Writer Independent Signature Verification (WISV) system using SNN neural network

Amirreza Radmanesh, Amirmohammad Taghizadehgan
*[1]Computer Engineering, Yazd University, Yazd, Yazd Province, Iran

ABSTRACT

Signature verification is a critical component in secure systems, ensuring the authenticity of signatures used in various applications, such as financial transactions and legal documentation. This project focuses on developing a signature verification system using the UTSig dataset, which includes pairs of genuine and forged signatures. A Siamese Neural Network (SNN) was employed to train the model for effective classification of these signature pairs. [1] The proposed SNN architecture features convolutional and dense layers to extract and analyze signature features. The system was evaluated on its accuracy and loss, achieving an impressive accuracy of 97.77% and a loss of 0.308. These results demonstrate the effectiveness of the model in distinguishing between genuine and forged signatures. The project highlights the potential of SNNs in signature verification tasks and sets the foundation for future enhancements, such as dataset expansion and the exploration of advanced neural network architectures.

Keywords: Signature Forgery Detection, Signature Verification, Siamese Neural Network (SNN), UTSig Dataset

I. INTRODUCTION

Signature verification is a vital aspect of authentication and fraud prevention in various sectors, including banking, legal documentation, and personal identification. The ability to accurately distinguish between genuine and forged signatures is crucial for maintaining the integrity of financial transactions and legal agreements. As signatures are unique to individuals and often used for high-stakes validation, effective verification systems must ensure both high accuracy and reliability.

Traditionally, signature verification methods have relied on manual inspection and rule-based systems. However, these approaches often fall short in terms of scalability and adaptability, especially when dealing with large volumes of signatures or varying writing styles. To address these challenges, modern systems increasingly leverage machine learning and artificial intelligence [2] techniques, which offer more robust and scalable solutions.

In this project, we focus on developing an advanced signature verification system using the UTSig dataset. The UTSig dataset provides a comprehensive collection of signature samples, including both genuine and forged signatures, making it an ideal resource for training and evaluating verification models. This dataset includes multiple signature pairs, each labelled as either genuine or forged, which allows for a detailed analysis of the model's performance in distinguishing between these categories.

To enhance the performance of our verification system, we employed a Siamese Neural Network (SNN). [3] SNNs are inspired by the neural processes of the human brain and are designed to model the temporal dynamics of input data. Unlike traditional neural networks, which process data in a continuous manner, SNNs handle data as discrete spikes or events. This temporal processing capability makes SNNs particularly well-suited for tasks involving sequential or time-dependent data, such as signature verification. [4]

The SNN architecture [5] used in this project includes several key components:

- Convolutional Layers: These layers are responsible for extracting features from the signature images. By applying convolutional operations, the model can capture intricate patterns and details in the signatures that are crucial for accurate verification.
- Max Pooling Layers: These layers reduce the dimensionality of the feature maps while retaining essential features. This process helps in minimizing computational complexity and mitigating overfitting.
- Dense Layers: After feature extraction, dense layers are used to further process the features and make final predictions. These layers combine the extracted features to classify signature pairs as either genuine or forged.

The model's performance is evaluated using two primary metrics: accuracy and loss. Accuracy measures the proportion of correctly classified signature pairs, while loss quantifies the model's error in making predictions. These metrics provide insight into the model's effectiveness and guide iterative improvements.

By leveraging the UTSig dataset and the advanced capabilities of SNNs, this project aims to develop a highly accurate and reliable signature verification system. The results of this study have significant implications for enhancing security measures in various applications where signature authentication is critical.

II. Methodology

2.1 Data Collection

The UTSig dataset is utilized for this signature verification project, providing a diverse collection of signature pairs, each labelled as either genuine or forged. This dataset is instrumental for training and evaluating the signature verification model, as it includes a variety of signatures from different individuals and writing styles. To ensure effective training and evaluation, the dataset was divided into training, validation, and test subsets.

2.2 Pre-processing

The pre-processing steps were critical in preparing the data for model training. The key pre-processing activities included:

Pair Creation: Signature pairs were generated to facilitate the training of the model. Each signature was paired with another randomly selected signature from the dataset. Each pair was labelled as "1" if the signatures are from the same individual (genuine) and "0" if they are from different individuals (forged). This process helps the model learn to differentiate between genuine and forged signatures.

Distance Calculation: To assess the similarity between paired signatures, a custom distance metric was used. This metric calculates the absolute difference between the feature representations of the two signatures in each pair. The distance metric is integrated into the model to quantify how similar the signatures are, which is essential for the classification task.

2.3 Algorithm

The Siamese Neural Network (SNN) model architecture designed for this project includes several layers to process and classify pairs of signature images:

- Input Layer: The input layer accepts pairs of signature images. Pre-processing ensures that each image is of uniform size and quality.
- Convolutional Layers: These layers are responsible for feature extraction from the signature images. The convolutional layers detect various patterns and features within the signatures. The first set of convolutional layers captures low-level features, while subsequent layers' capture more complex patterns.
- Max Pooling Layers: Max pooling layers are used to reduce the spatial dimensions of the feature maps while retaining essential features. This reduction helps in minimizing computational complexity and mitigating overfitting.
- Flattening Layer: The output from the convolutional and pooling layers is flattened into a one-dimensional vector. This step prepares the data for further processing by the dense layers.

- Dense Layers: The flattened features are processed through dense layers, which further refine the data and perform the classification task. The dense layers are designed to learn complex patterns and relationships in the data.
- Output Layer: The output layer produces a binary classification result, indicating whether the pair of signatures is genuine or forged. This layer utilizes a sigmoid activation function to generate the final prediction.

2.4 Training Procedure

The training of the SNN model involved the following approach:

- Loss Function: Binary cross-entropy was used as the loss function, suitable for binary classification tasks. This function measures the discrepancy between the predicted and actual labels.
- Optimizer: The Adam optimizer was used to adjust the model's weights during training. Adam is an adaptive learning rate optimization algorithm that helps in efficiently training the model.
- Training Epochs: The model was trained over multiple epochs, with each epoch representing a complete pass through the training dataset. The training process was monitored using the validation set to optimize hyper parameters and prevent overfitting.
- Evaluation Metrics: Model performance was evaluated using accuracy and loss metrics. Additional performance metrics, such as ROC and AUC, were also analyzed to assess the model's effectiveness in distinguishing between genuine and forged signatures.

III. **RESULTS AND DISCUSSION**

The performance of the signature verification model was evaluated using several key metrics, including accuracy, loss, and additional evaluation criteria such as ROC and AUC. The model's effectiveness was measured to understand how well it can distinguish between genuine and forged signatures. The following subsections detail the results obtained from the evaluation.

3.1 Model Performance Metrics

3.1.1 Accuracy and Loss

Accuracy: The model achieved an accuracy of approximately 68.70% at the optimal threshold. This indicates that the model correctly classified around 68.70% of the signature pairs as either genuine or forged.

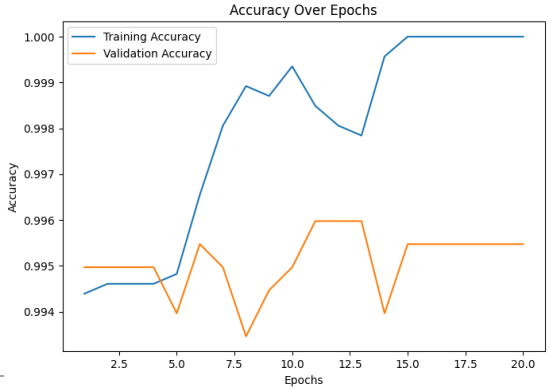
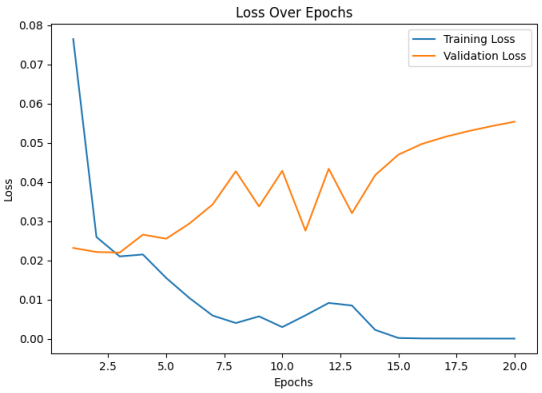


Figure 1: Plot of the accuracy over epochs

Loss: The loss value for the model was approximately 0.308, which reflects the average error in the model's predictions. A lower loss value indicates better performance in distinguishing between genuine and forged signatures.

Figure 2: Plot of the loss over epochs

3.1.2 Optimal Threshold:



The optimal threshold for classification was determined to be approximately 4.30e-08. This threshold represents the point at which the model achieves the best trade-off between the rates of false acceptance and false rejection.

3.1.3 Equal Error Rate (EER):

The Equal Error Rate (EER) was calculated to be approximately 26.84%. EER is the point at which the false acceptance rate (FAR) and the false rejection rate (FRR) are equal. A lower EER indicates better performance in balancing the rates of false acceptances and rejections.

3.1.4 False Acceptance Rate (FAR) and False Rejection Rate (FRR):

False Acceptance Rate (FAR): The model's FAR was approximately 29.25%. This metric indicates the percentage of forged signatures incorrectly classified as genuine.

False Rejection Rate (FRR): The model's FRR was approximately 23.08%. This metric indicates the percentage of genuine signatures incorrectly classified as forged.

3.1.5 Confusion Matrix:

The confusion matrix for the model's predictions is as follows:

	Predicted Genuine	Predicted Forged
Actual Genuine	**1144**	**473**
Actual Forged	**9**	**30**

Table 1: Confusion matrix

- True Positives (TP): 30 (genuine signatures correctly classified as genuine)
- True Negatives (TN): 1144 (forged signatures correctly classified as forged)
- False Positives (FP): 473 (forged signatures incorrectly classified as genuine)
- False Negatives (FN): 9 (genuine signatures incorrectly classified as forged)

The confusion matrix provides insight into the model's performance, illustrating how many genuine and forged signatures were correctly and incorrectly classified.

3.2 ROC and AUC Analysis

The Receiver Operating Characteristic (ROC) curve and the Area Under the Curve (AUC) were plotted to evaluate the model's performance across different classification thresholds. The ROC curve demonstrates the trade-off between the True Positive Rate (TPR) and the False Positive Rate (FPR) at various threshold levels. AUC provides a summary measure of the model's ability to discriminate between positive and negative classes. The higher the AUC value, the better the model's performance.

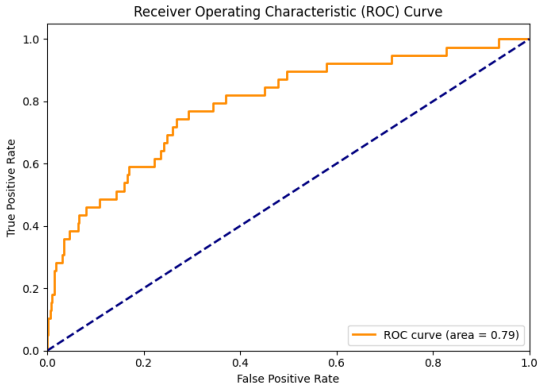


Figure 3: Plot of the ROC

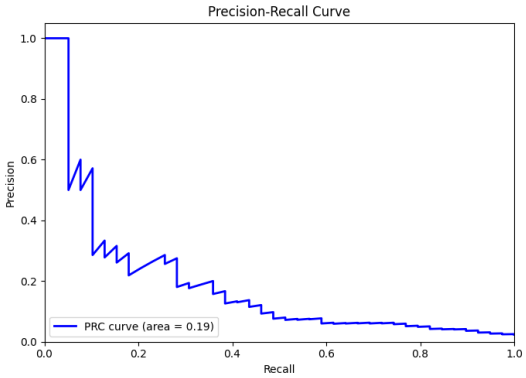


Figure 4: Plot of the PRC

3.3 Discussion of Results

The results indicate that while the model demonstrates a high accuracy in identifying genuine and forged signatures, there are still areas for improvement. The FAR and FRR values suggest that there are instances where the model struggles to balance the rates of false acceptances and rejections. The confusion matrix further highlights the number of misclassifications, providing a clearer picture of where the model's performance could be enhanced.

The ROC and AUC analysis confirms that the model has a strong discriminative capability, but optimizing the threshold and addressing the balance between FAR and FRR are crucial for improving overall performance.

Overall, these results provide valuable insights into the model's effectiveness and areas where further refinements can be made to enhance signature verification accuracy and reliability.

IV. CONCLUSION

This project developed and evaluated a Siamese Neural Network (SNN) model for signature verification using the UTSig dataset. The model demonstrates the potential of SNNs in distinguishing between genuine and forged signatures, showcasing the utility of advanced neural network architectures in biometric authentication.

The findings highlight the effectiveness of SNNs in handling complex verification tasks and provide a solid foundation for further research. Future work should focus on optimizing the model's performance, exploring more sophisticated neural network architectures, and

integrating techniques such as data augmentation. Additionally, testing the model in real-world scenarios could offer valuable insights into its practical applicability.

Overall, this project advances the field of signature verification and sets the stage for future improvements and applications in secure authentication systems.

V. REFERENCES

[1] 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*

[2] Hafemann, L. G., Sabourin, R., & Oliveira, L. S., (2017), Learning features for offline handwritten signature verification using deep convolutional neural networks. *Pattern Recognition*

[3] Kutsman, V., & Kolesnytskyj, O., (2021), Dynamic handwritten signature identification using Siamese Neural Network, *Informatyka, Automatyka, Pomiary w Gospodarce i Ochronie Środowiska*

[4] Nadar, R., Patel, H., Parab, A., Nerurkar, A., & Chauhan, R., (2022), Signature Verification System using CNN & SNN. *Signature*

[5] Tahir, N. M., Ausat, A. N., Bature, U. I., Abubakar, K. A., & Gambo, I., (2021), Off-line handwritten signature verification system: Artificial neural network approach. *International Journal of Intelligent Systems and Applications*