SCNN: Siamese Convolutional Neural Network for Handwritten Signature Verification

Maram A. Alharthi†, Khloud K. Alghamdi†, Samar O. Alosaimi† and Manal A. Alghamdi†

† *College of Computer and Information system, Umm Al-Qura University,* Makkah, KSA

Summary
Signatures have become more appealing in recent years because of its need in daily use including checks for banks, commercial transactions, attendance, and so on. Signature has been also used for identity authentication in many fields. The handwritten signature is known as a behavioral biometric trait of users that will always be different for everyone. Two persons with identical names will usually have different signatures. A signature difference from one person to another is an identification advantage for the person. Thus, it is important to have signature verification systems. Existing systems can be divided into two types; online and offline systems. The offline system uses an image of the signature with a high chance of forging the image from the original one. Focusing on that, this paper employs the Siamese convolutional neural network (SCNN) model to verify the offline signature and detect forgery. By using different kernel sizes, epoch, and learning rates, we outperform the related works with clear margins.

Keywords:
Signature, Verification, CNN, Siamese network, image

1. Introduction

Offline signature is commonly accepted as a bio-metric feature for authenticating individuals and documents, making automated signature authentication is an important research topic in the area of pattern recognition. Signature has been one of the most prominent and widely used authentication method to verify documents, bank checks, people, etc. Verifying offline signatures as a personal function of citizens is seen as one of the bio-metric behavioral variables that today play a significant role in human authentication. Offline signature is a skill that his author specializes in, usually written on paper with an ink pen, a writing pad and electronic pen are not used, so the task of verification is most challenging tasks in contrast to other types of signature authentication such as hands, iris, expression, fingerprints, palm printing or online signature where the forgery can be detected through record the sequence of the electronic pen coordinates while signing, writing speed, calligraphy, pressure, etc.

In this work, we extended the Siamese convolutional neural network (SCNN) model [1] to verify the offline signature and detect forgery. The distinctive in the Siamese neural network is that it does not classify the input image to one class of classes as in the regular CNN network, instead it takes a reference image for the genuine signature corresponding to the input image and the Siamese network compute the degree of similarity between the two images with a value between 0 and 1. One represents that images are (genuine, genuine) and zero represents that images are (genuine, forged). This network learns the similarity function rather than learning to classify a specific image into a specific category. For example, in the event that we have a new customer in the bank, we only need one copy of his signature and save it as a reference image. When obtaining another signature in his name, the network will simply calculate the similarity and make sure the signature is correct.

The structure of the paper is as follows: Section 2 introduces the related works of signature verification. Section 3 highlights summary of the structure of the model used. Section 4, describes our experiments on the work and the dataset used. Section 5 shows results and the discussion of results. Lastly, Section 6 illustrates the conclusion and future works.

2. Literature Review

YILMAZ et al [2] proposed a two-channel CNN model of one image with a hybrid user-independent/dependent offline signature verification technique.
Arenas et al [3] presented a DAY-CAN that aimed to use the writer-independent method to classify and verify off-line signatures of 3 users. Using a private dataset that was built manually, they obtained an overall accuracy of 99.4% and 99.3%.

In 2016 Rantzsch et al [4] proposed signature embedding based on a deeply learned similarity metric to verify writer independent signatures. They used VGG-16 pre-trained model for training and achieved an accuracy of 93.39% for Japanese offline signature, and 81.76% for the Dutch offline signature. Justino et al [5] provided a robust and basic system for verification offline signatures based on hidden markov model (HMM). A code book was

Manuscript received May 5, 2025
Manuscript revised May 20, 2025
https://doi.org/10.22937/IJCSNS.2025.25.5.11

created in the verification phase, by extracting a sequence of feature vectors from the signature. Then HMM was used to calculate the likelihood of the observations given the model. After computing the likelihood, a simple threshold was used to find genuine signatures and forgeries signatures.

In another study, Yapici et al [6] proposed a deep learning (DL) method to prevent signature fraud. The deep learning method that used in their work was based on previous work by LeCun et al [7]. They trained two separate CNN for writer dependent (WD) and writer independent (WI). They used GPDS synthetic Signature dataset that consists of 4000 signatures of various persons. The WD model achieved 75% of success, while the WI achieved 62.5% of success. Another study [8] provides a comparison between various two different pre-trained CNN architecture, VGG and AlexNe, which are performed remarkably well and used widely in Computer Vision problems. Using the GPDS-160 dataset, they obtained an Equal Error Rate of 2.74%.

3. Methodology

This paper extended the Siamese convolutional neural network (SCNN) that have been previously used by Dey et al [1] for offline signature verification. In that paper, the model contains four convolution layers with different kernel sizes including 11 × 11, 5 × 5 and 3 × 3. The convolution layers and fully connected layers are followed by ReLU activation function. The filters N × H × W, where N represents a number of the kernel.
Their convolution layers are:

- The first convolution layer: take the input image with fixed size (155,200) and use filter 96 × 11 × 11 and one pixel as stride. Then apply the normalization for each image in the Local Response Normalization layer and apply Max pooling.
- The second convolution layer: take the input image after normalized and pooled with filter 256 × 5 × 5. Select one pixel as stride and two for padding.

The third and fourth convolution layers of consecutive 384 × 3 × 3 and 265 × 3 × 3.

Also, their model included two fully connected layers and three pooling layers. They used Dropout in the last two pooling layers and the first fully connected layer. The model has two networks with the same architecture and parameters are shared. At the end, a fully connected layer is used to define the output. They used Euclidean distance in similarity metric and used contrastive loss to compute the loss.

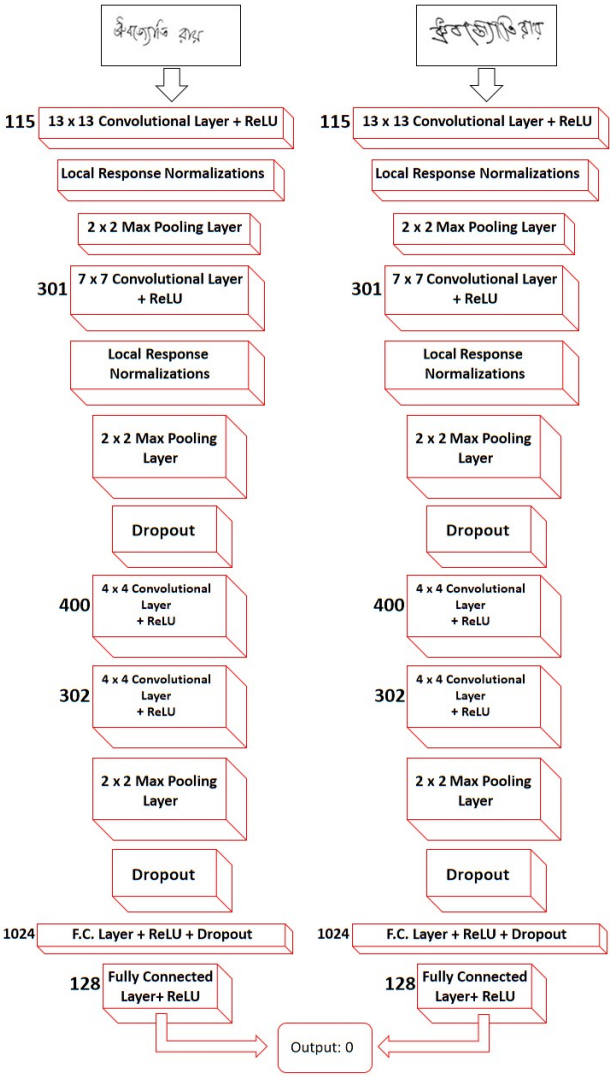

Fig. 1 Architecture of our Siamese convolutional neural network (SCNN) model for offline signature verification.

4. Experiment

4.1 Dataset

The Bengali dataset consists of 100 signers. There are 24 genuine and 30 forged signatures for each signer that means 2,400 genuine and 3,000 forged signatures. The Bengali dataset is a subset of the BHSig260[1] dataset and it can be used separately. To label the dataset, we divided it manually into two folders. One folder contains the genuine images other contains the forged images. It can be

[1] https://drive.google.com/file/d/0B29vNACcjvzVc1RfVkg5dUh2b1E/view

accessed from the link[2].

4.2 Preprocessing

For training the neural network, images should have an equal size which is not the case in the used dataset. So, we resize all the images to have a fixed size of 155 × 220.

4.3 Our experiments

The dataset contains 5400 images, which was divided into training and testing sets as follows. For generating a training file with genuine-forged signature pairs, we selected randomly number, then choose a randomly genuine signature with another randomly genuine signature for same person and take a label is 1. Simultaneously to achieve balance in the training file, choose a randomly genuine signature with a randomly forged signature for same person and take a label is 0. After that, selected 0.15 from training file for testing. We adopted all values of parameters in SigNet model [1], but we tried to change the value of the learning rate (lr), they adopted 1e-4. The main change in our work is change kernel size of filters which are shown in table 1. Their model used the RMSprop, while we used tried both RMSprop and Adam.

Table 1: The main changes in our work.

	SigNet	Our experiment
First filter	96 × 11 × 11	115 x 13 x 13
Second filter	256 × 5 × 5	301 × 7 × 7
Third filter	384 × 3 × 3	400 × 4 × 4
Forth filter	256 × 3 × 3	302 × 3 × 3

5. Result and Discussion

We compared our results with the SigNet[1], which achieved 0.86 in Bengali dataset with 128 batch-size and 20 epoch when the value of learning rate was 1e-1.

[2]https://www.dropbox.com/s/7ma4qa5882z2cjc/Bengali.rar?dl=0

We focus on this dataset to achieve higher accuracy by extending their model with our methodology.
We trained our model using 30, 50, 70 and 100 epoch with 32 and 64 batch-size.
Using their lr value, 32 batch-size and train 100 epoch, we obtained better accuracy and loss, with 0.988 accuracy and 0.008 loss with RMSprop. While using Adam optimizer achieved 0.996 accuracy and 0.008 loss.
By changing the value of lr to 1e-3, we obtained a higher accuracy of 1.0 and less loss of 7e-7 when trained 100 epoch and 32 batch-size with RMSprop (see Figure 2).
Changing the lr to 1e-2 causes the accuracy to decrease.
Table 2 shows detailed results of our experiments using lr= 1e-3 with 32 batch-size. As shown in the table, Adam optimizer achieved worse results that RMSprop.

Table 1: Our experiments when lr is 1e-3 with 32 batch-size.

	Adam	***RMSprop***
30 epoch	Accuracy: 0.816	Accuracy: 0.953
	loss: 0.107	loss: 0.050
50 epoch	Accuracy: 0.613	Accuracy: 0.996
	loss: 0.226	loss: 0.002
70 epoch	Accuracy: 0.839	Accuracy: 0.996
	loss: 0.069	loss: 0.001
100 epoch	Accuracy: 0.984	Accuracy: 1.0
	loss: 0.002	loss: 7e-07

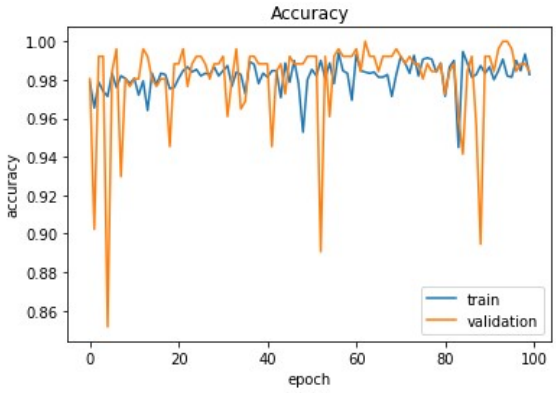

Fig. 2 Train & validation accuracy in 32 batch size with 100

epoch, RMSprop and lr is 1e-3.

6. Conclusions

In this paper, we extended the SigNet model [1] with changes in the architectural to improve the quality of the Siamese convolutional neural network (SCNN) in validating the signature and detecting forgery. The accuracy reached to 1.0 with the least possible loss when we experiment 100 epoch, RMSprop, lr is 1e-3 and 32 batch-size. In future work we plan to use various datasets with advanced convolutional neural networks. We want to run this structure on other images such as fingerprint and iris images.

References

[1] S. Dey, A. Dutta, J. I. Toledo, S. K. Ghosh, J. Lladós, and U. Pal, "SigNet: Convolutional siamese network for writer independent offline signature verification," *arXiv preprint arXiv:1707.02131*, 2017.

[2] M. Berkay Yilmaz and K. Ozturk, "Hybrid user-independent and user-dependent offline signature verification with a two-channel CNN," in *Proceedings of the IEEE Conference on Computer Vision and PatternRecognition Workshops*, pp. 526–534, 2018.

[3] [3] J. O. Pinzón-Arenas, R. Jimenez-Moreno, and C. G. Pachón-Suescun, "Offline signature verification using DAG-CNN.," *International Journal of Electrical & Computer Engineering (2088-8708)*, vol. 9, 2019.

[4] H. Rantzsch, H. Yang, and C. Meinel, "Signature embedding: Writer independent offline signature verification with deep metric learning," in *International symposium on visual computing*, pp. 616–625, Springer, 2016.

[5] E. J. Justino, A. El Yacoubi, F. Bortolozzi, and R. Sabourin, "An off-line signature verification system using HMM and graphometric features," in *Proc. of the 4th international workshop on document analysis systems*, pp. 211–222, Citeseer, 2000.

[6] M. M. Yapici, A. Tekerek, and N. Topaloglu, "Convolutional neural network based offline signature verification application," in *2018International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT)*, pp. 30–34, IEEE, 2018.

[7] Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in neural information processing systems*, pp. 396–404, 1990.

[8] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, "Analyzing features learned for offline signature verification using Deep CNNs," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 2989–2994, IEEE, 2016.

Maram A. Alharthi received the Bachelor's degree in computer science from the Computer Science Department, Taif University, Saudi Arabia. She is currently doing a Master's degree in Artificial Intelligence at Umm Al-Qura University. Her areas of scientific interests are machine learning, deep learning and image processing.

Khloud K. Alghamdi received the Bachelor's degree in computer science from the Computer Science Department, Al-Baha University, Saudi Arabia. She is currently doing a Master's degree in Artificial Intelligence at Umm Al-Qura University. Her areas of scientific interests are image processing and NLP.

Samar O. Alosaimi received the Bachelor's degree in computer science from the Computer Science Department, Taif University, Saudi Arabia. She is currently doing a Master's degree in Artificial Intelligence at Umm Al-Qura University. Her areas of scientific interests are machine learning and deep learning.

Manal A. Alghamdi an assistant professor at the department of computer science, UQU, Saudi Arabia.