Signature Verification System using VGG16 By Transfer Learning

MD. Imtiaz Habib
Computer Science and Engineering
American International University
Bangladesh
Dhaka, Bangladesh
imtiazhabib17@gmail.com

Kazi Tanvir
Computer Science and Engineering
American International University
Bangladesh
Dhaka, Bangladesh
kazitanvir.bangladesh@gmail.com

MD.Mehedi Hasan Zamil
Computer Science and Engineering
American International University
Bangladesh
Dhaka, Bangladesh
mahimehedi4@gmail.com

MD Raiyan Khan
Computer Science and Engineering
American International University
Bangladesh
Dhaka, Bangladesh
raiyankhan1982@gmail.com

Tajul Islam
Computer Science and Engineering
American International University
Bangladesh
Dhaka, Bangladesh
Islamtaz30@gmail.com

***Abstract*— With the fast growth of computer science and information technology, the demand for a person's identification in many organizations, institutions, banks or online business etc. is increasing day by day. Similarly, signature verification and recognition technologies are commonly used to identify forgery and fraud. For many decades, signature verification have been crucial biometric features. We used Transfer Learning to implement VGG16 in this paper. It has also been looked at to verify and identify offline photos of signatures. The ICDAR 2011 Signature Dataset was used as a standard for the research, which included English signatures from various people. The experimental results show that the VGG16 design can achieve good accuracy.**

Keywords—Signature, VGG16, Dataset, Model, Accuracy

I. Introduction

Nowadays signature verification is an important biometric technology. Signature verification is a type of software that compares signatures and checks for authenticity. This saves time and energy and helps to prevent human error during the signature process and lowers chances of fraud in the process of authentication. In this research we have used ICDAR 2011 Signature Dataset. Dataset contains genuine and fraud signature samples. There is two main directory (test and train). It has 1649 image for training (128 Different Classes, 69 Genuine, 69 Forged) and 500 image for testing (42 Different Classes, 21 Genuine, 21 Forged). Each unique signature has 10 different version of it. We are using deep learning model for signature verification task. We are trying to improve the accuracy of other existing models. We are using VGG16 architecture by transfer learning approach. VGG16 is a pre trained CNN (Convolutional Neural Network) model. We have achieved 97.14% accuracy model.

II. Dataset

A. Dataset source

CDAR 2011 Signature Dataset.

B. Contents of the dataset

The dataset contains two main directory (test and train). It has 1649 image for training (128 Different Classes, 69 Genuine, 69 Forged) and 500 image for testing (42 Different Classes, 21 Genuine, 21 Forged).

III. Methodology

We are using VGG16 architecture by transfer learning approach. VGG16 is a pre trained CNN (Convolutional Neural Network) model.

A. Model Diagram

We have collected a dataset in order to train our model. We have splitted the dataset into Test and Train. After that we have preprocessed our data. In this step we have resized the images to 224 x 224. Because as we are using VGG16, the image size is fixed to 224 x 224. After preprocessing our data, we have saved image data in Numpy array. Then we have imported the base model VGG16. We have implemented this architecture by using transfer learning approach. It has 16 convolutional layers. Here we are freezing first 5 layers of VGG16. As it is already a pre-trained model, we freeze the first five layers so that it does not change its weights. After training our model we have got our desired output with good accuracy.

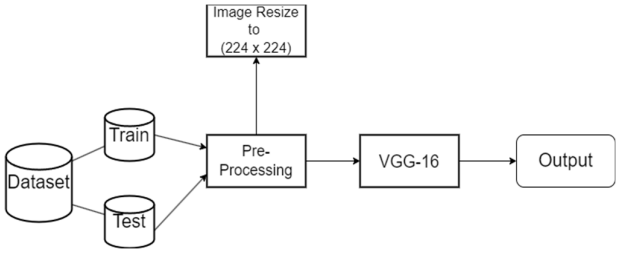

Fig: 1 ~ Model Diagram

B. *VGG16 Architecture*

- It has 16 convolutional layers.
- It has 3x3 convolutional layer at stride 1, padding 1 and ReLU activation function.
- It also has 2x2 max pooling at stride 2.

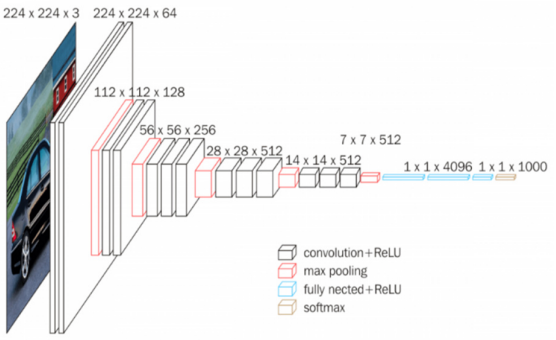

Fig: 2 ~ VGG16

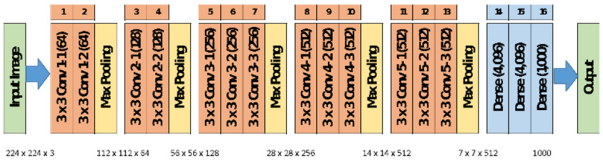

Fig: 3 ~ VGG16 Architecture

C. *Model Training*

We have trained our model using VGG16. As we know VGG16 is very power hungry architecture, it needs more time to train our model. For having less computational power, we have used Early Stop callback function. This callback allows us to specify the performance measure to monitor, the trigger, and once triggered, it will stop the training process. By using this callback function we got our result with 7 epochs.

D. *Some Mistakes*

As we have used Early Stop callback function, the training stopped after 7 epochs. It running for 100 epochs but we did not do it. So the result could be explored more.

IV. Result

The model had an accuracy of 97.14% with 7 epochs.

A. Validation Accuracy

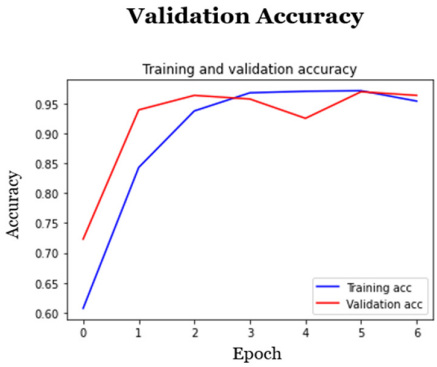

Fig: 4 ~ Validation Accuracy

Here we can see that the Validation Accuracy is more than 0.5.

B. Validation Loss

Fig: 5 ~ Validation Loss

Here we can see that the Validation Loss is less than 0.5.

Acknowledgment

Words cannot express my gratitude to my faculty for his invaluable support and feedback.

References

[1] Atefeh Foroozandeh, Ataollah Askari Hemmat and Hossein Rabbani, "Offline Handwritten Signature Verification and Recognition Based on Deep Transfer Learning" 2020 IEEE/ACS 17th International Conference on Computer Systems and Applications (AICCSA), 10 September 2020. Link: https://ieeexplore.ieee.org/document/9187481

[2] Wael Al-Ayoubi, "ICDAR 2011 Dataset" Qatar University, 18-21 September 2011. Link: https://www.kaggle.com/datasets/robinreni/signature-verification-dataset

[3] Aaron Courville, Ian Goodfellow, and Yoshua Bengio's Handbook, "Deep Learning", 2015. Link: https://www.deeplearningbook.org/