Face Recognition using Deep Learning

By Ruchir Sade
Cleveland State University

Abstract

Facial Recognition is the action of identifying a face in a visual input. To a human, this may seem trivial but when we need to analyze a huge amount of data, a computer has the edge, if it is equipped with an efficient machine learning algorithm. Deep learning is a type of neural network which uses multiple hidden layers used to identify hybrid features in data. The objective of this paper is to implement an efficient deep learning algorithm which can identify faces accurately using a training dataset to train the algorithm until an agreeable efficiency is reached and testing the performance on unknown images or video feed frames. It is also compared to the performance of other machine learning algorithms to determine the more efficient algorithm.

Summary

Deep learning algorithms are an evolved version of machine learning algorithms which are based on the concept of neural networks. Neural Networks depicts how the human brain architecture is constructed, with the neurons branching out. This architecture when mimicked by a computer has been termed Artificial Neural Networks. The use of multiple hidden layers helps in better feature extraction and in turn increases the potency of the algorithm. Deep learning has played a major role in Computer Vision in the last decade. Major advances in self driving cars have been made in the last few years and Computer Vision plays a large part in helping the control systems to act by recognizing roads, lanes, traffic lights and numerous other features to make the right decisions instantaneously.

When we take an image or video frame as input, the size of the input data will be large and classifying the data immediately would result in an algorithm that would not be close to perfection. Human faces cannot be generalized as a one-dimensional array to recognize them accurately enough. If we can teach the algorithm to learn about the features that constitute a human face, the machine learning algorithm can identify faces much more accurately. The use of a deep learning algorithm will detect features of the face and will improve the performance of the algorithm using the features in the subsequent layers until the final hybrid features are used to recognize the visual input and uses a bounding box to show the identified face.

The applications of facial recognition are vast and can range from using CCTV cameras to identify criminals, finding missing children, help forensic investigations and even helping the visually impaired. Most modern-day smart phones allow the user to access their phone by simply facing it. These features have been further improved to recognize whether the users have their eyes open or not to restrict access without the owner's conscious action. Facial Recognition is a highly useful tool with much potential and machine learning allows us to wield it. A highly trained neural network to recognize faces accurately would be just as accurate if not more and certainly much faster when dealing with a large amount of data when compared to the human mind.

Implementation

The model I chose to implement face recognition is a Deep Convolutional Neural Network. The Convolutional Neural Network architecture consists of a convolutional layer followed by pooling layers present after each convolutional layer. The convolutional layers act as the building blocks of the Convolutional Neural Network. They apply a filter to an input to create a feature map which results in the acquisition of higher order features. The convolution process is done by using a filter of a particular size which consists of weights which are convolved with their input to output the modified matrix with the depth determined by the filter used. The model updates these weights in each epoch to ascertain the weights which return the most accurate results. The kernel shifts across the image convolving the filter with the input, i.e., multiplying the matrices and storing the value in a new matrix which will have different dimensions due to the convolution which can padded to return the size of the original matrix if needed.

The pooling layer is used to find the distinct pixels in the output of the convolution layer. The pooling technique used in this study was max pooling which returns the max pixel value in the stride. A stride of 2x2 is used, which takes a matrix and drifts 2x2 pixels at a time and returns the maximum pixel value in each 2x2 grid. This results in the size of the output being half the input.

The Convolutional layer structure used in this study constitutes two convolutional layers each of which are followed by a pooling layer after which the image is flattened and passed through a dense layer of 64 nodes and finally another fully connected layer with softmax activation.

The model is trained in batches as the error values of a large number of images do not have to be stored. This is an important tool to train more complex algorithms as they are very demanding in processing power.

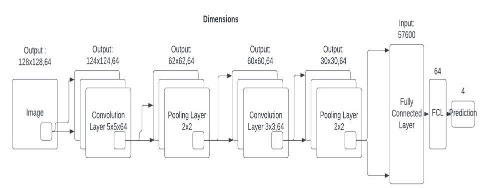
The dataset used is random faces of humans along with the pictures of three celebrities which are divided into training and testing sets with a validation split of 0.2, i.e., 20 percent of images are used for testing.

The filter used in the first convolutional layer was a 5x5 filter with a depth of 64. The filter used in the second convolutional layer was a 3x3 filter with a depth of 64 both using the relu activation function.

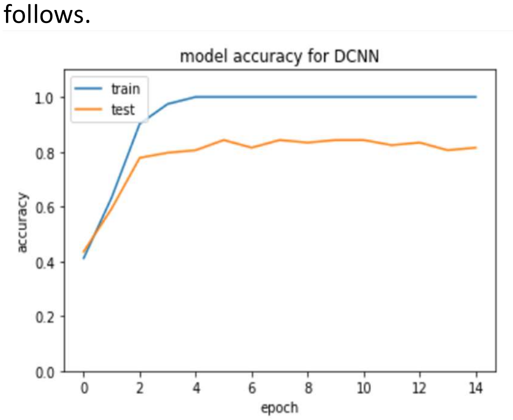
The final fully connected layers are used to return an array of length 4 which categorizes the images to their respective group using the softmax activation function.

To classify the images loss function was taken as sparse categorical cross entropy which is used in classification algorithms

Model Structure

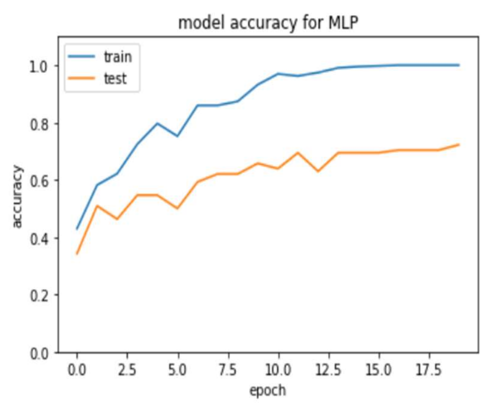


The model designed was able to recognize faces with a maximum accuracy of 84.25 percent and was able to reach it below 10 epochs. The accuracy graph documented, comparing training and testing accuracies is as follows.



0.8425925970077515

In order to compare performance and the ability of Deep Convolutional Neural Networks a Multi Layered Perceptron was implemented to compare the performance. This MLP consisted of three layers. The first layer after flattening had 128 hidden nodes, the second layer consisted of 64 hidden nodes and the final layer determined the class. This model was trained on the same dataset and the performance was documented.



0.7222222089767456

As seen from the results above it can be determined that on a dataset which consists of only human faces, a deep convolutional neural network performs far better than a multi layered perceptron.

Prospects

The performance of my model can be improved by increasing the number of images in the dataset. The current dataset consisted of a sum of 536 images which in my research showed was less than optimal. The accuracy of the model can reach a much higher level if it is trained and tested over a larger dataset.

The use of a more varied dataset can allow this model to perform even better on different orientation of human faces. Nearly all the images in the dataset are pictures of the face which makes the MLP perform only slightly worse than the DCNN.

My study may have started out as comparing DCNNs and MLPs but the research it led to has changed my focus to understand where DCNNs exemplify their superiority. The concept of Face detection and recognition was a field I came across which made me understand the true potential of DCNNs. The detection of faces in parts of images in different orientations is where the concept of DCNNs can be used to its full potential. The convolutional neural network recognizes features at each layer which get more detailed as we go deeper into the network letting it recognize more complex features like noses, eyes at later layers starting from horizontal and vertical lines.

Conclusion

In conclusion, this study gives insight into the potential on Deep Convolutional Networks and how it can be used to implement Face Recognition to differentiate similar looking features of a human face. The ability of a DCNN to find higher order features of images to accurately predict their respective types is demonstrated through this study.

References

https://machinelearningmastery.com/how-to-perform-face-detection-with-classical-and-deep-learning-methods-in-python-with-keras/

https://www.youtube.com/