Classifying Lesions in Clinical Dermatology Images Using Ensemble Networks

Sebastian Thomas
AC40001 Honours Project
BSc (Hons) Computing Science
University of Dundee, 2021
Supervisor: Prof. Emanuele Trucco

***Abstract* – The accurate classification of lesions in clinical dermatology is crucial in diagnosing problems such as skin cancer. Studies have previously found that the use of convolutional neural networks for classification of skin cancer achieves performance that is comparable to dermatologists' diagnoses. This paper looks at the use of ensemble neural networks to classify skin lesions into 8 classes and attempts to improve the accuracy compared to a single neural network. In this project we implement an ensemble approach by combining EfficientNet, ResNet-50 and NasNet architectures and we compare the performance to our own architecture. The answers from the networks in the ensemble were combined using model averaging. Evaluating on the test set shows that we have achieved an accurate classification of skin lesions with a satisfying test accuracy of 0.741. This ensemble is successful in averaging the models to give predictions better than two pre-trained models used in the ensemble. The accuracy score is close to that of the best performing model in the ensemble, so this experiment was a success.**

1 Introduction

1.1 Background and Motivation

Skin cancer is the most common form of cancer in the UK, but most skin cancers can be cured if they are diagnosed at an early stage [1]. According to Cancer Research UK melanoma skin cancer is the fifth most common cancer in the UK with approximately 16,200 new cases every year (2015-2017) [2]. This project aims to develop and implement deep learning techniques in biomedical image processing, specifically dermatology using ensemble networks. The use of deep neural networks to classify skin lesions promises to help dermatologists to provide more accurate diagnoses of skin lesions. This is important because it could help dermatologists to diagnose malignant skin lesions at an early stage which is crucial for curing them.

1.2 Problem Statement

This project focuses on the use of ensemble networks for classifying lesions, which is beneficial in providing more accurate results. The problem we are trying to solve is categorising images of skin lesions into eight different diagnostic categories: melanoma (MEL), melanocytic nevus (MN), basal cell carcinoma (BCC), actinic keratosis (AK), benign keratosis (solar lentigo / seborrheic keratosis / lichen planus-like keratosis) (BK), dermatofibroma (DF), vascular lesion (VL) and squamous cell carcinoma (SCC). The main goal is to employ convoluted neural networks (CNNs) to classify these skin lesions with an ensemble approach. The images in the dataset are from the HAM10000, MSK and BCN_20000 datasets [10][11][12]. These are all dermoscopic images.

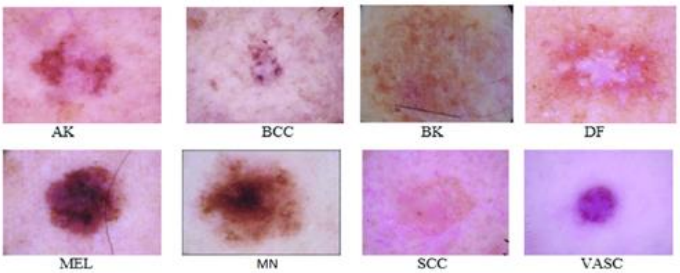

Figure 1 - Eight classes of skin lesions in 2019 ISIC dataset [7]

2 Background

2.1 Current method of skin lesion diagnosis

Classification of skin lesions is currently done visually by dermatologists. If they suspect that a lesion is malignant, then screening is followed by dermoscopic analysis, a biopsy and histopathological examination [3]. This visual examination can be difficult for dermatologists

because of the high variability in the appearance of lesions and this may lead to wrong detection of lesions [3]. Automated classification of skin lesions by using artificial intelligence offers a complementary solution to expert examination and can be used as a second opinion by dermatologists [4].

2.2 Recent Studies

The use of ensemble networks is comparable to a panel of experts agreeing on a classification. There are a few existing solutions that use ensemble networks to classify skin lesions. One of the papers the author studied is “skin lesion classification with ensembles of deep convolutional neural networks” [5]. This paper illustrates the importance of ensemble networks and the ways of fusing the outputs of four different deep neural network architectures. From this paper the author learned about data augmentation: if the volume of images required to train a convolutional neural network (CNN) is not large enough, then data augmentation can be used to add further images to the training set. This helps to increase accuracy avoid overfitting of the CNNs. This paper looks at many ensemble approaches like sum of probabilities, product of probabilities, simple majority voting and sum of maximal probabilities. Ensemble of neural networks is proven to be a good approach and results show that it outperforms individual networks regarding classification accuracy.

Another paper that was studied is “dermatologist-level classification of skin cancer with deep neural networks” by Esteva et al. [3]. This paper gave the author an introduction to the use of deep CNN in classifying skin lesion. It shows that deep learning CNN outperforms dermatologists at skin cancer classification. Even though this paper does not talk about ensemble networks it helps the author’s understanding of how CNNs can be used in classification. Learning about sensitivity–specificity curves and confusion matrices was very useful because it can be used in the project to show the effectiveness of the ensemble networks [3].

Another paper the author has looked at is “Ensemble Learning Based Multi-Color Space in Convolutional Neural Network” [6]. This paper is important because it reports a CNN using different colour spaces and it also details the process of converting from RGB colour space to other colour spaces, which may highlight different information in the images. Colour spaces were going to be investigated in the project as a possible way to create different input channels for different networks in an ensemble. However, the author did not get on to investigating this, so this is a recommendation for future work. The author has also learned of the different ways that confidence values can be combined such as voting, averaging and linear combination.

2.3 Convolutional Neural Networks

2.3.1 Basic Overview

CNNs are widely used for image processing, classification, and segmentation. CNNs are composed of multiple layers of artificial neurons. There are three types of layers in a convolutional neural network. Convolutional layers are the key building blocks of CNNs. They work by taking a small filter and moving this across the image to find certain features. The behaviour of each neuron is defined by the weights which are summed up and passed to the activation function [9]. Each of the layers in the CNN produces activation maps which can be used to extract relevant feature. Another layer in a CNN is the pooling layer. Pooling layers are usually used to reduce the size of inputs and speed up the computation [8]. Specific functions such as max pooling are used to reduce the dimensionality of the network. The last layer of CNNs is the fully connected layer. Fully connected layers aggregate information from final feature maps and generate a final classification.

2.3.2 Training

Training allows adjustment of weights in the neurons to extract desirable features from an image. In supervised learning, to train a network a training set and validation set is required. Validation set can be a subset of the training data which is only used for validating the model. Creating subsets of training data can be referred to as splits. Training split, validation split, test split is commonly used to refer to the sets of data used. When a network is being trained, each run through the training batch is called an epoch. These terminologies will be used in the paper.

3 Design

3.1 Initial Plan

The main objective of this project was to classify lesions in clinical dermatology images using ensemble networks. In addition to this the plan was to investigate lesion classification using different colour channels and colour spaces. A final objective was to learn about deep learning and develop a practical understanding of the main concepts behind machine learning.

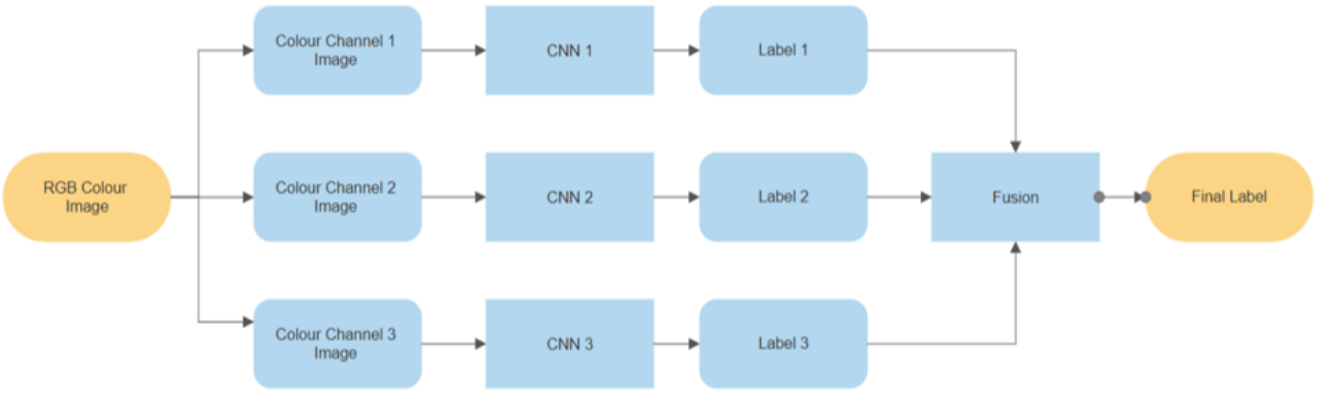

Figure 2 - The general idea behind the envisaged ensemble network

The design choices to be made included:

- whether the same CNN architecture should be used for all the channels, or different ones in different colour channels;

- whether the three networks can be trained with the same images (but different colour channels) or not;

- how the decisions from each colour channel should be fused.

The author planned to start by using the same CNN architecture for all the channels and if there was more time, the author would have explored different CNN architectures in different colour channels. Also, the author was planning to train the three networks with the same images to begin with; if there were significant differences then the author would have considered training different channels with different images. About information fusion from different channels, we expected to start with simple majority voting and later investigate probabilistic methods.

This plan is stated in the mid project progress report and you can find the timeline for this plan there.

3.2 Modified Plan

A lot had to be changed from the original plan for multiple reasons. Firstly, the author decided to experiment with his own architecture on the 2016 ISIC dataset to classify images to benign or malignant. This binary classification problem was investigated to start off simple and understand how CNNs work. In addition to this, the author experimented with his own architecture on the 2019 ISIC dataset to classify images into eight classes. This was investigated because then the results from this experiment could be used to compare with the results from the ensemble network and the individual pre-trained models. Before carrying out all these investigations a lot of time needed to be spent of further research because the author did not have any experience with Keras before.

After exploring basic architectures, the aim was to use pre-trained models and fine tune them to work on the 2019 ISIC dataset. The final and most important task was to ensemble three models using an ensemble method such as model averaging or simple majority voting. The initial plan included investigating colour spaces, however, this was not investigated due to problems with importing the OpenCV library on the server. Therefore, the author decided to investigate pre-processing functions instead.

The design choices made for the modified plan were to use different CNN architectures in each channel and train them with the same dataset. For ensemble method the plan is to look at simple majority voting and model averaging.

The timeline for the project was updated in March but due to changes in the project deliverables like the degree show being cancelled meant this timeline had to be updated again in April.

Figure 3 - Timeline for April

4 Materials and Methods

4.1 Technology

There is a wide variety of technologies that could have been used for this project. Several decisions had to be made to select the best technologies suited to the author. The author does not have previous experience in deep learning and the timescale for this project is not long, therefore this had to be considered when making the decisions. This section will detail the choices made and the reasoning behind them.

4.1.1 Programming Language

Python programming language was selected for this project because it is commonly used for deep learning. This meant that there is a large community of developers, therefore it is expected that there are plenty of tutorials and forums which the author can use to learn. Python is known for being beginner friendly while also being able to solve many complex tasks. Because of the short timescale of the project and the author's inexperience, this was the most sensible choice to make. The IDE used for this project is JupyterHub because this was the only option available on the server.

4.1.2 Neural Network API

Keras is a high-level neural network API which is written in Python. Keras is seen as user friendly and there are multiple tutorials available online which the author can use to learn. Keras can run on top of Tensorflow which can perform multiple machine learning tasks. An alternative backend is Theano, but this was not chosen because there were not many tutorials for beginners. Instead of Keras, an alternative is PyTorch, but this was not selected because it is not as beginner friendly as Keras. Also, Keras is seen as being better for experimenting with standard layers. So, the decision was made to use Keras running on top of Tensorflow 2.3.2.

4.1.3 Hardware

Throughout this project access to a Computer Vision and Image Processing (CVIP) server was available. The server used for this project was running two Nvidia GeForce RTX 2080 GPUs with 20 CPU cores and had 62GB of RAM. To access this server remote desktop connection was used to connect to the computing VPN and then the server.

4.1.4 Libraries

Several additional libraries were used in this project. Tensorflow hub library is used download and reuse pre-trained models with minimum amount of code. NumPy library is used in the project to work with the arrays and perform operations such as rounding and calculating mean. Matplotlib is used in this project to plot graphs such as model accuracy and loss and confusion matrix. Lastly, scikit-learn is used to import the confusion matrix and accuracy score functions.

4.2 Dataset

The datasets used in this project are both from The International Skin Imaging Collaboration (ISIC) archive. The datasets used in this project are 2016 ISIC dataset and 2019 ISIC dataset. These datasets are available to the public and the only information in this dataset is the images and the labels identifying the type of lesion. This meant no patient data was involved in this project, therefore no ethical considerations had to be made when using these datasets. A CSV file with labels for each image is also provided by ISIC which is used to split the datasets into the skin lesion categories and split into training, validation, and test sets. The 2016 ISIC dataset contains 900 images for training data and 379 images for test data all in JPEG format. The 2019 ISIC dataset contains 25,331 JPEG images for training data. The test data was not labelled for this dataset because it is current being use for a live challenge, therefore the test data had to be derived from the training set.

4.3 Training

4.3.1 Splits

For the 2016 dataset the training set was split into the subsets of training data and validation data. These subsets are referred to as splits. This gave a validation split of 0.1 and for the 2019 dataset a validation split and test split of 0.1 was created. Training was first done on simple architectures created by the author to see how well the CNNs perform. Then pre-trained models were used and fine-tuned to predict on the dataset.

4.3.2 Callback Functions

At first the number of epochs the author decided to run training for was set to a fixed number. This was not a good idea because different models require different number of epochs to give best result. This problem was solved by using early stopping callback function. Early stopping with a patience of 5 was used so that when the loss does not improve for five epochs the training stops. Model checkpoint function was used to save the best model after every epoch. If there is no improvement from the last epoch, then new model is not saved. Another callback function used in this

project is reduce learning rate on plateau. The learning rate starts at 0.0001 and the factor it reduces by is 0.2 with a patience of 3. So, if the loss does not improve for three consecutive epochs, then the learning rate is reduced. Reducing the learning is important a large learning rate means model will converge too quickly and a small learning rate takes longer for optimisation therefore a balance is required to obtain the optimal set of weights. The final callback function used in this project is CSV logger. This function saves the results from every epoch to a CSV file.

4.3.3 Metrics

In this investigation accuracy was used as a performance metric. Accuracy gives the percentage of correct predictions made. How the accuracy is calculated for each class is shown in Figure 9. Along with other accuracy other metrics such as precision, recall, f1 score and AUC should have been used because these are better metrics for dealing with imbalanced datasets. Accuracy is a skewed measure of performance for datasets that are highly imbalanced. This will be detailed in the recommendations for future work section of this paper.

$$Accuracy = \frac{TP + TN}{(TP + TN + FP + FN)}$$

Figure 9 - Formula for calculating accuracy (TP=true positives, TN=True Negatives, FP=False Positives, FN=False Negatives)

4.4 Ensemble Method

For this project, many ensemble methods were considered but model averaging seemed like the best option because of the short timescale of the project and the inexperience of the author. Model averaging guarantees the result will be better than the weakest model in the ensemble. Weighted averaging method could have been investigated to give even better accuracy because the author knows which model is the weakest and which is the strongest. Due to the timescale of the project this was not investigated. Simple majority voting method was included in the plan for investigation however the author realised there is a major disadvantage with this option. Only three models are used and there are eight classes of skin lesions so there is the possibility of all three models predicting a different class for some images. This is very likely to happen because the dataset being used in this project is very unbalanced. However, weights could be added to solve this. Adding more models and balancing the dataset would also solve this problem however, there was not enough time to implement this. Bootstrap aggregating and k-fold cross validation methods were also looked at, but these were difficult to implement even though the author understands the theory.

4.5 Evaluation

Validation accuracy and test set accuracy along with losses were used to evaluate the performance of the models. In addition, confusion matrices were used because they are often used to describe the performance for a classification model. Confusion matrices compare true values to values that are predicted by the model. It is basically a summary of the prediction results for the classification problem. These methods will be used to evaluate the performance of the models.

5 Experiments

This section of the paper details the experiments done. The first experiment the author conducted is classification of lesion images into benign and malignant classes using own architecture. This experiment helps the author to gain an understanding of deep learning. The second experiment conducted is classification of lesion images into eight classes using own architecture. This experiment also helps the author develop a practical understanding of the main concepts behind machine learning and the results can be compared to the ensemble results. Therefore, the last experiment is using pre-trained networks and ensemble method to give better results for classification. Using an ensemble of networks for classifying skin lesions is the main aim of this project.

5.1 Benign/Malignant Classification

For this classification problem I used two convolutional layers with filter sizes of 32 and 64 with a pooling layer after each convolutional layer. In the first convolutional layer we define the input size as 224x224x3. 224 is the image width and height and 3 is used because an RGB image is being inputted. ReLU activation function was used in the convolutional layers. This defines how the weighted sum of the inputs is changed into an output from the nodes in the layer. After each pooling in the next convolutional layer the filter size is increased because of the loss in pixels. A simple architecture is used because this is a binary classification problem and reducing complexity of architecture is a step to reduce overfitting. Overfitting is when the model performs well on the training set but fails to perform well on unseen datasets. After the final pooling layer, flatten

operation is performed to create a single feature vector. This operation converts anything greater than one dimension to one dimension. Dense layers only take vectors as an input so the 3D tensor from the convolutional layer must be flattened from 3D to 1D. This flattened feature map is passed through a dense layer with units set to 2 and activation set to softmax. This is the output layer where the predicted classes are outputted. Since this is a binary classification problem, units are set to 2 so that we get either benign or malignant as the output.

The softmax function gives us the probability, so it brings down the figures to between 0 and 1. An alternative to softmax that could have been used is the sigmoid function.

Before images are passed in pre-processing is applied to it. Data augmentation was investigated first however, there was not much difference in the confusion matrix produced. Using vgg16 pre-processing function gave better precision, so this was used in the experiment. Without this, the predictions on malignant class were not good.

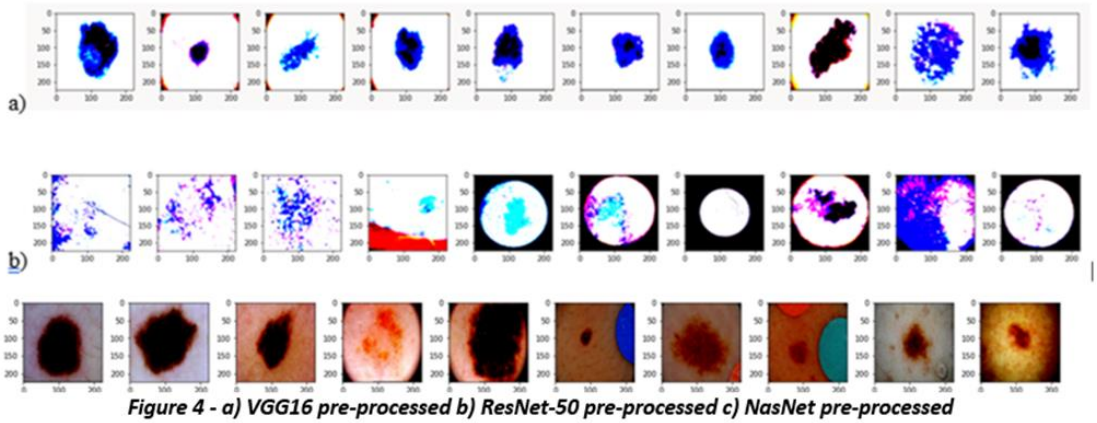

Figure 4 - a) VGG16 pre-processed b) ResNet-50 pre-processed c) NasNet pre-processed

5.2 Skin Lesion Classification

This classification problem is multi-class and the dataset used is the 2019 dataset which is much larger than the 2016 dataset. In this experiment the goal is to categorise the lesions into 8 classes which are mentioned in the introduction section of this report.

Like the previous experiment, the architecture used is made by the author instead of using a pre-trained model. For this problem, the architecture used on the 2016 dataset is too simple so there will be underfitting going on. Therefore, a deeper model is required so more convolutional layers will be added. To keep the architecture simple 3 convolutional layers were used. The filter size starts at 50 then increases to 75 and then 125 to learn more features. Relu is the activation function used throughout except in the output layer. Padding is set to 'same' in the convolutional layers so that the output has the same size as the input. The input size used in the first layer stays the same for the same reasons as stated for the last experiment. Flatten operation is performed after the convolutional layers. The hidden layer consists of 2 dense layers, both are followed by dropout operation.

Dense layers are used to improve the network's ability to classify extracted features. Dropout operation has been used in the convolutional layers and the hidden layer. Dropout is a way of regularisation to prevent overfitting. After the hidden layer is a final dense layer which is the output layer. So, the units are set to 8 because there are 8 classes and softmax is used for activation.

5.3 Ensemble Approach

In this experiment pre-trained models are used instead of building an architecture from scratch. The three models used in this experiment are ResNet-50, EfficientNet and NasNet-mobile. EfficientNet is used in this experiment so that results can be compared to related work. ResNet-50 is selected because it is known to perform well and is quite popular. NasNet-mobile was used because it was a good idea to see how well a smaller model that targets mobile devices perform. Another reason why these models were used was because they are all available to download from Tensorflow hub which means less code can be used. The pre-trained models are trained on the ImageNet data or CIFAR-10 data. Pre-trained networks are useful since they provide a good starting point because the features already learned

will be useful for experiment. This means they will be good at detecting high level features such as edges and patterns. To all these models a final dense layer needs to be added to give output of the 8 classes. The output layer used is the same as the one used in the previous experiment.

Before training the images need to be pre-processed. Each model has their own pre-processing functions. An example of what this looks like is shown in figure 4.

After getting results from each of the models, the model is saved and used later to give ensemble predictions. The ensemble method used in this experiment is detailed in the materials and methods section of this report. The test batch for the ensemble is pre-processed using vgg16 pre-processing function to keep it fair, so that one model does not have an advantage over the other.

6 Results and Discussion

6.1 Benign/Malignant Classification

For this experiment an accuracy of 0.79 is achieved on the test set with a loss of 2.2. Training ran for 14 epochs and then early stopping callback function stopped training. This allowed the model to stop when performance stops increasing. In figure 5 you can see that the loss is no longer improving so it is best to stop early instead of running for more epochs so that overfitting is reduced. The loss curve in Figure 5 shows a good learning rate. However, accuracy on the training set reaches 1.0 which means that there is overfitting. Furthermore, the large gap between training and validation curves gives a clear indication of overfitting going on.

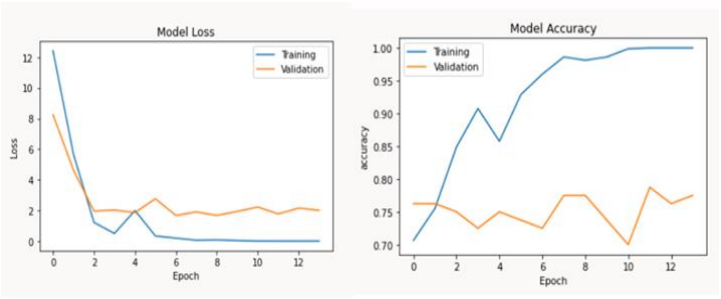

Figure 5 - Loss and Accuracy Graphs

The confusion matrix used to evaluate the performance on the test set shows that the network performed well on the benign class and not so well on the malignant class. This is caused by our dataset being imbalanced. The dataset contains four times as many benign images compared to malignant images. Therefore, better metrics to use would have been precision because it tells us the percentage of positive predictions that were correct. Recall and f1 score are also better metrics to consider.

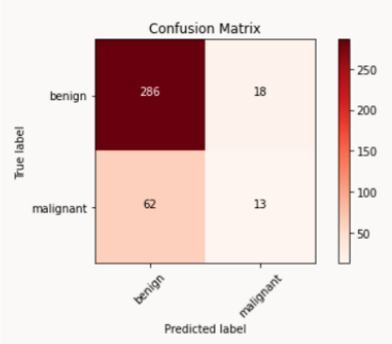

Figure 6 - Confusion matrix for benign/malignant classification

6.2 Skin Lesion Classification

From this experiment the accuracy achieved on the test set is 0.60 with a loss of 1.15. The accuracy is better than guessing but is still not a satisfactory performance especially since the dataset is highly imbalanced. Training was done for 14 epochs before the early stopping function acted. Accuracy is a flawed metric in this experiment too, so precision or recall should have been used. The model was not adequate in classifying all skin lesions because correct predictions were not made on classes like SCC and DF. A deeper model may be required to learn the underlying patterns in the images. Using transfer learning could improve the accuracy but precision still may not improve by much on account of the highly imbalanced dataset.

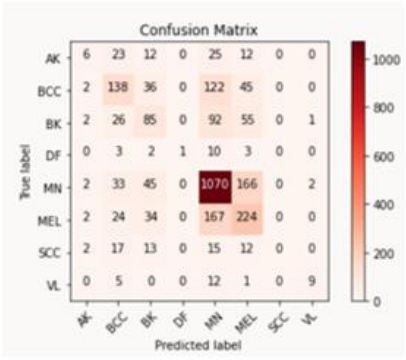

Figure 7 - confusion matrix for skin lesion classification

The confusion matrix for this experiment shows that the class MN had much more data compared to other classes such as DF, VL and SCC. A way to solve this problem is to try resampling the dataset. One way this could be done is by oversampling. For example, copies of the underrepresented classes could be made. Another way is under sampling, which means deleting instances of the overrepresented class. The confusion matrix for this experiment is shown in Figure 7.

6.3 Ensemble Approach

The ensemble model gave a good accuracy score of 0.741 on the test set. Evaluating on the same test set, ResNet-50 gave an accuracy of 0.81 and loss of 1.03. This was the best model in the ensemble followed by EfficientNet with an accuracy of 0.56 and loss of 1.56. The worst performance was given by NasNet which only gave an accuracy of 0.11 and loss of 11.71. The main reason for such a poor performance by NasNet was the pre-processing function used during training. NasNet was trained on data that with the NasNet pre-processing function applied. This suggests it does not work well with the vgg16 pre-processing function and does not perform well on data without the NasNet pre-processing function applied. ResNet-50 performs well because its pre-processing function is very similar to that of vgg16's. This shows that training models with their own pre-processing functions applied is not a good idea if they are working together as an ensemble.

Model averaging has proven to be a good ensemble method because the accuracy achieved is much greater than two of the models used in the ensemble and it is only 0.07 away from the accuracy of ResNet-50. In addition, using ensemble method gave much better results than using own architecture. This shows that the use of transfer learning and ensemble approach for this classification problem is useful in improving accuracy.

The confusion matrix for this experiment shows that imbalanced dataset is still causing a problem. However, comparing this to using the author's own architecture shows that the predictions have improved for the underrepresented classes and it can be clearly seen that predictions for BCC, BK, MEL, and MN are better. Even though the predictions are better than using own architecture, the predictions on the underrepresented classes still need to be improved. The dataset needs to be balanced to give excellent predictions.

As mentioned before resampling the dataset is a way of solving this problem. The short timescale of the project meant there was not enough time to do this.

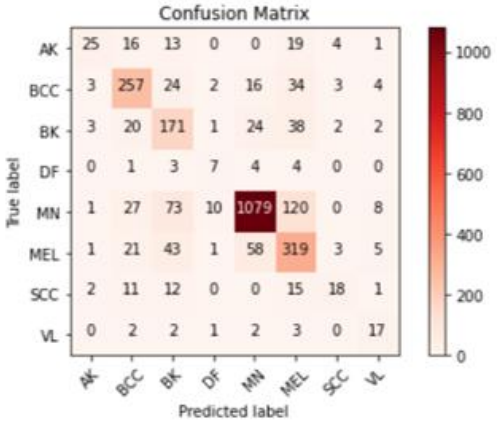

Figure 8 - confusion matrix for ensemble

7 Conclusion

The aim of this project was to classify lesions in clinical dermatology images using ensemble networks. In addition to this an objective was to learn about deep learning and develop a practical understanding of the main concepts behind machine learning. This project was successful in achieving these aims as the author has managed to apply his knowledge and create an ensemble network that classifies lesions.

The ensemble of networks managed to classify lesion images with an accuracy of 0.741. The ensemble method used by the author is quite simple and there is room for future work in exploring other ensemble methods. Initially, plans were made to investigate the effect of colour spaces, however this plan had to be changed due to the short timescale of the project. One major problem encountered during this project was the imbalanced dataset. Further work can be carried out in sorting this problem which is detailed in recommendations for future work section of this paper.

Overall, the author has achieved knowledge about deep learning and developed practical understanding by implementing solutions that classify skin lesions.

8 Recommendations for Future Work

There are many ways that this project could be improved. Firstly, introducing better metrics to measure the performance of model should be considered. Accuracy is a skewed measure of performance on highly imbalanced datasets. Introducing better metrics such as precision, recall and AUC is a better measure of performance. Precision-recall trade-off could be investigated, this leads to f1 score which is another metric that conveys balance between precision and recall. F1 score takes the harmonic mean of precision and recall. The formula for calculating these are shown in Figure 10.

$$PRE = \frac{TP}{TP + FP}$$

$$REC = TPR = \frac{TP}{P} = \frac{TP}{FN + TP}$$

$$F_1 = 2 \cdot \frac{PRE \cdot REC}{PRE + REC}$$

Figure 10 - Precision, Recall and F1 score formula [13]

Secondly, working with imbalanced data is an area that could be investigated because there are multiple ways to combat this. Imbalanced data problem could be solved by changing the performance metric, resampling data, generating synthetic samples. This is an area that the author would have looked at if there was more time. Imbalanced data meant that the accuracy metric is not good for measuring performance of models.

Thirdly, investigating the effect of colour spaces was an objective on the author's initial plan however the short timescale of the project meant the initial plan had to be modified and this was not included as an objective. Figure 2 gives an idea of how this could be implemented. After investigating pre-processing functions in this project, it is clear that model averaging is not the best ensemble method for this future work. Therefore, other ensemble methods like weighted averaging and majority voting will have to be explored too.

In addition to the last point, better ensemble methods are something that could be done to improve this project. Experimentations with a greater number of models and ensemble approaches such as simple majority voting, bagging, and boosting could be researched.

Lastly, another recommendation for future work is using the same pre-trained model in each channel and vary the training data so that models are trained on subsets of the training set. This investigation may also lead to interesting results and maybe more accurate and precise predictions.

9 Appraisal

This project has helped me gain an understanding of deep learning and the main concepts behind machine learning. I am proud to have been able to apply the knowledge gained. This has been a challenging project especially with all the self-teaching required and no face-to-face meetings due to the ongoing COVID-19 pandemic. I am glad to have finished this project to a really good quality in such a short timescale.

Acknowledgments

I would like to thank Prof. Emanuele Trucco for guidance and support throughout the project duration. He was always available to help me if I had any doubts or needed something reviewed.

References

[1] British Skin Foundation. 2021. What is skin cancer? [online] Available at: [Accessed January 2021].

[2] Cancer Research UK. 2021. Melanoma skin cancer statistics. [online] Available at: [Accessed January 2021].

[3] Esteva, A., Kuprel, B., Novoa, R. et al. Dermatologist-level classification of skin cancer with deep neural networks. Nature 542, 115–118 (2017). https://doi.org/10.1038/nature21056

[4] Hosny, K. M., Kassem, M. A., & Foaud, M. M. (2019). Classification of skin lesions using transfer learning and augmentation with Alex-net. PloS one, 14(5), e0217293. https://doi.org/10.1371/journal.pone.0217293

[5] Harangi, B. (2018). Skin lesion classification with ensembles of deep convolutional neural networks. Journal of Biomedical Informatics, 86, pp.25–32.

[6] J. Tan and N. Li, "Ensemble Learning Based Multi-Color Space in Convolutional Neural Network," 2019 IEEE Chinese Control Conference (CCC), Guangzhou, China, 2019, pp. 7924-7927, doi: 10.23919/ChiCC.2019.8865681

[7] Skin Lesions Classification Into Eight Classes for ISIC 2019 Using Deep Convolutional Neural Network and Transfer Learning - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/ISIC-2019-different-skin-lesions-examples_fig5_342325762 [accessed 25 Apr, 2021]

[8] Sharma, P., 2018. CNN Tutorial | Tutorial On Convolutional Neural Networks. [online] Analytics Vidhya. Available at: <https://www.analyticsvidhya.com/blog/2018/12/guide-convolutional-neural-network-cnn/> [Accessed 27 April 2021].

[9] Dickson, B., 2020. What are convolutional neural networks (CNN)?. [online] TechTalks. Available at: <https://bdtechtalks.com/2020/01/06/convolutional-neural-networks-cnn-convnets/> [Accessed 27 April 2021].

[10] Tschandl P., Rosendahl C. & Kittler H. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Sci. Data 5, 180161 doi.10.1038/sdata.2018.161 (2018)

[11] Noel C. F. Codella, David Gutman, M. Emre Celebi, Brian Helba, Michael A. Marchetti, Stephen W. Dusza, Aadi Kalloo, Konstantinos Liopyris, Nabin Mishra, Harald Kittler, Allan Halpern: “Skin Lesion Analysis Toward Melanoma Detection: A Challenge at the 2017 International Symposium on Biomedical Imaging (ISBI), Hosted by the International Skin Imaging Collaboration (ISIC)”, 2017; arXiv:1710.05006.

[12] Marc Combalia, Noel C. F. Codella, Veronica Rotemberg, Brian Helba, Veronica Vilaplana, Ofer Reiter, Allan C. Halpern, Susana Puig, Josep Malvehy: “BCN20000: Dermoscopic Lesions in the Wild”, 2019; arXiv:1908.02288.

[13] S. Raschka, “What is the best validation metric for multi-class classification?,” Sebastianraschka.com, 26-Apr-2021. [Online]. Available: https://sebastianraschka.com/faq/docs/multiclass-metric.html. [Accessed: 03-May-2021].