A Novel Architecture to verify Offline Hand-written Signatures using Convolutional Neural Network

Sultan Alkaabi
Institute of Informatics and Computing in Energy
Universiti Tenaga Nasional
Malaysia
PT20740@utn.edu.my

Salman Yussof
Institute of Informatics and Computing in Energy
Universiti Tenaga Nasional
Malaysia
Salman@uniten.edu.m

Sameera Almulla
College of Engineering and IT
University of Dubai
UAE
Salmulla@ud.ac.ae

Haider Al-Khateeb
Wolverhampton Cyber Research Institute (WCRI)
University of Wolverhampton
UK
H.Al-Khateeb@wlv.ac.uk

Abdulrahman AAlAbdulsalam
Dept of Information Technology
Colleges of Applied Sciences (CAS)
Oman
abdulrahmna.k.rus@cas.edu.om

***Abstract*— Hand-written signatures are marked on documents to establish legally binding evidence of identity and intent. However, they are prone to forgery, and the design of an accurate feature extractor to distinguish between highly-skilled forgeries and genuine signatures is a challenging task. In this paper, we propose a Convolution Neural Network (CNN) architecture for Signature Verification (SV). The algorithm is trained using two signatures, genuine and forged. Then the SV module performs a classification task to determine if any two signatures are of the same individual or not. The simulation results show that the proposed method can achieve 27% (relatively) better results than the benchmark scheme. The paper also integrated different data augmentation techniques for the signature data, which further improved the efficiency of the proposed method by 14% (relative).**

Keywords*— *Handwritten Signature, authentication, verification, Convolutional neural networks, forensics, signature verification, data augmentation

I. INTRODUCTION

Biometrics have long been used for authentication purposes with practical applications mainly relying on human physiological traits related to specific measurements and characteristics of the body such as fingerprints. In this study, however, we investigate signature recognition; behavioral biometrics that identifies individuals based on their hand-written text. More precisely we focus on static signatures already written on a paper and digitalized at a later time [1].

The need to perform and automate Hand-written Character Recognition (HCR) is increasing. HCR is a process in which hand-written images are received by the machine to interpret information from sources including photographs, images, and touch devices [2-3]. HCR is of two types: Online and offline. The online method transforms the strokes of the digital pen to an array of coordinates, whereas the offline method uses scanned characters as input images. The latter type of verification is used in offline personal verification and typically performed by banks or as part of a forensic process. There is an increasing demand for automated offline Signature Verification (SV) instead of manual verification to avoid human errors and to save time.

Offline HCR is considered a relatively more challenging task, in comparison with online HCR, since the captures features are limited to what has already been written excluding all real-time behavioral characteristics. Additionally, the variations in the writing patterns of the individual [4, 5] add to the problem. Therefore, offline HCR is arguably more susceptible to forgery. Although, it has many critical applications in daily life.

Several techniques have been proposed in the literature to improve the accuracy of offline HCR [6-7]. These methods could be divided broadly into two categories. In [8], the researchers explored several signal processing approaches such as pixel comparison [8], chord moment method [9] and a best feature selection approach [9] all of which performed for the purpose of verifying hand-written signatures. However, the signal processing technique was found to be inefficient in terms of features extraction due to the position and alignment of signatures in the images. To solve the issue of position and alignment, the researchers also explored gradient and projecting features [10] and Gabor filter for feature extraction [11], nevertheless, the accuracy in the verification of signatures was not improved.

Related work shows evidence that data-driven approaches including Machine Learning and Deep Learning have played an essential role in improving the accuracy of hand-written signatures [12,13]. For example, in [12, 14], the researchers surveyed the machine learning and deep learning techniques and concluded that CNN have better accuracy in terms of verifying the signatures in comparison to other neural network techniques.

Our proposed work is novel in terms of better learning of non-linearity in the datasets. CNNs are used for learning the features of the datasets. Unlike other approaches [15], the algorithms are trained with the original and forged data sets in parallel. Hence, the features are well learned and trained. The simulation results verify the efficiency of the proposed algorithm in terms of Equal Error Rate (EER).

To evaluate the proposed technique, a benchmark paper [14] is used for evaluation. It uses a genetic algorithm to handle the feature extraction of hand-written signatures.

978-1-7281-3012-5/19/$31.00 ©2019 IEEE

However, learning the non-linear features can be considered as one of the limitations. When the datasets have non-linearity, the results of the benchmark scheme are not satisfactory.

Therefore, in this paper, to overcome the limitations mentioned above, we emphasize the following contributions;

- To propose a novel CNN based architecture for SV compared to the existing schemes, in which the signatures are classified to different classes using neural networks.
- The proposed method tests a new approach in which it compares and trains two signatures (genuine and skilled forgeries) at a time.
- The results of the proposed CNN implementation the Keras library [16] will outperform the benchmark scheme in terms of EER.

The remainder of this paper is structured as follows; the proposed methodology is discussed in section II. Section III explains the standard datasets used for the evaluation of the algorithm. The simulation parameters are explained in section IV. Section V discusses the results of the proposed scheme compared to the benchmark scheme. Finally, we conclude our study in section VI.

II. The Proposed Architecture Overview

The proposed architecture trains the algorithm using two signatures, genuine and forged. The SV module performs a classification task to determine if any two signatures are of the same individual or not. If a signature is not verified, it will be rejected as a fraudulent imitation of the original. Our proposal acknowledges that false-positives can occur and the user will, therefore, be given the opportunity to use a contextual password as an alternate mechanism to authenticate. The metrics used to evaluate the proposed method are ERR and percentage accuracy.

The architecture of CNNs consists of two main parts: *Feature Extraction* and *Classification*. The feature extraction layer receives the data as input from previous layers and passes the features extracted from this layer to the next layer. The block diagram of the proposed SV system is shown in Fig 1. Each of the blocks is explained in detail below.

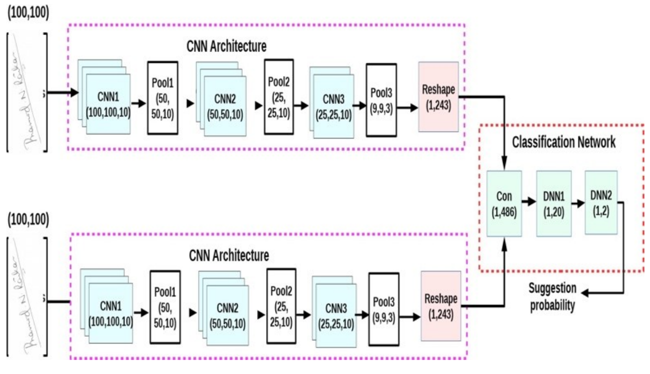

Figure 1: The proposed CNN-based Architecture

A. Feature Extraction

At first, a given signature image is converted to a binary image (digital image that has only two possible values for each pixel) by using a threshold α, which helps in handling the noise of the signature image. Thereafter, these images are resized to 100x100 to maintain a unified size for the features. Finally, these preprocessed images are passed through the network for SV.

Given the two signature images, we passed these images through the CNN's architecture as shown in the block diagram, which converts the 100x100 binary images into a single 243-dimensional vector. We worked with the assumption that vectors from the two images share similarities as long as the signatures have been originated by the same person. Hence, we have concatenated these two vectors of 243 dimensions to make a single 486-dimensional vector, which we have then passed through the classification network to obtain the final decision.

B. The Three Layer CNN Architecture

The proposed architecture consists of three CNNs stages namely convolution, max pooling and classification layers as discussed below;

- Convolution layer

This layer is responsible for convolving the feature maps of previous layers with kernels such as (Gabor or Gaussian). The convolved output of the kernel is passed through the activation functions such as (hyperbolic tangent, sigmoid, softmax, and rectified linear functions. The convolution layer can be mathematically represented as;

$$y_j^{a} = f\left(\sum_{i \, \mathcal{E} \, M_j} x_i^{t-1} k^t_{ij} + b^t_j\right) \qquad 1$$

In equation 1, y_j^a represents the output of the current layer that is dependent on the previous layers and represented as x_i^{t-1}. In which k^t_{ij} represents the kernel for the existing layer and b^t_j shows the bias value of the current layer. M_j is the value of selection of input maps. The value of M_j is convolved with kernel values for generating feature maps.

- Sampling layer

The operation of this layer is to downsample the input maps. The characteristics of the input and output layer remain the same in this layer. The size of the output map reduces due to down-sampling operation using down value in equation 2, depending on the mask of downsampling. Mathematically it can be represented as;

$$Y^{t}{}_{j} = f(\beta^t_j down(x_j^{t-1)} + b^t_j) \qquad 2$$

The down operation is responsible for subsampling function that sums up over n x n block of the maps and uses the highest values from n x n blocks. The output map dimension is reduced n times and it is passed through non-linear or linear activation functions.

C. The CNNs Architecture

The architecture of each CNNs is discussed below;

- *CNN1: First CNN layer architecture*

The functionality of the first layer of CNN is to reconstruct the hand-written input signatures. The first layer in CNN is designed such that the image of the signature is reconstructed efficiently. The CNN1 architecture consists of 13 convolutional layers with two upsampling and two max-pooling layers. The convolution layers in CNN1 uses a 2 x 2 kernel and a 2 x 2 max-pooling layers. The feature map is reduced to 50 x 50 x 10 using two max-pooling layers. The ReLU activation function is used in the convolution layers.

- *CNN2: Second CNN layer architecture*

The architecture of CNN2 consists of 2 x 2 kernels to add more non-linearity. The 2 x 2 max-pooling layer is used to further reduce the feature map to 2 x 2. The max-pooling layer further reduces the size of the image to (25, 25, 10) The ReLu activation function is used with the convolution layer.

- *CNN3: Third CNN layer architecture*

The CNN3 has a kernel shape of (3 x 3), and the output of the image is further reduced to (9, 9, 3). The output of the CNN3 is reshaped to (1,243) which is the dimensional vector for each signature. The sigmoid activation function is used as an activation function in this layer.

D. *Classification*

In the classification network, we convert the final 486-dimensional vector, which is obtained by concatenating the outputs from both CNNs architecture networks into a 2-dimensional vector of binary values. The 486-dimensional vector is first passed through a Deep Neural Network (DNN1) with the Relu activation to obtain a 20-dimensional vector. Then this 20-dimensional vector is passed through DNN2 with a Softmax activation function that represents the probability of authentication. The Softmax activation function is used to calculate the probability distribution of different events. This function calculates the probability of a specific class from different possible target classes.

III. Datasets

We considered two open access databases for the experiments. The first one is the CEDAR dataset [16], which consists of 55 signatures in total, including forged and genuine signatures from each of the signers. We considered 40 signer's data for training and the remaining 15 for the testing. The second dataset considered for the experiments is the NISDCC signature collection of the ICDAR 2009 online SV competition [17]. This dataset consisted of 60 authentic signatures, which are written by 12 authors. A total of 31 forgers produced forgeries for all the signatures with a ratio of 1 genuine to 5 forgeries.

To validate our results and as a benchmark, we considered the method proposed by [14] as the baseline scheme. In this method, the authors proposed a novel feature extraction method, which captures both dimensions of the signature and geometry. The benchmark scheme uses a genetic algorithm for extracting appropriate features sets and support vector machine-based classifier, is used for verification of the signature.

IV. Simulation Parameters

To evaluate the performance of the proposed network compared with the baseline scheme, we considered two experiments. In the first one, we used different signatures from different persons to train the network. In the second case, we used the genuine and forged signature to continue the training. For the testing phase, we considered two cases, (i) with different signatures (SV, (ii) with genuine and forged signatures (FS).

Table 1. EER value of the proposed scheme and benchmark scheme using trained and untrained datasets

EER(%)	Using trained dataset		Using different dataset	
Method	Benchmark Scheme (BS)	Proposed scheme	BS	Proposed
FS	11.2	8.7	13.4	9.1
SV	4.2	2.7	7.1	3.04

In order to observe database variability, we also considered two cases, which are (1) Within database (Training and testing are performed on the same database), (2) Across database (Training and testing are performed on the different databases). Equal Error Rate (EER) is used as the performance metrics, which is the error rate at which false acceptance rate and the false rejection rate are equal.

V. Results

As explained in the experimental setup, we computed EER for all the combinations as shown in Table 1. From the table, we can observe that the proposed method performs better than the baseline in all four cases, and in the best case scenario, it is a ~57 % (relative) improvement over the baseline. The drawback of the benchmark scheme is that it cannot learn the features of the signatures accurately. We can also observe that in both cases, EER for the FS case is higher than the normal SV case. The proposed method is performing best compared to the state-of-the-art scheme due to using CNNs that learn the features of the data accurately.

Table 2 also shows that the EER in the case of using different datasets is substantially more than the same database condition. It is interesting to see how the proposed method performs significantly better than the baseline in the case of a new test data condition, which makes it good for many applications.

Table 2. EER Comparison of the proposed method with and without Data Augmentation (DA) for different test conditions

EER(%)	Using trained datasets		Using untrained datasets	
Proposed method	without DA	with DA	without DA	with DA
FS	8.7	7.1	9.1	8.4
SV	2.7	2.42	3.04	2.91

A. *Data Augmentation (DA)*

To improve the robustness of the signature verification system towards the alignment and position of the signature, we considered the following data augmentation method. Each image of the signature is multi-folded to 12 times by rotating around the center of the image. The signature verification system is trained using the new data, where the rotated image version is used as a positive case and forged signature as a negative case. Table 2 shows the EER values of the proposed method with and without data augmentation. We can see that this kind of augmentation improves system performance. Table 2 also shows that DA is more effective in the case of FS compared to standard signature verification

VI. CONCLUSION

In this paper, we proposed a novel convolutional neural network architecture for signature verification. Experiments conducted at different testing conditions using two databases revealed that the proposed architecture is better than the selected baseline schemes in all testing conditions by at least ~27 % (relative). We have also showed that the proposed method offered substantial improvement when using different datasets for training/testing. Furthermore, we explored two types of data augmentation methods, which further improved the system performance. Feature work in this area could explore different generative models for SV.

REFERENCES

[1] Ferrer, M.A., Diaz, M., Carmona-Duarte, C. and Morales, A., 2016. A behavioral handwriting model for static and dynamic signature synthesis. IEEE transactions on pattern analysis and machine intelligence, 39(6), pp.1041-1053.

[2] Rani, N. S., Chandan, N., Jain, A. S., & Kiran, H. R. (2018). Deformed character recognition using convolutional neural networks. International Journal of Engineering & Technology, 7(3), 1599-1604.

[3] Jain, A., & Sharma, B. K. (2018). Analysis of activation functions for Convolutional Neural Network based mnist handwritten character recognition. International journal of advanced studies of scientific research, 3(9).

[4] D'souza, L., & Mascarenhas, M. (2018, August). Offline handwritten mathematical expression recognition using Convolutional Neural Network. In 2018 International Conference on Information, Communication, Engineering and Technology (ICICET) (pp. 1-3). IEEE.

[5] Enriquez, E. A., Gordillo, N., Bergasa, L. M., Romera, E., & Huélamo, C. G. (2018, November). Convolutional Neural Network vs traditional methods for Offline Recognition of Handwritten Digits. In Workshop of Physical Agents (pp. 87-99). Springer, Cham.

[6] Firdaus, S. A., & Vaidehi, K. (2020). Handwritten Mathematical Symbol Recognition Using Machine Learning Techniques. In Advances in Decision Sciences, Image Processing, Security and Computer Vision (pp. 658-671). Springer, Cham.

[7] Kumar, M., Jindal, M. K., Sharma, R. K., & RaniJindal, S. (2018, August). Performance Comparison of Several Feature Selection Techniques for Offline Handwritten Character Recognition. In 2018 International Conference on Research in Intelligent and Computing in Engineering (RICE) (pp. 1-6). IEEE.

[8] Indrajit Bhattacharya, Prabir Ghosh, and Swarup Biswas, "Offline signature verification using pixel matching technique," Procedia Technology, vol. 10, pp. 970–977, 2013.

[9] Medam Manoj Kumar and Niladri Bihari Puhan, "Off-line signature verification: upper and lower envelope shape analysis using chord moments," IET Biometrics, vol. 3, no. 4, pp. 347– 354, 2014

[10] KS Radhika and S Gopika, "Online and offline signature verification: a combined approach," Procedia Computer Science, vol. 46, pp. 1593–1600, 2015

[11] Sanghamitra Das and Abhinab Roy, "Signature verification using rough set theory based feature selection," in Computational Intelligence in Data MiningVolume 2, pp. 153–161. Springer, 2016.

[12] Naz, S., Umar, A. I., Ahmed, S. B., Ahmad, R., Shirazi, S. H., Razzak, M. I., & Zaman, A. (2018). Statistical features extraction for character recognition using recurrent neural network. Pakistan Journal of Statistics, 34(1).

[13] Elhoseny, M., Nabil, A., Hassanien, A. E., & Oliva, D. (2018). Hybrid rough neural network model for signature recognition. In Advances in Soft Computing and Machine Learning in Image Processing (pp. 295-318). Springer, Cham.

[14] Muhammad Sharif, Muhammad Attique Khan, Muhammad Faisal, Mussarat Yasmin, and Steven Lawrence Fernandes, "A framework for offline signature verification system: Best features selection approach," Pattern Recognition Letters, 2018.

[15] W. Alabbas, H. M. Al-Khateeb, and A. Mansour, "Arabic text classification methods: Systematic literature review of primary studies", in 2016 4th IEEE International Colloquium on Information Science and Technology (CiSt), Tangier: IEEE, 24-26 Oct 2016, pp. 361–367. DOI: 10.1109/CIST.2016.7805072. [Online]. Available: https://doi.org/10.1109/CIST.2016.7805072.

[16] keras.io, "Keras: The Python Deep Learning library," [Online]. Available: https://keras.io/. [Accessed 30 07 2019].

[17] Blankers, V. L., van den Heuvel, C. E., Franke, K. Y., & Vuurpijl, L. G. (2009, July). Icdar 2009 signature verification competition. In 2009 10th International Conference on Document Analysis and Recognition (pp. 1403-1407). IEEE.