Melanoma Skin Cancer Diagnosis Using Transfer Learning and Ensemble Learning

COMSATS University Islamabad, Pakistan
Muhammad Yahya Khan [1], Qasim Muhammad Ashraf [2]
FA17-ECE-027@ISBSTUDENT.COMSATS.EDU.PK [1]
FA17-ECE-032@ISBSTUDENT.COMSATS.EDU.PK [2]

Abstract—Early detection of Melanoma skin cancer is necessary for the patient because it can directly lead to the death of a person. If this cancer is detected in an early stage, then it can be cured easily. Image classification using machine learning is an effective technique to detect Melanoma using the images of lesions, it can help medical diagnostic centers in early detection of Melanoma. Machine learning techniques like Artificial neural network (ANN) and many more are used in classification of images. The objective is to classify benign and malignant cancer images using multiple neural network architectures and then use an ensemble network to find the best result out of them all.

I. INTRODUCTION

Advancement in technology and computer science has opened new doors for human beings. AI is a field of computer science which covers the broad idea of making intelligent machines using either both software and/or hardware which can act on their own without human intervention. ML is a field of AI using which we can create software that is able to predict an output based on the given input by training it. One of the techniques that can be used in ML is image classification, which is best suited for detecting skin cancer. It is a technique that uses a database of predetermined patterns to help us compare objects and then group them into categories. Remote sensing, robot navigation, and biological imaging are just a few of the uses for classification algorithms. Image classification allows for the categorization of pictures into benign and malignant categories. Supervised learning and unsupervised learning are two rule-based classification techniques used in image classification. We shall be able to anticipate the outcome or class of data in supervised learning, and a trained data set will be presented. Prior information is required before testing when using the supervised learning method, and this data must be obtained by the analyst. The user can detect faults and correct them with supervised learning, but it takes longer and costs more money, and the training data is chosen by the analyst. It may not involve all the conditions to detect the skin cancer. Supervised classification also involves human intervention. In unsupervised learning, no trained data will be provided but the classifier itself finds which category or class it belongs to. The user need not have any prior information, so no human intervention is required. Unsupervised classification is much faster when compared to supervised classification, but the accuracy is lower. There are no human errors in this procedure, and no prior knowledge is required.

II. BACKGROUND AND MOTIVATION

Skin is an important part of the human body; it protects us from harmful elements such as ultraviolet (UV) rays and help to maintain the body temperature. Melanoma is one of the most common type of skin cancer, and it begins in skin cells called melanocytes and the disease can start growing from anywhere on the body. Most moles do not develop into melanoma, but some do, and researchers have discovered genetic alterations in mole cells that may lead to the development of melanoma cells. UV light exposure is a primary risk factor for most melanomas. UV (ultraviolet) rays are mostly produced by sunlight. UV (ultraviolet) radiation can also be found in tanning beds and sunlamps. They come in a variety of colours, including brown, black, red, blue, and a blue-red combo. Melanoma is a skin cancer that only affects the top layer of the skin. The tumour is only a few millimetres to two millimetres thick, and the surface may or may not be fractured. Cancer cells can spread to other organs and tissues, including the liver, brain, and bones. Melanoma causes 55,500 cancer deaths annually which is 0.7% of all cancer deaths. The incidence and mortality rates of Melanoma differ from one country to another due to the variation of ethnic and racial groups [1]. It is critical to detect melanoma at an early stage. Statistics shows that the 5-year relative survival rate for people who has been diagnosed with Melanoma in an early stage is about 98%. However, about 20% to 50% of people having Melanoma in advanced stage will be alive 5 years after diagnosis [1].

The motivation here is to assist the medical diagnostic centres in detecting Melanoma at an early stage. It is critical to apply supporting imaging techniques in this situation since they have been found to improve and facilitate the diagnosing process. These techniques are build based on strategies invented by physicians to capture the melanoma at an early stage.

III. OBJECTIVES

Our aim is to distinguish between benign and malignant melanoma skin cancer, to apply transfer learning and ensemble learning techniques effectively using multiple Convolutional Neural Network architectures that are able to detect within a reasonable degree of accuracy and then apply ensemble technique using a new neural network with the input being predictions of all CNN architectures and compare the accuracy of the new network with the other CNN architectures used previously and then try to improve on the accuracy of our model.

IV. BLOCK DIAGRAM

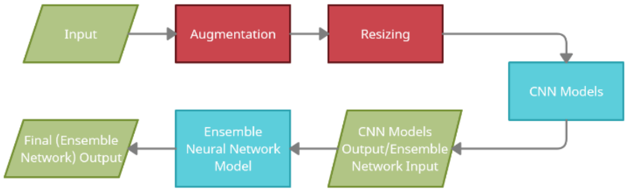

Input
Augmentation
Resizing
CNN Models
CNN Models Output/Ensemble Network Input
Ensemble Neural Network Model
Final (Ensemble Network) Output

Figure 1: Block Diagram

In this diagram we have explained the whole working of our project, first we have the inputs in the form of images, then we perform augmentation and after that we resize them according to the requirements of the CNN models.

When we have an output from the CNN models, we use it as input for our new model and the ensemble model gives us the final output in the form of benign or malignant melanoma.

V. METHODOLOGY

A. Dataset

A total of 33,126 dermoscopic training photos of unique benign and malignant skin lesions from over 2,000 patients are included in the dataset. Using a unique patient identity, each image is linked to one of these people. Histopathology was used to confirm all malignant diagnoses, while expert agreement, longitudinal follow-up, and histopathology were used to confirm benign diagnoses. A comprehensive article that describes all aspects of this dataset is available as a pre-print that has not yet been peer reviewed. The dataset was generated by the International Skin Imaging Collaboration (ISIC).[9]

After analysing the dataset, we found out that the images do not require segmentation, the dataset is already optimized for training models. The images in this dataset have different resolutions, so we needed to resize them for training different models.

When we used the ground truth given in the csv file, we found out that 32,542 images out of the 33,126 are of benign tumours and the remaining 584 images are of malignant tumours.

This ratio between malignant and benign images is not suitable for training neural networks efficiently.

If there is a big difference in ratio between classes, it creates class imbalance, and the neural network becomes biased.

In classification problems, it is better to have a 50-50 ratio in between classes or at least close to 50% to avoid class imbalance.

In our case, we have a ratio of 55-45 between malignant and benign images. We performed extensive augmentation on the 584 malignant images and increased their number to 40,880.

B. Pre-processing

Augmentation:

Image augmentation artificially creates training images using a variety of processing techniques or a combination of techniques, such as random rotation, shifts, shear, and flips, among others.

The dataset did not have an adequate number of images for malignant tumours, so we had to perform augmentation on the existing 584 images.

For this purpose, we used a community developed python library from GitHub: imgaug.[10]

We performed these operations on the malignant images' multiple times using different variables, sometimes with combining two operations.

On a single image, augmentation was performed a total number of 69 times. After performing these operations on 584 images, we got 40,880 images of malignant tumours.

In the following figure we can see a single image and its augmented variations:

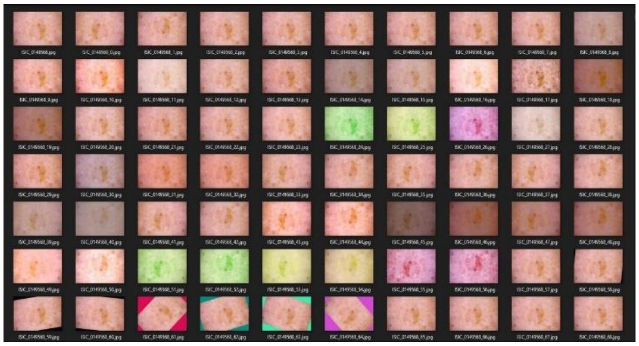


Figure 2: Augmentation

We used a python script to run this process. This augmented dataset was then used to train models.

The pre-built Keras image data generator also allows us to perform augmentation, but it would have cost us a lot of time as the image data generator would have to perform augmentation for every model separately. So, we instead used another library to perform augmentation on images first and then feed them into the neural networks.

C. Training (CNN)

There is a lot of stuff to consider before we start training these models.

First, the most important thing is the hardware we are using for training. In our case we have performed our training on a computer with the following specifications:

CPU: Xeon E5-1620 v2 (4 cores, 8 threads)

GPU: GTX 1080 (8 GB VRAM)

We have used CUDA to utilize the maximum potential of this GPU. GTX 1080 is a CUDA-enabled GPU from Nvidia, it has 2560 CUDA cores.

We could not train models like EfficientNet-B7 etc. due to the video memory limit of 8GB.

Some of these models require very high image sizes on input, for example: EfficientNet-B7 requires an image size of 600x600.

To run EfficientNet-B7 we would have to reduce the batch size to a small amount, that would make the model inefficient, reducing its accuracy.

As all these CNN models have been trained on the ImageNet dataset, they have been trained for 1000 classes. We use (include_top = False) to remove the top layer of the model, which allows to add a new output layer that has only 2 classes. Our objective is to classify between two classes (benign and malignant).

We have used the following functions while training models:

Activation: Sigmoid function has been used on the final layer of all models. The sigmoid function always returns a value between 0 and 1 so it is best suited for our classification problem.

Optimizer: Adam optimizer has been used for all models. Adaptive Moment Estimation is a technique for optimizing gradient descent algorithms.

When working with large problems with a lot of data or parameters, the method is extremely efficient. It is efficient and requires less memory. It's a combination of the 'gradient descent with momentum' and the 'root mean square propagation' algorithms on the surface.

Metric: Two functions have been used for all models, one of them is AUC.

The AUC (Area Under the Curve) of the ROC or PR curves is approximated. Binary classifiers' quality is measured by the AUC (Area under the curve) of the ROC (Receiver operating characteristic; default) or PR (Precision Recall) curves.

ROC-AUC and PR-AUC evaluate all a model's operational points, unlike accuracy and cross-entropy losses.

The other one is accuracy; it just calculates how often predictions equal labels.

Loss: Binary cross-entropy has been used for all models. It calculates the loss in cross-entropy between true and anticipated labels. Cross-entropy loss, commonly known as log loss, is a metric for evaluating the performance of a classification model whose output is a probability value between 0 and 1. As the predicted probability differs from the actual label, cross-entropy loss increases.

The following table shows image size, epochs, and batch size:

Model Name	**Image Size**	**Epochs**	**Batch Size**
EfficientNet-B0	**224 x 224**	**6**	**32**
EfficientNetB1	**240 x 240**	**6**	**32**
EfficientNetB2	**260 x 260**	**6**	**16**
EfficientNetB3	**300 x 300**	**6**	**16**
MobileNet	**256 x 256**	**3**	**32**
MobileNetV2	**256 x 256**	**3**	**32**
ResNet50	**224 x 224**	**6**	**32**
ResNet50V2	**224 x 224**	**6**	**32**
VGG16	**224 x 224**	**6**	**32**
VGG19	**224 x 224**	**6**	**32**
Xception	**299 x 299**	**6**	**16**

D. Training (Ensemble)

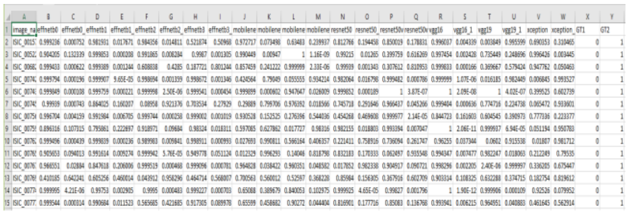
Ensemble Methods: Ensemble techniques are learning algorithms that create a group of classifiers and then categories incoming data points based on a (weighted) vote of their predictions. Ensemble techniques integrate several learning algorithms to produce greater prediction performance than any single learning algorithm.

All the CNN architectures that we have used to train models give output in the form of two nodes, one represents benign and the other represents malignant. The output is in the form [0.99, 0.01], one of these values is the probability of an image being benign and the other is the probability of it being malignant.

After we have trained all the 11 models, we then create notebooks on excel and store the predictions of all these 11 models on our dataset.

Then we merge all these notebooks into a single excel (.csv) file, after that we add the ground truth given in the dataset to this (.csv) file.

The file looks like this:



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
1	image_na	effnetb0	effnetb0_	effnetb1	effnetb1_	effnetb2	effnetb2_	effnetb3	effnetb3_	mobilene	mobilene	mobilene	mobilene	resnet50	resnet50_	resnet50v	resnet50v	vgg16	vgg16_1	vgg19	vgg19_1	xception	xception_	GT1	GT2
2	ISIC_0015	0.999236	0.000752	0.981931	0.017671	0.984356	0.014811	0.521874	0.50968	0.972717	0.073498	0.63483	0.239937	0.812766	0.194458	0.850019	0.178831	0.996037	0.004339	0.003849	0.995599	0.690353	0.310465	0	1
3	ISIC_0052	0.904205	0.132339	0.999853	0.000208	0.991865	0.008284	0.9987	0.001305	0.990449	0.00947	1	1.16E-09	0.99215	0.01265	0.399759	0.616269	0.997454	0.002428	0.735449	0.248696	0.996426	0.003445	0	1
4	ISIC_0068	0.999433	0.000622	0.999389	0.001244	0.608838	0.4285	0.187721	0.801244	0.857459	0.241222	0.999999	2.33E-06	0.99939	0.001343	0.307612	0.810953	0.999833	0.000166	0.369667	0.579424	0.947762	0.050463	0	1
5	ISIC_0074	0.999794	0.000196	0.999907	9.65E-05	0.998694	0.001359	0.998672	0.001346	0.424564	0.79049	0.055555	0.934214	0.982064	0.016798	0.999482	0.000786	0.999999	1.07E-06	0.016185	0.982449	0.006845	0.993527	0	1
6	ISIC_0074	0.999849	0.000108	0.999759	0.000221	0.999998	2.50E-06	0.999541	0.000454	0.999899	0.000602	0.947647	0.026009	0.999852	0.000189	1	3.87E-07	1	2.09E-08	1	4.02E-07	0.399525	0.602739	0	1
7	ISIC_0074	0.99939	0.000743	0.864025	0.160207	0.08958	0.921376	0.703534	0.27929	0.29889	0.799706	0.976392	0.018566	0.745718	0.291646	0.966437	0.045266	0.999404	0.000636	0.774716	0.224738	0.065472	0.933601	0	1
8	ISIC_0075	0.996704	0.004159	0.991984	0.006705	0.999744	0.000258	0.999002	0.001019	0.930528	0.152525	0.276396	0.544036	0.454268	0.469608	0.999977	2.14E-05	0.844723	0.161603	0.604545	0.390973	0.777336	0.223377	0	1
9	ISIC_0075	0.896316	0.107315	0.795861	0.222697	0.918971	0.09684	0.98324	0.018311	0.597085	0.627862	0.017727	0.98316	0.982155	0.018803	0.993394	0.007047	1	2.06E-11	0.999937	6.94E-05	0.051194	0.950783	0	1
10	ISIC_0076	0.999496	0.000439	0.999839	0.000236	0.989983	0.009841	0.998911	0.000993	0.027693	0.990811	0.566164	0.406357	0.221411	0.758916	0.736094	0.261747	0.96255	0.037344	0.0602	0.915538	0.01807	0.981712	0	1
11	ISIC_0076	0.905653	0.094013	0.991614	0.009274	0.999942	5.76E-05	0.949778	0.051124	0.012329	0.996293	0.14046	0.818798	0.832183	0.170333	0.062497	0.935548	0.994347	0.007477	0.982247	0.018063	0.212249	0.79535	0	1
12	ISIC_0076	0.966551	0.02884	0.847618	0.206006	0.999519	0.000468	0.999096	0.000781	0.964828	0.038422	0.960351	0.048582	0.017852	0.982338	0.904917	0.090721	0.998296	0.002205	2.40E-06	0.999997	0.336205	0.675447	0	1
13	ISIC_0076	0.410185	0.642241	0.605256	0.460014	0.043912	0.958296	0.464714	0.568007	0.700563	0.560012	0.52597	0.368228	0.85984	0.156305	0.367916	0.602709	0.903314	0.108325	0.632288	0.374715	0.182754	0.819612	0	1
14	ISIC_0077	0.999995	4.21E-06	0.99753	0.002905	0.9995	0.000483	0.999227	0.000703	0.65088	0.389679	0.840053	0.102975	0.999925	4.65E-05	0.99827	0.001796	1	1.90E-12	0.999906	0.000109	0.92526	0.079952	0	1
15	ISIC_0077	0.999544	0.000314	0.990684	0.011523	0.565685	0.421685	0.917305	0.089978	0.65599	0.458682	0.90272	0.044404	0.816901	0.177716	0.85083	0.136768	0.993941	0.006215	0.964951	0.040883	0.461645	0.562914	0	1

Figure 3: Final notebook form

As we can see from the figure above, we have 22 columns for 11 models, 1st column (A) contains image names, then 2 columns for each model and last 2 columns (X and Y) for ground truth.

Now that we have the outputs of all the 11 models that we have run, we are going to run a custom model using data from this notebook with the outputs of those 11 models as its inputs and malignant or benign classification as output.

The CNN model has 1 input layer and 6 dense layers. The following are the details:

- Input layer: It has 22 nodes
- 1st layer (Dense): It has 32 nodes
- 2nd layer (Dense): It has 64 nodes
- 3rd layer (Dense): It has 128 nodes
- 4th layer (Dense): It has 256 nodes
- 5th layer (Dense): It has 512 nodes
- Output layer (Dense): It has 2 nodes

VI. RESULTS

The following table shows the accuracies and losses of CNN models:

Network Name	**Train Loss**	**Validation Loss**	**Train Accuracy**	**Validation Accuracy**
EfficientNet-B0	**0.0130**	**1.5180**	**0.9956**	**0.7431**
EfficientNet-B1	**0.0101**	**1.7929**	**0.9968**	**0.7620**
EfficientNet-B2	**0.0103**	**2.8518**	**0.9998**	**0.7190**
EfficientNet-B3	**0.0089**	**1.8636**	**0.9971**	**0.7430**
MobileNet	**0.0174**	**0.8588**	**0.9949**	**0.7844**
MobileNetV2	**0.0280**	**0.9225**	**0.9900**	**0.8098**
ResNet50	**0.0124**	**1.6049**	**0.9964**	**0.7414**
ResNet50V2	**0.0127**	**1.8766**	**0.9957**	**0.7035**
VGG16	**0.0233**	**1.4696**	**0.9926**	**0.6960**
VGG19	**0.0269**	**2.0235**	**0.9908**	**0.7039**
Xception	**0.0073**	**1.3922**	**0.9977**	**0.7296**
Ensemble	**0.0914**	**0.1060**	**0.9645**	**0.9580**

The best result out of all CNN models is accuracy of 80.98% using mobilenet, and the worst accuracy is of 69.60% using vgg16. An average accuracy of 73.96% could be achieved just by using CNN models while the accuracy of our ensemble network is 95.80%. So, we can see that the accuracy was improved by 21.84%.

VII. CONCLUSION

We have learned a lot throughout the course of our research, Machine Learning is a field with huge potential and vast number of applications. Transfer learning is a very important concept that makes Machine Learning applications a lot easier for us. We have used multiple CNN architectures to classify images and then used an Ensemble network to classify these images. The best accuracy we could get out of all CNN architectures was around 80% but for the Ensemble network, we got an accuracy of around 95%. To conclude, we found out that using Ensemble learning technique we can get better results as compared to using just a single model.

REFERENCES

[1] Dirk Schadendorf, Alexander CJ van Akkooi, Carola Berking, Klaus G Griewank, Ralf Gutzmer, Axel Hauschild, Andreas Stang, Alexander Roesch, and Selma Ugurel. Melanoma. The Lancet, 392(10151):971–984, 2018

[2] Rotemberg, V., Kurtansky, N., Betz-Stablein, B., Caffery, L., Chousakos, E., Codella, N., Combalia, M., Dusza, S., Guitera, P., Gutman, D., Halpern, A., Helba, B., Kittler, H., Kose, K., Langer, S., Lioprys, K., Malvehy, J., Musthaq, S., Nanda, J., Reiter, O., Shih, G., Stratigos, A., Tschandl, P., Weber, J. & Soyer, P. A patient-centric dataset of images and metadata for identifying melanomas using clinical context. Sci Data 8, 34 (2021). https://doi.org/10.1038/s41597-021-00815-z

[3] Alexander Jung, imgaug, (2020), GitHub Repository, https://github.com/aleju/imgaug

[4] Shivangi Jain, Vandana jagtap, Nitin Pise, Computer Aided Melanoma Skin Cancer Detection Using Image Processing, Procedia Computer Science, Volume 48, Pages 735-740, ISSN 1877-0509,2015.

[5] Nayana Banjan, Prajkta Dalvi, Neha Prakash Athavale, Melanoma Skin Cancer Detection by Segmentation and Feature Extraction using combination of OTSU and STOLZ Algorithm Technique,2017.

[6] Joanna Jaworek-Korjakowska, Paweł Kłeczek, "Automatic Classification of Specific Melanocytic Lesions Using Artificial Intelligence", BioMed Research International, vol. 2016, Article ID 8934242, 17 pages, 2016. https://doi.org/10.1155/2016/8934242

[7] Goyal, M., Knackstedt, T., Yan, S., & Hassanpour, S. (2020). Artificial intelligence-based image classification for diagnosis of skin cancer: Challenges and opportunities. Computers in Biology and Medicine, 104065

[8] Nurshazlyn Mohd Aszemi and P.D.D Dominic, “Hyperparameter Optimization in Convolutional Neural Network using Genetic Algorithms” International Journal of Advanced Computer Science and Applications (IJACSA), 10(6), 2019. http://dx.doi.org/10.14569/IJACSA.2019.0100638

[9] L. Bi, J. Kim, E. Ahn, A. Kumar, M. Fulham and D. Feng, "Dermoscopic Image Segmentation via Multistage Fully Convolutional Networks," in IEEE Transactions on Biomedical Engineering, vol. 64, no. 9, pp. 2065-2074, Sept. 2017, doi: 10.1109/TBME.2017.2712771.

[10] Gessert, Nils, et al. "Skin lesion classification using ensembles of multi-resolution EfficientNets with meta data." MethodsX 7 (2020): 100864.

[11] A. H. Shahin, A. Kamal and M. A. Elattar, "Deep Ensemble Learning for Skin Lesion Classification from Dermoscopic Images," 2018 9th Cairo International Biomedical Engineering Conference (CIBEC), Cairo, Egypt, 2018, pp. 150-153, doi: 10.1109/CIBEC.2018.8641815