A novel CNN architecture for the classification of galaxies

MSc Research Project
MSc in Data Analytics

Vibha Vaid
Student ID: x20205635

School of Computing
National College of Ireland

Supervisor: Dr. Giovani Estrada

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Vibha Vaid	
Student ID:	x20205635	
Programme:	MSc in Data Analytics	2021-2022
Module:	MSc Research project	
Supervisor:	Giovani Estrada	
Submission Due Date:	19/09/22	
Project Title:	A novel CNN architecture for classification of galaxies	
Word Count 6999	**Page Count** 25	

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: ………

Date: ………

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):	

A novel CNN for the classification of galaxies

Vibha Vaid
x20205635
MSc in Data Analytics
National College of Ireland

19th September 2022

Abstract

The universe and its galaxies have always been an interesting subject to scientists and astronomers. It becomes essential for an astronomer or scientist to classify a galaxy or a deep space object. In this research, a novel Convolution neural network architecture is being developed to classify the morphology of the galaxies. To perform classification the dataset for this project is collected from Kaggle which was a part of the galaxy zoo project. To develop the novel CNN architecture, literature has been reviewed and important aspects of the paper are extracted. A concatenated CNN has also been developed in order to combine the 2 neural networks and produce better results. The CNN and concatenated CNN are compared with other existing models of transfer learning like vgg19, densenet121, and inceptionv3.The main goal of this research has been to develop a novel convolutional neural network and concatenated CNN which improve the accuracy of the classification of galaxies. The CNN architecture is developed from scratch by concatenating two different networks. The research follows the knowledge discovery in database (KDD) approach for classification. The models are tested on the accuracy, precision, and recall. It can be seen that inceptionv3 gives the most accurate results in comparison to other neural networks used. It was observed that the concatenated model did not converge and performed poorly in comparison to the other networks

Keywords: Image classification, vgg19, densenet121, inceptionv3, CNN, Concatenated CNN

1 Introduction

The universe is made up of billions of galaxies. Space organizations like NASA (National Aeronautics and Space Administration) has developed telescope which allows to study of galaxies beyond the sharp details and also explore the most distant object of the observable universe. A total of 170 billion galaxies have been identified. Galaxies are often categorized based on their visual appearances based on the morphology into shapes like elliptical, spiral, or irregular. It is a presumption that at the center of the galaxy there is a black hole. In the area of data analytics lack of a dataset becomes a major obstacle in solving developing an appropriate architecture. This is the problem that is never faced in the field of astronomy as there is plenty of data present in the field of astronomy from galaxies, stars, and planets. Astroinformatics combines the field of astronomy and data science.

1.1 Motivation and background

Data deluge was a problem faced by the researchers where the information was so high that they were not able to analyze and process much of the data. Kevin Schawinski was the co-founder of galaxy zoo which is the primary dataset that is being used in this research. Galaxy zoo is a crowdsource project that invites people to help in the classification of the galaxies. The results are to determine the different features of the objects and to classify them accurately. The presence of such a large volume of data is the prime motivation for this research project.

Galaxies are not distributed evenly in space. Few of the galaxies do not have closer neighbors while others occur in pairs. It's possible that clusters of galaxies can be grouped into structures called superstructures. The elliptical galaxies are smooth, have featureless light distribution, and look like an ellipse in a photographic image. The lenticular galaxies have a bright central bulge which has a disc-like structure around it. The irregular galaxies do not have a regular structure. The spiral galaxies are flattened discs with stars that form a spiral structure. It is believed that the shapes of the galaxies are caused by the gravitational attraction of the neighboring galaxies

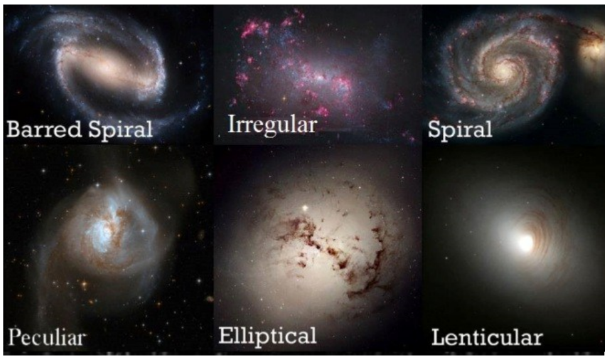

Figure 1: Different shapes of the galaxies

Computer vision is a scientific field that explains how a computer can gain a high-level understanding of digital images and videos. Deep learning algorithm makes it easier to classify the deep space object or galaxy. The accuracy and precision provided by the algorithm is high. It does not require any human intervention. The main aim of this research is to improve upon the existing deep learning architecture and obtain higher accuracy. To develop a novel CNN from the knowledge gained from the literature which was reviewed for this research.

1.2 Research question and objective

Time spent on the telescope can be expensive and it is a finite resource. Weather and atmospheric conditions can further add limitations to detecting the space object or a

galaxy. The problem being addressed here is to can the galaxies be correctly classified into their shapes which in turn will help the astronomers in their research. This research will help astronomers in dealing with the growing volume of data that is being generated daily.
a. "What is the most effective way to combine different convolutional layers to create a galaxy classifier?"
b. "How can different neural networks be combined to classify galaxies accurately?"
c. "What are the similarities and differences between the proposed CNN and VGG19, DenseNet121, and Inceptionv3 networks"?

Objective 1: Review of Classification of galaxies using deep learning techniques, Critically reviewing the techniques to be used in the project

Objective 2:Performing exploratory data analysis and filtering the images from 32 classes to the 3 predominant classes

Objective 3: Build a Deep Learning model using the VGG19 convolutional neural network and evaluate the performance of the system

Objective 4: Build a Deep Learning model using the densenet121 convolutional neural network and evaluate the performance of the system

Objective 5: Build a Deep Learning model using the Inceptionv3 convolutional neural network and evaluate the performance of the system

Objective 6: Build a novel convolutional neural network and evaluate the performance of the system

Objective 7: Build a novel concatenated convolutional neural network and evaluate the performance of the system.

The evaluation metrics for all the models are accuracy, precision, and recall
The roadmap for the remaining components of this research is as follows: Section 2 discusses past research and includes a critical assessment of the various methodologies used in the classification of galaxies using the convolutional neural network and machine learning techniques. A critical review of image classification of transfer learning methods. Section 3 discusses the different steps followed in the methodology of the project and the approach taken. Section 4 discusses the design specification and section 5 different implementation techniques used are discussed. Section 6 discusses the evaluation methods and a detailed discussion on the networks

2 Related work

Every scientific research is always backed by reviewing research papers and reading enough about the topic of discussion. In this section, the papers are reviewed which will help in the design and selection of the appropriate architecture.

2.1 Introduction

Early astronomers have made early observations and predictions across the globe. In the 6th century, certain astronomers and historians stated that the earth was more spherical than just flat. They were also able to accurately calculate the circumference of the earth. In the universe milky way is one of the trillion galaxies. Before the 16th century, the earth was thought to be in the Centre and all the other planets revolving around the earth. It was in the year 1543 that an astronomer named Nicolaus Copernicus proposed with his experiments that the sun was in the center and the planets were revolving around the sun. The morphology of the galaxies is mostly governed by their neighbors and often galaxies collide. The milky way is colliding with its neighboring galaxies.

2.2 Classification of galaxies using Convolutional neural network and machine learning techniques

Convolutional neural network are best suited when it comes to image classification and CNN's form multilayer perceptrons which means they form a fully connected network. In (Fielding et al. 2021) a CNN model is proposed which can be used to classify the galaxies into various shapes like elliptical, spiral, and irregular. Data modeling and image selection are done based on the samples of the data selected. In their model, the images are fed to the CNN model which is used to classify the galaxies. Convolution operations are used with filters to detect edges. The depth of any color image would be 3, which refers to RGB values. From each class the model is fed with 1098 images, batch size of 32 was taken, a total of 30 epochs it was observed that the model achieved an accuracy of 75% and validation

In (Fielding et al. 2021) to train the model to predict the galaxies the galaxy zoo dataset is used and the zoobot python library is used. EfficientNet B0, DenseNet121, and residual neural network 50 as core model architectures are the algorithms which are used. From all the methods which are proposed accuracy metrics are generated per decision tree to find out what is the architecture performance. Among all the methods which are used to produce the result, it was observed that DenseNet121 produced the most accurate results. On the other hand, it was also seen that DenseNet121 took more computational time than ResNet50. A effective comparison between EffectiveNetB0, Densenet121, Resnet50 is done. An artificial neural network resembles the functioning of a human brain where one neuron is connected to other neurons and the signals are passed to the human brain. In (Biswas & Adlak 2018) artificial neural network is used to classify the galaxy morphology into 3 categories. The given algorithm classifies the galaxies into 3 classes which are elliptical, irregular and spiral galaxies. The input which is given to the network can be used to calculate the weighted sum and pass the result forward to produce output close to binary value which can be 0 for the negative result or 1 for the positive result. The error in the system is back propagated to update the weight to get a binary output. Once the weights are found out then the data is tested. In the testing phase, input values are provided to the neural network where weights are assigned to axons. A residual neural network is a neural network that jumps from one layer to another using skip connections or shortcuts.A non-linear function in a contemporary model often contains two or three skips (RELU). The skips are included to minimize the degradation problem where adding more layers to the network results in training errors or to lessen vanishing gradient issues. In (Ba Alawi & Al-Roainy 2021)deep residual network is used to classify the star and galaxies. As Resnet is exceptionally good at image classification. To assess the performance of the proposed model, four versions of resnet models are used. ResNet-18, ResNet34, ResNet50, and ResNet152 are the resnet models used. The accuracy of the models obtained is greater than 90%. Resnet152 performs the best which implies that increasing the residual network improves the accuracy.

To understand the formation of priority is given to the global clusturs. In (Sookmee et al. 2020) five machine learning models were applied to the galaxy dataset which were K-Nearest neighbors, Random Forest, Neural network, SVM and Decision tree. Experiments explained that the KNN, random forest, and SVM were the best approaches for classifying the images of globular clusturs. At the end, when these models are applied to the unseen globular dataset it can be seen that there is a success of 6.32%.

In (Reza 2021) 5 different machine learning techniques are used to sample the dataset into four different categories which are spirals, elliptical, mergers, and stars using the Sloan digital sky survey. It is a challenge when it comes to classifying the mergers as a different categories as they can sometimes be confused as elliptical and spirals. It can be observed that the dataset is highly imbalanced. The 5 different algorithms are decision tree, random forest, K-nearest neighbors, extra trees, and artificial neural network. The highest test accuracy is seen in the ANN (98.2%) and the least accuracy is seen in the decision tree. Also, ensemble methods like ET/RF perform better in categorizing minority class samples than ANN. It is also observed that precision and recall are better for stars than that for mergers.

In machine learning Support vector machine is a supervised learning algorithm that analyzes data for classification and regression. In (Freed & Lee 2013) SVM algorithm is used to classify the galaxy morphologies. In the traditional approach, the SVM algorithm is used in the binary classifier which means it can place objects into one or two categories. This study has used numerous classifiers in a tiered structure to construct a multi-classification. A three-category sample was used in the given structure, and it was initially divided using a conventional binary classifier. When it comes to the second binary classifier it was used to separate the remaining two classes. The results of this paper demonstrate the high accuracy of the classification of the SVM. The results of 2 categories between the spiral and elliptical categories are worth noting. However, it becomes difficult for the SVM to classify the irregular galaxy.

XGBoost implements the grading boosting algorithm under the grading boosting framework. There was a low accuracy of star/galaxy classification which was set by the SDSS, A new machine technique called the gradient boosting XGBoost has been introduced. In (Chao et al. 2019) for bright and dark set a ten fold cross validation method is used. Following this grid search is used to adjust the XGBoost parameters. In the end, based on the accuracy of the classifier, the classification results are analyzed, which are then compared. The accuracy is improved by 10% while using XGBoost.

2.3 Image classification using VGG and Inception v3

The larger convolution kernels in AlexNet are replaced by a series of sequential 3 x 3 convolution kernels in VGG, which increases the depth of the network and validates the performance, which may be continuously improved by depending on the network topology. In (Tang 2021) a new improvised version of vgg16 is proposed, the dataset used is that of cifar10. To expand the dataset, methods such as random flipping, random translating, random erasing, and mix-up are used. The number of completely connected layers is decreased and a BN and dropout layer are introduced in the proposed vgg16 model. The learning rate of the model steadily declines as training time increases, and the model gradually converges. The final accuracy of the model is 96.7 %, which is lower than the baseline model, indicating that the new improved model slows down overfitting.

The visual geometry group at Oxford University created VGG16, where 16 is the number of layers in the neural network. With a 92.7 percent accuracy on the imagenet dataset, VGG16 has demonstrated success. In (Kavala & Pothuraju 2022) the three most prevalent grape leaf illnesses are categorized using vgg models along with healthy grape leaves. The testing data show that the offered models have a mean accuracy of 98%, proving that utilizing a neural network approach to diagnose certain diseases is preferable than doing it manually. As can be seen, the proposed vgg model outperformed the old one in terms of accuracy rate.

In (Mascarenhas & Agarwal 2021) a comparison is made between 3 methods are used for image classification which are vgg16, vgg19, and resnet50. The dataset used for comparison has 6000 images and the main operation in the methodology was to resize and reduce it so that it was consistent with CNN architecture. The results show that resnet50 performs the best when compared to the other two methods and with a total accuracy of 97.33%. The models described in the paper can be used to detect diseases. The models can be used for object detection which may have multiple models working in real-time. Inception v3 is an image recognization dataset that attained an accuracy of 78.1% on the imagenet dataset. In (Dere et al. 2021) for the extraction of features and the classification of various sports categories, Inception v3 is utilized in conjunction with neural networks.There are six different categories which are rugby, tennis, cricket, basketball, volleyball, and badminton which are being used for the classification and analysis. The performance of inceptionv3 is compared with random forest, K-nearest neighbors(KNN), and support vector machine(SVM). It can be observed that neural network achieves the best accuracy as they are capable of handling large datasets efficiently.

2.4 Image preprocessing and data augmentation

Tensorflow is a software developed by the Google Brain team that is used for machine learning and artificial intelligence tasks. Tensorflow includes a GPU (graphics processing unit) to help the model run faster. A classification of astronomical bodies such as a star, quasar, and galaxy is performed in (Ethiraj & Bolla 2021), TensorFlow is used to train the images, and a categorical variable with three different categories is used as the target variable. Image preprocessing is required to obtain a clean image free of distortions, and all images must be the same size to apply a convolution neural network. When the image size is small, the model training time is reduced without significantly reducing model performance. Identifying the best pre-processing methods can help improve the model's performance.

Data augmentation is used in (González et al. 2018) to avoid overfitting and increase the size of the dataset. Also using the techniques of data augmentation makes the dataset more robust. It demonstrates that using data augmentation improves the model's accuracy rather than decreasing it. Data augmentation that is well-founded can lead to better results. As a result, data augmentation is used to improve the model's accuracy.

3 Methodology

The main aim of this research is to devise an architecture that generates better precision, accuracy, and recall and critically compares the four deep learning methods for the classification of galaxies. A variety of data analysis methodologies were studied like KDD, CRISP-DM, and SEMMA. After a careful analysis, Knowlegde's discovery in the database(KDD) was selected as the final methodology to go ahead in this research.

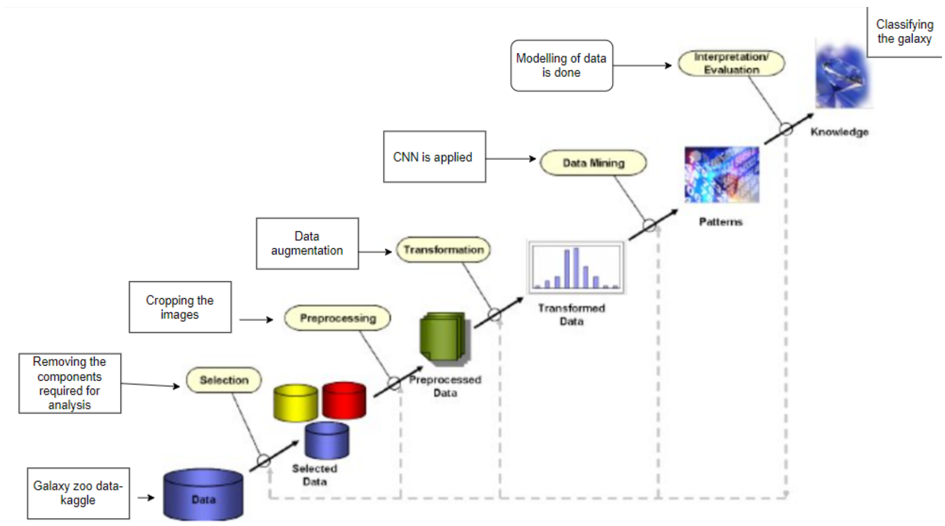

Figure 2: Design-specification of the flow[1]

3.1 Data selection

The images present on the Sloan digital sky survey are in FITS format. FITS is a format that holds data of some kind and image data. The dataset used in this research has been extracted from Kaggle and the galaxy zoo challenge [2].The zip file is divided into five separate folders images_training, solutions_training, images_test, all_ones_benchmark, all_zeros_ benchmark, central_pixel_benchmark. Images_training contains 61,578 images in JPG format. The morphology described in the dataset is a probability of each category as it is identified by crowdsourcing volunteer classification as a part of the galaxy zoo project. The given shapes of the galaxies are the probability for each category if

[1]http://www2.cs.uregina.ca/ dbd/cs831/notes/kdd/1_kdd.html
[2]https://www.kaggle.com/competitions/galaxy-zoo-the-galaxy-challenge

the number is high(which is close to 1) it indicates that many users have identified the category for the galaxy with a high level of confidence. If the number is on the lower side that which is closer to 0 it indicates that the feature is not present.
The galaxy zoo project was described by (Willett et al. 2013). The user is asked 11 questions which they answer depending on the decision tree the shape of the galaxy is classified. Each question had a predefined set of options, for example, question one: "Is the object a smooth galaxy, a galaxy with features/disk or a star?" had three possible responses: smooth, features or disk, star/artifact. The responses were then aggregated and converted to a probability based on the collated responses (Lintott et al. 2011)
There are a total of 32 classes in the dataset provided but for this particular research 3 major classes are chosen which are spiral, elliptical, and lenticular. Based on the classes of the galaxies the elliptical, spiral, and lenticular galaxies are found. The probability of a galaxy being elliptical will be decided if the class falls between 1.1 and 7.1, if the class falls between 1.1 and 7.2 it is lenticular and if the class is between 1.2 and 2.1 it is spiral. The total number of elliptical images is 7311, the Total number of spiral images is 4625 and the total number of lenticular images is 6625. The three different classes of images are previewed.

Figure 3: Eliptical galaxy

Figure 4: Lenticular galaxy

Figure 5: spiral galaxy

3.2 Data preprocessing and Exploratory data analysis

Data preprocessing is an important step in the model building. The galaxies selected in the data selection step is taken and are divided into the respective folders. Three different paths are created to access the required three classes. The original images are of a size of the image are 424 ×424 and it is cropped to a size of 180 × 180 as the images are more focussed on the center. The data is divided into the train and validation dataset in a ratio of 80:20 where 80% of the images are used to train the dataset and 20% images are used to test the dataset.

3.3 Data augmentation

Data augmentation is a technique in data analysis where data is increased by modifying the angle or by rotating the images in different directions. Data augmentation is done to increase the dataset if there are fewer images in the dataset to be trained on. In this research data augmentation techniques are used to create a larger dataset by re-scaling, rotating horizontally flipping, and zooming the images. Data augmentation is done to avoid overfitting, it also makes the data more robust.

3.4 Data transformation

The data was downloaded in the JPG format and three classes of galaxies were selected for the analysis of this research then the data were divided into train and test data in a ratio of 80/20. The images are not converted to any other format TensorFlow works with JPEG format As the data was downloaded in a group order, the images were shuffled before dividing into test and train data.

3.5 Data mining and learning approach

Data mining is the process of extracting usable data from a large volume of data. In this research, data mining is applied in the extraction of the three classes of galaxies this data is later fed to the models which are to be applied in this research. In this research to classify the shapes of the galaxies, transfer learning and convolutional neural network (CNN) are used. Transfer learning is the method in which pre-trained data is used as the starting point for the new task, the data on which the model is trained comes from the imagenet. The models used in the research were vgg19, densenet121,

3.6 Data Interpretation and evaluation

The information is included as zip files, which are opened in Collab by using the unzip function. The validation dataset, which makes up 20% of the remaining data in this study, is used to test the trained model. The standard against which the model's estimation is made would be its accuracy on the test data.
The main objective of this research is to evaluate the models and convolution neural network architecture and the transfer learning for the classification of galaxies

4 Design specification

This research aims at designing an architecture for the classification of galaxies. This section explains the different design that is implemented to carry out the research for this project. The process diagram is shown in the figure. The images which are used for image classification are extracted from Kaggle which is a part of the galaxy zoo project. The images are selected precisely in three classes from the 32 classes mentioned in the CSV file. The three classes in which the galaxies are divided into shapes are elliptical, lenticular, and spiral. The images are resized from 424×424 to 180×180. These resized images are then passed for data augmentation. The images are augmented by rotating and flipping horizontally. The data is augmented so as avoid overfitting in the model. The classification is performed using 5 deep learning models VGG19, Densenet121, inceptionv3, CNN, and convolution CNN these networks are trained on the dataset given. The mode's performance is evaluated using accuracy, precision, and recall.

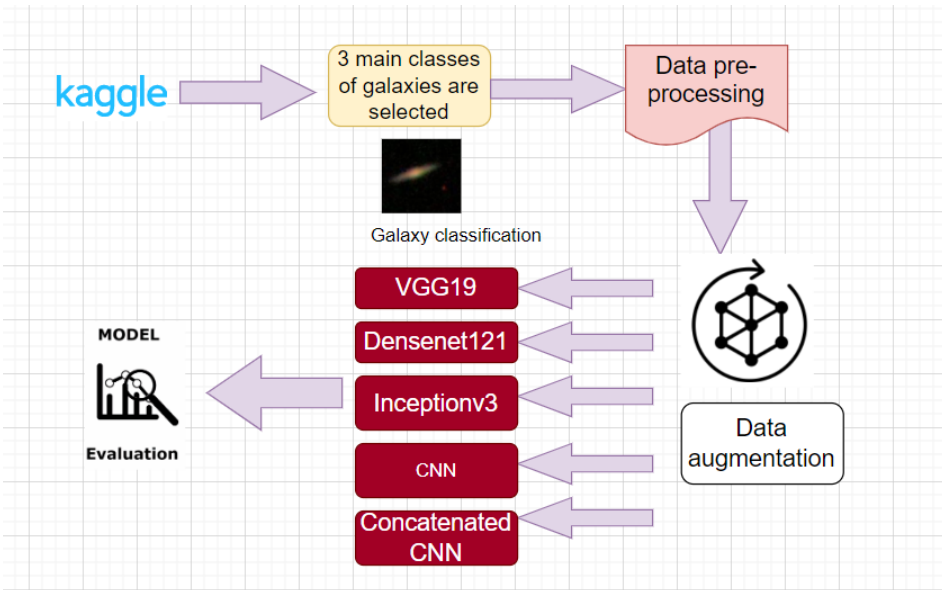

Figure 6: Design of the research

5 Implementation

This section describes the different models that were used to achieve the goal of this research to correctly classify the galaxy into their required shapes. The framework used to develop the deep learning models in this research were Tensorflow and Keras. Google collab and Jupyter notebook were used to code the main model and preprocess with a programming language such as python 3.7. The images are resized from 256×256 to 180×180 to just focus on the image of the galaxy before the model is created. The batch size used in this experiment was 32. Keras training functions are used to train the images. After the images are cropped, data augmentation is performed on the images where the images are flipped at an angle, rescaled, and horizontally rotated. The images are not changed to any other format like greyscale or RGB. The data was divided into train and validation data a total of 61,578 images were used which had a total of 32 classes which were then divided into 3 main classes of galaxies which are elliptical, spiral, and lenticular. Total images in elliptical were 7311, Total images in spiral were 4635 and finally lenticular was 6625. The training dataset of elliptical, lenticular, and spiral had 5848, 5300, and 3708. The validation dataset of elliptical, lenticular, and spiral had 1463, 1325, and 927 images. Categorical cross-entropy is used for all the output levels as the loss rate is set to 0.0001. Because the Adam optimizer incorporates the best elements of the AdaGrad and RMSProp algorithms, it can handle sparse gradients and is hence widely utilized. The architecture used in the projects was that of vgg19, densenet121, inceptionv3, and convolutional neural network which was created from scratch. All the research and experiments were performed using google collab with the given specification 2vCPU @ 2.2GHz, 13GB RAM, n1-higher-2 instance, and GPU instance downgraded to 64GB disk space.
The layers given below are added to the networks of vgg19, densenet121, inceptionv3, CNN, concatenated CNN

Layer Type	Corresponding Parameter
Polling layer	NA
Dropout layer	0.2
Batch Normalization layer	NA
Dense layer	1024
Dropout layer	0.2
Dense layer	1024
Batch Normalization layer	NA
Dropout layer	0.7
Dense layer	3

Figure 7: Layers to be added

5.1 VGG-19

VGG-19 classifies the images by utilizing the idea of transfer learning. The VGG-19 architecture comprises of 19 layers, 16 of which are convolutional layers, 3 of which are fully connected layers, 5 of which are maxpool layers, and 1 layer that is softmax. The network receives a fixed-size RGB image with dimensions of (224,224), indicating that the matrix was shaped (224,224,3). With a kernel size of (3 x 3), a stride of 1 pixel, and the ability to cover the entire image, VGG can do this. Implemented are three completely connected layers, two of which were 4096 pixels in size, and one dense layer with a 1000 by 1000 classification. The softmax function was utilized as the final layer's activation function.

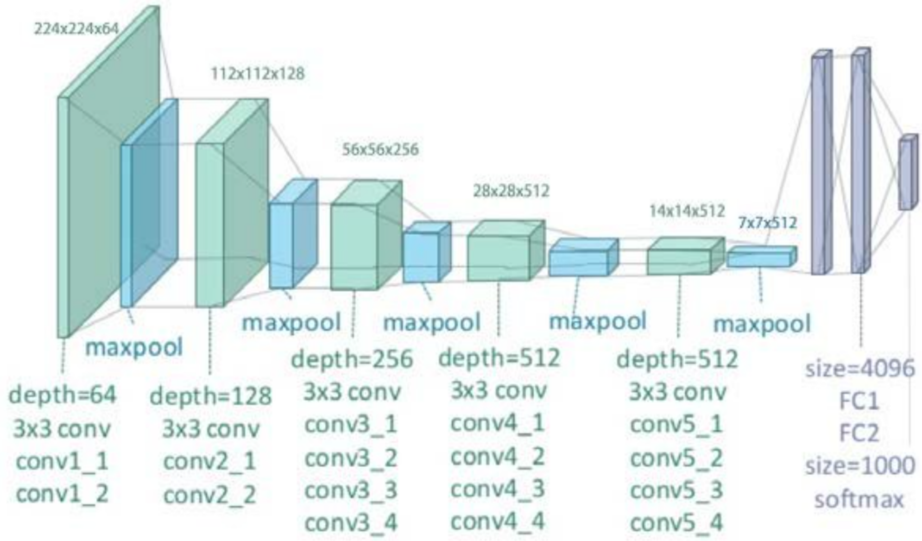

Figure 8: VGG-19 architecture[3]

5.2 Densenet121

A CNN called Densenet 121 has layers that are all connected to each other and to levels that are deeper in the network. Densenet features four average pooling layers and 120 convolutions. Deeper layers can be employed in the features extracted by distributing the weights of all the layers used in the same dense block and transition layers across several inputs. Densenet produces a better outcome than resnet and traditional CNN since it is more compact and performs at the cutting edge of technology.

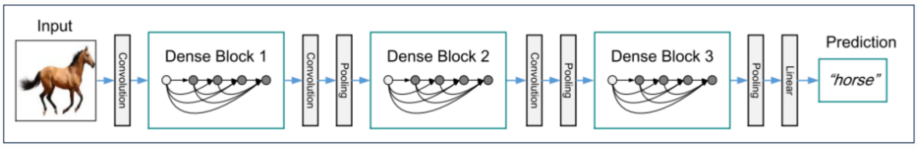

Figure 9: Densenet121 architecture[4]

[3]https://www.researchgate.net/figure/llustration-of-the-network-architecture-of-VGG-19-model-conv-means-convolution-FC-means_fig2_325137356

[4]https://towardsdatascience.com/creating-densenet-121-with-tensorflow-edbc08a956d8

5.3 Inception v3

Inceptionv3 is a CNN architecture which is another version from the inception family, it improvises on the factors like label smoothing, has factorized 7 x 7 convolutions, and the use of an auxiliary classifier to propagate label information lower down the network.

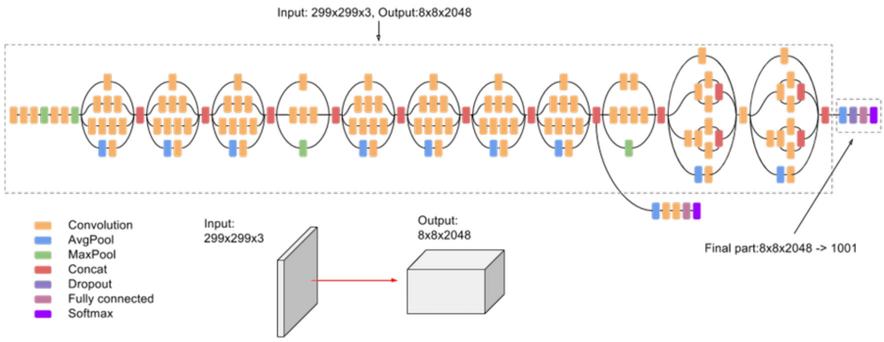

Figure 10: Inception v3 architecture[5]

5.4 Convolution neural network

To design the CNN network, different research papers were studied. In (Chen et al. 2019) a CNN classifier is proposed to improve image classification accuracy when it was applied to different images of breast cancer. In the model created for this research 4 convolution layers are used. The first convolution layer has 64 filters and relu is used as an activation function. The relu stands for rectified linear activation function for which will output the input itself if it is positive and output 0 otherwise. The third and fourth convolution layer has 128 filters respectively.The image is then flattened and a dropout layer of 0.7 is added.The dropout layer is added to avoid over fitting it randomly sets the input to 0.

[5]https://cloud.google.com/tpu/docs/inception-v3-advanced

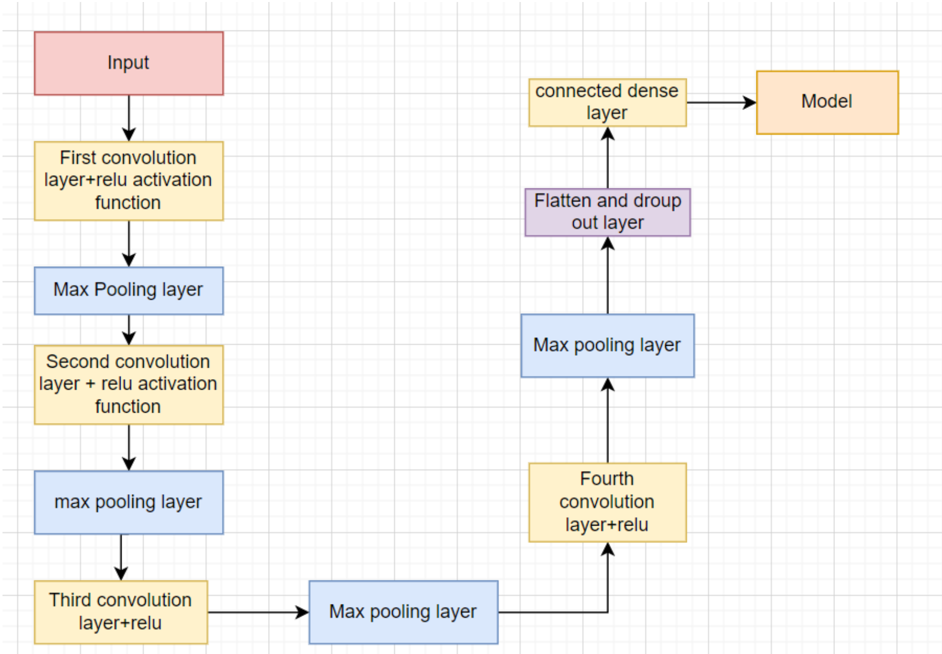

Figure 11: CNN architecture

5.5 Concatenated neural network

The concatenated neural network generally means concatenating two or greater output tensors from different network layers and concatenating them along the same channel. In attempting to concatenate the model, different techniques were tried which yielded a low accuracy. The final 2 networks which were concatenated are as follows.

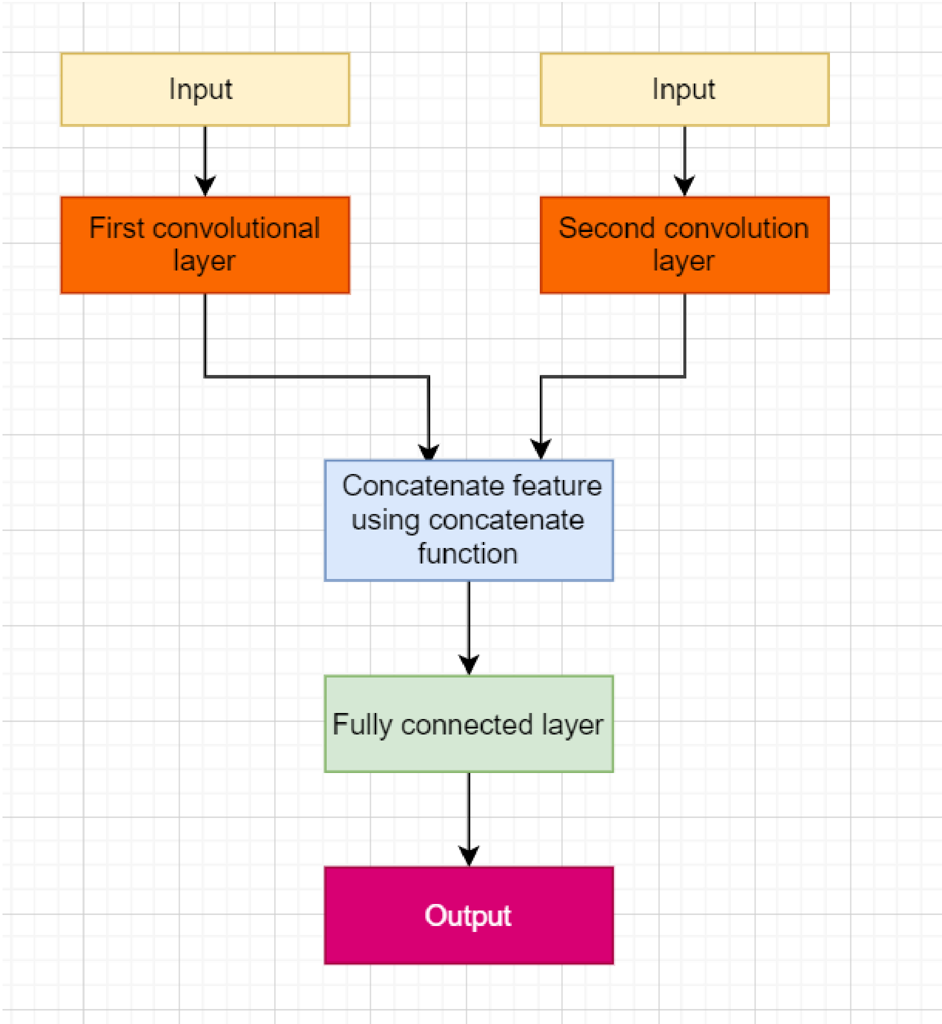

Figure 12: Concatenating CNN architecture

6 Evaluation

This section provides the context of the metrics which are to be used in the evaluation of the model's performance of the deep learning models which are being used. Upon reviewing the literature and work in the field of computer vision and deep learning it is seen that precision, Recall and accuracy (Jaiswal 2021) will be the right choice for measuring a model's performance. For all the models used in this research precision, recall and accuracy have been used.

6.0.1 Precision

The precision can be defined as the ratio of positive samples that are correctly classified to the total number of samples that are positive which are correct or incorrect.

6.0.2 Recall

The recall is determined by dividing the total number of positive samples by the proportion of positive samples that are accurately identified as positive. It is a ratio that defines the ability of a model to detect positive samples.

6.0.3 Accuracy

Accuracy is the most popular and common evaluation metric in the field of deep learning. Accuracy is a metric that is used to describe the classification performance of all the classes. The accuracy of the model is the percentage of correct prediction in comparison with the total prediction.

6.1 Experiment 1: Building a VGG19 model

VGG19 neural network is trained using the imagenet dataset and for this particular experiment, the VGG19 is accessed using the Keras API. A new set of layers is added to the network. The performance of VGG19 is measured using accuracy, precision, and recall. Figure 13 shows the developed architecture of the vgg19 network.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 vgg19 (Functional)          (None, 7, 7, 512)         20024384

 dropout (Dropout)           (None, 7, 7, 512)         0

 flatten (Flatten)           (None, 25088)             0

 batch_normalization (BatchN  (None, 25088)            100352
 ormalization)

 dense (Dense)               (None, 1024)              25691136

 batch_normalization_1 (Batc  (None, 1024)             4096
 hNormalization)

 activation (Activation)     (None, 1024)              0

 dropout_1 (Dropout)         (None, 1024)              0

 dense_1 (Dense)             (None, 1024)              1049600

 batch_normalization_2 (Batc  (None, 1024)             4096
 hNormalization)

 activation_1 (Activation)   (None, 1024)              0

 dropout_2 (Dropout)         (None, 1024)              0

 dense_2 (Dense)             (None, 3)                 3075

=================================================================
Total params: 46,876,739
Trainable params: 26,798,083
Non-trainable params: 20,078,656
_________________________________________________________________
```

Figure 13: VGG-19 architecture

The accuracy, precision, and recall of the network increase with the increase in the epochs. The loss decreases with an increase in the epochs. The accuracy achieved after adding the layers to vgg19 is 87% and the validation accuracy is 86.57%. The precision and recall achieved are 87.45% and 87.32%. The validation precision and recall achieved are 86.19% and 86.89%. The overall loss in the final epoch is 0.2002 and the validation loss is 0.2215. From the figure below the pattern of accuracy, loss, recall and precision can be seen.

Figure 14: Accuracy vs epoch graph on the left and the loss vs epoch graph on the right

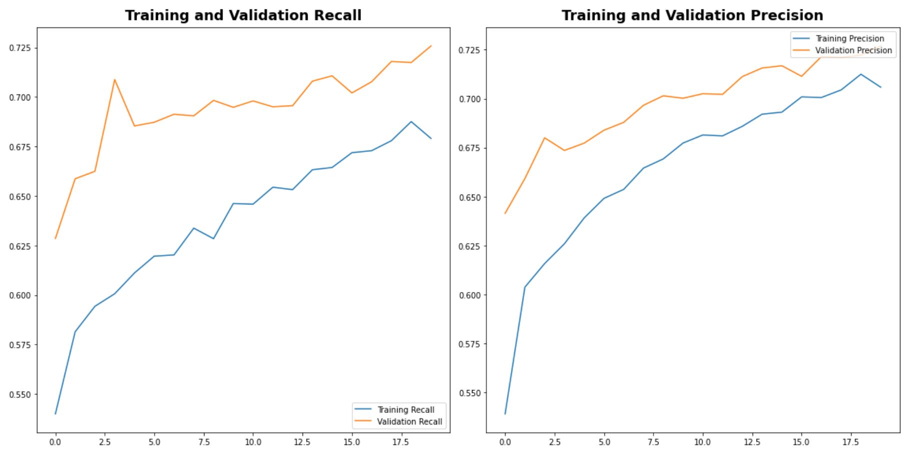

Figure 15: Recall vs epoch graph on the left and precision vs epoch graph on the right

6.2 Building a densenet121 model

Densenet121 neural network also uses the concept of transfer learning and the experiment is accessed using the Keras API. New layers are added to the already existing network of densenet121 and the performance of the modified densenet121 is compared with other neural networks.

```
Model: "sequential_5"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 densenet121 (Functional)    (None, 7, 7, 1024)        7037504

 dropout_4 (Dropout)         (None, 7, 7, 1024)        0

 flatten_2 (Flatten)         (None, 50176)             0

 batch_normalization_4 (Batc  (None, 50176)            200704
 hNormalization)

 dense_3 (Dense)             (None, 1024)              51381248

 batch_normalization_5 (Batc  (None, 1024)             4096
 hNormalization)

 activation_2 (Activation)   (None, 1024)              0

 dropout_5 (Dropout)         (None, 1024)              0

 dense_4 (Dense)             (None, 1024)              1049600

 batch_normalization_6 (Batc  (None, 1024)             4096
 hNormalization)

 activation_3 (Activation)   (None, 1024)              0

 dropout_6 (Dropout)         (None, 1024)              0

 dense_5 (Dense)             (None, 3)                 3075

=================================================================
Total params: 59,680,323
Trainable params: 59,492,227
Non-trainable params: 188,096
_________________________________________________________________
```

Figure 16: Densenet121 architecture

The accuracy, precision, and recall of the network increase with the increase in the epochs. The loss decreases with an increase in the epochs. The accuracy achieved after adding the layers to densenet121 is 89.35% and the validation accuracy is 88.087%. The precision and recall achieved are 89.33% and 89.32%. The validation precision and recall achieved are 87.99% and 88.18%. The overall loss in the final epoch is 0.1638 and the validation loss is 0.1803. From the figure below the pattern of accuracy, loss, recall and precision can be seen.

Figure 17: Accuracy vs epoch graph on the left and the loss vs epoch graph on the right

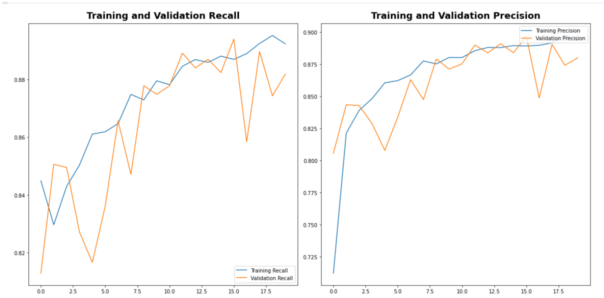

Figure 18: Recall vs epoch graph on the left and precision vs epoch graph on the right

6.3 Building an Inceptionv3 network

Inception v3 neural network is created and the network is accessed by the Keras API. New layers are added to the already existing network of Inceptionv3 and its performance is measured using precision, recall, and accuracy.

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 inception_v3 (Functional)   (None, 5, 5, 2048)        21802784

 dropout_3 (Dropout)         (None, 5, 5, 2048)        0

 flatten_1 (Flatten)         (None, 51200)             0

 batch_normalization_285 (Ba  (None, 51200)            204800
 tchNormalization)

 dense_3 (Dense)             (None, 1024)              52429824

 batch_normalization_286 (Ba  (None, 1024)             4096
 tchNormalization)

 activation_284 (Activation)  (None, 1024)             0

 dropout_4 (Dropout)         (None, 1024)              0

 dense_4 (Dense)             (None, 1024)              1049600

 batch_normalization_287 (Ba  (None, 1024)             4096
 tchNormalization)

 activation_285 (Activation)  (None, 1024)             0

 dropout_5 (Dropout)         (None, 1024)              0

 dense_5 (Dense)             (None, 3)                 3075

=================================================================
Total params: 75,498,275
Trainable params: 75,357,347
Non-trainable params: 140,928
_________________________________________________________________
```

Figure 19: Inception v3 architecture

The accuracy, precision, and recall of the network increase with the increase in the epochs. The loss decreases with an increase in the epochs. The accuracy achieved after adding the layers to inceptionv3 is 91.93% and the validation accuracy is 89.31%. The precision and recall achieved are 91.38% and 91.83%. The validation precision and recall achieved are 89.21% and 89.50%. The overall loss in the final epoch is 13.18% and the validation loss is 15.59%. From the figure and figure 15 below the pattern of accuracy, loss, recall and precision can be seen.

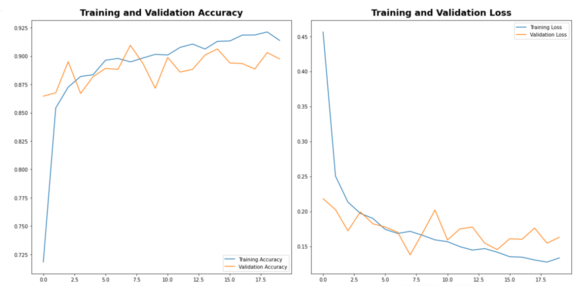

Figure 20: Accuracy vs epoch graph on the left and the loss vs epoch graph on the right

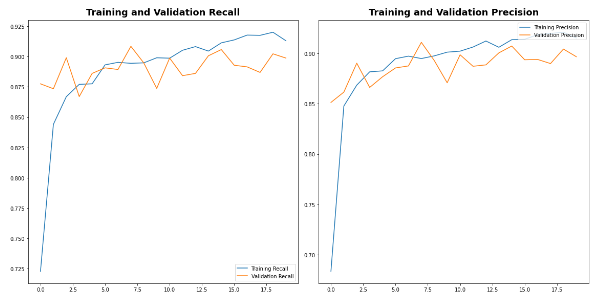

Figure 21: Recall vs epoch graph on the left and precision vs epoch graph on the right

6.4 Building a Convolution neural network model

CNN network used in this research is built from scratch with four different convolution layer. Activation function used in the network is relu. After the 4 convolution layer the image is flatten and a dropout layer is added to the network.Finally a dense layer to classify the 3 classes with activation as softmax is added.

```
________________________________________________________________________________________
 Layer (type)                   Output Shape         Param #     Connected to
========================================================================================
 input_5 (InputLayer)           [(None, 180, 180, 3  0           []
                                )]

 conv2d_21 (Conv2D)             (None, 178, 178, 64  1792        ['input_5[0][0]']
                                )

 max_pooling2d_11 (MaxPooling2D  (None, 89, 89, 64)  0           ['conv2d_21[0][0]']
 )

 conv2d_22 (Conv2D)             (None, 87, 87, 64)   36928       ['max_pooling2d_11[0][0]']

 conv2d_23 (Conv2D)             (None, 87, 87, 64)   36928       ['max_pooling2d_11[0][0]']

 max_pooling2d_12 (MaxPooling2D  (None, 43, 43, 64)  0           ['conv2d_22[0][0]']
 )

 max_pooling2d_13 (MaxPooling2D  (None, 43, 43, 64)  0           ['conv2d_23[0][0]']
 )

 concatenate_3 (Concatenate)    (None, 43, 43, 128)  0           ['max_pooling2d_12[0][0]',
                                                                  'max_pooling2d_13[0][0]']

 flatten_1 (Flatten)            (None, 236672)       0           ['concatenate_3[0][0]']

 dense_5 (Dense)                (None, 3)            710019      ['flatten_1[0][0]']

========================================================================================
Total params: 785,667
Trainable params: 785,667
Non-trainable params: 0
________________________________________________________________________________________
```

Figure 22: CNN architechure

The accuracy, precision, and recall of the network increase with the increase in the epochs. The loss decreases with an increase in the epochs. The accuracy achieved after adding the layers to CNN is 83.66% and the validation accuracy is 86.087%. The precision and recall achieved are 83.83% and 83.54%. The validation precision and recall achieved are 86.46% and 86.46%. The overall loss in the final epoch is 37.60% and the validation loss is 29.03%. From the figure below the pattern of accuracy, loss, recall and precision can be seen.

Figure 23: Accuracy vs epoch graph on the left and the loss vs epoch graph on the right

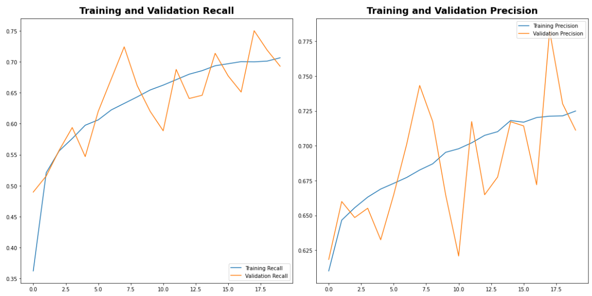

Figure 24: Recall vs epoch graph on the left and the precision vs epoch graph on the right

6.5 Building a concatenated Convolution neural network model

Concatenated CNN is built by concatenating 2 different neural networks and their features are concatenated using the concatenate function. First neural network has a conv layer attached to max pooling layer and conv layer attached to max pooling layer which is flattened and a droupout of 0.2 is added along with a dense layer of 32.

Network b is first flattened with the input and a dense layer is added and with relu as the activation function, a dropout layer of 0.2 is added to prevent overfitting and a dense layer is added. Both these layers are combined using the concatenation function and the output is taken using the dense layer and softmax activation function.

Model: "model"

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 180, 180, 3)]	0	[]
conv2d (Conv2D)	(None, 178, 178, 32)	896	['input_1[0][0]']
max_pooling2d (MaxPooling2D)	(None, 89, 89, 32)	0	['conv2d[0][0]']
conv2d_1 (Conv2D)	(None, 87, 87, 64)	18496	['max_pooling2d[0][0]']
max_pooling2d_1 (MaxPooling2D)	(None, 43, 43, 64)	0	['conv2d_1[0][0]']
flatten_1 (Flatten)	(None, 97200)	0	['input_1[0][0]']
flatten (Flatten)	(None, 118336)	0	['max_pooling2d_1[0][0]']
dense_1 (Dense)	(None, 64)	6220864	['flatten_1[0][0]']
dropout (Dropout)	(None, 118336)	0	['flatten[0][0]']
dropout_1 (Dropout)	(None, 64)	0	['dense_1[0][0]']
dense (Dense)	(None, 32)	3786784	['dropout[0][0]']
dense_2 (Dense)	(None, 32)	2080	['dropout_1[0][0]']
concatenate (Concatenate)	(None, 64)	0	['dense[0][0]', 'dense_2[0][0]']
dense_3 (Dense)	(None, 3)	195	['concatenate[0][0]']

Total params: 10,029,315
Trainable params: 10,029,315
Non-trainable params: 0

Figure 25: Concatenated CNN architecture

The accuracy, precision, and recall of the network increase with the increase in the epochs. The loss decreases with an increase in the epochs. The accuracy achieved after adding the layers to CNN is 83.66% and the validation accuracy is 86.087%. The precision and recall achieved are 83.83% and 83.54%. The validation precision and recall achieved are 86.46% and 86.46%. The overall loss in the final epoch is 37.60% and the validation loss is 29.03%. From the figure below the pattern of accuracy, loss, recall and precision can be seen.The graphs of the precision vs epoch is not that smooth it can be due to the fact normalisation layer is missing in the neural network b.

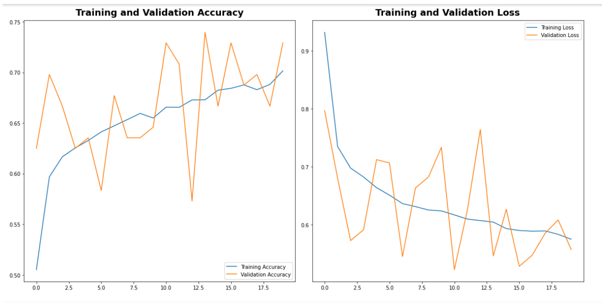

Figure 26: Accuracy vs epoch graph on the left and the loss vs epoch graph on the right

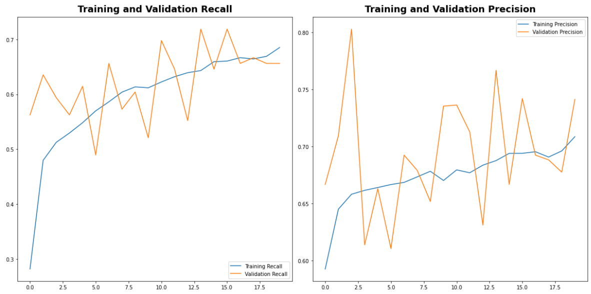

Figure 27: Recall vs epoch graph on the left and precision vs epoch graph on the right

6.6 Discussion

This research aimed to develop a Convolutional neural network that can produce better accuracy than the transfer learning network. A comparison of the results of all the networks was done to find out which neural network produces the best results to classify the galaxies. To design the CNN network different combinations of layers was taken. The final model was developed which yielded the highest accuracy as given in section 5.4.

The two neural networks described in section 5.5 are combined such that the two networks are compatible that is the size of the output of one of both the neural networks can be combined to produce the desired results.

The model's performance is compared using accuracy, precision, and recall. It takes into account the total number of correct predictions. The higher the value of the accuracy metrics the better it can classify correctly for all the classes. In the figure given below the accuracy, precision, and recall of all three can be depicted and can be seen. It can be seen that inceptionv3 performs the best and gives the highest validation accuracy.

Model type	Accuracy	Precision	Recall
VGG19	86.57	86.19	86.89
Densenet121	88.08	87.99	88.19
Inceptionv3	89.31	89.21	89.50
CNN	86.08	86.46	86.46
Concatenated CNN	72.92	74.12	65.62

Figure 28: Comparison of accuracy, recall, and precision

It can be seen that the concatenated CNN does not perform that well in comparison to the other methods. Changing the filter size or the activation function can be tried to improve the accuracy of the model

7 Conclusion

This research was aimed at developing a novel CNN network that can accurately classify the images of galaxies. In this research VGG19, Densenet121, inception v3, CNN, and concatenated CNN were used to classify the galaxies. It can be that inceptionv3 performs the best and delivers an accuracy of 91.93%. The lowest losses achieved are in inceptionv3. The aim of this research is finally fulfilled. The novel CNN created yielded an accuracy of 86.08% which can be used in the future for image classification. For future work, more networks can be concatenated to produce a more accurate result. A model can be tried on different image datasets. More astronomical data can be added to classify the deep space objects. A mobile application can be created for the classification of image data using the same architecture.A attempt to make a better concatenated model can be attempted.

References

Ba Alawi, A. E. & Al-Roainy, A. A. (2021), Deep residual networks model for star-galaxy classification, *in* '2021 International Congress of Advanced Technology and Engineering (ICOTEN)', pp. 1–4.

Biswas, M. & Adlak, R. (2018), Classification of galaxy morphologies using artificial neural network, *in* '2018 4th International Conference for Convergence in Technology (I2CT)', pp. 1–4.

Chao, L., Wen-hui, Z. & Ji-ming, L. (2019), 'Study of star/galaxy classification based on the xgboost algorithm', *Chinese Astronomy and Astrophysics* **43**(4), 539–548.
URL: *https://www.sciencedirect.com/science/article/pii/S0275106219300815*

Chen, G., Chen, Y., Yuan, Z., Lu, X., Zhu, X. & Li, W. (2019), Breast cancer image classification based on cnn and bit-plane slicing, *in* '2019 International Conference on Medical Imaging Physics and Engineering (ICMIPE)', pp. 1–4.

Dere, S., Fatima, M., Jagtap, R., Inamdar, U. & Shardoor, N. B. (2021), Anomaly detection in astronomical objects of galaxies using deep learning, *in* '2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)', IEEE.

Ethiraj, S. & Bolla, B. K. (2021), Classification of astronomical bodies by efficient layer fine-tuning of deep neural networks, *in* '2021 5th Conference on Information and Communication Technology (CICT)', pp. 1–6.

Fielding, E., Nyirenda, C. N. & Vaccari, M. (2021), A comparison of deep learning architectures for optical galaxy morphology classification, *in* '2021 International Conference on Electrical, Computer and Energy Technologies (ICECET)', pp. 1–5.

Freed, M. & Lee, J. (2013), Application of support vector machines to the classification of galaxy morphologies, *in* '2013 International Conference on Computational and Information Sciences', pp. 322–325.

González, R., Muñoz, R. & Hernández, C. (2018), 'Galaxy detection and identification using deep learning and data augmentation', *Astronomy and Computing* **25**, 103–109.
URL: *https://www.sciencedirect.com/science/article/pii/S2213133718300325*

Jaiswal, G. (2021), Performance analysis of incremental learning strategy in image classification, *in* '2021 11th International Conference on Cloud Computing, Data Science Engineering (Confluence)', pp. 427–432.

Kavala, S. A. P. N. & Pothuraju, R. (2022), Detection of grape leaf disease using transfer learning methods: Vgg16 vgg19, *in* '2022 6th International Conference on Computing Methodologies and Communication (ICCMC)', pp. 1205–1208.

Lintott, C., Schawinski, K., Bamford, S., Slosar, A., Land, K., Thomas, D., Edmondson, E., Masters, K., Nichol, R. C., Raddick, M. J., Szalay, A., Andreescu, D., Murray, P. & Vandenberg, J. (2011), 'Galaxy Zoo 1: data release of morphological classifications for nearly 900 000 galaxies', **410**(1), 166–178.

Mascarenhas, S. & Agarwal, M. (2021), A comparison between vgg16, vgg19 and resnet50 architecture frameworks for image classification, *in* '2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)', Vol. 1, pp. 96–99.

Reza, M. (2021), 'Galaxy morphology classification using automated machine learning', *Astronomy and Computing* **37**, 100492.
URL: *https://www.sciencedirect.com/science/article/pii/S2213133721000469*

Sookmee, P., Suwannajak, C., Techa-Angkoon, P., Panyangam, B. & Tanakul, N. (2020), Globular cluster classification in galaxy m81 using machine learning techniques, *in* '2020 17th International Joint Conference on Computer Science and Software Engineering (JCSSE)', pp. 98–103.

Tang, L. (2021), Image classification based on improved vgg network, *in* '2021 IEEE 6th International Conference on Signal and Image Processing (ICSIP)', pp. 316–320.

Willett, K. W., Lintott, C. J., Bamford, S. P., Masters, K. L., Simmons, B. D., Casteels, K. R. V., Edmondson, E. M., Fortson, L. F., Kaviraj, S., Keel, W. C., Melvin, T., Nichol, R. C., Raddick, M. J., Schawinski, K., Simpson, R. J., Skibba, R. A., Smith, A. M. & Thomas, D. (2013), 'Galaxy Zoo 2: detailed morphological classifications for 304 122 galaxies from the Sloan Digital Sky Survey', *Monthly Notices of the Royal Astronomical Society* **435**(4), 2835–2860.
URL: *https://doi.org/10.1093/mnras/stt1458*