



كلية الحاسوب والذكاء الاصطناعي
Faculty of Computers & Artificial Intelligence

HELWAN UNIVERSITY

FACULTY OF COMPUTERS AND ARTIFICIAL INTELLIGENCE
COMPUTER SCIENCE DEPARTMENT



[A Comparative Study for Classification of Skin Cancer Using Four Approaches]

A graduation project dissertation by:

[Mohamed Maher Fouad (20180527)]

[Ahmed Galal Abd Elhameed (20180027)]

[Youssef Medhat Galal (20180721)]

[Mohamed Fathy Youssef (20180525)]

[Ahmed Ehab Mohamed (20180024)]

[Youssef Ahmed Ali (20180696)]

Submitted in partial fulfilment of the requirements for the degree of Bachelor of Science in
Computers & Artificial Intelligence, at the Computer Science Department, the Faculty of
Computers & Artificial Intelligence, Helwan University

Supervised by:

[Amr S. Ghoneim]

July 2022



كلية الحاسوب والذكاء الاصطناعي
Faculty of Computers & Artificial Intelligence



جامعة حلوان
كلية الحاسوب والذكاء الاصطناعي
قسم علوم الحاسوب

دراسة مقارنة لتصنيف أنواع سرطان [الجلد بأربع تقنيات]

رسالة مشروع تخرج مقدمة من:

- [محمد ماهر فؤاد (20180527)]
- [احمد جلال عبد الحميد (20180027)]
- [يوسف مدحت جلال (20180721)]
- [محمد فتحي يوسف (20180525)]
- [احمد ايها ب محمد (20180024)]
- [يوسف احمد علي (2018696)]

رسالة مقدمة ضمن متطلبات الحصول على درجة البكالوريوس في الحاسوب والذكاء الاصطناعي،
قسم علوم الحاسوب، كلية الحاسوب والذكاء الاصطناعي، جامعة حلوان

تحت إشراف:

(عمرو صبري غنيم)

يوليو / تموز 2022

Abstract

Skin cancer is considered as one of the most common and dangerous types among other cancers. Currently, between 2 and 3 million non-melanoma skin cancers and 132,000 melanoma skin cancers occur globally each year which accounts for only about 1%. in our study we used ISIC 2019 dataset which contains 8 classes with 25331 dermoscopic images. We have 4 approaches, first pre-processing comparison which we tried different techniques with different sizes and chose the best one which is edge finning and normalization with image size 224x224 which achieved on resNet-50 accuracy 85% and balanced accuracy 78% and on densenet-201 88% accuracy and 82% balanced accuracies. Secondly, we compared between different pre-trained models and the highest models are EfficientNetV2s, EfficientNetV2M and DenseNet-201 which achieved 88% accuracy and 82% balanced accuracy. Thirdly, we make different combination of the pre-trained models to form ensemble. We did 3 trials the best one achieved 90% accuracy and 84% balanced accuracy. Fourth Our model used One-Vs-All technique to build the binary tree classifiers and obtained 95.7% for F1-Score, 94.6% Sensitivity and 99% for Specificity.

الملخص

يعتبر سرطان الجلد من أكثر أنواع السرطانات شيوعاً وخطورة. في الوقت الحالي ، يحدث ما بين 2 و 3 ملايين من سرطانات الجلد غير الميلانينية و 132000 سرطان الجلد على مستوى العالم كل عام وهو ما يمثل حوالي 1 % فقط. في دراستنا ، استخدمنا مجموعة بيانات ISIC 2019 التي تحتوي على 8 فئات مع 25331 صورة مناظير الجلد. لدينا 4 طرق ، أول مقارنة للمعالجة المسماقة التي جربناها بتقنيات مختلفة بأحجام مختلفة واخترنا أفضلها وهو تشذيب الحواف والتقطيع مع حجم الصورة 224x 224 والذي حقق دقة ResNet-50 85٪ ودقة متوازنة 78٪ وعلى 201-DenseNet دقة 88٪ و 82٪ دقة متوازنة. ثانياً ، قارنا بين النماذج المختلفة المدربة مسبقاً وأعلى الموديلات هي EfficientNetV2s و EfficientNetV2M و DenseNet-201 التي حققت دقة 88٪ و 82٪ دقة متوازنة. ثالثاً ، نقوم بعمل مجموعة مختلفة من النماذج المدربة مسبقاً لتشكيل فرقه. لقد أجرينا 3 تجارب ، حقق أفضلها دقة 90٪ و 84٪ دقة متوازنة. رابعاً ، استخدمنا تقنية One-Vs-All لبناء مصنفات الأشجار الثنائية وحصل على 95.7٪ للحصول على درجة F1 و 94.6٪ حساسية و 99٪ للخصوصية.

Keywords

Skin cancer, dermoscopic, pre-processing, pre-trained, ensemble, classification, deep learning, ISIC 2019, multi classification, binary-based-tree, skin lesion, One-Vs-All.

Table of Contents

Table of Contents

Abstract.....	1
الملخص.....	1
Keywords.....	1

Table of Contents.....	1
List of Abbreviations	3
Glossary.....	4
Chapter 1: An Introduction.....	5
1.1 Overview	5
1.2 Motivation.....	8
1.3 Problem Statement.....	8
1.4 Work Methodology.....	8
Chapter 2: Related Work (Literature Review).....	8
2.1 Background	8
2.2 Literature Survey.....	9
Chapter 3: Dataset	16
Chapter 4: Pre-Processing.....	17
4.1 Image size	17
4.2 Augmentation.....	18
4.3 Normalization.....	18
4.4 Hair removal	19
4.5 Black Edge Removal.....	22
4.6 Manual Augmentation.....	23
Chapter 5: Pre-Trained Models.....	24
5.1 Background	24
5.1.1 Convolutional.....	24
5.1.2 Pooling	25
5.1.3 Flattening	25
5.1.4 Full Connection.....	26
5.2 Transfer Learning.....	26
5.2.1 Transfer Learning Definition	26
5.2.2 Basic Applications of Transfer Learning	27
5.2.3 Proposed method using Transfer Learning	27
5.3 Network Architecture:.....	28
5.3.1 ResNet50 (Detailed Explanation in Appendix):	28
5.3.2 DenseNet201(Detailed Explanation in Appendix):	29
5.3.3 InceptionV3(Detailed Explanation in Appendix):	30
5.3.4 MobileNet:	31
5.3.5 EfficientNetB0:	32
5.3.6 EfficientNetV2-S/M:.....	32

5.3.7 Xception:	33
5.3.8 VGG-19:	34
Chapter 6: Ensemble Learning	35
6.1 Ensemble Learning:	35
6.1.1 Weighted Soft Voting:	36
6.1.2 Nelder-Mead minimizing function:	36
Chapter 7: Binary tree-based multi-classifier	37
Chapter 8: Implementation & Results	38
8.1 Performance metrics	39
8.2 Conducted Result.....	40
8.2.1 First Approach	40
8.2.2 Second Approach	41
8.2.3 Third Approach.....	58
8.2.4 Fourth Approach	64
Chapter 9: Discussion, Conclusions, and Future Work	64
9.1 Summary & Conclusion.....	64
9.2 Future Work	66
References	66
Appendix I (<i>Pretrained Models</i>)	73

List of Abbreviations

- CAD - Computer-Aided Diagnoses system.
- ISIC-International Skin Imaging Collaboration.
- CNN-Convolution Neural Networks.
- GPU-Graphical Processing Unit.
- LPT -Learning by Passing tests.
- SGD-Stochastic Gradient Descent.
- AUC-Area Under Curve.
- NV-Melanocytic Nevi.
- BKL-Benign keratosis-like lesions.
- AK-Actinic keratoses and intraepithelial Carcinoma.
- SCC-Squamous Cell Carcinoma.
- VASC-Vascular lesions.
- DF- Dermatofibroma.

MEL-Melanoma.

BCC-Basal cell carcinoma.

ANN-Artificial Neural Network.

TL-Transfer Learning.

ILSVRC-ImageNet Large Scale Visual Recognition Challenge.

FLOPS- Floating-point operations per second.

OVO – One-Vs-One

OVA- One-Vs-All

WHO- world Health Organization

Glossary

Bowen's disease (akiec)	a very early form of skin cancer that's easily treatable. The main sign is a red, scaly patch on the skin. It affects the squamous cells, which are in the outermost layer of skin, and is sometimes referred to as squamous cell carcinoma <i>in situ</i> .
Actinic keratoses (akiec)	are dry scaly patches of skin that have been damaged by the sun. The patches are not usually serious. But there's a small chance they could become skin cancer also called solar keratoses
intraepithelial carcinoma (akiec)	recently described lesion characterized by replacement of endometrial surface epithelium or glands by malignant cells resembling high-grade invasive endometrial carcinoma.
basal cell carcinoma (bcc)	a type of skin cancer that most often develops on areas of skin exposed to the sun, such as the face. On brown and Black skin, basal cell carcinoma often looks like a bump that's brown or glossy black and has a rolled border. Basal cell carcinoma is a type of skin cancer
Dermatofibroma (df)	(Superficial benign fibrous histiocytoma) is a common cutaneous nodule of unknown etiology that occurs more often in women. Dermatofibroma frequently develops on the extremities (mostly the lower legs) and is usually asymptomatic, although pruritus and tenderness can be present
Melanoma (mel)	is a type of skin cancer that develops when melanocytes (the cells that give the skin its tan or brown color) start

	to grow out of control? Cancer starts when cells in the body begin to grow out of control. Cells in nearly any part of the body can become cancer and can then spread to other areas of the body.
melanocytic nevi (nv)	Melanocytic nevi represent proliferations of melanocytes that are in contact with each other, forming small collections of cells known as nests. Melanocytic nevi commonly form during early childhood. Their onset is believed by some authorities to be, at least in part, a response to sun (ultraviolet) exposure
vascular lesions (vasc)	Vascular lesions are relatively common abnormalities of the skin and underlying tissues, more commonly known as birthmarks. There are three major categories of vascular lesions: Hemangiomas, Vascular Malformations, and Pyogenic Granulomas.
benign keratosis-like lesions (bk1)	Seborrheic keratoses are noncancerous (benign) skin growths that some people develop as they age. They often appear on the back or chest but can occur on other parts of the body. Seborrheic keratoses grow slowly, in groups or singly. Most people will develop at least one seborrheic keratosis during their lifetime
Histopathology (histo)	is the diagnosis and study of diseases of the tissues
Biopsy	A small sample of tissue that's taken for testing
Hypodermis	The bottom layer of skin in the human body
Dermis	The inner layer of the two main layers of the skin
Epidermis	The outer layer of the two main layers of the skin

Chapter 1: An Introduction

1.1 Overview

Skin cancer is considered as one of the most common and dangerous types among other cancers [1], Skin cancer is the out-of-control growth of abnormal cells in the epidermis, the outermost skin layer, caused by unrepaired DNA damage that triggers mutations. These mutations lead the skin cells to multiply rapidly and form malignant tumors. The two main causes of skin cancer are the sun's harmful

ultraviolet (UV) rays and the use of UV tanning beds. You can reduce your risk of skin cancer by limiting or avoiding exposure to ultraviolet (UV) radiation. According to WHO The incidence of both non-melanoma and melanoma skin cancers has been increasing over the past decades. The three most frequent forms of skin cancer are Basal Cell, Squamous Cell, and Melanoma. Melanoma is the worst form of skin cancer, killing thousands of people each year. Currently, between 2 and 3 million non-melanoma skin cancers and 132,000 melanoma skin cancers occur globally each year which accounts for only about 1%. One in every three cancers diagnosed is a skin cancer and, according to Skin Cancer Foundation Statistics, one in every five Americans will develop skin cancer in their lifetime [2], Doctors ordinarily use the biopsy method for skin cancer detection. This procedure removes a sample from a suspected skin lesion for medical examination to determine whether it is cancerous or not. This process is painful, slow, and time-consuming. Computer-Aided Diagnoses system (CAD) provides a comfortable, less expensive, and speedy diagnosis of skin cancer symptoms. the process of defining and extracting features by a human expert, is a cumbersome and time-consuming task. Deep learning eliminates the need for feature engineering because it can automatically learn and extract useful features from the raw data. In our project we used the International Skin Imaging Collaboration (ISIC) dataset which has become a leading repository for researchers in machine learning for medical image analysis, especially in the field of skin cancer detection and malignancy assessment. They contain tens of thousands of dermoscopic photographs. The associated yearly challenges have resulted in major contributions to the field.

Deep learning has been applied to resolve very complex classification and segmentation tasks [3],[4]. The architecture of these networks is mainly based on convolutional layers. These layers filter and extract essential features of the images to learn to identify different lesions. For example, Zhou *et al.* [5] used different modality images to learn the features that determine dementia cases. Commonly named Convolutional Neural Networks (CNNs), they have been applied to many areas of interest, showing exceptional performance in image and video processing [6], [7]. Nowadays, CNNs use the power of GPUs to compute a large number of operations in a few seconds, allowing them to process large datasets to create reliable models to be applied to image classification, decision support systems and object recognition and segmentation. With the increase of publicly available datasets, deep networks have shown excellent performance on medical image analysis.

Transfer learning which is a machine learning method where a model developed for a task is reused as the starting point for a model on a second task. The reuse of a previously learned model on a new problem is known as transfer learning and the model is named a pre-trained model. A pre-trained model is a model created by someone else to solve a similar problem. Instead of building a model from scratch to solve a similar problem, you use the model trained on other problem as a starting point. It's particularly popular in deep learning right now since it can train deep neural networks with a small amount of data. This is particularly valuable in the field of data science, as most real-world situations do not require millions of labelled data points to train complicated models. In computer vision, neural

networks typically aim to detect edges in the first layer, forms in the middle layer, and task-specific features in the latter layers. The early and central layers are employed in transfer learning, and the latter layers are only retrained. It makes use of the labelled data from the task it was trained on, Transfer learning offers several advantages, the most important of which are reduced training time, improved neural network performance (in most circumstances), and the absence of a large amount of data.

In this investigation study we have implemented 4 approaches starting with image pre-processing which is an essential step of detection to remove noises such as hair noise, irrelevant details such as black circle around some images and enhance the quality of original image. We used different pre-processing techniques such as images resize, normalizations, hair removal, augmentation and edge fining we will talk about each of them later in processing section.

The first approach is choosing the-state-of-art deep CNN classifiers which were trained on ImageNet dataset and retrain them using transfer learning concepts on ISIC 2019 dataset by resizing the images to different sizes (32x32, 64x64 and 224) and applied different pre-processing methods.

In second approach we have chosen the best pre-processing methods according to the models' performance in the previously stated approach, we experimented on different pre-trained models (VGG-19, ResNet-152, ResNet-50, DenseNet-201, MobileNet, EfficientNet-B0, InceptionV3, EfficientNetV2M, EfficientNetV2S, NASNetMobile and Xception).

In third approach we combine best pretrained models according to their accuracy in the previous approach to form an ensemble, by using a voting ensemble (or a "majority voting ensemble") which is an ensemble machine learning model that combines the predictions from multiple other models. It is a technique that may be used to improve model performance, ideally achieving better performance than any single model used in the ensemble. A voting ensemble works by combining the predictions from multiple models. It can be used for classification or regression.

In fourth approach investigates how a binary tree technique can be used to diagnose skin cancer. Skin cancer is a multiclass classification problem, which is challenging for medical problems like ours. The binary classifiers and problem classes are represented as nodes in the graph or tree. The root node usually contains a classifier that divides all problem classes into two different groups. These groups are also iteratively broken into two pieces each, until only one unique class is left. The hierarchical structure used may have an impact on the quality of the solution obtained in the multiclass problem. As a result, the classes linked to each node may have a significant impact on the overall classification accuracy. We used the approach proposed by (Ana Carolina, Andre' C.P.L.F. de Carvalho. 2010) [6] in their paper. A common approach to generalizing binary learning techniques to handle problems with more than two classes, often known as multiclass classification problems, is to break the multiclass problem into several binary sub-problems, the outputs of which are combined to form the predicted class. This

technique yields a tree of binary classifiers, with each internal node representing a binary classifier separating two different groups of classes and the leaf nodes representing the problem classes.

1.2 Motivation

Even with dermoscopic images, dermatologists struggle to make an accurate diagnosis of skin cancer since various skin cancer forms can appear similar at first.

However, even skilled dermatologists are limited by their research and experience because they are only exposed to a fraction of all probable skin cancer symptoms during their lifetime. According to studies on clinical dermatology diagnostic accuracy, dermatologists with three to five years of expertise had a 62% accuracy rate. Dermatologists with more than ten years of expertise, on the other hand, can attain 80% accuracy. This study makes use of the International Skin Imaging Collaboration-2019 skin cancer dataset (ISIC-2019).

1.3 Problem Statement

Skin lesions pre-processing investigation and classification into 8 classes using pretrained models, ensemble method and Binary tree-based multi-classifier.

1.4 Work Methodology

- Pre-processing comparison.
- Pretrained models comparison.
- Soft voting Ensemble.
- Binary tree-based multi-classifier.

Chapter 2: Related Work (Literature Review)

2.1 Background

The research of skin cancer detection based on image analysis has advanced significantly over the years. Many different techniques have been tried, researchers have tried to improve the accuracy of diagnosis by employing different classification algorithms and techniques. Image classification took to new bounds when convolutional neural network (CNN) structure was introduced by Fukushima (1988) and later Le-Cunn (1990). They used CNNs for image classification. CNNs basically mimic the human visual cognition system and are considered to be the best state-of-the-art methods for image classification. Though there is a plethora of literature available on image classification, we limit our review of the literature to deep learning methods and ensemble learning for skin cancer images.

2.2 Literature Survey

The first breakthrough on skin cancer classification by a pre-trained GoogleNet Inception V3 CNN model came from Esteva et al [8] They used 129,450 clinical skin cancer images including 3,374 dermatoscopic images. The reported accuracy of classification is 72.1 ± 0.9 . In 2019 Khan et al [9]. They utilize two pretrained models of ResNet series like ResNet-50 and ResNet101 for deep features extraction only which returns an output of size $N \times 2048$ for both. the reported accuracy of classification is 89.8% for HAM1000, 95.60% for ISBI 2017, 90.20% for ISBI 2016. In 2021 Kandukuri et al [10]. this study is to diagnose the early stages of skin cancer using the method of Neural Architecture Search, improved by the technique of LPT (Learning by Passing tests) which chooses the best architecture over various deep learning models, The Model is optimized using SGD. the reported accuracy of classification is 88.09% by using PCDARTS-LPT on skin cancer dataset and 84.66% by using DARTS-LPT on HAM10000 dataset. In 2020 Kwasigroch et al [11]. this study Deploy one neural architecture search methods to the non-benchmarks' datasets for the medical problem of detecting malignant melanoma or classify the skin lesions to malignant and benign. The accuracy for search bases on loss was 76.38 and for the search bases on accuracy was 72.36. In 2021 Muhammad Attique et al [12]. Segmentation and classification of skin lesions use CAD system, A new CNN based method is presented in this work for skin lesion detection and recognition, two different tasks are performed, segmentation and classification. In the first task, segmentation is performed through Mask RCNN in this architecture ResNet50 is used along with feature pyramid network (FPN) as a backbone. In 2021 Khan et al [13]. they Develop fully automated system for Skin lesion classification and segmentation with the most discriminant features for deep feature extraction transfer learning approach is used from two pre-trained CNNmodels ResNet101, DenseNet201. In 2021 et al [14] .In this study Ensemble based CNN architecture where multiple CNN models, some of which are pre-trained and some are trained only on the data at hand, along with auxiliary data in the form of metadata associated with the input images, are combined using a meta-learner to Classify dermoscopic skin cancer images (M,B) , The Reported F1=0.23 , AUC-PR=0.16 ,AUC-ROC =0.87 . In 2021 Mijwil et al [15]. In this Study they Classify high resolution skin cancer images into malignant and benign by using (ConvNet) model applying with three architectures (InceptionV3, ResNet, and VGG19). The reported accuracy of classification is 86.90%. In 2020 Balaji et al [16]. Segmenting the affected regions in the skin and classifying their type according to the classes in the dataset which are malignant, benign and keratosis. The reported accuracy of classification is 72.7%. In 2021 Damian Michis et al [17]. in this study they use Segmentation Of skin lesion and classifying its type to Melanoma or Nevi, pre-processing for noise reduction, hair removal, and image segmentation they use a kNN-CV algorithm RBFNN (Radial basis function neural network) as classifiers. the reported 7-Point =95.42%, MedNode =94.7%, PH2=94.88%. In 2021 Thurnhofer-Hemsi et al [18]. in this study they Use ensemble of CNN to classify skin lesion Images into seven classes use Classifiers MobileNetV2, GoogLeNet. The reported accuracy of classification is 83.6%. In 2020 Nils Gessert et al [19]. they Use ensemble of deep learning models to

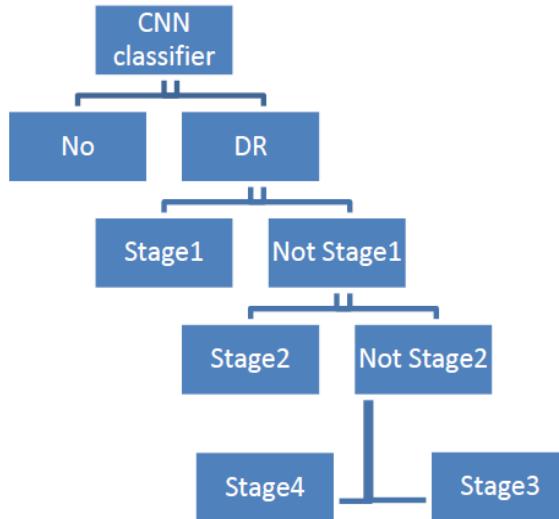
classify skin lesions to eight classes using dermoscopic images and meta data provided for the patient in the dataset in terms of the two tasks in the ISIC 2019 Challenge, with Classifiers CNN pretrained models such as (EfficientNets, SENet154 two ResNext with weakly supervised learning WKL). For Mel class the AUC in task1 0.928 And in task2 0.931. In 2021 Yu et al [20]. in This paper is discussing different procedures for remote diagnosis through IOT devices and then compare different ML models and DL model that have worked in skin disease classification, finally after detecting the best model for every disease implement the Targeted Ensemble Machine Classification Approach. The reported accuracy of classification is 98.48%. In 2019 Albahar et al [21]. in this study they propose a new prediction model that classifies skin lesions into benign or malignant lesions based on a novel regularizer technique along with CNN architecture CNN with a novel regularizer which is based on standard deviation of the weight matrix of the classifier. The reported accuracy of classification is 97.49%. In 2019 Kadampur et al [22]. The paper illustrates A model-driven architecture in the cloud named DLS (Deep Learning Studio), that uses deep learning algorithms in its core implementations, is used to construct models that assist in predicting skin cancer whether it's malignant or benign. In 2020 Naresh et al [23]. The aim of the paper is to provide insights about different categories of skin lesions and methodologies implemented to classify and predict skin cancers, and the role of dermatologists while developing the models also this paper provides a leading path for researchers and dermatologists to understand the basics algorithms in ML and broad extent of uses So this research doesn't have its own method for classification it only makes a literature review about other methods and their achievements. In 2018 Gessert et al [24]. Classify skin lesions based on the classes presented in the dataset to seven classes. Classifiers SENet154, ResNeXt101, DenseNet201, DenseNet161, DenseNet169, SE-Resnet101, PolyNet, SVM for final classification. In 2021 Ahmed et al [25].in this study they Classify skin lesion images by using ensemble of the most relevant pretrained model to classify the images into 8 classes. The reported accuracy of classification is 0.93. In 2019 Lee et al [26]. In this paper they Design the WonDerM pipeline, that resamples the pre-processed skin lesion images, builds neural network architecture fine-tuned with segmentation task data, and uses an ensemble method to classify the seven skin diseases. The ensemble of models achieved an accuracy of 0.899 and 0.785 in the validation set and test set, respectively. In 2020 Islam et al [27]. in this study classifying the eight different classes of skin lesions, the proposed model can detect images that do not belong to any one of the eight classes where these images are classified as unknown images. the proposed model gained an accuracy of 96.10% in training and 90.93% during testing. In 2019 Moldovan [28]. in this paper is presented a method for the classification of skin cancer images that consists of two steps, and which is based on transfer learning and deep learning. The classification models are developed in Python using the PyTorch machine learning library and the dataset used as experimental support for testing and validating the transfer learning-based method is Human Against Machine with 10000 training images (HAM10000) dataset. In 2020 Lin et al [29]. in this study they proposed an ensemble CNNs for multi-class classification study based on integration of image pre-process, deep learning and risk management

on skin cancer dermoscopy images with Meta-data. basically, demonstrate general stacking ensemble flow by stages from data pre-processing, CNNs grouping, Meta-data concatenating and 1st CNNs ensemble and 2nd meta-classifiers ensemble. The result showed good performance, especially the contribution from 1st ensemble while they check break-down details. With 5% holdout validation dataset, the reported accuracy maximum 91% in ensemble model, compared with best single CNN (DenseNet121) 87.3%. in 2021 Zhang et al [30]. In this paper, to help dermatologists improve the efficiency of melanoma analysis, they employ the DenseNet model to complete the recognition of melanomas in skin lesion images. The proposed model is trained and evaluated with the ISIC2020 dataset. Besides, Experimental results show method achieves superior performance over the other deep-learning approaches. DenseNet model gains 0.925 with AUC metric, which is higher than approaches with VGG and ResNet backbone. in 2021 Jusman et al [31]. In this paper, they train Multi-layer Perceptron, a custom convolutional neural network, and VGG-16 for skin cancer classification on a large skin cancer dataset, HAM10000. The performance of each trained model is subsequently compared and analysed in terms of classification accuracy and computational time. The experimental setups reveal that the VGG-16 model can set the best classification accuracy among the compared networks while in terms of testing time, the VGG-16 and custom CNN models are being much faster than the Multi-layer Perceptron. The results of study are beneficial in providing systematic comparison and analysis of several neural networks in skin cancer classification. In 2020 Sabri et al [32]. in this paper they propose an improved ensemble learning method to classify skin cancer. Features used are the best combination of extracted features from different characteristics, i.e., shape, colour, texture, and skeleton of the lesion, then they classify these features using different algorithms to predict the classes. Globally, the experimented results show a promoting result. In 2020 Rahat et al [33]. In this paper, DenseNet-121 was used to classify 7 types of skin lesions using the “HAM10000” (Human Against Machine) dataset. Data augmentation was used to make performance of the classifier more efficient. This research can have a positive impact to support dermatologist in the clinic to make more accurate decision in case of skin lesion identification. In 2021 Kondaveeti et al [34]. in this paper they classify seven types of skin cancer. it is an approach where they used initialized weights of pre-trained networks trained on other data resources. final Model based on ResNet50 has achieved a multi-class accuracy of 90. In [35] they explored multiple techniques of building models to accurately classify skin cancer images such as parallel networks and transfer learning and achieved an accuracy of 82.8% using VGG16. In [36] they mimicked the medical strategy and train a deep-learning architecture to perform a hierarchical diagnosis and achieved an accuracy of 87.6% and balanced accuracy of 70% using DenseNet-161. In [37] proposed a method for skin lesion classification and Melanoma detection based on the DenseNet201, a DCNN using transfer learning and achieved Micro-Macro Precision of 87.13%, F1-Score of 87.07% and Micro-Macro ROC AUC of 98.84%. In [38] the research proposed a novel DL with class attention layer-based CAD technique for skin lesion detection and classification known as DLCAL-SLDC with Swallow Swarm Optimization algorithm based Convolutional Sparse

Autoencoder and used CapsNet for classification and achieved 98.5% accuracy, 94.5% sensitivity and 99.1% specificity. In [39] they investigated a pathological course of outlier lesions developing to be melanoma and try to meet the above challenges by proposing a novel neural network based on Efficient-B5 and achieved ROC AUC score of 91.9%. In [40] The proposed method contains three phases localization by tiny YOLOv2 model with ONNX and SqueezeNet model, 13-layer 3D-semantic segmentation model and in the third phase extracting features using ResNet18 then selected by ant colony optimization and the optimized features vectors are passed to O-SVM and O-NB for classification and achieved an accuracy of 97.8% and 99.1%. In [41] they analyzed the benefits of using the transfer learning of pre-trained ImageNet models for medical imaging tasks against a lightweight CNN and used Resnet-50 achieving 99.3% accuracy. In [42] they tested a deep learning approach for a multi-class classification with 8 major diagnostic categories by applying state-of-the-art pre-trained deep convolutional neural networks which can yield a higher diagnostic accuracy compared to dermatologists they used four pre-trained models DenseNet-201, ResNet-152, InceptionV3 and Inception-ResNetV2 with DenseNet-201 achieving the best scores which were Precision Micro-Macro of 89.01-85.24%, F1-score Micro-Macro of 89.01-85.13% and ROC AUC Micro-Macro of 98.79-98.16%. In [43] they created a method which may allow meds to proactively track skin lesions and detect cancer earlier. By creating a novel disease taxonomy, and a disease-partitioning algorithm that maps individual diseases into training classes, they were able to build a deep learning system for automated dermatology with using InceptionV3 and a partition algorithm they made two experiments three way and nine way in the three way they divided the lesions into benign, malignant and non-neoplastic and achieved $72.1 \pm 0.9\%$ accuracy with it and in the nine way they divided the lesions into 9 different classes and achieved an accuracy of $55.4 \pm 1.7\%$ accuracy with it. In [44] this paper proposed a new skin lesion image classification framework based on a skin lesion augmentation style-based GAN (SLA-StyleGAN) according to the basic architecture of style-based GANs and DenseNet201 for classification but instead of using DenseNet201 which is pre-trained on ImageNet it's instead pre-trained on ISIC 2018 and for classification ISIC 2019 were used they achieved an accuracy of 92.5%, balanced multiclass accuracy of 93.64%, Sensitivity of 68.2% and Specificity of 95.6%. In [45] they proposed a decision fusion method. Through transfer learning, based on multiple pre-trained CNNs they used the block to combine multiple CNNs and finally make decisions through multiple blocks with using StyleGANs to generate high-quality images to alleviate the problem of less and uneven distribution of the dermoscopy image dataset and improve the classification effect of CNNs and the best decision fusion achieved an accuracy of 99.5%, AUC of 98.9%, AP of 98.4%, Sensitivity of 98.3% and Specificity of 99.6%. In [46] they proposed a system of classification and detection of skin diseases that can be applied to Tele-dermatology they used MobileNetV1 and InceptionV3 and achieved an accuracy of 72% and of 58% respectively. In [47] their goal was to obtain an effective way for early diagnosis of skin cancer by classifying their dataset images as benign or malignant using InceptionV3 and ResNet-101 they achieved an accuracy of 90% and 89% respectively. In [48] they proposed a

powerful skin-lesion classification approach based on a fusion of handcrafted features (shape, skeleton, color, and texture) and features extracted from most powerful DL architectures they extracted features from AlexNet, GoogLeNet, Vgg16, ResNet18 and passed them to SVM achieving an accuracy of 98% on PH2 Dataset and 87.8% on ISIC 2017. In [49] they researched in the classification of lesion images into Melanoma and Non melanoma using enhanced VGG16-CNN model and achieved an accuracy of 93%. In [50] they proposed a method for capturing a skin sample using a mobile device and assigning it a label depending on its lesion they first used localization, image segmentation then information fusion of both the segmented images by employing the maximum mutual information (MMI) method after that they extracted features from DenseNet-201 and passed the vectors to a multi-class weighted ELM classifier and they tested it on multiple datasets and the best result achieved was 95.26% on ISBI2016. In [51] they proposed a novel framework for dermoscopy image recognition via both a deep learning method and a local descriptor encoding strategy. Specifically, the deep representations of a rescaled dermoscopy image are first extracted via a very deep residual neural network (ResNet-50) pre-trained on a large natural image dataset. Then these local deep descriptors are aggregated by orderless visual statistic features based on fisher vector (FV) encoding to build a global image representation. Finally, the FV encoded representations are used to classify melanoma images using a support vector machine (SVM) with a Chi-squared kernel and achieved an accuracy of 86.54%. In [52] they systematically study melanoma and notice that using deeper, wider and higher resolution convolutional neural networks can obtain better performance. Based on these observations, they propose an automated melanoma detection model by analysis of skin lesion images using EfficientNet-B6 and achieve an accuracy of 91%. In [53] they proposed a new Computer-Aided Detection (CAD) system for the detection and classification of dangerous skin lesions (melanoma type) is presented, During the preprocessing step, a lesion image is enhanced, filtered, and segmented, with the aim to obtain the Region of Interest (ROI); in the next step, the feature extraction is performed. Handcraft features such as shape, color, and texture are used as the representation of the ABCD rule, and deep learning features are extracted using a Convolutional Neural Network (CNN) architecture, which is pre-trained on Imagenet (an ILSVRC Imagenet task). MI measurement is used as a fusion rule, gathering the most important information from both types of features. Finally, at the Classification step, several methods are employed such as Linear Regression (LR), Support Vector Machines (SVMs), and Relevant Vector Machines (RVMs) and achieved an accuracy of 92.4%. In [54] a novel model that relies upon the autoencoder, spiking, and convolutional neural networks is proposed to ensure a useful decision support tool in this study. In the proposed approach, the dataset is reconstructed using the autoencoder model. The original dataset and structured dataset were trained and classified by the MobileNetV2 model that consists of residual blocks, and the spiking networks, the proposed method was able to achieve an accuracy of 95.27%. In [55] they proposed an approach and findings offer a framework for future studies across the spectrum of image-based diagnostics to improve human-computer collaboration in clinical practice. They used 34-layer residual network (ResNet34), a particular type of convolutional

neural network (CNN), on the training dataset of a publicly available image benchmark of pigmented lesions containing seven diagnostic categories and achieved an accuracy of 80.3%. In [56] they proposed a Global-Part Convolutional Neural Network (GP-CNN) model, which treats the fine-grained local information and global context information with equal importance. The Global-Part model consists of a Global Convolutional Neural Network (G-CNN) and a Part Convolutional Neural Network (P-CNN). Specifically, the GCNN is trained with downsampled dermoscopy images, and is used to extract the global-scale information of dermoscopy images and produce the Classification Activation Map (CAM). While the PCNN is trained with the CAM guided cropped image patches and is used to capture local-scale information of skin lesion regions. Additionally, they present a data-transformed ensemble learning strategy, which can further boost the classification performance by integrating the different discriminant information from GPCNNs that are trained with original images, colour constancy transformed images, and feature saliency transformed images, respectively. The proposed method achieved an accuracy of 92.6%. The adoption of a tree structure, with the root node containing all K classes and each sub-node containing the related class information and classifier, is an outstanding



approach. Tree exploration is used to classify test data x , starting at the root node and proceeding until x reaches a leaf node belonging to a specific class. Because of its high classification results and cheap computational complexity, binary tree-based hierarchical multi-class classification using top-down approaches has been extensively accepted (Hernández et al., 2014) [91]. They can be classified using two criteria: a class decomposition approach and a binary classification algorithm that generates a binary tree.

The study On the Grading of Diabetic Retinopathies Using a Binary-Tree-based Multiclass Classifier of CNNs (Mohamed M. Adly., 2019) [92] described an ensemble system based on a criterion obtained from the confusion-matrix values between any two successive DR phases as showed on the Fig.

The overall structure of Mohamed M. Adly ensemble model.

Ana Carolina Lorena and André C.P.L.F. de Carvalho study how metrics of class separability can be

Title	Year of Publication	Dataset(s)	Classifier	Results
Automated Multi-class Classification of Skin Lesions Through Deep Convolutional Neural Network with Dermoscopic Images [86]	2020	ISIC-17, ISIC-18, and ISIC-19	DCNN being named as Classification of Skin Lesions Network (CSLNet) and (t-SNE)	ACC = 92.4 Recall = 48.3 Precision = 97.7
Classification of Skin Disease Using Deep Learning Neural Networks with MobileNet V2 and LSTM [87]	2021	HAM10000	MobileNet V2 and LSTM.	ACC = 85.3 Recall = 88.2 Precision = 92
Skin Cancer Disease Images Classification Using Deep Learning Solutions [88]	2021	ISIC-19, and ISIC-20	InceptionV3	Recall = 86.1 Precision = 87.7
Skin Lesion Classification using Convolution Neural Network with Novel Regularizer [89]	2019	ISIC 2019	CNN with a novel regularizer that classifies skin lesions into benign or malignant	ACC = 97.5 Recall = 94.3 Precision = 93.6
Deep Convolutional Neural Network VGG-16 Model for Differential Diagnosing of Papillary Thyroid Carcinomas in Cytological Images: A Pilot Study [90]	2019	Pathology-Proven Dataset	VGG-16	ACC = 86 Recall = 86 Precision = 100

used in the building of binary-tree-based multiclass classifiers, adjusting the decomposition to the specific multiclass problem[93].

A Summary of The Literature Review: Recent Publications (From 2019)

Chapter 3: Dataset

The well-known dataset from the ISIC 2019 challenge was used to test and evaluate the proposed method. The ISIC 2019 dataset contains images of HAM10000 and the BCN_20000 and MSK dataset. HAM10000 contains 10000 images with a size of 600×450 . While the BCN_20000 contains 19424 images of size 1024×1024 . And MSK contains 12500 images with different sizes in range 1024×720 to 6600×4400 . The benign classes are Melanocytic nevi (nv, 12875 images), Benign keratosis-like lesions (bkl, 2624 images), Actinic keratoses and intraepithelial Carcinoma (ak, 867 images), Squamous cell carcinoma (scc, 628 images), Vascular lesions (vasc, 253 images) and Dermatofibroma (df, 239 images). Whereas the malignant are Melanoma (mel, 4522 images) and Basal cell carcinoma (bcc, 3323 images). the dataset classes distribution as shown in Fig. 1 And dataset classes sample in Fig. 2.

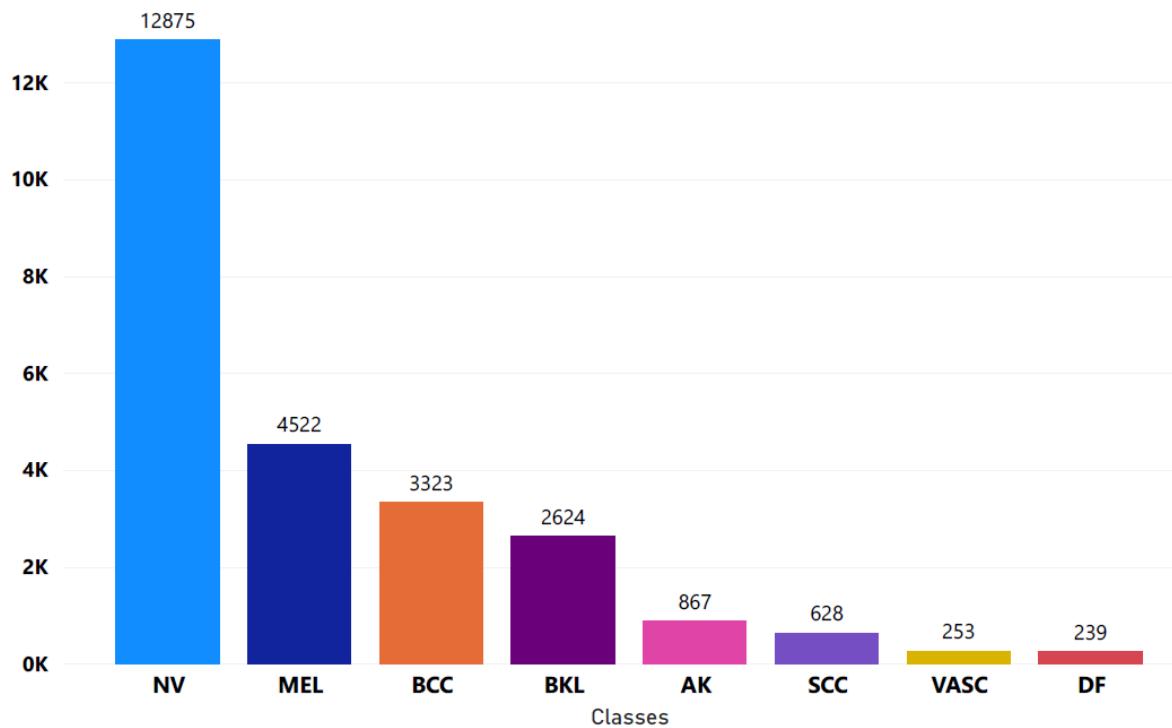


Fig. 1. Classes of ISIC 2019 Dataset

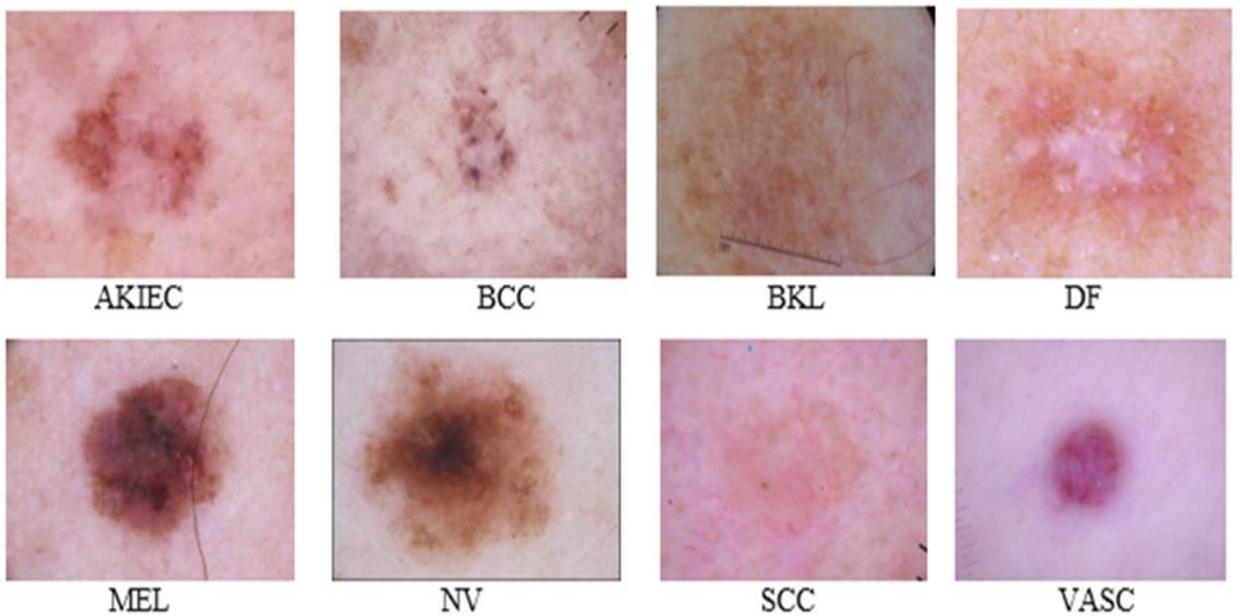


Fig. 2. Sample of images from each class

Chapter 4: Pre-Processing

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model. When creating a machine learning project, it is not always a case that we come across the clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put in a formatted way. A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models. Data preprocessing is required tasks for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model. We Trained DenseNet201 and ResNet50 using transfer learning on the whole dataset with different image sizes (32*32, 64*64, 224*224) and the images are preprocessed by different techniques (Hair Removal, Normalization, Edge Fining) and several combinations between them.

4.1 Image size

CNNs usually use fixed size and square images as input, and different size images can get different feature extraction effect. All images are cropped to the desired size before entering the CNN, but most cropping causes skin lesions to deform, and shape contour information is an important basis for discriminating skin cell damage classes. Relatively complete dermoscopy images are helpful for CNNs to extract features. However, if the image is large, the time and computational cost of training CNNs is high. We try to pre-process and crop the image without affecting the recognition target and adjust the image size with the centre as the origin. By comparing three sizes of dermoscopy images, we choose a size of 32x32, 64x64 and 224x224 as

input in the pre-processing Comparison and used 224x224 as input in the second approach in the pre trained comparison and ensemble.

4.2 Augmentation

augmentation is performed within the Data generator Kera's function. The used parameters are as follow; randomly flip images horizontally and vertically are allowed. The rotation range is 90, the zooming range is 0.1 and both width and height shift range are 0.1

4.3 Normalization

It is the most important Data Transformation technique widely used. The numerical attributes are scaled up or down to fit within a specified range. In this approach, we are constraining our data attribute to a particular container to develop a correlation among different data points. Normalization can be done in multiple ways, we used two types of Normalization, pixel normalization and Z-Score normalization (Standardization).

We first used pixel normalization for most image data, the pixel values are integers with values between 0 and 255. Neural networks process inputs using small weight values, and inputs with large integer values can disrupt or slow down the learning process. As such it is good practice to normalize the pixel values so that each pixel value has a value between 0 and 1. It is valid for images to have pixel values in the range 0-1 and images can be viewed normally. This can be achieved by dividing all pixel's values by the largest pixel value; that is 255. This is performed across all channels, regardless of the actual range of pixel values that are present in the image.

Pixel Normalization Formula:

$$x' = \frac{(x - x_{\min})255}{x_{\max} - x_{\min}}$$

Secondly, we used Z-Score normalization which is a strategy of normalizing data that avoids this outlier issue. In this technique, values are normalized based on the mean and standard deviation of the data. The essence of this technique is the data transformation by the conversion of the values to a common scale where an average number/mean equals zero and a standard deviation is one. However, we didn't use the standard deviation and mean of our dataset instead used those of ImageNet which are mean (0.485, 0.456, 0.406) and std (0.229, 0.224, 0.225) because it's highly advised to use with models that are pre-trained on ImageNet.

Z-score Formula:

$$z = \frac{x - \mu}{\sigma}$$

4.4 Hair removal

There have been numerous advancements towards utilizing deep networks, ANNs, AI, etc. in tasks like detecting the skin disease, type of tumor, etc. However, it becomes difficult for the networks to learn the features since, most of the skin images are occluded by hair. Thus, there is a need for pre-processing of the skin images to remove this obstructing hair, so this step-in pre-processing aims to remove the hair noise from the skin image with the help of Morphological filtering.

1. Convert the colour image to a grayscale version.
2. Applying Morphological Black-Hat transformation on the grayscale image
3. Creating the mask for InPainting task
4. Applying inpainting algorithm on the original image using the mask prepared from the grayscale image in step 3

1-Convert the image to grayscale version:

Consider a colour image, given by its red, green, blue components R, G, B. The range of pixel values is often 0 to 255. Colour images are represented as multi-dimensional arrays - a collection of three two-dimensional arrays, one each for red, green, and blue channels. Each one has one value per pixel and their ranges are identical. For grayscale images, the result is a two-dimensional array with the number of rows and columns equal to the number of pixel rows and columns in the image, this step is done by using the OpenCv2 [57] library in python. Fig.3 and Fig.4 shows the original and the gray scale image



Fig.3

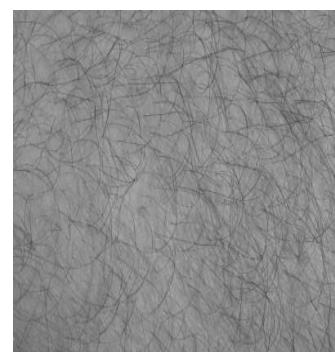


Fig.4

2-Applying Morphological Black-Hat transformation on the grayscale image:

2.1 Morphological Transformations:

Morphological transformations are some simple operations based on the image shape [58]. It is normally performed on grayscale images. It needs two inputs, one is our original image, second one is called **structuring element** or **kernel** which decides the nature of operation, in skin lesions case the structuring element selected is cross struct element with size (17,17)

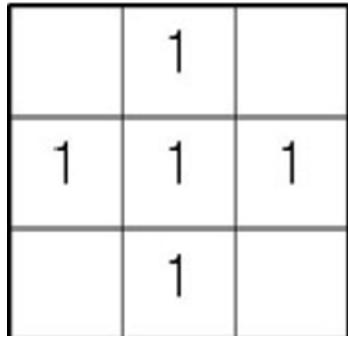


Fig.5 example on 3*3 Cross Structuring Element

Two basic morphological operators are Erosion and Dilation. Then its variant forms like Opening, Closing, Gradient etc. also comes into play

2.2 Erosion

The basic idea of erosion is just like soil erosion only, it erodes away the boundaries of foreground object (Always try to keep foreground in white). So, what it does? The kernel slides through the image (as in 2D convolution). A pixel in the original image (either 1 or 0) will be considered 1 only if all the pixels under the kernel is 1, otherwise it is eroded (made to zero). So, what happened is that all the pixels near boundary will be discarded depending upon the size of kernel. So, the thickness or size of the foreground object decreases, or simply white region decreases in the image. It is useful for removing small white noises (as we have seen in colour space chapter), detach two connected objects etc.

2.3 Dilation

It is just opposite of erosion. Here, a pixel element is '1' if at least one pixel under the kernel is '1'. So, it increases the white region in the image or size of foreground object increases. Normally, in cases like noise removal, erosion is followed by dilation. Because, erosion removes white noises, but it also shrinks our object. So, we dilate it. Since noise is gone, they won't come back, but our object area increases. It is also useful in joining broken parts of an object.

2.4 Closing

Closing is reverse of Opening, **Dilation followed by Erosion**. It is useful in closing small holes inside the foreground objects, or small black points on the object.

2.5 Black Hat

Simply It is the difference between the closing of the input image and input image. Fig.6 shows the image after applying the black Hat transformation

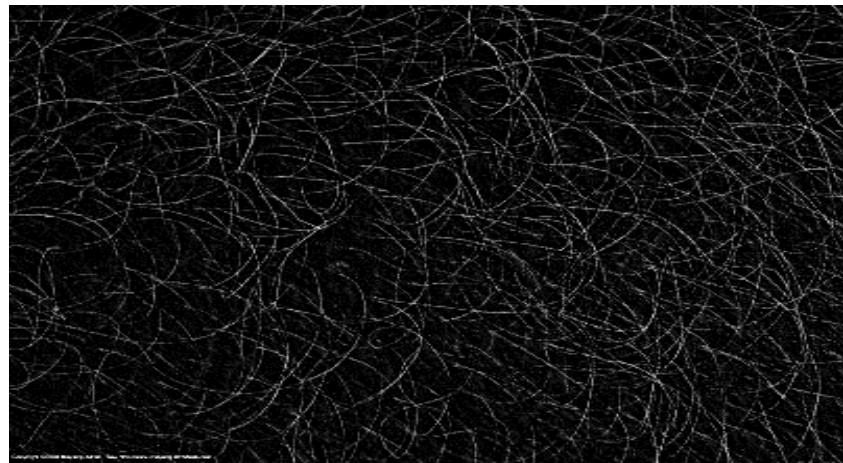


Fig.6 BlackHat filtering result

3- Creating the mask for InPainting task

The inpainting mask can be done by using the “cv.THRESH_BINARY” [59]with specified threshold values for every pixel, the same threshold value is applied. If the pixel value is smaller than the threshold, it is set to 0, otherwise it is set to a maximum value The first argument is the source image, which **should be a grayscale image**. The second argument is the threshold value which is used to classify the pixel values. The third argument is the maximum value which is assigned to pixel values exceeding the threshold. Fig.7 shows the the thresholded image for the inpainting step

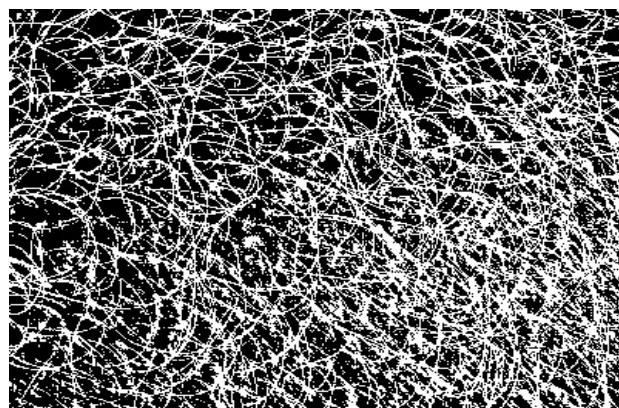


Fig.7 Thresholded image

4- Applying inpainting algorithm on the original image using the mask

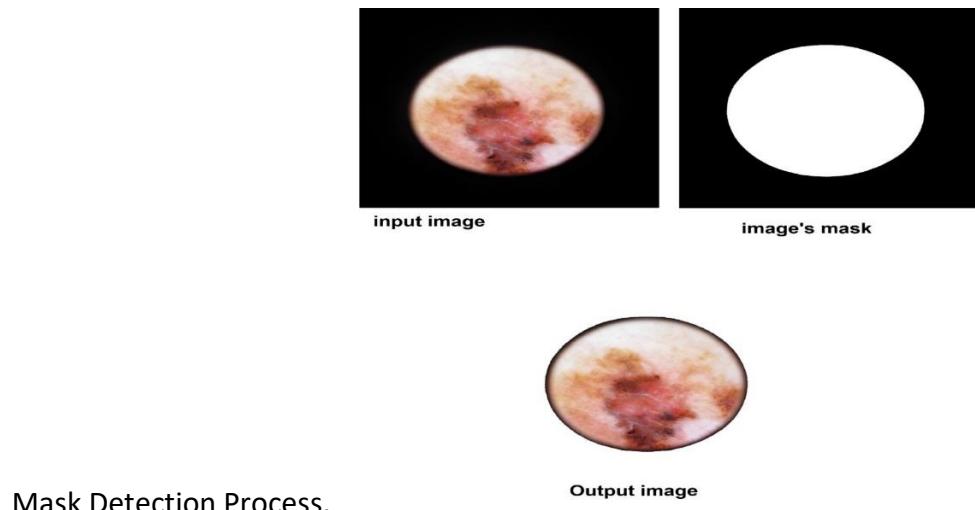
An Image Inpainting Technique Based on the Fast-Marching Method*** by Alexandru Telea in 2004[60]. It is based on Fast Marching Method. Consider a region in the image to be inpainted. Algorithm starts from the boundary of this region and goes inside the region gradually filling everything in the boundary first. It takes a small neighborhood around the pixel on the neighborhood to be inpainted. This pixel is replaced by normalized weighted sum of all the known pixels in the neighborhood [61]. Fig.8 shows the final image



Fig.8 (final result after inpainting)

4.5 Black Edge Removal

In this section, we aim to remove the black edge part that surrounds the skin, as shown in the following Fig, by using thresholding, which produces a binary image in which all pixels with intensities above or below the thresholding value are turned on while the other pixels are turned off, followed by an open operation that removes small objects and lines by eroding then dilating the image using the same structure element, and finally a close operation to fill small gaps. The resulting picture is known as the mask, and it indicates the ROI portion of the image (i.e., the skin). This algorithm's last stage is cropping the image depending on the external contour region



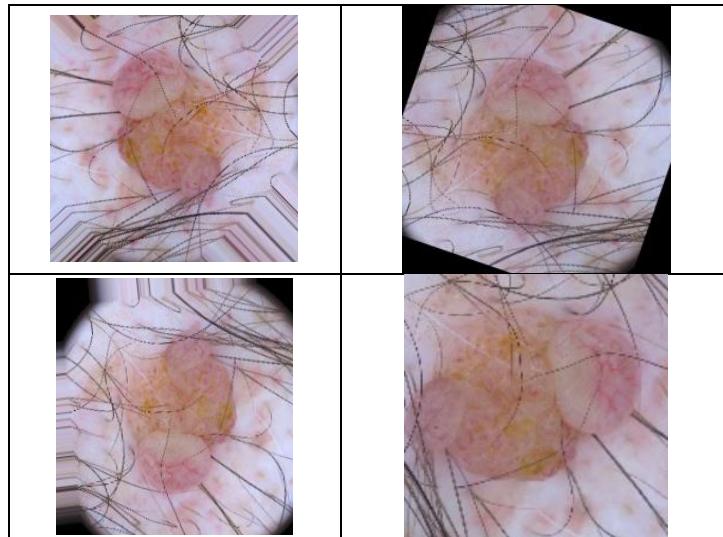
Mask Detection Process.

4.6 Manual Augmentation

Our manual augmentation method was done manually written and applied before training process and it includes brightness changing in range (0.7 and 1), and rotation between (50 and 270), Vertical flip, horizontal flip, zooming by .15 and combinations of them.

Comparing manual augmentation Operations to Data-generator Augmentation Operations

Data-generator Images	Manual implemented augmentation methods



Chapter 5: Pre-Trained Models

5.1 Background

Before discussing how different pretrained models work and the other methods we mentioned earlier let us take a look about the very basic components and calculations of Convolutional Neural Networks (CNN)

- Convolutional
- Pooling
- Flattening
- Full Connection

5.1.1 Convolutional

The first operation, Convolutional, which is a step to extract the important features from the image by performing matrix multiplication between two matrices the image matrix and the kernel or the filter matrix and finally the result of this operation is called a feature map.

Different kernel sizes according to your problem of classification and the type of the images, this filter is traversed through the whole image matrix with fixed step size which is called stride and multiply the filter values with the pixels values to get the feature map as a result

Fig.9 shows the Convolutional process

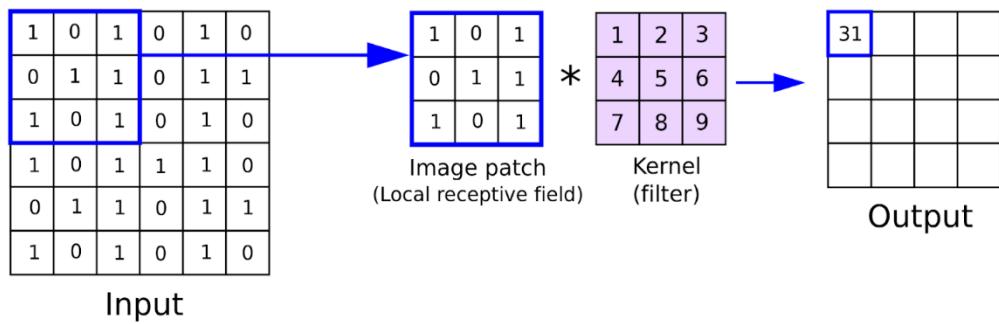


Fig.9 Convolutional operations

5.1.2 Pooling

Pooling operation comes after the convolutional operation to reduce the number of parameters, in other word it reduces the dimensionality of each feature map by keeping only significant information.

Pooling has two main types

- Max Pooling
- Average Pooling

Max Pooling: Max pooling selects the brighter pixels from the image which means the pixels with the greatest value in feature map.

Average Pooling: Pooling with the average values. as the name suggests, it retains the **average values** of features of the feature map, Fig.10 shows max pooling operation

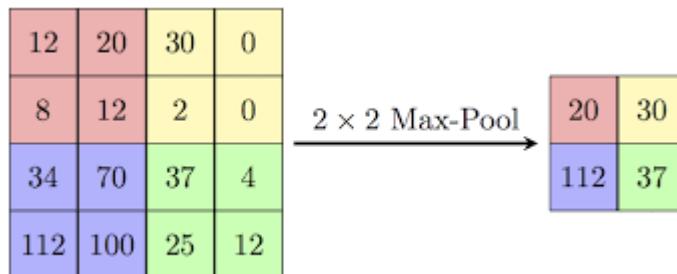


Fig.10 Max Pooling Operation

5.1.3 Flattening

To feed the features that were generated from the previous step, then we need to convert the matrices into one vector Fig11 (shows the flattening operation after pooling operation)

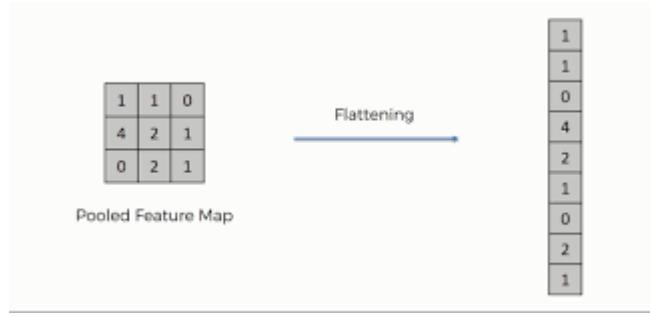


Fig.11 flattening operation

5.1.4 Full Connection

The fully connection layer takes the input from the preceding convolution/pooling layer and produces an N dimensional vector where N is the number of classes to be classified. Thus, the layer determines the features most correlating to a particular class based on the probabilities of the neurons, Fig.12 shows the fully connected layer in a neural network.

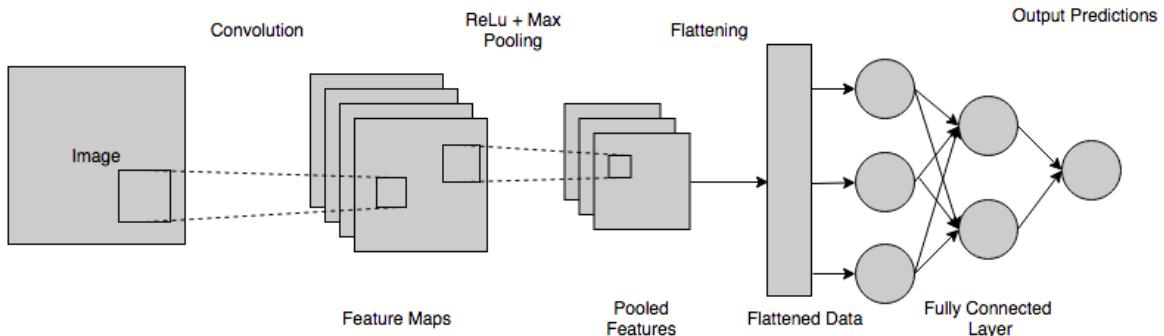


Fig12 (Fully connected layer with other layers)

5.2 Transfer Learning

Transfer learning [62] is a method for feature representation from a pre-trained model facilitating us that we don't need to train a new model from scratch. A pre-trained model is usually trained on a huge dataset such as ImageNet and the weights obtained from the trained model can be used for any other related application with your custom neural network. These newly built models can directly be used for predictions on relatively new tasks or can be used in training processes for related applications. This approach not only reduces the training time but also lowers the generalization error.

5.2.1 Transfer Learning Definition

The definition of transfer learning is given in term of domains and A domain \mathcal{D} Consists of a: feature space X and a marginal probability distribution $P(X)$, where $X = x_1, \dots, x_n \in X$. Given a specific

domain, $\mathcal{D} = \{X, P(X)\}$ a task consists of two components: a label space Y and an objective prediction function $f : X \rightarrow y$.

The function f is used to predict the corresponding label $f(x)$ of a new instance x . This task is denoted by $T = \{y, f(x)\}$, is learned from the training data consisting of pairs $\{x_i, y_i\}$, where $x_i \in X, y_i \in Y$

Given a source domain D_s and learning task T_s , a target domain D_t and a learning task T_t , where $D_s \neq D_t$ or $T_s \neq T_t$, transfer learning aims to help improve the learning of the target predictive function $F_t(\cdot)$ in D_t using the knowledge in D_s and D_t . Fig.13 (shows example on TL process)

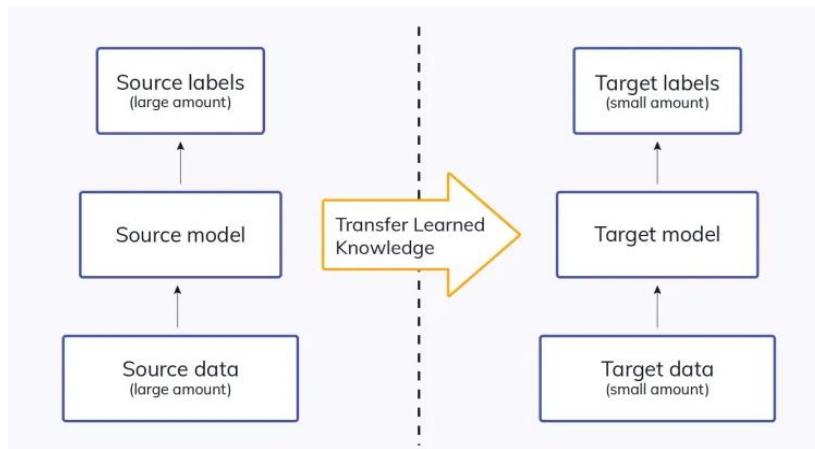


Fig.13 TL Overview

5.2.2 Basic Applications of Transfer Learning

Some of the common applications of transfer learning can be seen as pre-trained models trained on ImageNet can be used for real-world image-based classification problems; this is achieved because the model is trained on 1000 classes. Let's say you are developing software that predicts a kind of flower, there you can use a pre-pre-trained model to predict the kind of flower.

5.2.3 Proposed method using Transfer Learning

In this project, the CNN is trained by transfer learning from the pretrained weight of ImageNet classification. ImageNet Large Scale Visual Recognition Challenge (ILSVRC) is the classification challenge involving 14 million training images of approximately 20,000 classes [64].

The model is trained and tested on the state-of-art CNNs [65], namely DenseNet201, ResNet50, InceptionV3, EfficientNetV2M, EfficientNetV2S, EfficientB0, VGG19, MobileNet, ResNet152, Xception and NASNetMobile to perform the eight-class classification of the skin lesion images.

The proposed method is done on several steps

- Instantiate a base model and load pre-trained weights into it.
- Run your new dataset through it and record the output from the base model. This is called **feature extraction**.
- Use that output as input data for a new, smaller model, and this is done by omitting the classification layer in the original model which is designed to classify ImageNet classes which is 1000 class and is replaced by a global average layer followed by flattening and SoftMax activation layer with eight nodes which the new or the smaller model fig.14 example on ResNet50.

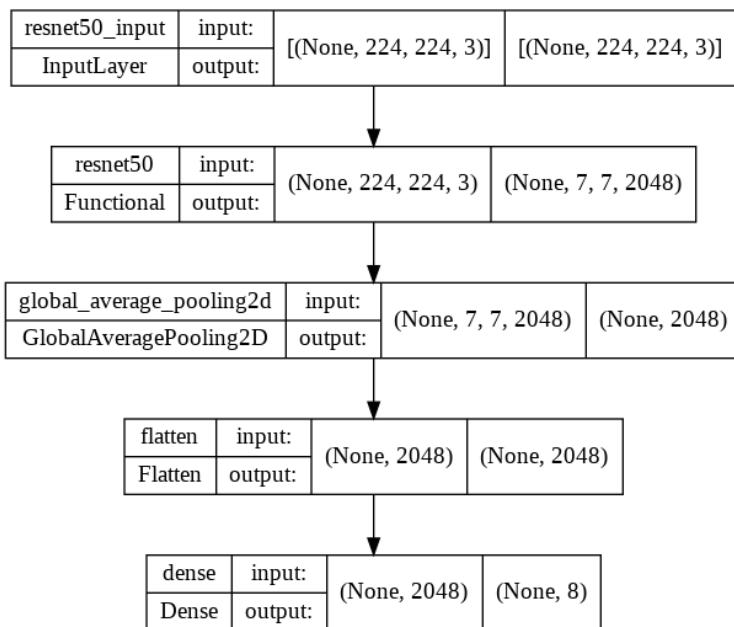


Fig.14 ResNet50 architecture after modifying

5.3 Network Architecture:

5.3.1 ResNet50 (Detailed Explanation in Appendix):

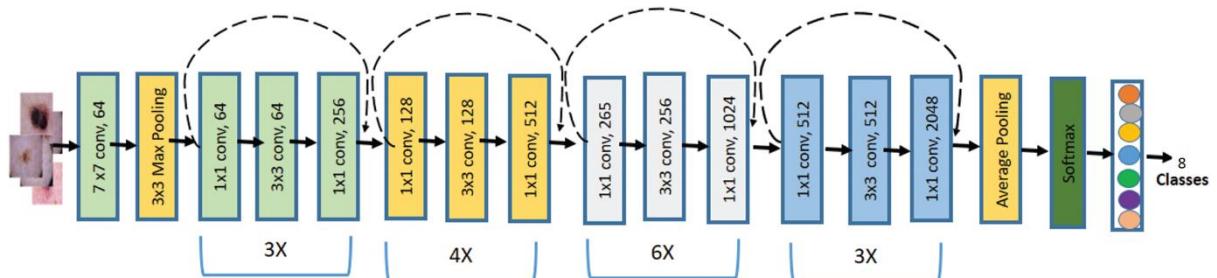


Fig.15 shows ResNet50 architecture

Residual Network (ResNet) is a typical model of neural network used as an integral part for various computer vision tasks. This ResNet model won the 2015 ImageNet challenge. The network is 50 layers deep and takes the input image size of 224×224 pixels [66]. Generally, in a deep CNN, many layers are stacked and trained. The network learns many low levels, middle and high-level features. In residual learning, rather than learning some features, we learn some residual.

Residual can be simply interpreted as reduction of features learned from input layer. It has been shown that training residual networks is easier compared to training simple deep CNNs. It also helps to tackle the problem of degrading accuracy.

Resnet-50 architecture contains five phases, each phase contains two blocks where the first one is a convolution block, and the second block is an identity block. Each convolution block consists of three convolution layers also each identity block has three convolution layers. In the end, there is a fully connected layer and a classification output layer which are replaced with three new layers for two classes when transfer learning is applied. Fig. 4 shows the diagram of the updated Resnet-50 with transfer learning. In our proposal ResNet was trained on the whole dataset pre-processed with edge fining algorithm, with both Hair removal and edge fining, and with hair removal only.

5.3.2 DenseNet201(Detailed Explanation in Appendix):

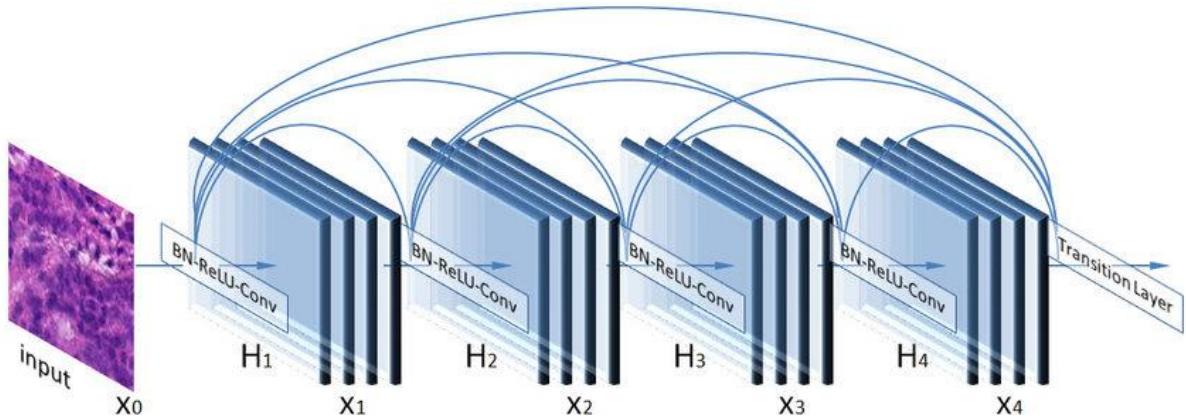


Fig.16 DenseNet Architecture

Dense Convolutional Network (DenseNet) [67] is connecting each layer to every other layer in a feed-forward fashion. They alleviate the vanishing-gradient problem, strengthen feature propagation, encourage feature reuse, and substantially reduce the number of parameters. DenseNet works on the idea that convolutional networks can be substantially deeper, more accurate, and efficient to train if they have shorter connections between layers close to the input and those close to the output. Fig.16 shows DenseNet Architecture

DenseNet201 Architecture contains the following

- Basic convolution layer with 64 filters of size 7X7 and a stride of 2
- Basic pooling layer with 3x3 max pooling and a stride of 2
- Dense Block 1 with 2 convolutions repeated 6 times
- Transition layer 1 (1 Conv + 1 AvgPool)
- Dense Block 2 with 2 convolutions repeated 12 times
- Transition layer 2 (1 Conv + 1 AvgPool)
- Dense Block 3 with 2 convolutions repeated 48 times
- Transition layer 3 (1 Conv + 1 AvgPool)
- Dense Block 4 with 2 convolutions repeated 32 times
- Global Average Pooling layer- accepts all the feature maps of the network to perform classification
- Output layer which will be in our case only 8 classes instead of 1000

In our proposal DenseNet was trained on four times

- On images without any pre-processing.
- On images pre-processed with Edge finning algorithm.
- On images pre-processed with hair removal algorithm.
- On images pre-processed with hair removal and edge finning algorithms

5.3.3 InceptionV3(Detailed Explanation in Appendix):

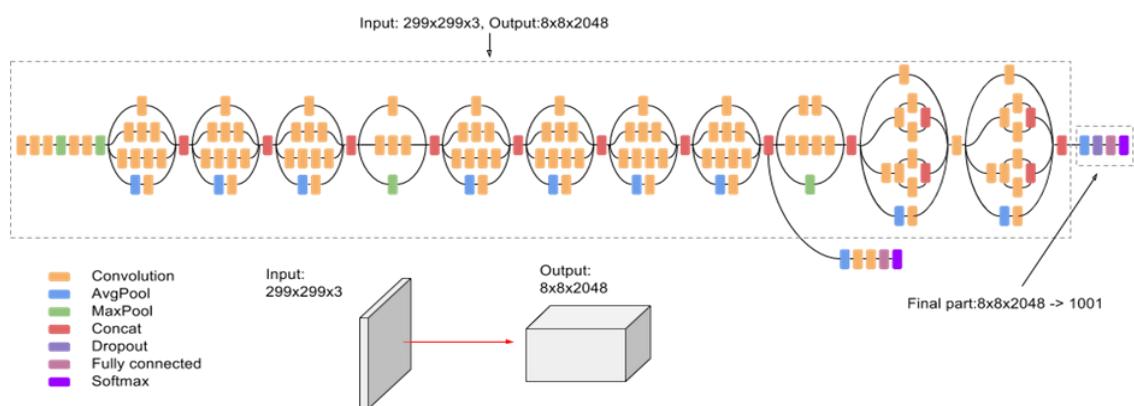


Fig.17

Inception v3 is an image recognition model that has been shown to attain greater than 78.1% accuracy on the ImageNet dataset. The model is the culmination of many ideas developed by multiple researchers over the years. It is based on the original paper [68]

The model itself is made up of symmetric and asymmetric building blocks, including convolutions, average pooling, max pooling, concatenations, dropouts, and fully connected layers. Batch normalization is used extensively throughout the model and applied to activation inputs. Loss is computed using SoftMax. The Inception-v3 is composed of a 42-layer deep neural network. Fig.17 shows overview on Inception V3 architecture

Notice that the input layer is modified to take images with size 224*224, InceptionV3 was trained on images pre-processed with edge finning

5.3.4 MobileNet:

As the name applied, the MobileNet model is designed to be used in mobile applications, and it is TensorFlow's first mobile computer vision model. MobileNets are based on the principle of streamlined architectures which use depth-wise separable convolutions followed by point-wise convolutions that considerably reduce the number of learnable parameters and assist in building lightweight deep neural networks [69].

Effectively, it decreases the total number of floating point calculations required which is supportive for embedded and mobile computer vision applications where there is shortage of computational power. The architecture was proposed by Google, MobileNet Model was trained on images pre-processed with edge finning algorithm.

Fig.18 shows the depth-wise and point-wise convolution operations.

Fig.19 depicts the architecture of the MobileNet CNN model.

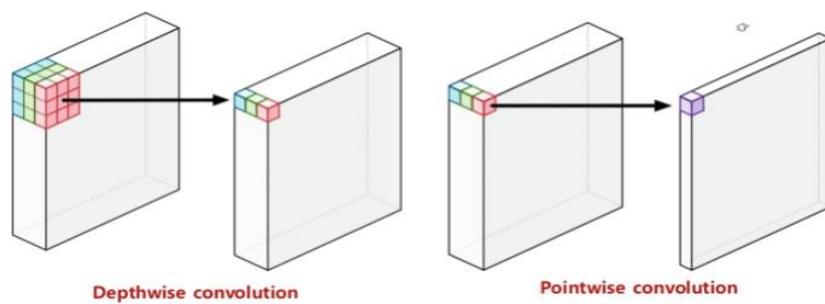


Fig.18

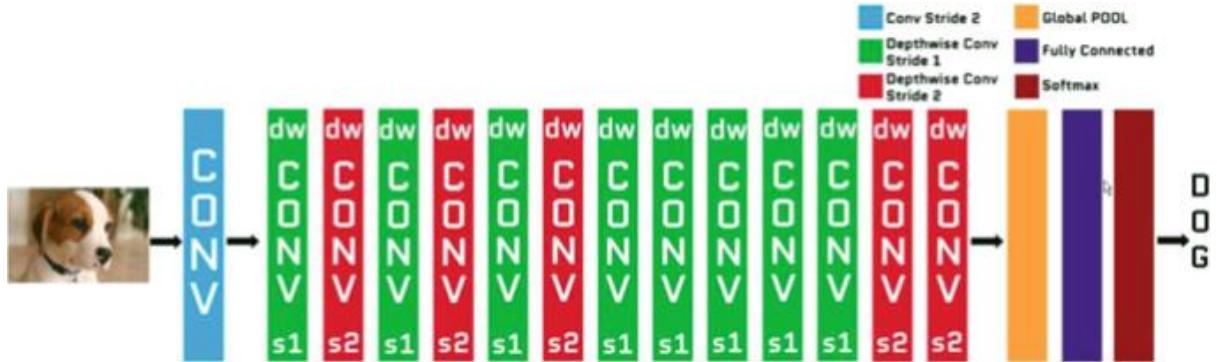


Fig.19

5.3.5 EfficientNetB0:

In [70] they proposed a novel model scaling method that uses a simple yet highly effective *compound coefficient* to scale up CNNs in a more structured manner. Unlike conventional approaches that arbitrarily scale network dimensions, such as width, depth and resolution, their method uniformly scales each dimension with a fixed set of scaling coefficients. The effectiveness of model scaling also relies heavily on the baseline network. So, to further improve performance, we have also developed a new baseline network by performing a neural architecture search using the AutoML MNAS framework, which optimizes both accuracy and efficiency (FLOPS). The resulting architecture uses mobile inverted bottleneck convolution (MBCConv), similar to MobileNetV2 [71] and MnasNet [72] but is slightly larger due to an increased FLOP budget. We then scale up the baseline network to obtain a family of models, called *EfficientNets*.

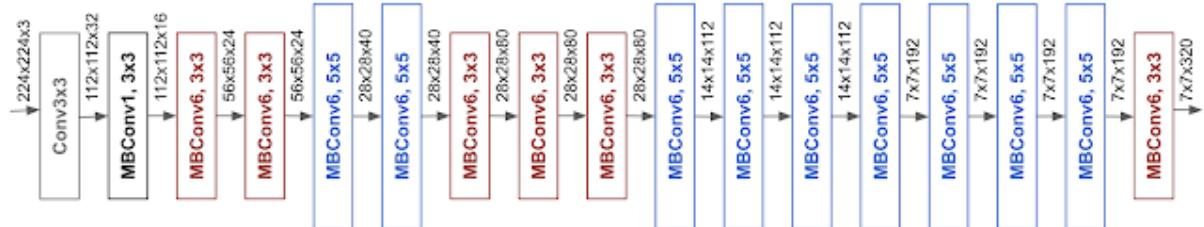


Fig.20

We have compared our EfficientNet with other existing CNNs on ImageNet [64]. In general, the EfficientNet models achieve both higher accuracy and better efficiency over existing CNNs, reducing parameter size and FLOPS by an order of magnitude. For example, in the high-accuracy regime, our EfficientNet-B0 reaches state-of-the-art 77.1% top-1 and 93.3% top-5 accuracy on ImageNet. The model was trained on images pre-processed with edge fining algorithm.

5.3.6 EfficientNetV2-S/M:

In [73] they designed a search space enriched with additional ops such as Fused-MBConv and apply training-aware NAS and scaling to jointly optimize model accuracy, training speed, and parameter size.

their found networks, named EfficientNetV2, train up to 4x faster than prior models, while being up to 6.8x smaller in parameter size.

The architecture for their searched model EfficientNetV2-S. Compared to the EfficientNet backbone, their searched EfficientNetV2 has several major distinctions: The first difference is EfficientNetV2 extensively uses both MBConv [70] and the newly added fused-MBConv [74] in the early layers. Secondly, EfficientNetV2 prefers smaller expansion ratio for MBConv since smaller expansion ratios tend to have less memory access overhead. Thirdly, EfficientNetV2 prefers smaller 3x3 kernel sizes, but it adds more layers to compensate the reduced receptive field resulted from the smaller kernel size. Lastly, EfficientNetV2 completely removes the last stride-1 stage in the original EfficientNet, perhaps due to its large parameter size and memory access overhead.

they scale up EfficientNetV2-S to obtain EfficientNetV2-M/L using similar compound scaling as [70], with a few additional optimizations: First they restrict the maximum inference image size to 480, as very large images often lead to expensive memory and training speed overhead; Second as a heuristic, they also gradually add more layers to later stages (e.g., stage 5 and 6) to increase the network capacity without adding much runtime overhead.

they proposed an improved method of progressive learning: in the early training epochs, they train the network with small image size and weak regularization (e.g., dropout and data augmentation), then they gradually increase image size and add stronger regularization. Built upon progressive resizing, but by dynamically adjusting regularization, their approach can speed up the training without causing accuracy drop. With the improved progressive learning, their EfficientNetV2 achieves strong results on ImageNet, CIFAR-10, CIFAR100, Cars, and Flowers dataset. On ImageNet, they achieved 85.7% top-1 accuracy while training 3x - 9x faster and being up to 6.8x smaller than previous models, both models V2M and V2S are trained on images pre-processed with edge fining algorithm.

5.3.7 Xception:

Xception is a deep convolutional neural network architecture that involves depthwise Separable Convolutions. It was developed by Google researchers. Google presented an interpretation of Inception modules in convolutional neural networks as being an intermediate step in-between regular convolution and the depthwise separable convolution operation (a depthwise convolution followed by a pointwise convolution). In this light, a depthwise separable convolution can be understood as an Inception module with a maximally large number of towers. This observation leads them to propose a novel deep convolutional neural network architecture inspired by Inception, where Inception modules have been replaced with depthwise separable convolutions [75].

They showed that this architecture, dubbed Xception, slightly outperforms Inception V3 on the ImageNet dataset (which Inception V3 was designed for), and significantly outperforms Inception V3

on a larger image classification dataset comprising 350 million images and 17,000 classes. Since the Xception architecture has the same number of parameters as Inception V3, the performance gains are not due to increased capacity but rather to a more efficient use of model parameters [76], Fig.21 shows the detailed architecture of the original Xception Module.

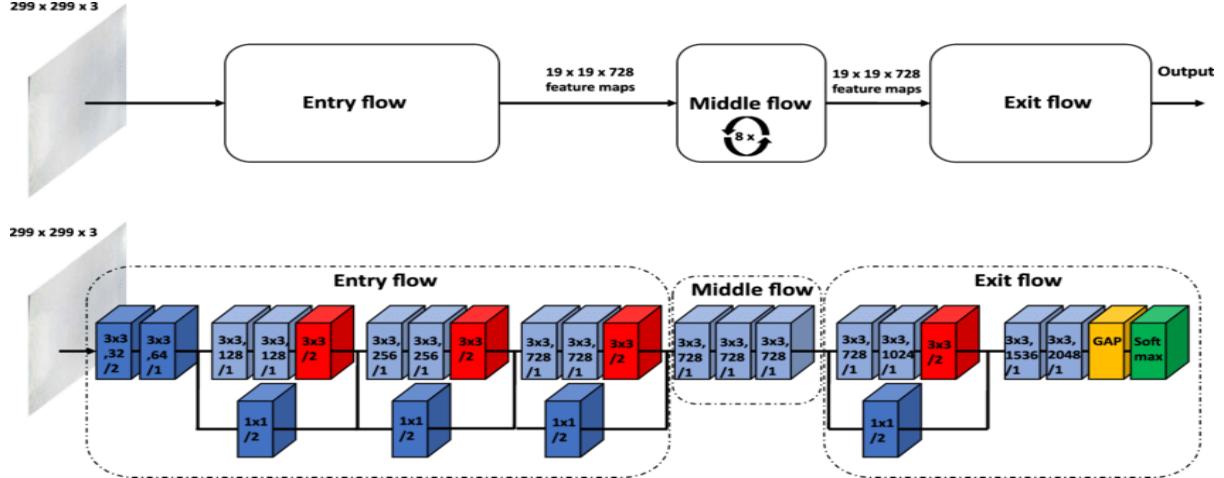


Fig.21 Xception Modules

5.3.8 VGG-19:

VGG-19 is a convolutional neural network that is trained on more than a million images from the ImageNet database. The network is 19 layers deep and can classify images into 1000 object categories, such as a keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images.

VGG19 is a variant of the VGG model which in short consists of 19 layers (16 convolution layers, 3 Fully connected layer, 5 MaxPool layers and 1 SoftMax layer). There are other variants of VGG like VGG11, VGG16 and others. VGG19 has 19.6 billion FLOPs.

The layers with trainable weights are only the convolution layers and the Fully Connected layers. Maxpool layer is used to reduce the size of the input image where SoftMax is used to make the final decision [77].

In simple language, VGG is a deep CNN used to classify images. The layers in the VGG19 model are as follows Fig.22:

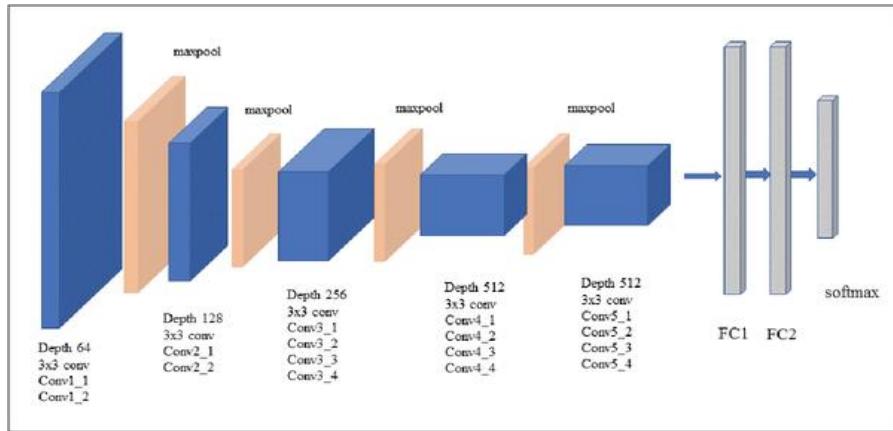


Fig. 3. VGG-19 network architecture

Fig.22 VGG-19 Architecture

Chapter 6: Ensemble Learning

6.1 Ensemble Learning:

In our study we did the following in the ensemble approach:

1. Choose the best models among those models according to their performance.
2. Combine all the predictions of those models using Ensemble methods.
3. Ensemble the models with the highest classification accuracy (above 85%).
4. Ensemble the models above 85% classification accuracy.
5. Ensemble Rest of the models.

Ensemble learning refers to the procedures employed to train multiple learning machines and combine their outputs, treating them as a “**committee**” of decision makers [78]. The principle is that the decision of the committee, with individual predictions combined appropriately, should have better overall accuracy, on average, than any individual committee member. Numerous empirical and theoretical studies have demonstrated that ensemble models very often attain higher accuracy than single models.

The members of the ensemble might be predicting real-valued numbers, class labels, posterior probabilities, rankings, clusterings, or any other quantity. Therefore, their decisions can be combined by many methods, including averaging, voting, and probabilistic methods. The majority of ensemble learning methods are generic, applicable across broad classes of model types and learning tasks, in the proposed solution the adopted method of ensemble is the weighted soft voting classifier.

6.1.1 Weighted Soft Voting:

In weighted soft voting [79], we predict the class labels based on the predicted probabilities P on the test set for the different classifiers previously stated using the output from the SoftMax layer, assign weight to each classifier and calculate the final prediction using this formula:

$$\hat{y} = \text{argmax}_i \sum_{j=1}^m w_j p_{ij},$$

where w_j is the weight that can be assigned to the j th classifier, the assigned weights are initialized with one and then calculated for each classifier using the “Nelder-Mead” minimizing function. Fig.23 shows an overview about the majority voting.

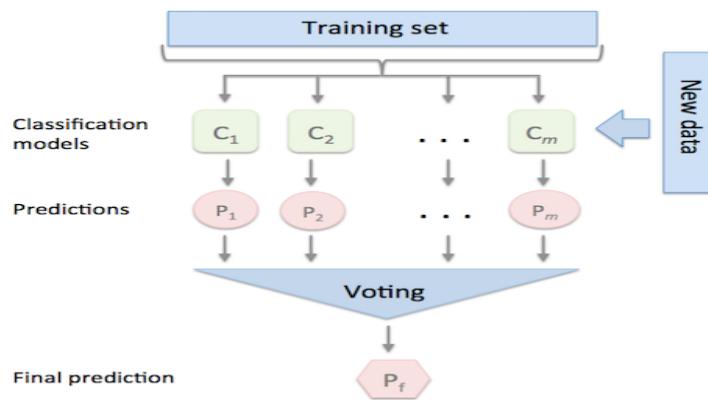


Fig.23 Majority Voting

6.1.2 Nelder-Mead minimizing function:

The Nelder–Mead technique was proposed by John Nelder and Roger Mead in 1965 [80], as a development of the method of Spendley et al.[81] It is one of the best algorithms that can be used to solve parameter estimation problems, and statistical problems.

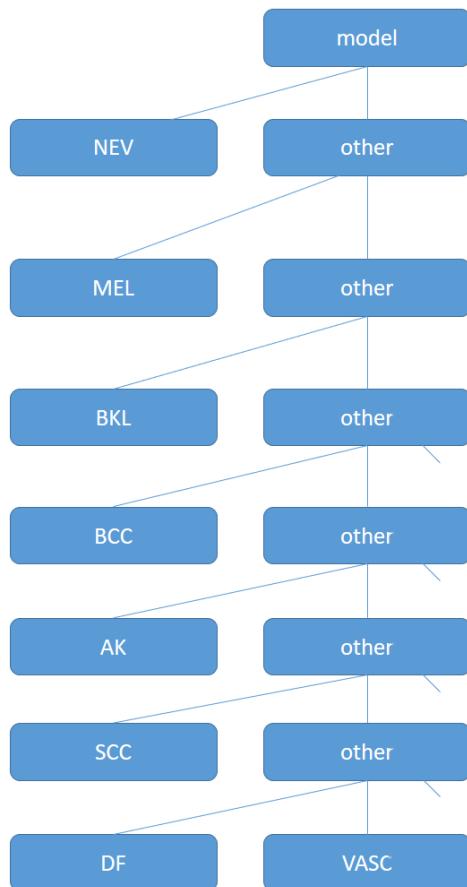
This algorithm can also be used to work with discontinuous functions which occur frequently in statistics. It is a simple algorithm, and it is easy to understand as well. Used to minimize the parameters of a non-linear function in case of a multidimensional unconstrained optimization.

Chapter 7: Binary tree-based multi-classifier

To design our model, we leveraged the architecture of the DenseNet201 model to create a multi-classifier for all classes. The confusion matrix was then created in order to detect the error ratio of the classes in our situation, as we utilized the ISIC 2019 Dataset, which has 8 classes, and we discovered the following class ratio order: 1-Nevus 2-Mel 3-BKL 4-BCC 5-AK 6-SCC 7-VASC 8-DF.

We used these classes to create multi-classifiers using the One-Vs-All technique as shown in model's pseudo-code

As mentioned above we applied the augmentation methods explained to implement our binary models and multi-classification models



Our proposed tree structure.

```

2
3     input: test data, k-1 binary models, label appendarray
4
5     for data in test data
6         predict_0(data)
7             check if image is the target type
8                 append target label to array
9             else
10                predict_1(data)
11                check if image is the target type
12                    append target label to array
13                else
14                    predict_2(data)
15                    ....
16
17                else
18                    predict_k-1(data)
19                    check if image is one of the last two classes
20                        append the class label to array
21                    else
22                        append the other class label to array
23 End of code..
24
25
26

```

Pseudo-Code For our Binary tree based Model's Implementation

Chapter 8: Implementation & Results

In the first 3 approaches the hyperparameters are the same as in Table.1, the learning rate was scheduled to decay over the training process using the cosine decay scheduler [82], augmentation is performed within the Data generator Keras function [83]. The used parameters are as follow; randomly flip images horizontally and vertically are allowed. The rotation range is 90, the zooming range is 0.1 and both width and height shift range are 0.1 and fill-mode is set to nearest. Dataset is split into 80% train (20264 image) , 10% validation (2534 image) and 10% test (2533image).

Hyper Parameter	Value
Initial Learning Rate	0.0001
Batch Size	32
Decay Steps	20000
Loss Function	Categorical Cross Entropy
Epochs	30
Optimizer	Adam

Table.1 Hyper parameters

8.1 Performance metrics

In a classification problem, only one metric such as Accuracy cannot help us evaluate the complete model efficiency effectively. Hence, we measure the Accuracy, Precision, Recall, F1 Score and Support for every class of the skin lesion disease.

We also plot the Confusion Matrix to check how well our model performs on every class.

We will now understand how the above-mentioned metrics are calculated and what they mean [84] [85],
TP = True Positive; FN = False Negative, FP = False Positive; TN = True Negative.

Accuracy

Accuracy is the fraction of predictions our model has correctly guessed.

It is defined as:

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

Precision

Precision metric answers the following question: What proportion of positive identifications was actually, correct?

It is defined as:

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

Sensitivity/Recall

Sensitivity is also called as the True Positive Rate (TPR) or Recall. It answers the following question: What proportion of actual positives was identified correctly?

It is defined as:

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

Balanced Accuracy:

Balanced Accuracy is used in both binary and multi-class classification. It's the arithmetic mean of sensitivity and specificity, its use case is when dealing with imbalanced datasets, also it was the metric used for evaluation in ISIC 2019 Challenge, calculated using this formula:

$$\text{Balanced Accuracy} = \frac{\sum_{j=0}^m \text{Recall}_j}{m}$$

Where m is the number of classes

F1 Score:

F1 Score is also known as the F-score or F-measure. The F1 score is calculated as a weighted average of the precision and recall. Its best value is 1 and worst is 0. The contribution of recall and precision in the calculation of the F1 score are equal.

It is defined as:

$$\text{F1 Score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

Specificity

Specificity is calculated as the number of correct negative predictions divided by the total number of negatives, calculated using this formula:

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

ROC AUC

ROC_AUC stands for “Receiver Operator Characteristic Area Under the Curve”. It summarizes the trade-off between the true positive rates and the false-positive rates for a predictive model. ROC yields good results when the observations are balanced between each class.

8.2 Conducted Result.

8.2.1 First Approach

In Table.2 a comparison between different pre-processing techniques and images size using two pretrained models (ResNet50, DenseNet201) and from the following experiment the best combination of pre-processing methods and image size (224*224) was the edge fining and normalization, and this setup was adapted in the next approach.

Based on our observations in Table.2 we also found the highest accuracy achieved by ResNet-50 with input size 32*32 was without any pre-processing as any pre-processing methods applied to this size cause a huge damage to the image pixels which result in low classification accuracy, for ResNet50 64*64 the highest model achieved 78.8% classification accuracy and 64% for balanced accuracy with images normalization methods , for ResNet-50 224*224 the highest model achieved 86% for classification accuracy and 77% for balanced accuracy on images pre-processed by the edge fining algorithm, for DenseNet201 with input size 32*32 achieved 74% for classification accuracy and 58% for balanced accuracy , for input size 64*64 the images pre-processed with normalization was the

highest model, for input size 224*224 images pre-processed with edge fining and normalization was the highest model .

Models/ Methods	No Pre- processing	Hair Removal	Normalization	Hair Removal & Normalization	Edge Fining	Edge Fining & hair	Edge Fining & Normalization	Edge Fining & hair & Normalization
ResNet50 32*32	ACC:72.72 BACC:56	ACC:72.48 BACC:57	ACC:73 BACC:54	ACC:71.3 BACC:54	ACC:72.2 BACC:54	ACC:70.59 BACC:54	ACC:69.1 BACC:53	ACC:70 BACC:53
ResNet50 64*64	ACC:78.68 BACC:66	ACC:78 BACC:64	ACC:78.8 BACC:64	ACC:78 BACC:65	ACC:77.3 BACC:64	ACC:77.97 BACC:64	ACC:75.6 BACC:61	ACC:77 BACC:63
ResNet50 224*224	ACC:85 BACC:77	ACC:83.85 BACC:75	ACC:84.92 BACC:76.5	ACC:84.29 BACC:73	ACC:86 BACC:77	ACC:86 BACC:76	ACC:85 BACC:78	ACC:85 BACC:75
DenseNet201 32*32	ACC:74 BACC:57	ACC:74 BACC:57	ACC:74 BACC:58	ACC:73.5 BACC:54	ACC:72.7 BACC:55	ACC:71.46 BACC:56	ACC:69.6 BACC:55	ACC:71 BACC:56
DenseNet201 64*64	ACC:80 BACC:69	ACC:81 BACC:68.8	ACC:81 BACC:71	ACC:79.5 BACC:66	ACC:80.22 BACC:66	ACC:77.77 BACC:63	ACC:78.1 BACC:66	ACC:78 BACC:65
DenseNet201 224*224	ACC:87.37 BACC:77	ACC:85.5 BACC:76	ACC:87 BACC:77.3	ACC:85.79 BACC:75	ACC:87.6 BACC:78	ACC:86.77 BACC:79	ACC:88 BACC:82	ACC:86 BACC:78

Table.2

8.2.2 Second Approach

In this approach we chose the edge fining and normalization to run all the following models to see what the best model is including from the previous approach ResNet-50 and DenseNet201, from the following experiments we observed that the highest performance was achieved by EfficientNetV2M with classification accuracy 88% and the lowest performance was VGG-19 with classification accuracy 76%

1. DenseNet-201: in Fig.24 shows classification report and in Fig.25 shows ROC-AUC curve and in Fig.26 shows the confusion matrix

	precision	recall	f1-score	support
0.0	0.77	0.80	0.79	87
1.0	0.90	0.90	0.90	332
2.0	0.78	0.79	0.78	262
3.0	0.87	0.83	0.85	24
4.0	0.86	0.77	0.81	452
5.0	0.91	0.95	0.93	1287
6.0	0.79	0.65	0.71	63
7.0	0.96	0.85	0.90	26
accuracy			0.88	2533
macro avg	0.85	0.82	0.83	2533
weighted avg	0.88	0.88	0.88	2533

Fig.24 classification report (DenseNet201)

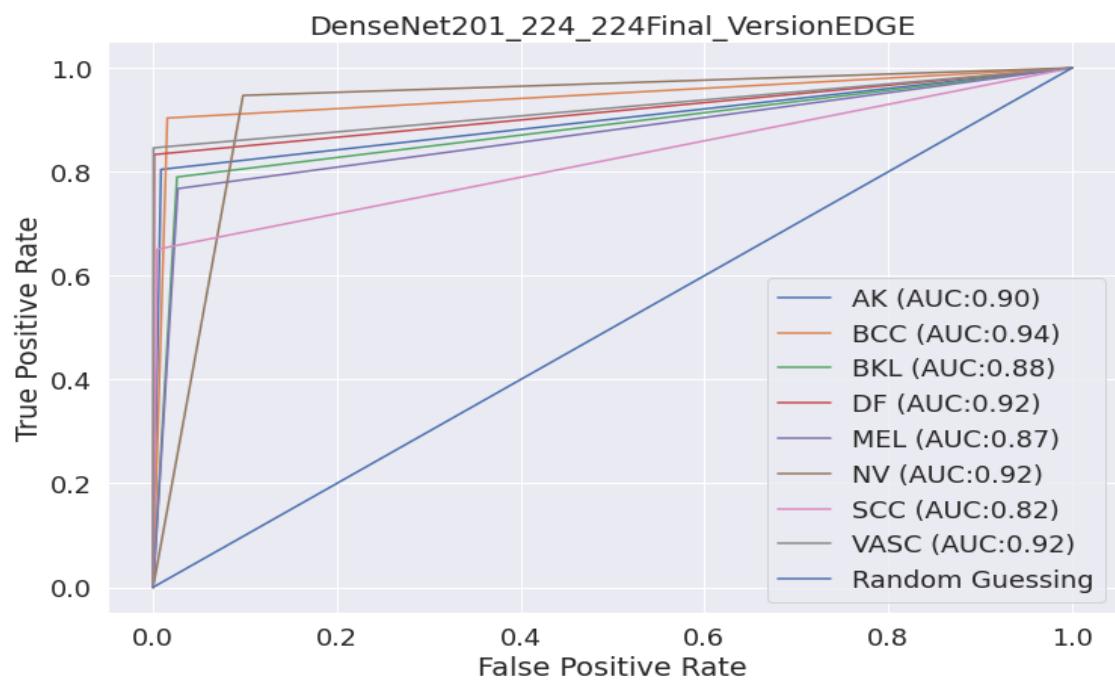


Fig.25 ROC-AUC (DenseNet201)

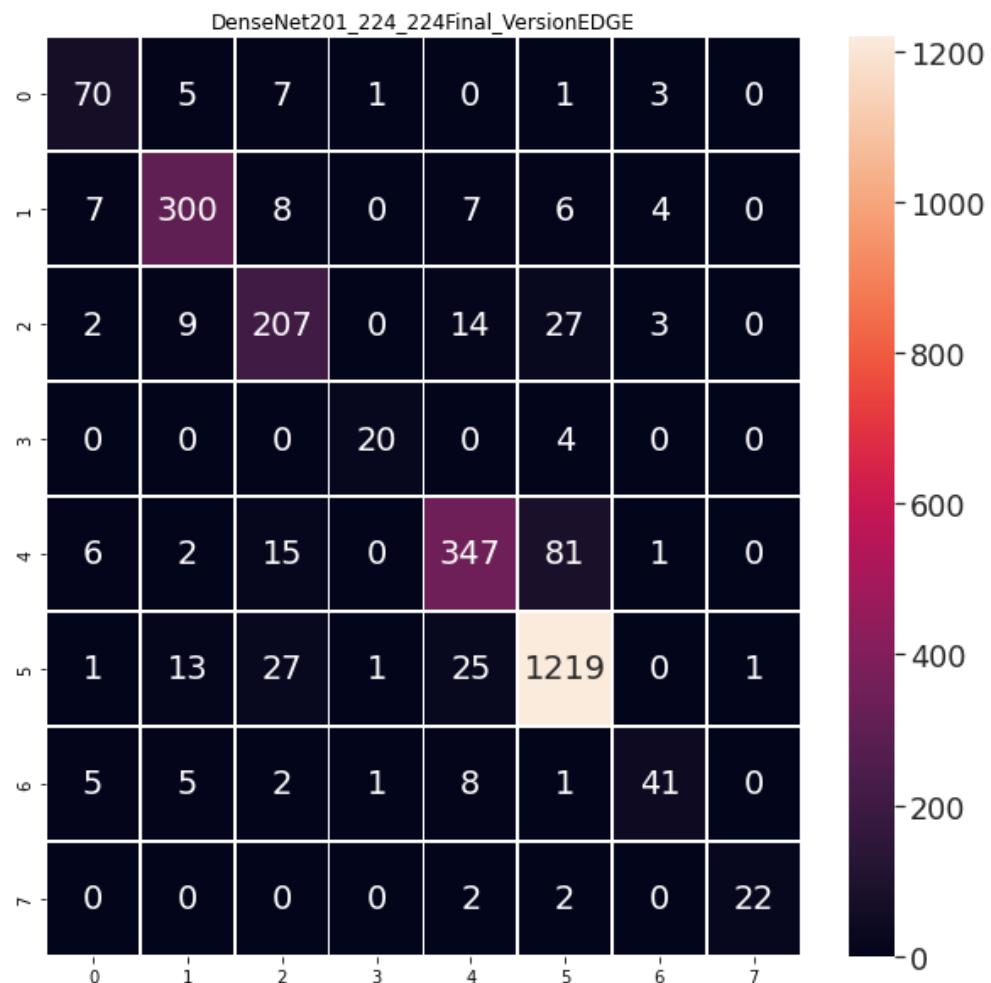


Fig.26 confusion matrix (DenseNet201)

2. ResNet-50: in Fig.27 shows classification report and in Fig.28 shows ROC-AUC curve and in Fig.29 shows the confusion matrix

	precision	recall	f1-score	support
0.0	0.71	0.71	0.71	87
1.0	0.87	0.81	0.84	332
2.0	0.73	0.80	0.76	262
3.0	0.86	0.79	0.83	24
4.0	0.85	0.72	0.78	452
5.0	0.89	0.94	0.91	1287
6.0	0.71	0.70	0.70	63
7.0	1.00	0.81	0.89	26
accuracy			0.85	2533
macro avg	0.83	0.78	0.80	2533
weighted avg	0.85	0.85	0.85	2533

Fig.27 classification report (ResNet-50)

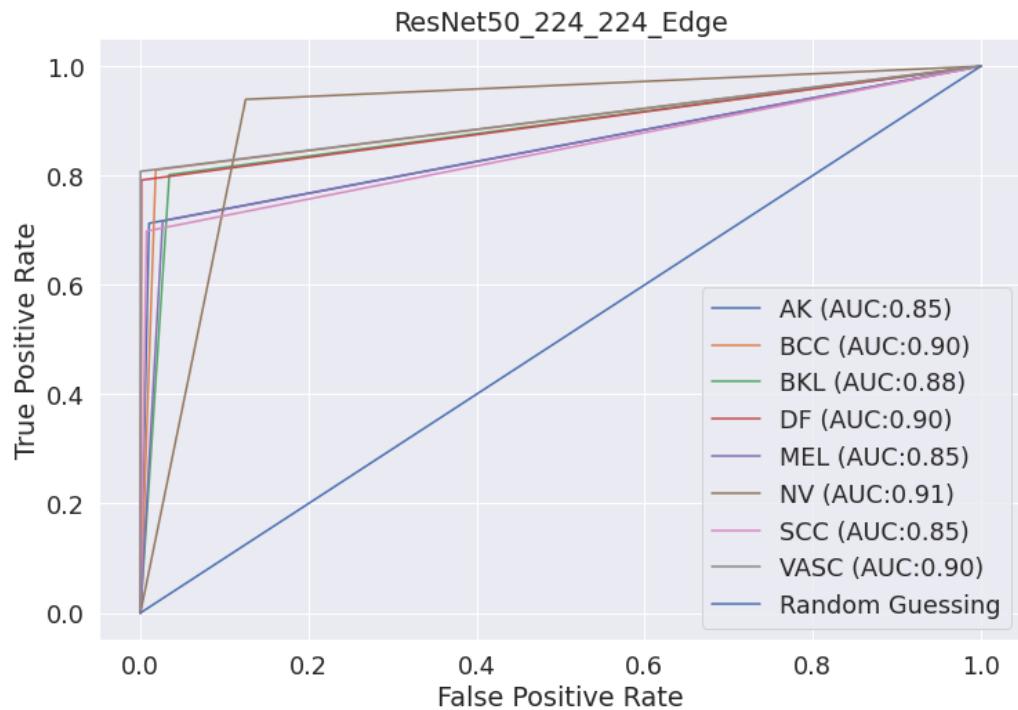


Fig.28 ROC-AUC (ResNet-50)

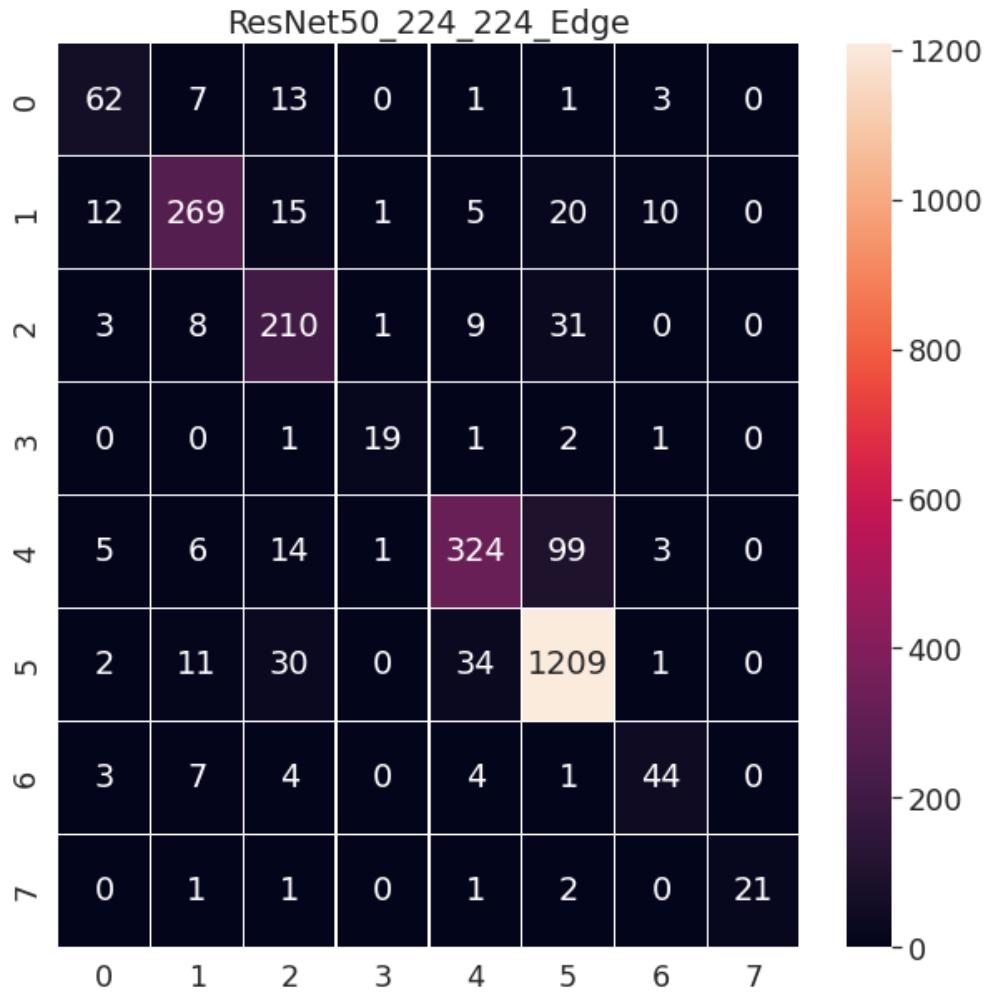


Fig.29 Confusion Matrix (ResNet-50)

3. InceptionV3: in Fig.30 shows classification report and in Fig.31 shows ROC-AUC curve and in Fig.32 shows the confusion matrix

	precision	recall	f1-score	support
0.0	0.74	0.68	0.71	87
1.0	0.88	0.85	0.87	332
2.0	0.72	0.81	0.76	262
3.0	0.95	0.75	0.84	24
4.0	0.86	0.71	0.78	452
5.0	0.89	0.94	0.92	1287
6.0	0.77	0.75	0.76	63
7.0	0.95	0.77	0.85	26
accuracy			0.86	2533
macro avg	0.85	0.78	0.81	2533
weighted avg	0.86	0.86	0.86	2533

Fig.30 Classification Report (ResNet-50)

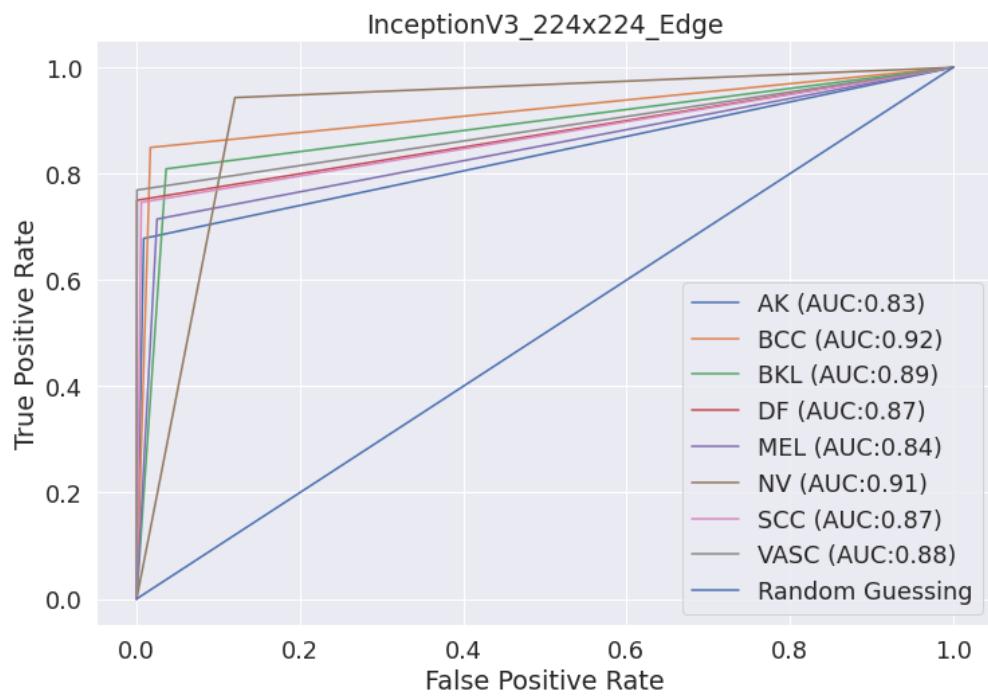


Fig.31 ROC-AUC (ResNet-50)

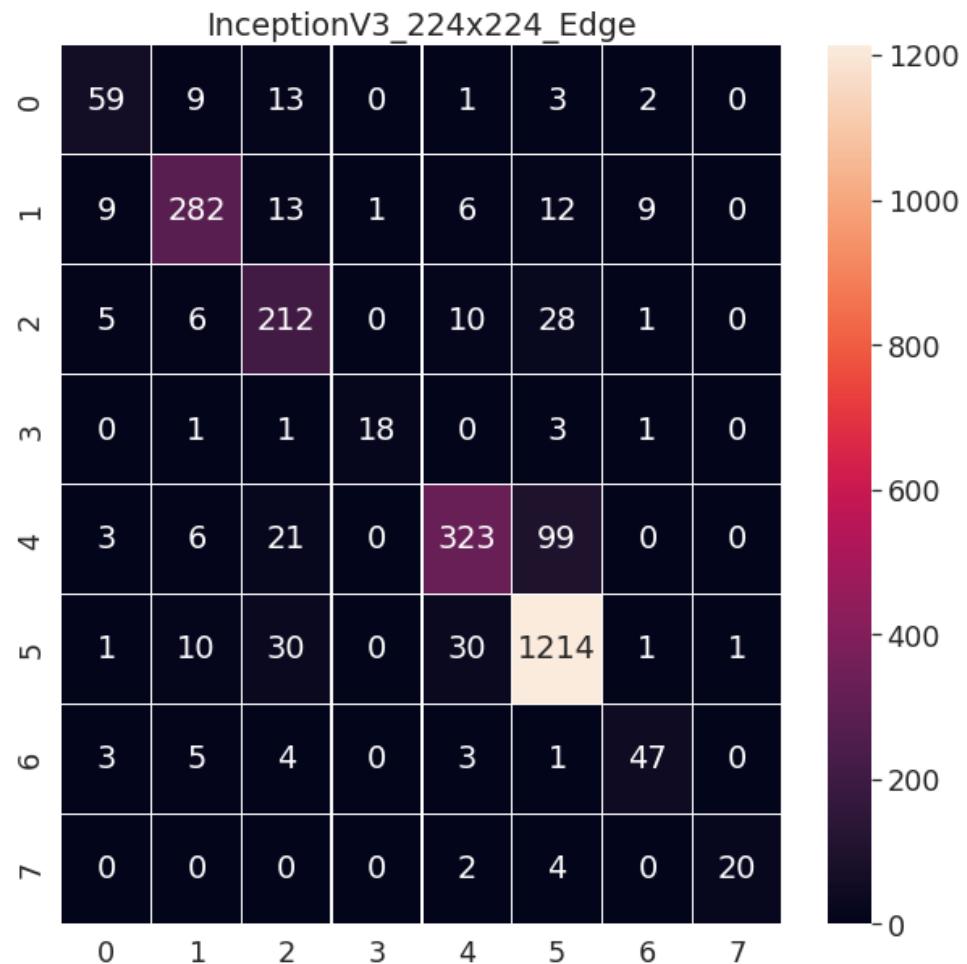


Fig.32 Confusion Matrix (ResNet-50)

4. EfficientNetV2M: in Fig.33 shows classification report and in Fig.34 shows ROC-AUC curve and in Fig.35 shows the confusion matrix

	precision	recall	f1-score	support
0.0	0.76	0.75	0.75	87
1.0	0.91	0.86	0.88	332
2.0	0.74	0.86	0.80	262
3.0	0.79	0.92	0.85	24
4.0	0.90	0.77	0.83	452
5.0	0.91	0.95	0.93	1287
6.0	0.75	0.71	0.73	63
7.0	0.95	0.81	0.88	26
accuracy			0.88	2533
macro avg	0.84	0.83	0.83	2533
weighted avg	0.88	0.88	0.88	2533

Fig.33 Classification Report (EfficientNetV2M)

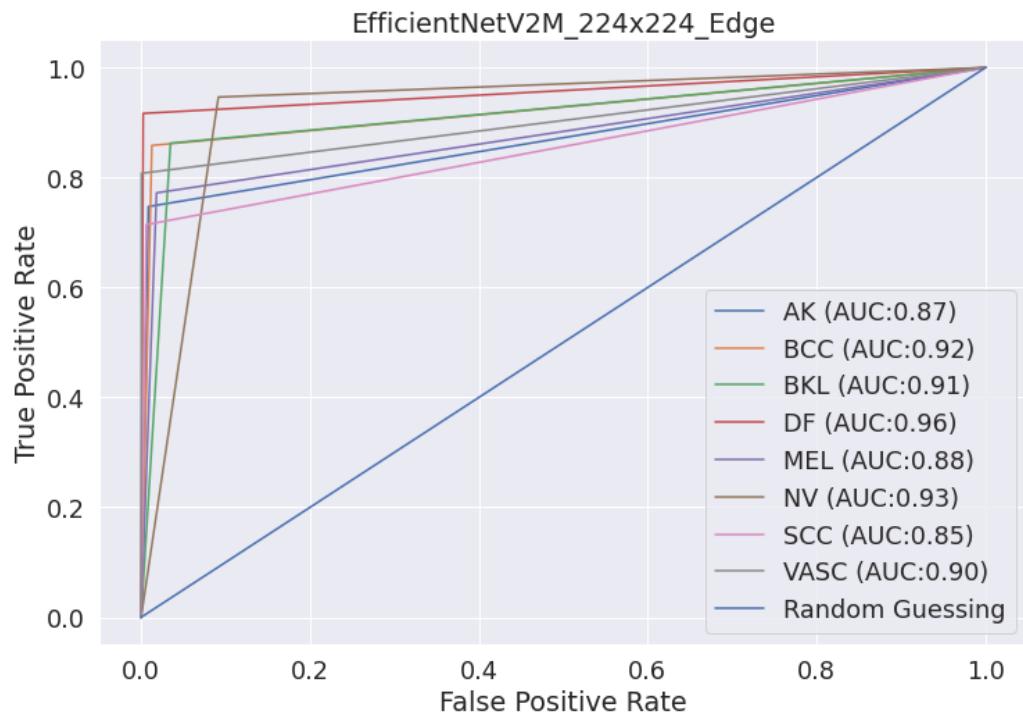


Fig.34 ROC-AUC(EfficientNetV2M)

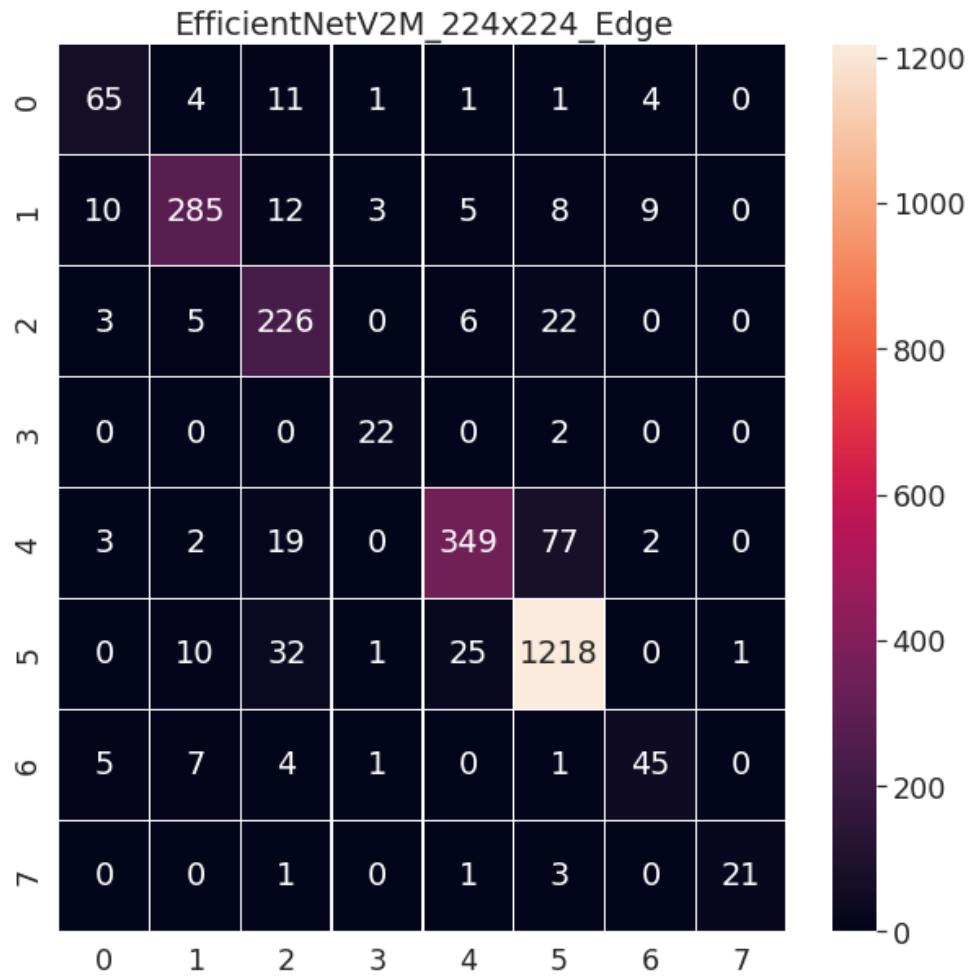


Fig.35 Confusion Matrix (EfficientNetV2M)

5. EfficientNetV2S: in Fig.36 shows classification report and in Fig.37 shows ROC-AUC curve and in Fig.38 shows the confusion matrix

	precision	recall	f1-score	support
0.0	0.70	0.75	0.72	87
1.0	0.89	0.84	0.86	332
2.0	0.77	0.79	0.78	262
3.0	0.85	0.92	0.88	24
4.0	0.88	0.78	0.83	452
5.0	0.91	0.95	0.93	1287
6.0	0.71	0.70	0.70	63
7.0	0.92	0.88	0.90	26
accuracy			0.88	2533
macro avg	0.83	0.83	0.83	2533
weighted avg	0.88	0.88	0.88	2533

Fig.36 Classification Report (EfficientNetV2S)

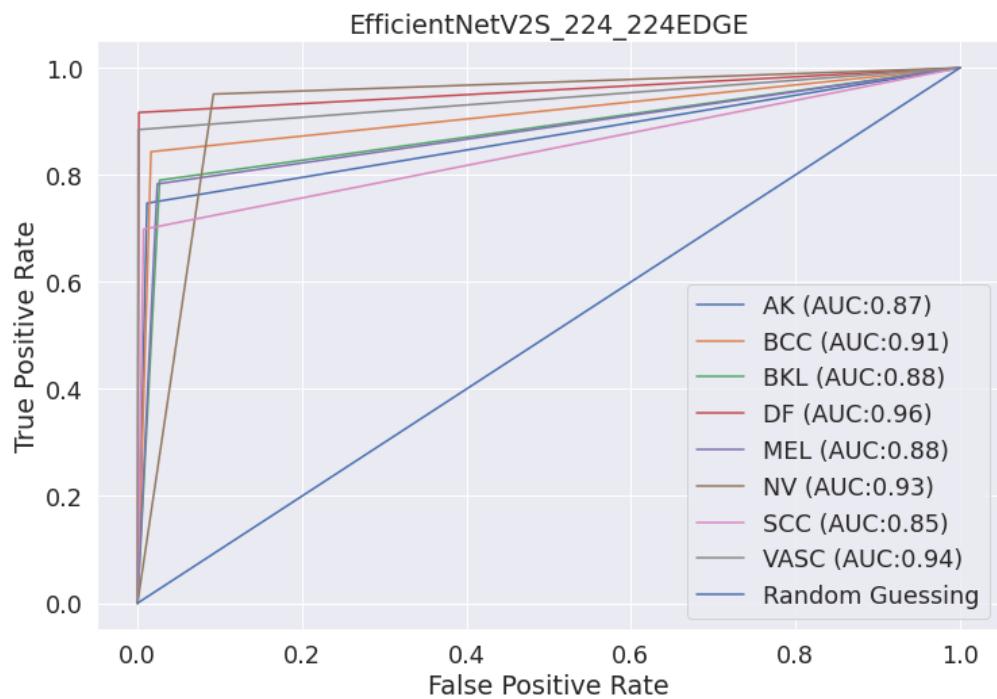


Fig.37 ROC-AUC (EfficientNetV2S)

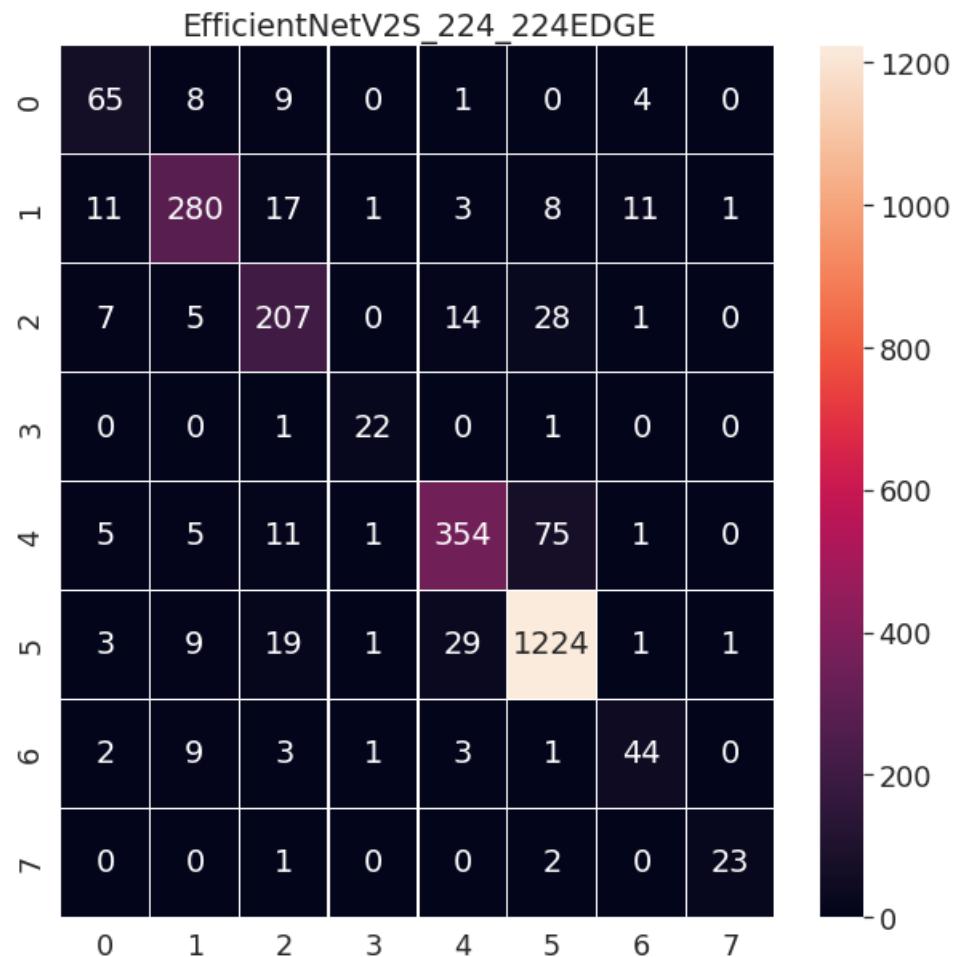


Fig.38 Confusion Matrix (EfficientNetV2S)

6. NASNetMobile: in Fig.39 shows classification report and in Fig.40 shows ROC-AUC curve and in Fig.41 shows the confusion matrix

	precision	recall	f1-score	support
0.0	0.71	0.67	0.69	87
1.0	0.89	0.84	0.86	332
2.0	0.80	0.74	0.77	262
3.0	0.85	0.71	0.77	24
4.0	0.84	0.73	0.78	452
5.0	0.88	0.96	0.91	1287
6.0	0.68	0.57	0.62	63
7.0	0.95	0.81	0.88	26
accuracy			0.85	2533
macro avg	0.82	0.75	0.78	2533
weighted avg	0.85	0.85	0.85	2533

Fig.39 Classification Report (NASNetMobile)

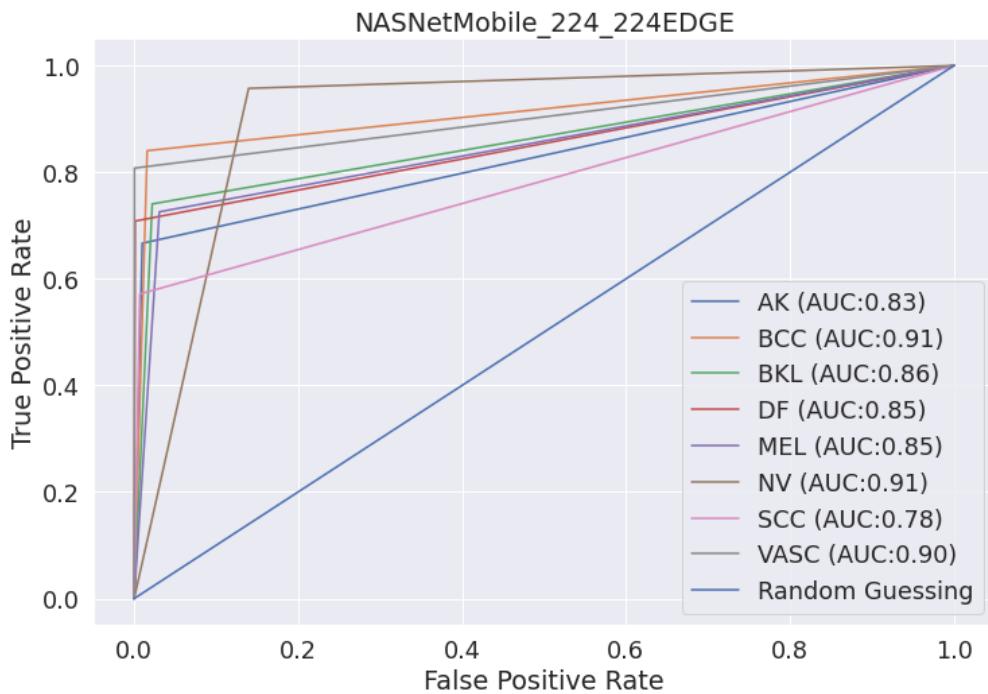


Fig.40 ROC-AUC (NASNetMobile)

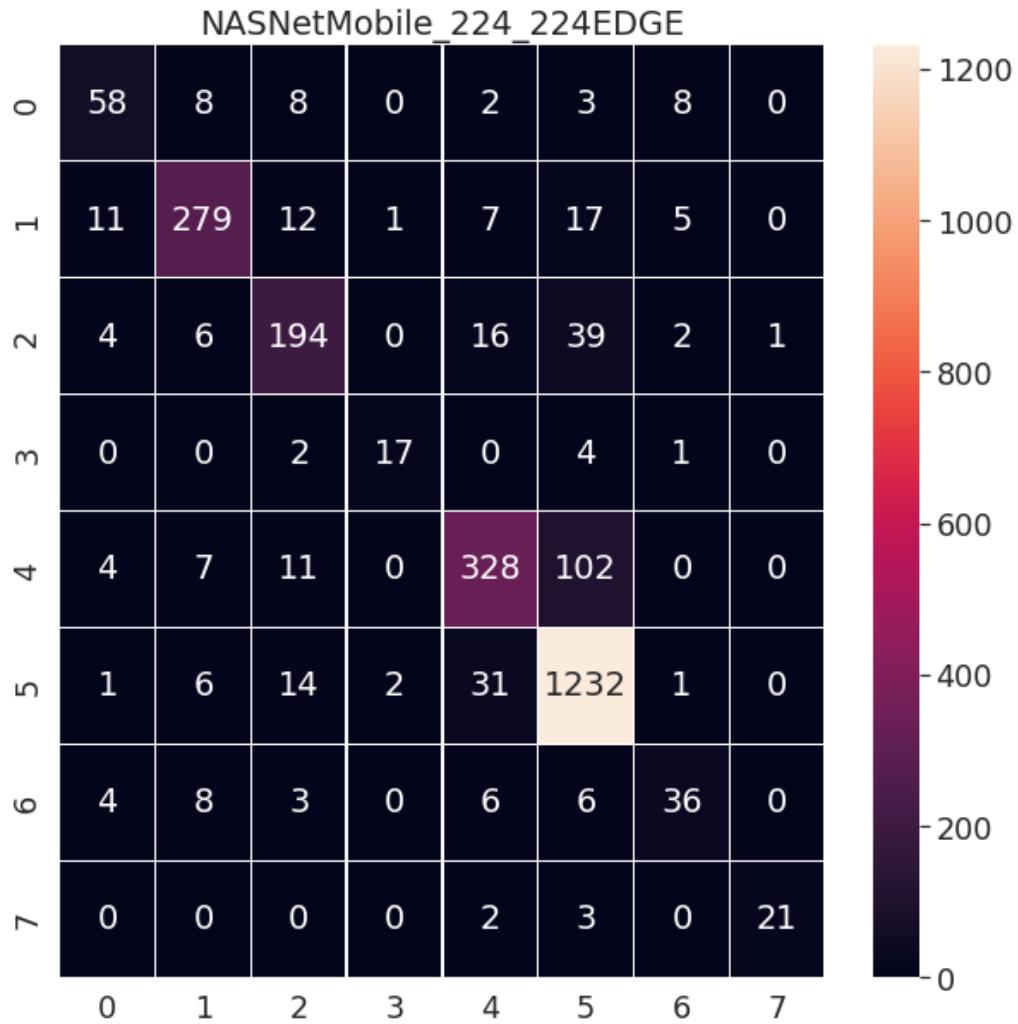


Fig.41 Confusion Matrix (NASNetMobile)

7. EfficientNet-B0: in Fig.42 shows classification report and in Fig.43 shows ROC-AUC curve and in Fig.44 shows the confusion matrix

	precision	recall	f1-score	support
0.0	0.71	0.69	0.70	87
1.0	0.90	0.83	0.86	332
2.0	0.74	0.76	0.75	262
3.0	0.80	0.83	0.82	24
4.0	0.81	0.68	0.74	452
5.0	0.87	0.93	0.90	1287
6.0	0.70	0.63	0.67	63
7.0	0.95	0.77	0.85	26
accuracy			0.84	2533
macro avg	0.81	0.77	0.79	2533
weighted avg	0.84	0.84	0.84	2533

Fig.42 Classification Report (EfficientNet-B0)

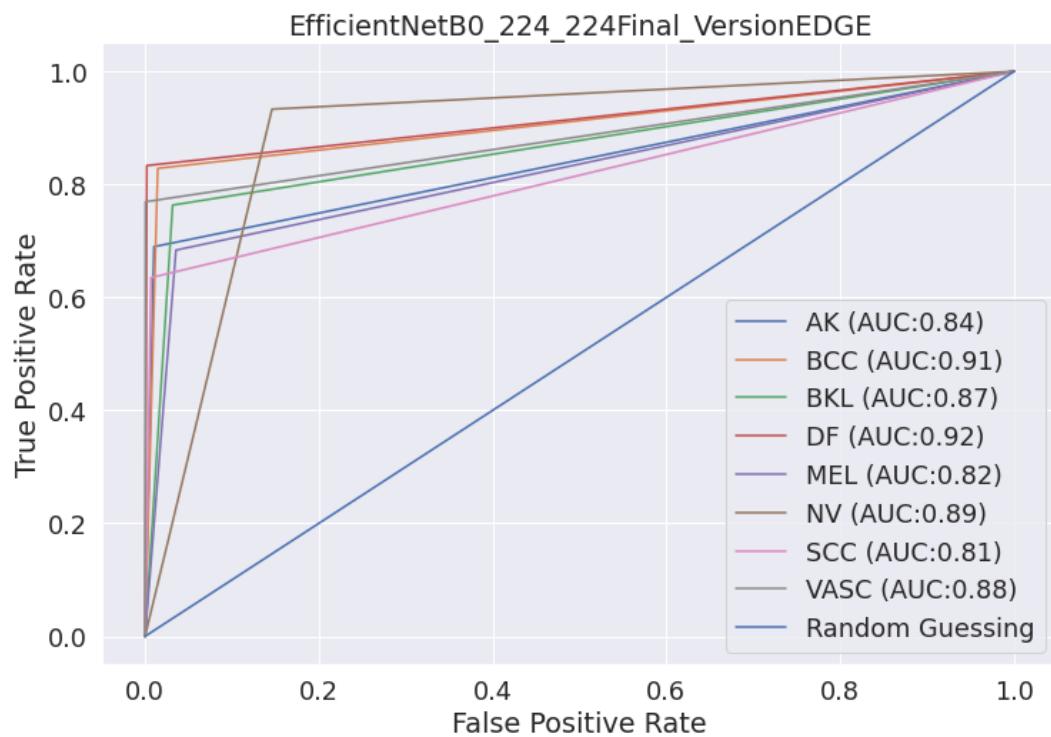


Fig.43 ROC-AUC (EfficientNet-B0)

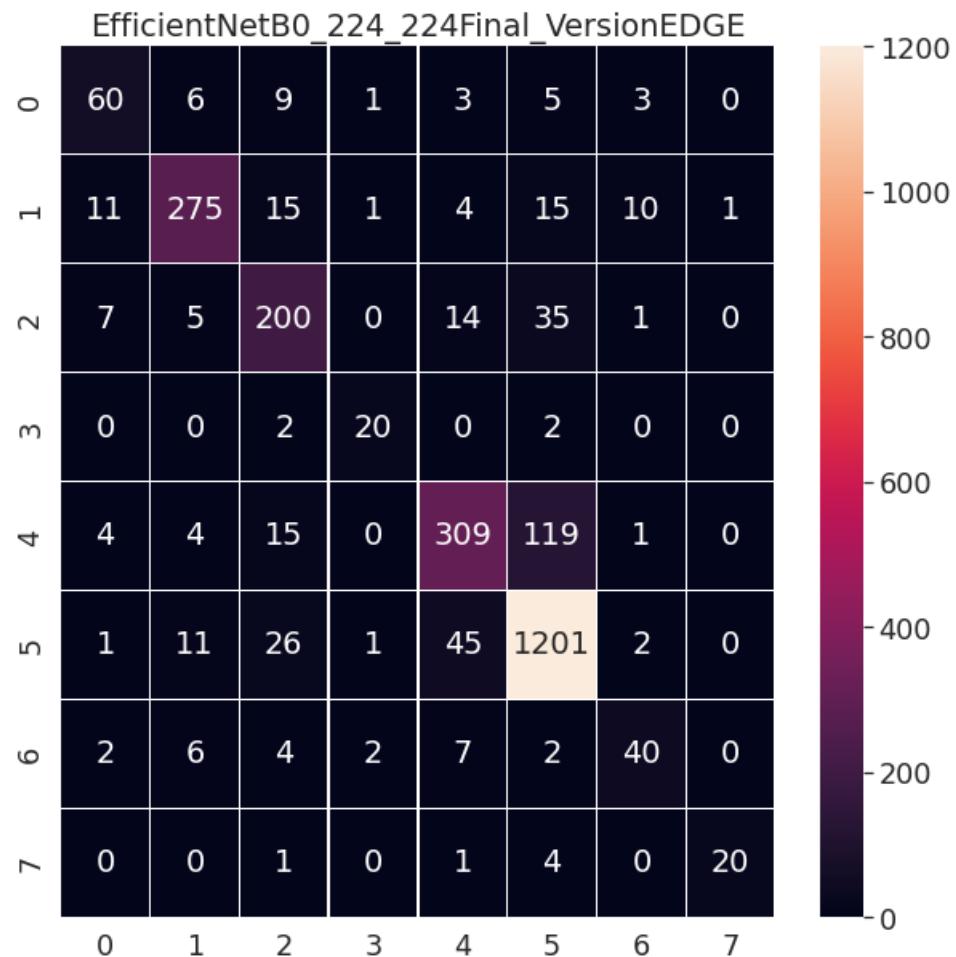


Fig.44 Confusion Matrix (EfficientNet-B0)

8. ResNet-152: in Fig.45 shows classification report and in Fig.46 shows ROC-AUC curve and in Fig.47 shows the confusion matrix

	precision	recall	f1-score	support
0.0	0.72	0.72	0.72	87
1.0	0.88	0.84	0.86	332
2.0	0.74	0.77	0.76	262
3.0	0.78	0.75	0.77	24
4.0	0.78	0.72	0.75	452
5.0	0.89	0.93	0.91	1287
6.0	0.76	0.62	0.68	63
7.0	1.00	0.73	0.84	26
accuracy			0.85	2533
macro avg	0.82	0.76	0.79	2533
weighted avg	0.85	0.85	0.85	2533

Fig.45 Classification Report (ResNet-152)

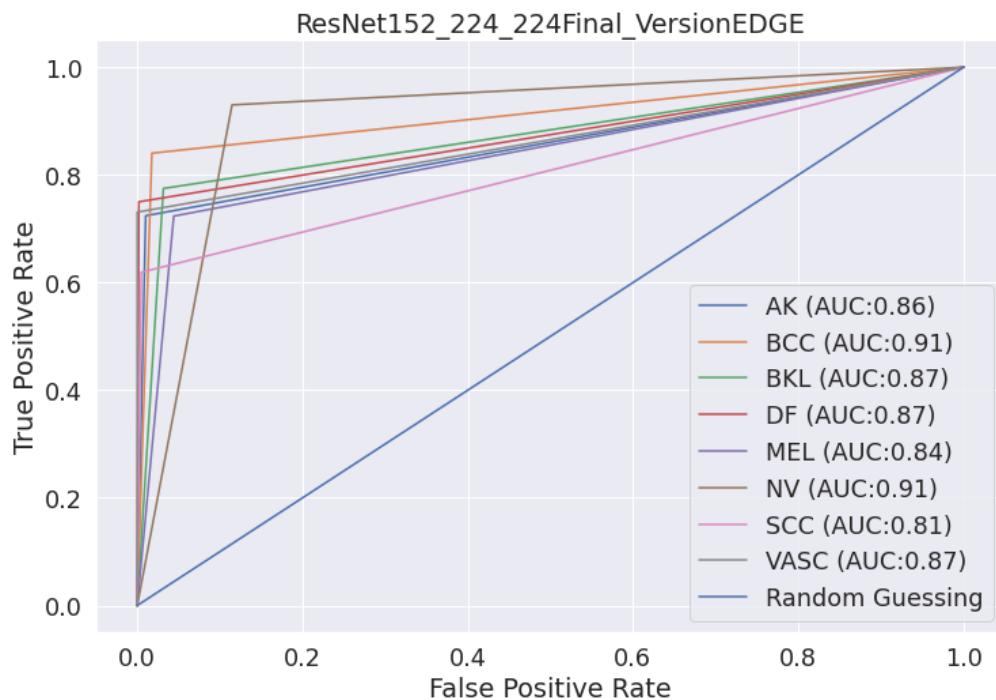


Fig.46 ROC-AUC (ResNet-152)

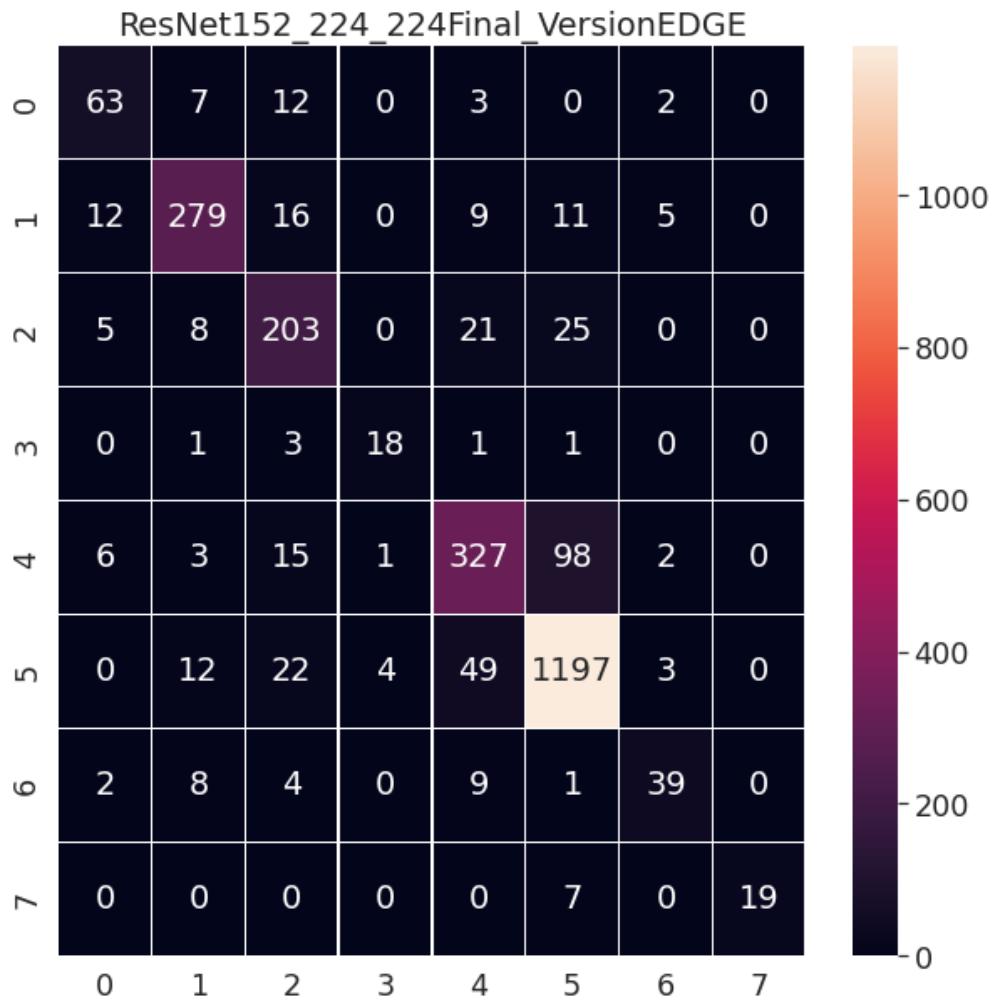


Fig.47 Confusion Matrix (ResNet-152)

9. VGG-19: in Fig.48 shows classification report and in Fig.49 shows ROC-AUC curve and in Fig.50 shows the confusion matrix

	precision	recall	f1-score	support
0.0	0.52	0.46	0.49	87
1.0	0.73	0.77	0.75	332
2.0	0.58	0.62	0.60	262
3.0	0.65	0.54	0.59	24
4.0	0.71	0.55	0.62	452
5.0	0.83	0.90	0.87	1287
6.0	0.50	0.33	0.40	63
7.0	0.90	0.73	0.81	26
accuracy			0.76	2533
macro avg	0.68	0.61	0.64	2533
weighted avg	0.75	0.76	0.75	2533

Fig.48 Classification Report (VGG-19)

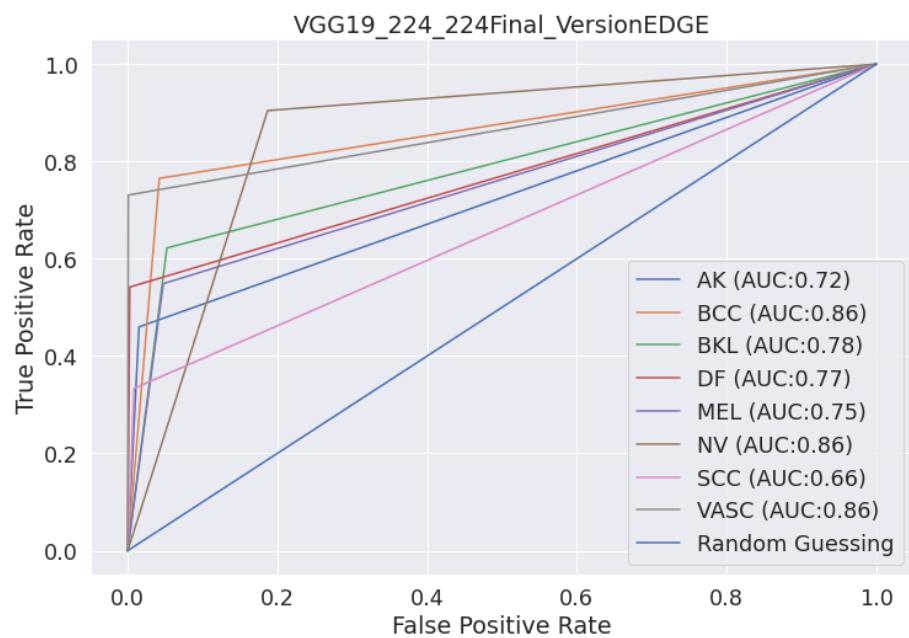


Fig.49 ROC-AUC (VGG-19)

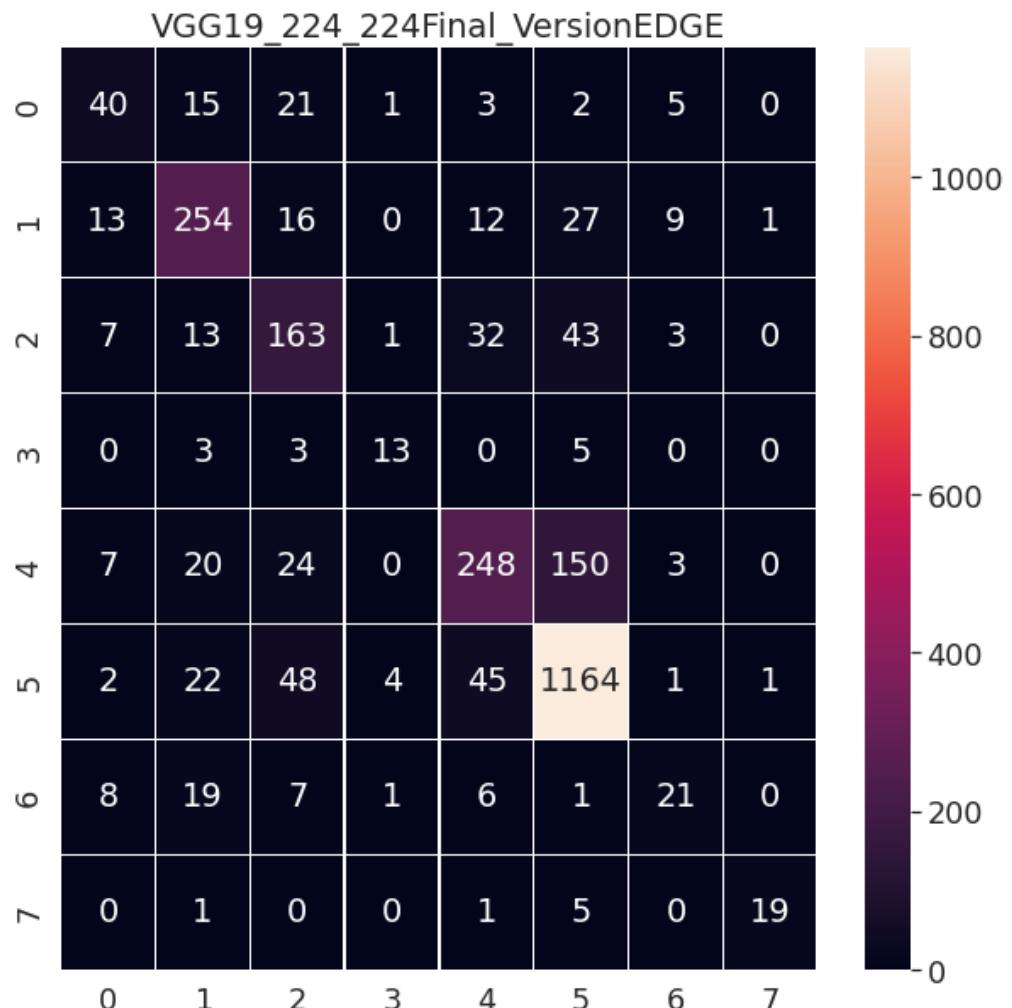


Fig.50 Confusion Matrix (VGG-19)

10. Xception: in Fig.51 shows classification report and in Fig.52 shows ROC-AUC curve and in Fig.53 shows the confusion matrix

	precision	recall	f1-score	support
0.0	0.74	0.67	0.70	87
1.0	0.89	0.89	0.89	332
2.0	0.78	0.79	0.78	262
3.0	0.84	0.88	0.86	24
4.0	0.85	0.76	0.80	452
5.0	0.90	0.95	0.92	1287
6.0	0.81	0.70	0.75	63
7.0	1.00	0.81	0.89	26
accuracy			0.87	2533
macro avg	0.85	0.80	0.83	2533
weighted avg	0.87	0.87	0.87	2533

Fig.51 Classification Report (Xception)

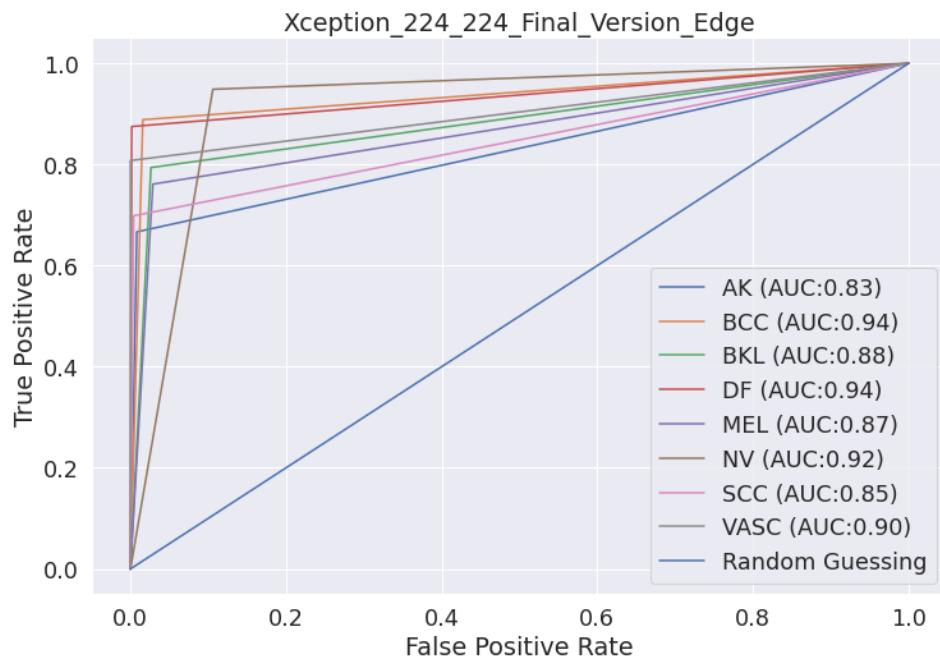


Fig.52 ROC-AUC (Xception)

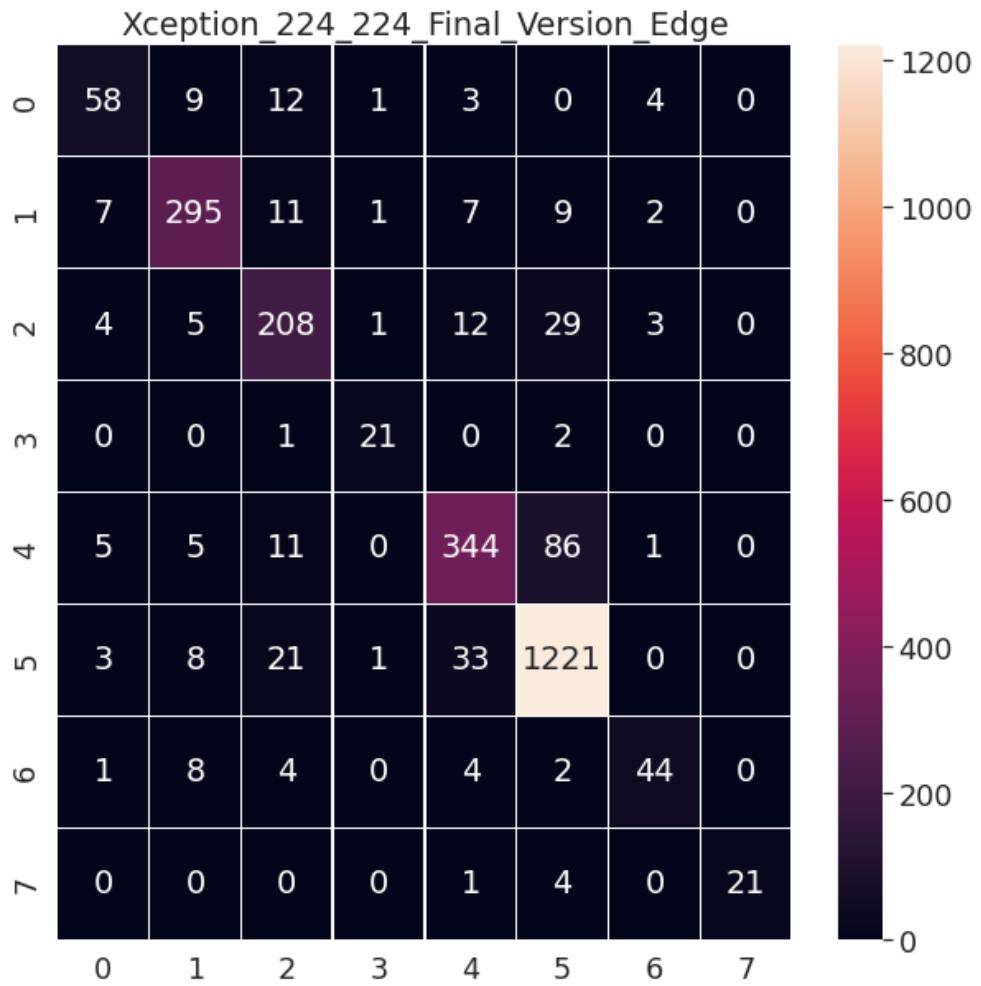


Fig.53 Confusion Matrix (Xception)

11. MobileNet: in Fig.54 shows classification report and in Fig.55 shows ROC-AUC curve and in Fig.56 shows the confusion matrix

	precision	recall	f1-score	support
0.0	0.73	0.51	0.60	87
1.0	0.87	0.79	0.83	332
2.0	0.63	0.77	0.70	262
3.0	0.74	0.83	0.78	24
4.0	0.78	0.64	0.71	452
5.0	0.87	0.93	0.90	1287
6.0	0.79	0.54	0.64	63
7.0	0.95	0.81	0.88	26
accuracy			0.82	2533
macro avg	0.80	0.73	0.75	2533
weighted avg	0.82	0.82	0.82	2533

Fig.54 Classification Report (MobileNet)

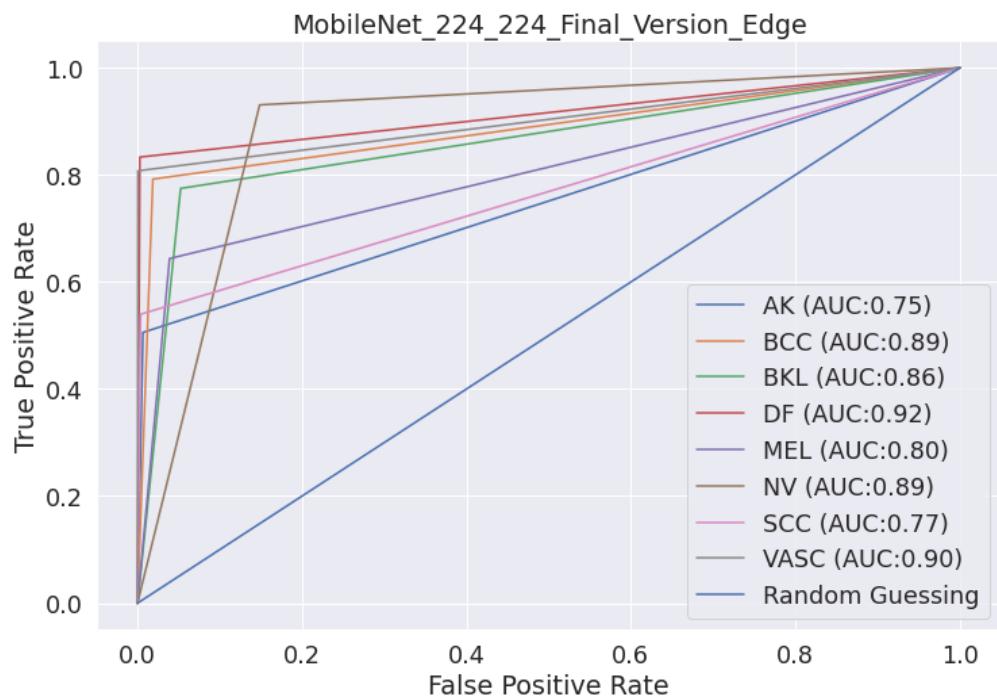


Fig.55 ROC-AUC (MobileNet)

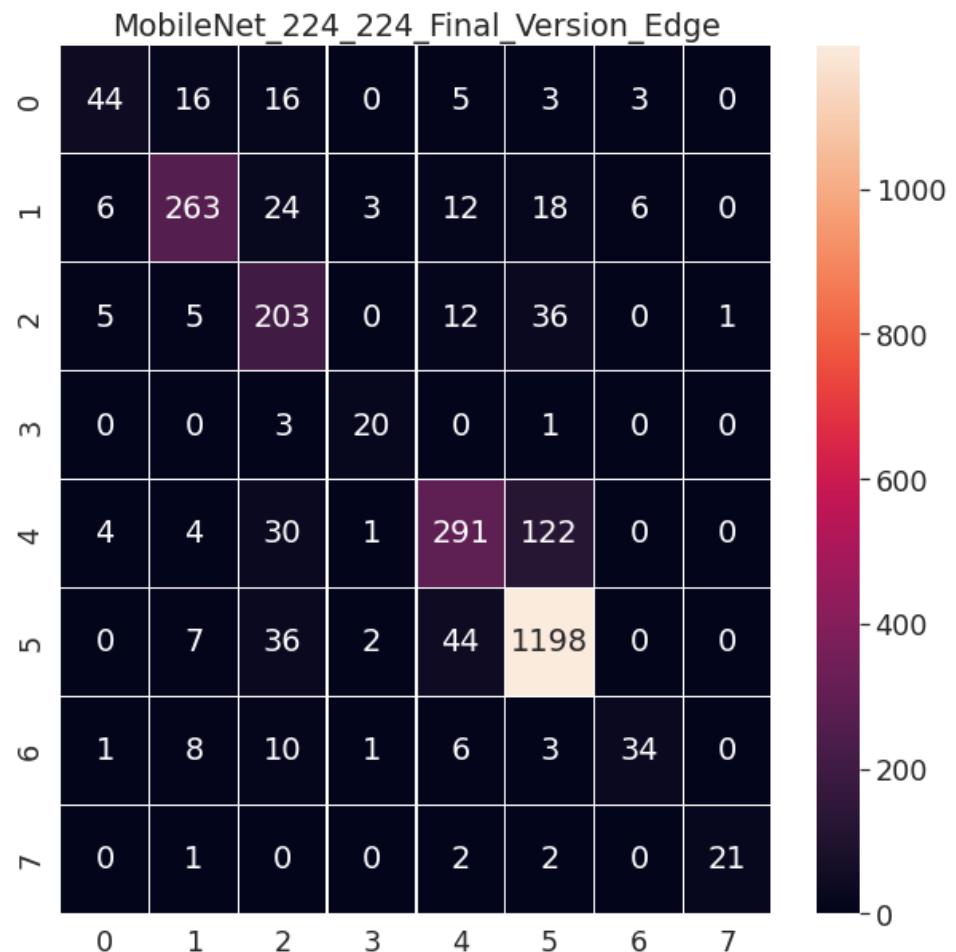


Fig.56 Confusion Matrix (MobileNet)

model	accuracy	sensitivity	specificity	precision	recall	AUC
VGG19	76%	61%	95.5%	68%	61%	78%
ResNet152	85%	76%	97%	82%	76%	86.6%
EfficientNet_B0	84%	76.6%	96.9%	81%	77%	86.8%
Dense201	88%	81.7%	97.7%	85%	82%	89.7%
ResNet50	85%	78%	97%	83%	78%	87.8%
Inceptionv3	86%	78%	97%	85%	78%	87.7%
EfficientNetV2M	88%	82%	97.8%	84%	83%	90%
EfficientNetV2S	88%	82%	97%	83%	83%	90%
NASNetMobile	85%	75%	97%	82%	75%	86%
Xception	87%	80%	97.6%	85%	80%	89%
MobileNet	82%	72.8%	96.6%	80%	73%	84.7%

Table.3 models by their performance

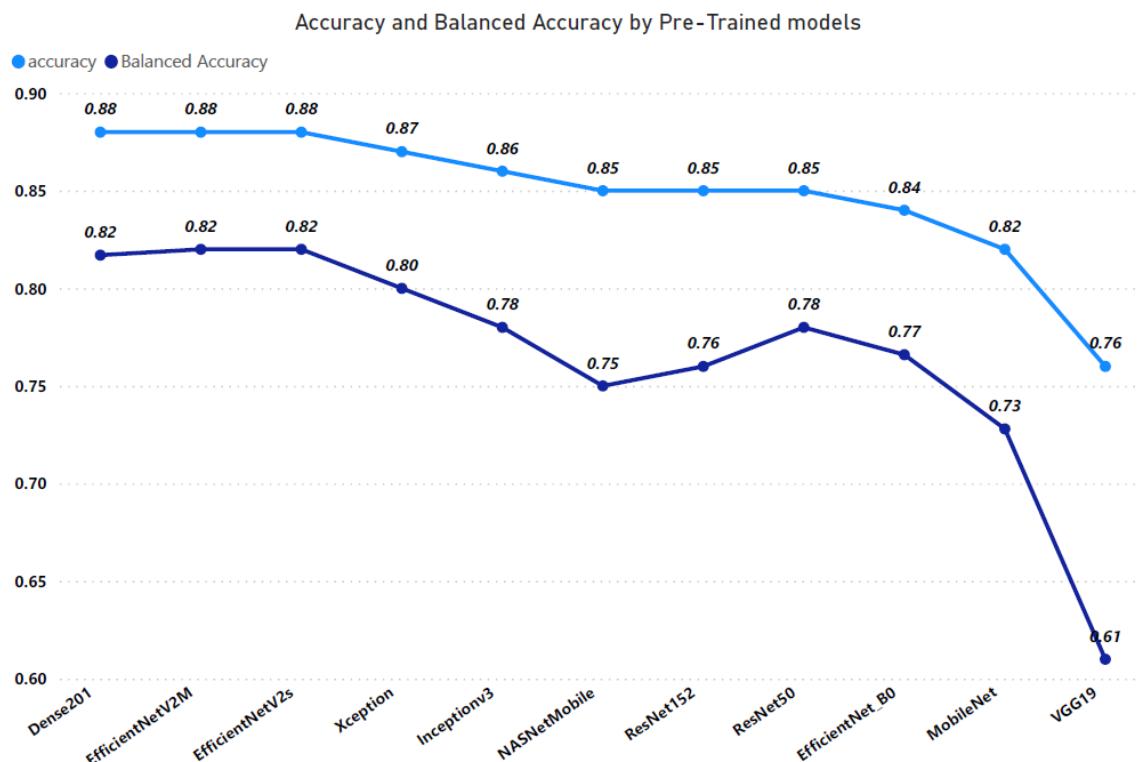


Fig.57

8.2.3 Third Approach

Ensemble using models trained in the previous approach and with the addition of some models from the first approach that had an input size of 224*224 applying weighted soft majority voting on them, we used the following models

- DenseNet-201 With edge fining and normalization
- DenseNet-201 With No pre-processing and Normalization

- DenseNet-201 With Hair Removal and Normalization
 - ResNet-50 With Edge Fining algorithm and Normalization
 - DenseNet-201 With Hair Removal, Edge Fining and Normalization
 - ResNet-50 With Hair Removal, Edge Fining and Normalization
 - ResNet-50 With Hair Removal Algorithm and Normalization
 - InceptionV3
 - EfficientNetV2M
 - EfficientNetV2S
 - NASNetMobile
 - EfficientNetB0
 - ResNet152
 - VGG19
 - Xception
 - MobileNet
1. The first trial we ensembled the highest performance models by accuracy which are (Xception, NasNetMobile, EfficientNet-V2S, EfficientNet-V2M, InceptionV3, ResNet152, DenseNet201 With edge fining and normalization, DenseNet201 with No pre-processing and Normalization, ResNet50 with Edge Fining algorithm and Normalization) which are above or equal 85% and achieved 90% accuracy and 84% balanced accuracy as shown in the following figures.

	precision	recall	f1-score	support
0.0	0.81	0.78	0.80	87
1.0	0.92	0.92	0.92	332
2.0	0.83	0.85	0.84	262
3.0	0.95	0.88	0.91	24
4.0	0.90	0.79	0.84	452
5.0	0.92	0.97	0.94	1287
6.0	0.89	0.75	0.81	63
7.0	0.95	0.81	0.88	26
accuracy			0.90	2533
macro avg	0.90	0.84	0.87	2533
weighted avg	0.90	0.90	0.90	2533

Fig.58 Classification Report

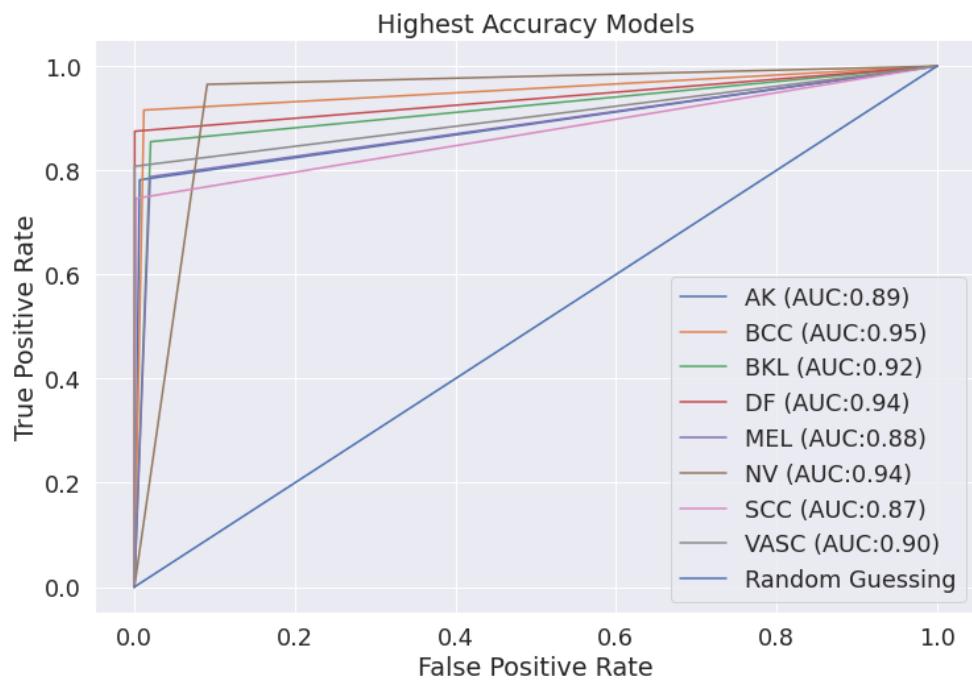


Fig.59 ROC-AUC

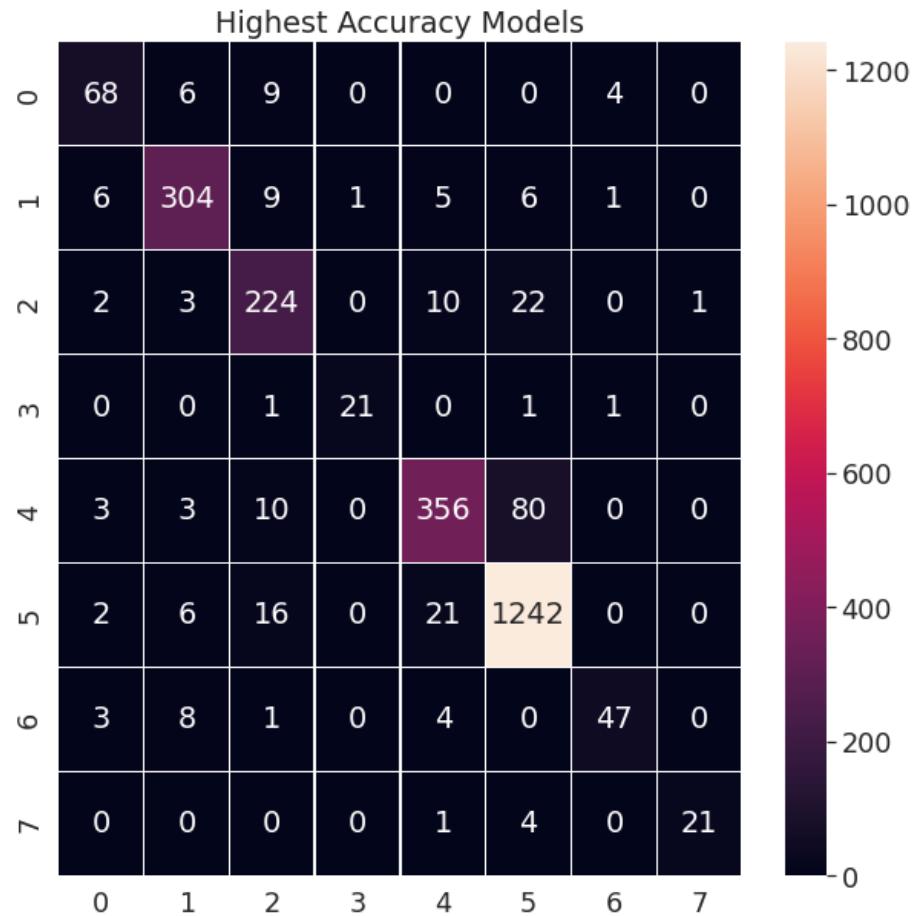


Fig.60 Confusion Matrix

2. The second trial we ensembled the lowest performance models by accuracy which are (MobileNet, VGG-19, DenseNet201 with Hair Removal and Normalization, DenseNet201 with Hair Removal and Edge Fining and Normalization,ResNet50 with Hair Removal and Edge Fining and Normalization,ResNet50 With Hair Removal Algorithm and Normalization, EfficientNetB0, VGG19, MobileNet) below 85% classification accuracy and achieved 87% accuracy and 80% balanced accuracy as shown in the following figures.

	precision	recall	f1-score	support
0.0	0.81	0.76	0.79	87
1.0	0.90	0.89	0.90	332
2.0	0.83	0.76	0.80	262
3.0	1.00	0.83	0.91	24
4.0	0.89	0.69	0.78	452
5.0	0.87	0.97	0.92	1287
6.0	0.93	0.62	0.74	63
7.0	1.00	0.85	0.92	26
accuracy				0.87
macro avg	0.90	0.80	0.84	2533
weighted avg	0.87	0.87	0.87	2533

Fig.61 Classification Report

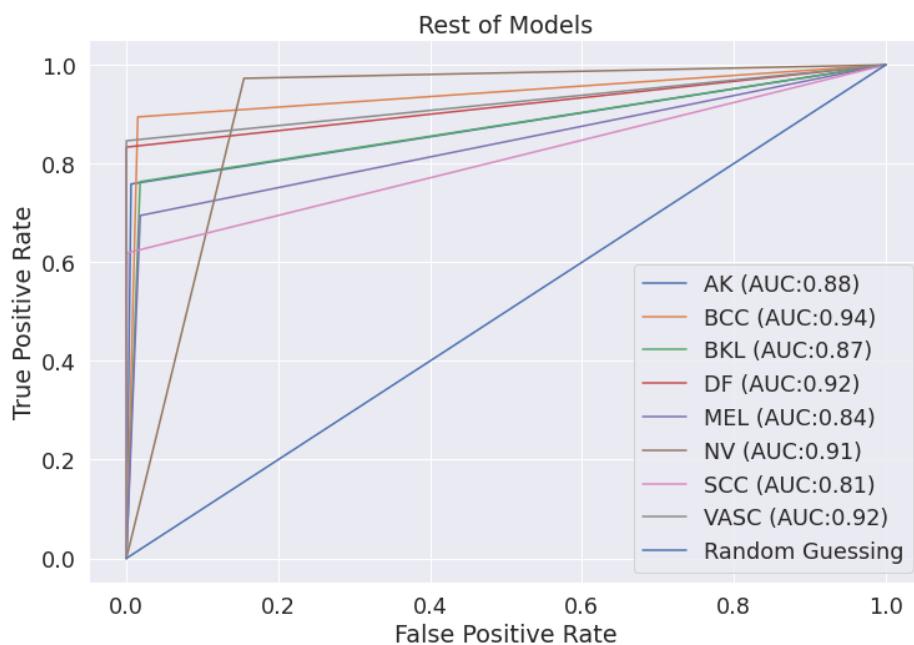


Fig.62 ROC-Auc

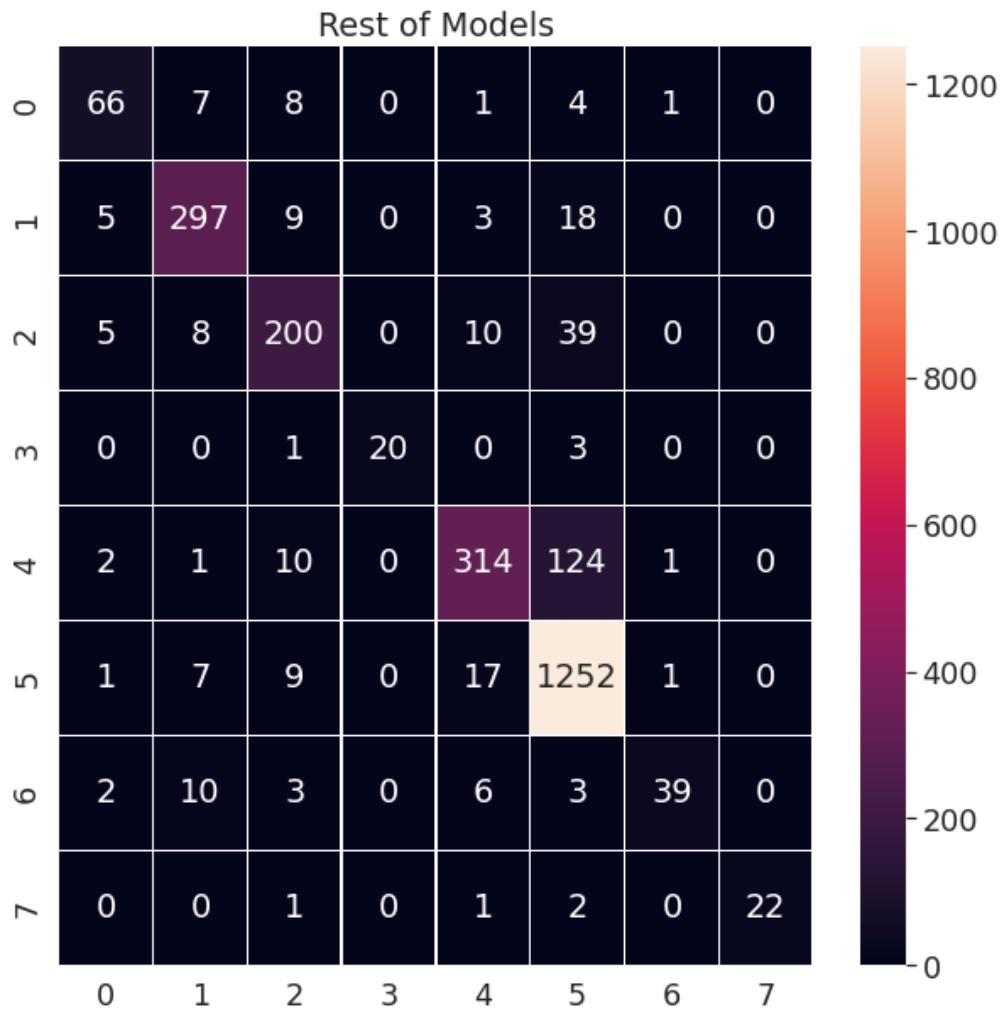


Fig.63 Confusion Matrix

3. The last trial we combined the two previous trials to form the final ensemble model which achieved 89.5% accuracy and 83% balanced accuracy as shown in the following figures.

	precision	recall	f1-score	support
0.0	0.82	0.78	0.80	87
1.0	0.91	0.92	0.91	332
2.0	0.82	0.82	0.82	262
3.0	1.00	0.88	0.93	24
4.0	0.90	0.77	0.83	452
5.0	0.91	0.97	0.94	1287
6.0	0.92	0.70	0.79	63
7.0	1.00	0.85	0.92	26
accuracy			0.90	2533
macro avg	0.91	0.83	0.87	2533
weighted avg	0.90	0.90	0.89	2533

Fig.64 Classification Report

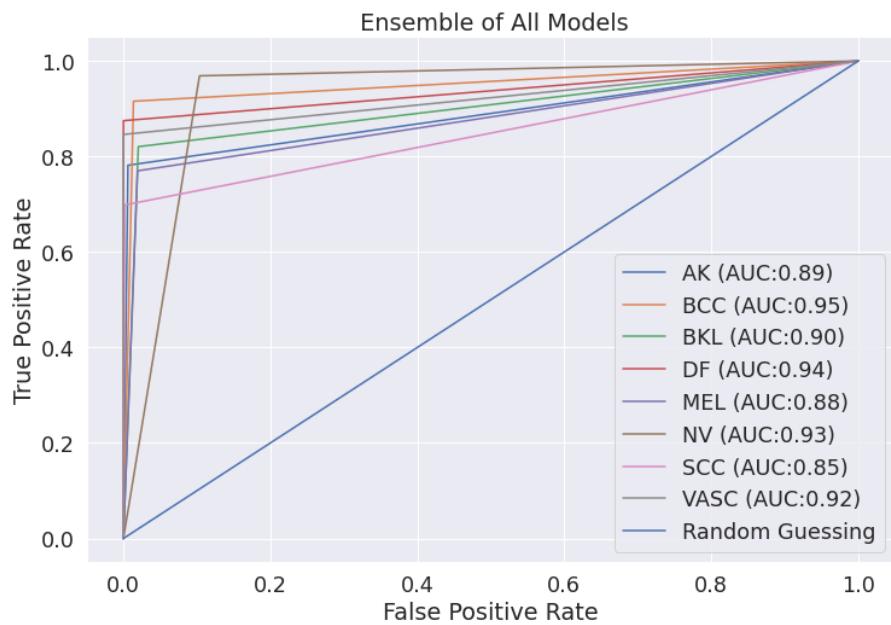


Fig.65 ROC-AUC

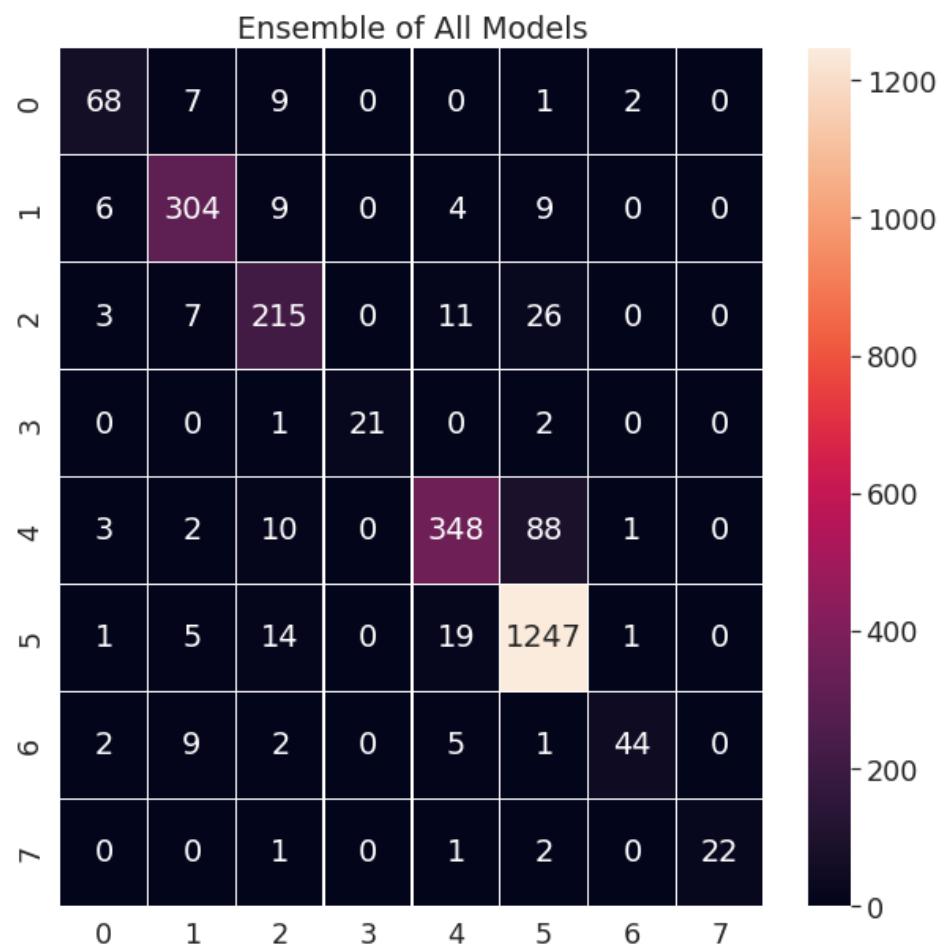


Fig.66 Confusion Matrix

8.2.4 Fourth Approach

We used two different type of augmentation methods (data-generator and manual augmentation) to train our multi-classification and binary based multi-classifier which results are stated as the table below

Model	ACC(%)	BACC(%)	F1-Score(%)
Multiclass-Classification with runtime augmentation	87	79	86
Binary-Tree-based Multiclass Classifier with real-time augmentation	86	77	86
Multiclass-Classification with balanced data	94	92	93
Binary-Tree-based Multiclass Classifier with balanced data	95	94	96

Chapter 9: Discussion, Conclusions, and Future Work

9.1 Summary & Conclusion

- pre-processing

we tried many image sizes such as 32x32, 64x64 and 224x224 and observed that the best image size for pre-trained models is 224x224. And the best combination for pre-processing techniques is edge fining and normalization which achieved with image size 224x224 on ResNet-50 85% accuracy and 78% balanced accuracy and densnet-201 88% accuracy and 82% balanced accuracy. As shown is the Table.2.

- Pretrained Models

In this approach we chose the highest accuracy models based on ILSVRC-ImageNet Large Scale Visual Recognition Challenge top accuracy models.

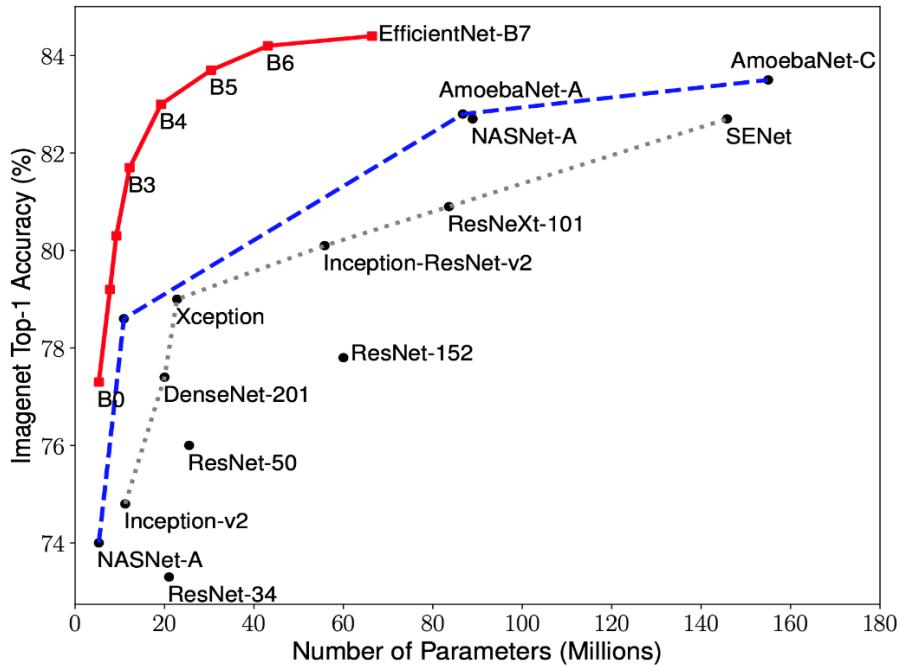


Fig.66

As shown in Fig.57 the best models are DenseNet-201, EfficientNetV2M and EfficientNetV2S which achieved 88% accuracy and 82% balanced accuracy. And all model's comparison shown in Table.3

- Ensemble

After implementing previous experiments using different pre-Trained models, we observed that the first trial is the best combination of pre-trained models to form the ensemble model which achieved 90% accuracy and 84% balanced accuracy as shown in the next Table.4

Ensemble model/Metrics	Accuracy (ACC)	Balanced Accuracy (BACC)	F1 Score (F1)	Precision (P)	AUC
<i>First trial</i>	90	84	87	90	91
<i>Second trial</i>	87	80	87	90	85
<i>Third trial</i>	89.5	83	87	91	90

Table.4

- Binary tree-based multi-classifier

As stated in table above it's been proven that our Binary-Tree-based Multiclass Classifier with balanced data has out-performed ensemble model and all other traditional multiclass-classifier

using the same balanced data which proves the great effect results from segmenting our multiclass-model into multiple binary classifiers.

9.2 Future Work

The future work can be summarized into four main points, first is to use bigger image size as it heavily affects classification accuracy as shown in table.3. secondly to try other ensemble methods and combination as bagging, boosting, and stacking. Thirdly, we could apply our method on other publicly available datasets on skin cancer. Fourthly we aim to implement the Binary-Tree-based Multiclass Classifier based on an automatically designed CNN architecture using genetic algorithm.

References

- [1] Taufiq, M.A., Hameed, N., Anjum, A., Hameed, F. (2017). m-Skin Doctor: A Mobile Enabled System for Early Melanoma Skin Cancer Detection Using Support Vector Machine. In: Giokas, K., Bokor, L., Hopfgartner, F. (eds) eHealth 360°. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol 181. Springer, Cham. https://doi.org/10.1007/978-3-319-49655-9_57.
- [2] <https://www.skincancer.org/skin-cancer-information/skin-cancer-facts/#:~:text=An%20estimated%201.8%20million%20cases,in%20the%20U.S.%20each%20year.&text=The%20latest%20figures%20suggest%20that,nonmelanoma%20skin%20cancer%20every%20month>
- [3] F. P. dos Santos and M. A. Ponti, ``Robust feature spaces from pre-trained deep network layers for skin lesion classification," in *Proc. 31st SIBGRAPI Conf. Graph., Patterns Images (SIBGRAPI)*, Oct. 2018, pp. 189–196.
- [4] A. H. Shahin, A. Kamal, and M. A. Elattar, ``Deep ensemble learning for skin lesion classification from dermoscopic images," in *Proc. 9th Cairo Int. Biomed. Eng. Conf. (CIBEC)*, Dec. 2018, pp. 150–153.
- [5] T. Zhou, K.-H. Thung, X. Zhu, and D. Shen, ``Effective feature learning and fusion of multimodality data using stage-wise deep neural network for dementia diagnosis," *Hum. Brain Mapping*, vol. 40, no. 3, pp. 1001–1016, 2019.
- [6] N. Liu, L. Wan, Y. Zhang, T. Zhou, H. Huo, and T. Fang, ``Exploiting convolutional neural networks with deeply local description for remote sensing image classification," *IEEE Access*, vol. 6, pp. 11215–11228, 2018.
- [7] J. Li, G. Zhou, Y. Qiu, Y. Wang, Y. Zhang, and S. Xie, ``Deep graph regularized non-negative matrix factorization for multi-view clustering," *Neurocomputing*, vol. 390, pp. 108–116, May 2020.

- [8] 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>
- [9] M. A. Khan, M. Y. Javed, M. Sharif, T. Saba and A. Rehman, "Multi-Model Deep Neural Network based Features Extraction and Optimal Selection Approach for Skin Lesion Classification," 2019 International Conference on Computer and Information Sciences (ICCIS), 2019, pp. 1-7, doi: 10.1109/ICCISSci.2019.8716400.
- [10] Kandukuri, Nikhil & Sakthivel, Sangeetha & Xie, Pengtao. (2021). Neural Architecture Search For Skin Cancer Detection. 10.21203/rs.3.rs-953342/v1.
- [11] A. Kwasigroch, M. Grochowski and A. Mikołajczyk, "Neural Architecture Search for Skin Lesion Classification," in IEEE Access, vol. 8, pp. 9061-9071, 2020, doi: 10.1109/ACCESS.2020.2964424.
- [12] Muhammad Attique Khan, Yu-Dong Zhang, Muhammad Sharif, Tallha Akram, Pixels to Classes: Intelligent Learning Framework for Multiclass Skin Lesion Localization and Classification, Computers & Electrical Engineering, Volume 90, 2021.
- [13] Khan, M.A.; Sharif, M.; Akram, T.; Damaševičius, R.; Maskeliūnas, R. Skin Lesion Segmentation and Multiclass Classification Using Deep Learning Features and Improved Moth Flame Optimization. *Diagnostics* **2021**, *11*, 811. <https://doi.org/10.3390/diagnostics11050811>.
- [14] <https://arxiv.org/abs/2103.12068> -----
- [15] Mijwil, M.M. Skin cancer disease images classification using deep learning solutions. *Multimed Tools Appl* **80**, 26255–26271 (2021). <https://doi.org/10.1007/s11042-021-10952-7>
- [16] V.R. Balaji, S.T. Suganthi, R. Rajadevi, V. Krishna Kumar, B. Saravana Balaji, Sanjeevi Pandiyan, Skin disease detection and segmentation using dynamic graph cut algorithm and classification through Naive Bayes classifier, Measurement, Volume 163, 2020 , 107922, ISSN 0263-2241, <https://doi.org/10.1016/j.measurement.2020.107922> .
- [17] Moldovanu, S.; Damian Michis, F.A.; Biswas, K.C.; Culea-Florescu, A.; Moraru, L. Skin Lesion Classification Based on Surface Fractal Dimensions and Statistical Color Cluster Features Using an Ensemble of Machine Learning Techniques. *Cancers* **2021**, *13*, 5256. <https://doi.org/10.3390/cancers13215256> .
- [18] K. Thurnhofer-Hemsi, E. López-Rubio, E. Domínguez and D. A. Elizondo, "Skin Lesion Classification by Ensembles of Deep Convolutional Networks and Regularly Spaced Shifting," in IEEE Access, vol. 9, pp. 112193-112205, 2021, doi: 10.1109/ACCESS.2021.3103410.
- [19] Nils Gessert, Maximilian Nielsen, Mohsin Shaikh, René Werner, Alexander Schlaefer, Skin lesion classification using ensembles of multi-resolution EfficientNets with meta data, MethodsX, Volume 7, 2020 , 100864, ISSN 2215-0161, <https://doi.org/10.1016/j.mex.2020.100864> .
- [20] H. Q. Yu and S. Reiff-Marganiec, "Targeted Ensemble Machine Classification Approach for Supporting IoT Enabled Skin Disease Detection," in IEEE Access, vol. 9, pp. 50244-50252, 2021, doi: 10.1109/ACCESS.2021.3069024.

- [21] M. A. Albahar, "Skin Lesion Classification Using Convolutional Neural Network With Novel Regularizer," in IEEE Access, vol. 7, pp. 38306-38313, 2019, doi: 10.1109/ACCESS.2019.2906241.
- [22] Mohammad Ali Kadampur, Sulaiman Al Riyae, Skin cancer detection: Applying a deep learning based model driven architecture in the cloud for classifying dermal cell images, Informatics in Medicine Unlocked, Volume 18, 2020, 100282, ISSN 2352-9148, <https://doi.org/10.1016/j.imu.2019.100282>.
- [23] S. Naresh Kumar, B. Mohammed Ismail, Systematic investigation on Multi-Class skin cancer categorization using machine learning approach, Materials Today: Proceedings, 2020 , ISSN 2214-7853, <https://doi.org/10.1016/j.matpr.2020.11.484>.
- [24] Gessert, Nils & Sentker, Thilo & Madesta, Frederic & Schmitz, Rüdiger & Kniep, Helge & Baltruschat, Ivo & Werner, René & Schlaefer, Alexander. (2018). Skin Lesion Diagnosis using Ensembles, Unscaled Multi-Crop Evaluation and Loss Weighting.
- [25] S. A. A. Ahmed, B. Yanikoğlu, Ö. Göksu and E. Aptoula, "Skin Lesion Classification With Deep CNN Ensembles," 2020 28th Signal Processing and Communications Applications Conference (SIU), 2020, pp. 1-4, doi: 10.1109/SIU49456.2020.9302125.
- [26] Lee, Yeong Chan & Jung, Sang-Hyuk & Won, Hong-Hee. (2018). WonDerM: Skin Lesion Classification with Fine-tuned Neural Networks.
- [27] M. K. Islam et al., "Melanoma Skin Lesions Classification using Deep Convolutional Neural Network with Transfer Learning," 2021 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA), 2021, pp. 48-53, doi: 10.1109/CAIDA51941.2021.9425117.
- [28] D. Moldovan, "Transfer Learning Based Method for Two-Step Skin Cancer Images Classification," 2019 E-Health and Bioengineering Conference (EHB), 2019, pp. 1-4, doi: 10.1109/EHB47216.2019.8970067.
- [29] T. -C. Lin and H. -C. Lee, "Skin Cancer Dermoscopy Images Classification with Meta Data via Deep Learning Ensemble," 2020 International Computer Symposium (ICS), 2020, pp. 237-241, doi: 10.1109/ICS51289.2020.00055.
- [30] Y. Zhang and C. Wang, "SIIM-ISIC Melanoma Classification With DenseNet," 2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), 2021, pp. 14-17, doi: 10.1109/ICBAIE52039.2021.9389983.
- [31] Y. Jusman, I. M. Firdiantika, D. A. Dharmawan and K. Purwanto, "Performance of Multi Layer Perceptron and Deep Neural Networks in Skin Cancer Classification," 2021 IEEE 3rd Global Conference on Life Sciences and Technologies (LifeTech), 2021, pp. 534-538, doi: 10.1109/LifeTech52111.2021.9391876.
- [32] M. A. Sabri, Y. Filali, H. El Khoukhi and A. Aarab, "Skin Cancer Diagnosis Using an Improved Ensemble Machine Learning model," 2020 International Conference on Intelligent Systems and Computer Vision (ISCV), 2020, pp. 1-5, doi: 10.1109/ISCV49265.2020.9204324.

- [33] S. Rahat Hassan, S. Afrogé and M. Binte Mizan, "Skin Lesion Classification Using Densely Connected Convolutional Network," 2020 IEEE Region 10 Symposium (TENSYMP), 2020, pp. 750-753, doi: 10.1109/TENSYMP50017.2020.9231041.
- [34] H. K. Kondaveeti and P. Edupuganti, "Skin Cancer Classification using Transfer Learning," 2020 IEEE International Conference on Advent Trends in Multidisciplinary Research and Innovation (ICATMRI), 2020, pp. 1-4, doi: 10.1109/ICATMRI51801.2020.9398388.
- [35] S. Bassi and A. Gomekar, "Deep Learning Diagnosis of Pigmented Skin Lesions," 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2019, pp. 1-6, doi: 10.1109/ICCCNT45670.2019.8944601.
- [36] C. Barata and J. S. Marques, "Deep Learning For Skin Cancer Diagnosis With Hierarchical Architectures," 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019), 2019, pp. 841-845, doi: 10.1109/ISBI.2019.8759561.
- [37] A. K. Waweru, K. Ahmed, Y. Miao and P. Kawan, "Deep Learning in Skin Lesion Analysis Towards Cancer Detection," 2020 24th International Conference Information Visualisation (IV), 2020, pp. 740-745, doi: 10.1109/IV51561.2020.00130.
- [38] Adla, D., Reddy, G.V.R., Nayak, P. *et al.* Deep learning-based computer aided diagnosis model for skin cancer detection and classification. *Distrib Parallel Databases* (2021).
<https://doi.org/10.1007/s10619-021-07360-z>
- [39] W. jiahao, J. Xingguang, W. Yuan, Z. Luo and Z. Yu, "Deep Neural Network for Melanoma Classification in Dermoscopic Images," 2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE), 2021, pp. 666-669, doi: 10.1109/ICCECE51280.2021.9342158.
- [40] M. A. Anjum, J. Amin, M. Sharif, H. U. Khan, M. S. A. Malik and S. Kadry, "Deep Semantic Segmentation and Multi-Class Skin Lesion Classification Based on Convolutional Neural Network," in IEEE Access, vol. 8, pp. 129668-129678, 2020, doi: 10.1109/ACCESS.2020.3009276.
- [41] Alzubaidi L, Duan Y, Al-Dujaili A, Ibraheem IK, Alkenani AH, Santamaría J, Fadhel MA, Al-Shamma O, Zhang J. 2021. Deepening into the suitability of using pre-trained models of ImageNet against a lightweight convolutional neural network in medical imaging: an experimental study. *PeerJ Computer Science* 7:e715 <https://doi.org/10.7717/peerj-cs.715>
- [42] Rezvantalab, Amirreza & Safigholi, Habib & Karimijeshni, Somayeh. (2018). Dermatologist Level Dermoscopy Skin Cancer Classification Using Different Deep Learning Convolutional Neural Networks Algorithms. <https://doi.org/10.48550/arXiv.1810.10348>
- [43] 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>

- [44] C. Zhao, R. Shuai, L. Ma, W. Liu, D. Hu and M. Wu, "Dermoscopy Image Classification Based on StyleGAN and DenseNet201," in IEEE Access, vol. 9, pp. 8659-8679, 2021, doi: 10.1109/ACCESS.2021.3049600.
- [45] C. Zhao, R. Shuai, L. Ma, W. Liu, D. Hu and M. Wu, "Dermoscopy Image Classification Based on StyleGAN and DenseNet201," in IEEE Access, vol. 9, pp. 8659-8679, 2021, doi: 10.1109/ACCESS.2021.3049600.
- [46] I. K. E. Purnama et al., "Disease Classification based on Dermoscopic Skin Images Using Convolutional Neural Network in Teledermatology System," 2019 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM), 2019, pp. 1-5, doi: 10.1109/CENIM48368.2019.8973303.
- [47] A. Demir, F. Yilmaz and O. Kose, "Early detection of skin cancer using deep learning architectures: resnet-101 and inception-v3," 2019 Medical Technologies Congress (TIPTEKNO), 2019, pp. 1-4, doi: 10.1109/TIPTEKNO47231.2019.8972045.
- [48] Filali, Y., EL Khoukhi, H., Sabri, M.A. *et al.* Efficient fusion of handcrafted and pre-trained CNNs features to classify melanoma skin cancer. *Multimed Tools Appl* **79**, 31219–31238 (2020). <https://doi.org/10.1007/s11042-020-09637-4>
- [49] Mary Adewunmi. Enhanced Melanoma Classifier with VGG16-CNN. *ScienceOpen Posters*. DOI: 10.14293/S2199-1006.1.SOR-.PPN1W6K.v1
- [50] Khan MA, Muhammad K, Sharif M, Akram T, Albuquerque VHC. Multi-Class Skin Lesion Detection and Classification via Teledermatology. *IEEE J Biomed Health Inform*. 2021 Dec;25(12):4267-4275. doi: 10.1109/JBHI.2021.3067789. Epub 2021 Dec 6. PMID: 33750716.
- [51] Z. Yu *et al.*, "Melanoma Recognition in Dermoscopy Images via Aggregated Deep Convolutional Features," in IEEE Transactions on Biomedical Engineering, vol. 66, no. 4, pp. 1006-1016, April 2019, doi: 10.1109/TBME.2018.2866166.
- [52] R. Zhang, "Melanoma Detection Using Convolutional Neural Network," 2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE), 2021, pp. 75-78, doi: 10.1109/ICCECE51280.2021.9342142.
- [53] Almaraz-Damian J-A, Ponomaryov V, Sadovnychiy S, Castillejos-Fernandez H. Melanoma and Nevus Skin Lesion Classification Using Handcraft and Deep Learning Feature Fusion via Mutual Information Measures. *Entropy*. 2020; 22(4):484. <https://doi.org/10.3390/e22040484>
- [54] Mesut Togaçar, Zafer Cömert, Burhan Ergen, Intelligent skin cancer detection applying autoencoder, MobileNetV2 and spiking neural networks, Chaos, Solitons & Fractals, Volume 144, 2021, 110714, ISSN 0960-0779, <https://doi.org/10.1016/j.chaos.2021.110714>.
- [55] Tschandl, P., Rinner, C., Apalla, Z. *et al.* Human–computer collaboration for skin cancer recognition. *Nat Med* **26**, 1229–1234 (2020). <https://doi.org/10.1038/s41591-020-0942-0>

- [56] P. Tang, Q. Liang, X. Yan, S. Xiang and D. Zhang, "GP-CNN-DTEL: Global-Part CNN Model With Data-Transformed Ensemble Learning for Skin Lesion Classification," in IEEE Journal of Biomedical and Health Informatics, vol. 24, no. 10, pp. 2870-2882, Oct. 2020, doi: 10.1109/JBHI.2020.2977013.
- [57] "OpenCV: Geometric Transformations of Images." *OpenCV: Geometric Transformations of Images*, docs.opencv.org,
https://docs.opencv.org/4.x/da/d6e/tutorial_py_geometric_transformations.html.
- [58] "OpenCV: Morphological Transformations." *OpenCV: Morphological Transformations*, docs.opencv.org, https://docs.opencv.org/4.x/d9/d61/tutorial_py_morphological_ops.html.
- [59] "OpenCV: Image Thresholding." *OpenCV: Image Thresholding*, docs.opencv.org,
https://docs.opencv.org/4.x/d7/d4d/tutorial_py_thresholding.html.
- [60] "OpenCV: Image Inpainting." *OpenCV: Image Inpainting*, docs.opencv.org,
https://docs.opencv.org/4.x/df/d3d/tutorial_py_inpainting.html.
- [61] Telea, Alexandru. "An image inpainting technique based on the fast marching method." Journal of graphics tools 9.1 (2004): 23-34.
- [62] "A Comparison of 4 Popular Transfer Learning Models." Analytics India Magazine, analyticsindiamag.com, 1 Sept. 2021, <https://analyticsindiamag.com/a-comparison-of-4-popular-transfer-learning-models/>.
- [63] Lin, Yuan-Pin; Jung, Tzzy-Ping (27 June 2017). "Improving EEG-Based Emotion Classification Using Conditional Transfer Learning". Frontiers in Human Neuroscience. 11: 334. doi:10.3389/fnhum.2017.00334. PMC 5486154. PMID 28701938. CC-BY icon.svg Material was copied from this source, which is available under a Creative Commons Attribution 4.0 International License.
- [64] O. Russakovsky et al., ImageNet large scale visual recognition challenge. Int. J. Comput. Vis. (2015)
- [65] Team, Keras. "Keras Documentation: Keras Applications." Keras Applications, keras.io, <https://keras.io/api/applications/>.
- [66] arXiv:1512.03385
- [67] arXiv:1608.06993
- [68] arXiv:1512.00567
- [69] A.G. Howard et al., MobileNets. arXiv Prepr. arXiv1704.04861 (2017)
- [70] <https://doi.org/10.48550/arXiv.1905.11946>
- [71] <https://arxiv.org/abs/1801.04381>
- [72] <https://arxiv.org/abs/1807.11626>
- [73] <https://doi.org/10.48550/arXiv.2104.00298>

- [74] Gupta, S. and Tan, M. Efficientnet-edgetpu: Creating accelerator-optimized neural networks with automl. <https://ai.googleblog.com/2019/08/efficientnetedgetpu-creating.html>, 2019.
- [75] https://openaccess.thecvf.com/content_cvpr_2017/papers/Chollet_Xception_Deep_Learning_CV_PR_2017_paper.pdf
- [76] <https://arxiv.org/abs/1610.02357>
- [77] Sudha, V. & Ganeshbabu, Dr. (2020). A Convolutional Neural Network Classifier VGG-19 Architecture for Lesion Detection and Grading in Diabetic Retinopathy Based on Deep Learning. Computers, Materials & Continua. 66. 827-842. 10.32604/cmc.2020.012008.
- [78] Brown G. (2011) Ensemble Learning. In: Sammut C., Webb G.I. (eds) Encyclopedia of Machine Learning. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-30164-8_252
- [79] Raschka, Sebastian. "EnsembleVoteClassifier: A Majority Voting Classifier - Mlxtend." *EnsembleVoteClassifier: A Majority Voting Classifier - Mlxtend*, rasbt.github.io, http://rasbt.github.io/mlxtend/user_guide/classifier/EnsembleVoteClassifier/
- [80] Nelder, John A.; R. Mead (1965). "A simplex method for function minimization". *Computer Journal*. 7 (4): 308–313. [doi:10.1093/comjnl/7.4.308](https://doi.org/10.1093/comjnl/7.4.308)
- [81] Spendley, W.; Hext, G. R.; Hinsworth, F. R. (1962). "Sequential Application of Simplex Designs in Optimisation and Evolutionary Operation". *Technometrics*. 4 (4): 441–461. [doi:10.1080/00401706.1962.10490033](https://doi.org/10.1080/00401706.1962.10490033)
- [82] "Tf.Keras.Optimizers.Schedules.CosineDecay | TensorFlow Core v2.9.1." *TensorFlow*, www.tensorflow.org, https://www.tensorflow.org/api_docs/python/tf/keras/optimizers/schedules/CosineDecay
- [83] "Tf.Keras.Preprocessing.Image.ImageDataGenerator | TensorFlow Core v2.9.1." *TensorFlow*, www.tensorflow.org, https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/image/ImageDataGenerator
- [84] Nahata, Hardik & Singh, Satya. (2020). Deep Learning Solutions for Skin Cancer Detection and Diagnosis. 10.1007/978-3-030-40850-3_8.
- [85] A. Baratloo, M. Hosseini, A. Negida, G. El Ashal, Part 1: Simple Definition and Calculation of Accuracy, Sensitivity and Specificity. (Emergency, Tehran, Iran, 2015)
- [86] Iqbal, I., Younus, M., Walayat, K., Kakar, M. and Ma, J., 2021. Automated multi-class classification of skin lesions through deep convolutional neural network with dermoscopic images. *Computerized Medical Imaging and Graphics*, 88, p.101843.
- [87] Srinivasu, P., SivaSai, J., Ijaz, M., Bhoi, A., Kim, W. and Kang, J., 2021. Classification of Skin Disease Using Deep Learning Neural Networks with MobileNet V2 and LSTM. *Sensors*, 21(8), p.2852.

- [88] Mijwil, M., 2021. Skin cancer disease images classification using deep learning solutions. *Multimedia Tools and Applications*.
- [89] Albahar, M., 2019. Skin Lesion Classification Using Convolutional Neural Network With Novel Regularizer. *IEEE Access*, 7, pp.38306-38313.
- [90] Guan, Q., Wang, Y., Ping, B., Li, D., Du, J., Qin, Y., Lu, H., Wan, X. and Xiang, J., 2019. Deep convolutional neural network VGG-16 model for differential diagnosing of papillary thyroid carcinomas in cytological images: a pilot study. *Journal of Cancer*, 10(20), pp.4876-4882.
- [91] Hernández, J., Sucar, L. and Morales, E., 2014. Multidimensional hierarchical classification. *Expert Systems with Applications*, 41(17), pp.7671-7677.
- [92] Adly, M.M., Ghoneim, A.S. and Youssif, A.A., 2019. On the grading of diabetic retinopathies using a binary-tree-based multiclass classifier of cnns. *International Journal of Computer Science and Information Security (IJCSIS)*, 17(1).
- [93] Lorena, A. and de Carvalho, A., 2010. Building binary-tree-based multiclass classifiers using separability measures. *Neurocomputing*, 73(16-18), pp.2837-2845.

Appendix I (*Pretrained Models*)

ResNet

It was first introduced by kaiming He and his colleagues, before discussing the architecture and the inner details of the ResNet architecture let us discuss what was the problem that ResNet paper was trying to solve which is called degradation problem.

Degradation problem means the accuracy keeps increasing when the architecture gets deeper, it was known that when the models get deeper it will have a higher performance than shallow ones , but during 2015 ,when the CNN architecture goes deeper the error is higher but the reason was the not the over fitting problem or the gradient vanish problem as the batch normalization is already addressed.

To address what is the problem that really happened let us look from a math perspective, which proves that the shallower networks are just a subset of its counterpart deeper network, which means the deeper network can learn a sophisticated representation of mapping function between input and output and the shallow networks are just a subset of those solutions.

So if we have a neural network with 50 layers and the shallow network has 20 layers , so we can construct the deeper network based on the shallow network so the first 20 layers will be the shallow network and the rest of layers will be identity mapping so the big or deeper network should now be the same behavior as the shallow network , so from the mathematic perspective there are no reason for the degradation problem so the authors suggested that there are a difficulties in approximating identity mapping by multiple non-linear layer so the residual block fig() is introduced to solve this problem.

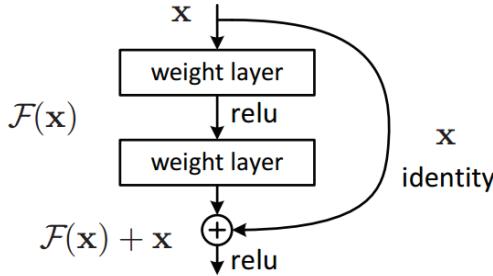


Fig.67

In degradation problem, we know that shallower networks perform better than the deeper counterparts that have few more layers added to them .so, why not skip these extra layers and at-least match the accuracy of the shallow sub networks.

So here $(x) = F(x) + x$, $H(x)$ is the Desired mapping (Original mapping, output) $F(x)$ represents the residual block, X represents the identity mapping or the input.

One of the notes here about the residual architecture that the vanishing gradient from the deeper network is solved from the shortcut connection.

Fig.68 shows a simple comparison between networks with plan structure and with residual structure to see how the shortcut connections results in a less error rate

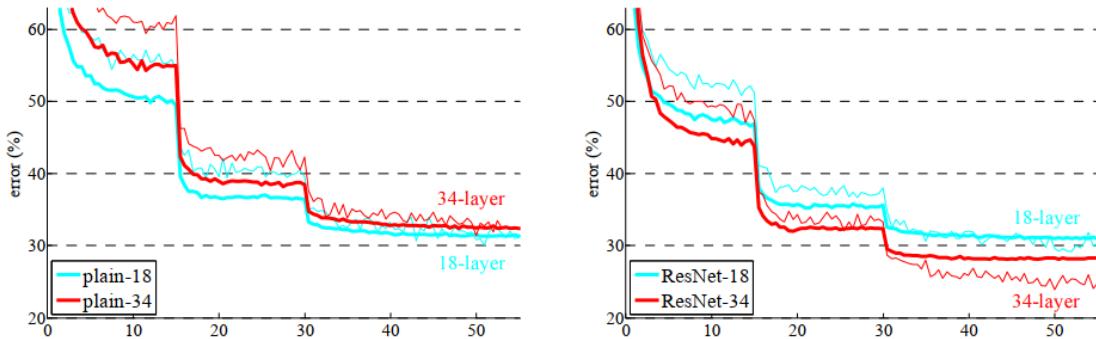


Fig.68 (Figures mentioned in the original paper)

DenseNet appendix

Densely connected convolutional neural networks are introduced in 2018 by Gao Huang basically its idea is inspired from the ResNet architecture which we have discussed earlier.

An Advantage of ResNet is that the gradient can flow directly through identity function from later layers to the earlier layers, however, the identity function and the output of $H(L)$ are combined by summation, which may impede the information flow in the network. so, the dense net architecture is going to solve this problem by making every two layers in the network connected to each other, so each layer receives signals from all its preceding layers so here the input is concatenated by channel wise concatenation.

Fig.69 shows how connections are different according to different deep CNN architectures, in traditional CNNs we have L layers with L Connections ,in Dense Net the number of connections will be calculated using this formula $L * (L + 1)/2$

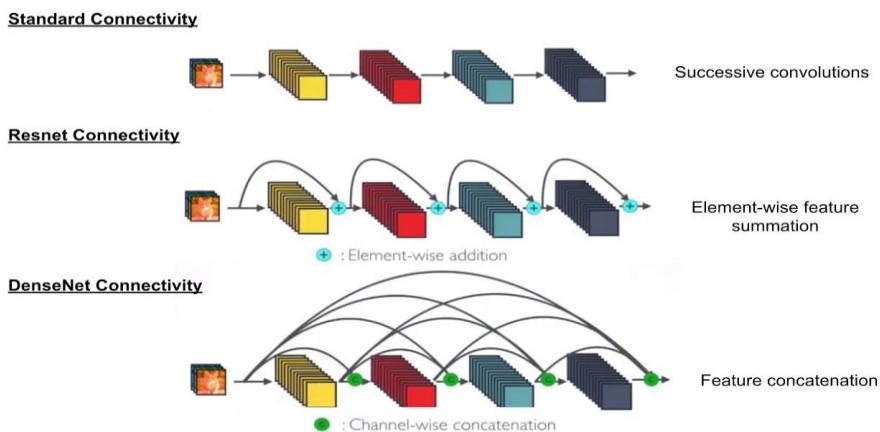
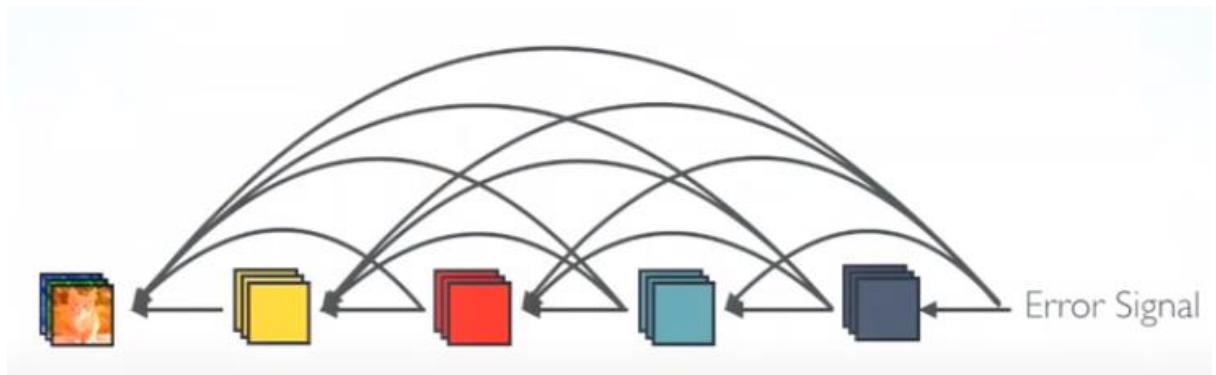


Fig.69

show different connectivity

So, the main advantages of the Dense Net Architecture can be divided into four main points

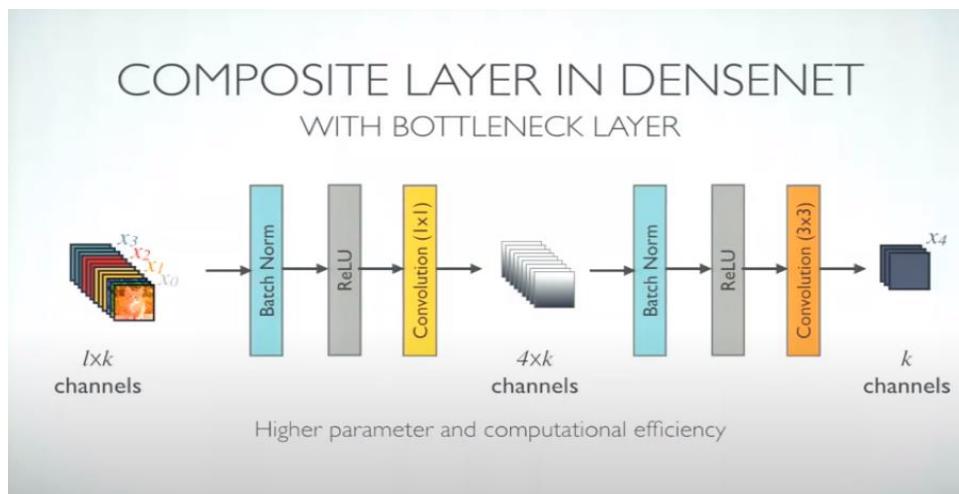
- Alleviate the vanishing-gradient problem
- Strengthen feature propagation or strong gradient flow fig.69
- Encourage feature reuse
- Substantially reduce the number of params (will be discussed later in this section)



For the first two points the reason why, they are an advantage of this architecture is that each layer has direct access to the gradients from the loss function and the original input signal, leading to an implicit deep supervision.

But as every layer has a direct access to the preceding layers so there will be a low information bottleneck in the network, so the authors make each layer much thinner and then obtaining much compact model and this gives high parameter and computational efficiency and to do though every layer produces a fixed number of feature maps.

Between every two dense blocks there is Batch Norm layer, Relu, and Convolutional(3*3),but by keep concatenating features in the network the input to deeper layers will become very wide and this may introduce too much computations in deeper layers ,so to address this problem a bottle neck layer is added also called transition layer to reduce the number of feature to 4k using cheap convolution layer with 1*1 with will results in increasing the parameters efficiency fig()



One of the other main strengthen points in Dense Net architecture that differs from the standard connectivity that the classifier uses features of all complexity levels not the high-level features at the end of the classifiers or the low level features at the early layers of network fig.70

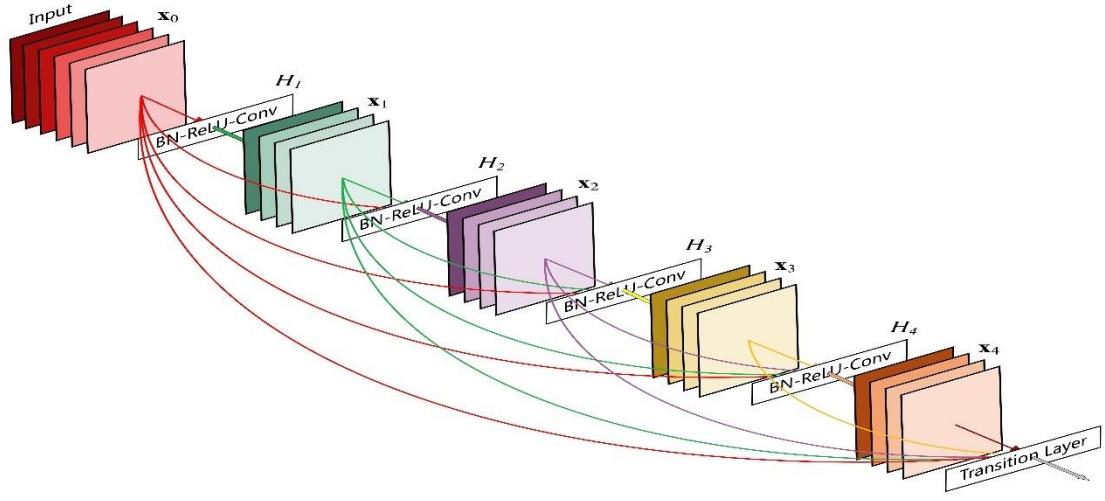


Fig.70 (shows overview on dense architecture and the bottleneck layer)

Inception V3

Inception v3 mainly focuses on burning less computational power by modifying the previous Inception architectures. This idea was proposed in the [], published in 2015. It was co-authored by Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, and Jonathon Shlens.

In comparison to VGGNet, Inception Networks (GoogLeNet/Inception v1) have proved to be more computationally efficient, both in terms of the number of parameters generated by the network and the economic cost incurred (memory and other resources). If any changes are to be made to an Inception Network, care needs to be taken to make sure that the computational advantages aren't lost. Thus, the adaptation of an Inception network for different use cases turns out to be a problem due to the uncertainty of the new network's efficiency. In an Inception v3 model, several techniques for optimizing the network have been put suggested to loosen the constraints for easier model adaptation. The techniques include factorized convolutions, regularization, dimension reduction, and parallelized computations.

Inception v3 Architecture

The architecture of an Inception v3 network is progressively built, step-by-step, as explained below:

1. **Factorized Convolutions:** this helps to reduce the computational efficiency as it reduces the number of parameters involved in a network. It also keeps a check on the network efficiency.
2. **Smaller convolutions:** replacing bigger convolutions with smaller convolutions definitely leads to faster training. Say a 5×5 filter has 25 parameters; two 3×3 filters replacing a 5×5 convolution has

only 18 ($3 \times 3 + 3 \times 3$) parameters instead. Fig.71

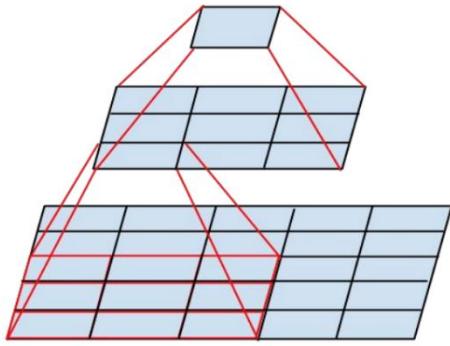


fig.71

In the middle we see a 3×3 convolution, and below a fully connected layer. Since both 3×3 convolutions can share weights among themselves, the number of computations can be reduced.

3. Asymmetric convolutions: A 3×3 convolution could be replaced by a 1×3 convolution followed by a 3×1 convolution. If a 3×3 convolution is replaced by a 2×2 convolution, the number of parameters would be slightly higher than the asymmetric convolution proposed. Fig.72

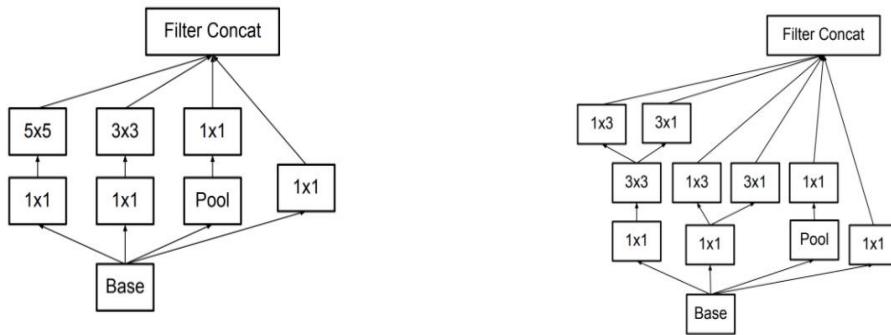


Fig.72 Asymmetric convolutional4.

4-Auxiliary classifier: an auxiliary classifier is a small CNN inserted between layers during training, and the loss incurred is added to the main network loss. In GoogLeNet auxiliary classifiers were used for a deeper network, whereas in Inception v3 an auxiliary classifier act as a regularizer.