IFERP
ISSN (Online) 2394-2320
International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 9, Issue 9, September 2022

Classification Facial Skin and Treatment Suggestions for Good Skin Using Deep Learning with Region of Interest (ROI) Patches

[1] Satwik Sunnam, [2] Avuku Obulesh
[1] [2] Dept. of Artificial Intelligence, Vidya Jyothi Institute of Technology (Autonomous), Hyderabad, Telangana, India.
Corresponding Author Email: [1] sunnamsatwik19@gmail.com, [2] avuku06@gmail.com

Abstract*— Skin is the outermost layer of the human body. It serves as a protective layer for the internal organs of the human body. We should keep that healthy by knowing the respective skin type of ourselves rather than using different skin type products for skin without knowing our skin type can lead to damage to skin cells and maybe be responsible for epidermal diseases. There are four different skin types: Normal skin, Oily skin, dry skin, combination skin. We have taken two skin types: Normal skin, Oily skin of faces. The main objective of this paper is to classify facial skin as normal skin or oily skin. After classification, this paper suggests treatment for respective skin. For this task we used deep learning and open CV. In order to enhance performance, we exploit knowledge related to the human face structure. We train our model by employing automatically created facial regions of interest (ROI) to this end. By jointly learning the network parameters and optimized network output combination weights, each facial region appropriately contributes to the final classification result.

Index Terms: skin, deep learning, ROI, classification

I. INTRODUCTION

Skin types are classified into four categories normal skin, oily skin, dry skin, combination skin. Each of them has separate characteristics so we can easily differentiate each of them. Oily skin can be easily identified based on the following characteristics: it has enlarged, clearly visible pores, glossy shine, and thick skin. Normal skin can be identified as fine pores, soft and smooth texture, no blemishes. Dry skin is generally characterized by a feeling of tightness and roughness. It may also acquire an ashy grey colour, with occurrence of desquamation, itching, redness and small cracks. It may be temporary mainly caused by external factors such as weather, low humid conditions. Or it may be a long-term skin type for some people. Combination skin can be characterized by pore that looks larger than normal, Black heads, Shiny skin. Each different skin should be treated by different skin products based on skin type and use recommended products by dermatologist. To address that we came with a technique and developed a model which can differentiate skin types and recommending the skin products based on skin types. Skin texture and colour are critical indicators that people use to determine a range of culturally related characteristics regarding one another, such as fitness, ethnic origin, age, appearance, and income. Skin tone is a visual cue that people are present in photographs and videos. As a result, substantial study in the form of specialist and intelligent systems has been conducted over the last two decades. Concentrated on the identification of skin in videos and photographs. Skin identification is the mechanism of distinguishing between "normal skin" and "oil skin" regions in a photograph. Digital picture entails classifying pixels binary and doing fine segmentation to describe the skin's regional borders. At the moment, skin detection is a sophisticated method that requires not only model preparation, but additionally, various other processes, such as data pre- and post-processing. Skin detection is used in a wide variety of application domains: it is a prerequisite for facial recognition [1], and monitoring [2] as well as body tracking [3]. Detection of hands [4] and acknowledgment of gestures [5], biometric authentication (e.g., palm print recognition) [6] and objectionable content filtering [7] medical imaging. The aim of this work is to conduct a detailed review of how various expert systems (including artificial intelligence, deep learning, and machine learning systems) are being developed to address the issue of skin detection. There are just a few surveys on this subject: Although the works in [8, 9] are very old and only cover methods suggested prior to 2005, the surveys in [10, 11, 12] are more recent and have in-depth benchmarking databases and output data spanning almost two decades. In either case, none of the above surveys have a Comparing technique that fairly requires the use of the same experimental procedures and datasets. The aim of this work is not to conduct a survey. Latest research in this area has been enriched by the addition of deep learning methods [13-16] but also, and perhaps most significantly, to propose a structure for a fair contrast of methods.

The article organization is as follows. Section II describes complete proposed method steps. Section III discusses results after the performance evaluation of suggested methods and

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 9, September 2022

Section IV gives conclusions of this article.

II. PROPOSED METHODOLOGY

2.1. Model Architecture:

Our model consists of three convolution blocks with a max pool layer in each of them There's a fully connected layer with 128 units on top of it that's activated by "relu" activation function. Convolution neural networks are widely used in computer vision & where we deal with images. There are two variants in convolution networks 1-dimensional and 2-dimensional mostly, 2-dimensional is used to deal with images & 1-dimensional networks to deal with textual data. Convolutional layers will learn the local parameters in small windows in two dimensions. The important feature of convolution layers can learn spatial archeries of patterns by preserving spatial relationships. The Fig.1 shows the block diagram of the proposed one.

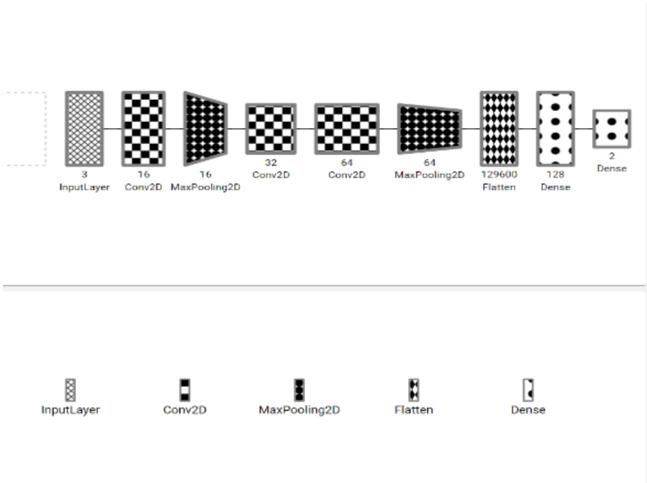


Fig.1: The architecture of the proposed model.

2.2 Feature Extraction:

Features are the characteristics of the object or place which makes the differentiability to specify the object/place uniquely.

Feature extraction is defined as the method/process of extracting important features of the image so the model neural networks will concentrate on important/relevant features rather than all features of the image. To get the important features from the image we need some kind of filters/abstractors which can be applied on every image in the dataset which are fed into the network so filters are applied on every fixed position of the image and extract the key features. Which is typically shown below.

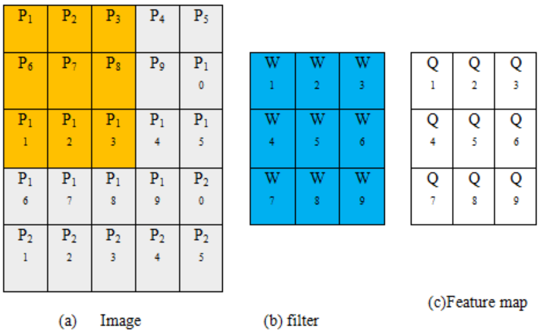


Fig. 2: Derivation of feature map with filter.

The Fig.2 represents the process of evaluating feature maps. Here Fig.2(a) represents input image (b) represents 3x3 filter and (c) represents feature map.

As we are using coloured images, we have s colour channels named (RGB) representing Red, Green, Blue. We apply the filter size of 3 on every image of the dataset. So, our filter size will look like 3*3*3 and it will cover each and every pixel of the image so that'll convert all the pixels into numerical representation. These representations will help in creating a feature map of every image in the dataset. Feature map is the end product of feature extraction.

2.3 Activation Function (ReLU):

In every ANN, every node will accept the output from the previous node. The transmission between the nodes are the features of the image. To filter the relevant information of our task we need to pass the important features of the image rather than all features from the previous layer to the next layer. So, we need some mathematical function to filter these features. Activation function comes into picture. There are normally three different kinds of activation functions: tanh, softmax, ReLU.In deep learning neural networks are fed with large amounts of data so it'll take a long time to compile; so high convergence speed of the model is important. The ReLU function is very fast in calculation & its convergence is faster than other two activation functions. It also helps us to get rid of the gradient vanishing problem which is the main cons of the tanh, sigmoid functions.

$$F(x)=Max(0,x) \qquad (1)$$

2.4 Pooling layer:

It's a kind of filter which downsamples the features that are obtained from the convolution layer. Pooling layer is always followed by the convolution layers so the output from each convolution layer is pooled by the pooling layer & sent to the next convolution layer. It helps in reducing the dimensionality of features & avoids overfitting. But the important point in the pooling layer is although it is reducing the dimension of features still it will store the important information of the image. The Fig.3 shows the process of max. pooling.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 9, September 2022

1	0	2	3
4	6	6	8
3	1	1	4
2	2	3	6

(a) Feature map

6	8
3	6

(b) Max pooling result

Fig. 3: Process of max pooling.

2.5 Flattening:

To classify our images into respective classes we need to flatten our data into a vector. Normally we use a dense layer for this task & it's generally added at the end of the layers. So, we can map each vector to a different class.

2.6 Fully Connected Layer:

Fully connected layer works as a classifier. It acts as a connection between all the layers of the model.

The connection between the layers is the weights of each layer which is blinding part of the model. It is located at the end of the rear part of the model.

After extracting relevant features of the image, the task is to classify the image so at the end they are fed into the fully connected layer which helps in classifying the image.

2.7 Compiling & Training the model:

We need to tune the weights of the layers of CNN so that we can reach the least point of loss.

"**Adam**" is one of the best algorithms & gives great performance on most deep learning problems.

We are giving the ROI done image to the model to predict the face type and our model have reached accuracy around of 90 percentage and we have predefined face washes which are curated by the dermatologist based on skin type after predicting the skin type, we are recommending the face wash based on the face type.

2.8 ROI(Region of Interest):

Image is represented as a mathematical array for the computer. To change the values in the array we want to access the values in that array. Removing the unwanted or not interested parts in the image by accessing that array is simply defined as selecting our region of interest.

Accessing each and every image array from a large dataset is a naive approach and so OpenCV has developed a function named select ROI which stores the upper bounds and lower bounds of the interested part of our image so that we can access our selected region rather than having the entire image. Which is basically slicing the image so that we can remove the not interested parts of the image.

2.9 Data Augmentation:

Deep learning models must be trained on large amount of data to increase the performance and generalization. It is a technique that applied on available data to create new training data. Transformation of images happens in data augmentation like shift, zoom flips & much more. Data augmentation will helps model such that it will never see the same image twice & helps preventing overfitting & increases the generalization of the model.

III. RESULTS AND DISCUSSIONS

The model is trained with 260 images where 130 images for each class. With augmentation the number of images increased for getting more accuracy. After training the model, accuracy, training loss and validation loss parameters are evaluated and plotted in diagrams like Fig.4.

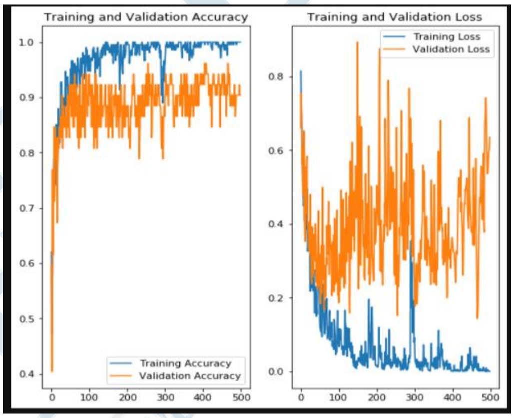


Fig. 4: Accuracy and loss with respect to training and validation.

Fig.5: Suggestions after classification of skin.

We have trained our model in three different epochs. We have received these accuracies of the model after training in different epochs which are described below:

3.1 Model trained under 200 epochs

As we have seen the Fig.6 training & validation accuracy for training & validation accuracy the it's kept on raising under 50 epochs when it has crossed 50 epochs the model training accuracy and validation accuracy has deviated training accuracy has gone to the top of the crest and it's been reached around of 99~100 accuracy. But the validation accuracy is keeps on suffering after crossing the 50 epochs it's been suffering in range of 80 % accuracy and it suddenly dropped the accuracy to ~70 % around 100 epochs and the accuracy is maintained in the range of 75-80 % accuracy until 180 epochs at final stages of epochs at 200 epochs the accuracy of the validation data has reached around 86% accuracy. Model performance is not only determined by the accuracy but also loss produced by the model during training and validation under training.

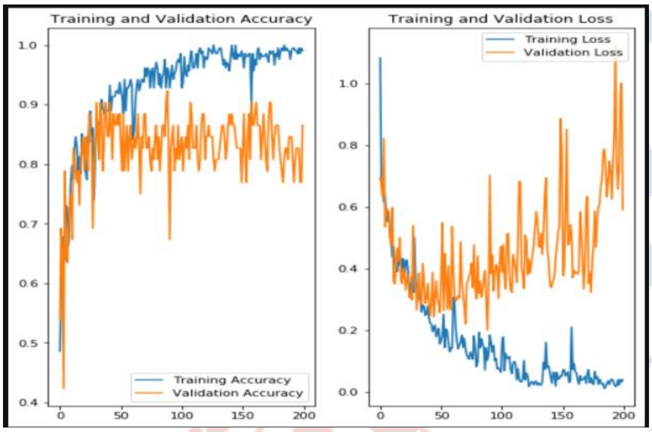


Fig.6: Accuracy and loss with respect to training and validation for 200 Epochs.

Model's training loss started around 100% in the starting of the model and started downhill and sticked down to 5% loss after completing the all epochs. As we see the validation loss it's started around of 80% and kept on decreasing and reached 30% in 50 epochs and kept on suffering between 30% and 60% accuracy under 100 epochs and it's been in the range of 30% -80 % accuracy under 150 epochs and there is sudden raise of accuracy and it's kept on raising and suffered between 80 % and ~90 % accuracy and it's settled around at 60%

```
y_pred=model.predict(val_ds)
predicted_cateogories=tf.argmax(y_pred,axis=1)
true_cat=tf.concat([y for x,y in val_ds],axis=0)
confusion_matrix(predicted_cateogories,true_cat)

array([[27,  3],
       [ 4, 18]], dtype=int64)
```

Fig.7: Confusion matrix for 200 epochs.

3.1.1 Confusion Matrix

A confusion matrix is a table that is often used to describe the performance of a classification model on a set of validation data for which the true values are known. As we are worked under binary classification problem and our label names are oily skin and normal skin, we have got a 2*2 confusion matrix Which can be shown below:

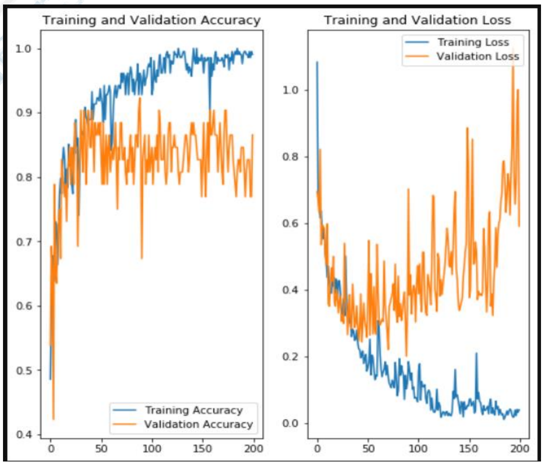
Let's draw an example confusion matrix which tells us what is it going to say about performance: Using this matrix, we can find the accuracy of the model which can be given by this formula given below

$$Accuracy = \frac{TP+TN}{P+N} \qquad (2)$$

Where TP- True positive, TN- True negative, P- predicted positive, N- predicted negative. By doing that calculation we will get accuracy around 86 % for 200 epochs.

```
outside_pics/oilyc1.jpeg Your Face is most likely belongs to oily skin with a 99.93 percent confidence.
you can choose these recommended facewashes :)
Plum Green Tea Pore Cleansing Face Wash
Biotique BIO Honey Gel Face Wash
Natural Vibes ~ Ayurvedic Tea Tree Face Wash
Forest Essentials Mashobra Honey, Lemon and Rosewater Facial Cleanser
outside_pics/normalc1.jpeg Your Face is most likely belongs to normal skin with a 99.58 percent confidence.
you can choose these recommended face washes :)
Himalaya Herbals Purifying Neem Face Wash
Neutrogena Deep Clean Facial Cleanser
St. Botanica Vitamin C Gentle Foaming Brightening Face Wash
Mcaffeine Neem Face Wash
Pondâ€™s Pure White Anti-Pollution + Purity Face Wash
Clean & Clear Foaming Face Wash
```

Fig.8: Suggestions given after classification of skin for 200 epochs.



We have taken images rather than from training and validation data and we have seen the accuracy of the prediction by the model. As we can see the Fig.8 actual label of the image is oily and its prediction is oily skin with 99.93 accuracy and it's given suggestions of face washes for that skin type.

Also, for another skin type it's predicted the same actual label i.e., Normal skin and its accuracy is 99.58 and it gave suggestions of face washes for normal skin type.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 9, September 2022

3.2 Model trained under 300 epochs

As we can see in Fig.9 the training accuracy is in between 96%-100% under 100 epochs and there is a sudden drop of the accuracy to 93% of the training data at the 100th epoch. If we see the accuracy of the training data under 100-200 epochs the accuracy is suffering in between 95-98% and if we see the accuracy of training in between 200-300 epochs it's same as 100-200 epochs but at end of the epochs it's settled down 98% accuracy.

Now we have to look at validation accuracy if we subdivide the epochs and analyse if we try to observe under 100 epochs the accuracy started at around 78 % and suffered between 85-90% accuracy. There is a sudden drop of the accuracy at 100th epoch to 78%. If we are able to see the accuracy of the validation data under 100-200 epochs the accuracy has suffered between 82-88% accuracy and it reached a peak of 92% in between these epochs. Validation data accuracy under 200-300 epochs accuracy suffered between the 83-90% accuracy and in between it reached a peak of 92% in between these epochs and finally settled down around 82%. As we see the training loss it's suffered between 5% -10% under 100 epochs and it reached around 20% in between 100-200 epochs and settled down to 10% at the end of 200 epoch after we can look at 200-300 epochs it's suffered in the range 10%. Validation loss started around 60% and settled down to 80% loss in between the 100 epochs and in between 100-200 epochs the loss is around 30%-40% and in between it has touched a peak of 100% loss and also more than that. If we see in between 200-300 epochs the loss is in between in the range 40%-50% and it too touched a peak of 100% loss and finally it ended up with 30% loss.

```
y_pred=model.predict(val_ds)
predicted_cateogories=tf.argmax(y_pred,axis=1)
true_cat=tf.concat([y for x,y in val_ds],axis=0)
confusion_matrix(predicted_cateogories,true_cat)

array([[26,  4],
       [ 5, 17]], dtype=int64)
```

Fig.9: Confusion matrix for the model for 300 epochs.

The accuracy of the model which is trained under 300 epochs is around 82%. By doing the calculation which is mentioned above we can find out the accuracy.

We have taken images rather than from training and validation data and we have seen the accuracy of the prediction by the model. As we can see the Fig.10 actual label of the image is oily and its prediction is oily skin with 99.94 accuracy and it's given suggestions of face washes for that skin type.

```
outside_pics/oilyc1.jpeg Your Face is most likely belongs to oily skin with a 99.94 percent confidence.
you can choose these recommended facewashes :)
Plum Green Tea Pore Cleansing Face Wash
Biotique BIO Honey Gel Face Wash
Natural Vibes ~ Ayurvedic Tea Tree Face Wash
Forest Essentials Mashobra Honey, Lemon and Rosewater Facial Cleanser
outside_pics/normalc1.jpeg Your Face is most likely belongs to normal skin with a 99.94 percent confidence.
you can choose these recommended face washes :)
Himalaya Herbals Purifying Neem Face Wash
Neutrogena Deep Clean Facial Cleanser
St. Botanica Vitamin C Gentle Foaming Brightening Face Wash
Mcaffeine Neem Face Wash
Pondâ€™s Pure White Anti-Pollution + Purity Face Wash
Clean & Clear Foaming Face Wash
Aroma Magic Neem And Tea Tree Face Wash
Lotus Herbals Tea Tree And Cinnamon Anti-Acne Oil Control Face Wash
```

Fig.10: Suggestions after classification of skin for 300 epochs.

Also, for another skin type it's predicted the same actual label i.e., Normal skin and its accuracy is 99.94 and it gave suggestions of face washes for normal skin type.

3.3 Model trained under 500 epochs

As we can see in Fig.11 the training accuracy is in between 82%-95% under 100 epochs. If we see the accuracy of the training data under 100-200 epochs the accuracy is suffering in between 95-98% and if we see the accuracy of training in between 200-300 epochs it's same as 100-200 epochs but at end of the epochs it's settled down 98%-99% accuracy. In case of 400-500 epochs the accuracy is nearby to 100% on training data.

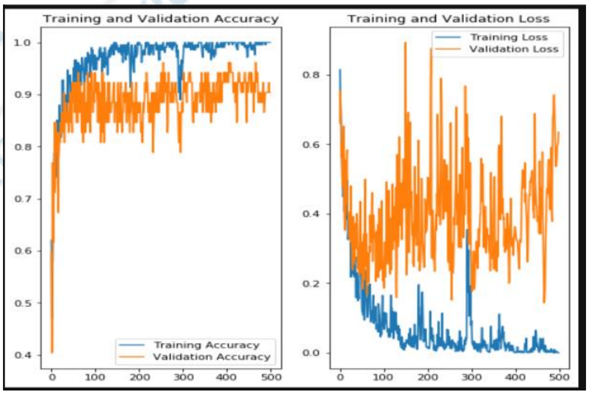


Fig.11: Accuracy and loss of training and validation for 500 epochs.

Now we have to look at validation accuracy if we subdivide the epochs and analyse if we try to observe under 100 epochs the accuracy started at around 30% and suffered between 85-90% accuracy. If we are able to see the accuracy of the validation data under 100-200 epochs the accuracy has suffered between 82-90% accuracy and it reached a peak of 92% in between these epochs. In the case of 400-500 epochs validation accuracy is in between 91-93% accuracy and finally settled down. Validation data accuracy under 200-300 epochs accuracy suffered between the 83-90% accuracy and in between it reached a peak of 92% in between these epochs and finally settled down around 82%.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 9, September 2022

As we see the training loss it's suffered between 80% -10% under 100 epochs and it reached around 10%-5% in between 100-200 epochs we can look at 200-300 epochs it's suffered in the range 10% and in case of 400-500 epochs it's settled down to 5% loss. Validation loss started around 80% and settled down to 20% loss in between the 100 epochs and in between 100-200 epochs the loss is around 30%-40% and in between it has touched a peak of 90% loss. If we see in between 200-300 epochs the loss is in the range 20%-30% and it too touched a peak of 80% loss and finally it ended up with 30% loss. In the case of 400-500 epochs, it suffered between 30% -80% and settled around 40%.

```
y_pred=model.predict(val_ds)
predicted_cateogories=tf.argmax(y_pred,axis=1)
true_cat=tf.concat([y for x,y in val_ds],axis=0)
confusion_matrix(predicted_cateogories,true_cat)

array([[28,  2],
       [ 3, 19]], dtype=int64)
```

Fig. 12: Confusion matrix for 500 epochs.

The accuracy of the model which is trained under 500 epochs is around 90% by doing the calculation which is mentioned above. We can find out the accuracy by using Fig.12.

```
outside_pics/oilyc1.jpeg Your Face is most likely belongs to oily skin with a 99.99 percent confidence.
you can choose these recommended facewashes :)
Plum Green Tea Pore Cleansing Face Wash
Biotique BIO Honey Gel Face Wash
Natural Vibes ~ Ayurvedic Tea Tree Face Wash
Forest Essentials Mashobra Honey, Lemon and Rosewater Facial Cleanser
outside_pics/oilyc2.jpeg Your Face is most likely belongs to oily skin with a 100.00 percent confidence.
you can choose these recommended facewashes :)
Plum Green Tea Pore Cleansing Face Wash
Biotique BIO Honey Gel Face Wash
Natural Vibes ~ Ayurvedic Tea Tree Face Wash
Forest Essentials Mashobra Honey, Lemon and Rosewater Facial Cleanser
```

Fig.13: Suggestions after classification of skin for 500 epochs.

We have taken images rather than from training and validation data and we have seen the accuracy of the prediction by the model. As we can see the Fig.13 actual label of the image is oily and its prediction is oily skin with 99.99% accuracy and it's given suggestions of face washes for that skin type. Also, for another skin type it's predicted the same actual label i.e., Normal skin and its accuracy is 100% and it gave suggestions of face washes for normal skin type.

Table 1: Accuracy of existing and proposed models for different Epochs.

	200 Epochs (Round Off value)	300 Epochs (Round Off value)	500 Epochs (Round Off value)
Alex Net	82%	71%	86%
Google Net	80%	83%	88%
Proposed	86%	82%	90%

Table 1 describes the comparison of existing models and proposed one and respective discussions given above.

IV. CONCLUSION AND FUTURE

As we have discussed the detail analysis of each model and its accuracy and losses and we can say that the model which is trained under 500 epochs is better one and we can further enhance the model with large number of training examples and a greater number of epochs so that it can find complex patterns and make predictions and also train models under GPU or TPU for faster computation rather than CPU. As we have worked under two types of skin named normal skin type and oily skin type, we can make our work wider and make work on other skin types so that we can use this model in the market for real recommendation of face products just taking a picture of the face.

REFERENCES

[1] Hsu, R. L., Abdel-Mottaleb, M., & Jain, A. K. (2002). Face detection in color images. IEEE Transactions on Pattern Analysis and Machine Intelligence. https://doi.org/10.1109/34.1000242

[2] De-La-Torre, M., Granger, E., Radtke, P. V. W., Sabourin, R., & Gorodnichy, D. O. (2015). Partially -supervised learning from facial trajectories for face recognition in video surveillance. Information Fusion. https://doi.org/10.1016/j.inffus.2014.05.006

[3] Argyros, A. A., & Lourakis, M. I. A. (2004). Real-time tracking of multiple skin-colored objects with a possibly moving camera. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://doi.org/10.1007/978-3-540-24672-5_29

[4] Roy, K., Mohanty, A., & Sahay, R. R. (2017). Deep Learning Based Hand Detection in Cluttered Environment Using Skin Segmentation. 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), 640–649. https://doi.org/10.1109/ICCVW.2017.81

[5] Han, J., Award, G. M., Sutherland, A., & Wu, H. (2006). Automatic skin segmentation for gesture recognition combining region and support vector machine active learning. Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition, 237–242. https://doi.org/10.1109/FGR.2006.27

[6] Sang, H., Ma, Y., & Huang, J. (2013). Robust Palmprint Recognition Base on Touch-Less Color Palmprint Images Acquired. Journal of Signal and Information Processing, 04(02), 134–139. https://doi.org/10.4236/jsip.2013.42019

[7] Lee, J.-S., Kuo, Y.-M., Chung, P.-C., & Chen, E.-L. (2007). Naked image detection based on adaptive and extensible skin color model. Pattern Recognition, 40, 2261–2270. https://doi.org/10.1016/j.patcog.2006.11.016

[8] C. Prema, & Manimegalai, M. (2012). Survey on Skin Tone Detection using Color Spaces. International Journal of Applied Information Systems (IJAIS), 2(2), 18–26.

[9] Kakumanu, P., Makrogiannis, S., & Bourbakis, N. (2007). A survey of skin-color modeling and detection methods. Pattern Recognition, 40(3), 1106–1122. https://doi.org/10.1016/j.patcog.2006.06.010

[10] Chen, W., Wang, K., Jiang, H., & Li, M. (2016). Skin color modeling for face detection and segmentation: a review and a new approach. Multimedia Tools and Applications, 75(2), 839–862
[11] Mahmoodi, M. R., & Sayedi, S. M. (2016). A Comprehensive Survey on Human Skin Detection. International Journal of Image, Graphics and Signal Processing (IJIGSP), 8(5), 1–35. https://doi.org/10.5815/ijigsp.2016.05.01
[12] Naji, S., Jalab, H. A., & Kareem, S. A. (2018). A survey on skin detection in colored images. Artificial Intelligence Review. https://doi.org/10.1007/s10462-018-9664-9
[13] Xu, T., Zhang, Z., & Wang, Y. (2015). Patch-wise skin segmentation of human body parts via deep neural networks. Journal of Electronic Imaging, 24(4), 043009. https://doi.org/10.1117/1.JEI.24.4.043009
[14] Zuo, H., Fan, H., Blasch, E., & Ling, H. (2017). Combining Convolutional and Recurrent Neural Networks for Human Skin Detection. IEEE Signal Processing Letters, 24(3), 289–293.
[15] Kim, Y., Hwang, I., & Cho, N. I. (2017b). Convolutional neural networks and training strategies for skin detection. IEEE International Conference on Image Processing (ICIP), 3919–3923. https://doi.org/10.1109/ICIP.2017.8297017
[16] Ma, C., & Shih, H. (2018). Human Skin Segmentation Using Fully Convolutional Neural Networks. IEEE 7th Global Conference on Consumer Electronics (GCCE), 168–170. https://doi.org/10.1109/GCCE.2018.8574747