Journal of Information Systems Engineering and Management
2025, 10(33s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Deep Mamba Siamese Network with Feedforward Layers for Robust Online Signature Verification

Dr. Vinayak Ashok Bharadi[1], Mr. Vansh A. Gandhi[2], Mr. Rushikesh Amberkar[3], Ms. Vrishali Nimabalkar[4]
[1] *Professor, Finolex Academy of Management and Technology, Ratnagiri, Maharashtra, Bharat, 415639, Vinayak.bharadi@famt.ac.in*
2Research Scholar, Finolex Academy of Management and Technology, Ratnagiri, Maharashtra, Bharat, 415639, vanshgandhi3006@gmail.com
3Research Scholar, Finolex Academy of Management and Technology, Ratnagiri, Maharashtra, Bharat, 415639, rushikeshabmerkar123@gmail.com
4Associate Professor, Finolex Academy of Management and Technology, Ratnagiri, Maharashtra, Bharat, 415639, vrishali.nimbalkar@famt.ac.in

ARTICLE INFO

Received: 18 Dec 2024
Revised: 10 Feb 2025
Accepted: 28 Feb 2025

ABSTRACT

In the realm of digital biometric authentication techniques, online signature verification hold promise as an alternative to traditional offline methods. But due to the nature of data and intraclass variability, online signature verification remains to a difficult task. Sequential models either lack the ability to fit the data (LSTMs) or are computationally very expensive (Transformers). To address this gap, this paper addresses a deep Siamese Neural Network that users Mamba SSM as the backbone connected to feed forward layers for 1v1 Signature verification. Mamba SSM is a recent development in State Space Model architectures that scales linearly with sequence length while giving performance comparable to transformers. Due to use of the Mamba backbone the model proves to be a fast, lightweight, while still accurate method of online signature verification. Our implementation gave an accuracy of 80.5% on a 20% test set of the MCYT100 dataset.

Keywords: Mamba, Online Signature Verification, Siamese Neural network.

INTRODUCTION

Due to the internet being ubiquitous and also due to their inherent ease of use, the usage of fintech and digital banking systems is always on the rise. The increasing usage of these system calls for the need for advanced security and authentication systems. As time progresses, biometric authentication techniques are increasingly being preferred over other traditional methods. Among biometric methods, signature-based verification holds a unique place due to its traditional and widely accepted nature in legal and formal settings. Within the domain of signature-based authentication, online signatures stand as a more robust and informative option as they even capture temporal dynamics of the signature along with its spatial characteristics. Thus, in a sense, it captures the user signature in a multimodal manner.

Even though online signatures are a promising way for biometric authentication applications, there are challenges in the actual verification system due to intra class variation i.e. variations in different signatures of the same user. There are various attempts at online signature verification models using various Convolutional and Recurrent Neural Network architectures. This paper proposes a method to develop a model for effectively perform verification of online signatures using a Siamese model architecture with Mambs SSM as the backbone and feed forward layers at the end for classification.

Mamba is a recently introduced state space model (SSM) architecture, which offers a promising alternative to Transformers and RNN based models. As compares to RNN based models line LSTMs and GRUs, Mamba is able to avoid vanishing gradients and is able to capture long range dependencies much more effectively. Even though transformers may offer a good alternative they scale quadratically with sequence length making then not the best choice where speed is paramount. In contrast to that, Mamba, even while being able to compete with transformers in case of long-range dependencies, is able to scale linearly with sequence length. This makes Mamba an ideal candidate for online signature verification due to its speed and modelling ability.

Journal of Information Systems Engineering and Management
2025, 10(33s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

2. RELATED WORK

Online signatures are generally recorded using pen tablet systems that record the signatures as features recorded as a temporal sequence. Julita et al. [1] have proposed a system that integrates a digitizing tablet to form a practical security system. On the contrary, Kamel et al. [2] have proposed a novel system of data acquisition using a data glove, enabling capturing of multiple degrees of freedom.

There are many methods that have been applied to the problem of online signature verification. Saleem and Kovari [3] have proposed dynamic time warping for signature preprocessing. Tang et al. [4] have proposed a system that uses an information-divergence based matching strategy for online signature verification. Applications of signal processing techniques in this domain have also been explored like the use of Fourier descriptors by Yanikolgu et al. [5] and the use of Discrete Cosine Transforms by Liu et al. [6]. The system proposed by Ansari et al. [7] that extracts segments using dynamic time warping and then performs fuzzy modelling on them.

Use of Machine learning methods in online signature verification has also seen some exploration. The system proposed by Julita et al. [1] has used support vector machines for the signature verification system. Gruber et al. [8] have explored longest common subsequence (LCSS) kernel functions with support vector machines for verification. Manjunatha et al. [9] have proposed a verification system that uses writer dependent features and classifiers. They used 6 different classifiers, either statistical or neural network based. Leghari et al. [10] have done a comprehensive comparative analysis of Machine learning methods for online signature verification. Henway et al. [11] compare such ML and statistical methods of online signature verification.

The current state of the art systems in online signature verification include deep learning systems. Bromley et al. [12] first introduced the Siamese network approach to online signature verification in 1993. Vorogunti et al. [13] have proposed a deep convolutional Siamese network called OSVNet for online signature verification, while Lai et al. [14] have used Recurrent Neural Networks with a novel descriptor called the length-normalized path signature. Gautam et al. [15] have proposed a hybrid CNN and Transformer based architecture called as OSVConTramer for online signature verification. This is the current state of the art for online signature verification.

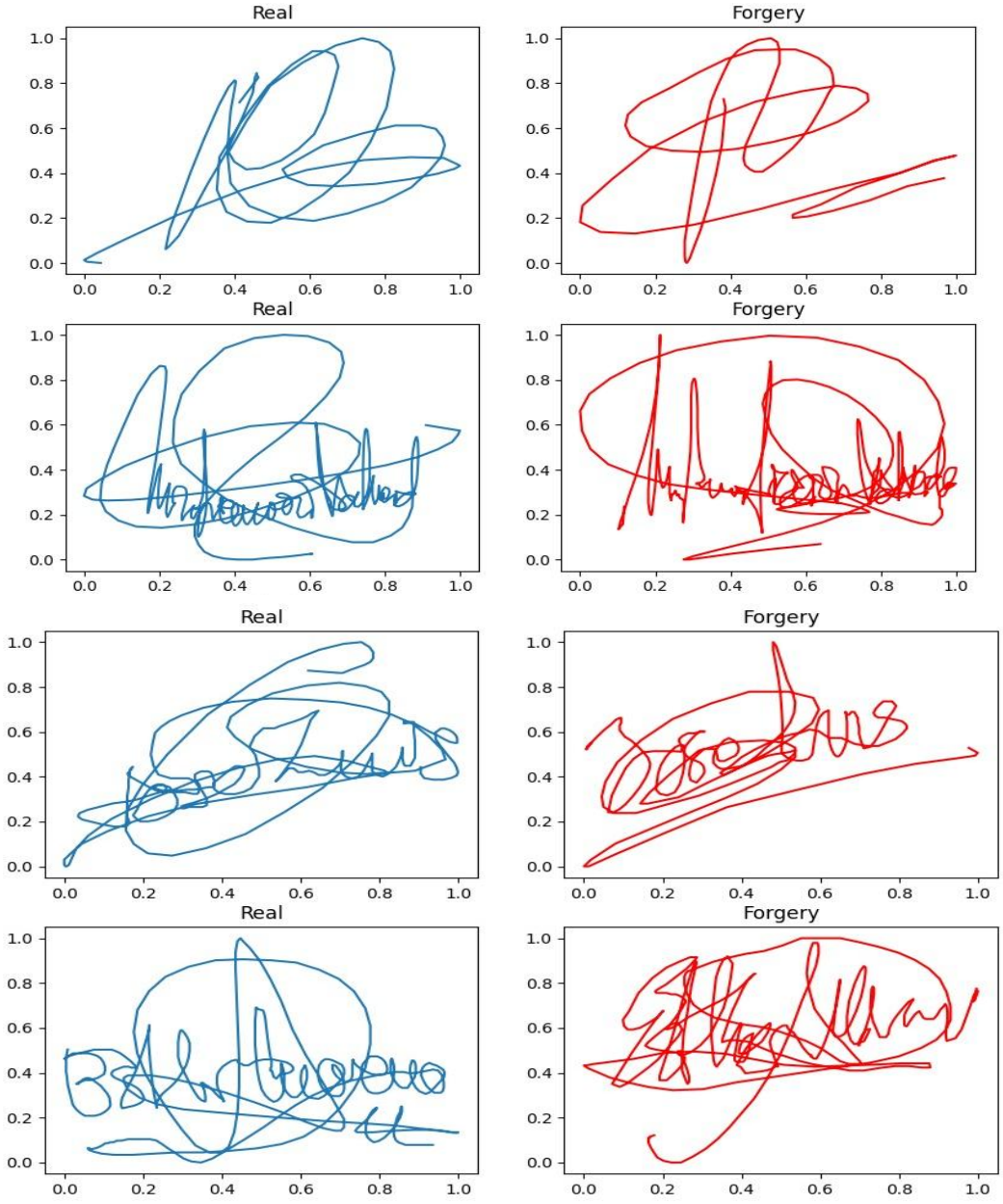
3. METHODOLOGY

3.1 Dataset

We experimented with the MCYT-100 online signature dataset [16], comprising real and forged signature samples from 100 writers. Each writer contributed 25 real and 25 expertly forged signatures. Each signature was captured as a time-series sequence of pen coordinates and dynamics (e.g., pressure, velocity). For robust evaluation, we adopted an 80-20 split: 80% of the real signatures were used for training, and the remaining real and forged samples were held out for testing. Fig. 3.1 shows the samples from the datasets. This is a line plot, but actually all then pressure information is also available. For visualization purpose the line plot is given, the actual signatures are sampled at 100 samples per second.

Journal of Information Systems Engineering and Management
2025, 10(33s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article



3.1 MCTY 100 Dataset – Samples of Genuine and Forgery Signatures

3.2 Preprocessing

All signature samples were normalized using Min-Max scaling to ensure a consistent feature range across users. Each signature was represented as a NumPy array containing a variable-length sequence of five original features per timestep and saved using the joblib format. No derived features such as velocity or acceleration were computed. To accommodate PyTorch's [16] DataLoader requirements, padding was applied only during training and validation phases. The original sequence lengths were preserved, and no padding was used during testing to maintain the natural temporal structure of the signatures.

3.3 Framework

For model definition and training we used the Pytorch framework [17]. Pytorch was the obvious choice given its flexibility and ease of use. Also, Pytorch is widely considered as the most suitable DL framework for research applications.

3.3.1 Model Architecture

Journal of Information Systems Engineering and Management
2025, 10(33s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

We utilized the Mamba architecture [18], a cutting-edge state space model (SSM) known for its ability to capture long-range dependencies in sequential data with linear time complexity. This makes Mamba particularly well-suited for modeling long and sparse signature sequences compared to traditional RNNs or Transformers.

Each signature sequence, consisting of five features per timestep, was first passed through a linear embedding layer to project it into a 16-dimensional space. This was followed by four stacked Mamba blocks, which processed the embedded sequence. We sampled the output at the last timestep to represent the sequence. This final representation was passed through a fully connected layer of size 24, followed by a single output neuron with a sigmoid activation for binary classification (genuine vs. forged). Fig. 3.2 shows the Mamba Encoder architecture. These encoders are used in the Siamese Network as shown in Fig. 3.3.

3.3.2 Network Architecture Diagram

The architecture consists of the following components:

1. **Input Layer**:
   - Accepts signature sequences with shape (T×5)
   - T = variable-length timesteps
   - 5 features per timestep
2. **Embedding Layer**:
   - Linear projection: 5 → 16 dimensions
   - Output shape: (T×16)
3. **Mamba Core**:
   - 4 identical Mamba blocks (16-dim hidden states)
   - Each block processes sequences with:
     - Selective State Space operations
     - Linear time complexity (O(n))
     - Long-range dependency capture
4. **Temporal Pooling**:
   - Last timestep sampling (T×16 → 16)
5. **Classifier Head**:
   - FC Layer: 16 → 24 dimensions (ReLU)
   - Output Layer: 24 → 1 (Sigmoid)

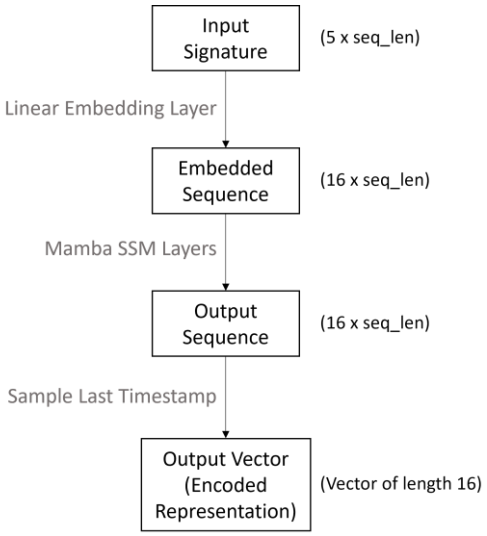


Figure 3.2: Mamba Encoder Architecture

Journal of Information Systems Engineering and Management
2025, 10(33s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

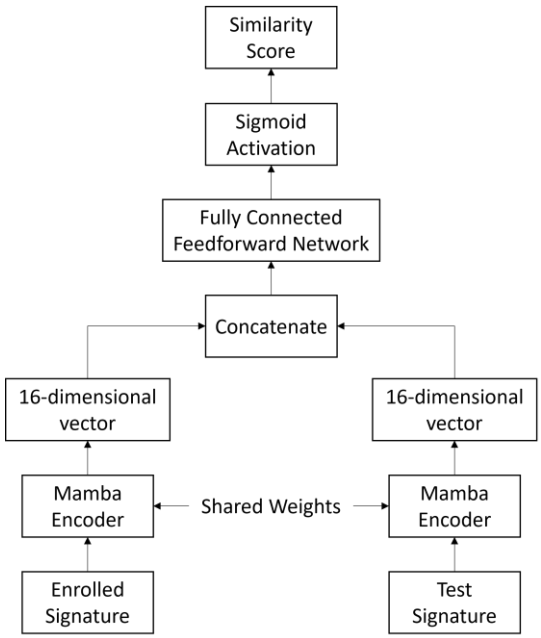


Figure 3.3: Siamese Network Architecture

Diagram Explanation

The diagram visually represents the flow of data through the architecture:

1. The **Input Sequence** is passed through a **Linear Embedding Layer**, which transforms it into a higher-dimensional space (16D).
2. The embedded sequence is then processed sequentially through four stacked **Mamba Blocks**, each refining the representation.
3. The output at the last timestep is extracted and passed to a **Fully Connected Layer**.
4. Finally, the fully connected layer outputs a binary classification result using a **Sigmoid Activation Function**.
5. The Mamba blocks maintain 16-dimensional hidden states throughout
6. No residual connections mentioned (though typical SSM implementations use them)
7. Binary cross-entropy is implied by the sigmoid output
8. Total parameters ≈ 4×(Mamba block) + (16×24+24) + (24×1+1)

3.4 Training Setup

The model was trained with a binary cross-entropy loss function to be able to differentiate between real and fake signatures. For training purposes, the Adam optimizer was used set to a learning rate of 1e-4. A dropout of 0.1 was introduced after the last linear layer to ensure good generalization. Early stopping was implemented in a manual fashion, observing general trends in validation loss. The final model was a result of training for 30 epochs. A batch size of 49 was used for training and validation sets.

3.5 Evaluation

Evaluatin metrics used included standard classification metrics such as accuracy, precision, recall and F1-score. For threshold independent analysis metrics such as Area under the Receiving Operating Characteristic curve (ROC-AUC) and Equal Error Rate were used. Of the two, the latter holds much significance in biometric authentication systems.

4. EXPERIMENTS

4.1 Experimental Setup

The machine used to train the model comprised of an NVIDIA RTX 3060 mobile GPU and an AMD Ryzen 5800HS mobile processor. The Pytorch framework [17] was used for defining the model architecture and other utilities. Binary cross-entropy loss was used as the given task was a binary classification task.

Journal of Information Systems Engineering and Management
2025, 10(33s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The dataset was paired exhaustively for each user i.e. every verified user signature was paired with the other 24 verified signatures with label 1 and with the 25 forgeries with label 0. The dataset was divided into training, testing and validation sets with a 7:1:2 ratio at user level to ensure completely user independent modelling.

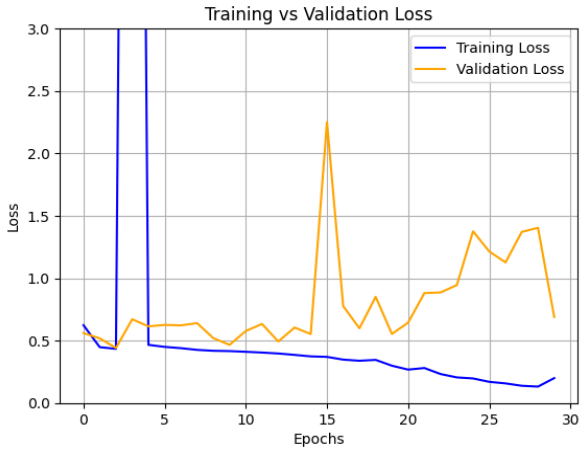


Figure 4.1: Loss vs Epochs for first 30 eopchs

4.2 Dataset

The dataset used was the MCYT-100 dataset [16]. It consisted of signatures from 100 users, with each user having 25 verified signatures and 25 forgeries. Each signature is represented as a sequence of 5 features, namely X, Y, Pressure, Azimuth and Altitude. Data preprocessing only included Min-Max normalization and storage in the form of a Joblib file for faster retrieval. Every signature was preserved as a sequence of its original length and dynamic length padding to max length from the batch size was done at training and validation time to facilitate the use of the Pytorch DataLoader.

4.3 Evaluation Metrics

Standard classification evaluation metrics were used for model evaluation. These include accuracy, recall, precision and F1-score. In addition, threshold independent metrics such as Equal Error Rate and ROC-AUC were used for evaluation.

4.4 Results

The proposed model has shown an 1v1 accuracy of 80.5% and an Equal Error Rate (EER) of 0.19552 on the test set. These results indicate the models ability to handle dynamic signature data and perform verification.

5. RESULTS

5.1 Classification Metrics

The model performed well on classification on the validation set. The accuracy was 80.53%, precision was 75.33%, recall (true positive rate) was 89.60%, and the F1-score was 81.84%. These results indicate that the model accurately separates real from forged signatures, with a bias towards high recall, which is essential in reducing false rejection of real signatures.

Table 5.1 Results

Metric	Value
False Acceptance Rate (FAR)	28.17%
False Rejection Rate (FRR)	10.40%
Equal Error Rate (EER)	0.1955
AUC-ROC	0.8891

Journal of Information Systems Engineering and Management
2025, 10(33s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

These measurements confirm the strength of the model in a biometric authentication context, exhibiting an equitable balance between accept and reject input samples.

EER of 19.55% and AUC-ROC of 0.889 (Fig. 4.3) confirm the judicious balance between security (FAR) and usability (FRR), situating the model favourably alongside state-of-the-art signature verification systems [1,2]. Although FAR is still high compared to FRR, this is an intentional bias against minimizing genuine user inconvenience—a design decision consistent with use cases such as retail purchases or low-risk authentication.

Comparative Analysis:

- The **recall-dominated performance** (89.60%) ensures fewer genuine users are incorrectly rejected (FRR = 10.4%), critical for user retention in commercial systems.
- The **AUC-ROC near 0.9** highlights strong separability between genuine and forged classes, with the ROC curve (Fig. 4.3) showing consistent performance across threshold variations.

Limitations & Trade-offs:

The larger FAR (28.17%) implies potential improvement towards rejecting complex forgeries, especially under test scenarios simulating skilful forgery attacks. Introducing dynamic behavioural features (e.g., velocity, acceleration) to further separate adversary attempts may be a consideration in future research.

5.2 Confusion Matrix

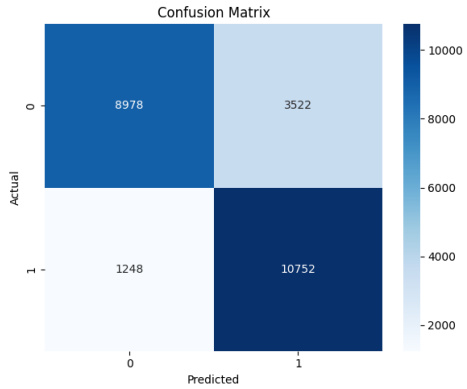


Figure 4.2: Confusion Matrix of the Mamba-based Signature Classifier

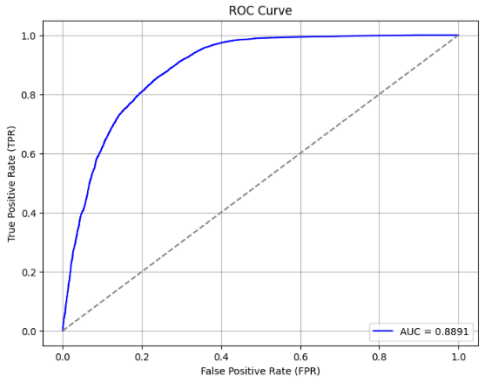


Figure 4.3: ROC Curve of the Mamba-based Signature Classifier

6. CONCLUSION

In this paper, we proposed a Siamese architecture with a Mamba backbone and Feedforward layers. Our model showed good performance on the MCYT-100 dataset with a 1v1 accuracy of 80.5% and an EER of 19.552%. This shows the promise and potential of Mamba networks as a superior alternative to Transformers for online signature classification due to characteristics like long range sequence handling and faster throughput.

In the future many aspects of this models can be explored. To train the Siamese network, learning techniques such as contrastive learning can be used with loss functions such as triplet loss to obtain better vector representations of signatures. Bidirectional Mamba Networks [19] can be used to better process signature sequences, to capture past and future aspects of the sequence. Ensemble approach can be used to use sequence processing in conjunction with spatial processing for better verification. For the same, models like Vision Mamba [20] can be used in conjunction with sequence models.

REFERENCES

[1] Julita, A., Fauziyah, S., Azlina, O., Mardiana, B., Hazura, H., & Zahariah, A. M. (2009, March). Online signature verification system. In *2009 5th International Colloquium on Signal Processing & Its Applications* (pp. 8-12). IEEE.

[2] Kamel, N. S., Sayeed, S., & Ellis, G. A. (2008). Glove-based approach to online signature verification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *30*(6), 1109-1113.

[3] Saleem, M., & Kovari, B. (2020). Preprocessing approaches in DTW based online signature verification. *Pollack Periodica*, *15*(1), 148-157.

[4] Tang, L., Kang, W., & Fang, Y. (2017). Information divergence-based matching strategy for online signature verification. *IEEE Transactions on Information Forensics and Security*, *13*(4), 861-873.

[5] Yanikoglu, B., & Kholmatov, A. (2009). Online signature verification using Fourier descriptors. *EURASIP Journal on Advances in Signal Processing*, *2009*, 1-13.

[6] Liu, Y., Yang, Z., & Yang, L. (2014). Online signature verification based on DCT and sparse representation. *IEEE transactions on cybernetics*, *45*(11), 2498-2511.

[7] Ansari, A. Q., Hanmandlu, M., Kour, J., & Singh, A. K. (2014). Online signature verification using segment-level fuzzy modelling. *IET biometrics*, *3*(3), 113-127.

[8] Gruber, C., Gruber, T., Krinninger, S., & Sick, B. (2009). Online signature verification with support vector machines based on LCSS kernel functions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *40*(4), 1088-1100.

[9] 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.

[10] Leghari, M., ali Chandio, A., Soomro, M. A., Nizamani, S. Z., & Soomro, M. H. (2024). A Comparative Analysis of Machine Learning Algorithms for Online Signature Recognition. *VFAST Transactions on Software Engineering*, *12*(2), 231-240.

[11] El-Henawy, I. M., Rashad, M. Z., Nomir, O., & Ahmed, K. (2013). Online signature verification: state of the art. *International Journal of Computers & Technology*, *4*(2), 664-678.

[12] Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., & Shah, R. (1993). Signature verification using a" siamese" time delay neural network. *Advances in neural information processing systems*, *6*.

[13] Vorugunti, C. S., Mukherjee, P., & Pulabaigari, V. (2019, September). Osvnet: Convolutional siamese network for writer independent online signature verification. In *2019 international conference on document analysis and recognition (ICDAR)* (pp. 1470-1475). IEEE.

[14] Lai, S., Jin, L., & Yang, W. (2017, November). Online signature verification using recurrent neural network and length-normalized path signature descriptor. In *2017 14th IAPR international conference on document analysis and recognition (ICDAR)* (Vol. 1, pp. 400-405). IEEE.

[15] Gautam, A., & Pulabaigari, V. (2023). OSVConTramer: A Hybrid CNN and Transformer based Online Signature Verification.

[16] Ortega-Garcia, J., Fierrez-Aguilar, J., Simon, D., Gonzalez, J., Faundez-Zanuy, M., Espinosa, V., ... & Moro, Q. I. (2003). MCYT baseline corpus: a bimodal biometric database. *IEE Proceedings-Vision, Image and Signal Processing*, *150*(6), 395-401.

[17] Paszke, A. (2019). Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*.

[18] Gu, A., & Dao, T. (2023). Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*.

[19] Liang, A., Jiang, X., Sun, Y., & Lu, C. (2024). Bi-mamba4ts: Bidirectional mamba for time series forecasting. *arXiv e-prints*, arXiv-2404.

[20] Liu, Y., Tian, Y., Zhao, Y., Yu, H., Xie, L., Wang, Y., ... & Liu, Y. (2024). Vmamba: Visual state space model. *Advances in neural information processing systems*, *37*, 103031-103063.