Classification of Deep Space Objects using Deep Learning Techniques

MSc Research Project
MSc in Data Analytics

Cillín Ó Foghlú
Student ID: 18186751

School of Computing

National College of Ireland

Supervisor: Dr Catherine Mulwa

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Cillín Ó Foghlú		
Student ID:	18186751		
Programme:	MSc in Data Analytics	**Year:**	2020
Module:	Research Project		
Supervisor:	Dr Catherine Mulwa		
Submission Due Date:	August 2020		
Project Title:	Classification of Deep Space Objects using Deep Learning Techniques		
Word Count:	………11045……………… **Page Count**………27……………………………………		

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ………………Cillín Ó Foghlú…………………………………………………………………
Date: ………………25/09/2020…………………………………………………………………………

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Classification of Deep Space Objects using Deep Learning Techniques

Cillín Ó Foghlú

18186751

Abstract

Throughout the ages people have gazed into the nights sky and wondered. This research looked at the use of modern computer image recognition algorithms and reviewed some of the best performing against each other to see how they adapted to deep space object imagery. Images from both the Slone Deep Space Survey and the Space Telescope Science Institute Kepler images were used as training input to deep learning models – ResNet50, VGG16, Xception and MobileNet. The model's performance is validated against unseen images and the level of accuracy used to ascertain their performance. Using pre-trained models as a base then project shows that these models can be trained to leant new features and classify deep space objects with accuracy of 80% plus. This will allow astronomers to focus their limited telescope time on the objects of greatest interest. The resents of review literature and identified gaps are also presented.

Keywords: CNN, Slone Deep Space Survey, STScI, Image Classification, Astronomy

1 Introduction

In many fields of data analysis, the biggest limitation is the availability of new data to analyse. Many researchers spend months, if not years, gathering sufficient data to meet their analytic needs. In the field of astronomy, the problem is the reverse. People has looked up at the stars for centuries and in the last 50 years we have used more and more powerful telescopes to capture data on what we saw. The field of Astroinformatics is where data analytics and Astronomy meet, by expert knowledge, statistical analysis, machine learning methodologies, and deep learning to assist astronomers to mine existing datasets and learn more about the universe. Advancements in machine learning and especially deep learning allow this to be completed at speeds which were impossible only a relatively few years ago.

1.1 Motivation and Background

In 2007 astrophysicist Kevin Schawinski from Oxford University had over 900,000 images of galaxies from the Sloan Digital Sky Survey to review, when he decided that there had be a better way to complete the work. He along with a fellow at Oxford college, Chris Lintott looked to armatures to assist in the work and Galaxy Zoo was the result. The work which had been expected to take years to complete was completed in just 6 months. In 2000 the Slone Deep Space Survey telescope was commissioned and since then has being completing surveys of the night sky. Data releases 16 of the Slone survey was announced in 2019 (Ahumada et al. 2020) and detailed the additions to existing released data. Up to 2018 it was estimated that SDSS had imaged about one-third of the night sky across five broad bands (*ugriz*), accounting for almost half a billion unique objects. The Dark Energy Spectroscopic Instrument (US Department of Energy Office of Science, 2018), started its commissioning process in April 2019 and when in production, will gather data on tens of millions of galaxies and quasars to construct a 3D map of the universe out to 100 billion light years. The LSST (Large Synoptic

Survey Telescope, 2020) is expected to be commissioned in the 2020's and the initial 10-year project will collect more a 500-petabyte dataset of images, Figure 1, shows the expected growth per night in the volumes of data for existing and future telescopes - Very Large Telescope (VLT), Sloan Digital Sky Survey (SDSS), Visible and Infrared Telescope for Astronomy (VISTA), Large Synoptic Survey Telescope.

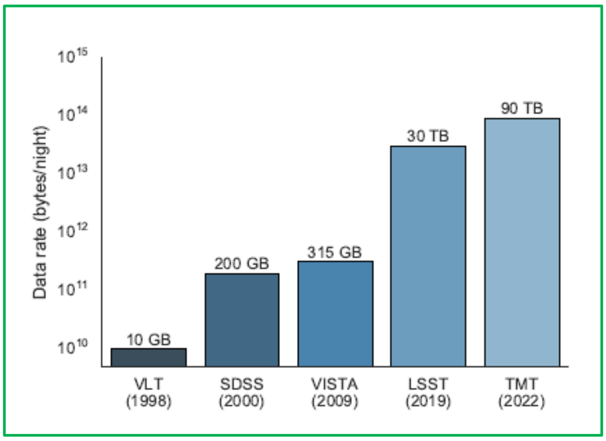


Figure 1: Increasing data volumes of existing and upcoming telescopes

Deep Space imagery provided one of the largest freely available data assets for data scientists to explore. It provides the vast amounts of imagery and associated classifications, as well as a growing need for new methodologies and tools to process the data in support of a real-world need. By classifying objects in space, data scientists can assist astronomers better identify objects which can be investigated using their limited resources. As an amateur photographer who has dabbled in astrophotography from time to time, I was drawn to this field, both out of curiosity and from a technical / image classification point of view. The projects goal is to give everyone a way to be part of the exploration in much the same way that SETI is in the search of extra-terrestrial life,

1.2 Research Question and Objectives

Telescope time is a costly and finite resource. It is further limited by adverse weather and sometimes atmospheric conditions. The problem addressed here was how best to maximize the output from deep space imagery through more selective approach in subject selection. This will assist astronomers to gain greater understanding of our universe. This research addresses the problem faced by astronomers currently, namely, how to deal with the growing amount of data being captured today and to identify tools to allow future new telescopes, improvements in astronomical imagery and process increasing volumes of data, while also ensuring that objects of astronomical interest are thoroughly investigated using the finite resources at their disposal.

> ***RQ:*** *""To what extent can the current best practices in deep learning and image classification (ResNet50, VGG16, Xception and MobileNet) be trained to classify deep space imagery to support astronomer to learn more about deep space objects?"* In addition, an assessment of current images from deep space telescopes was conducted with a view to their suitability as input to deep learning classification models.

The objectives and what each delivered is outlined in Table 1. This covers the main objectives as well as several minor objectives.

Table 1 : Research Objectives

Objectives	Description	Evaluation Methods
Objective 1	A review of the machine learning algorithms used currently to assist in imagery classification	
Objective 2	Data extraction and pre-processing	
Objective 3	Identification of new image recognition development in computer vision analysis	
Objective 4	Implementation of deep learning and image classification models for astronomical deep space imagery	
Objective 4 a	Acquire data for training and testing of the chosen design	
Objective 4 b	Assessing and tuning the classification model	
Objective 4 c	Build and evaluate using MobileNet model	Accuracy
Objective 4 d	Build and evaluate using ResNet50 model	
Objective 4 e	Build and evaluate using VGC16 model	
Objective 4 f	Build and evaluate using Xception model	
Objective 5	Comparison of model's performance	
Objective 6	Comparison of models' performance against industry state of the art models	

1.3 Contributions to the body of knowledge

The research question was investigated through the use computer vision tools to recognise images from the Slone Deep Space Survey and Space Telescope Science Institute (STScI) images of Galaxies, Quasars and Stars. Images were PNG and JPEG formats. The images extracted from FITS files were from the SDSS archives which came from the Data Release 12 (DR12) servers, which contained all observations up to July 2014 - as well as jpeg images from the later DR16 servers – which were released in 2019 and contained observations up to August 2018. The STScI contain NASA images from both Hubble and Kepler space telescope. The work carried out here processed these images to allow their input into several artificial neural networks and based on training datasets to teach these models to classify the objects in the images. The models were then assessed to state-of-the-art results which were achieved against the ImageNet dataset, to assess their performance.

The rest of the report is structured as follows: Section 2 is a critical review of relevant research completed to date and identifies the rationale behind the identification of the gaps identified by this research. Section3 describes how this project followed a well-established scientific methodology. Section 4 presents the design, pre-processing. Section 5 the implementation steps are described. Section 0 details the comparison and evaluation of the model chosen, Section 7 concludes the project and identifies future work which could be completed as a result of this project. Section 8 acknowledges the support this work received.

2 Literature Review into the field of Astronomical Imagery Classification

A cornerstone to scientific research is context and this is best done by reviewing the existing work in the field to identify gaps or opportunities to advance the field of research. To look deeper into space is to look back in time and man continues to peer deeper and deeper into an unending universe.

2.1 Introduction

Research into the field of astronomy dates back many centuries and it is considered the oldest of the natural sciences. While there is no definition as to when modern astronomy started, it is commonly accepted that it was with Nicolaus Copernicus and his model of the solar system and the work by Johannes Kepler, in the 17th century, who described the details of the motion of the planets around the sun for our solar system. The invention of the reflecting telescope by Sir Isaac Newton which led to a catalogue of over 3000 stars (Flemsteed, 1725) being published, the First Astronomer Royal. The birth of radio astronomy was in 1931 when Karl Guthe Jansky discovered "Cosmic Static" being emitted by the Milky Way (Jansky 1958) In 1937 when the first purpose build radio telescope was constructed by Grote Reber in Wheaton, Illinois and he described his findings "Radio-frequency investigations of astronomical interest" (Reber & Greenstein, 1947).

2.2 Deep Space Imagery

Currently deep space image surveys capture data in one of two forms, spectroscopic or photometric. Spectroscopic measures the wavelengths of photons across thousands of wavelengths, thus allowing for the identification of chemical compounds, such the presence of water. Photometry uses a Charge-Coupled Device, CCD, to measure only a handful of broad-band filters, resulting in less detailed data than photometry. As always, there is a trade-off between the sensitivity of spectroscopic, not being able to measure faint or distant objects, and cost in only being able to measure a smaller number of objects in a single image. Fainter objects are further away, letting astronomers look further back in time. This allows them to peer into the early universe and understand the fundamentals of life and our existence. Much of the resulting images still require manual manipulation and classification, generally due to several artefacts and other distortions which make it difficult for machine learning to process currently. These can include, digital noise due to the nature of long exposures, merging galaxies or galaxies along the same line of sight or even space dust.

2.3 Critical Review of methods used to classify astronomical objects

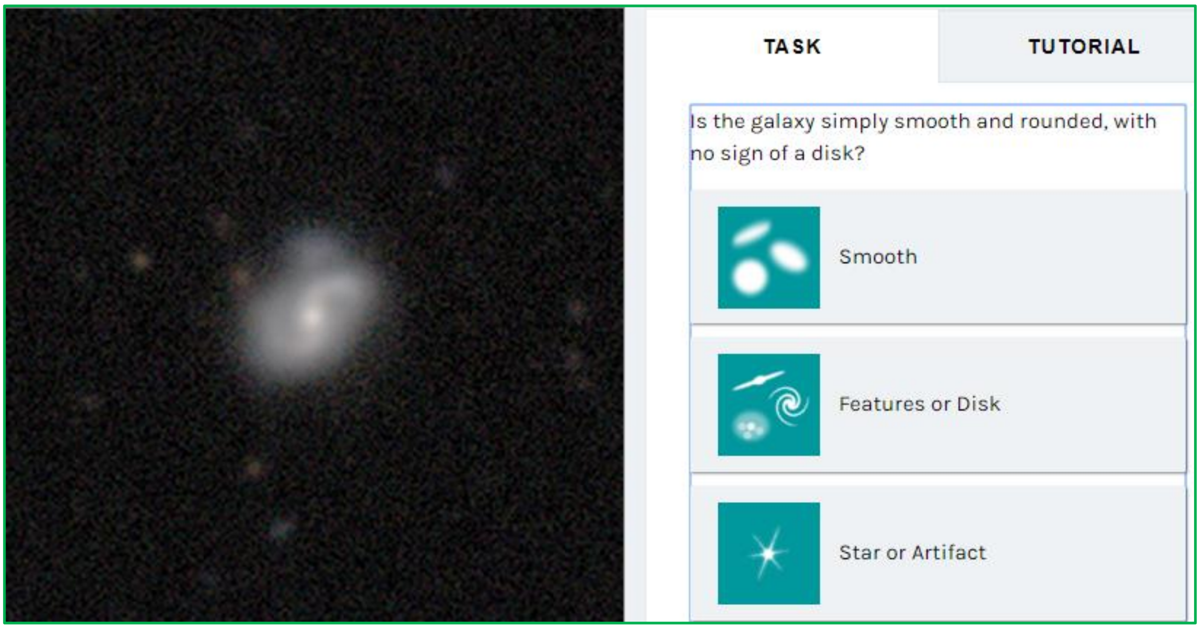
There were two categories which the classification of the data fell into, it was either traditional methods, using experts or large number of amateurs and requiring multiple positive results to determine high probability of accuracy or machine learning algorithms, which is relatively new to this subject area.

2.4 Non-Machine Learning Methods

Traditionally the interpertation and classification of astronomical imagery required an expert with a deep understanding of the subject. This was a slow process and open to human error. The result was that there was a backlog in inages which needed to be reviewed and interperated.

This problem was partially overcome in the early part of the 21st centruary with the Galaxy Zoo proect.

Galazy Zoo (Lintott et al. 2008) has helped astomnomers and obtained more than 4×10^7 individual classifications made by $\sim 10^5$ participants classifications of galaxy morphologies through direct visual inspection of images. Participants classified the shape of a galaxy and how dense it was, see Figure 2 The same images were presented to a number of different participants and this allowed the probability of the classification being correct to be increased. It is estimated that the human eye classification used here actieved a 94.5% accuracy when using 2 classes of galaxies but lewered to 65.2% when 11 classes were presented.



Is the galaxy simply smooth and rounded, with no sign of a disk?

Figure 2: Sample of Image from Galazy Zoo Tutorial

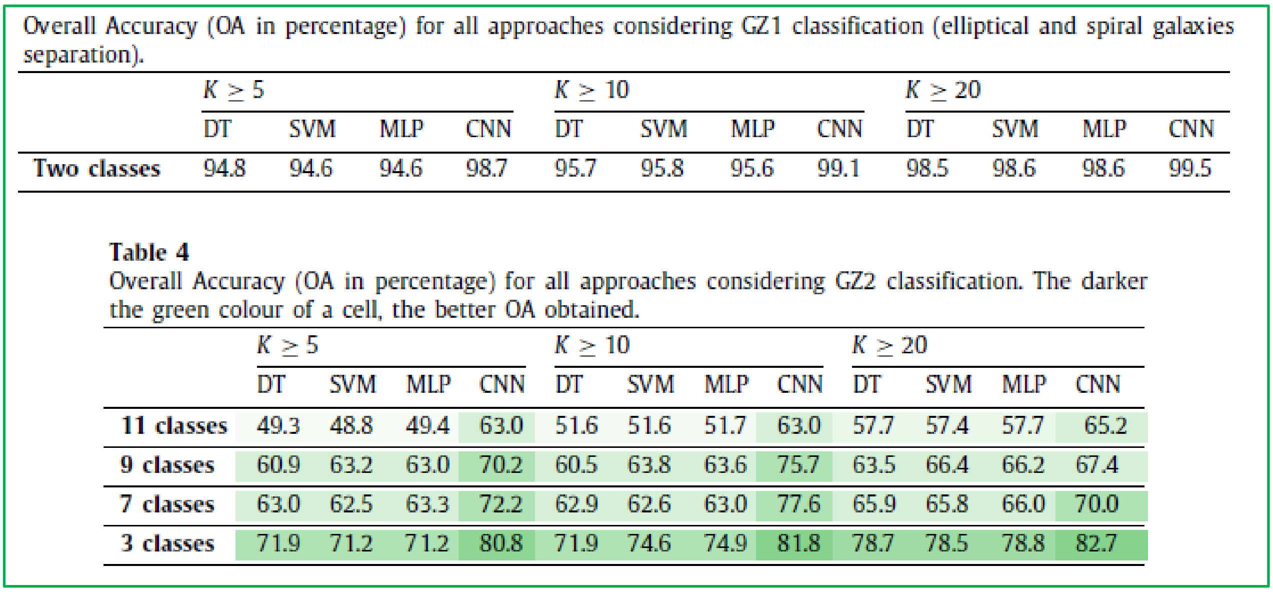
Redshift, is a key feature used to determain the distance of a galaxy from Earth and research into the distribution of galaxies using statistical methods (Wray and Gunn 2008) gave a “*rms z of 0.025 for red galaxies and 0.030 for blue galaxies (all with z < 0:25*)”. The research review how supervised neural networks had bene used to compute a photo-z’s from a range of iparameters from Petrosian radii, cnecentration index, surface brightness and axial ratios. The paper looked at using surface brightness and the Sérsic index (measure of radial light profile) as well as five-band photometry (u,g,r,i,z) which range from near ultra-violet to near infrared.. The data used as a subsection of the SDSS catelogue containing 55,405 galaxies. The statistical analysis used was acknowledged as being of great importance as it was all based on emperical data, it acknowledged the scale of the problems, which in 2008, that astronomers faced with very large data sets. The work completed was deemed as impractable for large data sets.

2.4.1 Machine Learning Methods

In their paper on Big Universe, Big Data (Kremer et al. 2017) identified that the Sloan Digital Sky Survey generated 200Gb every night and the volumns of data being produced every night amount to what would have taken a decate in the past. While the paper also looked at some of the algorythms being used, it also identified one issue which most machine learning, ML, alogrythms need to be very careful when classifying objects, nanely bias within the data. In order to allow for the bias importance-weighting was used – giving more weight to samples which are under represented and lesser weights to samples with higher representation. KNN models were used but proved to be computationally expensive – details on the hardware used is not provided in the paper – when spatial data structure were used as is the case with KNN and the unsuitabile for GPU's to process as this model was identified as not parallisable. The team then moved onto developing a new tree structure to run across numerour GPU's thus reducing the bottleneck. This model runs across multiple devices and there is only soutable to universities or institutions. This goal of this project is to find a way for non-accidemics to participate in the classification and ideinification of objects in deep space surveys there the model from Kremer et al was deemed as not appicable and further research was required. Using the ZsoltFrei Catalogue from the department of Astrophysical Sciences at Princeton University, which contained 113 different galaxies (Abd Elfattah et al. 2014) explored empirical decompossition to extract features and neural networks as a classifier for galaxies morphology. The types of neural networks used were multi-layer perceptron neural network based classifier, Generalized feed-forward networks, Recurrent networks. In order to assess the performance of the models used, Mean-Square Error (MSE); Normalized Mean Square Error (NMSE); Correlation Coefficient (R); and Error Percentage were used as measures. 26 images were selected for training and the remaining for testing. It was not surprising to note that this model returned a 99% accuracy given the small number of images and that the data used was of extreme high quality. It was also concluded that only a small number of features were required to successfully classify the images. While the paper did give a good foundation on the use of neural networks potential in classification of images, the conclusions were deemed to be based on too small a dataset and no details on the feature set was included in the paper.

A more comprehensive data set was used (Barchi et al. 2020), the SDSS Data Release 7 (Eisenstein, et al., 2011) and the Galaxy Zoo catalogues (Lintott, et al., 2008) used 1 million galaxies were used, with 80% for training and the balance for testing. The work provided a catalogue of 670,560 galaxies with morphological metrics and classifications. The results show that both deep and traditional ML gave a 94.5% accuracy with two classes of galaxies and 82% when using 3 classes. The work is a comprehensive review of the state of ML and image classification in the field of astronomy. The techniques used include Convolutional Neural Networks and what they referred to as Traditional ML and Deep Learning which looked at the impact of parameters on the Traditional ML One point made in this report is that Deep ML models require a large amount of data and can be difficult to train and tune. This difficulty in training and tuning still leaves a place for traditional machine learning. Algorithms used in this paper covered CyMorph - a non-parametric galaxy morphology system which determines Concentration (C), Asymmetry (A), Smoothness (S), Entropy (H) and Gradient Pattern Analysis (GPA) metrics. For machine learning Decision Trees, Support Vector Machines and Multilayer Perceptron models were used in Python. For their Deep Learning, Residual Networks (ResNet) (He, et al., 2016)) and GoogleNet (Szegedy, et al., 2015) were deployed. Different models returned overall accuracy as detailed below in Table 2 below, where a parameter K as the area of the galaxy's Petrosian ellipse divided by the area of the Full Width at Half Maximum (FWHM).

Table 2 : P.H. Barchi et al . Results



Overall Accuracy (OA in percentage) for all approaches considering GZ1 classification (elliptical and spiral galaxies separation).

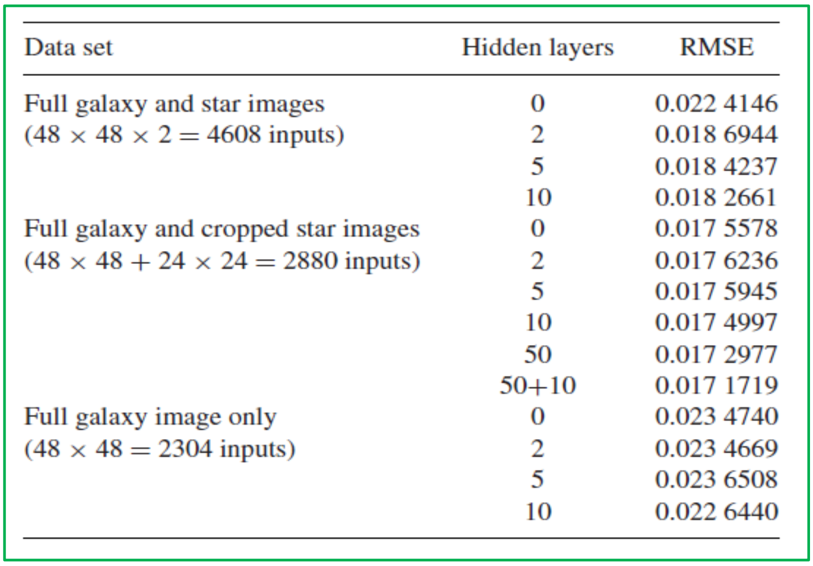
	$K \geq 5$				$K \geq 10$				$K \geq 20$			
	DT	SVM	MLP	CNN	DT	SVM	MLP	CNN	DT	SVM	MLP	CNN
Two classes	94.8	94.6	94.6	98.7	95.7	95.8	95.6	99.1	98.5	98.6	98.6	99.5

Table 4
Overall Accuracy (OA in percentage) for all approaches considering GZ2 classification. The darker the green colour of a cell, the better OA obtained.

	$K \geq 5$				$K \geq 10$				$K \geq 20$			
	DT	SVM	MLP	CNN	DT	SVM	MLP	CNN	DT	SVM	MLP	CNN
11 classes	49.3	48.8	49.4	63.0	51.6	51.6	51.7	63.0	57.7	57.4	57.7	65.2
9 classes	60.9	63.2	63.0	70.2	60.5	63.8	63.6	75.7	63.5	66.4	66.2	67.4
7 classes	63.0	62.5	63.3	72.2	62.9	62.6	63.0	77.6	65.9	65.8	66.0	70.0
3 classes	71.9	71.2	71.2	80.8	71.9	74.6	74.9	81.8	78.7	78.5	78.8	82.7

SkyNet was publicly released (Graff et al. 2014) as a tool to assist astronomers to train feed-forward NN in both supervised and unsupervised methods and allowed for regression, classification, density estimation, clustering and dimensionality reduction. The authors identified that while neural networks had been in use for over 20 years, their difficulty in training, use of backpropagation and complexity limited their deployment. Key points made in this research was that the number of layers in a NN was dependant on the number of input data points and there was a difficulty in choosing the correct number – too few led to the model not being able to learn properly and too many meant that the model would over-fit the training data and slow down the training. Table 3 below details the RMSE on ellipticity predictions for networks with different architectures, evaluated on the 60,000 image pairs of the MDM Challenge. All networks have two outputs: the galaxy ellipticities e1 and e2.

Table 3 : SkyNet Neural Network Performance



Data set	Hidden layers	RMSE
Full galaxy and star images	0	0.022 4146
($48 \times 48 \times 2 = 4608$ inputs)	2	0.018 6944
	5	0.018 4237
	10	0.018 2661
Full galaxy and cropped star images	0	0.017 5578
($48 \times 48 + 24 \times 24 = 2880$ inputs)	2	0.017 6236
	5	0.017 5945
	10	0.017 4997
	50	0.017 2977
	50+10	0.017 1719
Full galaxy image only	0	0.023 4740
($48 \times 48 = 2304$ inputs)	2	0.023 4669
	5	0.023 6508
	10	0.022 6440

Another paper using the SDSS imagery was presented by (du Buisson, et al., 2015) looked at supernova and the identification of artefacts in the images. They noted that the initial work was still done by humans – the removal is noise and unwanted artefacts from the images. Using features trained from eigen image analysis (principal component analysis, PCA) of single-epoch g, r and i difference images, a recall of 96% was achieved. The results also showed that random forests performed best, followed by the *k*-nearest neighbour and the SkyNet (Graff et al. 2014) artificial neural network (ANN) algorithms, compared to naive Bayes and kernel SVM's. This research showed that PCA based ML could match humans for accuracy and that where a multi epoch approach was used that ML could outperform the humans especially at low signal-to-noise. While accuracy was the measure here, it is also clearly identified that human scanning of future imagery from LSST, and more advanced telescopes in the future, would not be feasible but to the number of images being produced every night.

Research by National Astronomical Observatories (Zhao and Zhang 2008) reviewed the effectiveness of a number of decision tree formats - REPTree, Random Tree, Decision Stump, Random Forest, J48, NBTree, AdTree using the WEKA package. They found that "*applied in discriminating active objects (quasars, BL Lac objects and active galaxies) from non-active objects (stars and galaxies), ADTree is the best only in terms of accuracy, Decision Stump is the best only considering speed, J48 is the optimal choice considering both accuracy and speed*". Again, the requirement for large amounts of data required to train neural networks was highlighted. This is not a problem in the field of astronomy; however, it is the requirement to also have large labelled catalogues to go with the data which is the issue. This is becoming more available as professionals continue to process and release catalogues. The underlining assumption is that the classification of the objects used to train the networks is accurate and sufficiently diverse in categories to be of value.

The use of CNN's (Pasquet-Itam & Pasquet, 2018) to detect quasars and to predict photometric redshifts of quasars on data from the SDSS found that their CNN was able to give a precision of 0.988 for a recall of 0.90 compared with a recall of 0.97 for a random forest. Moreover, the research identified 175 new quasar candidates to be investigated. The design CNN used in this model was 4 layers deep, however the library used is not identified. It is also not clear if architecture was 4 hidden layers or 4 layers in total. It seems unusual in the number of layers given the number of parameters identified - 1 802 032 in the convolution layers and 11 468 80 in the fully connected layers. It would therefore be impossible to reproduce the findings or to build upon this research and tune the network further. It was assumed the reason for the high number of parameters was due to flattening the images.

Using over 477,000 objects from SDSS spectroscopic data and decision trees, (Ball, et al., 2006) classified 143 million nonrepeat photometric objects from the SDSS Data Release 3 and was the first public release of object classification of the entire SDSS dataset. Supervised ML was used to manage the decision tree on the Xeon Linus Cluster Tungsten at NCSA. This supercomputer had 1280 nodes each with 2GB or RAM and a peak double-precision performance of 6.4 Gflops. Over 7000 decision trees were analysed, and a probability was given to each result, with the highest being assigned as the classification. The best classification error achieved was noted as being 3:07% ± 0:08%. Currently SDSS is up to Release 16, which is a cumulative release up to August 2018 and was made public in December 2019.

With the SDSS Release 12 (Kheirdastan & Bazarghan, 2016) explored Probabilistic Neural Network (PNN), Support Vector Machine (SVM) and k-means clustering to automate the classification of stellar spectral clusters. They found that PNN's, a mathematical tool to emphasize variation and bring out strong patterns to reduce the dimensionality of data set, outperformed SVM and K-means. The algorithms varied in results, depending on the size of the datasets input, however overall, the team reported an 80% correct classification

SDSS, Galaxy Zoo and private datasets from Next Generation Virgo (NGVS) and Fornax (NGFS) surveys were used (González, et al., 2018) who used the DARKNET deep learning framework and YOLO (You only look once) for real-time detection to process an image in near real time – less than 3 seconds. One of the benefits of this research was that all data had been uploaded to Github so others could recreate and take advantage of it. This was the first identified paper where this was noted. They used industry standard FITS files with 3 colour channel images for 38,732 galaxies. 5 training sets, each with a different filter to compare against, a maximum accuracy rate of 90.23% was achieved. The design of the CNN was provided, 23 convolution layers and 5 maxpool layers, this helped in the tuning of the CNN used by this project.

2.4.2 Image Recognition and Machine Learning

In 2017 a paper published by Toronto University titled ImageNet Classification with Deep Convolutional Networks (Krizhevsky, Sutskever, and Hinton 2017) changed computer image recognition into high gear. It reduced the error rate in the classification of ImageNet dataset by 50% and proposed that deep Convolutional Neural Networks were the solution to image classification. The architecture presented was called AlexNet, Figure 3, consisted of ReLu layers for non-linearity activation functions, data augmentation, dropout to reduce over-fitting and successive convolutional layers with pooling followed by fully connected layers.

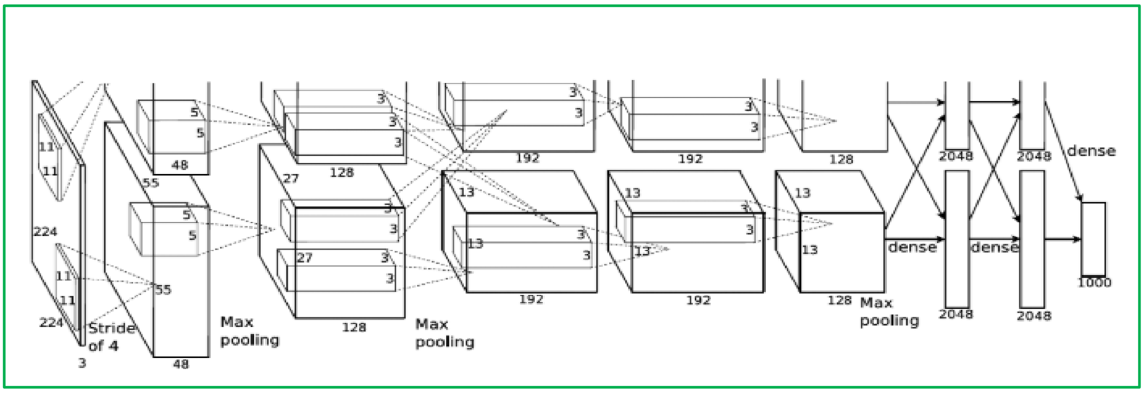


Figure 3 : AlexNet Architecture

A paper presented (Nkwentsha, et al., 2020) at the 2020 International SAUPEC/ RobMech / PRASA Conference in which they outlined how CNN, using weights from InceptionV3, were used as automate the classification of X-ray imaged. Like with astronomy, they had converted the greyscale images into 3 colour channels and achieved an accuracy of 66.7% and were able to up this using padding to 71.34%. Padding is the process where extra pixels are added to images to give the CNN the opportunity to take the first and last columns of images into account when processing in groups of 3 or more pixels when filtering to identify significant attributes of an image. The research used existing database of images to find a similarity to new images and therefore classify the new image. The use of support vector machines as an existing method to classify images was identified into 6 classes as used by (Zare, et al., 2013) and that KNN performed extremely week in x-ray classifications (Pelka O, 2018).

At the 2nd ICETC conference (Barik & Mondal, 2010) proposed using image segmentation to aid in computer image recognition. They used manual methods to segment the images – this is in line with using expert astronomers in this projects case – and used the histogram to find edges of the objects. They processed to use Maximum Ownership Labelling to mark the edges of the objects within the images followed by a Graph based approach to segmentation. The number of images used was small – only 4 are identified as being the training set and the average number of background clutter objects varied from 100 to 1000. They found that the computer vision methods were able to achieve from 79 – 95.77 where more than 100 objects were in the background and humans over 95 %, however there was only 30-50 objects in the background. It appeared that this research needed much more analysis and larger training sets to show realistic results. It is also possible that the method used is not conjunctive to larger training sets as used with CNN's generally.

As the IEEE Canadian Conference on Electrical and Computer Engineering (Azhar, et al., 2016) showed how Naïve Bays could be used to classify potholes with a 90% accuracy. As before the images were grayscale and 200x200 in size, which were divided into 8x8 cells and further into 4x4 blocks for processing. Training was completed using 50 images with 70 for testing. The image count is low as this suites Naïve Bays, however, would not necessarily scale to the volumes and subtility of imagery classification for astronomical imagery where more factors would need to be considered to classify an object.

Computer vision was looked at from an evolutionary perspective and a comparison with human vision by (Li & Shi, 2018). They listed out 4 steps for pattern recognition systems - 1 data collection, 2 Pre-treatment, 3 Feature Selection and extraction, 4 Classification and decision. This process follows the KDD methodology in how to move through the phased to a large extent. The over findings of this research do not articulate findings in relation to the topic in a clear manner. The research does find a match in how vision evolved for humans and computers – to some degree – however the conclusions are unclear at best and difficult to relate to the topic.

In recent years TensorFlow tens (Abadi et al., 2016) has been used, as a common library within Python and other platforms, to process deep learning and image recognition. (Mattmann & Zhang, 2019) proposed how it could be used for data classification. In the research TensorFlow was used on 2,620,000 celebrity images which were 244.244 in size. This was to recreate the work completed using the IMDB database where only 2,622 celebrity images were used , along with super computers to achieve a 97% accuracy rate, however it also loaded the MatLab's ConvNet (Vedaldi and Lenc 2016) weights and did not follow a full TensorFlow training pattern. The key points taken form the research was that TensorFlow was a good aid in the creation of CNN's and allowed easier paths to splitting the training and testing data over other methods. It also highlighted that there were tuning options within this toolset which were of advantage.

2.5 Conclusion and Identified Gaps

The amount of data which is and will be generated by telescopes is growing faster that humans can process using traditional manual ways. Traditional methods of catalogue objects, like decision trees, have a place, but will not be able to keep up with the volumns of data or additional complexity as new data becomes available with every new itteration of deep space survey technology. While the use of neural networks is nothing new, be it that is is a relativly recent addition to the toolkits available to astronamers, the complexity of building models and categorised data required to train a model is only recently become available digitally. The availability of newer libraries for computer vision and image recognition is making the area of nearual networks a more enticing prospect to process large datasets with high degree on accuracy.

TensorFlow was developed by the Google Brain team (Abadi et al., 2016) as an open-source machine learning library to assist developers in the building and deployment of AI models covering image recognition, machine vision and natural language processing. Version 2.1 was released in January 2020 which added improved support for GPU's and improvements, bug fixes in the design. It is being worked on and improved upon currently with nightly updates being made available. TensorFlow also supports Keras, another open-source library for neural networks written to support Python, which is designed to support fast experimentation of deep neural networks by making adding layers to the network a simple process. Since TensorFlow was a purposely designed ANN API which has been written to deal with image and computer vision recognition, also since the version of TensorFlow upgraded in 2020, a clear opportunity was identified to utilise this new version and the newest SDSS catalogue to build a model to classify astronomical imagery.

3 Research Methodology, Exploratory Analysis and Data Preparation

The purpose of research is defined as to "*increase the knowledge of humans*" (Clark-Carter, 2010), or the "*objective of research is to find out questions through the application of a systematic and scientific way*" (Bhattacharyya, 2006). By following a structured methodology, researchers show that they have "*followed similar standards to scholarly journals*" (Murray, 2006) thus ensuring that a rigerous methodology and criteria were implemented to support the research findings.

Several data analysis methodologies were reviewed for this project as outlined in (Shafique and Qaiser 2014), including KDD (Usama, et al., 1996), CRISP-DM and SEMMA (Azevedo, 2008). After consideration it was decided to follow the KDD methodology as the candidate was familiar with it. Also, KDD focuses on the extraction of knowledge from data in the context of large databases, which is shown in this paper to be appropriate to the problems being addressed. KDD was modified to deep space

3.1 Deep Space Methodology

Following the Deep Space Methodology, the project completed the research in a scientific manner. Each of the subsections below discuss how the project implemented each of the steps within this methodology, as described in described in Figure 4 and how each was adapted to this projects research needs. The initial step was to gain an understanding of the domain in question, evaluate prior knowledge and identify the goals of the end-user. This is covered in Chapter 2 where prior research was critically evaluated and the end-users goal identified in the research question.

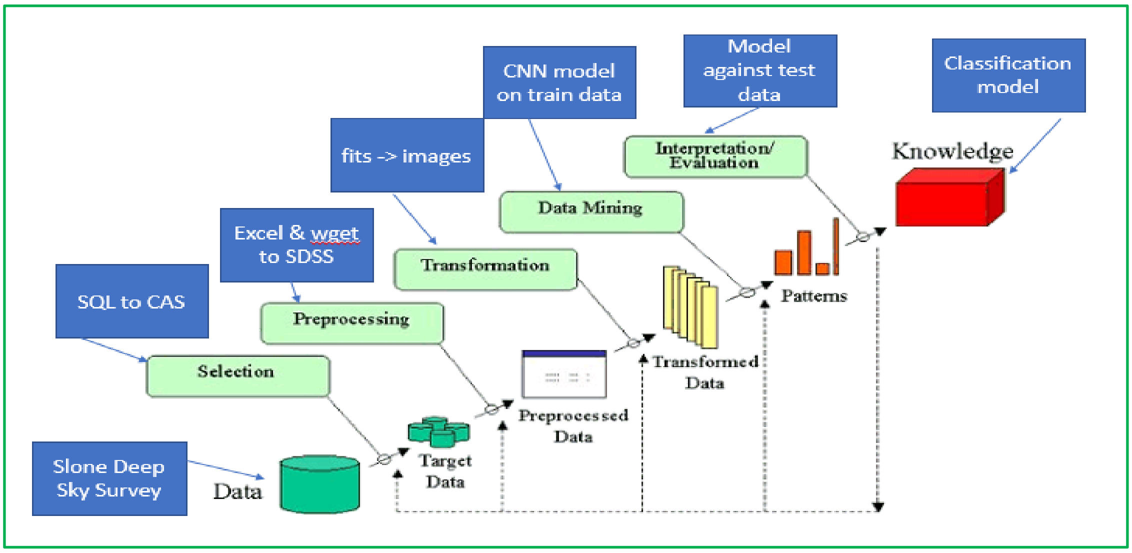


Figure 4 : Modified KDD Process for Deep Space Object Classification

3.1.1 Data Selection

The common format for astronomical files to be saved in is called a FITS format. FITS is an acronym for Flexible Image Transport System and usually contains both data and image data.

Standard images are taken in 3 colours – Red Green Blue, however, astronomical imagery is taken in greyscale in with differing filters in place to pick up different spectrum which can later be combined to make the full spectrum image of the object. The SDSS camera, see Figure 5, takes images in 5 filters which the camera takes, u (ultraviolet), g (green), r (red), i and z (both infrared).

Figure 5 : Image if the SDSS Camera with filters[1]

In this case a CAS job was run on the SDSS Catalogue Archive Server, using an SQL online query. The results provided a list of objects along with their Right Ascension (RA) and the Declination (DEC) along with the class of the object, being either a star, galaxy or a quasar and other details of the imagery in the archive which provided the basis for all further data selection for this work.

3.1.2 Data Pre-Processing and Exploratory Analysis

The objects were also filtered into the 3 classifications using excel and the same number of each classification was queried from SDSS. Data selection was all based on the results of the CAS output. For data pulled from the SDSS severs, the first 25,000 returned objects were used from each category. For the fits files this was 5 filters to make up a single full image resulting on over 130k images. Splitting of images into training and testing datasets was completed randomly, for the FITS images, this meant that a contiguous block from each band was selected in order to ensure that all filters could be used for an image.

Through an API call to the SD16 JPEG web service it was possible to extract jpegs which were "cut-outs" of larger FITS files and download them locally. These images were centred on the object of interest. Again 25,000 images for all 3 categories were extracted. For images extracted from the Kepler repository on Space Telescope Science Institute (STScI) servers, jpeg images were downloaded directly also based on the RA, DEC and CLASS from the SDSS's CAS

3.1.3 Data Transformation

The fits files were opened in Python and the images for each filter along with the object classification were extracted and saved as jpg files for later processing. SDSS jpeg files in groups ordered by classification, the array was shuffled prior to being split into an 80/20 for training and testing. required the addition of a ".jpg" suffix to their names to allow the windows PC to recognise them as image files. TensorFlow processing software only support a small number of image types, of which both PNG and JPEG are supported. The STScI jpeg images required no additional processing prior to loading into the models. As the data was initially downloaded in

[1] airandspace.si.edu

groups ordered by classification, the array was shuffled prior to being split into an 80/20 for training and testing.

3.2 Data Mining and Learning Approach

Data mining is the process used to extract usable data from a larger set of raw data. In this case the process used was to identify the desired objects by classification from available catalogue files and to extract this data into usable formats which were later fed into the models. If generally covers the analysis of the data to identify the required patterns, the collection of this data and sometimes the processing of the required data into data warehousing applications.

In this project Convolutional Neural Networks were used to mine the data and to extract the features which allowed the models classify dep space objects. The models used were MobileNet, ResNet50, VGG16 and Xception networks. The learning approach taken was to load the trained weights for the models which had been trained on the ImageNet dataset, to remove the final layer and add new layers to allow the models adapt to the new classes of images.

3.3 Data Interpretation and Evaluation

The images were structured into subdirectories locally by source, into training / validation and by classification. This allowed the TensorFlow function to "crawl" the directory structure and identify classification based on directory names. This was only possible using new functions which were in pre-release from TensorFlow master bench on GitHub.

For the purposes of this project it was decided that accuracy of the trained model would be tested against the validation dataset, which was the remaining 20% of the overall data. The accuracy of the model's performance on new data would provide the measure against which the model was evaluated.

The goal of this project was to evaluate, in a scientific method, that computer vision and classification models could be used to assist in the identification and classification of deep space imagery. A proven and well accepted methodology was laid out and followed in the pursuit of this goal and the results are detailed below in Chapter 4 along with design specifications. Full tracking of the progress of the models and the results were included in the associated Configuration Manel, which was submitted along with this pager and other supporting documentation.

4 Design Process Flow

Computer vision has evolved rapidly over the past two decades however the ability to scale well due to complicated architecture and the number of connected weights required (Sun, et al., 2020) has been an issues outside a cloud and mainframe environments. With the advent of deep learning and convolutional neural networks this has changed. Their research covered a review of the performance of 22 models widely used in image classification and found that convolutional neural networks performed best, for image classification. As a result, the research here focused on modifying existing trained models and adopted these models to answer the research question proposed. The architecture of the solution is described in Figure 6 below.

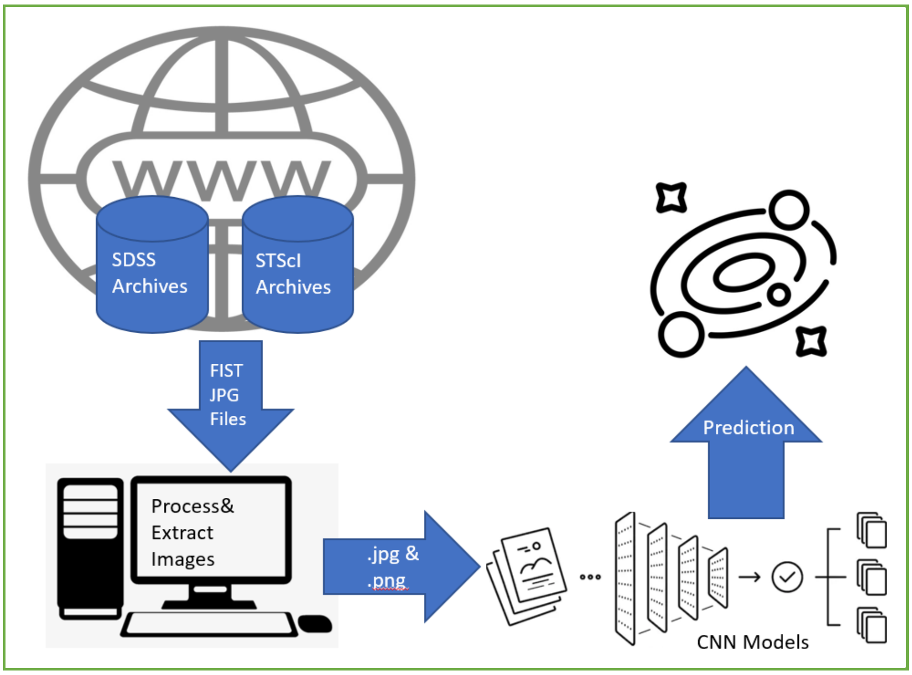


Figure 6 : Design Process Flow

4.1 Convolutional Neural Networks

Convolutional Neural Networks are primarily used in image classification. They work by taking input, implementing filters which extract features, see Figure 7 on the convolutional layers and using dense fully connected layers "learn" through the adjustment of weights and biases of nodes as both forward and back propagations are performed. The output layer then provided a classification, sometimes the softmax() function is used to force the output to pick the "most likely" classification.

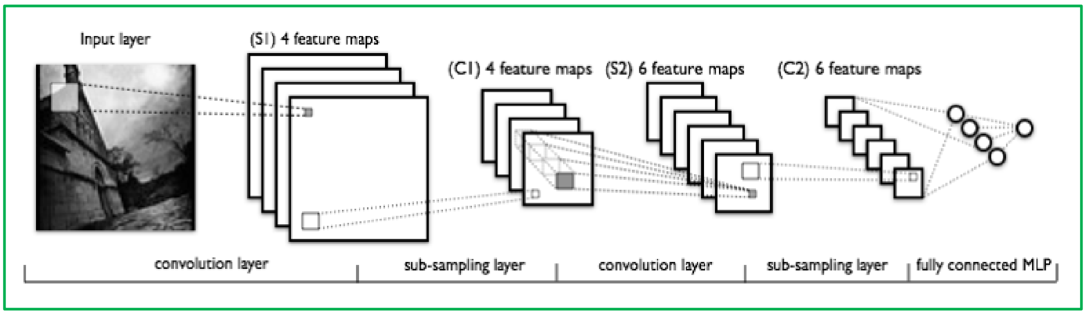


Figure 7 : Generalized structure of a CNN

4.2 Keras and TensorFlow

TensorFlow is an opensource free software platform, developed by Google's Brain Team, for machine learning. It has a community of developers and resources, both from the TensorFlow community and Google, which can make deploying a machine learning tool possible for most users who do not have a deep understanding of the mathematics behind the workings of a ANN. Its framework provided levels of abstraction to allow users focus on their implementation over the coding behind the functions. It offers API's for many different programming languages, e.g. Python in this case.

Keras is an open source neural network library written in Python which runs on top of TensorFlow. It "*has the low-level flexibility to implement arbitrary research ideas while offering optional high-level convenience features to speed up experimentation cycles* [2]."

4.3 Design for Data Acquisition

The design for the solution was in two parts, the initial stages are described below, Figure 8. Tasks completed on the internet are highlighted in green, while tasks completed locally are highlighted in blue and detail the flow of the process from start to acquiring the "fits" files. Initial stage was to query the catalogue servers of SDSS via an on-line SQL interface which returned a csv file giving details of the deep space objects in its archives. This file was then taken into excel to build up the required URL's of the FITS files. From initial query to final query the data re-run had changed, so an update to the concatenation command in excel was required.

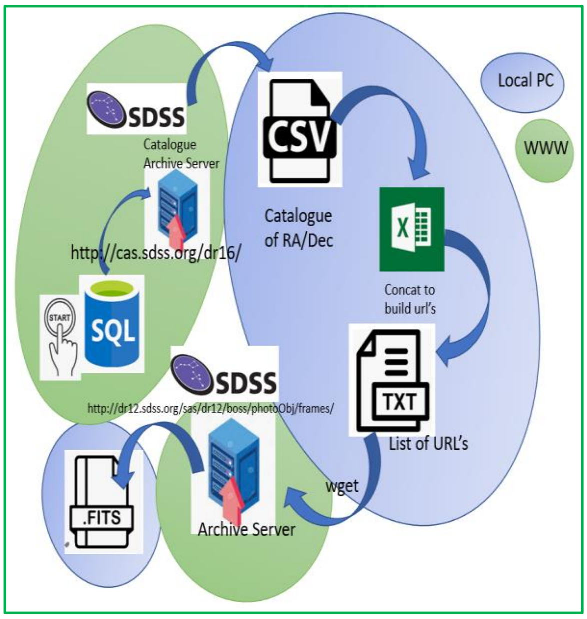


Figure 8 - Steps in preparing and acquiring FITS files form SDS

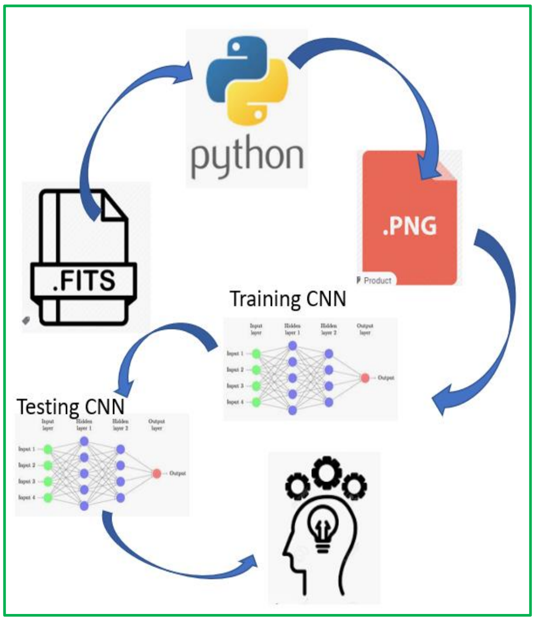


Figure 9 - Steps in processing the data to completion following data acquisition

4.4 Design for Processing and Modelling of Data

Fits files were processed using Python script to extract the plot images and to save them as PNG files, one per filter. These are later, again using python scripts, combined into 3 filter RGB files to provide a second type of input, see Figure 9. SDSS jpg files were directly extracted from the SDSS DR16 data servers and saved locally. Kepler jpg images were extracted using a modified script provided as an API to that sites data servers. All models were run using Python and imported libraries are described in the Configuration Manual

4.5 Data Presentation

Once the models had completed, their results were output to screen as both graphs and numeric output. The model's performance was measured for accuracy during training as well as

[2] https://keras.io/

validation, also measured was the model's loss function for both training and validation. This showed if a model had become "stuck" and ceased to learn during the process.

5 Implementation of Deep Space Classification Models

The following section outlined how the KDD methodology was modified and implemented in this project.

5.1 Data Source Selection

An SQL query was run on the SDSS Catalogue Archive Server, CAS, which returned output in csv format. This listed all fields which were later used to identify the required FITS files, the JPEG from both SDSS and STScI servers. A sample of the output is included in the Configuration Manual along with the methods used to combine the results of the CAS query into URL's which allowed image selection and download.

5.2 Data Pre-Processing

Using the output from, 5.1 above, the excel file was filtered by classification. Using the concatenation function within excel the URL for the files were generated to extract the required images and fits files. The full catalogue, of over 11 million objects, was deemed to be too large to work with on a home PC. Table 4 lists the number of files by type and source used in this research.

Table 4 : File volumns by Institute download

Institute	File Format	Star	Quasar	Galaxy	Total
SDSS	FITS Filters	44,055	44,177	44.020	132,252
	JPEG	25,000	25,000	25,000	75,000
	RGB	10,454	9,011	8,447	27,912
STScI	JPEG	25,000	25,000	25,000	75,000

Using the "wget -i download.txt", where the URL of the requested image was a line in the test tile, format the required FITS images were downloaded from SDSS. All image files were given a suffix of the first letter of their classification to allow for ease of identification later. Once the fits files were downloaded then the URL's were stripped off the path and only the file remained, this was done to make them easier to handle in the future.

This process was repeated with a different format used to make the URL combination for the jpg files from the same site, but this time it was the DR16 servers. The process to extract the Kepler images required a text file which was used as input to a python script made available by the STScI support teams to allow direct download of jpg images from their site.

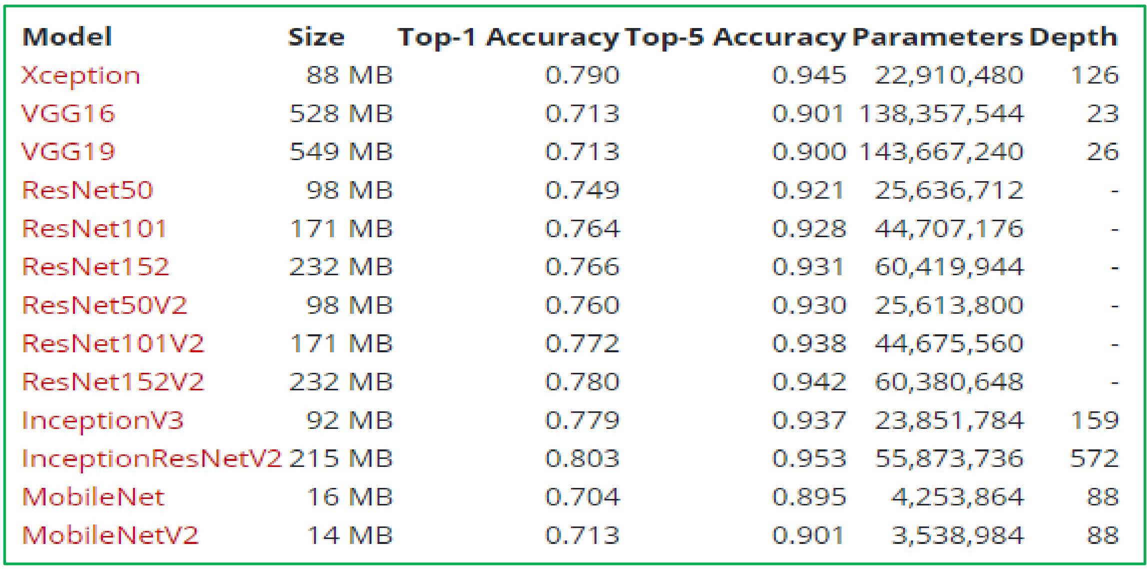
5.3 Data Transformation

A Python script was written to extract image plots from the FITS files and these were placed in folders corresponding to their classification. Another Python script was written to combine 3 of the fits filters plots into a single RGB image. This was completed by identifying a "g" frame and from that merging the red and blue filters to produce a 3 filter RGB image in PNG format. The Kepler files required a renaming to ".jpg" for TensorFlow to accept them as image files. This was required as the files downloaded without extensions. A Python script was created to rename all files by adding the extension to them and making them usable for future work. No other transformation was carried out of the jpg images, as they were already in the required formats. In order to prepare the data to be used as input to each data set was split into a test and train folder structure which was subsequently subdivided. This resulted in 4 top level folders, one for each dataset, and each with 2 subfolders for Test and Train data within which each had a folder for each classification. Images were split 80% for Training and 20% for Testing. Details on folder structure are included in the Configuration Manual.

5.4 Data Mining

CNN models mine data through a process of feature extraction followed, generally, a flattening layer then a varying number of Rectified Linear Unit (ReLU) layers and finally a softmax layer. The exact number of each of the Convolutional and ReLu layers is more an art than a scientific process (Ma, Dang, and Li 2014). The conclusions for Ma's research was that the number of hidden layers used for image recognition tended to be a more trial and error approach than a hard-scientific approach. However, in their book Deep Learning, Adaptive Computation and Machine Learning series, (Goodfellow, et al., 2016) identify that "*Empirically, greater depth does seem to result in better generalization for a wide variety of tasks*". There was also the concern that too deep a network could lead to stalled learning and unduly lengthy training. There are several deep learning models which are packaged with Keras as standard. They all provided model performance against the ImageNet validation dataset (Table 5) with the top-1 and top-5 accuracy refers to the model's performance against that dataset.

Table 5 - Keras Applications Performance against ImagNet



Model	Size	Top-1 Accuracy	Top-5 Accuracy	Parameters	Depth
Xception	88 MB	0.790	0.945	22,910,480	126
VGG16	528 MB	0.713	0.901	138,357,544	23
VGG19	549 MB	0.713	0.900	143,667,240	26
ResNet50	98 MB	0.749	0.921	25,636,712	-
ResNet101	171 MB	0.764	0.928	44,707,176	-
ResNet152	232 MB	0.766	0.931	60,419,944	-
ResNet50V2	98 MB	0.760	0.930	25,613,800	-
ResNet101V2	171 MB	0.772	0.938	44,675,560	-
ResNet152V2	232 MB	0.780	0.942	60,380,648	-
InceptionV3	92 MB	0.779	0.937	23,851,784	159
InceptionResNetV2	215 MB	0.803	0.953	55,873,736	572
MobileNet	16 MB	0.704	0.895	4,253,864	88
MobileNetV2	14 MB	0.713	0.901	3,538,984	88

[3]

Using some of the above models as a basis the top layer, which gave the classification if used against the ImageNet dataset, was removed, the weights for ImageNet were imported and the models were frozen, which meant that the initial models parameters were untrainable. This gave a model which already was trained to extract filters from images. This model was then extended to have additional layers concatenated to it which allowed the model to adapt to the new images and refine its "learning" to cover deep space objects.

[3] https://keras.io/api/applications/

6 Evaluation, Results and Discussion

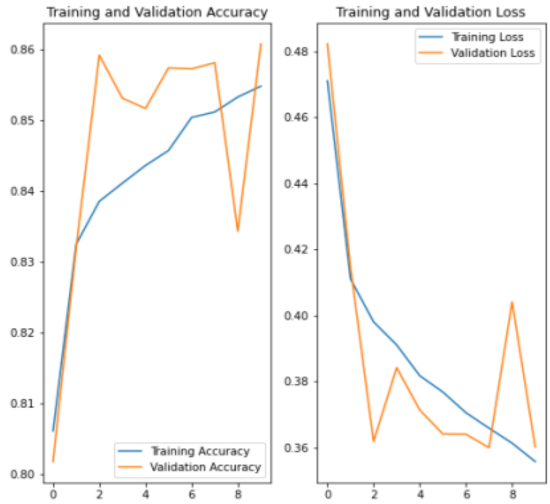
This section covers the performance of all models which were evaluated, reviews their performance against the imagery from SDSS and STScI and compares their performance to ImageNet performance, which is considered as “state of the art” baseline by which CNN’s model performance are assessed.

6.1 MobileNet Model

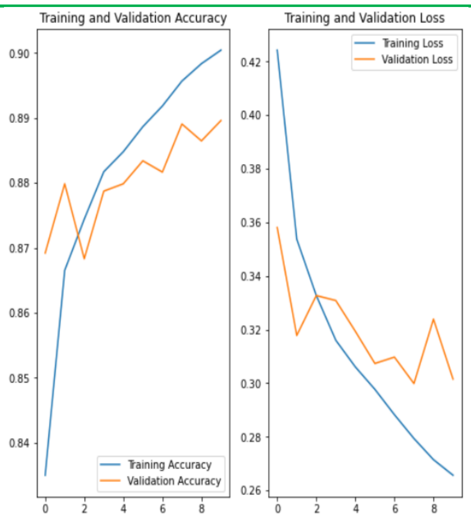
MobileNet (Howard, et al., 2017) model developed at Google and designed for embedded application on mobile devices. Its architecture was developed in a streamlined fashion to use dept-wise separable convolutions when building a light weight deep neural learning network. As was shown in Table **5** it is relatively small compared to the other models used in this project. This was the smallest and quickest to train of the models, with only 5.8M parameters. It performed comparatively well to all other models against the STScI jpg images in both accuracy and validation. The ease to work with this model and its small size make it an ideal model for standard home computers to utilise, and this is further enhanced using later versions of TensorFlow which support GPU processing.

This model reached the best accuracy score at 90% after 10 epochs on the new jpeg images from SDSS. Figure 10 shows that the training dataset/s learning curve was on an upward trajectory still and that a higher accuracy was feasible. There was also a sharp decent in the loss trajectory which indicated that the rate of learning was slowing, however it had not bottomed out. The SDSS jpeg images results reached 85% given the same number of epochs. However, it would have taken many more epochs to bring the SDSS filter images to the same level of accuracy which could only have been achieved with many more images to compensate for potential overfitting. RGB files did produce better results than the filters, however not significant enough to use as they only returned accuracy of 50%.

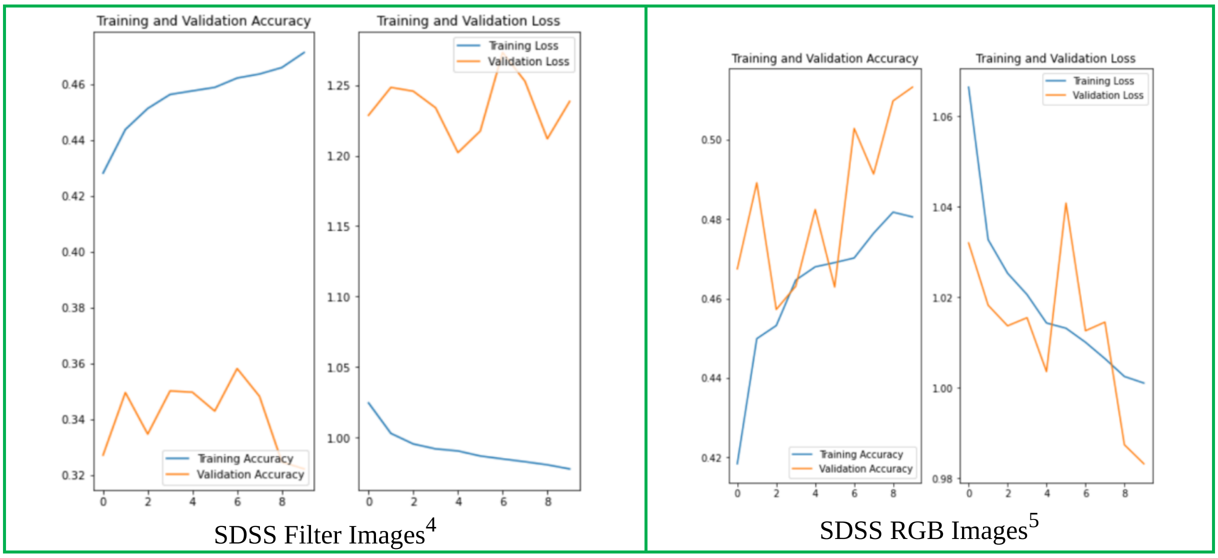
The results are on par with the results from this models training using ImageNet, see Table 5 Table **5** - Keras Applications Performance against ImagNetand the results of the Galaxy Zoo results of 90.23% as identified in Section Image Recognition and Machine Learning.



STSci JPG Images



SDSS JPG Images



SDSS Filter Images[4]

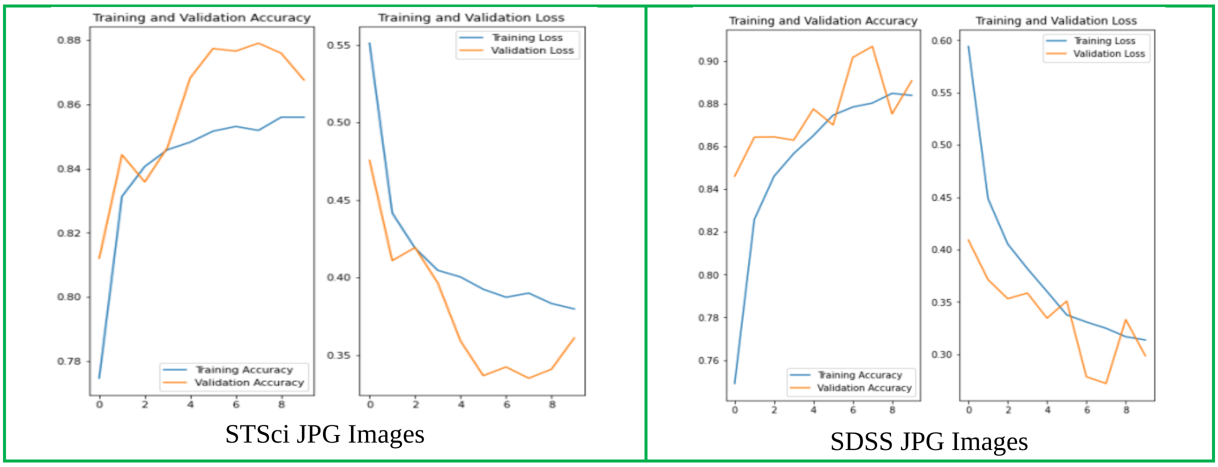
SDSS RGB Images[5]

Figure 10 - Performance of the MobileNet models

This addressed Objective 4b of the projects goals and completed the action giving the required results.

6.2 The ResNet50 Model

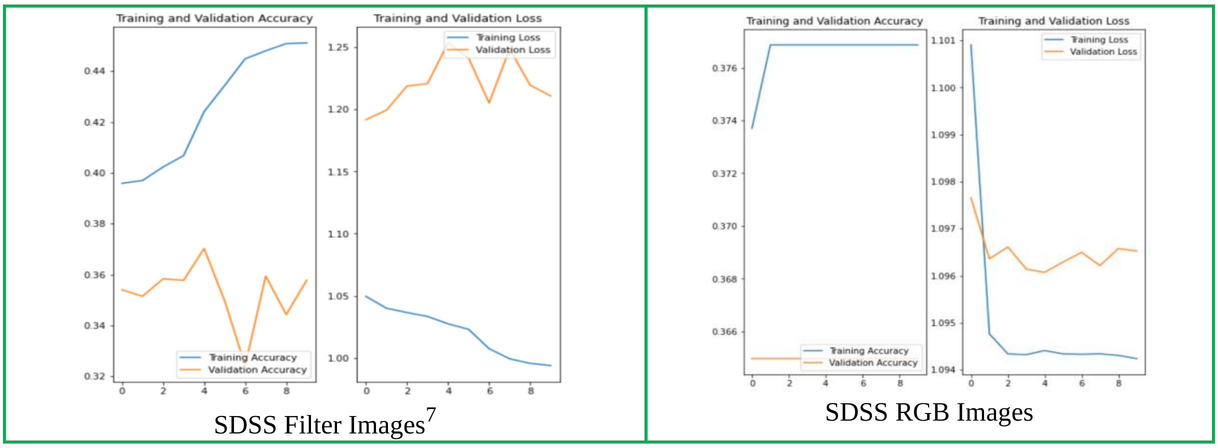
The ResNet50[6] , (He, et al., 2016) which has layers divided into blocks, and over 23 million trainable parameters. ResNet50 is credited with overcoming some of the issues identified with very deep learning, that of degradation problem and vanishing/exploding gradient problem. The model also performed on-par with other models when processing jpg files. While it had an additional 1,000,000 parameters which were tuneable as part of the training, this did not make a significant improvement in the results. Figure 11 shows the performance of the model against images from both sources and for both jpeg and filter datasets. All models showed signs of plateauing at 85% - 89% on jpeg images. The SDSS Filter images also plateaued 45%. While this model did perform best for validation accuracy, at 89%, it did not show the same potential as MobileNet to continue to refine its learning and improve its accuracy. Again, the RGB images only performed at a 50% accuracy, below the jpg formatted files.



STSci JPG Images

SDSS JPG Images

[4] Note different scale for this plot
[5] Note different scale for thi splot
[6] http://ethereon.github.io/netscope/#/gist/db945b393d40bfa26006



SDSS Filter Images[7]

SDSS RGB Images

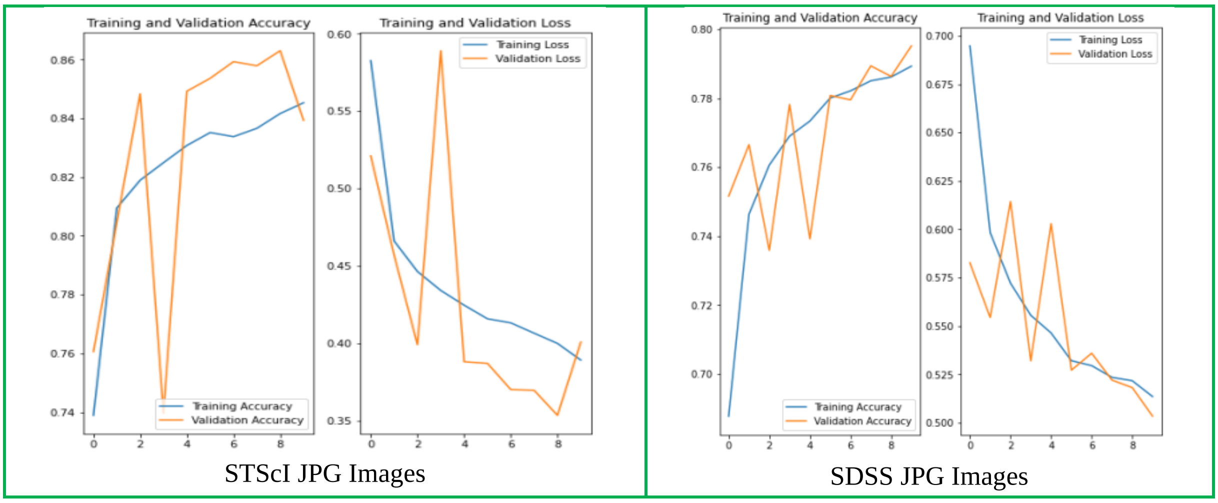
Figure 11 - Performance of the ResNet50 models

The model stalled on the second epoch using the RGB images and made no progress in the accuracy of the training or validation output. Therefore, it was shown that these images were not suitable to use for deep learning using this model. This addressed Objective 4c of the projects goals and completed the action giving the required results.

6.3 VGG16 Model

This model was developed as a solution to the ImageNet challenge (Simonyan & Zisserman, 2014) to research how CNN's accuracy on large image datasets. Their research looked at how the increase of the weighted layers, using small convolutional filters (3x3) could improve the accuracy of the CNN. Their models also generalised well with other datasets. Results in Figure 12 show the performance of the VGG models on the datasets.

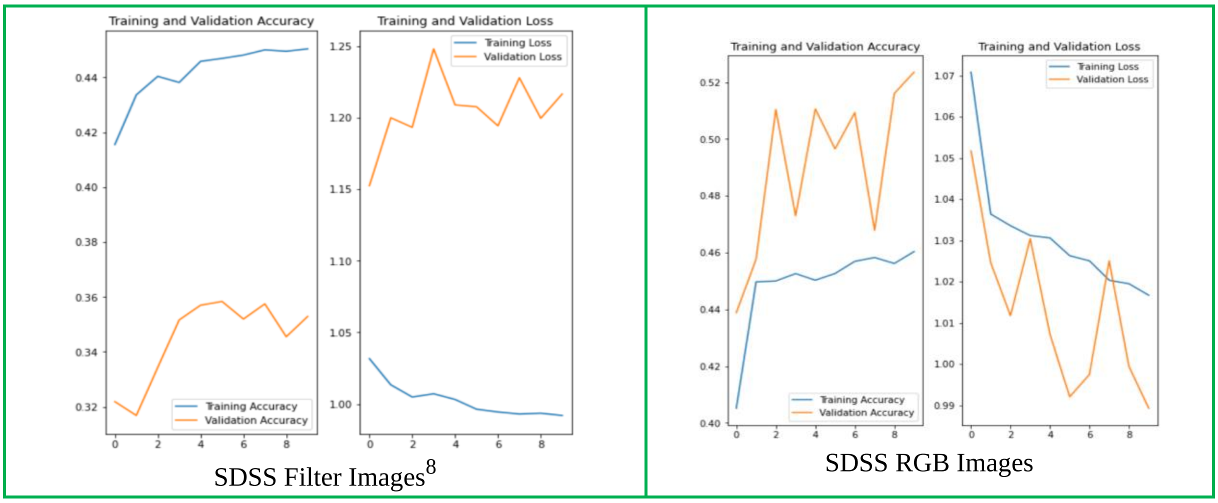
The models performed on par with the results from the ImageNet baseline, Table 5, on the new images presented and even showed potential for further refinement with more epochs as the loss rate was on a good downwards trajectory and the accuracy was improving. There was a loss in validation accuracy towards the end of the 10 epochs which could be investigated in future research. Based on the results provided the RGB files are better then the filter images for classification, but again did not meet the same bar as the jpg files.



STScI JPG Images

SDSS JPG Images

[7] Note different scale for this plot



SDSS Filter Images[8]

SDSS RGB Images

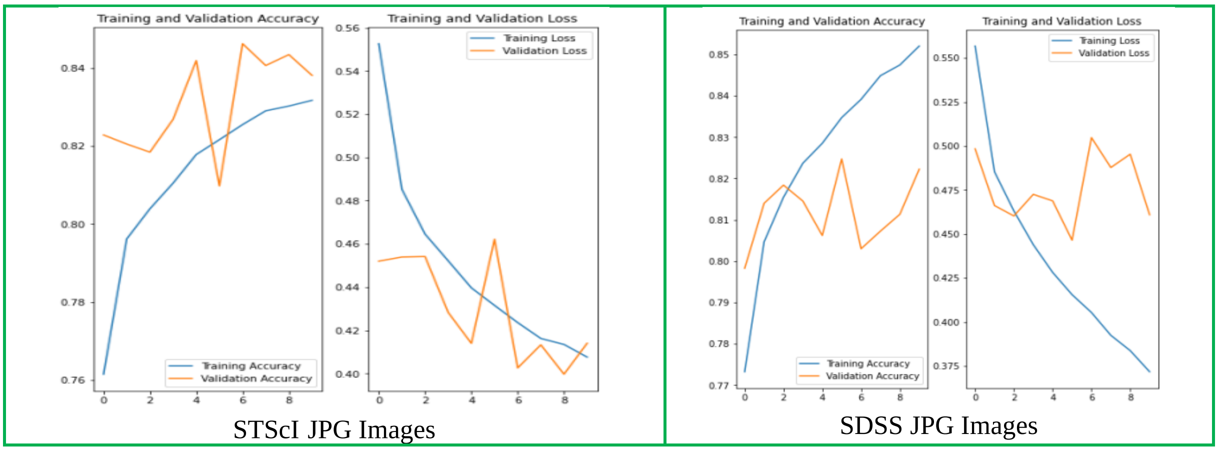
Figure 12 - Performance of the VGG16 models

This addressed Objective 4d of the projects goals and completed the action giving the required results.

6.4 Xception Model

Developed by (Chollet, 2016) as an improvement of the Inception V3 model. The new model outperformed IncpetionV3 on a classification dataset of 350 million images and while it used the same number of parameters this was accounted for by the greater efficient use of the model's parameters. The model was described as "*depthwise separable convolution can be understood as an Inception module with a maximally large number of towers*".

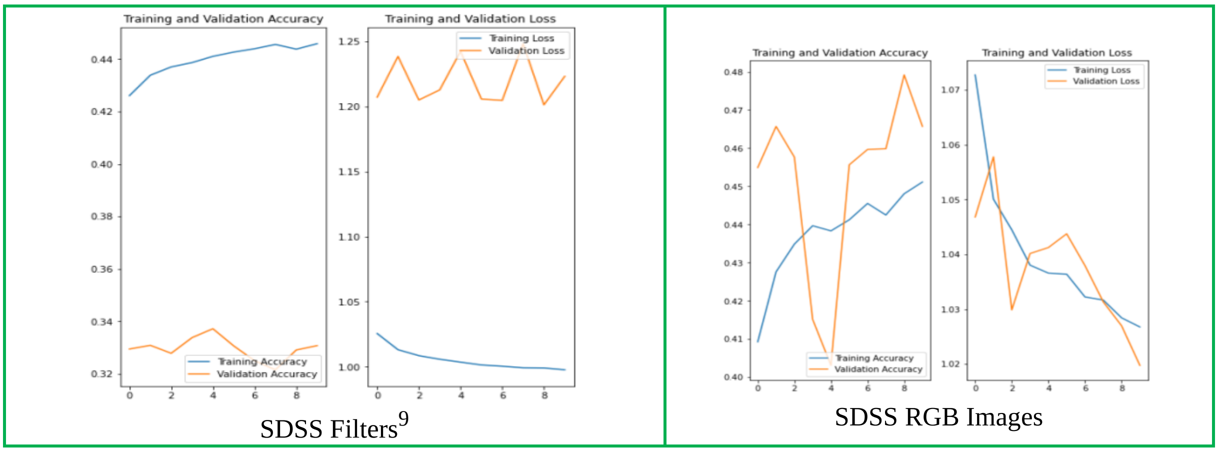
Figure 13 shows the performance of this model against the 3 different datasets. It reached 83% and 85% accuracy in classifications for jpg images and only 45% for the filters, however the processing on the jpg images showed a good upwards trend on the accuracy and the model has not reached a 0 loss, therefore there was further potential to improve this result with further training identified. The model shows potential to continue to learn with more epochs and is a candidate for further research and achieve greater accuracy. While the RGB images did not provide the performance of the jpg images, they did provide a slow upwards trend in accuracy and the potential for greater accuracy with more images and epochs being used.



STScI JPG Images

SDSS JPG Images

[8] Note different scale for this plot



SDSS Filters[9]

SDSS RGB Images

Figure 13 - Performance of the Xception models

This addressed Objective 4e of the projects goals and completed the action giving the required results.

6.5 General Review of Models Performance

All models had the same changes made to them, as in the final output layer which would have been used to show the results of the classification with the ImageNet images was removed and the same number of additional layers were inserted. This allows for a like-for-like comparison on the models' performance. With this training, size does matter when it came to processing time. On a standard CPU models took up to 5 days to train with little more than 35-40% accuracy. The need for large volumes of images when training models from scratch was evident.

Models performance on jpg images is on par with the results of the ImageNet league table. MobileNet and ResNet50 performed the best out of the models used, though Xception seemed to show better potential to improve accuracy over ResNet50. What the research did show is that the models trained on the ImageNet dataset can generalise extremely well and can be easily adapted to new problems and new sources of images. The performance of all models showed that Keras and TensorFlow have great potential to assist in image classification, which has raised ethical concerns in the adaption of this technology in the use of facial recognition. While it was not a concern for this research, the ability to train CNN's on images to make classifications has and will continue to be an ethical question. Nvidia already as developers to agree with its ethical guidelines when using their products – NVIDIA CUDA® and "NVIDIA CUDA® Deep Neural Network library" (CuDNN) both of which are required to allow for GPU processing with TensorFlow on a Windows machine.

Each model was trained for 10 epochs, this was to give all models the same baseline performance. While it was identified that further training would have improved the models performance, there was also the risk that further training with the same images could have led to overfitting. While the standard process to avoid this is to augment the data by rotating it, it was felt that to do so for these images would not create sufficient variety as to make this a valuable exercise. Stars and quasars are already round objects and rotating them would have added little to no value to the learning process. Galaxies already were rotated in all planes therefore to do so again was deemed unnecessary. Perhaps with a larger dataset this might have added value, or with wider fields of view where more objects were in the image. However, the

[9] Note different scale for this plot

addition of additional objects could have impacted the accuracy as was seen in the processing of the filter images.

6.6 Comparison of Developed Models

The models used for this experiment were previously trained with the ImageNet dataset. This is a dataset of 14 million images and just over 21,000 classes. The datasets used in this research is in 3 classes and training/validation loads of between 15,000 and 75,000 were presented to the models. All pre-trained models were loaded "as is" without the weights to support ImageNet classifications.

The results of each models training and validation with the inclusion of an additional 5 layers is laid out in Table 6. The top results are highlighted in bold.

Table 6 : Model Paranaters and Results of Training

Model	**MobileNet**	**ResNet50**	**VGC16**	**Xception**
Layers	85 + 5 added	173+ 5 added	18+ 5 added	131+ 5 added
Trainable params	2,625,539	3,674,115	2,101,251	3,674,115
Non-trainable params	3,228,864	23,587,712	14,714,688	20,861,480
Total params	5,854,403	27,261,827	16,815,939	24,535,595
Models Performance after 10 Epochs with the addition of 5 new layers				
STScI - jpg				
Accuracy	0.8548	0.8559	0.8453	0.8316
Val accuracy	0.8608	0.8675	0.8393	0.8380
SDSS – jpg				
Accuracy	**0.9004**	0.8838	0.7893	0.8520
Val accuracy	0.8896	**0.8906**	0.7951	0.8222
SDSS - filters				
Accuracy	0.4712	0.4511	0.4503	0.4458
Val accuracy	0.3223	0.3579	0.3529	0.3307
SDSS - RGB				
Accuracy	0.4806	0.3769	0.4604	0.4511
Val accuracy	0.5133	0.3650	0.5235	0.4657

All model was loaded with the pre-trained weights from the ImageNet dataset training. This meant that the model already "knew" how to extract features from images to assist in classification.

MobileNet, as the best model after 10 epochs, was trained for 50 Epochs, see Table 7, to validate that additional training would provide an improved accuracy in object identification and this was proven to correct.

Table 7 : Results after 50 Epochs for ModileNet and jpg files

Data Source/type	Training Accuracy	Validation Accuracy
SDSS Jpeg	96.30%	87.07%
STScI Jpeg	89.03%	86.05%

These results are comparable to the results from ImageNet's training as in Table 5. In both cases around 20 epochs seemed to be the ideal number as while training accuracy continued to improve, validation accuracy on unseen data did not improve with more training. The performance of all models against what is considered industry benchmark is detailed below in Table 8 which completed objective 6 as described in Section 1.2 of the project report

Table 8 : Models Performance against State of Art Models

Model	Best performance	Image Type	Performance against ImageNet
MobileNet	0.9004	SDSS jpg	0.901
ResNet50	0.8906	SDSS jPg	0.921
VGC16	0.4712	SDSS filters	.0901
Xception	0.8520	SDSS jpg	0.945

All the objectives (chapter 1, section 1.2) have been implemented and the results presented has solved the research question (Chapter 1, section 1,2) as proposed.

7 Conclusion and Future Work

The goal of this project was to investigate the ability of a ANN to utilise the existing imagery and to show if it could be trained to provide a meaningful addition to the astronomers toolset in the identification and classification of deep space objects.

Using development versions of TensorFlow made it possible to move processing to the GPU and this did cut training times down. Leveraging the existing models with their already pre-trained filters made processing on a home PC possible. All models were trained for 10 epochs and a standard baseline for the models' performance was established. This can be used as the basis for further training and for new objects to be added to the dataset once a catalogue of same is made available.

One of the main limiting factors currently, in the field of ANN's is the requirements for pre-labelled data. The models require large volumes of prelabelled images to train on and new labelled images to validate against. The ability to identify exceptions from the norm would allow ANN's to make a leap forward in image recognition in fields like medicine or even to identify a new deep space object which is not a star, galaxy or quasar.

Based on the models used in this project, the conclusion is that FITS files from SDSS cover too wide an arc of space and that there are too many objects in them for an ANN to properly identify which objects are of importance or not. It may be possible to do so with many more thousands of examples, but given that using 120k files across multiple spectrum gave little better than a one in 3 change of selecting the correct classification out of 3 possible classifications, it would seem that this is not a usable solution. Using TensorFlow to capture the images from a monitor it may be possible to so real-time identification of objects. It would also be possible to expand the research if access to the galaxy morphology which was carried

out by Galaxy Zoo, which was the inspiration for this research in the first place, were made available, which project classified
galaxies based on their morphology.

8 Acknowledgement

I would like to extend my thanks to Doctor Catherine Mulwa for her supervision, and support throughout the research.

I would especially like to acknowledge the support from my family, in particular my wife, who made this journey possible and put up with the numerous evening courses along my educational journey. To my parents who taught me that the pursuit of learning is a lifelong process.

This project would not have been possible the continued work of the team at the Slone Deep Space who have made so much data available. The Space Telescope Science Institute, STScI, which also provided both instructions and the jpg files used in this project https://stsci.edu/. Also, the team at Google for making TensorFlow available as an open-source licensed package.

Funding for SDSS has been provided by the Alfred P. Sloan Foundation, the Participating Institutions, the National Science Foundation, and the U.S. Department of Energy Office of Science.

9 Bibliography

Azhar, K., Murtaza, F., Yousaf , H. & Habib, H. a., 2016. *Computer vision based detection and localization of potholes in asphalt pavement images.* Vancouver, IEEE.

Ball, N. M., Brunner, R. J., Myers, A. D. & Tcheng, D., 2006. Robust Machine Learning Applied to Astronomical Data Sets. I. Star-Galaxy Classification of the Sloan Digital Sky Survey DR3 Using Decision Trees. *The Astrophysical Journal,* 605(1), pp. 497-509.

Barik, D. & Mondal, M., 2010. *Object Identification For Computer Vision using Image Segmentation.* s.l., IEEE Xplore.

Bhattacharyya, D. K., 2006. *Research Methodology.* 2nd ed. New Delhi: Excel Books.

Brownlee, J., 2019. *A Gentle Introduction to the Rectified Linear Unit (ReLU).* [Online] Available at: https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/
[Accessed 06 2020].

Chapman, P. et al., 2000. *CRISP-DM 1.0 Step-by-step data mining guide.* s.l.:SPSS.

Chollet, F., 2016. *Cornell University.* [Online] Available at: https://arxiv.org/abs/1610.02357[Accessed 28 06 2020].

Clark-Carter, D., 2010. *The Complete Student's Comapnion.* 3 ed. Hove, East Susses: Psychology Press.

Diederik P. Kingma, J. B., 2015. *Adam: A Method for Stochastic Optimization.* San Diego, Published as a conference paper at the 3rd International Conference for Learning Representations.

du Buisson, L., Sivanandam, N., Bassett, B. A. & Smith, M., 2015. Machine learning classification of SDSS transient survey images. *Monthly Notices of the Royal Astronomical Society*, 454(2), pp. 2026-2038.

Eisenstein, D. et al., 2011. Sdss-Iii: Massive Spectroscopic Surveys of the Distant Universe, the Milky Way, and Extra-Solar Planetary Systems. *The Astronomical Journal,* 142(3), p. 1.

Flemsteed, J., 1725. *Catalogus Britannicus.* London: s.n.

González, R. E., Muñoz, P. M. & Hernández, C. A., 2018. Galaxy detection and identification using deep learning and data augmentation. *Astronomy and Computing,* Volume 25, pp. 103-109.

Goodfellow, I., Bengio, Y. & Courville, A., 2016. *Back-Propagation and Other Differentiation Algorithms.* s.l.:MIT Press.

Goodfellow, I., Bengio, Y. & Courville, A., 2016. *Deep Learning (Adaptive Computation and Machine Learning series).* Cambridge, Ma: The MIT Press.

He, K., Zhang, X., Ren, S. & Sun, J., 2016. *Deep residual learning for image recognition..* s.l., IEEE.

Howard, A. G. et al., 2017. *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision.* [Online] Available at: https://arxiv.org/abs/1704.04861[Accessed 12 06 2020].

Hubel, D. H. & Wiesel, T. N., 1962. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology,* 160(1), pp. 106-154.

Kheirdastan, S. & Bazarghan, M., 2016. SDSS-DR12 bulk stellar spectral classification: Artificial neural networks approach. *Astrophysics and Space Science,* 361(9), p. 304.

Kingma, D. & Lei Ba, J., 2015. *ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION.* San Diego, International Conference on Learning Representations.

Large Synoptic Survey Telescope, 2020. *Legacy Survey of Space and Time.* [Online]Available at: https://www.lsst.org/[Accessed 03 March 2020].

LeCun, Y. et al., 1989. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation,* 1(4), p. 541–551.

Lintott, C. J. et al., 2008. Galaxy Zoo: morphologies derived from visual inspection of galaxies from the Sloan Digital Sky Survey. *Monthly Notices of the Royal Astronomical Society,* 389(3), pp. 1179-1189.

Li, X. & Shi, Y., 2018. *Computer Vision Imaging Based on Artificial Intelligence.* s.l., IEEE Xplore.

Martín Abadi, P. B. J. C. Z. C. A. D. J. D. M. D. S. G. G. I. M. I. M. K. J. L. R. M. S. M. D. G. M. B. S. P. T. V., 2016. TensorFlow: A System for Large-Scale Machine Learning. *12th {USENIX} Symposium on Operating Systems Design and Implementation*, Nov, pp. 265--283.

Mattmann, C. & Zhang, Z., 2019. *Deep Facial Recognition using Tensorflow.* Denvor, IEEE.

Ma, Y., Dang, J. & Li, W., 2014. *Research on deep neural network's hidden layers in phoneme recognition.* Singapore, IEEE.

Nkwentsha, X., Hounkanrin, A. & Nicolls, F., 2020. *Automatic classification of medical X-ray images with convolutional neural networks.* s.l., IEEE.

Pasquet-Itam, J. & Pasquet, J., 2018. Deep learning approach for classifying, detecting and predicting photometric redshifts of quasars in the Sloan Digital Sky Survey stripe 82. *Astronomy & Astrophysics,* Volume 611, p. A97.

Pelka O, N. F. F. C., 2018. Annotation of enhanced radiographs for medical image retrieval with deep convolutional neural networks. *PLoS One*, 12 11.

Post Grad Programme in Data Analytics, 2019. *Intro to Data Mining - slide 25.* Dublin: National College of Ireland.

Reber, G. & Greenstein, J. L., 1947. Radio-frequency investigations of astronomical interest. *The Observatory,* Volume 67, pp. 15-26.

Ruder, S., 2016. *An overview of gradient descent optimization algorithms,* Galway: Aylien Ltd, Dublin.

Simard, P. Y., Steinkraus, D. & Platt, J. C., 2013. *Best Practices for Convolutional Neural Networks.* Redmond, Microsoft Research.

Simonyan, K. & Zisserman, A., 2014. *Cornell University*. [Online] Available at: https://arxiv.org/abs/1409.1556
[Accessed 26 06 2020].

Sun, Y., Xue, B., Zhang, M. & Yen, G. G., 2020. Evolving Deep Convolutional Neural Networks for Image Classification. *IEEE Transactions*, April, pp. 394-407.

Szegedy, C. et al., 2015. *Going Deeper with Convolutions.* [Online]Available at: https://www.cv-foundation.org/openaccess/content_cvpr_2015/papers/Szegedy_Going_Deeper_With_2015_CVPR_paper.pdf
[Accessed 12 06 2020].

US Department of Energy Office of Science, 2018. *The Dark Energy Spectroscopic Instrument (DESI).* [Online] Available at: https://www.desi.lbl.gov/[Accessed 02 March 2020].

Usama, F., Gregory, P.-S. & Padhraic, S., 1996. The KDD process for extracting useful knowledge from volumes of data. *Communications of the ACM,* Issue 11, pp. 27-34.

Wikepedia, n.d. *Softmax function.* [Online] Available at: https://en.wikipedia.org/wiki/Softmax_function[Accessed 06 2020].

Zare, M. R., Mueen, A., Awedh, M. & Seng, W. C., 2013. Automatic classification of medical X-ray images: hybrid generative-discriminative approach. *The Institution of Engineering and Technology*, 5 July, pp. 523-532.