An Offline Signature Verification leveraging One-shot Learning

Usama Ejaz, Muneeb A. Khan, Heemin Park, Hyun-chul Kim
Department of Software, Sangmyung University, South Korea

Abstract

Offline signature verification is one of the most crucial tasks in biometrics and document authentication. Recently, multiple deep learning approaches have been proposed to authenticate the signature, but usually, they need a large training dataset. In this paper, we presented a deep learning model based on one-shot learning to verify the authenticity of offline signatures. To verify the capability of our model, we use cross-domain datasets with different languages. Furthermore, our designed model outperforms many existing models in terms of validation loss as 0.03 on most of the benchmark datasets.

1. INTRODUCTION

A signature is one of the most significant acceptable biometric trademarks to verify transactions in banking. There are two modes of signatures, such as online and offline. In the case of online signature, numerous artifacts involve in authenticating it. For instance, pen pressure, angle, speed, slope, and etc. While offline signature verification categorizes into writer-depended and writer-independent approaches [1, 2]. In the writer-dependant technique, an individual model needs to train a model for a single author. However, for the writer-independent approach, a single model is required to classify the genuine signature of all authors.

Deep Neural Networks (DNN) have been exploited in various fields and are often used to solve classification problems. It is handy to extract multiple features from different styles of handwriting and identity the forgery and genuine signature. Many machine learning algorithms have been presented for signature verification [3]. Usually, these models cost high computation resources, training time, and enormous data to train an efficient classifier to verify signatures with high accuracy. However, Siamese networks based on One-Shot learning, perform efficiently in the classification tasks, such as face verification [4] and signature verification [5], etc.

The Siamese Neural Network (SNN) comprises two Convolutional Neural Networks (CNN) with the same parameters and weights. To train the SNN, two different inputs are given to the model with the Euclidean distance function, which converts the images into the features embedding vectors and calculates the distance between the inputs images. Our work presents a signature verification model, which computes the proximity between the two sample input signatures with minimum loss and identifies it as genuine or forgery signatures accordingly.

The rest of the paper is arranged as follows: In Section 2, we describe the proposed methodology, dataset, and preprocessing of data. In Section 3, we illustrate the results, and the conclusion is presented in Section 4.

2. METHODOLOGY

In this section, we described the proposed model with workflow. We used SNN as a double CNN sub-network with equal weights, different input sample signatures, robust vector embedding, and contrastive loss function. We choose SNN because it works efficiently with less data, imbalanced classes, and detect forgery signatures with just a couple of samples. Furthermore, it is useful to encode specific features.

2.1 DATASETS

We performed our experiments on cross-domain challenging datasets, such as BHSig260 and Signature Verification Dataset (SVD). The BHSig260 datasets contain 160 contributors where 24 genuine and 30 forged. In the SVD dataset, 69 individuals participate and equally provide forged and genuine signature samples. In SVD, we opted 1980 samples for training while 504 samples for testing purposes with 80:20 ratio. The proposed model is also trained and tested on BGSig260 dataset with 6912 training sample while 1728 testing size, respectively. The training and testing samples are selected randomly to maintain unbiasedness and cross-validate the efficacy of the model.

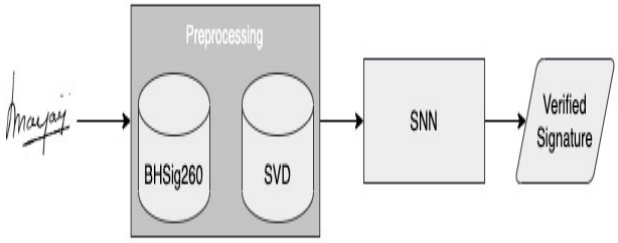

Figure 1. Workflow of Signature Verification

2.2 CNN AND SNN

CNN is the widely used and the most prominent deep learning architecture in computer vision. It comprises three layers. In the case of the convolution layer, it extracts the features from the input image. The non-linear layer maps the features and facilitates the model to adopt the non-linear function. The max-pooling layer lessens the spatial resolution by shifting the neighboring feature map with neighboring information (i.e., min, max, and average). Locally, each neuron in every layer is interconnected, thus forming fully connected layers [6].

SNN designes with two identical CNN sub-networks which share the same weights and parameters. The contrastive loss function computes the similarities or dissimilarities between the given pair vector by using Euclidean distance metrics. The main objective of SNN is to calculate the distance between two vectors instead of classifying them. Thus, the cross-entropy function is not suitable for this problem. The contrastive loss function weight up the network and how two images can be discriminated efficiently (as shown in Figure 1). The contrastive loss function defines as:

$$\mathrm{L(k,x,y)} = (1-k)\frac{1}{2}(D_w)^2 + (k)\frac{1}{2}\{\max(0, m - D_w)\}^2 \quad (1)$$

Where **k** shows whether both signature samples refer to the same class or not, while **x** and **y** are two signature input samples, $\mathbf{D_w}$ is the distance between both input samples formulated into the vector space, and m is the margin.

We formulate two SNN models which consist of simple SNN with three convolution layers with conventional CNN settings while extended version model consists of four convolution layers and applies local response normalization to normalize the local input regions. There is a max-pooling layer of 3 by 3 by following the convolution layer. We constructed the SNN with a combination of layers with optimized parameters. The overall proposed system architecture is shown in Figure 2.

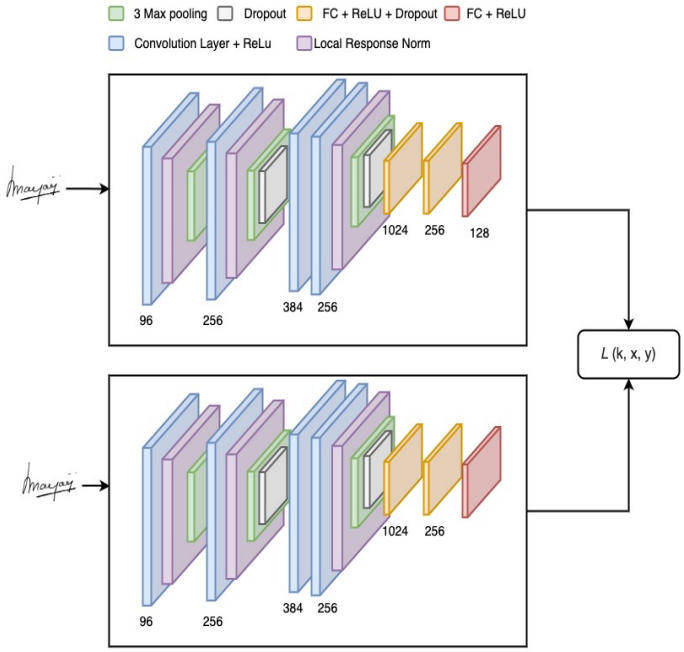

Figure 2. Architecture of Proposed Model

3. RESULTS AND DISCUSSION

The results are evaluated based on mainly training loss, validation loss, training time, consumption of resources, such as computational cost. Furthermore, we presented comparison of our models with the state-of-the-art models in Table1.

Table1: Comparison of proposed methodology with various state-of-the-art methods.

Sr#	Methods	Accuracy	Loss
1.	Pal *et al.* [7]	0.75	0.245
2.	Kumar *et al.*[8]	0.90	0.120
3.	Simple SNN	0.94	0.059
4.	**SNN-Extended**	0.97	0.030

3.1 VALIDATION LOSS

Accuracy and loss are the major metrics to evaluate the model performance. Usually, previous researchers focus on accuracy of the model to measure the metrics of signature verification task.

The experiments are performed using NVidia K80 GPU. Furthermore, we consider validation loss would be more reasonable than accuracy to counter the issue of model overfitting. Our model achieved the minimum training loss of 0.04; the validation loss remains 0.03 with only 30 epochs. Our model takes 42.6 seconds per epoch to validate the training accuracy (as shown in Figure 3).

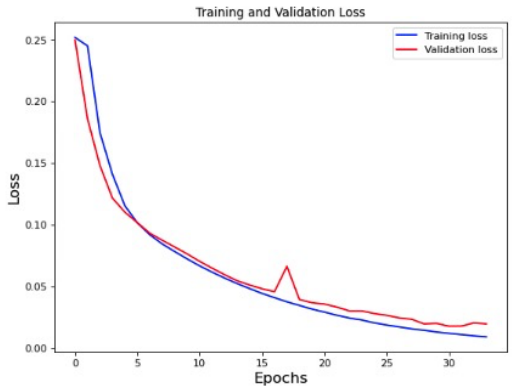

Figure 3 Training and validation loss at epochs 30

4. CONCLUSION

Our proposed methodologies used the SNN network to verify offline signatures with small-scale cross-domain datasets using One-Shot learning. The main task of the model is to identify the genuine and forgery signatures. We designed our model in such a way as it lessens the training time learned from the previous epoch. In future work, we consider using adversarial attacked datasets instead of the clean dataset to make an efficient classifier and use it as a real-world application.

REFERENCE

[1] Bertolini, D., Oliveira, L. S., Justino, E., & Sabourin, R. (2010). Reducing forgeries in writer-independent off-line signature verification through ensemble of classifiers. Pattern

[2] Manjunatha, K. S., Manjunath, S., Guru, D. S., & Somashekara, M. T. (2016). Online signature verification based on writer-dependent features and classifiers. Pattern Recognition Letters, 80, 129-136.

[3] Impedovo, D., & Pirlo, G. (2018). Automatic signature verification in the mobile cloud scenario: survey and way ahead. IEEE Transactions on Emerging Topics in Computing, 9(1), 554-568.

[4] Wang, W., Yang, J., Xiao, J., Li, S., & Zhou, D. (2014, November). Face recognition based on deep learning. In International Conference on Human-Centered Computing (pp. 812-820). Springer, Cham.

[5] Jagtap, A. B., Sawat, D. D., Hegadi, R. S., & Hegadi, R. S. (2020). Verification of genuine and forged offline signatures using Siamese Neural Network (SNN). *Multimedia Tools and Applications*, *79*(47), 35109-35123.

[6] Minaee, S., Boykov, Y. Y., Porikli, F., Plaza, A. J., Kehtarnavaz, N., & Terzopoulos, D. (2021). Image segmentation using deep learning: A survey. IEEE transactions on pattern analysis and machine intelligence.

[7] S. Pal, A. Alaei, U. Pal, M. Blumenstein, Performance of an off-line signature verification method based on texture features on a large indic-script signature dataset, in: DAS, 2016, pp. 72–77.

[8] R. Kumar, J. Sharma, B. Chanda, Writer-independent off-line signature verification using surroundedness feature, PRL 33 (3) (2012) 301–308.