# Automated Signature Verification based on Hybrid Features and Proposed Deep Model

*Dr. Gauri Borkhade[1], Dr.Vivekanandam[2], Dr. Shashi Kant Gupta[3]*

[1]Researcher Lincoln Academy, Malaysia; [2] Prof.Lincoln Academy, Malaysia; [3] Prof. Lincoln Academy, Malaysia

gborkhade@gmail.com, vivekanandam@lincoln.edu.my, raj2008enator@gmail.com

---

**Abstract:**

Signatures are behavioral biometric traits of a person, used to authenticate a person. In all the legal transactions and legal documents signature is required to authenticate its legality. In such cases there are chances to forge the signature by other person to get the benefits. Therefore in order to check the genuineness of the signature, signature verification system is needed. In the state-of-the art literature there are several algorithms are proposed by different authors but still few challenges are remained to address such as the detection of skilled forgery and detection of intra class variations. There are two types of signature verification system namely offline and online. For such authentication of signature this project presents an application software which facilitates the feature of offline signature verification using the convolution neural network approach. This software is able to train the network with new dataset of signature and validate the authenticity of new signature of trained class. User can also perform experiment and analyze the training and verification of model with features result analysis feature. This software consists very efficient user interface so that any non- technical person can use the software without any difficulty. The aim of signature verification is to discriminate genuine signature from forge signature. Forgeries are of three types first one is simple forgery, second one is random forgery and third one is skilled forgery. This software consists very efficient user interface so that any non- technical person can use the software without any difficulty. For measuring the performance of system, the neural network was trained with dataset of 27 person with each having at least 100 signature and training result was 97 % accuracy.

**Keywords**: Classifier; Signature verification; Accuracy; Prediction

---

## Introduction

The handwritten signature of a person is commonly accepted as a means of verifying the legality of documents such as certificates, checks, drafts, letters, approvals, visa, passport and is indispensable in countering the forgery and falsification of such documents in diverse financial, legal, bureaucratic, academic, and other commercial settings. Take for example, in any bank whenever cashier receives a cheque from client, such cheque is verified with signature in it. The cashier compares that signature with stored record of genuine signatures before proceeding with any legal transaction. This convention of using signature as the route for confirming the authenticity of documents has been followed from medieval time to present and will continue in future. Such authentication with signature is at times very critical and crucial in legal scenario. For instance, a signature in any contracts has a vital role to indicate the identity of person of interest and also to provide evidence of intent and informed consent. Any falsification and fraudulent regarding such signature may result severe damages in persons lives and assets. In such cases, a systematic approach to verifying the signature is very necessary to prevent such forgery. Traditionally, authentication of specimen signature is achieved by person, comparing and evaluating the specimen with copies of genuine signature specimens acquired previously or with the help of some sort of witness. In case of Nepal especially in banking sector, signature verification is important subject and very critical in various transaction and approvals processes. Bank verifies the signature of the applicant/drawer of the checks on the basis of specimen signature retained in the bank's custody. In modern banking ICDM, Image Capturing and Display Module is an important tool, which captures specimen signature of the applicant and displays as and when required. But such simple approach may not be sufficient in all cases as various advanced forgery and falsification techniques are emerging. This project tries to assist and improve the verification process of human signature using machine learning techniques.

The objective of this research work is to develop a software application which can assists and improves the verification of authenticity of signature by implementing convolution neural network machine learning technique. Software application for verifying the genuineness of handwritten signature can be very useful in various sectors and activities. For example, authentication of cheques in banking transaction, official documents of person's identity and assets, contracts and statements, confessions, academic and professional certification etc. All these sectors and activities can benefit from systematic and modern techniques of verification which minimizes if not eliminate in validating the documents with signature of person of interest.

**Literature Review**

The area of Handwritten Signature Verification has been broadly researched in the last decades and still remains as an open research problem. This project focuses on offline signature verification, characterized by the usage of static (scanned) images of signatures, where the objective is to discriminate if a given signature is genuine (produced by the claimed individual), or a forgery (produced by an impostor). We present an overview of how the problem has been handled by several researchers in the past few decades and the recent advancements in the field. Article published in International Journal of Scientific & Engineering examined signature verification using neural network approach and analyzed its strengths and weakness [6]. Paper presented method which uses geometric features extracted from preprocessed signature images, which trained neural network using error back propagation training algorithm for verification of signature. They used a feature vector of dimension 60 to uniquely characterize a candidate signature. Article used different technique for verification analyzing different error rate: False Acceptance Rate (FAR), False Rejection Rate (FRR) and Correct Classification Rate (CCR). Result of research was 12% FAR, 16.7% FRR and Correct Classification Rate is 85.7%. A research conducted in Stanford University used convolutional neural network for offline signature verification based on the VGG-16 architecture and ICDAR 2011 SigComp dataset to train their model with transfer learning [7]. The dataset includes both online and offline signatures (of which we only use the latter) for both Chinese and Dutch signers. The dataset was split into a training set and testing set of non- overlapping IDs. The Dutch training set included a total of 366 images for 10 IDS, with about 25 genuine signatures and 11 forged signatures for each ID. The result from research was validation accuracy of 67.1%, FAR of 33.0%, FRR of 33.0 %. The main limitation of research was limited dataset and with large data, the result would have been better.

Research paper of topic “offline handwritten signature retrieval using curvelet transform” proposed a new method for offline handwritten signature retrieval based on curvelet transform [8]. It focused on applications of image processing with similarity retrieval of an image from large collections of images. In such case image indexing become important for efficient organizational and retrieval of images. The proposed 4 system used a curvelet based texture feature extraction. The performance of the system was tested with an image database of 180 signatures. The result obtained indicated that the proposed system was able to identify signatures with greater accuracy even when part of signature was missing.

**Machine Learning**

Machine learning is a subsidiary of artificial intelligence that facilitates a techniques where machine can make decision based on its experience and improve and learn with time and use without explicitly programmed. Machine learning focuses on the development of computer programs that can access the data and use it to learn for themselves. Machine learning algorithms are often categorized as supervised and unsupervised machine learning algorithms.

**Supervised Learning**

Supervised machine learning is the search for algorithms that reason from externally supplied instances to produce general hypotheses, which then make predictions about future instances [9]. It can apply what has been learned in the past to new data using labeled examples to predict future events. Classification algorithms like decision tree, Naive Bayes, Neural Networks, k-Nearest Neighbors and Support Vector Machines (SVM) are some examples of supervised learning algorithms.

**Unsupervised Learning**

Unsupervised learning is the type of machine learning algorithm where there is no any defined or labeled class and it itself draws the inferences from datasets [10]. Unsupervised learning studies how systems can infer a function to describe a hidden structure from unlabeled data. Clustering algorithms like K-means, mixture models, hierarchical clustering, anomaly detection, Hebbian Learning Network, Generative Adversarial Networks etc. are some examples of unsupervised learning algorithms.

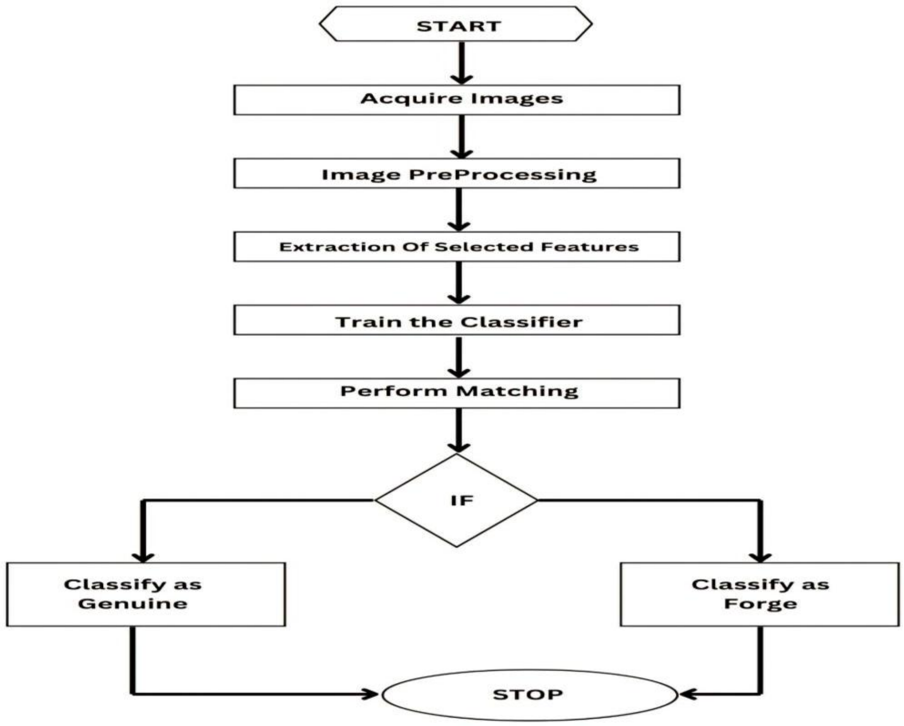
**Artificial Neural Network**

Artificial Neural Networks are composed of multiple nodes, which imitate biological neurons of human brain. The neurons are connected by links called axons and they interact with each other. The nodes accept input data and perform operations on the data. The result of these operations is passed to other neurons. The output at each node 5 is called its activation or node value. Each link is associated with weight. ANNs are capable of learning, which takes place by altering weight values.

Single layer perceptron, Multi layers perceptron, Backpropagation network, Recurrent neural network, Self-Organizing Map(SOM), Hebbian network, Kohenon network, Hopfield Network, Convolutional neural network etc. are the different types of neural network architectures each suitable for different purposes [11]. Neural networks in practice like sales forecasting, industrial process control, customer research, and targeted management etc. Neural networks in medicine like modelling and diagnosing the cardiovascular system, in business etc. are few of the many application area of artificial neural network [12]. Convolution Neural Network A Convolutional Neural Network (CNN) is comprised of one or more convolutional layers (often with a subsampling step) and then followed by one or more fully connected layers as in a standard MLP. The architecture of a CNN is designed to take advantage of the 2D structure of an input image or other 2D input such as a speech signal. This is achieved with local connections and tied weights followed by some form 6 of pooling which results in translation invariant features. Another benefit of CNNs is that they are easier to train and have many fewer parameters than fully connected networks with the same number of hidden units. Architecture of a CNN consists of a number of convolutional and subsampling layers optionally followed by fully connected layers. For an example, in a CNN where input is image, the input to the convolutional layer be m x m x r image where m is the height and width of the image and r is the number of channels, e.g. an RGB image has r = 3. The convolutional layer will have k filters (or kernels) of size n x n x q where n is smaller than the dimension of the image and q can either be the same as the number of channel r or smaller and may vary for each kernel. The size of the filters give rise to the locally connected structure which are each convolved with the image to produce k feature maps of size m – n + 1. Each map is then subsampled typically with mean or max pooling over p x p contiguous regions where p ranges between 2 small images and is usually not more than 5 larger inputs. Either before or after the subsampling layer an additive bias and sigmoidal nonlinearity is applied to each feature map.

**Methodology**

The system consists of two modules core implementation of convolutional neural network and cross platform desktop application with various functionalities like signature verification, training features for new signature classes and training result analysis. At the initial phase of project, application development and CNN implementation were done separately and integrated together to single system later in project. Data Collection and Pre-processing is in order to train the classification model, signature data which in this case was images of signature we sourced the dataset signature- classification-using-siamese-pytorch from kaggle. Once downloaded, the data underwent extensive preprocessing to prepare it for analysis. Preprocessing steps were crucial in transforming raw data into a clean and structured format, allowing for more accurate and reliable analysis and preprocessing was done on them with cropping the specimens, cleaning and scaling to appropriate size, gray scaling and giving proper name and putting them in separate directory. Implementation of convolutional neural network for this research work is based on the Inception architecture of Google Net [13]. For the project, implemented neural network is Inception-v3 trained tensor flow model. Training an Inception-v3 network from scratch is a computationally intensive task and depending on computer setup may take several days or even weeks which is not possible with limited resources. In order to overcome this problem transfer learning mechanism was adopted and retraining was performed on the Inception v3 model, which is trained for the ImageNet Large Visual Recognition Challenge using the data from 2012, with collected training data. The top layer receives as input a 2048-dimensional vector for each image. Training is done on a softmax layer on top of this representation. Assuming the softmax layer contains N labels, model corresponds to learning N + 2048*N model parameters Corresponding to the learned biases and weights. Training is done with collected data of signatures and training result was analyzed until satisfactory result was achieved.




The system design shown in the Data Flow Diagram (DFD) Level-2 involves multiple processes and entities. The User initiates Data Acquisition , which collects and stores data. The data undergoes Image Resizing and Format Conversion before forming the Training Data Set. The system displays training progress and accuracy graphs. Test data is acquired and compared to the trained data in the Comparing process. Results are generated and accessed by the Admin. The flow ensures seamless data processing, model training, testing, and result analysis for efficient system functionality. Our research was based on this dataset as well as two publicly available datasets, SigComp2011 [20] and CEDAR [21]. First, we pre-processed all training and test images to get the signatures ready for the identification stage. The images were then converted to greyscale, histogram equalised, blurred, and resized. Then, we proposed an approach based on the fusion of appearance-based features (Linear Discriminative Analysis [LDA]), texture-based features (Grey-Level Co-occurrence Matrix [GLCM]), and frequency- based features (Fast Fourier Transform [FFT]) to build a hybrid feature vector for each image for identification.

Finally, we developed a Deep Neural Network model to identify each person. The resultant hybrid feature vectors were fed into the proposed deep neural network architecture to be trained to use them later to classify the new features

The validation of signature in many cases are highly critical and any inaccuracy in the authentication may result serious consequences and damages. With the advancement in technology, new and complex forgery and fraud techniques are emerging. In order to avoid such scenario and prevent potential damage, modern robust approach must be adopted in verifying the genuineness of signature. Adopting such approach will assists person in making decision over authenticity of signature and prevents mistakes.

Application development and system modules integration desktop application was developed with python programming language and anaconda python data science platform. For graphical user interface, python's de-facto standard GUI package Tkinter was used. The convolutional neural network module was integrated to main application. Other features such as verification of signature, training features for new classes of signature holder, analyzing results with graph representation, whole system after integration and eliminate present errors. Performance and functionality testing were done in order detect and eliminate bugs and errors in system. These detected bugs and problems were solved with changes in codes and sometimes in changes in system configuration and even system design when needed, exploiting the iterative and continuous improvement nature of agile development.

The system design shown in the Data Flow figure 1. (DFD) Level-2 involves multiple processes and entities. The User initiates Data Acquisition (1.0), which collects and stores data. The data undergoes Image Resizing (2.1) and Format Conversion (2.2) before forming the Training Data Set (3.0). The system displays training progress and accuracy graphs. Test data is acquired and compared to the trained data in the Comparing (5.2) process. Results are generated and accessed by the Admin. The flow ensures seamless data processing, model training, testing, and result analysis for efficient system functionality.

**Result**

A customized version of 2D CNNs called 1D Convolutional Neural Networks (1D CNNs) was used with a kernel of size 3, a padding value equal to 1 (to give the kernel extra space to cover the vector, padding was applied to the frame of the feature vector), and stride of 1 that modifies the amount of movement over the feature vector in which the filter would move one unit at a time. Leaky Rectified Linear Units (Leaky ReLU) function, a non-linear activation function, was applied after the CNN layers. When compared to conventional activation functions, the Leaky ReLU function can speed up the training of deep neural networks. A new pooling layer was added after the Leaky ReLU layers. The Max-Pooling method was then used to obtain the maximum output. For 1D temporal data, the maximum value over the window determined by the pool size was used to down sample the input representation. The window was moved a little. When the "valid" padding option was used, the output had the following shape: output shape = (input shape - pool size + 1) / strides. To take the output of the preceding layers, "flatten" them, and create a single vector that can be an input for the following stage, one flattened layer was added.

One dense layer was used after the flattened layer with the Softmax activation function, as a classifier of the resultant vectors. The stated number of neurons in the dense layer would impact the output shape.
The dense layer carried out the action: Activation (dot [input, kernel] + bias) was equal to output.This section presents the experimental findings from using these techniques on three different datasets of handwritten signatures. Additionally, the outcomes of the system are compared with other deep neural networks such as VGG16 and VGG19 and four types of machine learning algorithms as well as state-of-art methods that utilize and incorporate the same datasets. The proposed model was trained with a learning rate of 0.001, epochs of 100, and a batch size of 64. The total number of parameters obtained from this model on the SigComp2011 dataset is equal to (6,985,521), (6,398,427) on the CEDAR dataset. The three datasets were used separately to compare the effectiveness and performance of this proposed approach with different approaches. These are SigComp2011, which is considered more challenging as it contains Chinese and Dutch signatures.
The total number of images in this dataset is 937, and the popular CEDAR dataset has 2640 images while the private dataset SigArab contains 372 Arabic signatures. The datasets were divided randomly into

training (70%) and test (30%) partitions. Four performance metrics—accuracy, precision, recall, F-score, and loss metrics—were used to evaluate the efficacy of this proposed methodology.

**Conclusions**

In conclusion, CNNs provide a highly accurate and efficient method for automating signature verification. They enhance security, speed up the verification process, and reduce the risk of fraud across various applications. As technology advances, CNNs will continue to improve, making signature verification even more reliable. The signature verification project using CNN achieved high accuracy in distinguishing genuine and forged signatures. The system effectively handled offline verification and showed consistent performance. This study introduced an authentic signature identification method based on the integration of static (offline) signature data and a proposed deep-based model.

In this method, three types of signature features were fused: LDA as appearance-based features, FFT as frequency-based features, and GLCM as texture-based features. To assess the efficiency and performance of this model, two well-known datasets, SigComp2011 and CEDAR, and our private dataset SigArab were employed. First, four pre-processing stages were carried out on the datasets. Then, LDA, GLCM, and FFT were used as feature extraction methods to build a hybrid feature vector for each image. These features were then fed into the proposed model. The primary goal of the proposed model was to categories the hybrid features derived from the earlier stage to identify each signer. This model was built with 25 layers, nine of which were convolutional layers (with filters equal to 16, 32, 64, 128, 256, 512, and 395, respectively), six Max-pooling 1D layers, eight Leaky-ReLU 1D layers, one Dense 1D layers, and one Flatten layer. The suggested technique enhanced the identification accuracy and outperformed other modern techniques, with an accuracy value of 99.7% for both public datasets and 99.23% for the private dataset. Additionally, it depicted a high precision rate reaching 1.00 for both public datasets, which can be attributed to the architecture of the model and the choice of effective signature features

**References**


1. Surabhi Garhawal and N. Shukla, "A Study on Handwritten Signature Verification Approaches," International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), vol. 2, no. 8, 2013.

2. W. Hussein, A. S. Mostafa and O. Ibrahim, "Image Processing Based Signature Verification Technique to Reduce Fraud in Financial Institution," MATEC Web of Conferences, p. 76, 2016.

3. M. Hanmandlu, "Neuro-Fuzzy Approaches to Signature Verification," 2006.

4. D. S. A. Daramola and P. T. S. Ibiyemi, "Offline Signature Recognition using Hidden Markov Model (HMM)," International Journal of Computer Applications , vol. 10, 2010.

5. B. Zhang, "Offline signature verification and identification by hybrid features and Support Vector Machine," International Journal of Artificial Intelligence and Soft Computing, vol. 2, pp. 302-320, 2011.

6. A. Pansare and S. Bhatia, "Off-line Signature Verification Using Neural Network," Internatio nal Journal of Scientific Engineering Research , vol. 3, no. 2, 2012.

7. G. Alvarez, B. Sheffer and M. Bryant, "Offline Signature Verification with Convolutional Neural Networks," Stanford.edu, Stanford, California, 2016.

8. S. B. Kotsiantis, "Supervised Machine Learning: A Review of Classification Techniques," Department of Computer Science and Technology, University of Peloponnese, Peloponnese, Greece, 2007. 23

9. M. T. Hagan, H. B. Demuth and M. H. Beale, Neural Network Design, Oklahoma, 2014.

10. L. Torrey and J. Shavlik, "Transfer Learning," University of Wisconsin, Madison WI, USA, 2008.

11. N. Wang, L. W., J. Y., S. P., R. S., . A. D., . E. D., V. V. and R. A., "Going deeper with convolutions," Proceedings of the IEEE, pp. 1-9, 2015.

12. K. N. Rao, G. K. Naidu and P. Chakka, "A Study of the Agile Software Development Methods, Applicability and Implications in Industry," International Journal of Software Engineering and Its Applications, vol. 5, 2011.

13. Soleimani, M., & Naderan, M. (2018).Title: Offline signature verification using deep convolutional neural networks. Source: 2018 4th Iranian Conference on Signal Processing and Intelligent Systems Summary: This paper discusses an offline signature verification system that employs deep CNNs to automatically extract features from signatures and achieve better performance than traditional handcrafted features.

14. Hafemann, L. G., Sabourin, R., & Oliveira, L. S. (2017).T itle: Offline handwritten signature verification—Literature review. Source: 2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA)Summary: A comprehensive survey of offline handwritten signature verification methods, including CNN-based approaches. It emphasizes the role of deep learning, particularly CNNs, in improving the accuracy of signature verification systems.

15. Zahid, M., Akram, U., & Zia, S. (2019). Title: Automatic Signature Verification Using CNN and HOG Features.Source: 2019 International Conference on Communication Technologies (ComTech)Summary: This paper presents a hybrid approach combining CNNs with Histogram of Oriented Gradients (HOG) features to enhance signature verification performance, showing that CNNs can learn robust feature representations from signature data.

16. Ooi, C. H., Khalid, N. E. A., & Samsudin, A. (2019).Title: Signature verification using a hybrid CNN and RNN approach.Source: International Journal of Advanced Computer Science and applicationsSummary: This study explores the combination of CNNs and Recurrent Neural Networks (RNNs) for signature verification. The CNN extracts spatial features while the RNN processes the sequential information, resulting in a more accurate verification system.

17. Xinyu Li, Jing Xu, Xiong Fan, Yuchen Wang and Zhenfeng Zhang, "Puncturable Signatures and Applications in Proof-of-Stake Blockchain Protocols", July 16,2020; DOI 10.1109/TIFS.2020.3001738,IEEE.

18. Jacopo Pegoraro , Francesca Meneghello , Michele Rossi "Multiperson Continuous Tracking and Identification From mm-Wave Micro-Doppler Signatures",September 13,2020.

19. Anamika Jain1 · Satish Kumar Singh1 · Krishna Pratap Singh1, “ Handwritten signature verification using shallow convolutional neural network ”, April 07 ,2020.

20. Azmi AN, Nasien D, Omar FS (2017) Biometric signature verification system based on free manchain code and k-nearest neighbor. Multimed Tools Appl 76(14):15341–15355.