Comparative Analysis of Transfer Learning CNN for Face Recognition

Janvi Nandre
IT, International Institute of Information Technology, Pune

Abstract - Face recognition (FR) is described as the technique of identifying persons using face visuals. This technology is widely used in biometrics, surveillance technologies, security intelligence, law enforcement, real-time attendance systems, smart cards, etc. The facial recognition system is constructed in two stages. The first stage is a procedure that picks up or extracts facial features, and the second step is pattern classification. Deep learning, particularly the convolutional neural network (CNN), has recently made great contributions in Face Recognition technology.Deep learning employs numerous processing layers to develop data representations with varying degrees of feature extraction. Since the achievements of DeepFace, DeepID in 2014, this developing technology has altered the study landscape of facial recognition (FR).Deep learning has greatly improved state-of-the-art performance and promoted successful real-world applications thanks to its hierarchical network for stitching together pixels into stable face representation. The performance of several of the most prominent CNN architectures for face recognition is investigated in this research. For our study, we used transfer learning to implement pre-trained CNN models such as VGG16, ResNet-50, and MobileNet for face recognition. Training and validation accuracy and loss were leveraged as criteria to improve the CNN algorithm's performance. The models were trained and evaluated on our local dataset, and all models achieved a classification accuracy of 100 %. Face recognition was implemented using both a static and a video-based approach. Our models can be used in real-time attendance and surveillance systems.

The proposed method's Python code will be made accessible at
https://github.com/JanviNandre/Face-recognition-CNN-transfer-learning

Keywords - face recognition; machine learning; neural networks; convolution neural network (CNN), transfer learning; deep learning; VGG16, ResNet-50, MobileNet

1. Introduction

The market for facial recognition technology is rapidly expanding as a result of breakthroughs in AI, ML, and deep learning technologies. A system that recognizes people based on their faces is known as facial recognition. Face recognition requires simply a digital imaging device to generate and collect the photos and data needed to develop and record the biometric facial pattern of the subject to be recognized. Biometric face recognition, unlike other methods of authentication such as password-based, email verification, or fingerprint identification, leverages unique mathematical and dynamic patterns, making it one of the safest and most successful. There are several methods for implementing FR. One significant innovation is the use of CNN for deep learning. There are various techniques of utilise CNN. The first technique is to learn the model from the ground up. In this situation, the pre-trained model's architecture is employed and trained using the dataset. In circumstances where the dataset is huge, the second option is to use transfer learning with features from pre-trained CNN.Finally, CNN can be applied via transfer learning by preserving the convolutional base in its original form and then feeding the classifier with its outputs. When the dataset is short or the problem to be classified is similar, the pre-trained model is employed as a fixed feature extraction strategy. In our study, we used transfer learning to create a face recognition system employing three different CNN architectures.

2. Literature Overview

Previously, the researcher developed many algorithms and methodologies for facial recognition. These are covered in the following section: Masooli et al. [1] proposed a training method for fine-tuning a SotA SeNet-50 architecture to extract resolution-robust deep features. The SeNet-50 model was trained on photos from the VGGFace2 dataset with a randomly chosen resolution and a CL Teacher-student strategy from literature. They evaluated the model on IJB-Face, IJB-C, QMUL-SurvFace, TinyFace, and SCface datasets and obtained good cross-resolution recognition accuracy results. Sanchez et al. [2] proposed a system for real-time facial identification that requires moderate hardware and a combination of deep learning algorithms like FaceNet and classic classifiers like SVM, KNN, and RF to function in an unconstrained environment. The model detected faces using YOLO-Face and used preprocessing techniques such as Bicubic interpolation for picture resampling, the L2 method for normalization, and color adjustments. FaceNet with the softmax+ cross-entropy loss function was used for the recognition stage. Face detection is performed on the FBBD, WIDER FACE, and Honda/UCSD CelebA data sets and is evaluated based on accuracy, precision, and recall rate, whereas face recognition is performed on the LFW and YTF data sets, with the YTF dataset achieving 99.1 percent recognition accuracy at a real-time speed of 24 FPS. Almday et al. [3] conducted a study review of FR performance using pre-trained CNN (AlexNet and ResNet-50 models) for feature extraction, followed by support vector machine and then employed transfer learning with pre-trained CNN (AlexNet model) for both feature extraction and classification. The evaluation was performed on the ORL, Georgia Tech Face, GTAV, FLFW, FEI, LFW, and YTF datasets. ResNet with SVM used the least amount of time to train, however, transfer learning alexnet consistently performed well.

Kumar et al. [4] developed an image recognition ANN with OpenCV and Euclidean distance, as well as an android-based navigation system with ultrasonic sensors, to address the issue of smart navigation for the visually impaired people. The model had a 95% accuracy rate for obstacle detection and a 90% accuracy rate for face recognition. Razavian et al.[5] demonstrated that deep-Net features can serve as the prime candidate in vision tasks. They retrieved features from the OverFeat network and used them as a general representation for various object classification tasks, scene recognition, and attribute description. For recognition of a variety of vision tasks, the feature representation of shape 4096x1 was used in SVM classifiers or simple linear classifiers. They also claimed that the features retrieved using OverFeat and trained on the ImageNet database can be used for a variety of visual identification tasks. Prakash, R., et al. [5] proposed an automatic facial recognition system based on a Convolutional Neural Network (CNN) and a transfer learning methodology. The CNN with weights learned from the VGG-16 pre-trained model. For classification, the collected features are passed into the Fully connected layer with softmax activation. To evaluate the performance of the proposed technique, two publicly available databases of face images–Yale and AT&T–are employed. AT&T dataset photos get a 100% face recognition accuracy, while Yale dataset photos get a 96.5 % face accuracy. The results reveal that face recognition using CNN with transfer learning outperforms the PCA technique in terms of classification accuracy. Ding and Tao [6] suggested a broad method based on convolutional neural networks (CNN) to tackle the challenges of video-based face recognition (VFR). CNN learns concealed highlights by utilizing prepared content that includes misleadingly veiled data and still images. They presented a trunk-branch ensemble CNN model (TBE-CNN) to improve CNN highlights for detecting differences and obstacles. TBE-CNN extracts data from face images and zones selected around facial segments. They proposed a better triplet misfortune capability to amplify the impact of discriminative representations learned by TBE-CNN. TBE-CNN was evaluated using three different video face datasets: YouTube, PaSC, and COX Face. Bah et al. [7] research's introduced a new method for improving the accuracy of a face recognition system by combining the Local Binary Pattern (LBP) algorithm with several advanced image processing strategies such as Contrast Adjustment, Bilateral Filter, Histogram Equalization, and Image Blending with 0.5 alpha value and 181 x 181 pixels values. The proposed method outperformed other handcrafted FR algorithms in terms of face detection and accuracy. However, the technique fails to address the issue of occlusion and mask faces. Umrani et al. [8] thoroughly analyzed numerous feature-based automatic facial recognition systems and the aspects that influence them, such as stance, variation, and illumination. Face recognition approaches in the forensic realm have been discussed. The research has addressed future dealings for face recognition technology and provided a comprehensive study of numerous common datasets. Yassin et al.[9] investigated some well-known approaches for FR each approach and provided a taxonomy of the strategies as well as a comparison study in terms of complexity, robustness, accuracy, and discrimination. They also examined a number of prominent datasets used for FR. Nuredin Ali [10] used transfer learning on VGGFace to recognize faces with dark skin, primarily Ethiopian faces. On local datasets, the accuracy was greater than 95%.

3. Preliminaries :

3.1 Convolutional Neural Networks (CNN)

Convolutional neural networks are a type of neural network that is commonly used to solve image processing tasks. You've definitely seen them in action anywhere a computer identifies things in an image, but convolutional neural networks can also be used in natural language processing applications. One of the key reasons they're so significant in deep learning and artificial intelligence nowadays is that they're useful in these fast-growing domains.

An input layer, hidden layers, and an output layer are part of a standard neural network.

The input layer accepts different forms of input, while the hidden layers conduct calculations on these inputs. The results of the calculations and extractions are subsequently delivered by the output layer. Each of these layers has neurons that are linked to neurons in the layer before it, and each neuron has its own weight. This means you're not making any assumptions about the data entering the network. Normally, this is great, but not when working with images or languages.

Convolutional neural networks operate in a unique way because they consider data as spatial. Instead of being coupled to every neuron in the previous layer, neurons in the layer are only connected to neurons nearby and all have the same weight. The network maintains the spatial aspect of the data collection due to the simplification of the links. The filtering process that occurs in this type of network is referred to as convolutional. Consider this: an image is complex. A convolutional neural network simplifies it so that it can be processed and understood more easily. A convolutional neural network, like a regular neural network, has many layers. There are a couple of layers that make it unique, the convolutional layer and the pooling layer. However, like other neural networks, it will also have a ReLU or a rectified linear unit layer and a fully connected layer. It is distinguished by two layers: the convolutional layer and the pooling layer. However, it will have a ReLU or rectified linear unit layer and a fully connected layer, just as other neural networks. The ReLU layer serves as an activation function, ensuring non-linearity as data flows through the network's layers. Without it, the data given into each layer would lose the dimensionality that we desire. Meanwhile, the fully connected layer allows you to classify your data collection. The most significant layer is the convolutional layer. It operates by applying a filter on an array of picture pixels. This results in what is known as a convolved feature map. The pooling layer comes next. This decreases or downsamples the sample size of a specific feature map. This also speeds up processing because it minimizes the number of parameters that the network must process. The result is a pooled feature map. There are two methods for accomplishing this: max pooling, which takes the maximum input of a specific convolved feature, and average pooling, which just takes the average. These processes are equivalent to feature extraction, in which the network constructs a picture of the visual input based on its own mathematical principles. To do categorization, you must first enter the fully linked layer. You'll need to flatten everything out to accomplish this.

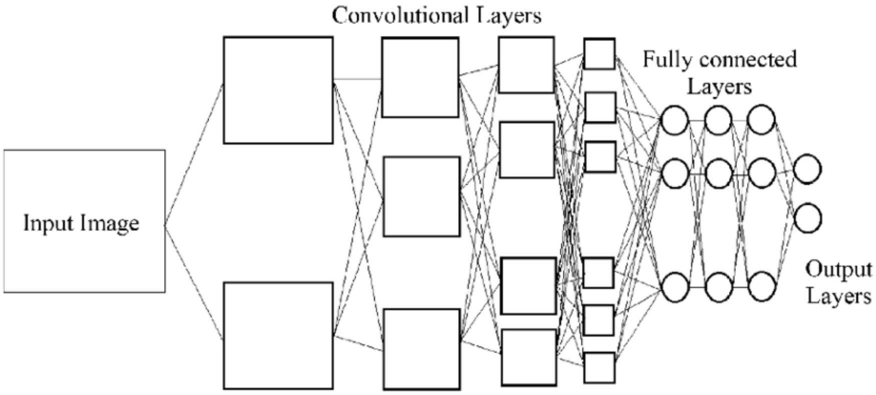


Figure 1 CNN basic architecture

3.2 Transfer Learning

Transfer learning is a branch of AI-ML that seeks to apply knowledge obtained from one work (the source task) to a different but similar activity (target task).

Transfer learning entails attempting to apply what has been learned in one task to improve generalization in another. We transfer the weights learned by a network at "task X" to a different "task Y." To obtain transfer learning, three conditions must be met: the development of an open-source third-party pre-trained model, Repurposing of the model, and fine-tuning for the Scenario.

3.3 VGG16

VGG16 is a convolutional neural network model proposed by the University of Oxford. In ImageNet, a dataset of over 14 million images classified into 1000 classes, the model achieves 92.7 percent top-5 test accuracy. It was one of the well-known models submitted to the ILSVRC-2014. It outperforms AlexNet by substituting huge kernel-sized filters (11 and 5, respectively, in the first and second convolutional layers) with numerous 33 kernel-sized filters one after the other.

3.3 ResNet50

ResNet50 is a convolutional neural network with 50 layers of depth. Microsoft built and trained it in 2015, and the model performance results may be found in their publication titled Deep Residual Learning for Image Recognition. This model has also been trained on over 1 million photos from the ImageNet collection. It, like the VGG-16, can categorize up to 1000 objects and was trained on 224x224 pixel-coloured images.

3.4 MobileNet

The MobileNet model is TensorFlow's first mobile computer vision model, and it is intended for usage in mobile applications. Depthwise separable convolutions are used by MobileNet. When compared to the network with ordinary convolutions of the same depth in the nets, it greatly reduces the number of parameters. As a result, lightweight deep neural networks are created.

3.5 Face recognition

Facial recognition is a system that can identify people based on their faces. It is based on complex mathematical AI and machine learning algorithms that gather, record, and evaluate face traits in order to match them with photos of people and, in some cases, data about them in a database. Facial recognition is a subset of biometrics, which also includes, palm printing, fingerprint scanning, eye scanning, and signature identification.

4. Methodology and Experiments

The primary purpose of this research was to investigate FR performance using convolutional neural networks. First, we used the Keras library and TensorFlow backend to import the pre-trained models and weights. The pre-trained model was then frozen and rebuilt using our new customized fully connected layer. We used the Dense() function to add new Hidden layers of neurons, as well as the Flatten() function to add a layer to flatten our image input, making it 1D, and grouped our layers into a single object. The model was then built using the Adam optimizer, accuracy as metrics, and a loss function. During the preprocessing stage, the dataset images were augmented using the ImageDataGenerator() function from the Keras package. The model was then trained on our local dataset with an input image size of 224 by 224 pixels. The epoch was set to 5 for all models. Finally, the models were tested using images from the test set that were produced at random.

5. Results

In this research, we used two dataset approaches. The first dataset contains 118 celebrity pictures with different scales, illumination, pose, and resolution. There are 93 photos in the training set and 25 in the testing set. We took a different strategy for the second dataset, implementing a program that takes samples from a webcam using OpenCV. Then, using the HAAR classifier, it finds faces and returns cropped images containing faces. Using this method, we may collect an unlimited number of photos for a specific class, each with a different stance and lighting.

Figure 2 Our Local Dataset Images

On our local datasets, all of the models obtained 100 percent recognition accuracy. This high accuracy could be attributed to the quantity of our dataset. The accuracy and loss figures for each model are listed below. MobileNet was the quickest to train and provide accurate results, but ResNet-50 took longer to train because to its complexity.

The training and testing accuracy and loss for our models is as follows :

5.1 VGG16

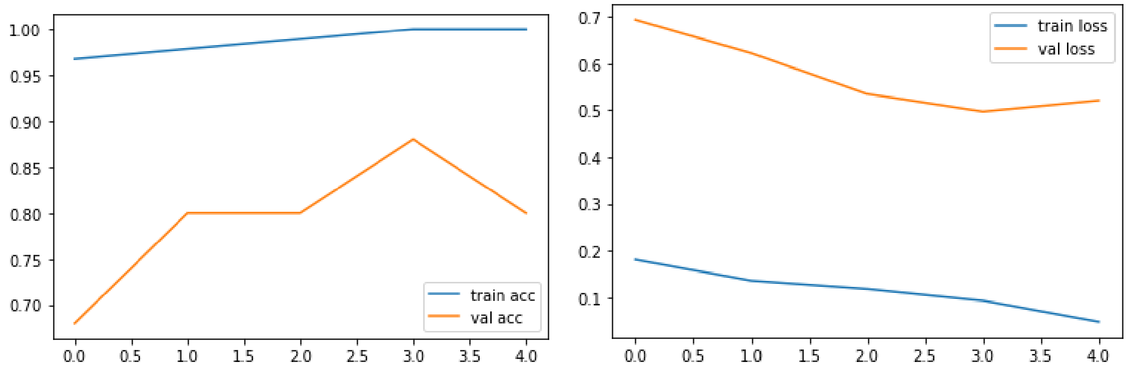


Figure 3 VGG16 Accuracyand Loss

5.2 ResNet-50

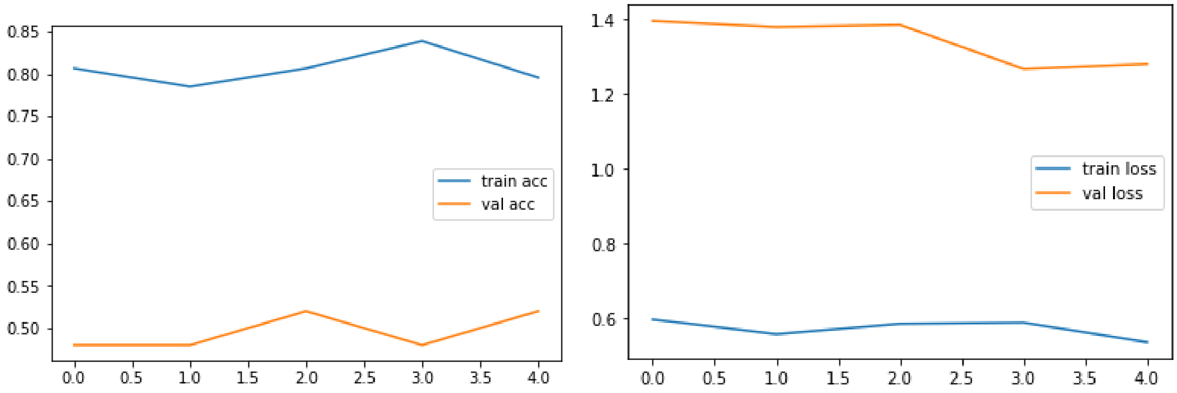


Figure 4 ResNet-50 Accuracy and Loss

5.3 MobileNet

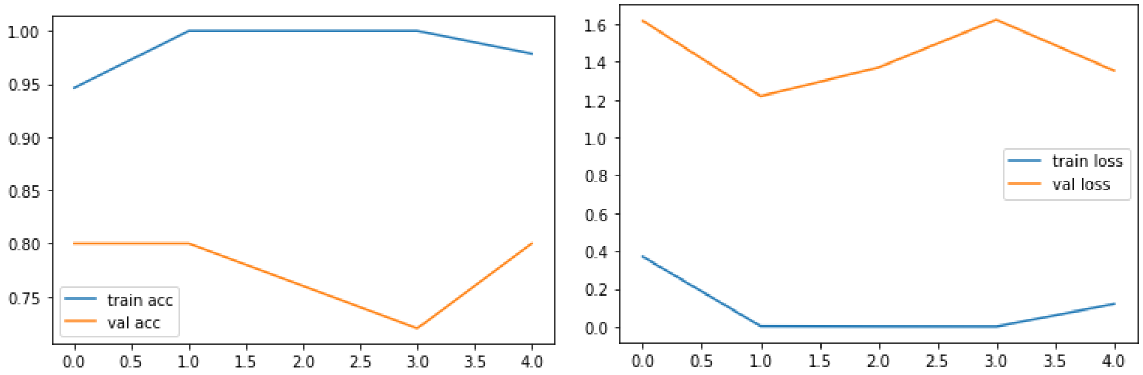
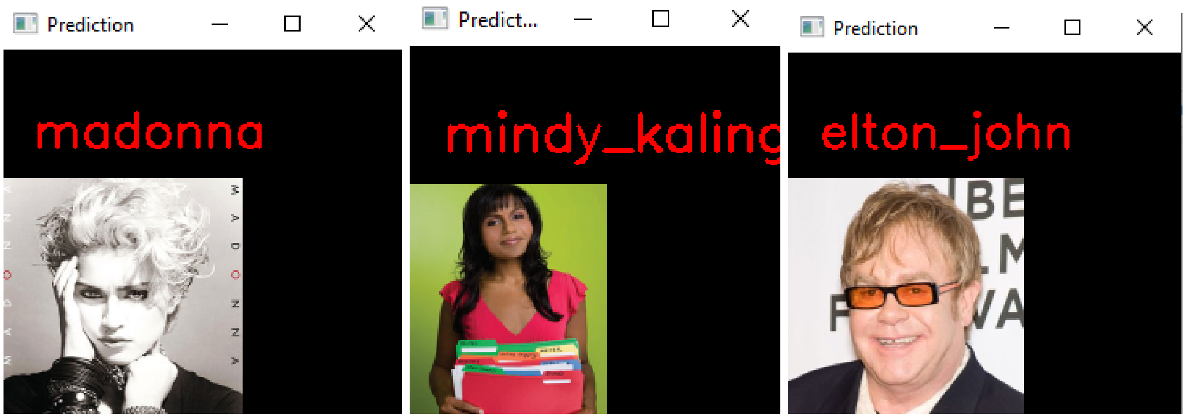


Figure 5 MobileNet Accuracy and Loss

5.4 Classification results :



madonna

mindy_kaling

elton_john

Figure 6 Classified Images using our models

Face recognition was performed using a random sample from the test set, and our models were then employed as classifiers. The detected image is labelled with the person's name on the window, and this technique may be utilised for both image and video face recognition. We have also built a separate function that can perform real-time recognition. All 3 models gave 100% recognition accuracy.

5. Conclusion

In this study, we used transfer learning to create a face recognition system. We learned about the CNN architecture and its pre-trained models, such as VGG16, ResNet-50,, and MobileNet, which are trained on the ImageNet dataset. We compared all three models in terms of accuracy, loss, training durati,on, and recognition loss. Our models can be used in real-time attendance systems as well as static image face recognition systems. Our future goal is to develop a real-time face recognition system for use in crime detection and forensics.

References

1. Massoli, F. V., Amato, G., & Falchi, F. (2020). Cross-resolution learning for face recognition. *Image and Vision Computing*, *99*, 103927.
2. Sanchez-Moreno, A. S., Olivares-Mercado, J., Hernandez-Suarez, A., Toscano-Medina, K., Sanchez-Perez, G., & Benitez-Garcia, G. (2021). Efficient Face Recognition System for Operating in Unconstrained Environments. *Journal of Imaging*, *7*(9), 161.
3. Almabdy, S., & Elrefaei, L. (2019). Deep convolutional neural network-based approaches for face recognition. *Applied Sciences*, *9*(20), 4397.
4. Kumar, P. M., Gandhi, U., Varatharajan, R., Manogaran, G., Jidhesh, R., & Vadivel, T. (2019). Intelligent face recognition and navigation system using neural learning for smart security in Internet of Things. Cluster Computing, 22(4), 7733-7744.
5. Prakash, R. M., Thenmoezhi, N., & Gayathri, M. (2019, November). Face recognition with convolutional neural network and transfer learning. In *2019 International Conference on Smart Systems and Inventive Technology (ICSSIT)* (pp. 861-864). IEEE.
6. Ding, C., & Tao, D. (2017). Trunk-branch ensemble convolutional neural networks for video-based face recognition. *IEEE transactions on pattern analysis and machine intelligence*, *40*(4), 1002-1014.
7. Bah, S. M., & Ming, F. (2020). An improved face recognition algorithm and its application in attendance management system. *Array*, *5*, 100014.
8. Jayaraman, U., Gupta, P., Gupta, S., Arora, G., & Tiwari, K. (2020). Recent development in face recognition. *Neurocomputing*, *408*, 231-245.

9. Kortli, Y., Jridi, M., Al Falou, A., & Atri, M. (2020). Face recognition systems: A survey. *Sensors*, *20*(2), 342.
10. Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and StefanCarlsson. ”CNN features off-the-shelf: an astounding baseline forrecognition”. CoRR, abs/1403.6382, 2014

Janvi Nandre – Third Year IT undergraduate, I2IT Pune, nandrejanvi@gmail.com