Int. j. adv. multidisc. res. stud. 2023; 3(1):552-559

Received: 06-12-2022
Accepted: 16-01-2023

International Journal of Advanced Multidisciplinary Research and Studies

ISSN: 2583-049X

A Comparative Study on Face Recognition Using Deep Learning Approach

[1] Arulnesan Priscilah Nivetha, [2] Mohammed Satheek Suhail Razeeth, [3] Prasanth Keerthana
[1, 3] Trincomalee Campus, Eastern University, Sri Lanka
[2] South Eastern University, Sri Lanka

Corresponding Author: **Arulnesan Priscilah Nivetha**

Abstract
Biometric systems are utilized to examine and confirm an individual's identity for verification. There are various biometric methods such as fingerprint scans, voice recognition and iris scan are available. Face recognition is one of the significant methods that has been used in many kinds of applications for security and surveillance purposes nowadays. There are several methods available from the early days to recent times for face recognition. Deep learning is one of the most used techniques in different applications of computer vision. It is a technique that facilitates automatic feature learning and classifying of images. In this paper, a CNN-based framework has been proposed and evaluated with some of the transfer learning frameworks and with the Google Teachable Machine-created model using a newly created dataset of faces 1500 images. Among all the methods, MobileNetV2 and DenseNet169 transfer learning models obtained fine performance with an accuracy of 100% with almost no loss.

Keywords: Face Dataset, Face Recognition, Google Teachable Machine, Convolutional Neural Network (CNN), Transfer Learning

1. Introduction

Many arising applications, from policing business endeavors, demand the business to have an authentication system to control access to particular information or confidential matters or equipment related to their organization. Numerous human authentication systems have been created, such as secured pins, passwords, signatures, and biometric systems. Biometric systems such as voice recognition, face recognition, fingerprint scanning, and iris recognition play a significant role. Of all the biometric systems, face recognition has become popular and trustworthy. A new sense of urgency in technology has become essential for adopting the latest and flawless face recognition process.

Face recognition has turned into a well-known area of exploration in computer vision which is used to analyze and understand the images in many emerging applications. It is a technique for recognizing or confirming the identity of a person using his or her face in an image or a video. Such face recognition systems aim to execute the system model for a specific face and separate it from an enormous number of stored face databases with many variations. But recognition of faces is very challenging due to the variety in looks, postures and brightening of the environments.

Face recognition techniques have moved essentially throughout the long term and numerous researchers have been chipping away at face recognition. Several methods have been used from traditional techniques up to the present. But in later times, many methods have been framed for face recognition with the assistance of deep neural networks. Deep neural networks are the subfield of Artificial intelligence (AI) that include neural networks for sorting out answers for the issues managing computerized reasoning [1]. These networks imitate the neocortex of the human cerebrum which has several neurons. These neurons are utilized to fabricate the neural network in deep learning models [1] and consist of several parameters and layers in the middle of input and output [2]. Deep learning facilitates automatic feature learning and consists of different types of neural network models. Deep learning approaches such as the convolutional neural network (CNN) have recently displaced traditional face recognition methods. CNN is a sort of artificial neural network that utilizes the convolution approach to deal with extricating characteristics from input data to expand the number of characteristics [3]. This was developed around the 1980s [4]. CNNs are comprised of multiple layers where each input image has to be passed through a series of layers. It contains convolutional layers, pooling layers and fully connected layers where each performs some predetermined functions on its input data [5]. Fig 1 shows a typical architecture of a CNN [6].

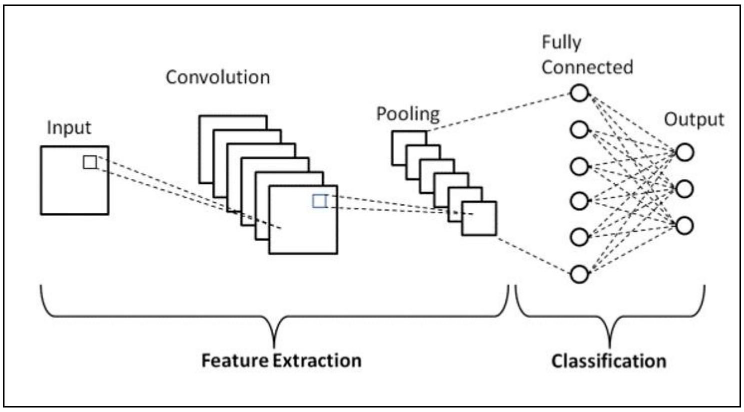

Fig 1: Typical CNN architecture

In this study, a convolutional neural network (CNN) based framework has been proposed to classify the face images of a newly created face dataset which consists of face images from five randomly chosen individuals. This proposed model has been analyzed with the Google Teachable Machine model and a set of pre-trained transfer learning-based models such as VGG16, VGG19, MobileNetV2, DenseNet121 and DenseNet169. The main aim of this paper is to classify the face images using the proposed model and analyze them with other models in terms of performance metrics.

The rest of the paper is organized as follows. The literature is reviewed in Section II. The experimental methods are presented in Section III. The comparison of the methods adopted in this study and its outcomes are presented in Section IV and lastly, the conclusion and future works of the paper are presented in Section V.

2. Literature Review

In this section summarization of some studies related to face recognition is presented.

In the comparative study [7], two distinct datasets were utilized, which are the FERET dataset of 400 images and the KREMIC dataset of 780 images. The skin color was used to detect the faces from the image background and then converted into a grayscale image. Histogram local feature-based extraction was applied to extract features from detected faces. Voting and Random Subspace ensemble learning algorithms were used along with classifiers such as ANN, k-NN, SVM, RF, C4.5 Decision Tree, Random Tree, REP Tree, LAD Tree, NB, Rotation Forest and CART for classification. K-fold cross-validation technique was applied to each algorithm for the performance evaluation, where the best performance was obtained with a value of 10 for k. The classifier performance has been evaluated using error rate using the performance metrics such as accuracy, F-measure, ROC Area and Kappa. For the FERET database, the Voting and Random Subspace ensemble methods with RF obtained the best outcome at 99.25%. There are likewise remarkable outcomes taken by both ensemble methods, with 99% and 98.75% accuracies. For the KREMIC database, voting ensemble methods with the RF algorithm obtained the best outcome at 96.79%. Any remaining techniques like ANN, k-NN, SVM, NB and Rotation Forest obtained above 97% of accuracy for the FERET database and above 92% of accuracy for the KREMIC database. Tree algorithms have less accomplishment than other algorithms when ensemble methods are applied in both datasets.

The study [8] used a dataset with a total of 9,000 human faces of real and fake images which are classified into four different classes and are available in the Kaggle repository. Then the images were normalized to ImageNet standard and five augmentation techniques were applied to generate more data. The dataset has been trained using the proposed model and other four pre-trained models such as VGG16, ResNet50, MobileNet, and InceptionV3 for a total of 150 epochs. The proposed model has been created with 8 convolutional layers followed by a fully connected hidden layer with softmax activation function in output layers, Adam optimizer for network optimization, the learning rate of 0.0001 and the callback feature of the library Keras was used to save the best validation accuracy. 100% of training accuracy and 95.21% of validation accuracy were obtained in the proposed model. Based on the overall results, InceptionV3 and ResNet50 were the best models then come MobileNet model in third, then the proposed model was fourth and VGG16 was ranked last. [9] Presented a study on Arab ethnicity classification based on the face image datasets of three distinct classes: Gulf Cooperation Council countries (GCC), Levant and Egypt. Other datasets such as Racial Faces in-the-Wild (RFW), BUPT-Transferface, FERET and UTK were used for the experiments as well. Dlib's pre-trained face detector based on a modification to the standard Histogram of Oriented Gradients with Linear SVM was applied to the images to detect the faces and different data augmentation techniques were used. For the classification problem supervised and unsupervised learning was carried out. In the supervised learning model, a pre-trained CNN model with ResNet-50 layers architecture has been used with hyperparameters tuning. Deep clustering methods such as DEC, IDEC, and DynAE were used in the unsupervised learning model. All models were evaluated on the above distinct datasets using evaluation metrics. The classification model obtained the best accuracy of 0.5697 with the unbalanced Arab dataset and 0.5212 with the balanced Arab dataset. This model could be able to identify up to 88% of Arabs when combined with other datasets, but the classification performance was not good in classifying Arab labels together. The clustering model obtained poor performance. Further RFW dataset obtained the best accuracy and higher NMI and ARI on all models.

A transfer-learning-based face recognition framework has been proposed [10] to detect autistic children using a dataset of 2936 images of normal and autistic children available in the Kaggle repository which were then split into training, validation and testing datasets. In the training process, CNN, pre-trained improved CNN and other machine learning models were used. Classifiers such as Adaboost, Decision Tree (DT), Gradient Boosting (GB), K-Nearest Neighbour (KNN), Logistic Regression (LR), Multi-layer Perceptron (MLP), Naïve Bayes (NB), Random Forest (RF), Support Vector Machine (SVM), Gradient Boosting (XGB), Convolutional Neural Network (CNN) and pre-trained CNNs such as DenseNet121, ResNet50, VGG16, VGG19, MobileNet-V1 and MobileNet-V2 were used. Three batch normalization (BN) and two fully connected (FC) layers are attached in a steady progression before the output layer in each model. The improved MobileNet-V1 model is the improved version of traditional MobileNet-V1 with some added layers. The improved MobileNet-V1 model obtained the highest accuracy, AUC, f-measure, g-mean, sensitivity and specificity of 83%, 17% fall-out and miss rate and outperformed other classifiers in the ROC curve during the validation set evaluation. Likewise improved MobileNet-V1 model obtained the best outcome of 90.67% for all evaluation metrics, the lowest fall-out and miss rates and outperformed other classifiers in the ROC curve during the test set evaluation. The clustering technique was applied to the autism dataset to identify the sub-types using a k-means algorithm which then lead to a multiclass classification using the improved MobileNet-V1 model which outperformed the base MobileNet-V1 model classification and obtained the best performance when k=2. Further, they introduced a few restrictions, for example, hardly any facial images were utilized and their quality was not promising and improved MobileNet-V1 has not given a more steady prescient performance for a greater number of autistic sub-types.

3. Methods

Here, we discuss the compositional parts and the process engaged with building and fostering our model utilizing a deep learning approach. The dataset is introduced by gathering images from random people. Once the dataset was prepared, various models were applied to solve the classification problem. Several performance matrices were used to evaluate each model.

A. Data

The dataset for this study comprises 1500 images of human faces. Five individuals were arbitrarily chosen and 300 images with various stances, backgrounds, brightening and expressions were utilized by every individual. The components of the pictures were 224 × 224, with 224 pixels in height and 224 pixels in width. The dataset was split into the training dataset and testing dataset with an 80:20 ratio.

B. Google Teachable Machine: Deep learning model

A teachable Machine is a web-based tool that makes making machine learning models quick, simple, and open to everybody [11].

1. The collected dataset was trained using this tool and tested for performance by changing the learning rate, batch size, and epochs.
2. The performance was evaluated in terms of performance metrics such as accuracy and confusion matrix.

C. The proposed deep learning model

The proposed CNN-based deep learning model consists of eight layers, three convolutional layers and three pooling layers, a fully connected layer, and then an output layer. It was obtained by changing the factors and structures which then achieved at a learning rate of 0.001, a batch size of 32 and 10 epochs.

1. Pre-processing: The images have been grouped into five distinct types, labeled and uploaded to Google Drive properly.
2. Data augmentation: The amount of the dataset has been increased by using ImageDataGenerator by rescaling, rotating, shearing, flipping, shifting and zooming the images.
3. The dataset has been trained using the own created model. Rectified Linear Unit Layer (ReLU) activation function has been used in all convolutional layers and the fully connected layer and SoftMax activation function have been used in the output layer. Dropout layers are also used to normalize the network from overfitting. Batch Normalization was applied to the layers to improve the performance and Adam optimizer has been used to optimize the network.
4. Callback functions such as model checkpoints and early stopping have been applied to the model [12].
5. The performance of the proposed model was evaluated using training and validation accuracy and training and validation loss.

D. Transfer learning models

Different pre-trained models such as VGG16 [13], VGG19 [14], MobileNetV2 [15], DenseNet121 [16] and DenseNet169 [17] were used.

1. Pre-processing: The input image size was changed for each model accordingly. The training set was used for model training and contained a sum of 1200 images from five distinct classes and the test set was kept isolated with 300 images which were then uploaded to Google.
2. Data augmentation: The images were rescaled, sheared, zoomed, and flipped using ImageDataGenerator.
3. Each model was trained using the training dataset, tested separately and evaluated using training and validation accuracy and training and validation loss. For all the models Adam optimizer and SoftMax activation function have been used. All the models were implemented for up to 5 epochs.

   The Keras library in Python was used and implemented in the Google Colab tool for the proposed and transfer learning models.

4. Results and Discussion

In this section, the outcomes of the execution of the methods referenced in the above-proposed works will be specified.

A. Google Teachable Machine: Deep learning model

The Teachable Machine has been evaluated by changing the learning rate, batch size and epochs.

Fig 2-7 show the confusion matrices for the learning rate of 0.1 while changing the batch size and epochs.

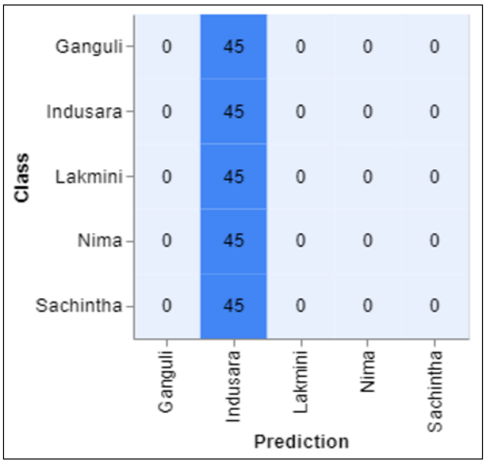

Fig 2: Confusion Matrix for a batch size of 16 & epochs of 5

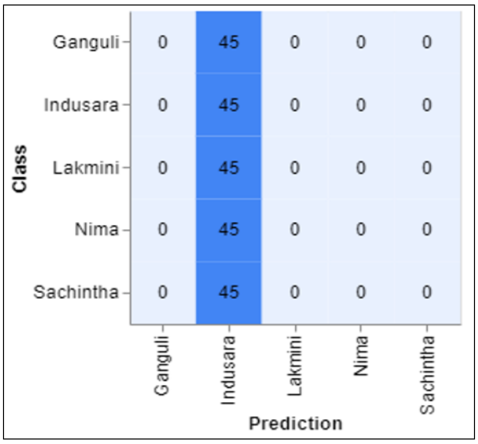

Fig 3: Confusion Matrix for a batch size of 32 & epochs of 5

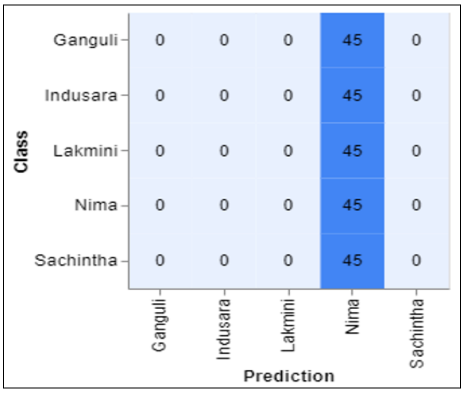

Fig 4: Confusion Matrix for a batch size of 16 & epochs of 10

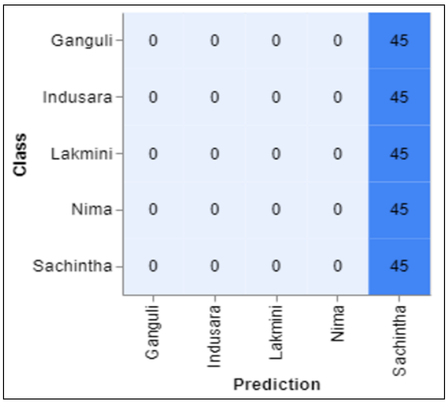

Fig 5: Confusion Matrix for a batch size of 32 & epochs of 10

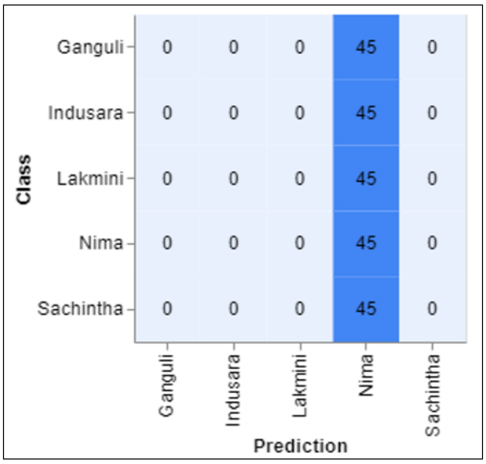

Fig 6: Confusion Matrix for a batch size of 16 & epochs of 15

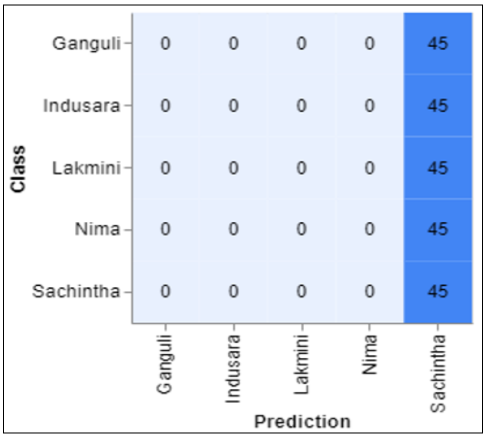

Fig 7: Confusion Matrix for a batch size of 16 & epochs of 20

When classifying the data using this deep learning model designed by the Teachable machine, the above figures clearly showed that the accuracy of classification didn't reach 100%.

Therefore, the learning rate has been changed to 0.01 and different batch sizes with epochs were carried out.

Fig 8-10 shows the confusion matrix, accuracy graph and loss graph respectively for the learning rate of 0.01 with a batch size of 16 and epochs of 5. It clearly shows that the designed model with a 0.01 learning rate with a batch size of 16 can classify the data accurately at epochs 5 since the accuracy reached 100% and the loss reached zero.

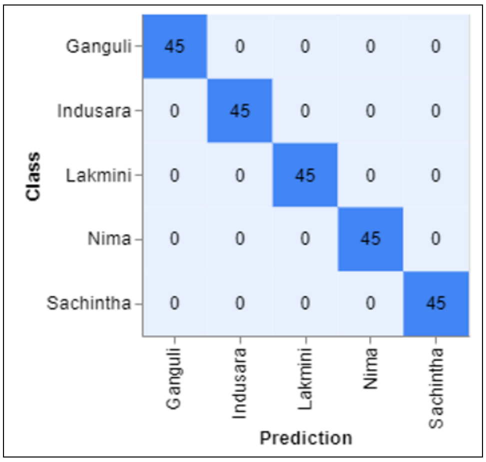

Fig 8: Confusion Matrix for learning rate of 0.01, batch size of 16 & epochs of 5

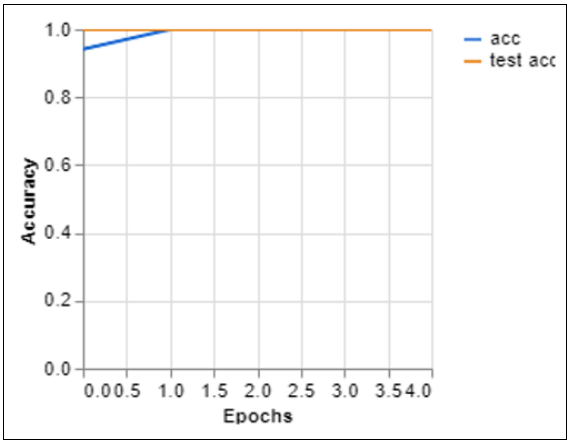

Fig 9: Accuracy graph for learning rate of 0.01, batch size of 16 & epochs of 5

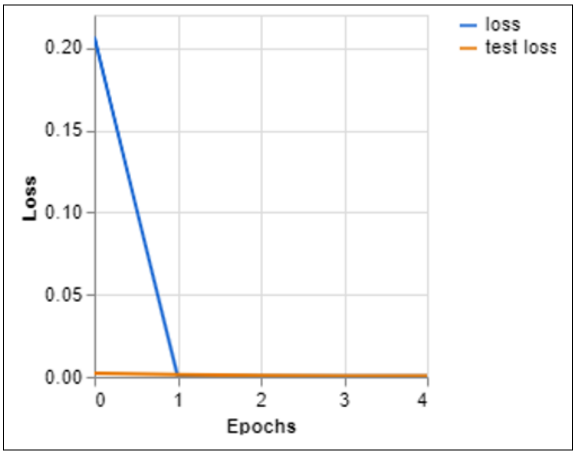

Fig 10: Loss graph for learning rate of 0.01, batch size of 16 & epochs of 5

B. The proposed deep learning model

Fig 11 shows the learning curve of the proposed CNN-based deep learning model during the training and validation stage. It is seen that both the validation and training accuracy have been increasing. However, the training accuracy has some gradual increment where there were some ups and downs in the validation accuracy curve with the number of epochs.

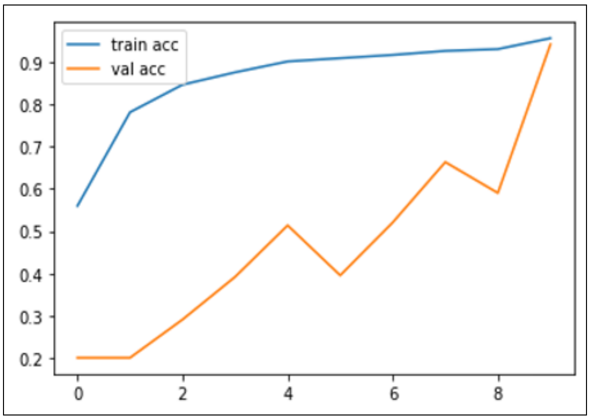

Fig 11: Training and validation accuracy curve for the proposed model

Fig 12 shows the loss curve of the proposed CNN-based deep learning model during the training and validation stage. It is seen that both the validation and training loss have been decreasing. However, the training loss has some gradual decrement where there were some ups and downs in the validation loss curve with the number of epochs.

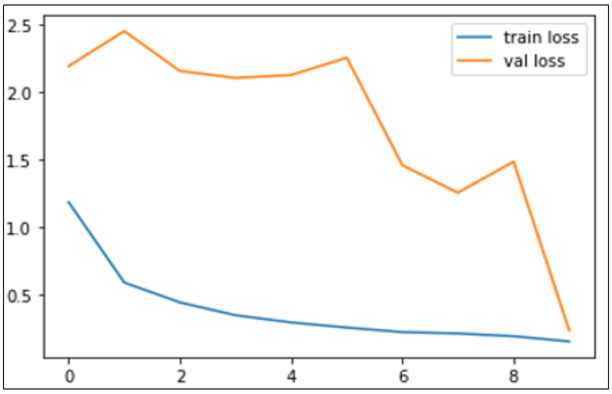

Fig 12: Training and validation loss curve for the proposed model

Overall, it is proven that the proposed model works well.

C. Transfer learning models

1) VGG16 Model

Fig 13 illustrates the training and validation accuracy of the VGG16 model. Fig 14 illustrates the training and validation loss of the VGG16 model. The training and validation accuracy increased and reached 100%, whereas the training and validation loss decreased with the increased number of iterations.

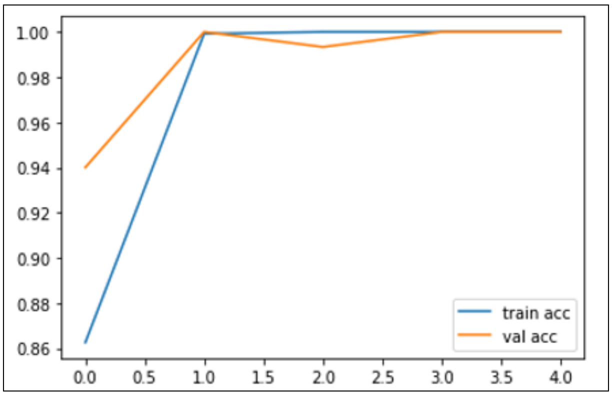

Fig 13: Validation and training accuracy graph of the VGG16 model

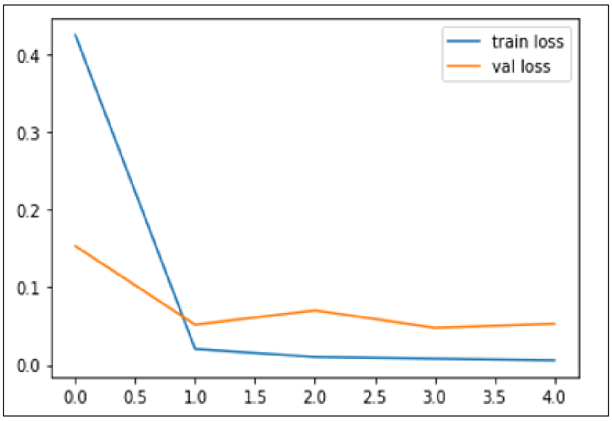

Fig 14: Validation and training loss graph of the VGG16 model

2) VGG19 Model

Fig 15 illustrates the training and validation accuracy of the VGG19 model. Fig 16 illustrates the training and validation loss of the VGG19 model. The training and validation accuracy increased and the training accuracy reached 100%, whereas the validation accuracy reached 93.67%. The training and validation loss decreased with the increased number of iterations.

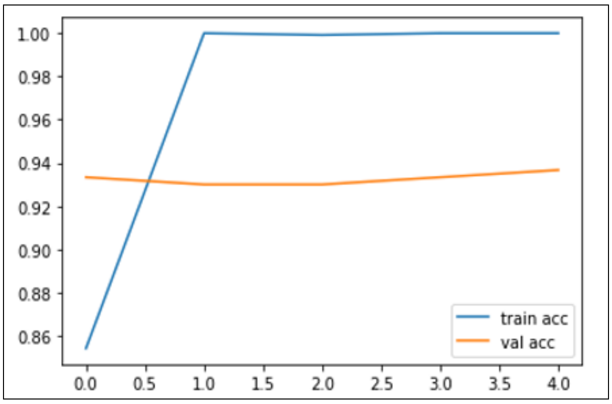

Fig 15: Validation and training accuracy graph of the VGG19 model

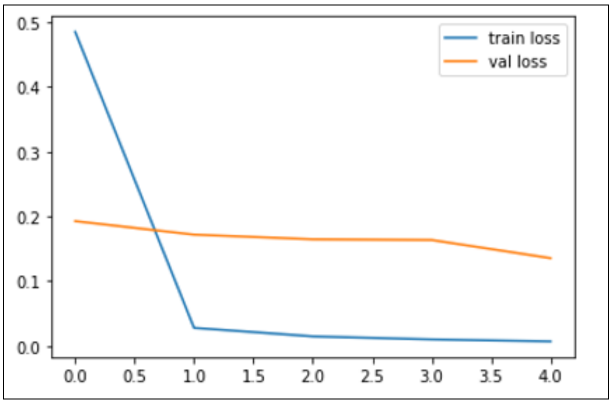

Fig 16: Validation and training loss graph of the VGG19 model

3) MobileNetV2 Model

Fig 17 illustrates both the training and validation accuracy of the MobileNetV2 model. Fig 18 illustrates both the training and validation loss of the MobileNetV2 model. The training and validation accuracy increased and reached 100%. The training and validation loss also decreased with the increased number of iterations.

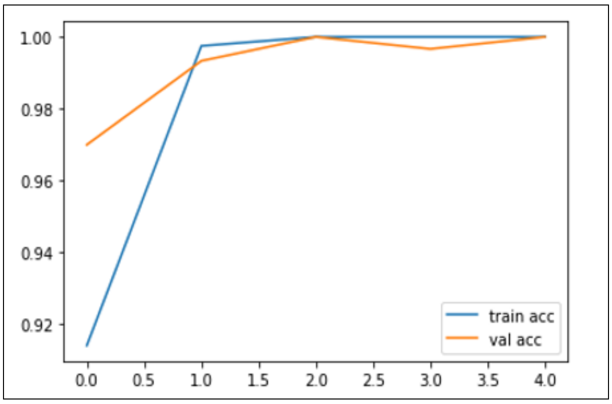

Fig 17: Validation and training accuracy graph of MobileNetV2 model

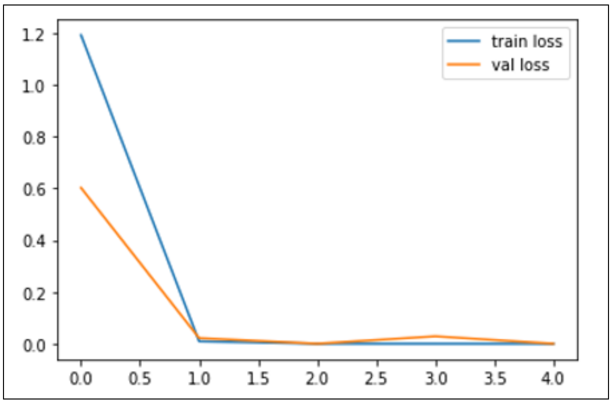

Fig 18: Validation and training loss graph of MobileNetV2 model

4) DenseNet121 Model

Fig 19 shows both the training and validation accuracy of the DenseNet121 model. Fig 20 shows both the training and validation loss of the DenseNet121 model. The training and validation accuracy increased and the training accuracy reached 100%, whereas the validation accuracy had a down in between, however finally reached 98% of accuracy. The training loss decreased with the increased number of iterations, but there were some flaws in between the validation loss.

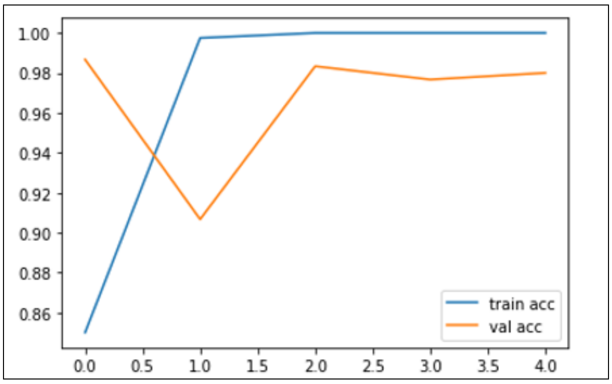

Fig 19: Validation and training accuracy graph of the DenseNet121 model

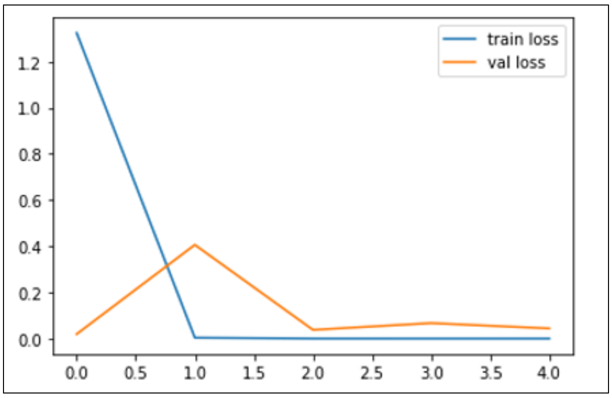

Fig 20: Validation and training loss graph of the DenseNet121 model

5) DenseNet169 Model

Fig 21 shows both the training and validation accuracy of the DenseNet169 model. Fig 22 shows both the training and validation loss of the DenseNet169 model. The training and validation accuracy increased and reached 100%. The training and validation loss also decreased with the increased number of iterations.

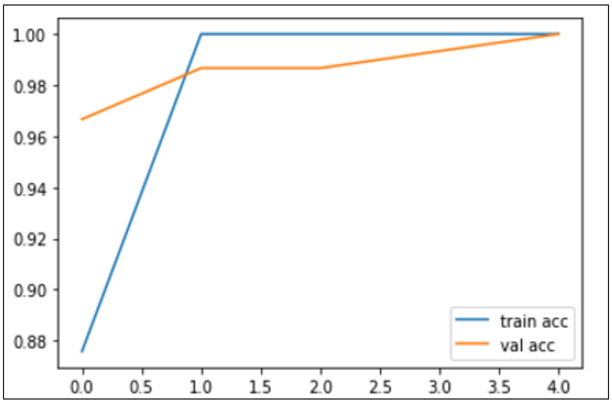

Fig 21: Validation and training accuracy graph of the DenseNet169 model

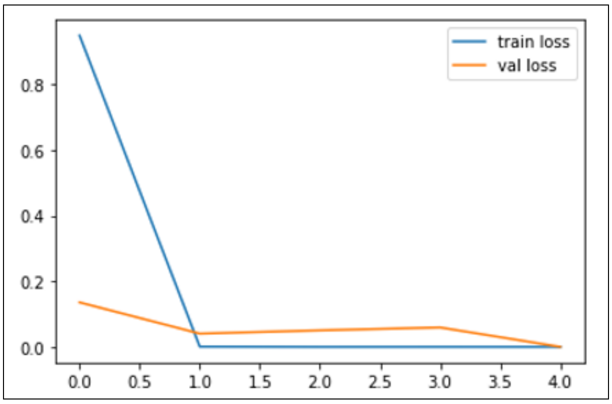

Fig 22: Validation and training loss graph of the DenseNet169 model

Table 1: Analysis of Transfer learning models

	VGG16	VGG19	MobileNetV2	DenseNet121	DenseNet169
Training accuracy	100%	100%	100%	100%	100%
Validation accuracy	100%	93.67%	100%	98%	100%
Training loss	0.0056	0.0064	0.0000	0.0000	0.0000
Validation loss	0.0529	0.1352	0.0000	0.0440	0.0000

Based on the results obtained from each model, the Teachable Machine deep learning model offered the best outcome with 100% accuracy and without any loss. However, to achieve that the model has to have several layers of convolutional layers and max pooling layers and the model was trained with a reasonable amount of data. So, it could be an inefficient model as it consists of several layers compared to other models and as well as considering a huge amount of dataset.

The proposed own model was able to attain a training accuracy of 95.55% and a validation accuracy of 94.10%. Likewise, the same model was able to obtain a training loss of 0.1569 and a validation loss of 0.2387.

Table 1 shows the outcomes of transfer learning models. VGG16, MobileNetV2 and DenseNet169 achieved an accuracy of 100% in both training and validation. However, MobileNetV2 and DenseNet169 outperformed VGG16 considering the validation loss.

Considering the performance obtained in all carried methods, MobileNetV2 and DenseNet169 obtained the best result for the particular newly created face dataset.

5. Conclusion

The necessity to have a face recognition system to improve security in all aspects for individuals or organizations is immensely important today. The traditional methods for recognizing faces have been now superseded with CNN-based deep learning methods. In this work, a comparative study of different CNN-based models and techniques has been evaluated. First, the face dataset was created by capturing the images from five randomly chosen individuals which consist of 1500 images. Once the dataset was created, initially it was trained and tested using Google Teachable Machine. The learning rate, batch size and epochs were changed in different combinations. Then a proposed own model has been used. After that five-transfer learning pre-trained models such as VGG16, VGG19, MobileNetV2, DenseNet121 and DenseNet169 were used for classification. MobileNetV2 and DenseNet169 outperformed all other methods.

In the future, this study can be enhanced by recognizing faces in real-time as it is done with some still images only and the dataset can be expanded as well. Further, the proposed model can be compared with other existing face recognition methods.

6. References

1. Onyema EM, Shukla PK, Dalal S, Mathur MN, Zakariah M, Tiwari B. Enhancement of Patient Facial Recognition through Deep Learning Algorithm: ConvNet, Journal of Healthcare Engineering, 2021, p8.
2. Sharma N, Sharma R, Jindal N. Machine Learning and Deep Learning Applications: A Vision, Global Transitions Proceedings. 2021; 2:24-28.
3. Pandey IR, Raj M, Sah KK, Mathew T, Padmini MS. Face Recognition Using Machine Learning, International Research Journal of Engineering and Technology (IRJET). 2019; 6(4):3772-3776.
4. Mandal M. Analytics Vidhya. [Online], May 1, 2021. Available: https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/. [Accessed July 23, 2021].
5. Wang ZJ, Turko R, Shaikh O, Das N, Hohman F, Kahng M, *et al*. Chau, CNN Explainer: Learning Convolutional Neural Networks with Interactive Visualization, IEEE Transactions on Visualization and Computer Graphics. 2021; 27(2).
6. Gurucharan M. upGrad, 28 July 2022. [Online]. Available: https://www.upgrad.com/blog/basic-cnn-architecture/.
7. Yaman MA, Subasi A, Rattay F. Comparison of Random Subspace and Voting Ensemble Machine Learning Methods for Face Recognition, symmetry, 2018; 10:p19.
8. Salman FM, Abu-Naser SS. Classification of Real and Fake Human Faces Using Deep Learning, International Journal of Academic Engineering Research (IJAER). 2022; 6(3):1-14.
9. Al-Humaidan NA, Prince M. A Classification of Arab Ethnicity Based on Face Image Using Deep Learning Approach, IEEE Multidisciplinary, Rapid Review, And

Open Access Journal. 2021; 9:50755-50766.
10. Akter T, Ali MH, Khan MI, Satu MS, Uddin MJ, Alyami SA, *et al*. Improved Transfer-Learning-Based Facial Recognition Framework to Detect Autistic Children at an Early Stage, Brain Sciences, 2021.
11. "Teachable Machine," [Online]. Available: https://teachablemachine.withgoogle.com.
12. Dwivedi R. Analytics India Magazine Pvt Ltd, August 29, 2020. [Online]. Available: https://analyticsindiamag.com/tutorial-on-keras-callbacks-modelcheckpoint-and-earlystopping-in-deep-learning/.
13. "Keras," Keras, [Online]. Available: https://keras.io/api/applications/vgg/#vgg16-function.
14. "Keras," Keras, [Online]. Available: https://keras.io/api/applications/vgg/#vgg19-function.
15. "Keras," Keras, [Online]. Available: https://keras.io/api/applications/mobilenet/#mobilenetv2-function.
16. "Keras," Keras, [Online]. Available: https://keras.io/api/applications/densenet/#densenet121-function.
17. "Keras," Keras, [Online]. Available: https://keras.io/api/applications/densenet/#densenet169-function.
18. Teoh K, Ismail R, Naziri S, Hussin R, Isa M, Basir M. Face Recognition and Identification using Deep Learning Approach, Journal of Physics: Conference Series, 2021.
19. Shamrat FJM, Billah MM, Jubair MA, Chakraborty S, Alauddin M, Ranjan R. A Deep Learning Approach for Face Detection using Max Pooling, in 5th International Conference on Trends in Electronics and Informatics (ICOEI 2021), Tirunelveli, India, 2021.