Department of Electrical and Computer Engineering
North South University

Senior Design Project 499B Report

Automatic Skin Cancer Detection System Using Deep Learning Approach

Tonmoy Debnath	**1921254042**
Imran Hossain	**1721398642**
Md Naser Bin Hossain	**1912620642**
Afif Bin Jashim	**1620853643**

Faculty Advisor:

Riasat Khan

Assistant Professor

ECE Department

Fall, 2022

Automatic Skin Cancer Detection System Using Deep Learning Approach

Individual Contribution Table

Section	Contributing Member Name	
IEEE/LaTEX formatting	Imran Hossain	
Turnitin check	Imran Hossain	
Grammarly check	Imran Hossain	Grammarly score-92
Abstract	Tonmoy Debnath	
Keywords	Tonmoy Debnath	
Introduction Motivation	Afif BIn Jashim	
Paper Review 1	Tonmoy Debnath	Melanoma Skin Cancer Detection Using Deep Learning and Advanced Regularizer
Paper Review 2	Imran Hossain	Deep Convolutional Neural Network for Melanoma Detection using Dermoscopy Images
Paper Review 3	Md Naser Bin Hossain	Detection Of Skin Cancer Using Deep Neural Networks.
Paper Review 4	Afif Bin Jashim	Skin Cancer Classification Using Image Processing and Machine Learning
Paper Review 5	Afif Bin Jashim	Melanoma skin cancer detection using deep learning and classical machine learning

		techniques: A hybrid approach
Paper Review 6	Tonmoy Debnath	Skin Cancer Detection Using Combined Decision of Deep Learners
Paper Review 7	Imran Hossain	Early detection of skin cancer using deep learning architectures: resnet-101 and inception-v3
Paper Review 8	Md Naser Bin Hossain	Deep Learning-Based System for Automatic Melanoma Detection
Introduction Second Last Paragraph	Imran Hossain	
Introduction Contribution	Tonmoy Debnath	
Introduction Last Paragraph	Afif Bin Jashim	
Proposed System	Tonmoy Debnath Md Naser Bin Hossain	
Results and Discussion	Tonmoy Debnath Md Naser Bin Hossain	
Figure title and format	Afif Bin Jashim	
Equations format	Imran Hossain Md Naser Bin Hossain	
Conclusions	Imran Hossain	
References Formatting in IEEE format	Imran Hossain Afif Bin Jashim	

Automatic Skin Cancer Detection System Using Deep Learning Approaches

Tonmoy Debnath
Electrical and Computer Engineering
North South University
Dhaka, Bangladesh
tonmoy.debnath@northsouth.edu

Md Naser Bin Hossain
Electrical and Computer Engineering
North South University
Dhaka, Bangladesh
naser.hossain@northsouth.edu

Imran Hossain
Electrical and Computer Engineering
North South University
Dhaka, Bangladesh
imran.hossain09@northsouth.edu

Afif Bin Jashim
Electrical and Computer Engineering
North South University
Dhaka, Bangladesh
afif.jashim@northsouth.edu

***Abstract*—Melanoma skin cancer can be detected traditionally using epiluminescence microscopy or dermoscopy by oncologists and dermatologists. However, there is a system like melanoma detection which can easily identify melanoma by assessing dermoscopic pictures with the aid of deep learning techniques. In that case, It will be easy to detect skin cancer within a few seconds. In this paper, an automatic skin cancer detection system has been proposed using a deep learning approach. We used the Harvard Dataverse dataset named HAM10000, which contains 10,015 dermoscopic pictures. We applied some traditional preprocessing techniques to get average color information and normalize all color channel information. Then we reshape images and obtain data for classification. We also used data augmentation to prevent overfitting. Finally, we implemented the Convolutional Neural Network, MobileNetV2, Visual Geometry Group 16, Dense Convolutional Network 201, Roboflow, vision Transformers, and knowledge distillation models to classify skin cancer. The Convolutional Neural Network, Visual Geometry Group 16, Dense Convolutional Network 201, MobileNetV2, Roboflow (ImageNet), vision transformer, and knowledge distillation models obtained training accuracies of 78%, 89%, 98%, 91%, 81%, 85%, and 89% respectively. Finally, the Dense Convolutional Network 201 approach attained better accuracy, 98% on training, 85% on validation, and 84% on the test samples.**

Keywords*—*Augmentation, Convolutional Neural Network, Dense Convolutional Network 201, Melanoma, Preprocessing, Visual Geometry Group

I. Introduction

Medical professionals utilize dermoscopy or epiluminescence microscopy to find melanoma skin cancer. Our system is melanoma detection which can easily identify melanoma by assessing dermoscopic pictures with the aid of deep learning techniques. We used two dermoscopic images, one set is about malignant, and another is about benign. We collected a dataset from ISIC (International Skin IMAGE Collaboration). This dataset contains 10015 images. Unfortunately, melanoma cases have increased globally during the past 40 years [1]. Melanoma accounts for around 60% of skin cancer mortality and seems to be detected in 1% of malignant tumors [1]. Malignant melanoma incidences have tripled during the past three decades in Europe and the United States [2]. The expenditure on medicating skin cancer was 3.6 billion from 2002 to 2006; however, from 2007 to 2011, the yearly cost increased to 8.1 billion [3]. As a result, melanoma's death rate is growing daily. Melanoma, which spreads more quickly than other kinds of cancer to other organs, such as the brain and skeletons, is treatable if discovered early. However, treatments grow more difficult, and the likelihood of recovery nearly disappears if it spreads to other body parts and the skin. Therefore, it is important for the early detection of malignant melanoma. That's why we created this system to detect malignant melanoma easily. We use Convolutional Neural Network, Visual Geometry Group 16, and Dense Convolutional Network 201 models to obtain better accuracy.

Melanoma skin cancer inspection requires highly qualified dermatologists and oncologists, which takes time and is expensive. In addition, recognizing skin cancer lesions require more training and expertise. Deep learning has thus been utilized to research automatic skin cancer detection in recent years. Some recent works on automatic skin cancer detection have been discussed in the following paragraphs.

For instance, in [4], the authors used deep learning models to detect skin cancer (Melanoma) using two dermoscopic images, one set is about malignant with 1497 images, and another is about benign with 18,00 images. They collected a dataset from ISIC (International Skin IMAGE Collaboration) containing resized images of 224×224 resolution. In preprocessing, they have used initializing, creating train, validation, and test sets, normalization, one-hot encoding, removing noise, and data augmentation. Finally, they used CNN models to detect skin cancer. In the result part, the authors obtained 93.58% accuracy in the CNN model.

In [5], Kaur and inha developed a fast and accurate classifier for an automatic skin cancer detection system using deep learning approaches. First, Dataset was collected from a Canadian hospital which consists of 2,150 malignant or benign images. Next, data augmentation and dropout techniques were used in the preprocessing step. Next, the authors used the Alex Net and VGG16 models to compare the results. Finally, the authors achieved 0.8295 accuracies from their model.

In [6], Rahi et al. introduced an automatic skin cancer detection model with a HAM10000 skin lesions image dataset from the ISIC repository. In this work, the authors applied various deep learning models, e.g., CNN, VGG16, ResNet, and DenseNet algorithms, to obtain better and more accurate findings. For example, the authors observed that the validation accuracy of the CNN model is 0.79. On the other hand, the authors found the validation accuracy is 0.90 from both ResNet 50 and DenseNet121 models and also observed the validation accuracy of the VGG 11_BN model is 0.85.

In [7], Javaid-Masot et al. proposed a system of detecting melanoma using picture classification and segmentation techniques. In this paper, the authors proposed an automatic system to determine dermoscopic images with the help of machine learning. The authors used linear scaling to unit variance to find the results. This paper collected images from the ISIC database, comprising RGB dermoscopic images. The images were preprocessed before uploading them to the system. But the authors found that dataset images had an imbalance as approximately 80% of images were benign and 20% were malignant. Finally, the authors found that accuracy for Quadratic discrimination, SVM, and random classifiers was 85%, 88%, and 93%, respectively.

In [8], the authors developed a Melanoma skin cancer detection system for better self-examination using deep learning and machine learning. In this experiment, a public dataset was used from ISIC (International Skin Imaging Collaboration) size of 23000 images, and only 640 photos of size 124×124 were used. Before applying any model, data preprocessing is important. So, the first case is removing hair. The author used the Color enhancement technique, 2-D derivatives of Gaussian for hair removal and segmentation. For the highest accuracy, the author uses Convolutional neural networks, conventional machine learning classifiers, and the appearance of a skin lesion. In this experiment, the author applied KNN and SVM CNN. The lowest accuracy achieved for KNN is only 57.3%, and SVM gives higher accuracy 71.8%. Finally, CNN provides the highest accuracy of 85.5%.

In [9], the authors applied three deep-learning models on a public dataset with 25000 images to identify skin cancer. All of the dataset's photographs have been downsized to 224x224. In the training phase, 80% of the photos were utilized. To avoid overfitting, they applied data augmentation techniques. They employed VGG16, proposed-ensemble, ResNet, and CapsNet. Based on the results, the proposed-ensemble model has the highest accuracy (93.5%).

In [10], Demir and colleagues used two deep-learning models using a public dataset of 3097 pictures to detect skin cancer. The images in the collection have been reduced to 224x224 pixels in size. Two thousand four hundred thirty-seven images were used in the training phase. They used data augmentation approaches to prevent overfitting. They have applied ResNet101 and InceptionV3 to detect skin cancer. The InceptionV3 model has the highest accuracy (87.42%).

In [11], Adegun and Viriri introduced Automatic Melanoma Detection using deep learning approaches with some updated models. Melanoma is the deadliest variety of skin cancer. To overcome these issues, the authors of this paper proposed a deep learning-based method for automatically recognizing and segmenting melanoma lesions. A lesion classifier is a revolutionary method that has been created to categorize skin lesions into melanoma and non-melanoma based on the results of pixel-wise classification. The most extensively researched cutting-edge literature techniques, including FrCN, CDNN, FCN, and UNET, were suggested for use in a few models. The accuracy and dice coefficient on the PH2 and the ISIC 2017 datasets were 0.95, 0.92, and 0.95, 0.93, respectively.

We have examined a variety of publications and articles on the topic of implementing deep learning to identify melanoma. Most of them worked on older deep-learning algorithms to identify melanoma using images. In addition, some papers omitted the data augmentation portion. Because of this, we have tried applying several new models to our chosen dataset and applied data augmentation to all of the models.

In this research, we used dermoscopic pictures and deep learning approaches to identify melanoma skin cancer disease. We used the 10,015 dermoscopic images from the HAM10000 dataset from Harvard Dataverse. In preprocessing, we applied to reshape techniques, normalization, and data augmentation to prevent overfitting. We used a total of seven neural network models to detect skin cancer. In addition, Convolutional Neural Network, Visual Geometry Group 16, Dense Convolutional Network 201, MobileNetV2, Roboflow (ImageNet), vision Transformers, and knowledge distillation have been applied. Finally, Dense Convolutional Network 201 obtained better accuracy than other models.

Deep learning is used in this paper to detect melanoma skin cancer. This work has made the following notable contributions:

- A major contribution of this project is to resize all the images of the HAM10000 dataset from Harvard Dataverse, which contains 100,15 dermoscopic images, into 224×224. The dataset includes seven classes.
- Convolutional Neural Network, Visual Geometry Group 16, Dense Convolutional Network 201, MobileNet, Roboflow (ImageNet), Vision Transformer, and Knowledge Distillation have been applied to the dataset to classify melanoma skin cancer detection.
- To prevent overfitting, data augmentation has been applied to every model.
- To accomplish fast detection with real-time data, an Android application has been developed utilizing the MobileNet architecture.

This research's novelty is using old and new deep learning models to construct an android app that automatically detects skin cancer using the HAM10000 dataset.

The proposed system, which includes a dataset, preprocessing, and deep learning models, has been detailed in Section II, along with the appropriate tables, figures, and flowcharts. In Section III, the research's results and discussion part have been discussed. Finally, section IV provides some guidance for the paper's future improvement.

II. Proposed System

In this section, we have briefly discussed datasets, preprocessing, and deep learning models.

A. Dataset

The HAM10000 melanoma detection dataset has been used in this work and is collected from the Harvard Dataverse repository [12]. It contains 100,15 dermoscopic images. In this dataset, there are seven types of lesions Vascular_lesions, Dermatofibroma, Melanocytic_nevi, Benign_keratosis-like_lesions, Actinic_keratoses, melanoma, and Basal_cell_carcinoma. Those are types of melanoma and benign. In Fig. 1, all kinds of lesions are shown with numbers. In Fig. 2, a sample dataset is shown.

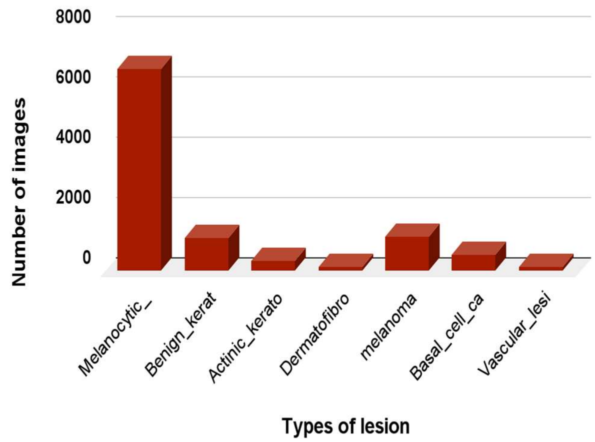

Fig. 1. Number of images in each type of lesion in the used dataset.

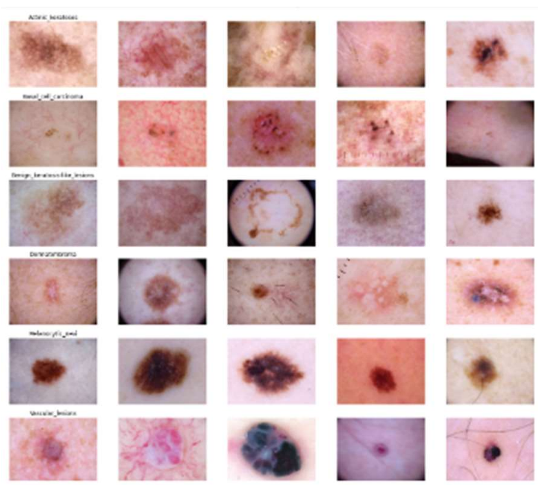

Fig. 2. Sample images of the employed dataset.

B. Dataset Preprocessing

- **Binary classification:** In the dataset, there are seven types of lesions. Each type of lesion is initialized by 0 or 1. 0 is for benign, and one is for malignant, as demonstrated in Fig. 3. We used this classification in the first phase. Now, we have used seven classes for variety.

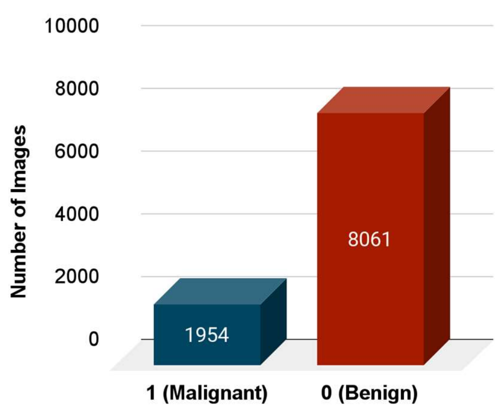

Fig. 3. Sample images of the employed dataset.

- **Normalization:** (450,600,3) is the shape of an image array. Here, 3 means three channels which are red, green, and blue. (0,1) is taken for the mean across the axis.
- **Resize images:** All images are reshaped into (64,64) for Convolutional Neural Network. (256, 192) has been used for Visual Geometry Group 16 and Dense Convolutional Network 201.
- **Data augmentation:** For data augmentation, we have used the following features.

TABLE I. DATA AUGMENTATION

Types	Value
Rotation	40
Width-shift	0.2
Height-shift	0.2
Shear	0.2
Zoom	0.2
Horizontal-flip	True
Fill	nearest

In Table I, all types of data augmentation parameters and their corresponding parameters are shown.

C. Deep Learning Models

Three models have been used in this project. All models have been described below.

Convolutional Neural Network: In this project, a Convolutional Neural Network has been used to build the model. After preprocessing, we make a Convolutional Neural Network with (64,64) reshaped images.

64,64,3 is the input feature map. 3 means three colors: red, green, and blue. In the first layer, there are 16 filters with 3×3 dimensions. In the next layer, max pooling has been added with a 2×2 dimension. That's the whole part of the first convolution.

In the second convolution, there are 32 filters with 3×3 dimensions, and a 3×3 max pooling layer has been added in the next layer.

In the third convolution, there are 64 filters with 3×3 dimensions, and a 3×3 max pooling layer has been added in the next layer.

Then, 512 hidden units and ReLU activation were used to construct a layer that was fully connected. A dropout rate of 0.5 has been added to the layer. Adam optimizer has been used with a 0.001 learning rate. 30 epochs have been used to train the model. In Fig. 4, a sample architecture shows this model. In this model, the number of total parameters is 2,124,839.

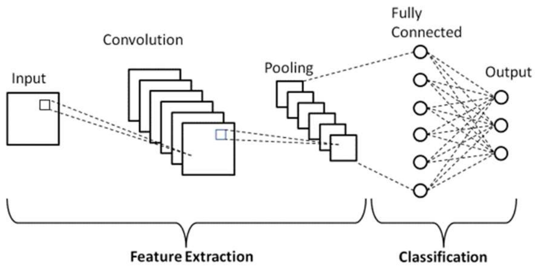

Fig. 4. CNN Architecture.

Adam Optimizer:

The following equations are called the Adam optimizer.

$$m_t = \beta_1 \times m_{t-1} + \{(1-\beta_1) \times g_t\} \quad (1)$$

$$v_t = \beta_1 \times v_{t-1} + \{(1-\beta_2) \times g_t^2\} \quad (2)$$

$$\widehat{m_t} = m_t \div (1-\beta_1) \quad (3)$$

$$\widehat{v_t} = v_t \div (1-\beta_2) \quad (4)$$

$$\Theta_{t+1} = \Theta_t - \{(\eta \times \widehat{m_t})/(\sqrt{\widehat{v_t}} + \epsilon)\} \quad (5)$$

where, 1 and 2 are respectively 0.9 and 0.999 default decay rates in Keras. And m_t and v_t are, respectively, a previous gradient average with exponential decay and a previously squared gradient average with exponential decay.

Visual Geometry Group 16: In this project, Visual Geometry Group 16 has been used to build another model. After preprocessing, we built a Visual Geometry Group 16 with (256,192) reshaped images.

We have been loaded with a pre-trained model with (256,192,3) input shape. In that model, there are 19 layers. At first, the top fully connected layers were erased, and new fully connected layers were added with 512 hidden units, ReLU activation, and a 0.5 rate of dropout. All layers were frozen when fully connected layers were added. All layers were unfrozen after feature extension with 3 epochs. 30 epochs were used to build this model. Adam optimizer has been used with a 0.001 learning rate. In Fig. 5, a sample architecture shows this model. In this model, the number of total parameters is 14,980,935.

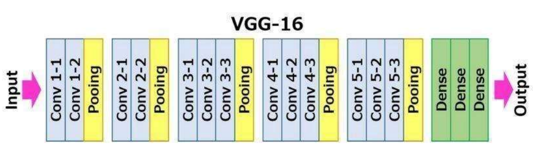

Fig. 5. VGG16 Architecture.

Dense Convolutional Network 201: In this project, a Dense Convolutional Network 201 has been used to build another model. After preprocessing, we built a Dense Convolutional Network 201 with (256,192) reshaped images.

We have been loaded with a pre-trained model with (256,192,3) input shape. In that model, there are 201 depth layers. At first, the top fully connected layers were erased, and new fully connected layers were added with 512 hidden units, ReLU activation, and a 0.5 rate of dropout. All layers were frozen when fully connected layers were added. All layers were frozen after feature extension with 3 epochs. 30 epochs were used to build this model. Adam optimizer has been used with a 0.001 learning rate. In Fig. 6, a sample architecture shows this model. In this model, the number of total parameters is 19,309,127.

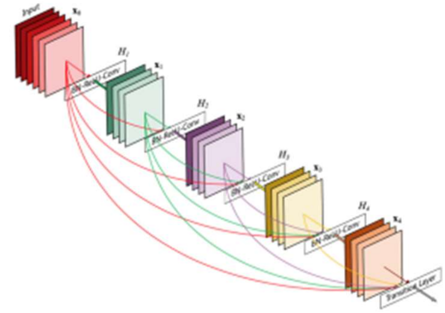

Fig. 6. DenseNet201 Architecture

MobileNetV2: In this work, a MobileNet has been used to build another model. After preprocessing, we made a Dense Convolutional Network 201 with (224,224) reshaped images.

We have been loaded with a pre-trained model with (224,224,3) input shape. In that model, there are 88 depth layers. The structure of MobileNetV2 is a residual structure that is inverted. First, there is the initial layer which has 32 filters. Then, new fully connected layers were added with 512 hidden units, ReLU activation, and a 0.5 rate of dropout. The connections of residuals are between the layers of the bottleneck. Thirty epochs were used to build this model. Adam optimizer has been used with a 0.001 learning rate. In Fig. 7, a sample architecture shows this model. In this model, the number of total parameters is 22,261,827.

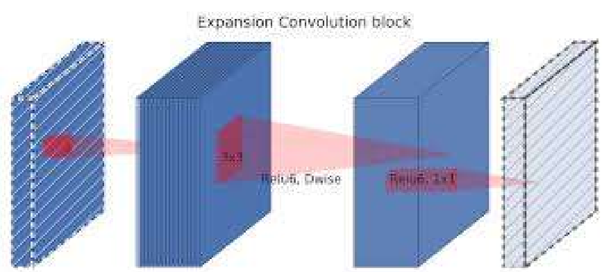

Fig. 7. MobileNetV2 Architecture

Roboflow: Another model in this project was created using Roboflow, an open-source framework. Models come in a variety of forms. These models are simple to utilize through this website. The website used ImageNet to train our dataset.

We initially spilled our entire dataset into train, test, and test after uploading it. Next, we resized all of the dataset's photos to 224×224. Grayscale and auto-orient were used throughout the preprocessing stage. We used to crop and rotate throughout the augmentation phase. Finally, we selected the train button. The website displayed the model's validation accuracy after a few hours.

Vision Transformer: Another model has been constructed in this project using a Vision Transformer. We created a Vision Transformer using (224,224) reshaped images following preprocessing.

Images in this design are scattered throughout a predetermined number of patches that are linearly embedded with a position. Thirty epochs were used to build this model.

Vectors linearly transmit these to a transformer encoder. We have been loaded with a pre-trained model with (224,224,3) input shape. In Fig. 8, a sample architecture shows this model.

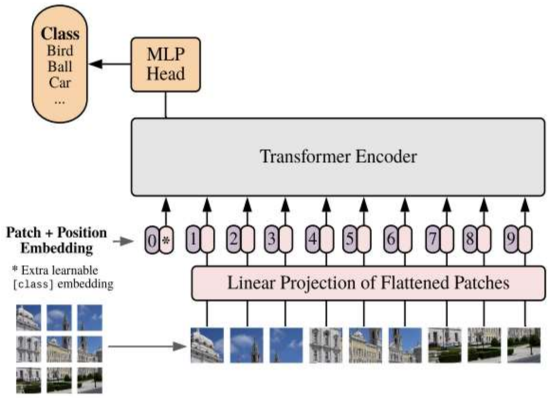

Fig. 8. Vision Transformer Architecture

Knowledge Distillation: In this project, a knowledge distillation has been used to build another model. After preprocessing, we made a knowledge distillation with (224,224) reshaped images.

This paper used ResNet50 as a teacher model and mobile net as a student model. We have been loaded with a pre-trained model of ResNet50 with (224,224,3) input shape. In that model, there are 48 depth layers. With 512 hidden units, ReLU activation, and a 0.5 rate of dropout, new completely connected layers have been created. Between the levels of the bottleneck are the connections of residuals. The total parameters are 23,542,786. This model was created using 30 epochs. Then a small network was built as a student model, a mobile network. It is also a model. A 0.001 learning rate was utilized using the Adam optimizer. In this model, the total parameters are 3.2 million. It is smaller than the teacher model. There is a value named temperature, which is a very important part of this model. Learning from the teacher model is beneficial for the student model. So, we used hyper-tuning on it to get better accuracy. The following equations are used in the KD model.

$$softmax(a_i) = exp\left(\frac{a_i}{T}\right) \div \sum_{j=1}^{k} exp\left(\frac{a_i}{T}\right) \quad (6)$$

$$p(teacher) = softmax(a_t \div T) \quad (7)$$

$$p(student) = softmax(a_s \div T) \quad (8)$$

In Fig. 9, a sample architecture shows this model.

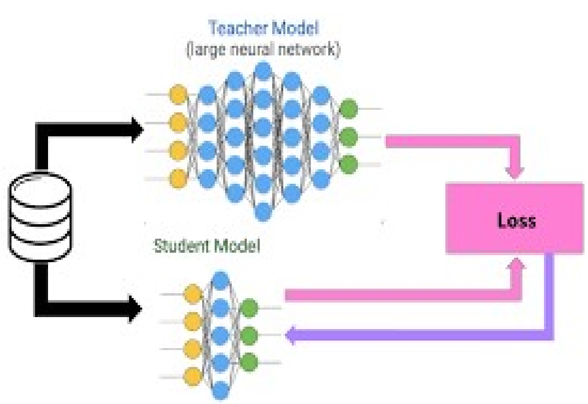

Fig. 9. Knowledge Distillation Architecture.

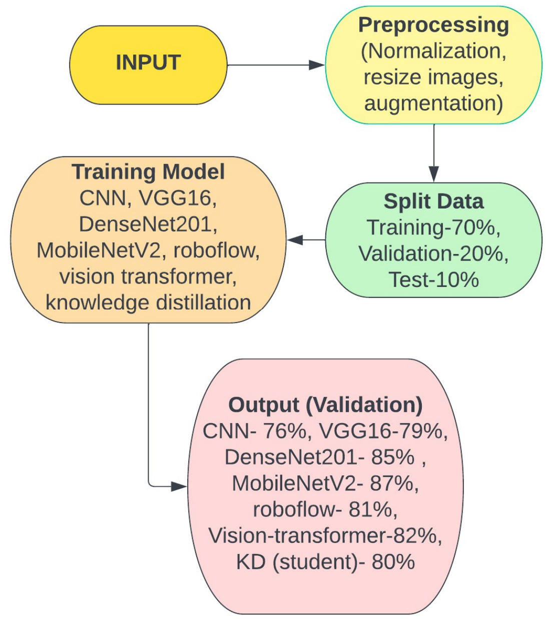

Fig. 10. Working sequences of the proposed automatic melanoma detection project.

In Fig. 10, the flowchart of the entire automatic melanoma skin cancer detection system is shown.

Three models have been used in this project. All models have been described below.

III. Results and Discussion

In Table II, the training accuracy, validation accuracy, and test accuracy of all models have shown.

TABLE II. ACCURACY OF ALL MODELS

Model Name	Training Accuracy	Validation Accuracy	Test Accuracy
CNN	0.7855	0.7637	0.7695
VGG16	0.8988	0.7971	0.7954
DenseNet201	**0.9870**	**0.8659**	**0.8433**
MobileNetV2	0.9121	0.8752	0.8399
Roboflow (ImageNet)		0.81	
Vision Transformer	0.8548	0.8269	0.8126
Knowledge Distillation (Student)	0.8961	0.8018	0.7761

In Table II, DenseNet201 has obtained better accuracy than other models.

TABLE III. COMPARISON OF TRAINING TIME

Dataset	GPU	Model	Overall time elapsed (Epoch 30)
HAM10000	Tesla T4 or P100	CNN	1293 seconds
		VGG16	676 seconds
		DenseNet201	847 seconds
		MobileNetV2	660 seconds
		Vision Transformer	972 seconds
		Knowledge Distillation (Student)	682 seconds

Fig. 11. CNN model’s accuracy and loss graph.

In Fig. 11, CNN models' training and validation accuracy increase with the number of epochs. On the other hand, the training and validation losses decrease with the number of epochs.

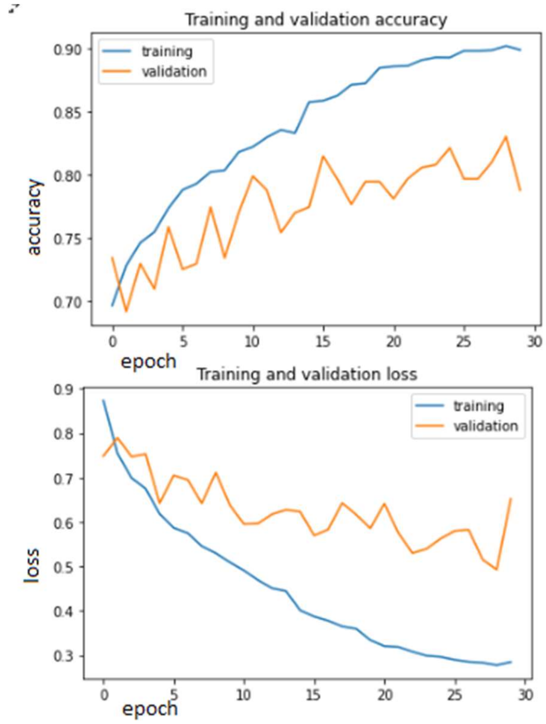

Fig. 12. VGG16’s accuracy and loss graph

In Fig. 12, the VGG16 models' training and validation accuracy rise as the number of epochs increases, but training and validation accuracy decline with the number of epochs in the loss graph.

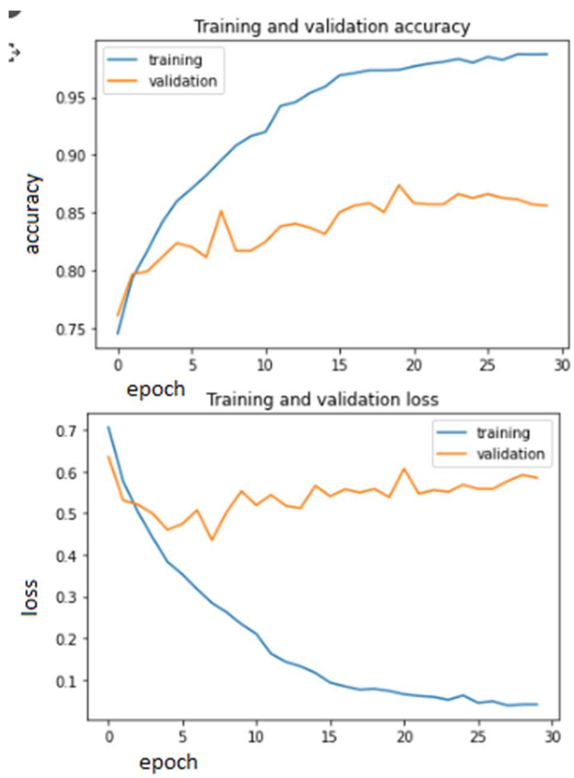

Fig. 13. DenseNet201’s accuracy and loss graph

In Fig. 13, with more epochs, Densnet 201 models are more accurate during training and validation. Additionally, as epochs are added, activity and validation losses get smaller.

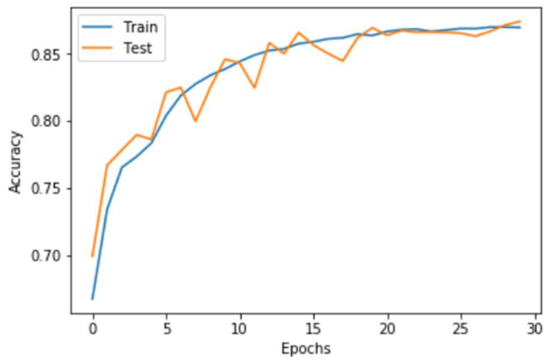

Fig. 14. MobileNetV2's accuracy and loss graph

In Fig. 14, the training and validation accuracy of MobileMetV2 models are increasing with the number of epochs.

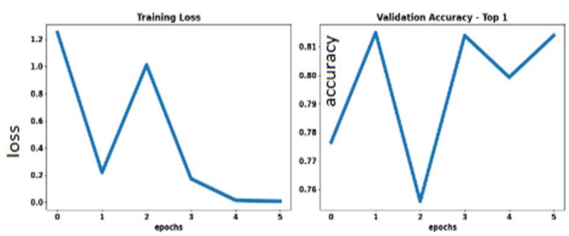

Fig. 15. Roboflow's accuracy and loss graph

In Fig. 15, the training and validation accuracy of Roboflow models are increasing with the number of epochs.

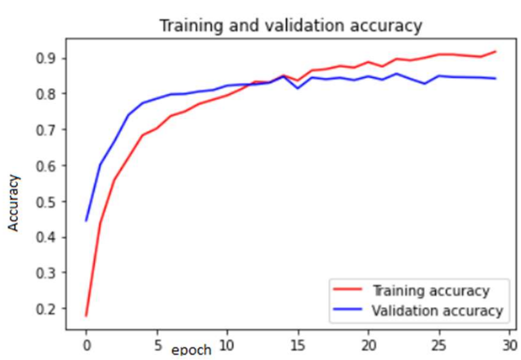

Fig. 16. KD's accuracy and loss graph

In Fig. 16, the training and validation accuracy of knowledge distillation student models are increasing with the number of epochs.

DenseNet201 has a better test, validation, and training accuracy in the result part.

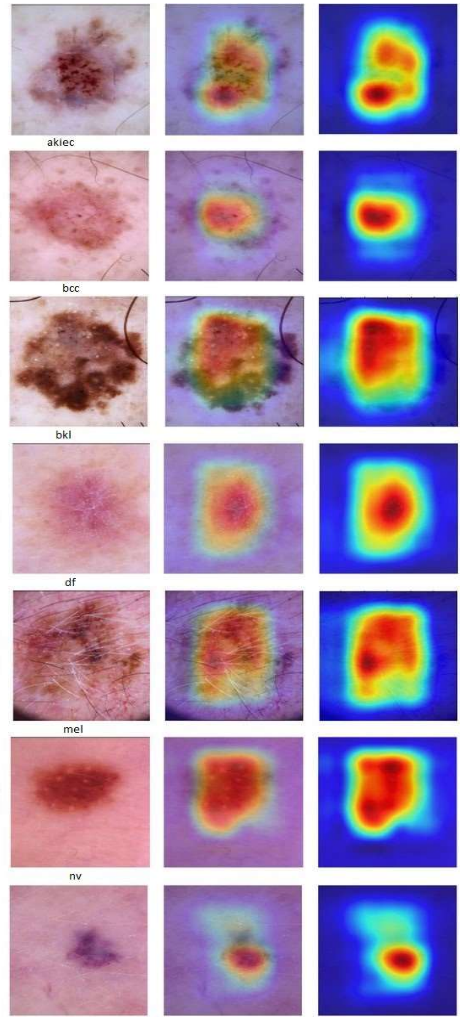

Fig. 17. Grad cam of Densenet201

Grad-CAM (Gradient-weighted Class Activation Mapping) is a technique used to visualize the regions in an image that contribute most to the prediction of a given deep learning technique. It provides a heatmap highlighting the areas in an image that the network considers most important for its prediction. Grad-CAM is implemented by computing the gradients of the target class concerning the feature maps of the last convolutional layer and then using these gradients to weigh the feature maps and generate the heatmap. This technique helps in understanding what the network has learned and can be used for debugging, analysis, and explainability of deep learning models. In Fig. 17, we have obtained best accuracy in Densenet201. We have applied GradCam on this model. It provides a graphic representation of the prediction outcome. It provides some concise illustrations of how and why it makes its prediction.

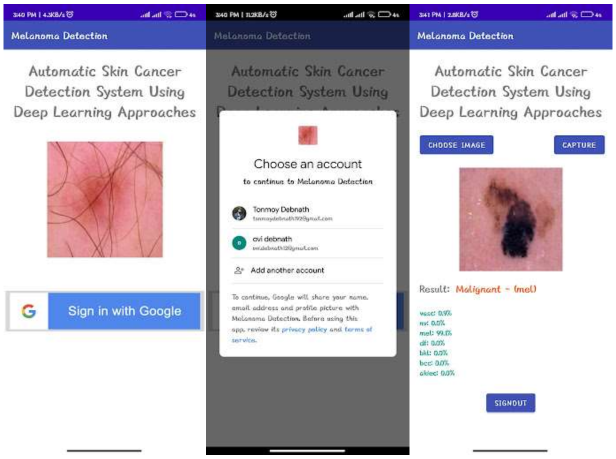

Fig. 18. Screenshots of mobile application

Screenshots of our mobile application are shown in Fig. 18. Three screenshots of our application have been included. We logged in using the Google Authentication System. The options are upload image and capture image. We can select an image by comparing two possibilities. The application will display the selected image's outcome when the user chooses it. Our results now include the confidence scores for every class.

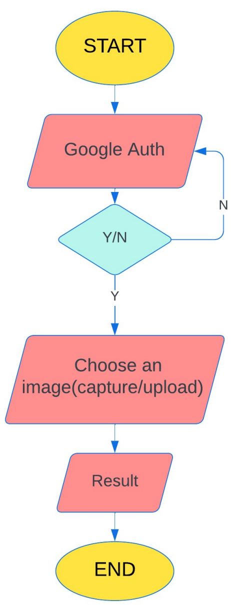

Fig. 19. Flowchart of mobile application

Fig 19 shows the flowchart of our mobile application.

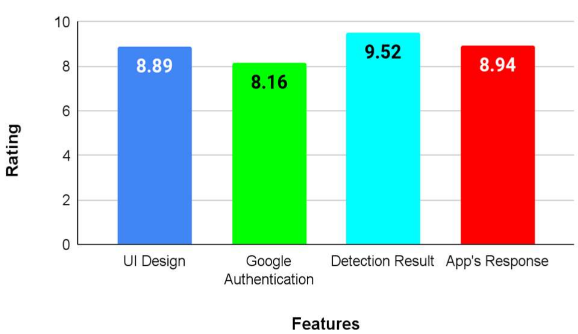

Fig. 20. App reviews rating.

Fig. 20 shows the review rating of our android application. Again, we have used MobleNetV2 to build this android application. Finally, we have organized a survey where users reviewed our android application's features. We have collected from a total of 19 users. Selected features are UI design, google authentication, detection result, and application response. Each rating is averaged and represented by a chart. For example, the highest average rating is 9.52 out of 10 in the detection result part.

TABLE IV. COMPARISON OF THIS WORK WITH EXISTING SYSTEMS

Author	Dataset	Network	Training Accuracy	Validation Accuracy	Test Accuracy
[4]	ISIC	CNN	0.98	0.89	0.93
[5]	Canadian hospital (open source)	Their Own Model	0.8389	0.8299	0.8295
[6]	HAM10000	ResNet50	0.9	0.89	0.87
[7]	ISIC-ISBI	Random Forest			0.939
[8]	ISIC	CNN	0.884		0.858
[9]	ISIC	Proposed ensemble	0.935	0.87	0.84
[10]	ISIC-Archive	Inception-V3	0.99	0.90	0.8742
[11]	ISIC	Their model	0.95	0.97	0.96
	PH2	Their model	0.95	0.95	0.93
This work	HAM10000	**DenseNet201**	**0.9870**	**0.8659**	**0.8433**

IV. CONCLUSIONS

Finally, we built three models with better accuracy using deep learning, which can easily detect melanoma skin cancer with the help of dermoscopic pictures. We collected our Dataset from Harvard, which contains 10015 dermoscopic images. The convolutional Neural Network model, Visual Geometry Group 16 model, Dense Convolutional Network 201, MobileNet, Roboflow (ImageNet), Vision Transformer, and Knowledge Distillation were used to implement our model. In our work, we implemented data preprocessing, data augmentation, and classification for training our model. As a result, dense Convolutional Network 201 achieved better accuracy than the other models, which were 98% on training, 85% on validation, and 84% on the test part. Our models can detect melanoma skin cancer quickly within a very low-cost range, revolutionizing medical science.

In the future, we will increase the accuracy of the models used in this work. We will also include or train more models to build a project with better accuracy.

REFERENCES

[1] R.-M. Szeimies, J. T. Lear, and V. Madan, “Non-melanoma skin cancer,” Lancet, vol. 375, pp. 673-685, 2010

[2] A. Waldmann et al., “Does skin cancer screening save lives?” Cancer, vol. 118, pp. 5395-5402, 2012

[3] S. J. Kempton, V. K. Rao, and J. T. Chen, “The economics of skin cancer,” Plastic and Reconstructive Surgery Global Open, vol. 4, 2016

[4] M. A. Hossin *et al*., “Melanoma Skin Cancer Detection Using Deep Learning and Advanced Regularizer,” International Conference on Advanced Computer Science and Information Systems, pp. 89-94, 2020.

[5] R. Kaur, H. GholamHosseini and R. Sinha, “Deep Convolutional Neural Network for Melanoma Detection using Dermoscopy Images,” International Conference of the IEEE Engineering in Medicine & Biology Society, pp. 1524-1527, 2020

[6] M. M. I. Rahi *et al*., “Detection of Skin Cancer Using Deep Neural Networks,” IEEE Asia-Pacific Conference on Computer Science and Data Engineering, 2019, pp. 1-7.

[7] A. Javaid, M. Sadiq and F. Akram, “Skin Cancer Classification Using Image Processing and Machine Learning,” International Bhurban Conference on Applied Sciences and Technologies, 2021, pp. 439-444

[8] J. Daghrir, L. Tlig, M. Bouchouicha and M. Sayadi, “Melanoma skin cancer detection using deep learning and classical machine learning techniques: A hybrid approach,” International Conference on Advanced Technologies for Signal and Image Processing, 2020, pp. 1-5

[9] A. Imran, A. Nasir, M. Bilal, G. Sun, A. Alzahrani and A. Almuhaimeed, "Skin Cancer Detection Using Combined Decision of Deep Learners," IEEE Access, vol. 10, pp. 118198-118212, 2022.

[10] A. Demir, F. Yilmaz and O. Kose, "Early detection of skin cancer using deep learning architectures: resnet-101 and inception-v3," Medical Technologies Congress, 2019, pp. 1-4

[11] A. A. Adegun and S. Viriri, "Deep Learning-Based System for Automatic Melanoma Detection," IEEE Access, vol. 8, pp. 7160-7172, 2020.

[12] P. Tschandl, C. Rosendahl and H. Kittler, “The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions,” Scientific Data, vol. 5, pp. 1-9, 2018.