Comparative study on different Deep Learning models for Skin Lesion Classification using transfer learning approach

Saswat Panda, Abhishek Sunil Tiwari, Manas Ranjan Prusty

School of Computer Science and Engineering
Vellore Institute of Technology, Chennai Campus, India – 600127

DOI: 10.29322/IJSRP.11.01.2021.p10923
http://dx.doi.org/10.29322/IJSRP.11.01.2021.p10923

ABSTRACT

Developing countries, specifically India, do not have sufficient hospitals and doctors to reach out to the population. Forget about skin specialists, there are still thousands of villages without even a basic hospital. But, there is one thing that reaches out to every person in this country which is the internet. This research paper focuses on training different pre-trained models on our dataset and suggests the best one for skin lesion classification. This model has been used with a mobile-compatible web app to detect skin cancer at the earliest stage possible. Hence, the objective of this research paper is to develop an intelligent system to detect skin cancer at the earliest stage possible by a skin lesion classifier. The dataset includes different categories of diseases like Actinic Keratosis, Vascular Lesions, Lentigo, Melanoma, and Dermatofibroma. The pre-trained models used in this research are the different transfer learning methods like VGG19, Xception, Densenet, Inception, MobileNet, NasNetMobile, and Resnet. Using transfer learning, Deep Neural Networks can be trained with presumably less amount of data. Also, transfer learning has been consistently proven to reduce training time as well as boost model accuracy. Hence, this paper has given the emphasis on using transfer learning models instead of building a model from scratch.

KEYWORDS: Convolutional Neural Network; Deep learning; Skin Lesion Classification; Transfer Learning.

I. INTRODUCTION

A lesion is any damage or abnormal change in the tissue of an organism, usually caused by disease or trauma. Similar abnormal changes on the surface of the skin are called skin lesions. If these skin lesions are identified and treated at a very early stage then only it is possible to control the incidence of skin cancer which is increasing by epidemic proportions. Worldwide the most common skin cancer is Basal Cell Carcinoma. But, the most number of deaths is caused by malignant melanoma. Sun exposure remains the most important risk factor for all skin neoplasms. Thus, patients should be taught basic "safe sun" measures such as sun avoidance during peak ultraviolet-B hours; proper use of sunscreen and protective clothing; and avoidance of suntanning. In addition to these, detecting cancers as early as possible is also an important aspect, which is one of the primary goals of this research paper.[1]

In India, skin cancers constitute about 1-2% of all diagnosed cancers. The most common type of cancer seen in India is the Squamous Cell Carcinoma(SCC), which is generally preceded by actinic keratosis[2]. In this study, actinic keratosis is included in the dataset as a precancerous stage to SCC. Also, various precancerous and cancerous Skin Lesions are included in the dataset used in the research.

Looking at previous works on skin lesion classification, the research by Tan et al [3] could achieve an average accuracy of 92% and 84% for benign and malignant skin lesion classification. Another research by Chaturvedi et al [4] classified the skin lesions into seven categories with a categorical accuracy of 83.15%. In that research, they used MobileNet for skin lesion classifications. In research by Jyothilakshmi et al [5], they have proposed and implemented an automated method for the detection of malignancy from various skin diseases. The technique was capable of performing the segmentation accurately for extracting the feature parameters similar work has been done [8]. A research paper by A.K.Verma et al [6] tries to predict skin disease by using UCI dataset and has an accuracy of 97%. The research by S. Kolkur [7] showed that texture is an interesting image feature that can be used for image classification. The research by Codella et al was capable of segmenting skin lesions, as well as analyzing the detected area and surrounding tissue for melanoma detection and could achieve an

This publication is licensed under Creative Commons Attribution CC BY.

accuracy of 76% [10]. In a research by Ge et al, a novel deep convolutional neural network(DCNN) architecture along with a saliency feature descriptor to capture discriminative features of the two modalities for skin lesions classification has been proposed[11], similar kind of work has been done in the paper [9]. Most of the previous research on skin lesion classification, i.e., multiclass skin lesion classification as well as benign and malignant lesion classification have used deep learning [12] [13] [14] [15] [16] [17] [18] [19] [20] [21]. This is because skin lesion classification is a classification of medical importance therefore a very high accuracy with least error is required which can be achieved by using Deep Learning. Therefore, in this research deep learning has been used. Deep Learning has been aided with the ensemble learning methods to reduce variance of error as well as generalization error. Not only that, this paper has used several transfertransfer learning aided deep learning models on the dataset of this research work to find out the best.

This research paper consists of five sections including the introduction section. Section 2 explains the various datasets and models used for the classification. Section 3 elucidates on the observation made by applying different models on the dataset. Section 4 describes the website and the results of this research by using one of the models in the website. Finally, the last section concludes the research paper by suggesting the best model for skin lesion classification and the impact of the website.

II. DATASET AND MODELS

1) About Dataset

The dataset in this paper includes six categories and they are melanoma, actinic keratosis, vascular lesion, lentigo, dermatofibroma, and normal skin. In this research, normal skin samples were collected and the rest images were taken from the ISIC dataset were taken[1]. All the collected images were then divided into a training and a validation set for individual categories depending upon the number of images in that category. The distribution of the dataset used in this paper can be seen in Table 1.

TABLE 1: Train and Validation split of Lesion Categories

Class	Train	Validation
Normal Skin	90	10
Actinic Keratosis	103	27
Vascular Lesions	117	25
Lentigo	63	8
Melanoma	202	20
Dermatofibroma	97	18

2) Dataset description

In this paper, six categories are used for classification. The description of these categories is given below:

a. **Actinic keratosis:** This is a rough, scaly patch on your skin (fig. 1) that develops from years of exposure to the sun. It's most commonly found on your face, lips, ears, back of your hands, forearms, scalp, or neck. Some squamous cell skin cancer develops from actinic keratoses. Because of this, the lesions are often called precancer. They are not life-threatening. But if they are not found and treated early, they have the chance to develop into skin cancer.

[1] https://www.isic-archive.com/#!/topWithHeader/onlyHeaderTop/gallery

This publication is licensed under Creative Commons Attribution CC BY.

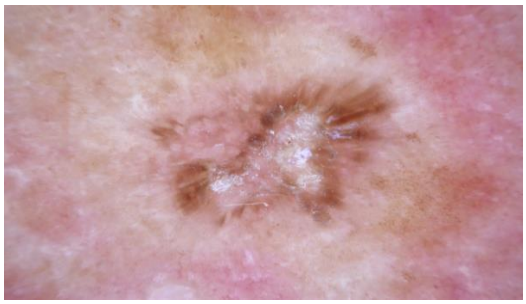

Figure 1: Actinic Keratosis sample

b. Vascular lesions: These are relatively common abnormalities of the skin and underlying tissues, more commonly known as birthmarks. Figure 2 shows a sample of Vascular lesions.

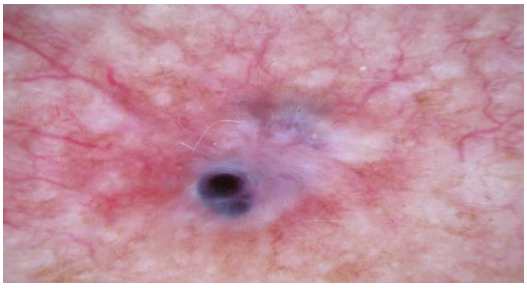

Figure 2: Vascular Lesions sample

c. Lentigo: It is a small pigmented spot on the skin (fig. 3) with a clearly defined edge, surrounded by normal-appearing skin. It is harmless hyperplasia of melanocytes. It is generally linear in its spread. Melanocytes normally reside at the cell layer directly above the basement membrane of the epidermis. The hyperplasia of melanocytes is restricted to this cell layer only. Thereby making it harmless. But, if lentigo maligna is not treated, it may become a type of invasive melanoma skin cancer called lentigo maligna melanoma. It could take ten years or more to completely become cancerous.

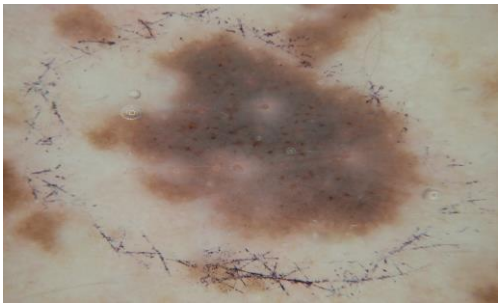

Figure 3: Lentigo sample

d. Melanoma: It is also known as malignant melanoma. It occurs when pigment-producing cells(the cells that give colour to the skin) become cancerous. It can occur anywhere in the body, especially where there is a new, unusual growth or change in an existing mole. Figure 4 shows a sample Melanoma.

This publication is licensed under Creative Commons Attribution CC BY.

Figure 4: Melanoma sample

e. Dermatofibromas: They are small, noncancerous (benign) skin growths(fig. 5) that can develop anywhere on the body but most often appear on the lower legs, upper arms, or upper back.

Figure 5: Dermatofibroma sample

2.3 Models Description

Sequential convolution network is a network in which all the layers are connected sequentially as shown in Fig. 6. Residual networks or residual blocks are those in which there are shortcut connections as in the Fig. 7 i.e the outputs of the one layer adds with the output of the other before passing it through activation function, this is done to solve the vanishing gradient problem in very deep neural networks. Dense block is the extension of the residual block in which, instead of adding/summation the output is simply concatenated. Spatially separable convolutional networks simply divide kernel to achieve speed for example a 3X3 matrix can be divided into 3X1 and 1X3, still the results will be same but now instead of 9 multiplications there will be only 6 multiplication, see Fig. 8. Depthwise separable convolutional networks are those in which a kernel/matrix/filter cannot be factored into a smaller kernel/matrix/filter as in the case of spatially separable. This network is most widely used. It works in two parts depthwise and pointwise. In depthwise, the number of channels (channel is a colorspace/depth like RGB) remains unchanged and convolution operation is applied to each channel at a time unlike standard convolution as displayed in Fig 9. In pointwise, 1X1X (no of the channel) convolution is applied to all channels and there are '*N*' such filters as shown in Fig. 10.

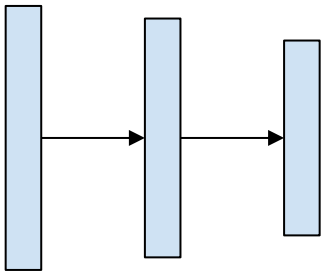

Figure 6: Sequential Convolutional Network

This publication is licensed under Creative Commons Attribution CC BY.

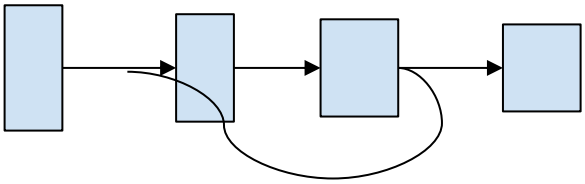

Figure 7: Shortcut connections

$$\begin{bmatrix} 3 & 6 & 9 \\ 4 & 8 & 12 \\ 5 & 10 & 15 \end{bmatrix} = \begin{bmatrix} 3 \\ 4 \\ 5 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}$$

Figure 8: A matrix factored into two matrix

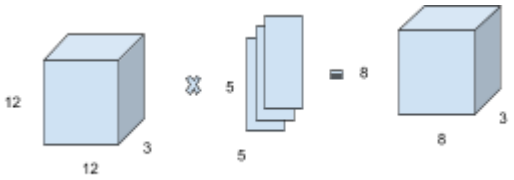

Figure 9: Depthwise Operation, convolution performed on each channel at a time

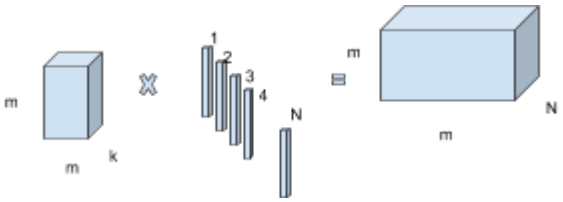

Figure 10: Pointwise Operation input multiplied by 1x1xm N such kernels to get N channels

a. Xception

Xception was proposed by François Chollet, the creator and chief maintainer of the Keras library. Xception is an extension of the Inception architecture which replaces the standard Inception modules with depth wise separable convolutions. Xception sports the smallest weight serialization at only 91MB [22].

b. DenseNet201

DenseNets consists of dense blocks and pooling operations, where each dense block is concatenated iteratively by previous feature maps. This architecture can be seen as an extension of ResNets, which performs an iterative summation of previous feature maps. However, this small modification has some interesting implications like DenseNets are more efficient in the parameter usage. All layers can easily access their preceding layers making it easy to reuse the knowledge from previously computed feature maps. DenseNets are very good fit for semantic segmentation as they naturally give rise to skip connections and multi-scale supervision. [23]

c. ResNet

Unlike traditional sequential network architectures:AlexNet, OverFeat, and VGG, ResNet is a "network in network" architecture also called micro-architecture. A collection of micro-architecture which includes standard CONV, POOL, etc. layers leads to the macro-architecture . It was introduced by He et al. 2015 [24]. Even though ResNet is far deeper than VGG16 and VGG19, the model size is smaller thanks to the usage of global average pooling instead of using fully-connected layers. The size of the model is 102MB for ResNet50.

d. InceptionV3

The "Inception" architecture was introduced by Szegedy et al. 2014 [25]. The goal of the inception module is to act as a "multi-level feature extractor" by computing 1×1, 3×3, and 5×5 convolutions within an equivalent module of the network. The output of those filters are then stacked along the channel dimension and before being fed into subsequent layers within the network. The network was first named as GoogLeNet, but subsequent manifestations have simply been called Inception vN where N refers to the version number put out by Google. The weights for Inception V3 are smaller than both VGG and ResNet, and cost only 96MB.

This publication is licensed under Creative Commons Attribution CC BY.

e. MobileNetV2

Mobilenet [26] has 30 layers with a convolutional layer with stride 2, depthwise layer, pointwise layer that doubles the number of channels, depthwise layer with stride 2.

f. VGG19

The VGG network architecture was introduced in 2014 [27]. This network is characterized by its simplicity, using only 3×3 convolutional layers stacked on top of every other in increasing depth. Reducing volume size is handled by max pooling. Followed by two fully-connected layers, each with 4,096 nodes are then followed by a softmax classifier. It is being found that training VGG takes a lot of time. So, in order to make training easier, the researchers first trained smaller versions of VGG with fewer weight layers first. The smaller networks converged and then used as initializations for the larger, deeper networks. This process is called pre-training.

g. InceptionResNetV2

Inception-RestNet-V2 is a convolutional neural network that trains over one million images from the ImageNet database. The network is deep in 164 layers and can classify 1000 object categories as keyboards, mouse, pencils and many animals. As a result, the network has learned great feature representations for a wide range of images. Input size of this network is 299 x 299. [24][25]

h. NasNetMobile

NasNetMobile is a pre-trained model. It is trained in a subset of the ImageNet database. Loading pre-trained loads on ImageNet is optional in this model. It is one of the lightest models, about 23 megabytes in size.[28]

2.4 Performance Measures

A confusion matrix is an important measure to check the performance of a classifier network learned using any machine learning algorithm. Table 2 shows the representation of a confusion matrix for a binary classification problem. True Negatives (TN) are the number of negative examples which are correctly classified as negative. False Positives (FP) are the number of negative examples which are incorrectly classified as positive. True Positives (TP) are the number of positive examples which are correctly classified as positive. False Negatives (FN) are the number of positive examples which are incorrectly classified as negative.

TABLE 2: Confusion Matrix

	Predicted Negative	**Predicted Positive**
Target Negative	TN	FP
Target Positive	FN	TP

The most commonly used performance measures derived from the confusion matrix are accuracy and error rate. The former is the ratio of the number of all the correctly classified samples to the total number of test samples whereas the later is the ratio of all the incorrectly classified samples to the total number of test samples as shown in Eq. 1 and 2 respectively. It may be highlighted that accuracy and error rate sum up to 1.

$$Accuracy = (TN + TP) / (TN + FP + FN + TP) \quad (1)$$

$$Error\ rate = (FP + FN) / (TN + FP + FN + TP) \quad (2)$$

Accuracy and error rate measures are very deceptive as these are data dependent. In case of imbalanced dataset where the number of majority samples is too large compared to the minority samples, the classifier gets biased towards the majority samples [29]. In such cases, the accuracy metric results in a very high value and the error rate being very low. These results project as if the classifier is an ideal one which actually is not the real scenario. In such cases, the minority class does not get properly classified yet these two metrics suggest the classifier to be an efficient one. In order to overcome such imbalanced dataset scenarios, there are some other evaluation metrics which state the actual performance of the classifier. These metrics are precision, recall, specificity, fall-out, F-measure and G-mean. These measures are defined as follows.

$$Precision = TP / (FP + TP) \quad (3)$$

This publication is licensed under Creative Commons Attribution CC BY.

$$Recall\ or\ Sensitivity\ or\ True\ Positive\ Rate = TP\ /\ (FN + TP) \quad (4)$$

$$Specificity\ or\ True\ Negative\ Rate\ =\ TN\ /\ (TN + FP) \quad (5)$$

$$Fall\text{-}out\ or\ False\ Positive\ Rate\ or\ (1 - Specificity)\ =\ FP\ /\ (TN + FP) \quad (6)$$

$$F - Measure\ =\ \frac{(1+\beta)^2 \times Recall \times Precision}{(\beta^2 \times Recall) + Precision} \quad (7)$$

$$G - mean\ =\ \sqrt{Precision \times Recall} \quad (8)$$

In Eq. 7, β is the co-efficient to adjust the relative importance of precision and recall. Usually precision and recall have equal importance and hence β = 1. F-measure is the harmonic mean and G-mean is the geometric mean of precision and recall. So, Eq. 7 can be simplified as follows.

$$F - Measure\ =\ \frac{Recall \times Precision}{Recall + Precision} \quad (9)$$

Precision is the measure of the exactness of the samples which were correctly classified positive out of the samples which are classified positive as shown in Eq. 3. Recall is the measure of the samples which were correctly classified positive out of the samples which were actually positive as shown in Eq. 4. Recall is also known as sensitivity or true positive rate. The importance and information of both precision and recall can be combined together as a measure known as F-measure. F-measure is the harmonic mean whereas G-mean is the geometric mean of precision and recall as shown in Eq. 7 and 8 respectively. Specificity is the measure of the samples which were correctly classified negative out of the samples which were actually negative as shown in Eq. 5. This is similar to sensitivity. It is also called the true negative rate. Fall-out is the measure of the samples which were incorrectly classified positive out of all the samples which were actually negative as shown in Eq. 6. Fall-out is also called false positive rate or 1-specificity.

III. OBSERVATION

As of initial performance metrics, training accuracies of different models have been noted along with their validation accuracy. While training, the gradient and backpropagate to update the weights were calculated whereas, in the validation process, backpropagation was not performed.

1) Model Wise observation:

In all the models, a total of 30 epochs have been used. This was the first-time models were allowed to learn from the dataset. Every time models were trained in this research, a training batch size of 16 and validation batch size of 10 was used.
Overfit: It occurs when the model is able to do well on the training set but performs very poorly on the test set/validation set. If training and validation curves are very far from each other then it means that the model is overfitting. Also, when the training loss decreases and the test loss does not change significantly then also it is said that the model is overfitting.
Underfit: It happens when the model is not even able to perform well on the training set. It was satisfactory to find that none of our models were underfitting.

a. Xception

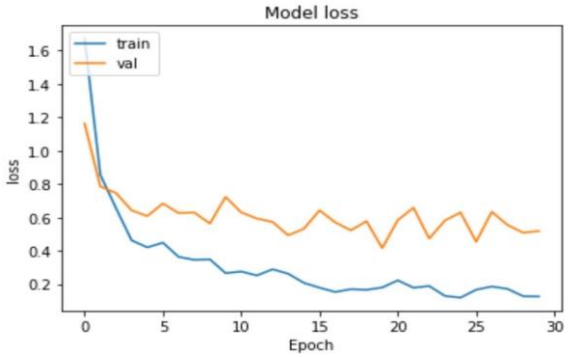

Figure 11: Graph for loss v/s epoch of Xception

From Fig. 11, we can observe that the model does not overfit since the difference in the train and test set loss curve is not much significant. The maximum accuracy obtained by this model on the training set is 94.8 %.

This publication is licensed under Creative Commons Attribution CC BY.

b. DenseNet201

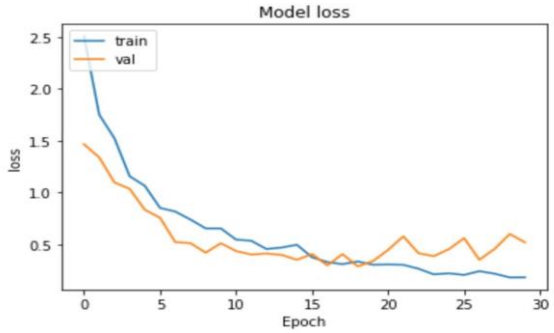

Figure 12: Graph for loss v/s epoch of DenseNet201

From Fig. 12 it can be noted that the model is not an overfit since there is a very minute difference in the train and test set loss, both the curves almost superimpose which is a sign of a good model. The maximum accuracy obtained by this model on the training set is 93.9 %.

c. InceptionV3

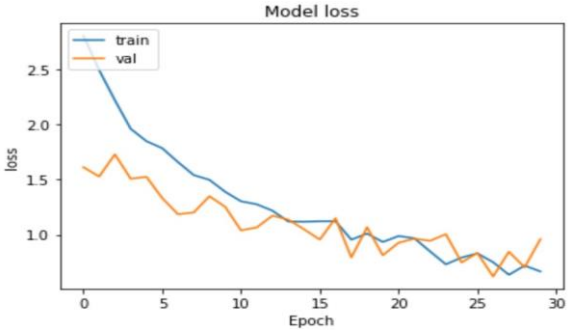

Figure 13: Graph for loss v/s epoch of InceptionV3

It can be clearly seen from Fig.13, that the model is not an overfit since there is not much difference in the train and test set loss. The maximum accuracy obtained by this model on the training set is 77.68 %.

d. MobileNetV2

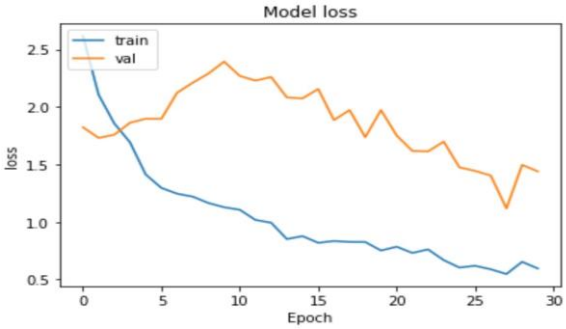

Figure 14: Graph for loss v/s epoch of MobileNetV2

The curve of the validation set is not smooth because of the small validation dataset. But, overall it is observed that with the number of epochs both validation and train loss curve gradually decreases and there is no significant difference between the train and validation curve. The graph has been provided in Fig. 14. The maximum accuracy obtained by this model is 79.17 %.

This publication is licensed under Creative Commons Attribution CC BY.

e. VGG19

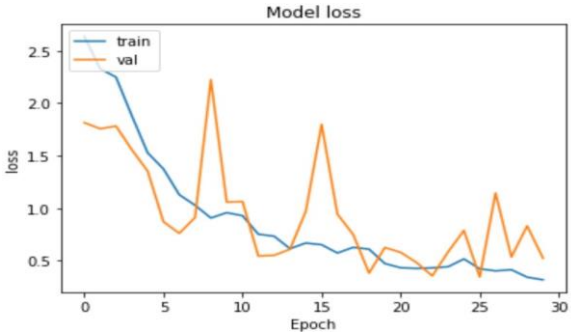

Figure 15: Graph for loss v/s epoch of VGG19

Figure 15 shows fluctuations in the validation loss curve which are due to the small validation dataset. But, overall the model is not an overfit and the maximum accuracy obtained by this model is 90.33%.

f. InceptionResNetV2

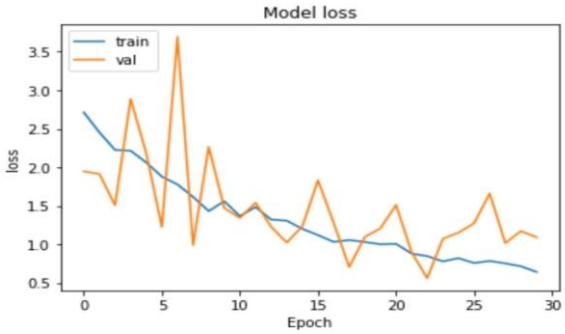

Figure 16: Graph for loss v/s epoch of InceptionResNetV2

The validation curve in Fig. 16 shows much fluctuation due to the small validation dataset. But, overall the model is not an overfit and the maximum accuracy achieved by the model is 79.46 %.

g. NasNetMobile

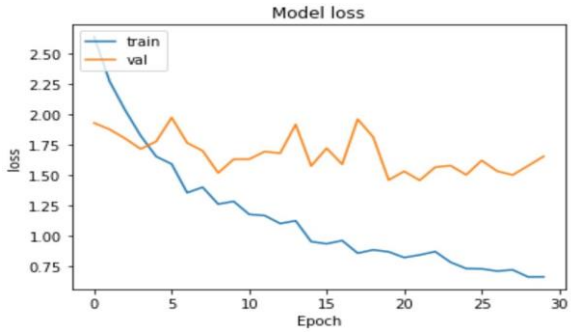

Figure 17: Graph for loss v/s epoch of NasNetMobile

From Fig. 17, it can be noted that the model is an overfit since there is much difference in the train and test set loss. Also, it can be noted that although the training loss decreases gradually, the validation loss either increases or remains constant after epoch number 7. The maximum accuracy obtained by this model on the training set is 79.32 %.

h. MobileNet

This publication is licensed under Creative Commons Attribution CC BY.

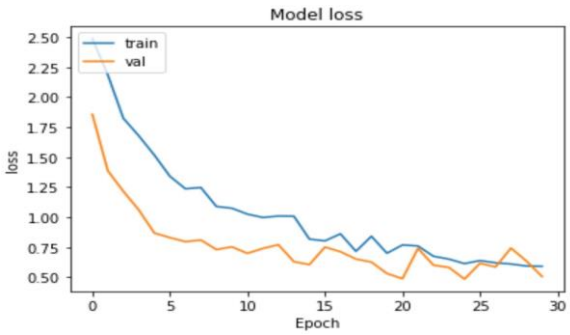

Figure 18: Graph for loss v/s epoch of MobileNet

From Fig. 18, it is observed that the model is not an overfit since there is not much difference in the train and test set loss. The maximum accuracy obtained by this model on the training set is 79.32 %.

Finally, based on accuracy, Xception model scored best training accuracy of (94.2%), followed by Densenet (93.9%), followed by vgg19 (90.33%), followed by InceptionResNetV2 (79.46%), followed by Mobilenet (79.3%), followed by mobilenetv2 (79.17%), followed by inceptionv3 (77.68%).

2) Overall Comparison:

Many times it has been seen that there is a tradeoff between recall and precision. It is uncertain and doubtful using them as metrics. So, to overcome this, F1 score is monitored which gives us a combined effect. As discussed in section 2.4 of this paper, F1 score is the harmonic mean of recall and precision. F1 score is being chosen as another metric to measure the performance of the model as shown in Table 3.

TABLE 3: Performance Metrics of different models for the training and Validation data

Model Name	Training Metrics			Validation Metrics		
	Precision	Recall	F1	Precision	Recall	F1
DenseNet201	96.9	95.5	96.0	90.1	86.4	87.8
MobileNetV2	97.0	95.2	95.9	83.2	77.1	79.6
InceptionResNetV2	95.9	94.9	95.3	82.9	78.0	79.6
MobileNet	96.1	93.8	94.7	78.5	75.4	76.0
Xception	96.2	94.0	94.0	74.0	71.2	72.0

This publication is licensed under Creative Commons Attribution CC BY.

NASNetMobile	92.3	90.2	91.0	79.0	72.9	75.0
InceptionV3	91.7	89.6	90.2	79.0	77.1	77.05
VGG19	77.4	68.2	67.0	69.2	65.3	61.5

From Table 3, it is observed that even though Xception performed the best when accuracy was the performance measuring parameter, DenseNet201 performed best when F1-score was considered. It is also observed that DenseNet201 was the second-best classifier when accuracy was the performance measuring parameter. Hence, from the research conducted using a variety of deep learning models using transfer learning approach, it is observed that DenseNet201 performed best for skin lesion classification considering both accuracy and F1-scare as the performance metrics.

IV. RESULT

The model DenseNet201 is deployed on a website which is open sourced at https://github.com/Abhishek-st/Skin-Lesion-Analysis. This repository can be used by developers for skin-lesion classification. When hosted the website can be directly used by anyone with the internet due to its mobile compatibility. Also, the tflite file of the model is provided in the github link which can be used by developers for building a mobile application. Here are some of the screenshots of the website classifying Skin-Lesions using our model:

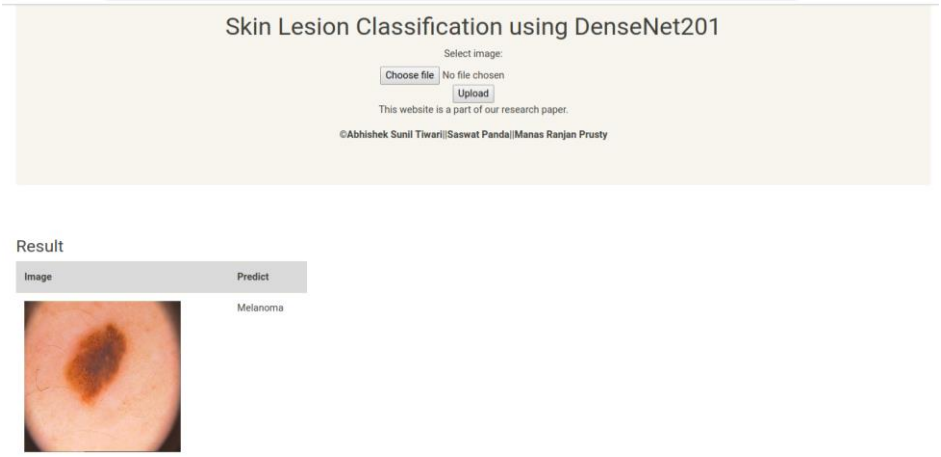

Figure 19: Correct classification of melanoma

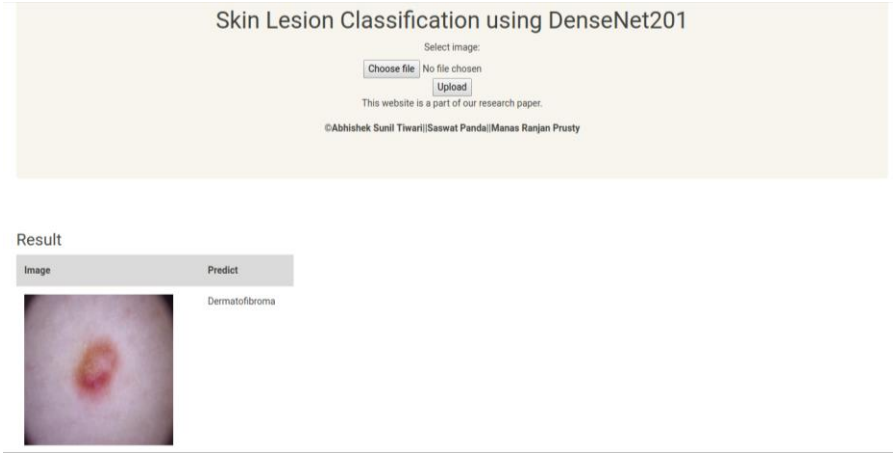

Figure 20: Correct classification of Dermatofibroma

This publication is licensed under Creative Commons Attribution CC BY.

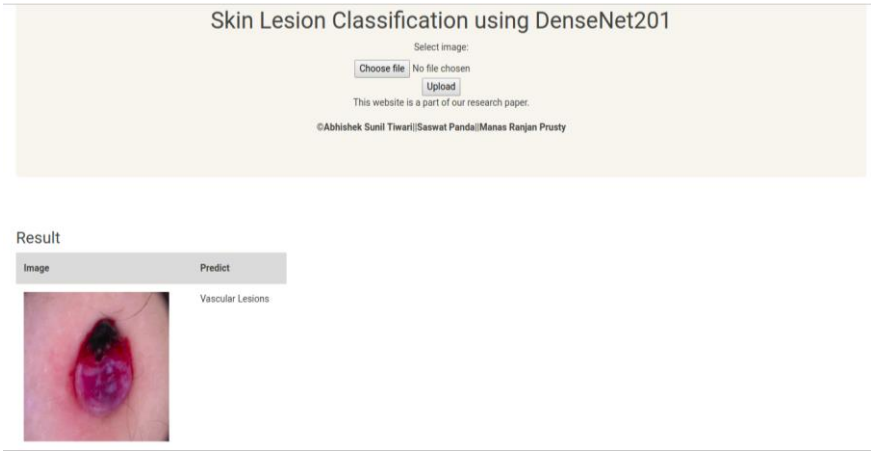

Figure 21: Correct classification of Vascular Lesion

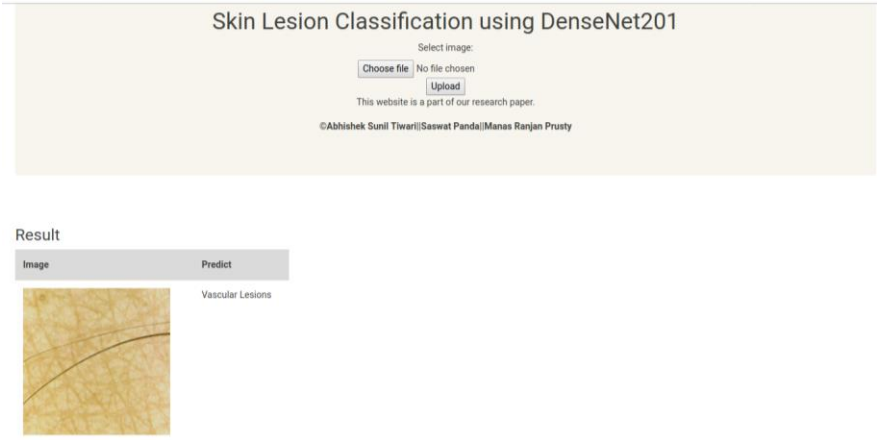

Figure 22: Normal skin wrongly classified as Vascular Lesion

IV. CONCLUSION

Thus, the objective of the research on finding an efficient model for skin lesion classification was fulfilled. In the observation, it is clear that DenseNet201 stands out from the rest. The model DenseNet201 had the second-best accuracy and the best F1-score. Hence, the best model among all the models used for training our dataset is DenseNet201. This model has been deployed with the help of a website that has been open sourced and can be used by people to detect the Skin Lesions at the earliest stage possible, especially in those countries where there is a shortage of doctors.

REFERENCES

[1] A. F. Jerant, J. T. Johnson, C. D. Sheridan, and T. J. Caffrey, "Early detection and treatment of skin cancer," *Am. Fam. Physician*, vol. 62, no. 2, pp. 357–368, 2000

[2] S. Deo, S. Hazarika, N. Shukla, S. Kumar, M. Kar, and A. Samaiya, "Surgical management of skin cancers: Experience from a regional cancer centre in North India," *Indian J. Cancer*, vol. 42, no. 3, p. 145, 2005.

[3] T. Y. Tan, L. Zhang, and M. Jiang, "An intelligent decision support system for skin cancer detection from dermoscopic images," *12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, pp. 2194–2199, 2016.

[4] S. Chaturvedi, K. Gupta and P. Prasad, "Skin Lesion Analyser: An Efficient Seven-Way Multi-Class Skin Cancer Classification Using MobileNet", *Advances in Intelligent Systems and Computing book series*, 1141 , pp 165-176, 2020

This publication is licensed under Creative Commons Attribution CC BY.

[5] K. K. Jyothi Lakshmi and J. B. Jeeva, "Detection of malignant skin diseases based on the lesion segmentation," *2014 International Conference on Communication and Signal Processing, Melmaruvathur,* pp. 382-386, 2014

[6] A. K. Verma, S. Pal, and S. Kumar, "Classification of skin disease using ensemble data mining techniques," *Asian Pac. J. Cancer Prev.,* vol. 20, no. 6, pp. 1887–1894, 2019.

[7] S. Kolkur and D. R. Kalbande, "Survey of texture based feature extraction for skin disease detection," *in 2016 International Conference on ICT in Business Industry & Government (ICTBIG),* pp. 1–6, 2016.

[8] M. S. Manerkar, S. Harsh, J. Saxena, S. P. Sarma, U. Snekhalatha, and M. Anburajan, "Classification Of Skin Disease Using Multi Svm Classifier," *in 3rd International Conference on Electrical, Electronics, Engineering Trends, Communication, Optimization and Sciences,* pp. 362–368, 2016.

[9] J. Rathod, V. Wazhmode, A. Sodha, and P. Bhavathankar, "Diagnosis of skin diseases using Convolutional Neural Networks," *in 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA),* pp. 1048–1051, 2018.

[10] N. C. Codella et al., "Deep learning ensembles for melanoma recognition in dermoscopy images," *IBM J. Res. Dev.,* vol. 61, no. 4/5, pp. 1–5, 2017.

[11] Z. Ge, S. Demyanov, R. Chakravorty, A. Bowling, and R. Garnavi, "Skin disease recognition using deep saliency features and multimodal learning of dermoscopy and clinical images," *in International Conference on Medical Image Computing and Computer-Assisted Intervention,* pp. 250–258, 2017.

[12] S. S. Han, M. S. Kim, W. Lim, G. H. Park, I. Park, and S. E. Chang, "Classification of the clinical images for benign and malignant cutaneous tumors using a deep learning algorithm," *J. Invest. Dermatol.,* vol. 138, no. 7, pp. 1529–1538, 2018.

[13] S. Kalouche, A. Ng, and J. Duchi, "Vision-based classification of skin cancer using deep learning," *2015 Conduct. Stanf. Mach. Learn. Course CS 229 Taught,* 2016.

[14] Y. Li and L. Shen, "Skin lesion analysis towards melanoma detection using deep learning network," *Sensors,* vol. 18, no. 2, p. 556, 2018.

[15] H. Liao, "A deep learning approach to universal skin disease classification," *Univ. Rochester Dep. Comput. Sci. CSC,* 2016.

[16] A. R. Lopez, X. Giro-i-Nieto, J. Burdick, and O. Marques, "Skin lesion classification from dermoscopic images using deep learning techniques," *in 2017 13th IASTED international conference on biomedical engineering (BioMed),* pp. 49–54, 2017.

[17] S. K. Patnaik, M. S. Sidhu, Y. Gehlot, B. Sharma, and P. Muthu, "Automated Skin Disease Identification using Deep Learning Algorithm," *Biomed. Pharmacol. J.,* vol. 11, no. 3, pp. 1429–1437, 2018.

[18] J. Premaladha and K. S. Ravichandran, "Novel approaches for diagnosing melanoma skin lesions through supervised and deep learning algorithms," *J. Med. Syst.,* vol. 40, no. 4, p. 96, 2016.

[19] D. A. Shoieb, S. M. Youssef, and W. M. Aly, "Computer-aided model for skin diagnosis using deep learning," *J. Image Graph.,* vol. 4, no. 2, pp. 122–129, 2016.

[20] J. Yap, W. Yolland, and P. Tschandl, "Multimodal skin lesion classification using deep learning," *Exp. Dermatol.,* vol. 27, no. 11, pp. 1261–1267, 2018.

[21] H. Zhou, F. Xie, Z. Jiang, J. Liu, S. Wang, and C. Zhu, "Multi-classification of skin diseases for dermoscopy images using deep learning," *in 2017 IEEE International Conference on Imaging Systems and Techniques (IST),* pp. 1–5, 2017.

This publication is licensed under Creative Commons Attribution CC BY.

[22] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1251–1258, 2017.

[23] F. Iandola, M. Moskewicz, S. Karayev, R. Girshick, T. Darrell, and K. Keutzer, "Densenet: Implementing efficient convnet descriptor pyramids Technical Report," 2014.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *in Proceedings of the IEEE conference on computer vision and pattern recognition,* pp. 770–778, 2016.

[25] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Robinowich, "Going deeper with convolutions," i*n Proceedings of the IEEE conference on computer vision and pattern recognition,* pp. 1–9, 2015.

[26] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam "Mobilenets: Efficient convolutional neural networks for mobile vision applications." *arXiv preprint arXiv:1704.04861*(2017)

[27] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *ArXiv14091556 Cs*, Apr. 2015, Accessed: May 01, 2020.

[28] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning Transferable Architectures for Scalable Image Recognition", *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 8697-8710, 2018

[29] M. R. Prusty, T. Jayanthi, and K. Velusamy, "Weighted-SMOTE: A modification to SMOTE for event classification in sodium cooled fast reactors" *Progress in Nuclear Energy,* vol. 100, 355-364, 2017.

AUTHORS

First Author - Saswat Panda, Student(School of Computer Science and Engineering), Vellore Institute of Technology(Chennai Campus), spsaswat16194@gmail.com

Second Author - Abhishek Sunil Tiwari, Student(School of Computer Science and Engineering), Vellore Institute of Technology(Chennai Campus), ankit2632000@gmail.com

Third Author - Dr. Manas Ranjan Prusty, Assistant Professor(School of Computer Science and Engineering), Vellore Institute of Technology(Chennai Campus), manas.iter144@gmail.com

Correspondence Author - Saswat Panda, spsaswat16194@gmail.com, saswat.panda2018@vitstudent.ac.in, +91-9348838922

This publication is licensed under Creative Commons Attribution CC BY.