figure 4.6: Showing the effect of randomly rotating the reconstructed matrix of source locations to investigate the proportion of chance findings.

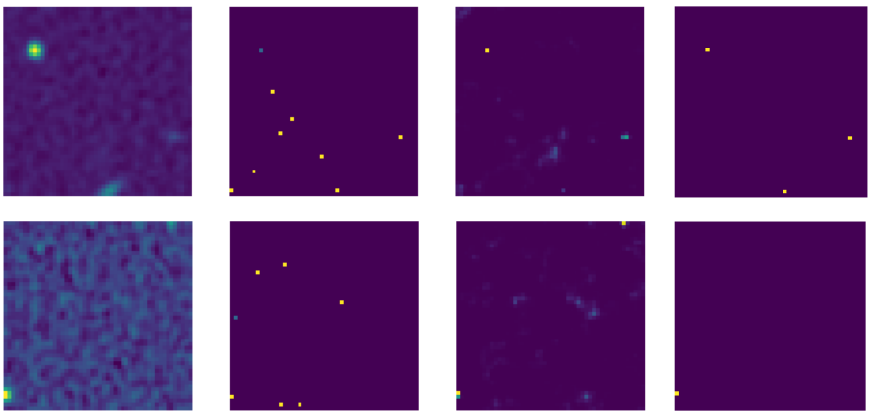

Figure 4.7: The top and bottom rows show a couple of examples of (first column) a real map for B1 at 8 h, (second column) the solutions when injected into the map given the SNR=1 threshold, (third column) the predicted locations by AutoSource after training on the original images only, and (fourth column) the predicted locations by PyBDSF.

improvements with augmentation of the SS/FS at the other two integration times, as the noise is reduced. The SS sources are the ones with the most varying morphology within the class (they have the greatest amount of extended emission and give rise to FRI/FRII type structures), however there are not many original examples of these. Additionally, the signal threshold is set at only twice the noise threshold so there are more solutions in the map, increasing the risk of sources being contaminated with noise. PyBDSF clearly outperforms AutoSource across the SS sources.

Across the FS sources, there are two datasets in which AutoSource performs better than PyBDSF (B2 at 8 h and B2 at 100 h), however for the remainder it does slightly worse than PyBDSF. The augmentation of the SS/FS sources always improves the F1 score across these FS sources, however it does not always improve when augmenting 'all' sources since most of these sources are SFGs, therefore proportionally there are fewer SS/FS sources to train on. We note that when using AutoSource the performance is better across the FS sources as these sources have a more defined morphology, which tend to be more compact compared to that of the SS sources.

A similar pattern is observed at the SNR of 2, as was observed at SNR=1 in regard to the effects of augmentation, where augmenting sources of the same type results in improved metrics for those sources.

Table 4.6 shows the confusion matrices across the B1 and B2 datasets for the 8 h and 1000 h integration times, when comparing the test results after using AutoSource trained on the augmentation of all sources, against PyBDSF. We exclude the true negative counts for brevity as this denotes the total number of pixels where there is no solution as well as no predicted source. Given that AutoSource sometimes produces reconstructed solutions that are spread over several pixels and that true positives are defined as matches that occur over less than 3 pixels of the true solution locations, AutoSource detects more true positives. However, it also detects more false positives compared to PyBDSF, but fewer false negatives. Therefore, it misses fewer sources compared to PyBDSF.

4.3.3 High significance source metrics at SNR=5

The opposite trend to what was observed at SNR=2 is seen at SNR=5, as shown in Figure 4.10 where now PyBDSF performs better on the SFGs/all sources on average, whereas AutoSource performs better on average on the SS and FS sources.

Therefore when there is a higher signal to noise, AutoSource can better extract the SS/FS sources compared to the SFG/point sources. For the majority of times, better results are achieved when augmenting either the SS and FS sources, or all. Whereas when the signal to noise is lower, the performance of AutoSource across these extended sources suffers, probably

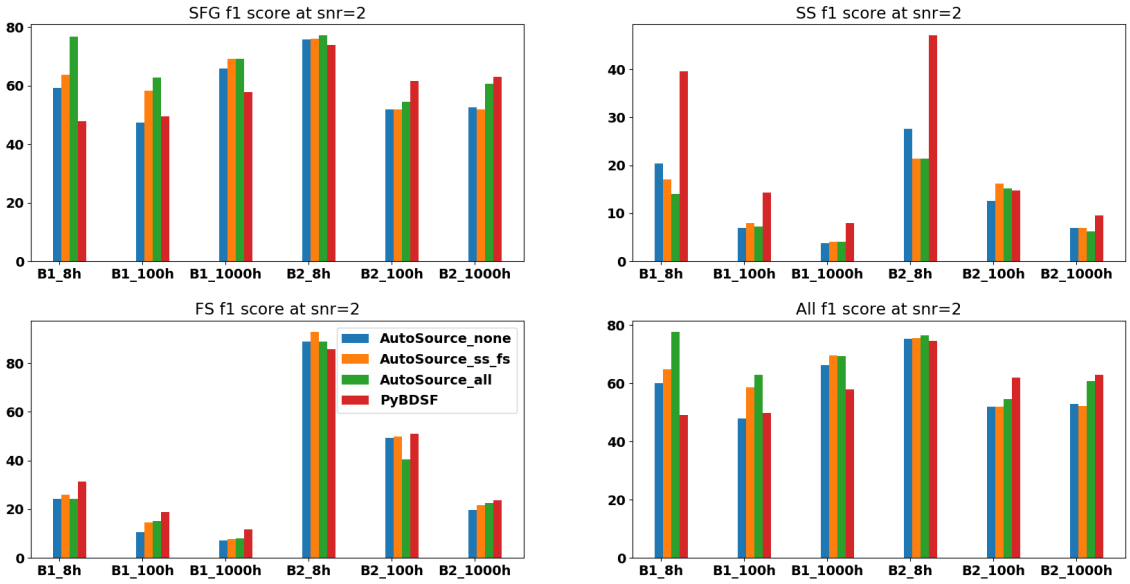

Figure 4.8: F1 scores at SNR=2, across the two frequencies B1 (560MHz) and B2 (1400MHz) and the three integration times. There are three results given from AutoSource, depending on the augmentation used when training. The blue bar represents no augmentation, orange represents augmenting the SS and FS sources, and the green bar represents augmenting all sources.

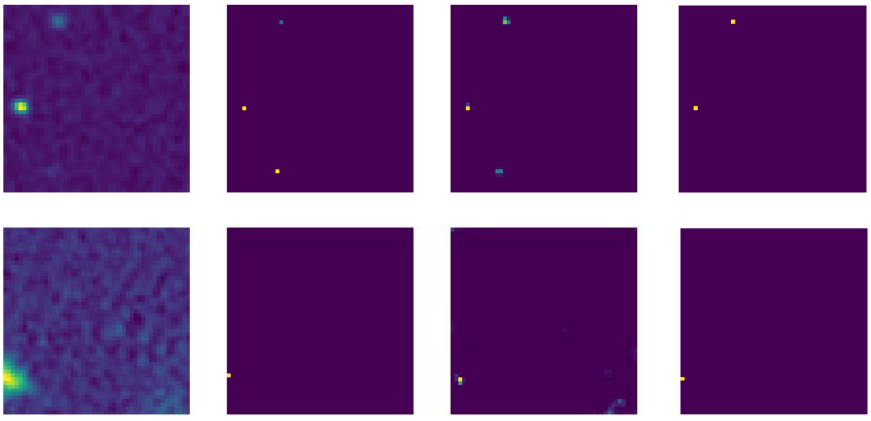

Figure 4.9: The top and bottom rows show a couple of examples of (first column) a real map for B2 at 8 h, (second column) the solutions when injected into the map given the SNR=2 threshold, (third column) the predicted locations by AutoSource after training on the original images only, and (fourth column) the predicted locations by PyBDSF.

Source type	SFG			SS			FS			All				
Method	tp	fp	fn	tp	fp	fn	tp	fp	fn	tp	fp	fn	fp/tp	fn/tp
B1_8 h														
(3)	1473	635	261	23	282	0	26	163	0	1522	561	316	0.37	0.21
(4)	314	73	611	19	57	1	8	35	0	341	46	663	0.14	1.94
B1_1000 h														
(3)	5351	2735	2026	58	2722	0	68	1551	0	5477	2592	2235	0.47	0.41
(4)	3326	506	4333	57	1306	0	50	765	0	3433	429	4555	0.13	1.33
B2_8 h														
(3)	628	52	319	3	22	0	12	3	0	643	56	340	0.09	0.53
(4)	130	13	79	4	9	0	9	3	0	143	8	89	0.06	0.62
B2_1000 h														
(3)	2476	1593	1608	11	330	1	42	289	0	2529	1531	1734	0.61	0.69
(4)	1897	290	1932	12	226	1	31	199	0	1940	245	2050	0.13	1.06

Table 4.6: Showing all of TP, FP, TN and FN across (3) = AutoSource augment all, (4) = PyBDSF, at B1 8 h, B1 1000 h, B2 8 h and B2 1000 h, at an SNR=2.

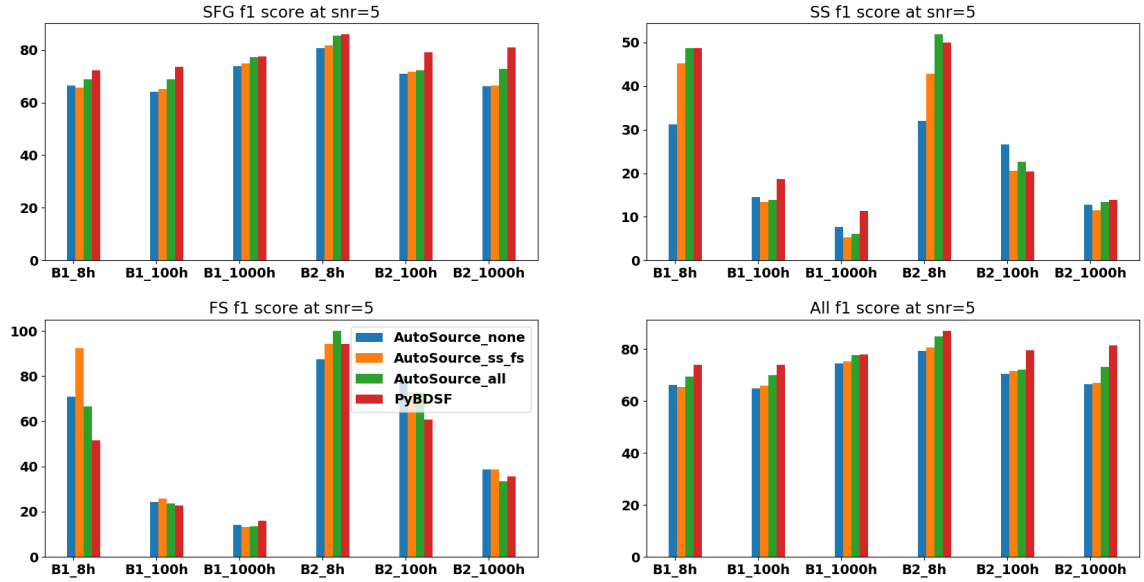

Figure 4.10: F1 scores at SNR=5, across the two frequencies B1 (560MHz) and B2 (1400MHz) and the three integration times. There are three results given from AutoSource, depending on the augmentation used when training. The blue bar represents no augmentation, orange represents augmenting the SS and FS sources, and the green bar represents augmenting all sources.

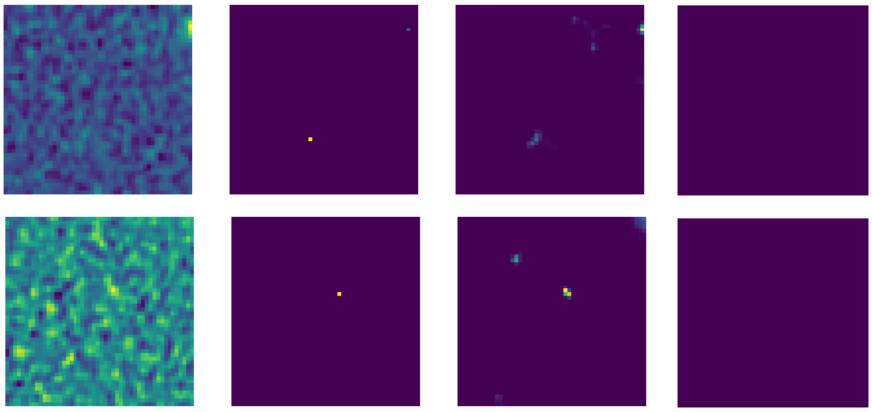

Figure 4.11: The top and bottom rows show a couple of examples of (first column) a real map for B2 at 8 h, (second column) the solutions when injected into the map given the SNR=5 threshold, (third column) the predicted locations by AutoSource after training on the original images only, and (fourth column) the predicted locations by PyBDSF.

because the emission from them tends to become lost in the noise, whereas the SFGs are recovered better compared to when using PyBDSF at lower SNR ratios.

Figure 4.11 shows that AutoSource can recover the SFG and SS source in the top row, as well as the SFG in the bottom row, however at the expense of a couple of false positives. Meanwhile, PyBDSF does not recover any sources.

Table 4.7 shows the confusion matrices at SNR=5, for the same datasets and runs as was included for SNR=2. There are fewer sources found by the source-finders overall as the SNR is higher compared to before, however a similar trend is seen to before, where AutoSource finds more true positives and false positives, whereas PyBDSF finds fewer true positives but more false negatives. Although, the ratio is not as pronounced when compared to what was observed at SNR=2, as the signal-to-noise is now higher.

Figure 4.12 shows the training and validation losses across the B1 and B2 frequencies, across all integration times. In the left panel (B1 frequency; 560MHz) the training and validation losses are roughly at the same level across the 8 h and 100 h integration times, whereas there is some underfitting observed in the 1000 h dataset. The underfitting generally indicates that a more complex architecture should be tried. In the right panel (B2 frequency; 1400MHz) the 8 h integration time loss curves are at the same level, whereas there is some level of overfitting observed across the 100 h and 1000 h integration times. The B2 dataset on average is noisier compared to the B1 dataset. It is interesting to note that for the same integration times across the two different frequencies, the same model tends to underfit on one dataset and overfit on the other. This indicates that using the same model across all frequencies and integration times is not ideal, that instead each model should be tuned to the specific dataset at hand. Nonetheless, the resulting metrics are still competitive with that of PyBDSF, and

Source type	SFG			SS			FS			All				
Method	tp	fp	fn	tp	fp	fn	tp	fp	fn	tp	fp	fn	fp/tp	fn/tp
B1_8 h														
(3)	444	175	225	20	41	1	11	11	0	475	164	256	0.35	0.54
(4)	304	60	172	18	38	0	8	15	0	330	40	193	0.12	0.58
B1_1000 h														
(3)	3478	1422	629	35	1075	0	45	573	0	3558	1311	738	0.37	0.21
(4)	3070	663	1124	57	892	0	45	473	0	3172	554	1247	0.18	0.39
B2_8 h														
(3)	332	44	70	7	13	0	8	0	0	347	47	80	0.14	0.23
(4)	128	13	29	4	8	0	8	1	0	140	8	34	0.06	0.24
B2_1000 h														
(3)	1980	974	514	13	168	0	29	115	0	2022	923	567	0.46	0.28
(4)	1857	280	587	12	149	0	28	102	0	1897	224	637	0.12	0.34

Table 4.7: Showing all of TP, FP, TN and FN across (3) = AutoSource augment all, (4) = PyBDSF, at B1 8 h, B1 1000 h, B2 8 h and B2 1000 h, at an SNR=5.

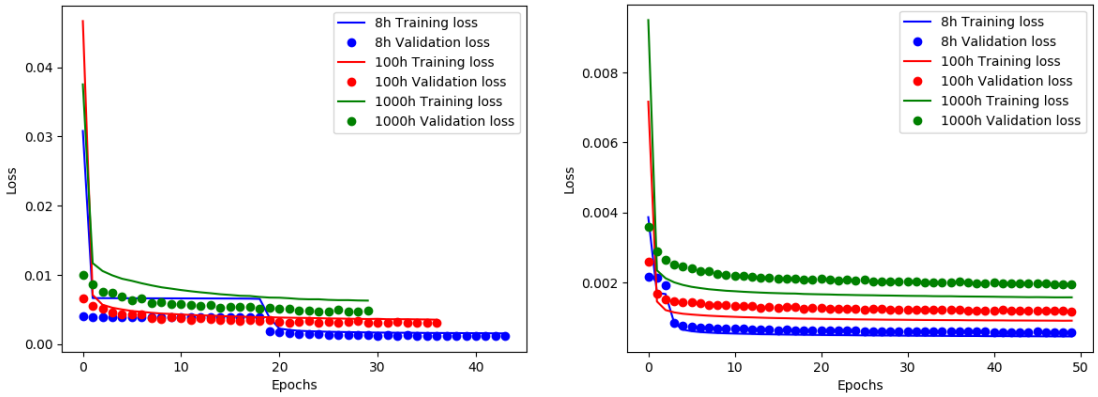

Figure 4.12: Training and validation losses across the 3 integration times at SNR=5 across B1 and B2 datasets in the left and rights panels respectively.

have the potential to outperform PyBDSF given a more optimally tuned model.

4.3.4 Execution times

Both source finders were run on a computing cluster using CPUs from 27 available Intel XEON CPU nodes, with a 3.5 GHz processor. There are six available cores per node.

Table 4.8 shows the source-finding execution times across AutoSource when performing no augmentation, augmenting the FS and SS sources, and augmenting all sources. PyBDSF takes more than two times longer to run compared to the AutoSource run when all images are augmented. The execution times across AutoSource are subject to variability depending on how many sources there are to augment, as well as the total training time, which depends

	B1_8 h (mins)
AutoSource none	27.5
AutoSource SS+FS	23.2
AutoSource all	63.6
PyBDSF	167.0

Table 4.8: Running times across AutoSource when augmenting different sets of images, as well as PyBDSF.

on the total number of images and epochs. The run where the SS and FS sources were augmented took a shorter time to train and test compared to the one where no augmentation was used, because there were more epochs of training completed; the run that did not utilise augmentation was affected by the early stopping condition at an earlier point during training.

4.4 Discussion and Conclusions

In the current work we have shown how the use of a simple autoencoder composed of three convolutional layers, a dropout layer and a dense layer, as shown in Figure 4.1 and Table 4.1, can be competitive with a state-of-the-art source-finder; PyBDSF. Both approaches have been tested across different frequencies, integration times and signal-to-noise ratios, and the recovery metrics across the different source types of SFGs, SS-AGN and FS-AGN sources were derived. The code used to obtain both the AutoSource and PyBDSF results is available on Github[5]. Given that AutoSource outputs continuous values in the reconstruction of the solution map, as defined by a reconstruction threshold that ranges between 0 and 1, whereas PyBDSF uses a fixed threshold, AutoSource could be more flexible as a method as it attributes a probability to finding a source at a particular location.

AutoSource also sometimes outputs the source location spread over a few pixels rather than being localised to a single one, which may provide additional information about the source; for example it could be more extended or diffuse. The fact that AutoSource spreads out the source location over several pixels, which occurs more frequently at the lower SNR ratios and at shorter exposure times, where there are more sources present and their emission is more likely to get mixed with the noise, results in more true positives and fewer false negatives. However, at the same time AutoSource also produces a larger number of false positives compared to PyBDSF. A similar trend is seen at higher SNR ratios, although fewer true positives, false positives and false negatives are found by both source-finders in comparison. For example, the SNR=5 dataset has fewer solutions but also the strongest signal. On the other hand,

[5]https://github.com/vlukic973/AutoSource

PyBDSF misses many more sources compared to AutoSource, as the false negative counts are almost always higher.

It is interesting to note that the metrics across the SS and FS sources tend to be relatively low across both PyBDSF and AutoSource. In fact, they decrease with increasing integration time, across all SNRs, with the dataset at the lowest frequency (B1) attaining the lowest metrics overall. Possible reasons could be that the SS and FS sources are smallest in number and their morphology is revealed as increasingly variable, as more extended emission is detected with the longer integration times.

In regard to how well the two methods extract SFGs, SS, FS and all source types combined across the SNRs, we see that PyBDSF performs better on average compared to AutoSource at SNR=1. AutoSource appears to be more severely affected by chance matches at this SNR compared to PyBDSF, however the sources have very low significance. In contrast, AutoSource is better at extracting the SFGs and all sources at SNR=2, whereas PyBDSF is better at extracting these at SNR=5. AutoSource is better at extracting the FS sources at an SNR of 5, whereas PyBDSF is better for the FS sources at SNR= 2. AutoSource is worse at extracting the SS sources at an SNR of 2, however half the time it is better than PyBDSF at extracting them at an SNR of 5.

We have seen that image augmentation improves the AutoSource performance when the relevant sources are augmented; generating more 'all' sources tends to improve the metrics across SFGs and 'all' sources as these sources are largely made up of SFGs, and generating more SS and FS sources tends to improve their recovery, but not that of SFGs and all sources. Augmentation may also not work to improve the results as expected when the datasets are noisier, the sources are few in number, or if their morphology is ambiguous.

PyBDSF takes longer to run in total (167 mins for the B1 dataset at 8 h), however it outputs characteristics of the source such as size, flux, among other properties, whereas AutoSource outputs the positions only.

Across the results for the low significance source metrics at SNR=2 and high significance source metrics at SNR=5, AutoSource usually outperforms or has very similar performance metrics to PyBDSF across the shortest integration time datasets (8 hrs). This may indicate that it can more successfully model the noise at these SNRs and integration time compared to PyBDSF. The only times that AutoSource performs visibly worse is in B2 at 8 hrs across the SS sources at an SNR=2, and across all B2 at 8 hrs at SNR=1. It appears that AutoSource has trouble modeling the noise as the SNR ratio decreases, especially for sources with more extended emission. Potential ways to improve the performance of AutoSource at lower SNR ratios could be to use a more complex network, and train for more epochs with a greater reconstruction threshold when using early stopping. However, one of the purposes of the

current work was to show how a simple convolutional network architecture can be used for source-finding in radio astronomy.

The injection of sources and, in turn, the ability to be found by the source-finders, largely depends on the characterisation of the background noise signal. In the current work, we use PyBDSF to estimate the background noise, therefore if there are more false negatives/positives these missed/extra sources will be contaminating the background signal to some extent. Sources displaying a more compact morphology are unlikely to affect the background signal by much, since the emission is localised to a very small area. However, the effect will be larger the more extended the source is. Some extended sources may have very faint and/or diffuse emission which can mingle with the noise.

It appears that AutoSource performs better overall at larger SNRs and shorter integration times compared to PyBDSF, most likely because it has learned to model the noise in these images better and the sources show a greater contrast against the background. The ratio of false positives to true positives is larger for AutoSource, however the ratio of false negatives to true positives is larger for PyBDSF. Therefore, AutoSource and PyBDSF perform better in terms of recall and precision respectively. As the SNR increases, AutoSource becomes increasingly better at recovering the extended (SS and FS) sources and tends to outperform PyBDSF across most datasets at the highest SNR of 5. However, at the same time PyBDSF becomes increasingly better at recovering the SFGs and sources as a whole. With a decreasing SNR, AutoSource is increasingly successful at recovering the more compact sources (SFGs) and all sources, whereas it performs worse with the extended sources, most probably because it has not successfully learnt to extract the extended source signals out from the noise at lower SNRs, on which PyBDSF does better.

Given that AutoSource tends to perform better in terms of recall (as shown in Appendix 4.5) overall compared to PyBDSF (therefore it finds fewer false negatives and hence picks up some sources that PyBDSF has missed), it could be used as part of a pipeline where AutoSource is run first to find the sources, then PyBDSF is run to extract the precision values for these sources, perform further filtering as well as characterise the sources.

The next step in developing AutoSource would be to derive properties from the sources found. One way to do this may be to correlate the features detected by lower layers to the values given in the catalogue, for example to match the total flux for a source in question to the emission detected by one of the feature maps. Previously we had attempted a regression technique to see if it could learn the continuous values provided in the catalogue, however our network failed to learn any property successfully. AutoSource could also be made up of individual models that are targeted to the dataset at hand, where the training and validation losses are better matched. Another possible extension to the current work would be to train an autoencoder to remove noise from data by generating 1000h maps from 8h or 100h ones.

4.5 Appendix

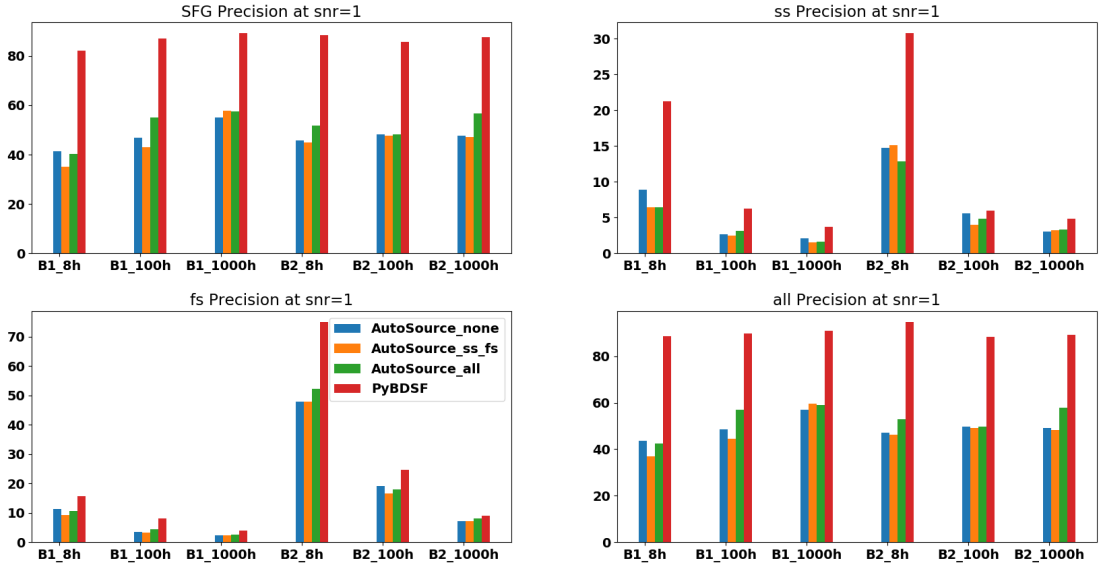

Figure 4.13: Precision values at SNR=1.

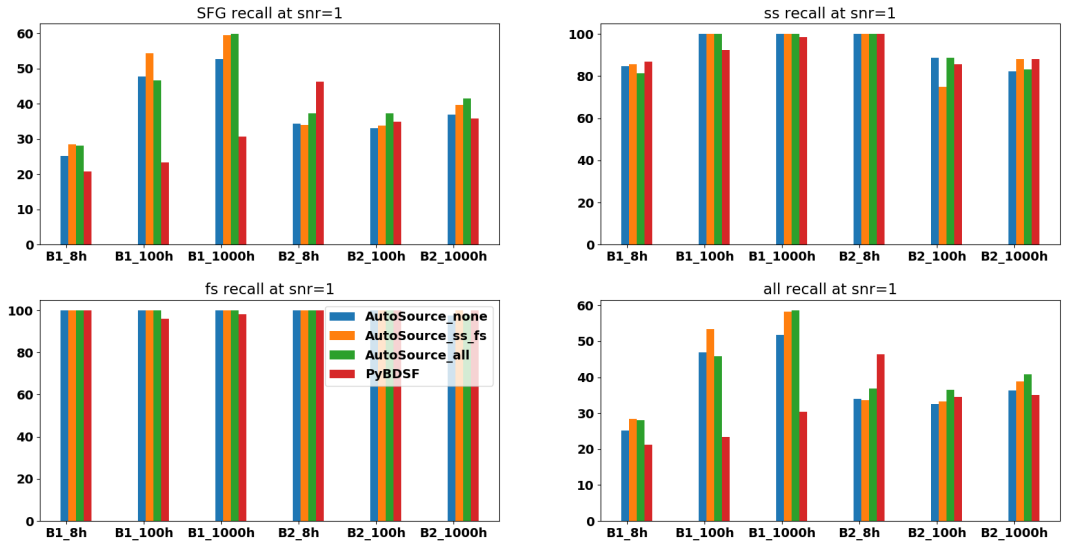

Figure 4.14: Recall values at SNR=1.

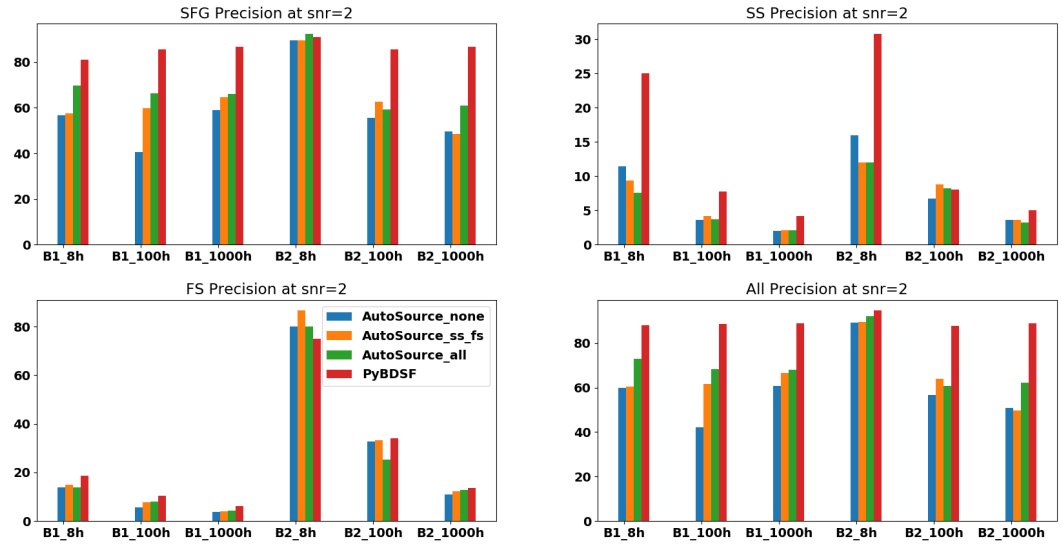

Figure 4.15: Precision values at SNR=2.

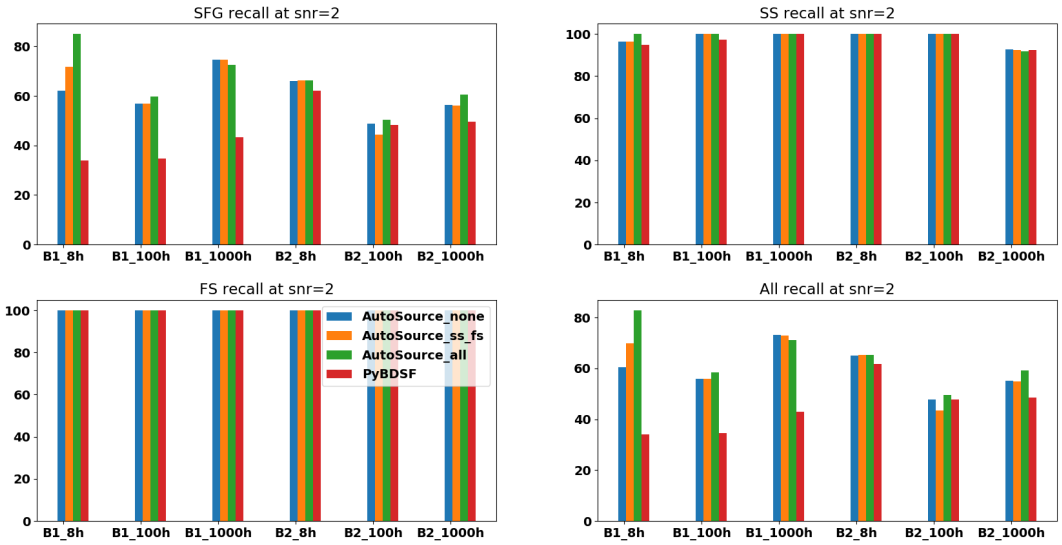

Figure 4.16: Recall values at SNR=2.

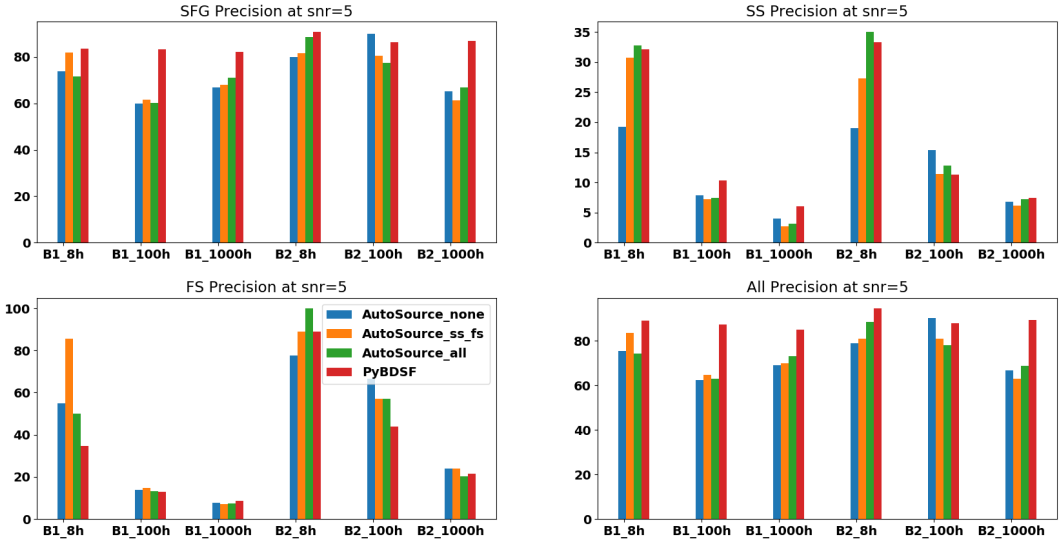

Figure 4.17: Precision scores at SNR=5.

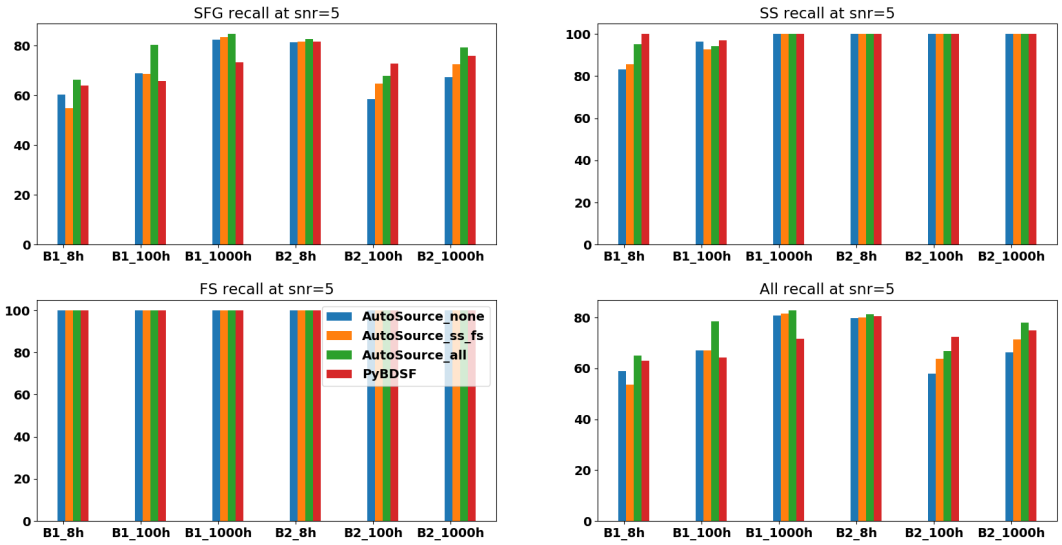

Figure 4.18: Recall scores at SNR=5.

5 Conclusions and Outlook

Radio astronomical continuum surveys are fundamental in furthering our understanding of the Universe as they facilitate the study of galaxy formation and evolution, and cosmology in general. Early radio surveys were able to provide conclusive evidence of the big bang model. Over the last few decades there has been an increase in the number of radio sources detected, owing to the increased capabilities of radio telescopes in terms of sensitivity and area. Larger samples of radio sources are very useful to enable better quality source population studies of the two fundamental galaxy types in terms of radio emission: AGN and SFGs, as well as understanding the evolution of the radio luminosity function. Surveys over the last few decades have provided larger samples of faint radio sources at higher redshifts, enabling a view into the earliest stages of cosmological evolution.

The cross-correlation of surveys between different wavelengths such as optical, infra-red and radio is important in gaining a more complete understanding of galaxies. For example, it is necessary to overlay radio and infrared emission to help determine whether components of radio emission belong to a single radio source, or whether they are independent sources. One important task in astronomy is that of classification In terms of radio sources, the range of radio galaxy classes in existence is indicative of different physical processes, such as those governing the power of the jets, how the radio galaxy is formed, what environment it exists in and what events preceded it.

The large number of radio sources detected will only increase in time, making manual analyses by astronomers increasingly obsolete. Large numbers of citizen scientists, who are not expert astronomers, can be taught to recognise patterns and therefore assist in characterising properties of astronomical objects in order to classify them. However, even the use of citizen scientists will be insufficient to keep up with the pace of increasing data volumes. Machine learning tools can be used, given their capacity to learn and distinguish features in data. The input from citizen scientists can be used as training set from which machine learning algorithms can learn from.

In regards to astronomy, machine learning tools have been proven useful in the estimation of photometric redshifts, classifying objects such as stars and galaxies, as well as different types of AGN, and the morphological classification of galaxies. A particular type of neural network – convolutional neural networks (CNNs) have had numerous successes in classification and

regression tasks in high-dimensional data such as image data. The use of traditional neural networks is generally unfeasible in the analysis of such data due to the size and complexity of the network, and hence number of parameters, needed to capture the patterns needed to discern features. CNNs on the other hand employ filters of smaller size that scan across the image and detect features, which greatly reduces the number of parameters. Additionally, the use of filters allows for parameter sharing and leads to translational invariance. Pooling layers summarise the information over a range of pixels before sending it to the following layer, which further decreases the number of parameters. The stacking of convolutional and pooling layers leads to a hierarchical extraction of features in images. The information is merged at the end and the network can calculate the error, based on the prediction and label. Machine learning will be the most indispensable tool for analysing large amounts of astronomical data in the future.

There are fundamental differences in images between astronomical/non-astronomical domains. Many neural network architectures have been developed to distinguish between everyday subjects such as dogs, cats and cars, of which there are thousands of classes and many examples within each class, and they tend to be well-defined in terms of appearance. Astronomical subjects on the other hand tend to have fewer classes and examples, and the subjects within the classes are usually more complex and abstract, as they can have very different emission patterns that also depends on the wavelength. As such, astronomical subjects generally need larger and/or higher resolution images to capture the differences within and between the classes. The fact that there are fewer examples of astronomical subjects leads to smaller training sets, hence motivating the use of citizen scientists to provide more labels. The difference in nature between astronomical and non-astronomical images is also an important consideration for applications of transfer learning, as the higher-level layers in neural networks trained on everyday images will need to be fine-tuned to capture the subtleties in classes based on the specific varieties of astronomical subjects.

The main focus of the current thesis has been on the classification of radio-loud AGN, namely radio galaxies. The fundamental categories that can be used to separate radio galaxies is that of compact and extended sources. Compact sources, also known as point sources, may be unresolved by the telescope, or they may be a radio galaxy in their own right (FR0), or another type of radio galaxy in the early stages of evolution. Extended sources are resolved by the telescope and display a much larger morphological variety. In Chapter 2, we explored the use of deep neural networks for classifying between compact and various classes of extended sources in radio astronomical data from the Radio Galaxy Zoo, consisting mainly of images from the FIRST survey. Beginning with a sample of more than 200,000 galaxies, we used the Python Blob Detector and Source-Finder (PyBDSF) to separate them according to the number of components they possess, under the assumption that there is one source per image. The first analysis involved the two-class problem of distinguishing between compact sources

and multiple-component extended sources. A 3 convolutional and 2 dense layer network produced the highest test classification accuracy. Using the knowledge gained from the 2-class problem, we applied the same architecture on the 4-class problem of classifying between compact sources and 3 classes of extended (1-, 2- and >3 component) sources, achieving a slightly lower test classification accuracy. Finally, we used the 4-class trained network on DR1 of the Radio Galaxy Zoo to see how well it matched predictions from citizen scientists, based on the number of peaks and components of the radio sources, and were able to slightly improve on the accuracies obtained compared to the 4-class problem. All three analyses also involved generating augmented images which served to increase the classification metrics. We note that the overall high accuracy on the RGZ DR1 is mainly influenced by the fact that the sources in the compact/single-component have the simplest morphology and only one component and are therefore the easiest to identify. The classification metrics tend to decrease as the number of components increases; hence the classification metrics on the (>3 component) sources is the poorest, which is most probably due to these sources having the most complex morphology and the fewest original examples. Although the use of image augmentation improves results, it is limited by the number of original examples of images one has in a particular class, especially if the class displays increased morphological variety and complexity. Additionally, the images were classified solely on the radio galaxy data, without integrating the infrared data, therefore it is not possible to determine more conclusively whether or not the radio emission in a particular image belongs to the same source. Additionally, since the majority of the images originated from the FIRST survey, it is known that the survey failed to show extended sources accurately due to the relatively poor angular resolution and low sensitivity to extended low surface brightness structures.

The limitations of the FIRST survey prompted the development of radio surveys that would achieve greater sensitivity and resolution. One such survey is the LOFAR survey. Due to the compact core and longer baselines, it is able to better capture fainter and highly detailed extended emission from further away. This aspect reveals a richer range of morphologies compared to previous surveys, and it is important to develop methods to preserve these fine-grained details. Despite the fact convolutional neural networks are translationally invariant, they are not rotationally invariant. The local features within an image are preserved, however how the features relate to each other on a global scale increasingly disintegrates due to the use of the pooling layer, as it summarises the pixel information over a small localised area and therefore results in some information loss. This downfall of convolutional networks motivated the development of Capsule networks, which are networks composed of groups of neurons whose aim is to preserve the relative location of features on a local and global scale, and hence achieve both translational and rotational invariance. In the context of radio galaxies, the orientation and pattern of the emission is important as it affects the morphological classification. As such, Chapter 3 compares the performance of Capsule networks against that of

simpler 4- and 8- layer convolutional neural network setups using the classes of Unresolved, FRI and FRII radio galaxies from the LOFAR LoTSS Hetdex survey, using 2900 original sources, which were augmented to have over 15,000 in total. In contrast to the sources in Chapter 2, these sources have optical IDs and have also had galaxies with radio emission due to star-formation removed. We explored using the original FITS file data as well as the sigma-clipped numpy arrays. The CNNs always supersede the performance of Capsule networks, with or without image augmentation. It appears as though the pooling operation is advantageous for the radio galaxy images at hand as it appears make the CNNs more robust to the presence of noise in images, as well as allowing more freedom in how the morphology can vary within classes. The relatively poorer performance of the Capsule networks may also be explained by the fact that the original sample size was insufficient for them to capture the underlying emission patterns.

In Chapter 4, we consider the problem of source-finding in radio astronomical data, using simulated images from the SKA across 3 integration times, each at 3 different frequencies. The data was originally intended for the SKA data challenge, to develop algorithms to find and characterise sources, that are suited to the real type of data that the SKA will produce. Three classes of galaxies are simulated, namely flat-spectrum AGN, steep-spectrum AGN and SFGs. The source-finder we developed is based on a deep learning approach, consisting of a simple convolutional autoencoder (AutoSource) with 3 convolutional layers, trained on the simulated maps that have been segmented into 50x50 pixel blocks and using the knowledge of the source locations to generate a solution map, which provides the training 'data' and 'labels' respectively. The performance of AutoSource is compared to that of PyBDSF, a state-of-the-art source finder based on fitting Gaussian components to sources, across different signal-to-noise ratios. We find that one method performs better than the other depending on the type of source, dataset (varying in integration time and frequency), and signal-to-noise ratio. There is not one method that performs better in all circumstances. AutoSource tends to detect more true positives as well as true negatives, however it also detects more false positives compared to PyBDSF. The competitive results produced by AutoSource indicate that deep learning techniques such as convolutional autoencoders hold promise in locating astronomical sources, especially with the use of image augmentation which can be used to produce more instances of rarer radio galaxy classes and therefore improve the source-finding ability for those source types. Furthermore, the technique of generating training images is very flexible as it is possible to produce segmented blocks that are overlapping to varying degrees, as well as being able to change the sizes of the blocks.

As evident from the introductory section discussing the application of machine learning techniques to astronomical datasets, the type of machine learning approach to take depends on the type of data, the complexity, the number of classes or variables, and the context. Deep learning algorithms have shown the most promise so far in performing classification and re-

gression tasks in high-dimensional data such as image data. Despite this, there is not yet one single architecture to use in all circumstances; it is necessary to experiment with different architectures and hyper-parameters to see which ones perform better than the others. The use of transfer learning has also proven to be useful, from using the stored weights from one network and applying it to a similar dataset.

The development of deep learning algorithms in the current work has largely been through trial-and error; we begin with simple neural network architecture and gradually increase the complexity to see whether the metrics improve. The validation and training losses have been monitored in order to see if the network was underfitting (indicating the data is more complex than the model can describe), or overfitting (the model has too many parameters relative to the data complexity). Additionally we looked at the features being learned by the network on a layer-by-layer level by outputting the activations. A future development to our methods would involve looking at their interpretability, for example examining which nodes are most relevant in assigning a class to a particular image.

In conclusion, we have developed deep learning algorithms to classify radio galaxies based on the number of components, as well as the Fanaroff-Riley class, and have also shown that deep learning algorithms can be competitive source finders. This application is novel and deserves further development. For example, incorporating the ability to attribute properties such as flux, size and angle to radio sources through regression.

In future, the continual development and refinement of deep learning models will be necessary as more surveys are performed, detecting sources at higher sensitivities and resolutions. The more we know about existing sources enables us to tailor the algorithms to mine data for unusual sources.

The use of transfer learning from networks trained on radio astronomical data should provide a more efficient and accurate way to obtain results from future astronomical datasets, compared to always training from scratch. The development of more algorithms that can perform cross-matching of sources between surveys will be of increasing importance as more sources are detected. Existing region-based convolutional networks should also be improved so they can be trained to pay attention to relevant parts of an image and be able to distinguish whether radio emission belongs to one, or multiple sources in an image. An integrated approach that accurately combines source-finding, cross-correlation, characterisation and classification would be an ideal development.

Other potential future developments include the generation or simulation of realistic radio images based on an exhaustive study of existing ones, to see whether a machine learning algorithm is able to detect the difference between the real and simulated ones. If the algorithm is unable to make the distinction, the simulated images can be used to increase the number

of images in a training set, which is essentially another image augmentation technique.

6 Bibliography

Ackermann S., Schawinski K., Zhang C., Weigel A. K., Turp M. D., 2018, http://dx.doi.org/10.1093/mnras/sty1398 , https://ui.adsabs.harvard.edu/abs/2018MNRAS.479..415A 479, 415

Alger M., 2016, Master's thesis, Australian National University

Alger M. J., et al., 2018, http://dx.doi.org/10.1093/mnras/sty1308 , https://ui.adsabs.harvard.edu/abs/2018MNRAS.478.5547A 478, 5547

Alhassan W., Taylor A. R., Vaccari M., 2018, http://dx.doi.org/10.1093/mnras/sty2038 Monthly Notices of the Royal Astronomical Society, 480, 2085

Andreon S., Gargiulo G., Longo G., Tagliaferri R., Capuano N., 1999. pp 3810 – 3815 vol.6, http://dx.doi.org/10.1109/IJCNN.1999.830761 doi:10.1109/IJCNN.1999.830761

Aniyan A. K., Thorat K., 2017, http://dx.doi.org/10.3847/1538-4365/aa7333 The Astrophysical Journal Supplement Series, 230, 20

Ankita Dubey A. C., 2017, International Journal of Scientific Research Engineering Technology (IJSRET), 6, 624

Arpit D., et al., 2017, in Precup D., Teh Y. W., eds, Proceedings of Machine Learning Research Vol. 70, Proceedings of the 34th International Conference on Machine Learning. PMLR, International Convention Centre, Sydney, Australia, pp 233–242, `http://proceedings.mlr.press/v70/arpit17a.html`

Auriemma C., Perola G. C., Ekers R. D., Fanti R., Lari C., Jaffe W. J., Ulrich M. H., 1977, , https://ui.adsabs.harvard.edu/abs/1977A

Avendi M., 2018, Another look into overfitting

Bach S., Binder A., Montavon G., Klauschen F., Muller K., Samek W., 2015, http://dx.doi.org/10.1371/journal.pone.0130140 PLoS One, 10

Bailey S., 2012, Publications of the Astronomical Society of the Pacific, 124, 1015

Baldi R. D., Capetti A., Giovannini G., 2015, http://dx.doi.org/10.1051/0004-6361/201425426 , https://ui.adsabs.harvard.edu/abs/2015AA...576A..38B 576, A38

Baldi P., Bauer K., Eng C., Sadowski P., Whiteson D., 2016a, http://dx.doi.org/10.1103/PhysRevD.93.094034 Phys. Rev. D, 93, 094034

Baldi R. D., Capetti A., Giovannini G., 2016b, http://dx.doi.org/10.1002/asna.201512275 Astronomische Nachrichten, https://ui.adsabs.harvard.edu/abs/2016AN....337..114B 337, 114

Ball N. M., Brunner R. J., 2010, http://dx.doi.org/10.1142/S0218271810017160 International Journal of Modern Physics D, https://ui.adsabs.harvard.edu/abs/2010IJMPD..19.1049B 19, 1049

Ball N. M., Brunner R. J., Myers A. D., Tcheng D., 2006, http://dx.doi.org/10.1086/507440 Astrophys. J., 650, 497

Ballard D. H., 1987, in Proceedings of the Sixth National Conference on Artificial Intelligence - Volume 1. AAAI'87. AAAI Press, pp 279–284, `http://dl.acm.org/citation.cfm?id=1863696.1863746`

Banfield J. K., et al., 2015, http://dx.doi.org/10.1093/mnras/stv1688 Monthly Notices of the Royal Astronomical Society, 453, 2326

Baron D., 2019, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2019arXiv190407248B p. arXiv:1904.07248

Baron D., Poznanski D., 2016, http://dx.doi.org/10.1093/mnras/stw3021 Monthly Notices of the Royal Astronomical Society, 465, 4530

Basson J. F., Alexander P., 2003, http://dx.doi.org/10.1046/j.1365-8711.2003.06069.x , https://ui.adsabs.harvard.edu/abs/2003MNRAS.339..353B 339, 353

Baum E. B., Haussler D., 1989, in Touretzky D. S., ed., , Advances in Neural Information Processing Systems 1. Morgan-Kaufmann, pp 81–90, `http://papers.nips.cc/paper/154-what-size-net-gives-valid-generalization.pdf`

Baum S. A., Zirbel E. L., O'Dea C. P., 1995, http://dx.doi.org/10.1086/176202 , https://ui.adsabs.harvard.edu/abs/1995ApJ...451...88B 451, 88

Beck R., Krause M., 2005, http://dx.doi.org/10.1002/asna.200510366 Astronomische Nachrichten, https://ui.adsabs.harvard.edu/abs/2005AN....326..414B 326, 414

Beck R., Wielebinski R., 2013, Magnetic Fields in Galaxies. p. 641, http://dx.doi.org/10.1007/978-94-007-5612-0_13 doi:10.1007/978-94-007-5612-0_13

Becker R. H., White R. L., Helfand D. J., 1994, in Crabtree D. R., Hanisch R. J., Barnes J., eds, Astronomical Society of the Pacific Conference Series Vol. 61, Astronomical Data Analysis Software and Systems III. p. 165

Becker R. H., White R. L., Helfand D. J., 1995, http://dx.doi.org/10.1086/176166 , https://ui.adsabs.harvard.edu/abs/1995ApJ...450..559B 450, 559

Begelman M. C., Rees M. J., Blandford R. D., 1979, http://dx.doi.org/10.1038/279770a0 , https://ui.adsabs.harvard.edu/abs/1979Natur.279..770B 279, 770

Bell Burnell J., 1969, PhD thesis, University of Cambridge

Bellman R., 2003, Dynamic Programming. Dover Books on Computer Science Series, Dover Publications, `https://books.google.be/books?id=fyVtp3EMxasC`

Belson W. A., 1959, Journal of the Royal Statistical Society. Series C (Applied Statistics), 8, 65

Bennett A. S., Simth F. G., 1962, http://dx.doi.org/10.1093/mnras/125.1.75 Monthly Notices of the Royal Astronomical Society, 125, 75

Bergstra J., Bengio Y., 2012, J. Mach. Learn. Res., 13, 281

Best P. N., Kauffmann G., Heckman T. M., Brinchmann J., Charlot S., Ivezić Ž., White S. D. M., 2005, http://dx.doi.org/10.1111/j.1365-2966.2005.09192.x , https://ui.adsabs.harvard.edu/abs/2005MNRAS.362...25B 362, 25

Bicknell G. V., 1994, http://dx.doi.org/10.1086/173748 , https://ui.adsabs.harvard.edu/abs/1994ApJ...422..542B 422, 542

Bicknell G. V., 1995, http://dx.doi.org/10.1086/192232 , https://ui.adsabs.harvard.edu/abs/1995ApJS..101...29B 101, 29

Bieri R., 2016, Theses, Université Pierre et Marie Curie - Paris VI, `https://tel.archives-ouvertes.fr/tel-01496864`

Bilicki, M. et al., 2018, http://dx.doi.org/10.1051/0004-6361/201731942 A&A, 616, A69

Bishop C. M., 1995, Neural Networks for Pattern Recognition. Oxford University Press, Inc., New York, NY, USA

Blake C., Wall J., 2002, http://dx.doi.org/10.1038/416150a , https://ui.adsabs.harvard.edu/abs/2002Natur.416..150B 416, 150

Blandford R. D., Rees M. J., 1974, http://dx.doi.org/10.1093/mnras/169.3.395 , https://ui.adsabs.harvard.edu/abs/1974MNRAS.169..395B 169, 395

Blanton E. L., Clarke T. E., Sarazin C. L., Randall S. W., McNamara B. R., 2010, http://dx.doi.org/10.1073/pnas.0913904107 Proceedings of the National Academy of Science, https://ui.adsabs.harvard.edu/abs/2010PNAS..107.7174B 107, 7174

Bonaldi A., Braun R., 2018, Square Kilometre Array Science Data Challenge 1 (http://arxiv.org/abs/1811.10454 arXiv:1811.10454)

Bonaldi A., Bonato M., Galluzzi V., Harrison I., Massardi M., Kay S., De Zotti G., Brown M. L., 2018, http://dx.doi.org/10.1093/mnras/sty2603 Monthly Notices of the Royal Astronomical Society, 482, 2

Bonato M., et al., 2017, http://dx.doi.org/10.1093/mnras/stx974 Monthly Notices of the Royal Astronomical Society, https://ui.adsabs.harvard.edu/abs/2017MNRAS.469.1912B 469, 1912

Bongiorno A., et al., 2016, http://dx.doi.org/10.1051/0004-6361/201527436 Astronomy and Astrophysics, https://ui.adsabs.harvard.edu/abs/2016AA...588A..78B 588, A78

Boser B. E., Guyon I. M., Vapnik V. N., 1992, in Proceedings of the Fifth Annual Workshop on Computational Learning Theory. COLT '92. ACM, New York, NY, USA, pp 144–152, http://dx.doi.org/10.1145/130385.130401 doi:10.1145/130385.130401, http://doi.acm.org/10.1145/130385.130401

Bosma A., 1978, PhD thesis, Groningen Univ., (1978)

Bottou L., 1998, Online Learning and Stochastic Approximations

Boyle B. J., Terlevich R. J., 1998, http://dx.doi.org/10.1046/j.1365-8711.1998.01264.x Monthly Notices of the Royal Astronomical Society, https://ui.adsabs.harvard.edu/abs/1998MNRAS.293L..49B 293, L49

Breiman L., 2001, http://dx.doi.org/10.1023/A:1010933404324 Machine Learning, 45, 5

Breiman L., Last M., Rice J., 2003, Random Forests: finding quasars. pp 243–254

Brett D. R., West R. G., Wheatley P. J., 2004, http://dx.doi.org/10.1111/j.1365-2966.2004.08093.x Monthly Notices of the Royal Astronomical Society, 353, 369

Bridle A. H., Perley R. A., 1984, http://dx.doi.org/10.1146/annurev.aa.22.090184.001535 , https://ui.adsabs.harvard.edu/abs/1984ARA

Brienza M., 2018, PhD thesis, University of Groningen

Brienza M., et al., 2016, http://dx.doi.org/10.1051/0004-6361/201526754 , https://ui.adsabs.harvard.edu/abs/2016AA...585A..29B 585, A29

Brienza M., et al., 2018, http://dx.doi.org/10.1051/0004-6361/201832846 , https://ui.adsabs.harvard.edu/abs/2018AA...618A..45B 618, A45

Britz D., 2019, Understanding Convolutional Neural Networks for NLP

Brown S., Rudnick L., 2011, http://dx.doi.org/10.1111/j.1365-2966.2010.17738.x Monthly Notices of the Royal Astronomical Society, 412, 2

Brüggen M., Bykov A., Ryu D., Röttgering H., 2012, http://dx.doi.org/10.1007/s11214-011-9785-9 , https://ui.adsabs.harvard.edu/abs/2012SSRv..166..187B 166, 187

Brüggen M., et al., 2018, http://dx.doi.org/10.1093/mnras/sty851 , https://ui.adsabs.harvard.edu/abs/2018MNRAS.477.3461B 477, 3461

Brunetti G., Jones T. W., 2014, http://dx.doi.org/10.1142/S0218271814300079 International Journal of Modern Physics D, https://ui.adsabs.harvard.edu/abs/2014IJMPD..2330007B 23, 1430007

Buehlmann P., van de Geer S., 2011, Statistics for High-Dimensional Data: Methods, Theory and Applications, 1st edn. Springer Publishing Company, Incorporated

Burbidge G. R., 1956, http://dx.doi.org/10.1086/146237 , https://ui.adsabs.harvard.edu/abs/1956ApJ...124..416B 124, 416

Burn B. J., 1966, http://dx.doi.org/10.1093/mnras/133.1.67 , https://ui.adsabs.harvard.edu/abs/1966MNRAS.133...67B 133, 67

Burns J. O., Rhee G., Owen F. N., Pinkney J., 1994, http://dx.doi.org/10.1086/173792 , https://ui.adsabs.harvard.edu/abs/1994ApJ...423...94B 423, 94

Calzetti D., 2001, http://dx.doi.org/10.1016/S1387-6473(01)00144-0 New Astronomy Reviews, https://ui.adsabs.harvard.edu/abs/2001NewAR..45..601C 45, 601

Carliles S., Budavári T., Heinis S., Priebe C., Szalay A., 2008, in Argyle R. W., Bunclark P. S., Lewis J. R., eds, Astronomical Society of the Pacific Conference Series Vol. 394, Astronomical Data Analysis Software and Systems XVII. p. 521 (http://arxiv.org/abs/0711.2477 arXiv:0711.2477)

Carliles S., Budavári T., Heinis S., Priebe C., Szalay A. S., 2010, http://dx.doi.org/10.1088/0004-637X/712/1/511 , https://ui.adsabs.harvard.edu/abs/2010ApJ...712..511C 712, 511

Carrasco-Davis R., et al., 2019, http://dx.doi.org/10.1088/1538-3873/aaef12 , https://ui.adsabs.harvard.edu/abs/2019PASP..131j8006C 131, 108006

Carroll B. W., Ostlie D. A., 2006, An introduction to modern astrophysics and cosmology

Caruana R., Lawrence S., Giles L., 2000, in Proceedings of the 13th International Conference on Neural Information Processing Systems. NIPS'00. MIT Press, Cambridge, MA, USA, pp 381–387, http://dl.acm.org/citation.cfm?id=3008751.3008807

Cattaneo A., et al., 2009, http://dx.doi.org/10.1038/nature08135 , https://ui.adsabs.harvard.edu/abs/2009Natur.460..213C 460, 213

Cauchy A.-L., 1847, Œuvres complètes, 1, 399

Clarke T., et al., 2014, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2014arXiv1401.0329C p. arXiv:1401.0329

Cohen M. H., Unwin S. C., 1984, in Fanti R., Kellermann K. I., Setti G., eds, IAU Symposium Vol. 110, VLBI and Compact Radio Sources. p. 95

Colafrancesco, S. Marchegiani, P. Emritte, M. S. 2016, http://dx.doi.org/10.1051/0004-6361/201424904 A&A, 595, A21

Condon J. J., Cotton W. D., Greisen E. W., Yin Q. F., Perley R. A., Taylor G. B., Broderick J. J., 1998, http://dx.doi.org/10.1086/300337 , https://ui.adsabs.harvard.edu/abs/1998AJ....115.1693C 115, 1693

Connolly A. J., Szalay A. S., Bershady M. A., Kinney A. L., Calzetti D., 1995, http://dx.doi.org/10.1086/117587 , https://ui.adsabs.harvard.edu/abs/1995AJ....110.1071C 110, 1071

Conselice C. J., 2012, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2012arXiv1212.5641C p. arXiv:1212.5641

Contopoulos I., Gabuzda D., Kylafis N., eds, 2015, The Formation and Disruption of Black Hole Jets Astrophysics and Space Science Library Vol. 414, http://dx.doi.org/10.1007/978-3-319-10356-3. doi:10.1007/978-3-319-10356-3.

Dabhade P., et al., 2019, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2019arXiv190400409D p. arXiv:1904.00409

Daniel 2019, Intro to optimization in deep learning: Gradient Descent

Davis J., Goadrich M., 2006, in Proceedings of the 23rd International Conference on Machine Learning. ICML '06. ACM, New York, NY, USA, pp 233–240, http://dx.doi.org/10.1145/1143844.1143874 doi:10.1145/1143844.1143874, `http://doi.acm.org/10.1145/1143844.1143874`

Davis Jr. L., Greenstein J. L., 1951, http://dx.doi.org/10.1086/145464 , https://ui.adsabs.harvard.edu/abs/1951ApJ...114..206D 114, 206

Delchambre L., 2016, http://dx.doi.org/10.1093/mnras/stw1025 Monthly Notices of the Royal Astronomical Society, 460, 2811

Deng L., Yu D., 2014, http://dx.doi.org/10.1561/2000000039 Foundations and Trends in Signal Processing, 7, 197

Deng J., Dong W., Socher R., jia Li L., Li K., Fei-fei L., 2009, in In CVPR.

Dickey J. M., Strasser S., Gaensler B. M., Haverkorn M., Kavars D., McClure-Griffiths N. M., Stil J., Taylor A. R., 2009, http://dx.doi.org/10.1088/0004-637X/693/2/1250 , https://ui.adsabs.harvard.edu/abs/2009ApJ...693.1250D 693, 1250

Dieleman S., et al., 2015a, Lasagne: First release., http://dx.doi.org/10.5281/zenodo.27878 doi:10.5281/zenodo.27878, `http://dx.doi.org/10.5281/zenodo.27878`

Dieleman S., Willett K. W., Dambre J., 2015b, http://dx.doi.org/10.1093/mnras/stv632 Monthly Notices of the Royal Astronomical Society, 450, 1441

Domínguez Sánchez H., Huertas-Company M., Bernardi M., Tuccillo D., Fischer J. L., 2018, http://dx.doi.org/10.1093/mnras/sty338 , https://ui.adsabs.harvard.edu/abs/2018MNRAS.476.3661D 476, 3661

Domínguez Sánchez H., et al., 2019, http://dx.doi.org/10.1093/mnras/sty3497 , https://ui.adsabs.harvard.edu/abs/2019MNRAS.484...93D 484, 93

Du Buisson L., 2015, Master's thesis, University of Cape Town

Duchi J., Hazan E., Singer Y., 2011, J. Mach. Learn. Res., 12, 2121

Eckert D., et al., 2015, http://dx.doi.org/10.1038/nature16058 , https://ui.adsabs.harvard.edu/abs/2015Natur.528..105E 528, 105

Ekers R. D., 2010, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2010arXiv1004.4279E p. arXiv:1004.4279

Ellingson S. W., 2011, http://dx.doi.org/10.1109/TAP.2011.2122230 IEEE Transactions on Antennas and Propagation, 59, 1855

Evans D. A., Worrall D. M., Hardcastle M. J., Kraft R. P., Birkinshaw M., 2006, http://dx.doi.org/10.1086/500658 , https://ui.adsabs.harvard.edu/abs/2006ApJ...642...96E 642, 96

Ewen H. I., Purcell E. M., 1951, http://dx.doi.org/10.1038/168356a0 , https://ui.adsabs.harvard.edu/abs/1951Natur.168..356E 168, 356

Fabian A. C., 1999, http://dx.doi.org/10.1073/pnas.96.9.4749 Proceedings of the National Academy of Sciences, 96, 4749

Fabian A. C., 2012, http://dx.doi.org/10.1146/annurev-astro-081811-125521 , https://ui.adsabs.harvard.edu/abs/2012ARA

Faisst A. L., Prakash A., Capak P. L., Lee B., 2019, http://dx.doi.org/10.3847/2041-8213/ab3581 , https://ui.adsabs.harvard.edu/abs/2019ApJ...881L...9F 881, L9

Fanaroff B. L., Riley J. M., 1974, http://dx.doi.org/10.1093/mnras/167.1.31P Monthly Notices of the Royal Astronomical Society, 167, 31P

Fang Y., Cui Y., Ao X., 2019, http://dx.doi.org/10.1155/2019/9196234 Advances in Astronomy, https://ui.adsabs.harvard.edu/abs/2019AdAst2019E..27F 2019, 9196234

Farabet C., Couprie C., Najman L., LeCun Y., 2013, http://dx.doi.org/10.1109/TPAMI.2012.231 IEEE Trans. Pattern Anal. Mach. Intell., 35, 1915

Fawcett T., 2006, http://dx.doi.org/https://doi.org/10.1016/j.patrec.2005.10.010 Pattern Recognition Letters, 27, 861

Feenberg E., Primakoff H., 1948, http://dx.doi.org/10.1103/PhysRev.73.449 Phys. Rev., 73, 449

Feigelson E. D., Laurent-Muehleisen S. A., Kollgaard R. I., Fomalont E. B., 1995, http://dx.doi.org/10.1086/309642 , https://ui.adsabs.harvard.edu/abs/1995ApJ...449L.149F 449, L149

Feretti L., 2003.

Ferrarese L., Celotti A., 2002, What Causes the FRI - FRII Dichotomy?, NOAO Proposal

Field G., Chaisson E., 1985, THE INVISIBLE UNIVERSE : Probing the frontiers of astrophysics. Birkhauser-Boston

Firth A. E., Lahav O., Somerville R. S., 2003, http://dx.doi.org/10.1046/j.1365-8711.2003.06271.x , https://ui.adsabs.harvard.edu/abs/2003MNRAS.339.1195F 339, 1195

Gaensler B. M., Beck R., Feretti L., 2004, http://dx.doi.org/10.1016/j.newar.2004.09.003 , https://ui.adsabs.harvard.edu/abs/2004NewAR..48.1003G 48, 1003

Galvin T. J., et al., 2019, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2019arXiv190402876G p. arXiv:1904.02876

Gao D., Zhang Y.-X., Zhao Y.-H., 2009, http://dx.doi.org/10.1088/1674-4527/9/2/011 Research in Astronomy and Astrophysics, 9, 220

Geach J. E., 2012, http://dx.doi.org/10.1111/j.1365-2966.2011.19913.x Monthly Notices of the Royal Astronomical Society, 419, 2633

George D., Huerta E., 2018, http://dx.doi.org/https://doi.org/10.1016/j.physletb.2017.12.053 Physics Letters B, 778, 64

Gheller C., Vazza F., Bonafede A., 2018, http://dx.doi.org/10.1093/mnras/sty2102 , https://ui.adsabs.harvard.edu/abs/2018MNRAS.480.3749G 480, 3749

Giovannini G., Feretti L., Gregorini L., Parma P., 1988, , https://ui.adsabs.harvard.edu/abs/1988A

Girshick R., Donahue J., Darrell T., Malik J., 2013, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2013arXiv1311.2524G p. arXiv:1311.2524

Gitti M., Brighenti F., McNamara B. R., 2012, http://dx.doi.org/10.1155/2012/950641 Advances in Astronomy, https://ui.adsabs.harvard.edu/abs/2012AdAst2012E...6G 2012, 950641

Glorot X., Bengio Y., 2010, in In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS'10). Society for Artificial Intelligence and Statistics.

Glorot X., Bordes A., Bengio Y., 2011, in Gordon G., Dunson D., Dudík M., eds, Proceedings of Machine Learning Research Vol. 15, Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics. PMLR, Fort Lauderdale, FL, USA, pp 315–323, `http://proceedings.mlr.press/v15/glorot11a.html`

Goodfellow I. J., Shlens J., Szegedy C., 2014, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2014arXiv1412.6572G p. arXiv:1412.6572

Goodfellow I., Bengio Y., Courville A., 2016, Deep Learning. MIT Press

Gopal-Krishna Wiita P. J., 2000, , https://ui.adsabs.harvard.edu/abs/2000A

Graff P., Feroz F., Hobson M. P., Lasenby A., 2014, http://dx.doi.org/10.1093/mnras/stu642 Monthly Notices of the Royal Astronomical Society, 441, 1741

Hale C. L., Robotham A. S. G., Davies L. J. M., Jarvis M. J., Driver S., Heywood I., 2019, http://dx.doi.org/10.1093/mnras/stz1462 Monthly Notices of the Royal Astronomical Society, 487, 3971

Hardcastle M. J., 2018, http://dx.doi.org/10.1093/mnras/stx3358 Monthly Notices of the Royal Astronomical Society, 475, 2768

Hardcastle M. J., et al., 2019, http://dx.doi.org/10.1051/0004-6361/201833893 , https://ui.adsabs.harvard.edu/abs/2019AA...622A..12H 622, A12

Hargrave P. J., Ryle M., 1976, http://dx.doi.org/10.1093/mnras/175.3.481 , https://ui.adsabs.harvard.edu/abs/1976MNRAS.175..481H 175, 481

Hartley P., Flamary R., Jackson N., Tagore A. S., Metcalf R. B., 2017, http://dx.doi.org/10.1093/mnras/stx1733 , https://ui.adsabs.harvard.edu/abs/2017MNRAS.471.3378H 471, 3378

Hastie T., Tibshirani R., Friedman J., 2009, The elements of statistical learning: data min-

ing, inference and prediction, 2 edn. Springer, http://www-stat.stanford.edu/~tibs/ElemStatLearn/

He K., Zhang X., Ren S., Sun J., 2015a, in The IEEE International Conference on Computer Vision (ICCV).

He K., Zhang X., Ren S., Sun J., 2015b, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2015arXiv151203385H p. arXiv:1512.03385

Heckman T. M., Best P. N., 2014, http://dx.doi.org/10.1146/annurev-astro-081913-035722 Annual Review of Astronomy and Astrophysics, 52, 589

Helou G., Soifer B. T., Rowan-Robinson M., 1985, http://dx.doi.org/10.1086/184556 , https://ui.adsabs.harvard.edu/abs/1985ApJ...298L...7H 298, L7

Hezaveh Y. D., Levasseur L. P., Marshall P. J., 2017, Nature, 548, 555 EP

Hill G. J., Lilly S. J., 1991, http://dx.doi.org/10.1086/169597 , https://ui.adsabs.harvard.edu/abs/1991ApJ...367....1H 367, 1

Hine R. G., Longair M. S., 1979, http://dx.doi.org/10.1093/mnras/188.1.111 Monthly Notices of the Royal Astronomical Society, 188, 111

Hoang T., Lazarian A., 2008, http://dx.doi.org/10.1111/j.1365-2966.2008.13249.x , https://ui.adsabs.harvard.edu/abs/2008MNRAS.388..117H 388, 117

Hochreiter S., 1991, Untersuchungen zu dynamischen neuronalen Netzen. Diploma thesis, Institut für Informatik, Lehrstuhl Prof. Brauer, Technische Universität München

Hochreiter S., 1998, http://dx.doi.org/10.1142/S0218488598000094 Int. J. Uncertain. Fuzziness Knowl.-Based Syst., 6, 107

Hocking A., Geach J. E., Sun Y., Davey N., 2018, http://dx.doi.org/10.1093/mnras/stx2351 , https://ui.adsabs.harvard.edu/abs/2018MNRAS.473.1108H 473, 1108

Hong K., 2019, Artifical Neural Network (ANN)

Hopkins A. M., Miller C. J., Connolly A. J., Genovese C., Nichol R. C., Wasserman L., 2002, http://dx.doi.org/10.1086/338316 , https://ui.adsabs.harvard.edu/abs/2002AJ....123.1086H 123, 1086

Hopkins A. M., et al., 2015, http://dx.doi.org/10.1017/pasa.2015.37 , https://ui.adsabs.harvard.edu/abs/2015PASA...32...37H 32, e037

Hossin M., M.N S., 2015, http://dx.doi.org/10.5121/ijdkp.2015.5201 International Journal of Data Mining & Knowledge Management Process, 5, 01

Huertas-Company, M. Rouan, D. Tasca, L. Soucail, G. Le Fèvre, O. 2008, http://dx.doi.org/10.1051/0004-6361:20078625 A&A, 478, 971

Hui J., Aragon M., Cui X., Flegal J. M., 2018, http://dx.doi.org/10.1093/mnras/stx3235 , https://ui.adsabs.harvard.edu/abs/2018MNRAS.475.4494H 475, 4494

Ishwara-Chandra C. H., Saikia D. J., 1999, http://dx.doi.org/10.1046/j.1365-8711.1999.02835.x Monthly Notices of the Royal Astronomical Society, 309, 100

Isola P., Zhu J.-Y., Zhou T., Efros A. A., 2016, 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 5967–5976

Iwasaki H., Ichinohe Y., Uchiyama Y., 2019, http://dx.doi.org/10.1093/mnras/stz1990 , https://ui.adsabs.harvard.edu/abs/2019MNRAS.488.4106I 488, 4106

Jackson, N. et al., 2016, http://dx.doi.org/10.1051/0004-6361/201629016 A&A, 595, A86

Jacobs C., Glazebrook K., Collett T., More A., McCarthy C., 2017, http://dx.doi.org/10.1093/mnras/stx1492 MNRAS, 471, 167

Jaroszynski M., Abramowicz M. A., Paczynski B., 1980, , https://ui.adsabs.harvard.edu/abs/1980AcA....30....1J 30, 1

Jarvis M., et al., 2016, in Proceedings of MeerKAT Science: On the Pathway to the SKA. 25-27 May. p. 6 (http://arxiv.org/abs/1709.01901 `arXiv:1709.01901`)

Johnson J. B., 1928, http://dx.doi.org/10.1103/PhysRev.32.97 Phys. Rev., 32, 97

Jolliffe I., Cadima J., 2016, http://dx.doi.org/10.1098/rsta.2015.0202 Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 374, 20150202

Kaastra J. S., et al., 2008, http://dx.doi.org/10.1007/s11214-008-9326-3 Space Science Reviews, 134, 1

Kang H., 2018, http://dx.doi.org/https://doi.org/10.1016/j.nuclphysbps.2018.07.036 Nuclear and Particle Physics Proceedings, 297-299, 259

Kapińska A. D., et al., 2017, http://dx.doi.org/10.3847/1538-3881/aa90b7 The Astronomical Journal, https://ui.adsabs.harvard.edu/abs/2017AJ....154..253K 154, 253

Karpathy 2016, CS231n: Convolutional Neural Networks for Visual Recognition

Katebi R., Zhou Y., Chornock R., Bunescu R., 2019, http://dx.doi.org/10.1093/mnras/stz915 Monthly Notices of the Royal Astronomical Society, 486, 1539

Kawakatu N., Kino M., Nagai H., 2009, http://dx.doi.org/10.1088/0004-637x/697/2/l173 The Astrophysical Journal, 697, L173

Keel W., 2007, Active galactic nuclei in the early universe. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 159–175, http://dx.doi.org/10.1007/978-3-540-72535-0_8 doi:10.1007/978-3-540-72535-0_8, https://doi.org/10.1007/978-3-540-72535-0_8

Kellermann K. I., Owen F. N., 1988, Radio galaxies and quasars. pp 563–602

Kellermann K. I., Pauliny-Toth I. I. K., 1966, http://dx.doi.org/10.1086/148844 The Astrophysical Journal, https://ui.adsabs.harvard.edu/abs/1966ApJ...145..954K 145, 954

Kellermann K. I., Verschuur G. L., 1988, Galactic and extragalactic radio astronomy (2nd edition)

Kembhavi A. K., Narlikar J. V., 1999, Quasars and active galactic nuclei : an introduction

Kennicutt R. C., Evans N. J., 2012, http://dx.doi.org/10.1146/annurev-astro-081811-125610 Annual Review of Astronomy and Astrophysics, 50, 531

Kereš D., Katz N., Weinberg D. H., Davé R., 2005, http://dx.doi.org/10.1111/j.1365-2966.2005.09451.x , https://ui.adsabs.harvard.edu/abs/2005MNRAS.363....2K 363, 2

Kerrigan J., et al., 2019, http://dx.doi.org/10.1093/mnras/stz1865 , https://ui.adsabs.harvard.edu/abs/2019MNRAS.488.2605K 488, 2605

Khan A., Huerta E., Wang S., Gruendl R., Jennings E., Zheng H., 2019, http://dx.doi.org/https://doi.org/10.1016/j.physletb.2019.06.009 Physics Letters B, 795, 248

Khandelwal R., 2018, Deep Learning Autoencoders

Kimball A. E., Ivezić Ž., 2008, http://dx.doi.org/10.1088/0004-6256/136/2/684 The Astronomical Journal, 136, 684

Kingma D. P., Ba J., 2014, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2014arXiv1412.6980K p. arXiv:1412.6980

Kohonen T., 1989, Self-organization and Associative Memory: 3rd Edition. Springer-Verlag, Berlin, Heidelberg

Kormendy J., Ho L. C., 2013, http://dx.doi.org/10.1146/annurev-astro-082708-101811 , https://ui.adsabs.harvard.edu/abs/2013ARA

Kotsiantis S. B., 2013, http://dx.doi.org/10.1007/s10462-011-9272-4 Artificial Intelligence Review, 39, 261

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., pp 1097–1105, http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

Laing R. A., Bridle A. H., 2002, http://dx.doi.org/10.1046/j.1365-8711.2002.05873.x Monthly Notices of the Royal Astronomical Society, 336, 1161

Lauberts A., Valentijn E. A., 1989, The surface photometry catalogue of the ESO-Uppsala galaxies

LeCun Y., Boser B., Denker J. S., Henderson D., Howard R. E., Hubbard W., Jackel L. D., 1989, http://dx.doi.org/10.1162/neco.1989.1.4.541 Neural Comput., 1, 541

LeCun Y., Bottou L., Orr G., Müller K., 2012, Efficient backprop. pp 9–48, http://dx.doi.org/10.1007/978-3-642-35289-8-3 doi:10.1007/978-3-642-35289-8-3

LeCun Y., Bengio Y., Hinton G., 2015, http://dx.doi.org/10.1038/nature14539 Nature, 521, 436

Lecun Y., Bottou L., Bengio Y., Haffner P., 1998, http://dx.doi.org/10.1109/5.726791 Proceedings of the IEEE, 86, 2278

Ledlow M. J., Owen F. N., 1996, http://dx.doi.org/10.1086/117985 , https://ui.adsabs.harvard.edu/abs/1996AJ....112....9L 112, 9

Lee C.-Y., Gallagher P. W., Tu Z., 2016, in Gretton A., Robert C. C., eds, Proceedings of Machine Learning Research Vol. 51, Proceedings of the 19th International Conference on Artificial Intelligence and Statistics. PMLR, Cadiz, Spain, pp 464–472, http://proceedings.mlr.press/v51/lee16a.html

Lelli F., et al., 2015, http://dx.doi.org/10.1051/0004-6361/201526613 , https://ui.adsabs.harvard.edu/abs/2015A

Liang M., Hu X., 2015, in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Libbrecht M. W. . N. W. S., 2015, Nat Rev Genet, 16, 321

Liou C.-Y., Huang J.-C., Yang W.-C., 2008, http://dx.doi.org/https://doi.org/10.1016/j.neucom.2008.04.030 Neurocomputing, 71, 3150

Liu R. H., et al., 2019, http://dx.doi.org/10.1093/mnras/stz2228 , https://ui.adsabs.harvard.edu/abs/2019MNRAS.489.1770L 489, 1770

Lloyd S. P., 1982, IEEE Transactions on Information Theory, 28, 129

Locatelli N., Vazza F., Domínguez-Fernández P., 2018,

http://dx.doi.org/10.3390/galaxies6040128 Galaxies, https://ui.adsabs.harvard.edu/abs/2018Galax...6..12 6, 128

Longair M. S., Riley J. M., 1979, http://dx.doi.org/10.1093/mnras/188.3.625 Monthly Notices of the Royal Astronomical Society, 188, 625

Lucas J., Calef B., Kyono T., 2018, in The Advanced Maui Optical and Space Surveillance Technologies Conference. p. 51

Lukic V., Brüggen M., Banfield J. K., Wong O. I., Rudnick L., Norris R. P., Simmons B., 2018, http://dx.doi.org/10.1093/mnras/sty163 Monthly Notices of the Royal Astronomical Society, 476, 246

Lukic V., Brüggen M., Mingo B., Croston J. H., Kasieczka G., Best P. N., 2019, http://dx.doi.org/10.1093/mnras/stz1289 , https://ui.adsabs.harvard.edu/abs/2019MNRAS.487.1729L 487, 1729

Mack K.-H., Klein U., O'Dea C. P., Willis A. G., Saripalli L., 1998, , https://ui.adsabs.harvard.edu/abs/1998A329, 431

Madau P., Dickinson M., 2014, http://dx.doi.org/10.1146/annurev-astro-081811-125615 Annual Review of Astronomy and Astrophysics, 52, 415

Madau P., Pozzetti L., Dickinson M., 1998, http://dx.doi.org/10.1086/305523 , https://ui.adsabs.harvard.edu/abs/1998ApJ...498..106M 498, 106

Mahatma V. H., et al., 2018, http://dx.doi.org/10.1093/mnras/sty025 , https://ui.adsabs.harvard.edu/abs/2018MNRAS.475.4557M 475, 4557

Mancuso C., Lapi A., Cai Z. Y., Negrello M., De Zotti G., Perrotta F., Danese L., 2015, in Advancing Astrophysics with the Square Kilometre Array (AASKA14). p. 82 (http://arxiv.org/abs/1412.5827 arXiv:1412.5827)

Mannor S., Peleg D., Rubinstein R., 2005, in Proceedings of the 22Nd International Conference on Machine Learning. ICML '05. ACM, New York, NY, USA, pp 561–568, http://dx.doi.org/10.1145/1102351.1102422 doi:10.1145/1102351.1102422, http://doi.acm.org/10.1145/1102351.1102422

Mao X., Shen C., Yang Y.-B., 2016, in Lee D. D., Sugiyama M., Luxburg U. V., Guyon I., Garnett R., eds, , Advances in Neural Information Processing Systems 29. Curran Associates, Inc., pp 2802–2810

Maragkoudakis A., Zezas A., Ashby M. L. N., Willner S. P., 2016, http://dx.doi.org/10.1093/mnras/stw3180 Monthly Notices of the Royal Astronomical Society, 466, 1192

Marconi A., Risaliti G., Gilli R., Hunt L. K., Maiolino R., Salvati M., 2004, http://dx.doi.org/10.1111/j.1365-2966.2004.07765.x , https://ui.adsabs.harvard.edu/abs/2004MNRAS.351..169M 351, 169

Masias M., Freixenet J., Llada X., Peracaula M., 2012, http://dx.doi.org/10.1111/j.1365-2966.2012.20742.x Monthly Notices of the Royal Astronomical Society, 422, 1674

Massardi M., Bonaldi A., Negrello M., Ricciardi S., Raccanelli A., de Zotti G., 2010, http://dx.doi.org/10.1111/j.1365-2966.2010.16305.x , https://ui.adsabs.harvard.edu/abs/2010MNRAS.404..532M 404, 532

McCulloch W. S., Pitts W., 1943, http://dx.doi.org/10.1007/BF02478259 The bulletin of mathematical biophysics, 5, 115

McMullin J. P., Waters B., Schiebel D., Young W., Golap K., 2007, in Shaw R. A., Hill F., Bell D. J., eds, Astronomical Society of the Pacific Conference Series Vol. 376, Astronomical Data Analysis Software and Systems XVI. p. 127

Mehdipour M., Costantini E., 2019, http://dx.doi.org/10.1051/0004-6361/201935205 , https://ui.adsabs.harvard.edu/abs/2019AA...625A..25M 625, A25

Meliani Z., Keppens R., 2009, http://dx.doi.org/10.1088/0004-637X/705/2/1594 The Astrophysical Journal, https://ui.adsabs.harvard.edu/abs/2009ApJ...705.1594M 705, 1594

Miljković O., 2009.

Miljkovic D., 2017. , http://dx.doi.org/10.23919/MIPRO.2017.7973581 doi:10.23919/MIPRO.2017.7973581

Miraghaei H., Best P. N., 2017, http://dx.doi.org/10.1093/mnras/stx007 Monthly Notices of the Royal Astronomical Society, 466, 4346

Mitton S., Ryle M., 1969, http://dx.doi.org/10.1093/mnras/146.3.221 , https://ui.adsabs.harvard.edu/abs/1969MNRAS.146..221M 146, 221

Mohan N., Rafferty D., 2015a, PyBDSF: Python Blob Detection and Source Finder, Astrophysics Source Code Library (ascl:1502.007)

Mohan N., Rafferty D., 2015b, PyBDSF: Python Blob Detection and Source Finder, Astrophysics Source Code Library (ascl:1502.007)

Montavon G., Orr G., Müller K., 2012, Neural Networks: Tricks of the Trade. No. LNCS 7700 in Lecture Notes in Computer Science Series, Springer Verlag

Montavon G., Samek W., Müller K.-R., 2017, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2017arXiv170607979M p. arXiv:1706.07979

Mortlock D. J., et al., 2011, http://dx.doi.org/10.1038/nature10159 , https://ui.adsabs.harvard.edu/abs/2011Natur.474..616M 474, 616

Murtagh F., Contreras P., 2012, http://dx.doi.org/10.1002/widm.53 Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2, 86

Nair V., Hinton G. E., 2010, in Proceedings of the 27th International Conference on International Conference on Machine Learning. ICML'10. Omnipress, USA, pp 807–814, `http://dl.acm.org/citation.cfm?id=3104322.3104425`

Nalepa J., Kawulok M., 2019, http://dx.doi.org/10.1007/s10462-017-9611-1 Artificial Intelligence Review, 52, 857

Ng A. Y., 2004, in Proceedings of the Twenty-first International Conference on Machine Learning. ICML '04. ACM, New York, NY, USA, pp 78– , http://dx.doi.org/10.1145/1015330.1015435 doi:10.1145/1015330.1015435, `http://doi.acm.org/10.1145/1015330.1015435`

Ngai E., Hu Y., Wong Y., Chen Y., Sun X., 2011, http://dx.doi.org/https://doi.org/10.1016/j.dss.2010.08.006 Decision Support Systems, 50, 559

Nicastro F., 2016, in XMM-Newton: The Next Decade. p. 27 (http://arxiv.org/abs/1611.03722 `arXiv:1611.03722`)

Nicastro F., et al., 2018, http://dx.doi.org/10.1038/s41586-018-0204-1 , https://ui.adsabs.harvard.edu/abs/2018Natur.558..406N 558, 406

Nielsen M., 2015, Neural Networks and Deep Learning. Determination Press, `https://books.google.be/books?id=STDBswEACAAJ`

Norris R. P., 2016, http://dx.doi.org/10.1017/S1743921316012825 Proceedings of the International Astronomical Union, 12, 103–113

Norris R. P., 2017a, http://dx.doi.org/10.1038/s41550-017-0233-y Nature Astronomy, 1, 671

Norris R. P., 2017b, http://dx.doi.org/10.1017/pasa.2016.63 Publications of the Astronomical Society of Australia, https://ui.adsabs.harvard.edu/abs/2017PASA...34....7N 34, e007

Norris R. P., the Emu team 2009.

Norris R. P., et al., 2006, http://dx.doi.org/10.1086/508275 The Astronomical Journal, 132, 2409

Norris R. P., et al., 2011, http://dx.doi.org/10.1071/AS11021 , https://ui.adsabs.harvard.edu/abs/2011PASA...28..215N 28, 215

Ntampaka M., et al., 2019, http://dx.doi.org/10.3847/1538-4357/ab14eb , https://ui.adsabs.harvard.edu/abs/2019ApJ...876...82N 876, 82

Padovani P., 2016, http://dx.doi.org/10.1007/s00159-016-0098-6 The Astronomy and Astrophysics Review, 24, 13

Padovani P., 2017, http://dx.doi.org/10.1038/s41550-017-0194 Nature Astronomy, https://ui.adsabs.harvard.edu/abs/2017NatAs...1E.194P 1, 0194

Padovani P., Bonzini M., Miller N., Kellermann K. I., Mainieri V., Rosati P., Tozzi P., Vattakunnel S., 2014, in Mickaelian A. M., Sanders D. B., eds, IAU Symposium Vol. 304, Multiwavelength AGN Surveys and Studies. pp 79–85 (http://arxiv.org/abs/1401.1342 arXiv:1401.1342), http://dx.doi.org/10.1017/S1743921314003391 doi:10.1017/S1743921314003391

Parmar A., Katariya R., Patel V., 2019, in Hemanth J., Fernando X., Lafata P., Baig Z., eds, International Conference on Intelligent Data Communication Technologies and Internet of Things (ICICI) 2018. Springer International Publishing, Cham, pp 758–763

Partridge B., 2011, in Carignan C., Combes F., Freeman K. C., eds, IAU Symposium Vol. 277, Tracing the Ancestry of Galaxies. pp 75–78, http://dx.doi.org/10.1017/S1743921311022502 doi:10.1017/S1743921311022502

Pearson K., 1901, http://dx.doi.org/10.1080/14786440109462720 The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 2, 559

Pearson W. J., Wang L., Trayford J. W., Petrillo C. E., van der Tak F. F. S., 2019, http://dx.doi.org/10.1051/0004-6361/201935355 , https://ui.adsabs.harvard.edu/abs/2019AA...626A..49P 626, A49

Peterson B. M., 1997, An Introduction to Active Galactic Nuclei

Peterson B. M., Wandel A., 2000, http://dx.doi.org/10.1086/312862 The Astrophysical Journal, https://ui.adsabs.harvard.edu/abs/2000ApJ...540L..13P 540, L13

Petrosian V., Bykov A. M., 2008, http://dx.doi.org/10.1007/s11214-008-9315-6 , https://ui.adsabs.harvard.edu/abs/2008SSRv..134..207P 134, 207

Polsterer K., Gieseke F., Igel C., Doser B., Gianniotis N., 2016, in Proceedings of the 24th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2016). i6doc.com, pp 405–410

Prandoni I., Seymour N., 2015, in Advancing Astrophysics with the Square Kilometre Array (AASKA14). p. 67 (http://arxiv.org/abs/1412.6512 arXiv:1412.6512)

Pratt L. Y., Mostow J., Kamm C. A., 1991, in Proceedings of the Ninth National Conference

on Artificial Intelligence - Volume 2. AAAI'91. AAAI Press, pp 584–589, http://dl.acm.org/citation.cfm?id=1865756.1865767

Pritchard J. R., Loeb A., 2012, http://dx.doi.org/10.1088/0034-4885/75/8/086901 Reports on Progress in Physics, https://ui.adsabs.harvard.edu/abs/2012RPPh...75h6901P 75, 086901

Quattoni A., Collins M., Darrell T., 2008, 2008 IEEE Conference on Computer Vision and Pattern Recognition, pp 1–8

Raccanelli A., et al., 2012, http://dx.doi.org/10.1111/j.1365-2966.2012.20634.x , https://ui.adsabs.harvard.edu/abs/2012MNRAS.424..801R 424, 801

Radhakrishnan V., 1999, in Taylor G. B., Carilli C. L., Perley R. A., eds, Astronomical Society of the Pacific Conference Series Vol. 180, Synthesis Imaging in Radio Astronomy II. p. 671

Rafferty D. M. N., 2016, PyBDSF Documentation

Ralph N. O., et al., 2019, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2019arXiv190602864R p. arXiv:1906.02864

Rees M. J., 1967, http://dx.doi.org/10.1093/mnras/137.4.429 , https://ui.adsabs.harvard.edu/abs/1967MNRAS.137..429R 137, 429

Rees M. J., 1982, in Heeschen D. S., Wade C. M., eds, IAU Symposium Vol. 97, Extragalactic Radio Sources. pp 211–221

Rees M. J., Phinney E. S., Begelman M. C., Blandford R. D., 1982, http://dx.doi.org/10.1038/295017a0 Nature, 295, 17

Regier J., McAuliffe J., Prabhat M., 2015.

Rhee J., Lah P., Briggs F. H., Chengalur J. N., Colless M., Willner S. P., Ashby M. L. N., Le Fèvre O., 2018, http://dx.doi.org/10.1093/mnras/stx2461 , https://ui.adsabs.harvard.edu/abs/2018MNRAS.473.1879R 473, 1879

Robotham A., Davies L., Driver S., S.Koushan Taranu D., Casura S., Liske J., 2018, ProFound: Source Extraction and Application to Modern Survey Data. https://github.com/asgr/ProFound

Rogstad D. H., Ekers R. D., 1969, http://dx.doi.org/10.1086/150089 , https://ui.adsabs.harvard.edu/abs/1969ApJ...157..481R 157, 481

Rogstad D. H., Shostak G. S., 1972, http://dx.doi.org/10.1086/151636 , https://ui.adsabs.harvard.edu/abs/1972ApJ...176..315R 176, 315

Rokach L., Maimon O., 2005, Clustering Methods. Springer US, Boston, MA, pp 321–352, http://dx.doi.org/10.1007/0-387-25465-X_15 doi:10.1007/0-387-25465-X_15, https://doi.org/10.1007/0-387-25465-X_15

Romano R. A., Aragon C. R., Ding C., 2006, in Proceedings of the 5th International Conference on Machine Learning and Applications. ICMLA '06. IEEE Computer Society, Washington, DC, USA, pp 77–82, http://dx.doi.org/10.1109/ICMLA.2006.49 doi:10.1109/ICMLA.2006.49, http://dx.doi.org/10.1109/ICMLA.2006.49

Ronen S., Aragón-Salamanca A., Lahav O., 1999, http://dx.doi.org/10.1046/j.1365-8711.1999.02222.x Monthly Notices of the Royal Astronomical Society, 303, 284

Rumelhart D. E., Hinton G. E., Williams R. J., 1986, http://dx.doi.org/10.1038/323533a0 , https://ui.adsabs.harvard.edu/abs/1986Natur.323..533R 323, 533

Russakovsky O., et al., 2014, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2014arXiv1409.0575R p. arXiv:1409.0575

Russell S., Norvig P., 2009, Artificial Intelligence: A Modern Approach, 3rd edn. Prentice Hall Press, Upper Saddle River, NJ, USA

S. R. Offner S., Liu Y., 2018, Nature Astronomy, 2, 896–900

Sabater J., et al., 2019, http://dx.doi.org/10.1051/0004-6361/201833883 , https://ui.adsabs.harvard.edu/abs/2019AA...622A..17S 622, A17

Sabour S., Frosst N., Hinton G. E., 2017, in Proceedings of the 31st International Conference on Neural Information Processing Systems. NIPS'17. Curran Associates Inc., USA, pp 3859–3869, http://dl.acm.org/citation.cfm?id=3294996.3295142

Sadeh I., Abdalla F. B., Lahav O., 2016, http://dx.doi.org/10.1088/1538-3873/128/968/104502 , https://ui.adsabs.harvard.edu/abs/2016PASP..128j4502S 128, 104502

Sadler E. M., Jenkins C. R., Kotanyi C. G., 1989, http://dx.doi.org/10.1093/mnras/240.3.591 , https://ui.adsabs.harvard.edu/abs/1989MNRAS.240..591S 240, 591

Saikia D. J., Jamrozy M., 2009, Bulletin of the Astronomical Society of India, https://ui.adsabs.harvard.edu/abs/2009BASI...37...63S 37, 63

Saito T., Rehmsmeier M., 2015, http://dx.doi.org/10.1371/journal.pone.0118432 PLOS ONE, 10, 1

Sarazin C. L., 2002, in Feretti L., Gioia I. M., Giovannini G., eds, Astrophysics and Space Science Library Vol. 272, Merging Processes in Galaxy

Clusters. pp 1–38 (http://arxiv.org/abs/astro-ph/0105418 arXiv:astro-ph/0105418), http://dx.doi.org/10.1007/0-306-48096-4_1 doi:10.1007/0-306-48096-4_1

Saripalli L., 2012, http://dx.doi.org/10.1088/0004-6256/144/3/85 The Astronomical Journal, 144, 85

Sarro L. M., Sánchez-Fernández C., Giménez Á., 2006, http://dx.doi.org/10.1051/0004-6361:20052830 , https://ui.adsabs.harvard.edu/abs/2006A

Savage R. S., Oliver S., 2007, http://dx.doi.org/10.1086/515393 The Astrophysical Journal, 661, 1339

Scheuer P. A. G., Williams P. J. S., 1968, http://dx.doi.org/10.1146/annurev.aa.06.090168.001541 , https://ui.adsabs.harvard.edu/abs/1968ARA

Schmidhuber J., 2015, http://dx.doi.org/10.1016/j.neunet.2014.09.003 Neural Networks, 61, 85

Schmidt M., 1963, http://dx.doi.org/10.1038/1971040a0 , https://ui.adsabs.harvard.edu/abs/1963Natur.197.1040S 197, 1040

Schoenmakers A. P., de Bruyn A. G., Röttgering H. J. A., van der Laan H., Kaiser C. R., 2000, http://dx.doi.org/10.1046/j.1365-8711.2000.03430.x Monthly Notices of the Royal Astronomical Society, 315, 371

Sérsic J. L., 1963, Boletin de la Asociacion Argentina de Astronomia La Plata Argentina, https://ui.adsabs.harvard.edu/abs/1963BAAA....6...41S 6, 41

Shabala S. S., Ash S., Alexander P., Riley J. M., 2008, http://dx.doi.org/10.1111/j.1365-2966.2008.13459.x , https://ui.adsabs.harvard.edu/abs/2008MNRAS.388..625S 388, 625

Shakeshaft J. R., Ryle M., Baldwin J. E., Elsmore B., Thomson J. H., 1955, , https://ui.adsabs.harvard.edu/abs/1955MmRAS..67..106S 67, 106

Shankar F., Weinberg D. H., Miralda-Escudé J., 2009, http://dx.doi.org/10.1088/0004-637X/690/1/20 , https://ui.adsabs.harvard.edu/abs/2009ApJ...690...20S 690, 20

Shen H., George D., Huerta E. A., Zhao Z., 2017, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2017arXiv171109919S p. arXiv:1711.09919

Shimwell, T. W. et al., 2017, http://dx.doi.org/10.1051/0004-6361/201629313 A&A, 598, A104

Shimwell, T. W. et al., 2019, http://dx.doi.org/10.1051/0004-6361/201833559 A&A, 622, A1

Shirasaki M., Yoshida N., Ikeda S., 2019, http://dx.doi.org/10.1103/PhysRevD.100.043527 , https://ui.adsabs.harvard.edu/abs/2019PhRvD.100d3527S 100, 043527

Silk J., 2011, in Carignan C., Combes F., Freeman K. C., eds, IAU Symposium Vol. 277, Tracing the Ancestry of Galaxies. pp 273–281 (http://arxiv.org/abs/1102.0283 arXiv:1102.0283), http://dx.doi.org/10.1017/S1743921311022939 doi:10.1017/S1743921311022939

Simonyan K., Zisserman A., 2014, CoRR, abs/1409.1556

Simpson C., 2017, http://dx.doi.org/10.1098/rsos.170522 Royal Society Open Science, 4, 170522

Snyder G. F., Rodriguez-Gomez V., Lotz J. M., Torrey P., Quirk A. C. N., Hernquist L., Vogelsberger M., Freeman P. E., 2019, http://dx.doi.org/10.1093/mnras/stz1059 , https://ui.adsabs.harvard.edu/abs/2019MNRAS.486.3702S 486, 3702

Srivastava N., Hinton G., Krizhevsky A., Sutskever I., Salakhutdinov R., 2014, Journal of Machine Learning Research, 15, 1929

Stehman S. V., 1997, http://dx.doi.org/https://doi.org/10.1016/S0034-4257(97)00083-7 Remote Sensing of Environment, 62, 77

Storm E., Jeltema T. E., Profumo S., Rudnick L., 2013, http://dx.doi.org/10.1088/0004-637X/768/2/106 , https://ui.adsabs.harvard.edu/abs/2013ApJ...768..106S 768, 106

Storrie-Lombardi M. C., Lahav O., Sodre Jr. L., Storrie-Lombardi L. J., 1992, http://dx.doi.org/10.1093/mnras/259.1.8P , https://ui.adsabs.harvard.edu/abs/1992MNRAS.259P...8S 259, 8P

Strateva I., et al., 2001, http://dx.doi.org/10.1086/323301 , https://ui.adsabs.harvard.edu/abs/2001AJ....122.1861S 122, 1861

Subrahmanyan R., Saripalli L., W. Hunstead R., 1996, http://dx.doi.org/10.1093/mnras/279.1.257 Monthly Notices of the Royal Astronomical Society, 279, 257

Sutherland W., Saunders W., 1992, http://dx.doi.org/10.1093/mnras/259.3.413 , https://ui.adsabs.harvard.edu/abs/1992MNRAS.259..413S 259, 413

Sutskever I., 2013, PhD thesis, Toronto, Ont., Canada, Canada

Szegedy C., Zaremba W., Sutskever I., Bruna J., Erhan D., Goodfellow I., Fergus R., 2013, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2013arXiv1312.6199S p. arXiv:1312.6199

Szegedy C., Ioffe S., Vanhoucke V., Alemi A., 2016, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2016arXiv160207261S p. arXiv:1602.07261

Tabatabaei F., 2017, Radio astronomers tune in to the star formation channel

Tabatabaei F. S., et al., 2017, http://dx.doi.org/10.3847/1538-4357/836/2/185 , https://ui.adsabs.harvard.edu/abs/2017ApJ...836..185T 836, 185

Tachibana Y., Miller A. A., 2018, http://dx.doi.org/10.1088/1538-3873/aae3d9 , https://ui.adsabs.harvard.edu/abs/2018PASP..130l8001T 130, 128001

Tagliaferri R., et al., 2003, http://dx.doi.org/https://doi.org/10.1016/S0893-6080(03)00028-5 Neural Networks, 16, 297

Tan C., Sun F., Kong T., Zhang W., Yang C., Liu C., 2018, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2018arXiv180801974T p. arXiv:1808.01974

Tang H., M. M. Scaife A., P. Leahy J., 2019

Tikhonov A. N., Arsenin V. Y., 1977, Solutions of ill-posed problems. V. H. Winston & Sons, Washington, D.C.: John Wiley & Sons, New York

Tishby N., Pereira F. C., Bialek W., 1999. pp 368–377

Torniainen I., et al., 2008, http://dx.doi.org/10.1051/0004-6361:20079222 , https://ui.adsabs.harvard.edu/abs/2008A

Troland T. H., Heiles C., 1982, http://dx.doi.org/10.1086/159544 , https://ui.adsabs.harvard.edu/abs/1982ApJ...252..179T 252, 179

Trombetti T., Burigana C., 2018, http://dx.doi.org/10.3389/fspas.2018.00033 Frontiers in Astronomy and Space Sciences, 5, 33

Vafaei S. A., Vos E. E., Bassett B. A., Hosenie Z., Oozeer N., Lochner M., 2019, http://dx.doi.org/10.1093/mnras/stz131 Monthly Notices of the Royal Astronomical Society, 484, 2793

Van Oort C. M., Xu D., Offner S. S. R., Gutermuth R. A., 2019, http://dx.doi.org/10.3847/1538-4357/ab275e , https://ui.adsabs.harvard.edu/abs/2019ApJ...880...83V 880, 83

Vasconcellos E. C., de Carvalho R. R., Gal R. R., LaBarbera F. L., Capelato H. V., Velho H. F. C., Trevisan M., Ruiz R. S. R., 2011, http://dx.doi.org/10.1088/0004-6256/141/6/189 The Astronomical Journal, 141, 189

Vincent P., Larochelle H., Bengio Y., Manzagol P.-A., 2008, in Proceedings of the 25th International Conference on Machine Learning. ICML '08. ACM, New York, NY, USA, pp 1096–1103, http://dx.doi.org/10.1145/1390156.1390294 doi:10.1145/1390156.1390294, http://doi.acm.org/10.1145/1390156.1390294

Vluymans S., 2018, PhD thesis, Ghent University

W. Shimwell T., et al., 2016

Wadadekar Y., 2005, Publications of the Astronomical Society of the Pacific, pp 79–85

Wald A., 1949, http://dx.doi.org/10.1214/aoms/1177730030 Ann. Math. Statist., 20, 165

Weaver W. B., Torres-Dodgen A. V., 1997, http://dx.doi.org/10.1086/304651 The Astrophysical Journal, 487, 847

Webber W. R., Simpson G. A., Cane H. V., 1980, http://dx.doi.org/10.1086/157761 , https://ui.adsabs.harvard.edu/abs/1980ApJ...236..448W 236, 448

Weir N., Fayyad U. M., Djorgovski S., 1995, http://dx.doi.org/10.1086/117459 , https://ui.adsabs.harvard.edu/abs/1995AJ....109.2401W 109, 2401

White S. D. M., Rees M. J., 1978, http://dx.doi.org/10.1093/mnras/183.3.341 , https://ui.adsabs.harvard.edu/abs/1978MNRAS.183..341W 183, 341

White R. L., et al., 2000, http://dx.doi.org/10.1086/313300 , https://ui.adsabs.harvard.edu/abs/2000ApJS..126..133W 126, 133

Willett K., 2015, in The Many Facets of Extragalactic Radio Surveys: Towards New Scientific Challenges. p. 8

Willett K. W., et al., 2013, http://dx.doi.org/10.1093/mnras/stt1458 , https://ui.adsabs.harvard.edu/abs/2013MNRAS.435.2835W 435, 2835

Williams, W. L. et al., 2019, http://dx.doi.org/10.1051/0004-6361/201833564 A&A, 622, A2

Willis A. G., Strom R. G., Wilson A. S., 1974, http://dx.doi.org/10.1038/250625a0 , https://ui.adsabs.harvard.edu/abs/1974Natur.250..625W 250, 625

Wong S. C., Gatt A., Stamatescu V., McDonnell M. D., 2016, 2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA), pp 1–6

Wright D. E., et al., 2015, http://dx.doi.org/10.1093/mnras/stv292 , https://ui.adsabs.harvard.edu/abs/2015MNRAS.449..451W 449, 451

Wrobel J. M., Walker R. C., 1999, in Taylor G. B., Carilli C. L., Perley R. A., eds, Astronomical Society of the Pacific Conference Series Vol. 180, Synthesis Imaging in Radio Astronomy II. p. 171

Wu J. F., Boada S., 2019, http://dx.doi.org/10.1093/mnras/stz333 , https://ui.adsabs.harvard.edu/abs/2019MNRAS.484.4683W 484, 4683

Wu C., et al., 2018, http://dx.doi.org/10.1093/mnras/sty2646 Monthly Notices of the Royal Astronomical Society, 482, 1211

Xi E., Bing S., Jin Y., 2017, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2017arXiv171203480X p. arXiv:1712.03480

Yatawatta S., 2008

Yip C. W., et al., 2004, http://dx.doi.org/10.1086/422429 , https://ui.adsabs.harvard.edu/abs/2004AJ....128..585Y 128, 585

Yosinski J., Clune J., Bengio Y., Lipson H., 2014, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2014arXiv1411.1792Y p. arXiv:1411.1792

Zahavy T., Kang B., Sivak A., Feng J., Xu H., Mannor S., 2016, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2016arXiv160202389Z p. arXiv:1602.02389

Zeiler M. D., 2012, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2012arXiv1212.5701Z p. arXiv:1212.5701

Zeiler M. D., Fergus R., 2013, arXiv e-prints, https://ui.adsabs.harvard.edu/abs/2013arXiv1311.2901Z p. arXiv:1311.2901

Zevin M., et al., 2017, http://dx.doi.org/10.1088/1361-6382/aa5cea Classical and Quantum Gravity, 34, 064003

Zhang Y., Zhao Y., 2014, in Manset N., Forshay P., eds, Astronomical Society of the Pacific Conference Series Vol. 485, Astronomical Data Analysis Software and Systems XXIII. p. 239

Zheng C., Pulido J., Thorman P., Hamann B., 2015, http://dx.doi.org/10.1093/mnras/stv1237 Monthly Notices of the Royal Astronomical Society, 451, 4445

Zhu X.-P., Dai J.-M., Bian C.-J., Chen Y., Chen S., Hu C., 2019, http://dx.doi.org/10.1007/s10509-019-3540-1 , https://ui.adsabs.harvard.edu/abs/2019Ap

de Gasperin F., Intema H. T., Frail D. A., 2018, http://dx.doi.org/10.1093/mnras/stx3125 , https://ui.adsabs.harvard.edu/abs/2018MNRAS.474.5008D 474, 5008

de Hulst H. V., 1945, Ned.Tijd.Natuurkunde, 11, 210

de Zotti G., Ricci R., Mesa D., Silva L., Mazzotta P., Toffolatti L., González-Nuevo J., 2005, http://dx.doi.org/10.1051/0004-6361:20042108 , https://ui.adsabs.harvard.edu/abs/2005AA...431..893D 431, 893

van Haarlem, M. P. et al., 2013, http://dx.doi.org/10.1051/0004-6361/201220873 A&A, 556, A2

van Velzen, Sjoert Falcke, Heino Schellart, Pim Nierstenhöfer, Nils Kampert, Karl-Heinz 2012, http://dx.doi.org/10.1051/0004-6361/201219389 A&A, 544, A18

van Weeren R. J., de Gasperin F., Akamatsu H., Brüggen M., Feretti L., Kang H., Stroe A., Zandanel F., 2019, http://dx.doi.org/10.1007/s11214-019-0584-z Space Science Reviews, 215, 16