Offline Signature Verification Using Image Processing

Bushra Shaik [1*], *Jyothi Manohar* Katikireddy [2], *Vamsidhar* Kambham [3], *K* Sravani [4].

[1] Btech Student, Electronics and Communication Engineering, GRIET, Hyderabad, Telangana, India.
[2] Btech Student, Electronics and Communication Engineering, GRIET, Hyderabad, Telangana, India.
[3] Btech Student, Electronics and Communication Engineering, GRIET, Hyderabad, Telangana, India.
[4] Assistant Professor, Electronics and Communication Engineering, GRIET, Hyderabad, Telangana, India.

Abstarct: A person's signature is merely a handwritten sign that closely resembles his/her name, frequently stylized and distinctive, and that expresses the person's identity, intent, and consent. Two types of verifications are present. They are online signature verification and offline signature verification. Generally, Offline Signature verification is less efficient and slower process compare to online verification when come to the situation having larger number of documents and files to verify with in less time. Over the years, many researchers have developed so many methods for signature verifications to help the people or organizations to find whether the signature of a particular person is forged or genuine. To overcome this problems; In this paper we introduced a simple method to improve the verification of the signature in Image Processing using Convolution Neural Networks(CNN).

Signature Verification it is used to authenticate various kinds of documents, including cheques, draughts, certificates, approvals, letters, and other legal ones, such verification is crucial for preventing document forgery and falsification. Previously, to verify a signature, it was manually checked against copies of real signatures. This straightforward approach might not be sufficient given that forgery and signature fraud techniques are becoming more sophisticated as a result of improving technology.

Keywords—Offline Signature, Image Processing, Convolution Neural Networks, high accuracy, Speed and Robust

1. INTRODUCTION

This template, A signature is a mark or a person's name, frequently stylized and handwritten, that they use to signify their identification and sincerity of purpose. In various financial, legal, bureaucratic, academic, and other commercial settings, the handwritten signature of a person is widely accepted as a method of confirming the legitimacy of documents such as certificates, checks, draughts, letters, approvals, visas, passports, etc. It is essential in preventing the forgery and falsification of such documents. For example take during the competitive exams like government exams and central

[*] Corresponding Author: bushrajabeenshaik@gmail.com

government exams; In earlier, days when there is no software verifications many people made forgery signatures during the exam registrations and while attending interviews etc., With this many people lost their opportunities. For instance, a signature plays a crucial function in any contracts to identify the party of interest and to show purpose and informed consent.So, to overcome this kind of issues the signature verification is introduced. Several security applications make use of biometrics technology. Such methods are created to identify a person based on physical or behavioural characteristics. In the first instance, identification is based on measurements of biological characteristics like a person's fingerprint, face, iris, etc. In the latter instance, behavioural characteristics like voice and handwriting are at issue.The two main applications of biometric systems are identification and verification. In the first instance, a system user presents a biometric sample and asserts their identification. The verification system's job is to make sure the user is who they say they are. The goal of the identification case, where a user supplies a biometric sample, is to locate the user's biometric sample among all other users registered with the system.

The most important component of an individual's ID in an increasingly innovative environment is their signature. The number of fake cases is also increasing dramatically as the years pass. Thus, the use of a signature check framework is a request for a chance to improve the interaction between confirmers and provide secure ways for the approval of authoritative archives. To distinguish between genuine and fake marks, use the mark confirmation frameworks. Traditionally, a person would authenticate a sample signature by comparing and assessing the sample with copies of real signature specimens they had already collected or with the assistance of a witness.

.

2. OVERVIEW

The task of verifying signatures is extremely important and frequently involves challenges such as high variability a person's signature may change significantly with age, behaviour and environment similarities between signatures of different people, and similarity in signature duplication or forgery. Such obstacles to authentication can be overcome by two methods Online verification and offline verification. Offline signature verification is done by using image pixels dimensions, size etc., by scanning the signed documents.

3. IMAGE PROCESSING

The process of applying specified procedures to a photograph in order to either enhance it or extract some useful information from it is known as image processing. The demand for researchers with the ability to analyse and analyse image data has grown as computer systems have become faster and more powerful and as cameras and other imaging devices have proliferated in a variety of different areas of life. The large amounts of data that may be involved, the time-consuming and error-prone nature of manual processing, the high-resolution images that take up a lot of disc space or virtual memory, and the collections of numerous images that must be processed together can all make automated processing and analysis as a computer programme advantageous or even necessary.

3.1 CONVOLUTION NEURAL NETWORKS

A convolutional neural network (CNN) is composed of one or more convolutional layers (often with a subsampling step), while a typical MLP is followed by one or more fully connected layers. The architecture of a CNN is designed to take advantage of a 2D input's 2D structure, such as an voice or picture signal. Local connections and linked weights are used to accomplish this., then some sort of Hidden Input Output 7 Pooling yields translation-invariant features. In comparison to fully linked networks with the same number of hidden units, CNNs also benefit from being easier to train and having a much smaller set of parameters. Convolution neural networks also have the advantage of being simpler to train and having a smaller number of parameters than fully linked networks with the same number of hidden units.

3.2 DEEP LEARNING

Deep learning attempts to mimic the capabilities of the human brain, but is still far from being able to match them. As a result, algorithms may cluster data and generate extremely accurate predictions. In recent years, methods that are not dependent on manually crafted feature extractors have attracted increased interest. As an alternative, feature representations can be learned from unprocessed data (in the case of photos, pixels). An illustration of this are the Deep Learning models.The results of employing these features to distinguish between authentic signatures and forgeries were not revealed by Ribeiro et al., who utilised RBMs to create a representation for signatures. Instead, they simply provided a visual depiction of the learned weights. Previous efforts using private datasets to apply representation learning to the job did not report significant results.. Khalajzadeh employed CNNs to verify Persian signatures, however their testing only took into account chance forgeries.

Artificial neural networks, also known as deep learning neural networks, try to imitate how the human brain functions by employing data inputs, weights, and bias. Together, these components allow things in the data to be appropriately identified, categorised, and described. Deep neural networks are made up of several interconnected layers of nodes, each of which improves the categorization or prediction offered by the one in front of it. Calculations move through a network according to a process known as forward propagation. The layers of a deep neural network that are visible are the input and output layers. After the data has been processed in the input layer, the deep learning model performs the final prediction or classification in the output layer.

Backpropagation is an alternative method that evaluates prediction errors using techniques like gradient descent before iteratively returning through the layers to change the weights and biases of the function in an effort to train the model. Together, forward and back propagation allow a neural network to forecast the future and correct any errors. Over time, the algorithm's precision keeps improving.

3.3 RECURRENT NEURAL NETWORKS(RNN*)*

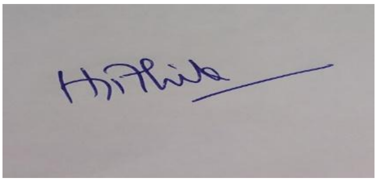
Because they use sequential or time series data, RNN are commonly used in applications for voice and natural language recognition. They are implemented into well-known programmes like Google Translate, Siri, and voice search. Recurrent neural networks (RNNs) learn from training data similarly to feedforward and convolutional neural networks (CNNs). They are distinct owing to their "memory," which enables them to change the present input and output by utilizing information from the past inputs. Unlike normal deep neural networks, which assume that inputs and outputs are independent of one another, recurrent neural networks' outputs are reliant on the previous sections of the sequence.

4. LITERATURE SURVEY

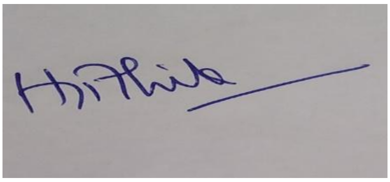
In depth study of 15 Research papers is done before proceeding with the project. Notable Research papers are mentioned below.[1] The authors propose a method that uses a shapematching algorithm to comparethe signature's form in comparison to another signature. It initially extracts the signature's attributes before applying a correspondence method to compare them to the reference signature. On a dataset of offline signatures, the suggested method is assessed and contrasted with other cutting-edge approaches.[2] This research suggests a deep learning-based method with two stages: forgery detection and signature identification. A deep convolutional neural network (CNN) is trained to identify real signatures in the first stage. The test signature is compared to the real signature in the second stage of training a CNN to recognise forgeries.[3] According to the writer of the signature, the authors suggested using a convolutional neural network (CNN) to extract the features of the signature and classify it into one of the pre-established classifications. The findings demonstrate that the suggested strategy performs better than the alternatives in terms of accuracy, precision, recall, and F1 score.[4] The suggested approach uses a deep neural network architecture to extract features from an offline signature image and then confirms its authenticity by contrasting it with a reference signature. The findings demonstrate that, in terms of accuracy, false acceptance rate, and false rejection rate, the suggested method performs better than the other methods. [6] Author paper here suggests an offline Hidden Markov Model (HMM) approach for signature recognition. The two stages of the suggested method are feature extraction and classification. The signature is segmented and its features, such as form and velocity, are extracted during the feature extraction stage. HMMs are used to model the signature features and identify the signature during the classification step. A 90% identification rate is achieved using the suggested strategy.[7] The author used a neural network methodology to create an offline signature recognition system. Two steps of feature extraction and classification make up this work. The signature is divided into segments for feature extraction, and attributes like height, width, and centre of gravity are taken out. The neural network is taught to identify the signature based on its attributes during the classification step. The findings indicate a 92.5% recognition rate.[8] The author of this article uses logistic regression and neural network techniques to provide a study on handwritten English character recognition. To train a logistic regression model for character recognition, the features in the logistic regression approach are retrieved as vertical and horizontal projections. A multilayer perceptron (MLP) and the backpropagation algorithm are employed in the neural network approach for training and recognition..[9] The author used a convolution neural network, it involves pre processing of signature images. This work consists of two stages feature extraction and classification. Inception V1 architecture had been used to extract features from the signature image. The author employs data augmentation techniques such as rotation and scaling to increase the size of the images. Softmax layer had been used to classify the signature images as genuine or forged. The result shows a recognition rate of 99.47%.[10] Here the author proposed three stages of the proposed method. They are pre processing, feature extraction and image generation. In the pre processing stage, the image gets segmented and normalized to remove any background noise. In feature extraction set of features had been extracted from an image which includes shape, texture and stroke-based features. In image generation, Generative Adversarial Network (GAN) had been used to generate multiple versions of the genuine signature image.[11] The author uses a 3`deep neural network which optimizes two tasks i. e. signature verification and feature learning. The network is trained using multitask loss function that combines the loss of two tasks.[12] The author used an interval symbolic representation of the signature image and fuzzy similarity measure. First, the signature image is converted into an interval symbolic representation where each pixel is represented by an interval of values. Later, a fuzzy similarity measure is applied to compare the intervals of two signature images and calculate the similarity score.[13] Here the author uses adaptive windowing techniques. The first step is preprocessing of an image which removes noise and enhances the contrast. Later it divides the image into overlapping blocks of fixed size and applies a set of statical features to each block to extract exact-specific characteristics. Adaptive windowing techniques involve shifting the block position and sizes within the signature image to generate multiple sets of the feature vector.

5. EXISTING WORK

In contrast to any physiological qualities of the individual signature, the handwritten signature is a behavioural biometric that is based on behaviour. The verification and because a person's signature varies with time, which results in irregularly increased errors, signature authentication may take a lengthy period. increased erroneous rejection rates result from inconsistent signatures for signers who did not do so consistently. Previously, the work was done using OpenCV technology, which has a greater error rate and cannot predict reliable results. Moreover, the output processing takes time and is not significantly faster. All of these flaws will be fixed with our proposed technology “TensorFlow” and the output will be quicker than before while also producing accurate results with a lower mistake rate.



Fig(1)Original Signature



Fig(2) Forged Signature

6. PROPOSED WORK

In this research, TensorFlow's Python programming language is used for verification.
CNN is necessary for obtaining the necessary features. to distinguish the key features that make the two class labels different. When a person regularly signs their name, their brain controls the nerve impulse without taking the particular into account as they sign. The person who forges someone else's signature, on the other hand, carefully evaluates every component of the original signature, which changes nerve impulses and causes some trembling in the signature. The characteristics of the signs that CNN collects and recognizes alter as a result of this difference.

6.1 Feature Extraction Results

Table 1: Training Accuracy

	optimizers SGD	**Optimizers RMSprop**	**Optimizers Adagrad**	**Optimizers Adam**
VGG16	0.8648	0.9645	0.8824	**0.9584**
Incept-ion	0.8048	**0.9827**	**0.9567**	**0.9922**
ResNet50	**0.9515**	**0.9991**	**0.9991**	**0.9974**
Xception	0.7730	0.9835	0.8215	**0.9939**

Table 2: Training Loss

	optimizers SGD	**Optimizers RMSprop**	**Optimizers Adagrad**	**Optimizers Adam**
VGG16	**0.4497**	**0.0918**	0.3716	**0.1069**
Incept-ion	0.4485	**0.0448**	0.2218	**0.0176**
ResNet50	0.1561	**0.0050**	**0.0324**	**0.0084**
Xception	0.5424	**0.0642**	0.4889	**0.0221**

Table 3: Validation Loss

	optimizers SGD	**Optimizers RMSprop**	**Optimizers Adagrad**	**Optimizers Adam**
VGG16	0.7091	0.9717	0.5111	**0.9556**
Incept-ion	0.5818	**0.4202**	**0.6020**	**0.6323**
ResNet50	**0.4182**	**0.5879**	**0.5818**	**0.4182**
Xception	0.5697	0.5818	0.5657	**0.5899**

Table 4: Validation Accuracy

Optimizers	**SGD**	**RMSprop**	**Adagrad**	**Adam**
VGG16	**0.5971**	**0.0793**	**0.9206**	**0.1127**
Incept-ion	**0.7371**	8.5688	**0.7872**	0.2959
ResNet50	1.2646	**0.6738**	1.4782	**0.7494**
Xception	**0.7339**	7.0186	**0.7754**	3.2455

7. IMPLEMENTATIONS

The analysis is done by using various steps :

1.***Data Acquisition***: Handwritten signatures are gathered, together with some identifying characteristics, to build a database for each and every individual. A unified database including the signatures of every participant is needed to assess the effectiveness of the signature verification system and compare the results obtained using various methodologies on the same database.

2.***Pre-Processing and resizing***: In the beginning, the image is resized to the appropriate size. Pre-processing is then carried out in accordance with the outcomes. No matter the size or slant supplied for the signature, the system must be able to retain good performance. The system's insensitivity is crucial since it will make it possible to correct the signature image.

*3.**Training Model***: The system's insensitivity is essential since it will allow the signature image to be fixed. To build CNN, We merge the TensorFlow backend with the Keras library. We train the model and evaluate its performance after loading the pre-processed picture directories.

4.Implementation: The Keras Python library has access to the file directory structure holding the signature images utilised in this investigation. The CNN was then created in Python using Keras and the TensorFlow backend to learn the patterns related to the signatures. The next stage in assessing whether the model was appropriate for the data was to test the model's precision and loss metrics. A signature from a holdout set was utilised as the evaluation tool to assess the accuracy of the model's predictions.

5.Dataset: The dataset employed in this study consists of 10 signatures, 5 of which are authentic and 5 of which are forgeries. The authentic signatures of 30 people may be found in this dataset, which was taken from the Kaggle Dataset. There are 300 images in total—150 authentic and 150 fake. The pictures are all in RGB format.

***6.Building a CNN*:**Using the Keras API, which serves as a wrapper for the Tensorflow and Theano backends, the appropriate modules are imported. The model was created using the Tensorflow backend. The first Python script that uses several classes for real and false signatures to train CNN. We train the model using the dataset using an 80-20 split (which can be altered depending on the circumstance).then, we store thein a.json file format. Furthermore, we keep the model's weights in a file The model and weights are loaded from the.json file and the.h5 file, respectively, in the second script. The model is then recompiled once a prediction is produced using a test specimen from the holdout set.

6.1Data gathering: Gather the photos of both authentic and fake signatures.

.

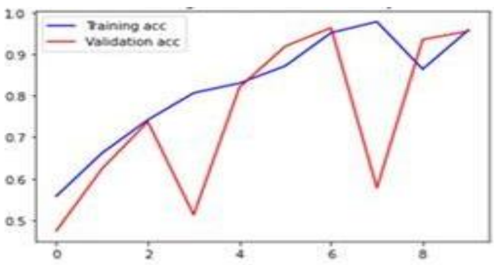
6.2Preprocessing: To avoid affecting the precision of the verification procedure, any noise or distortions in the collected signature photos should be removed during preprocessing. Operations like noise reduction, normalization, and image scaling fall under this category.

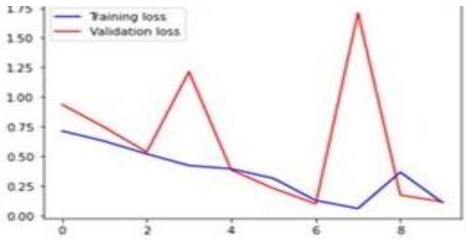
6.3Feature Extraction: The next step is to extract pertinent features from the signature image. The signature's size, curvature, shape, and orientation are all often used characteristics.

6.4Training: The retrieved characteristics from the real and fake signatures are used to train a classification algorithm. This can be done using machine learning methods like Support Vector Machines (SVM), Artificial Neural Networks (ANN), or Random Forest.

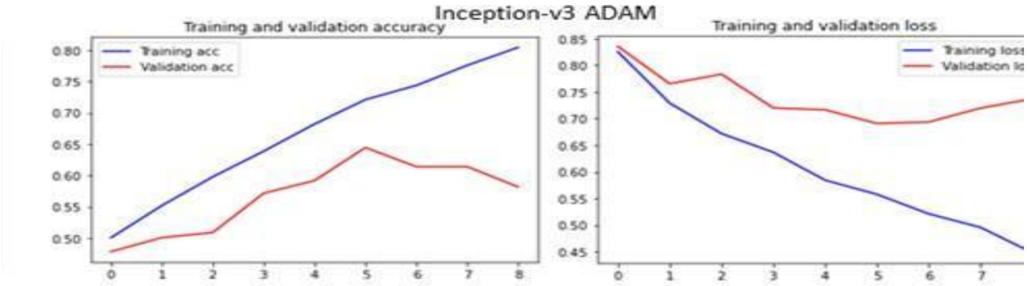
6.5Testing: The trained model is then put to the test on a different set of signature photos to see how well it predicts the future.

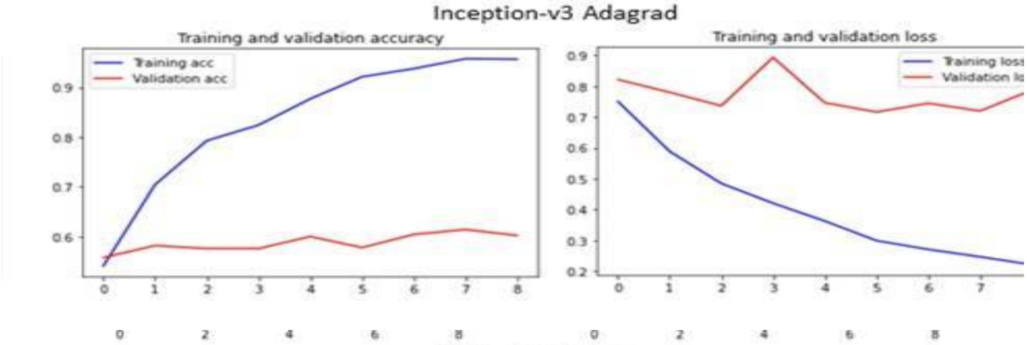
6.6Validation: To verify the model's effectiveness, the predictions are contrasted with the ground truth, which is the signature's real validity.

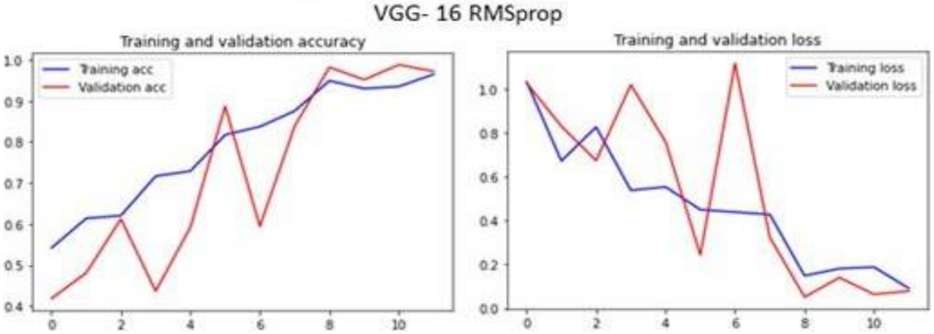




Fig(a) Training and Validation Accuracy Fig(b) Training and Validation



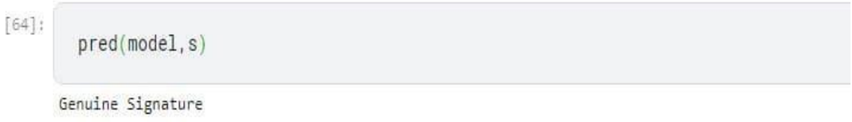




8. RESULTS

The CNN model's ability to discriminate between valid and invalid signatures from a bigger set without bias was evaluated using a smaller sample of signatures. The dataset was composed of 50 people were chosen at random. To investigate the effects of data augmentation on the generalization of the model and overall performance, a range of signatures with augmented data were utilized during training. To contrast the performance of genuine data vs. enhanced data during training, two unique sets with differing amounts of real signatures were used. The addition of data was also used.

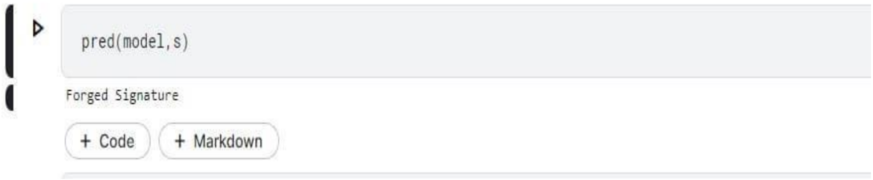
TenserFlow is used in this project because it produces the most accurate results for the input signatures that are provided. This method produced the output quickly within a matter of seconds so we can state that it is a reliable and time-saving method of verification



```
[64]:  pred(model,s)

Genuine Signature
```

From the below output , we can describe that the two inputs are forged which is fake signatures.



```
pred(model,s)

Forged Signature
```

9. CONCLUSION

Here, a writer-independent strategy is used to train the CNN, and a writer-dependent approach makes use of this trained CNN to extract features from offline signatures. The findings show that the curves of the signature images are precisely examined. the output of the Python programming language on Tensor Flow. A number of offline signature databases were used, and the findings demonstrate how well.Researchers have released a variety of offline signature verification methods over the past ten years. Although separating authentic signatures from expert forgeries is still a difficult operation, mistake rates have considerably decreased recently, in part because Deep Learning improvements have been applied to the problem. evaluating the most recent developments in the field. The VGG16 Adam model offered at least 95% training accuracy and 60% validation accuracy.

REFERENCES

[1] P. N. Narwade, R. R. Sawant, and S. V. Bonde, “Offline signature verification using shape correspondence,” International Journal of Biometrics, **vol. 10, no. 3, pp. 272–289, 2018**.

[2] Poddar J, Parikh V, Varti SK. Offline Signature Recognition & Forgery Detection using Deep Learning. *The 3rd International Conference on Emerging Data and Industry 4.0* **EDI40**, Warsaw, Poland, (April 6 - 9, 2020)

[3] Quazi Saad-ul Mosaher and Mousumi Hasan Offline Handwritten Signature Recognition Using Deep Convolution Neural Network **Vol**
7. *European Journal of Engineering and Technology Research* ISSN: 2736-576X (August 2022)

[4] José A. P. Lopes , Bernardo Baptista , Nuno Lavado and Mateus Mendes Offline Handwriiten Signature Verification Using Deep Neural Networks (2022).

[5] Daramola, S.A.; Ibiyemi, T.S. Offline signature recognition using hidden markov model (HMM). Int. J. Comput. Appl. (2010, 10, 17–22).