

Human Signature Verification System Using CNN with Tensor flow

ArunKumar G¹, Anand K², Dinesh P³, Abimanyu R⁴, Vijayakumari V⁵

^{1,2,3,4,5} Computer Science and Engineering, The Kavay Engineering College, Tamilnadu, India.

How to cite this paper:

ArunKumar G¹, Anand K², Dinesh P³,
Abimanyu R⁴, Vijayakumari V⁵; Human
Signature Verification System Using CNN
With Tensor flow", IJIRE-V4I03-235-242.



<https://www.doi.org/10.59256/ijire.2023040381>

Copyright © 2023 by author(s) and
5th Dimension Research Publication.

This work is licensed under the Creative
Commons Attribution International License
(CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>

Abstract: One of the most popular verification biometrics is the signature. On checks, forms, letters, applications, minutes, and other documents, handwritten signatures are required. A person's handwritten signature must be individually identified because each individual's signature is unique by nature. Signature verification is a popular technique for confirming anyone's identity while they are not present. Human verification can be inaccurate and occasionally unsure. The use of Convolutional Neural Networks (CNN) for Writer-Dependent models in signature verification is examined in this research. In order to create forged signatures, random distortions were created in real photos using an auto encoder and then fed to the classifier during training. In addition to demonstrating various test outcomes for varying the number of training sets of images, the study describes all image pre-processing procedures that were applied to the image. In the Persian dataset, the system's average test accuracy is 83% after 22 real photos were used to train it. When the model was trained on nine real photos, accuracy dropped by 9.4%.

Key Word: Offline Signature Verification, WD (Writer Dependent), CNN (Convolutional Neural Network), FAR (False Acceptance Ratio), FRR (False Rejection Ratio), Auto encoder

INTRODUCTION

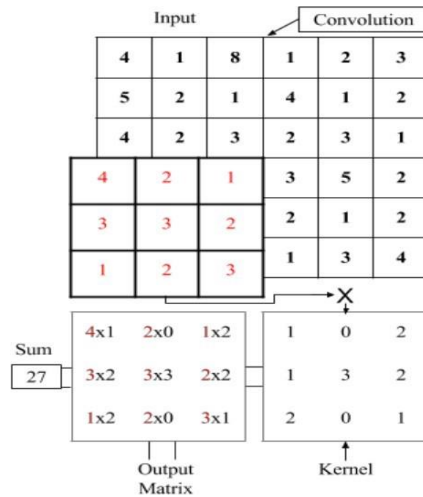
The process of using a user's handwritten signature as a behavioural biometric to verify their identity is known as handwritten signature verification. Automatic handwritten signature verification utilising various artificial models has been the subject of numerous investigations. The Related Works section of this research article reviews a few of the earlier studies. The fact that a handwritten signature has been one of the most widely accepted forms of identification in our society for hundreds of years and doesn't require sophisticated technology is one of the advantages that signature verification has over many other forms of biometric technologies, such as voice authentication, retina scan, fingerprint, etc. Banks and other institutions continue to accept user authentication procedures based on signature verification. Furthermore, the most popular method for verifying a user in a system uses digital PIN and password procedures. Depending on the data gathering method being used, signature verification can be separated into two primary categories: offline and online signature verification. In offline signature verification, a document containing the signature is scanned to obtain a digital image of it. Electronic gear, such as electronic signature capture pads, is used in online signature verification to capture the pen motions made during signing. These strategies can all be used in their respective sectors.

Convolutional Neural Network (CNN)

CNNs were first proposed by Yann LeCun and Yoshua Bengio in 1995 AD. A feed-forward neural network called CNN is capable of removing topological details from an input image. It takes features out of the picture and feeds those features into a classifier, which classifies the picture. Common geometric transformations including translation, scaling, rotation, and squeezing have little effect on CNNs and are generally invariant to distortions. Local receptive fields, shared weights, and spatial or temporal sub-sampling are three architectural concepts that CNNs combine to provide some level of shift, scale, and distortion invariance [3]. CNNs are typically taught through back propagation, much like a conventional ANN. Convolutional and max-pooling layers alternate in CNN layers. In order to extract features from local, corresponding fields, a convolutional layer is used. It is structured into feature maps, which are planes of neurons. Each unit in a network with a 5x5 convolution kernel has 25 inputs connected to the local receptive field, a 5x5 region in the preceding layer. All units of a feature map have a common trainable weight that is assigned to each connection. This characteristic, known as the weight sharing technique, is used in all CNN layers and allows for the reduction of the number of trainable parameters.

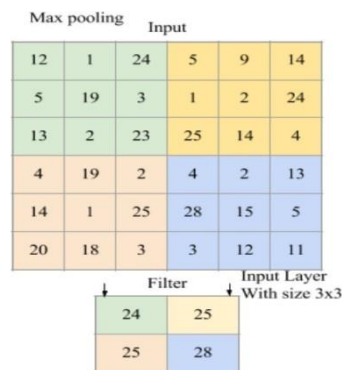
Convolution Layer:

A nxn value matrix called Filter (kernels), which is convolved during the convolution procedure, surrounds each pixel in the image as well as its immediate surroundings. The process involves multiplying the filter matrices with the same dimension area in the picture matrices element by element. The output of the sum is then placed at the same pixel position where the current convolution kernel was initially centred in the result image.



Max-Pooling Layer:

Max-pooling layer's primary goal is to down-sample an image. The picture matrix is split into several areas of pixels by the Max-Pooling operation, each region being spaced N pixels apart both vertically and horizontally. Each divided matrix's maximum value is outputted as a smaller output image.



Rectification Linear Unit (ReLU):

ReLU is an activation function on the outputs of the convolution layer. For each node, the mathematical expression for ReLU activation is:

$$f(x) = \max(0, x)$$

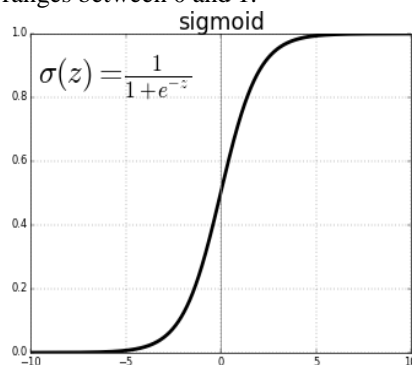
This sets all negative pixel values from the convolution to zero, while leaving each positive value unchanged.

Sigmoid Activation:

A sigmoid function is a mathematical function having a characteristic "S"-shaped curve or sigmoid curve. The equation is given by:

$$\text{sigma}(x) = \frac{1}{1 + e^{-x}}$$

The output of sigmoid function ranges between 0 and 1.



Auto encoder:

An encoder, a decoder, and a loss function make up an auto encoder. This method is well-liked for the unsupervised learning of complex distributions. An autoencoder's primary goal is to compress data by encoding it with mathematical formulas or other processes and then reverse the process to decompress the downscaled data back to its original size. Some data is lost during this operation, resulting in erratic variances in the original data. The more compression used, the more information is lost as a result.

II.LITERATURE REVIEW

Kumar demonstrated an offline technique for verifying signatures that makes use of the global and texture features of the signature. The plan is based on a method that pre-processes the signature to produce a binary picture from which the global and texture feature points are calculated and a feature vector is maintained. These feature points served as the foundation for all calculations. A classifier called an artificial neural network (ANN) was utilised to train and validate signatures.

In a survey that they presented, Batista et al. outlined the key methods for offline signature feature extraction and verification. They also provided solutions for the challenges associated with small data sets. Guerbai et al. presented a writer-independent One-Class SVM (OC-SVM) based Human Signature Verification system with the goal of reducing the issues caused by a large number of signees in the system. The suggested model only considers one class, namely real images of signatures, which is a positive trait. For training, the classifier exclusively used real signatures. The algorithm, however, faced a severe hurdle due to the small amount of legitimate photos.

Rezaei et al used fully convolutional networks (FNN) to perform signature verification on a Persian dataset. Their solution suggested a FNN with a 270 x 360 pixel image input size. There are no pre-processing techniques in the suggested procedure. The system had a 76.71% accuracy rate for predicting signatures.

In a Persian dataset, Khalajzadeh et al. performed signature verification using a Convolutional Neural Network (CNN), achieving 95% accuracy. To analyse signatures and balance the amount on a check, Miah et al. in 2015 created a model utilising the ANN. However, his suggested system did not take into account signatures made on actual checks.

Comparative study of various algorithms in literature review:

TITLE	DESCRPTION
Puncturable Signatures and Applications in Proof-of-Stake Blockchain Protocols	One of the most promising alternatives to the power-hungry proof-of-work protocols are "proof-of-stake blockchain protocols."
Multi person Continuous Tracking and	In this study, we explore the use of backscattered mm-wave radio signals
Identification From mm-Wave Micro-Doppler Signatures	for the simultaneous identification and tracking of people moving through indoor spaces.
Handwritten signature verification using shallow convolutional neural network	A person's identity can be shown incontrovertibly and uniquely by their handwritten signature. Due to its simplicity and originality, it holds a crucial position in the field of behavioural biometrics.
Biometric signature verification system based on freeman chain code and k-nearest neighbor.	A person's signature is one of their biometrics, and because it can change with age, attitude, and surroundings, among other things, it cannot be exactly matched to another person's signature.

III.EXISTING PROBLEM

The learning set and testing set were created by randomly dividing the data. The testing set contained all of the falsified samples. demonstrates the accurate verification rate that was attained during the experiments. The outcomes of the experiment (learning and testing) show the following intriguing systemic facts: Whether they were used to compare the sample to the prototype or the reverse, all of the features utilised for verification were almost equally successful. The most points are found in central line characteristics per image. Due to their concentration in the centre, some of these points are meaningless.

Therefore, a more effective tool is required to address this issue, and our study suggests one that might help people make informed decisions when authenticating handwritten signatures. The subject of handwritten signature verification has been extensively researched over the past few decades, however it is still an unsolved research issue.

IV.PROPOSED SOLUTION

Predicting whether a human handwritten signature is real or fake is the goal of the suggested system. More examples of photos are gathered, including both authentic and fake signatures, and they represent various classes. To stop hostile individuals from forging signatures, we suggested an offline signature verification method based on Deep Learning (DL).

The database containing the signatures is tested after training the CNN models on the images of the signatures, and

the outcome indicates whether the matching signature is real or fake. A extremely powerful and effective technique that may be used on an embedded device is convolutional neural networks. The effectiveness of the algorithm can be confirmed using the aforementioned tests. All of these tests yield very comparable results. The training of signature datasets obtained from multiple perspectives is a crucial parameter to take into account, according to algorithm testing. The public will be helped in using their signatures in the safest and most effective way possible with the help of this intelligent human signature verification technology.

Biometric Signature Verification System:

The phrase "biometric signature verification system based on freeman chain code and k-nearest neighbour" [1], Three issues were found in Stage 1's analysis of the problem's background. The first one is relevant to the SVS as a whole. Some solutions to this issue are defined since signatures are a sort of biometric that can alter with mood, environment, and age. A decent signature database needs to be updated at certain intervals in order to remain current and useful. In addition, one must sign consistently in order to create a string of signatures that are very similar to one another. The second issue is with the FCC generation, which was unable to extract from the damaged portions of the signature. In order to extract the FCC, just the greatest continuous portion of the signature is picked. The third issue had to do with verification in order to have a decent outcome. To get the desired results using k-NN, earlier steps, particularly pre-processing and feature extraction, must function effectively. Stage 3 of this research has two feature extraction components. The FCC functionality is discussed in the first section. Chain code representation, which records the direction in which the next pixel will be located and corresponds to the neighbourhood in the image, provides the outline for a signature image.

Multiple Neural Classifier:

The phrase "Signature verification using multiple neural classifiers" - On Sun's Spark System, a prototype recognition system was developed using C. Ten separate people's samples were obtained for the experiment. Each person provided fifteen authentic signature samples, which were then collected. In addition, 100 random forgeries were utilised to evaluate the system. A series of trials were run to gauge the approach's effectiveness. In every experiment, five randomly chosen samples of each person's actual signature were used to train the classification nets. These nets had the same number of output nodes as there were participants in the experiment.

Two Stage Neural Network Classifier:

"A new neural network classifier-based two-stage neural network signature verification technique" [3] - This research suggests a fresh method for off-line signature verification and identification. The entire system is built on a two-stage neural network classifier with a one-class-one-network arrangement and 160 features divided into three subsets. Only small, fixed-size neural networks need to be trained during the first stage's training procedure, but the training method for the second stage is simple. The majority of our design work went towards incorporating the majority of the intelligence into the system's structure. Use all features and let the neural networks determine which ones are relevant and which ones are not was the general rule of thumb when determining which features to include and which ones to leave out.

Image Processing:

1. **Image Aquisition & Resize:** Any shape of image, typically 1024x768, was taken with a digital camera. For speedier processes, the image was reduced in size to a shape with a maximum pixel length of 600 while preserving the original aspect ratio.
2. **Median Blur:** To remove any grainy noise from the acquired image, a 7x7 window with a median blur was applied to each RGB channel.
3. **Gray Scaling:** By calculating the mean of the channels in each pixel, the three channel RGB image was then transformed into a single channel grayscale image.
4. **Fast non-local means De-noising:** Quick OpenCV tool With Template window size = 10 pixels and Search window size = 21 pixels, Non local Means Denoising was used to reduce fine noise.
5. **Image Segmentation:** The image underwent global thresholding, with the thresholding value set to the average pixel value. Threshold Value = Average(image).
6. **Image Localization:** Cropped a rectangular portion of image that contains only the signature.
7. **Padding:** To make the image square without distorting the signature image, apply padding on the required side.
8. **Image Resize:** The image was resized using Nearest neighbourhood to 300x300, as per the input shape of CNN model.
9. **Image Negative:** Generated negative of the image by: Negative = 255 – Image.

CNN Architecture:

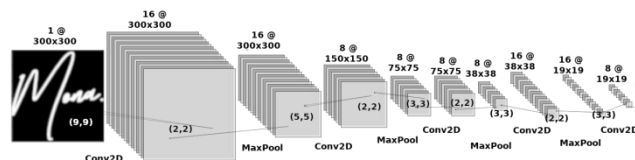
The suggested model has an architecture made up of five convolutional layers, each followed by a max-pooling layer. After flattening the output, an ANN was used to classify the data into binary categories. While an ANN was used to classify the image, convolutional layers were employed to extract features from signatures. The network received the 300x300 input image. 16 filters with a 9x9 pixel resolution were employed to convolve the image in the first convolution operation, which was followed by a 2x2 pixel max-pooling operation. 32 feature maps were sampled from the first 16 layers in the second convolution operation using a 5x5 filter, followed by max-pooling in a 2x2 region. 32 feature maps were

Human Signature Verification System Using CNN withTensor flow

sampled in the third convolution operation using a 3x3 filter and max-pooling in a 3x3 region. Using a filter of dimension 2x2, 16 feature maps from the preceding 32 layers were sampled in the fourth convolution operation, which was followed by max-pooling in a 2x2 zone. With the use of a 3x3 filter and a 2x2 region for max-pooling, 8 feature maps from the previous 16 layers were sampled in the fifth convolution operation. All five convolutional layers employed the Rectifier Linear Unit (ReLU) as its activation function.

After that, the tensor was reduced to a single dimension with 1936 neurons. 20% dropout in connection was used to connect 256 neurons in a second completely connected hidden layer. The model was made less overfit by the addition of dropout. Similar to the previous layer, another hidden layer with 128 neurons and a 40% dropout was fully connected. Implementation of both hidden layers.

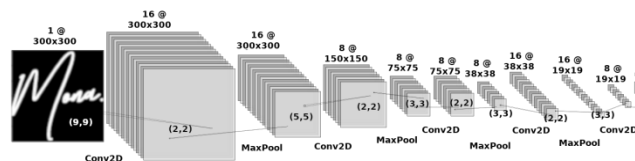
The activation of ReLU works. In order to identify output between the range of 0 and 1, the output layer of a single neuralnetwork utilising Sigmoid activation function was used. The degree of signature similarity to authentic signatures was determined by the neuron's final output. 1 denoting a precise match and 0 denoting no match.



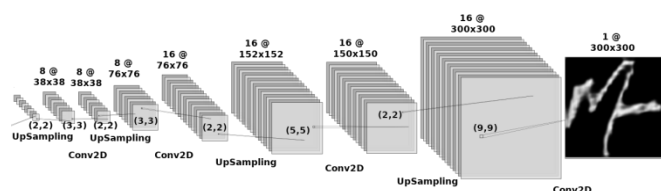
Proposed CNN Model Architecture

Auto Encoder Architecture:

By convolving, downsampling, and rescaling the images to their original dimensions, an auto encoder was utilised in the system to create fake images from real ones. As a result of the image data being compressed during encoding, figure 4's noisy image is created by further upsampling. Five convolution procedures, each followed by maximum pooling, made up the encoder component. The decoder component, in a similar vein, featured 5 convolution processes, each of which was followed by an up sampling operation. Binary Cross-entropy with ad delta optimisation was the loss function used. Fewer training epochs seem to be more effective for ada delta optimisation. The architecture of the proposed auto encoder and the kernel measurements made at each stage.



Encoder



Decoder

Dataset and Training:

Thirty individual people's Persian signatures were used in this study. Forgery was divided into three categories: Simple (66 photos per category), Skilled (6 images per category), and Opposite Hand (3 images per category). Each individual dataset had 27 real images. A set of 115 photos taken from 115 distinct people's simple forgeries, one from each person, were set as random images, or set false, during training because we were developing binary classifiers.

Training:

9 real photos, 9 fake images created using an auto encoder, and 115 random images were utilised in Training Case I for each WD classifier. For 20 epochs, a total of 133 pictures were trained. As test data, 3 Opposite Hand forgeries and 5 real photos were employed. Since the classification is binary, the pictures were trained using Adam (Adaptive momentum) optimisation with a learning rate of Binary Cross-entropy loss function.

















V.RESULTS















Institutions that demand better accuracy, for example, may set standards for high results, and vice versa. Table 1 displays the output of 30 classifiers trained on cases I and II. The False Acceptance Ratio (FAR) of the Signature is given by equation:

$$\text{FAR} = \frac{\text{No. of forgery signature accepted}}{\text{Total no. of forgery images}}$$

The False Rejection Ratio (FRR) of the Signature is given by equation:

$$\text{FRR} = \frac{\text{No. of genuine signature rejected}}{\text{Total no. of genuine signatures}}$$

S. NO	IMAGES	CASE I (9 Images)			CASE II (22 Images)		
		FAR	FRR	Acc	FAR	FRR	Acc
1.		0%	40%	79%	0%	20%	87%
2.		0%	20%	78%	33%	0%	87%
3.		0%	20%	86%	0%	0%	95%
4.		0%	0%	94%	0%	0%	99%
5.		0%	40%	72%	0%	40%	69%
6.		0%	20%	87%	0%	20%	87%
7.		0%	0%	98%	0%	0%	99%
8.		0%	60%	72%	0%	0%	94%
9.		0%	40%	67%	0%	0%	99%
10.		0%	20%	85%	0%	0%	99%
11.		66%	100%	24%	66%	20%	62%
12.		0%	60%	60%	66%	40%	53%
13.		0%	0%	85%	0%	20%	84%
14.		0%	60%	59%	0%	60%	63%
15.		66%	20%	59%	100%	0%	62%
16.		0%	20%	86%	66%	20%	71%

17.		0%	0%	99%	0%	0%	99%
18.		0%	0%	98%	0%	0%	99%
19.		0%	40%	74%	0%	40%	80%
20.		0%	20%	62%	0%	40%	77%
21.		0%	60%	62%	0%	0%	99%
22.		66%	20%	66%	66%	0%	65%
23.		33%	40%	66%	66%	20%	68%
24.		0%	80%	49%	0%	60%	66%
25.		0%	0%	87%	0%	0%	99%
26.		0%	20%	87%	0%	20%	87%
27.		0%	60%	61%	0%	20%	85%
28.		0%	0%	99%	0%	0%	99%
29.		0%	80%	54%	0%	60%	69%
30.		0%	20%	87%	0%	20%	92%

VI.CONCLUSION

Thus we have successfully developed a model using the AlexNet algorithm of one of the most powerful deep learning models, CNN (Convolutional Neural Network) to achieve an higher accuracy in terms of recognizing a signature and classifying it as to whether it is the corresponding person's original signature or a fake one.

This can be useful in various sectors which involves collecting authentic information of the customer, employee or any other person. Some of the sectors include banking, database related fields, healthcare etc.

References

1. 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.
2. Jacopo Pegoraro , Francesca Meneghello , Michele Rossi , "Multiperson Continuous Tracking and Identification From mm-Wave Micro-Doppler Signatures",September 13,2020; <https://www.ieee.org/publications/rights/index.html>

3. Anamika Jain¹ · Satish Kumar Singh¹ · Krishna Pratap Singh¹, “ Handwritten signature verification using shallow convolutional neural network ”, April 07,2020 ; <https://doi.org/10.1007/s11042-020-08728-6>
4. Azmi AN, Nasien D, Omar FS (2017) Biometric signature verification system based on freeman chaincode and k-nearest neighbor. *Multimed Tools Appl* 76(14):15341–15355
5. Bajaj R, Chaudhury S (1997) Signature verification using multiple neural classifiers. *Pattern Recogn* 30(1):1–7
6. Baltzakis H, Papamarkos N (2001) A new signature verification technique based on a two-stage neural network classifier. *Eng Appl Artif Intell* 14(1):95–103. [https://doi.org/10.1016/S0952-1976\(00\)00064-6](https://doi.org/10.1016/S0952-1976(00)00064-6).
7. Bouamra W, Djeddi C, Nini B, Diaz M, Siddiqi I (2018) Towards the design of an offline signature verifier based on a small number of genuine samples for training. *Expert Syst Appl* 107:182–195
8. Berkay Yilmaz M, Ozturk K (2018) Hybrid user-independent and user-dependent offline signature verification with a two-channel cnn. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp 526–534
9. Ferrer MA, Vargas JF, Morales A, Ordonez A (2012) Robustness of offline signature verification based on gray level features. *IEEE Trans Inf Forensic Secur* 7(3):966–977. <https://doi.org/10.1109/TIFS.2012.2190281>
10. Fierrez-Aguilar J, Nanni L, Lopez-Penalba J, Ortega-Garcia J, Maltoni D (2005) An on-line signature verification system based on fusion of local and global information. In: Kanade T, Jain A, Ratha NK (eds) *Audio- and video-based biometric person authentication*. Springer, Berlin, pp 523–532
11. Hadjadji B, Chibani Y, Nemmour H (2017) An efficient open system for offline handwritten signature identification based on curvelet transform and one-class principal component analysis. *Neurocomputing* 265:66 <https://doi.org/10.1016/j.neucom.2017.01.108>. <http://www.sciencedirect.com/science/article/pii/S0925231217310159>. New Trends for Pattern Recognition: Theory and Applications
12. Ismail, M.A.; Gad, Samia (Oct 2000). "Offline arabic signature recognition and verification". *Pattern Recognition*. 33(10):1727–1740.