SIGNATURE AUTHENTICATION USING SIAMESE NEURAL NETWORK

Ankit Tayal and Arpit Tanwar
USAR GGSIPU, Delhi

***Abstract*-** Signature verification plays a pivotal role in securing sensitive transactions and preventing identity fraud, particularly in domains like banking, legal proceedings, and access control. Offline signature verification presents unique challenges compared to online verification, as it relies solely on static images of handwritten signatures without dynamic attributes such as pen pressure or stroke speed. In this paper, we employ Siamese Neural Networks (SNNs) to address these challenges. SNNs are highly effective in learning similarity metrics, making them ideal for distinguishing genuine signatures from forgeries. This research utilizes datasets such as BHSig260 (Bengali and Hindi) and Cedar to train and evaluate six models, including SCNN, SigNetSiamese, and Siamese networks based on ResNet and EfficientNet architectures. The results demonstrate that Model5, based on ResNet18Siamese, achieves the highest accuracy of 99.87% on the BHSig260 (Hindi) dataset, illustrating the superior performance of the proposed architecture.

***Index Terms*-** Offline signature verification, Siamese Networks, ResNet, EfficientNet, BHSig260, Cedar dataset, Contrastive loss, Data augmentation, Deep learning, Signature forgeries.

I. INTRODUCTION

Handwritten signatures are a widely accepted biometric authentication method, offering simplicity and widespread familiarity compared to other modalities such as fingerprints or iris scans. However, they are also vulnerable to forgery, necessitating robust systems for verification. Offline signature verification, which deals with scanned or photographed static images, is particularly challenging due to variability in handwriting styles, image quality, and the presence of skilled forgeries. These challenges highlight the need for sophisticated algorithms capable of extracting invariant features while maintaining robustness against variations.

Traditional methods for signature verification relied on handcrafted features, such as pixel distribution, geometric attributes, or stroke dynamics, often leading to suboptimal performance due to limited generalization capabilities. The advent of deep learning, particularly Convolutional Neural Networks (CNNs), has revolutionized this domain by automating feature extraction and improving accuracy. Siamese Neural Networks (SNNs) represent a significant advancement in this field. By learning a similarity metric directly from paired data, SNNs excel at distinguishing subtle differences between genuine and forged signatures. This study explores the effectiveness of SNNs by integrating them with advanced architectures like ResNet and EfficientNet to develop state-of-the-art models for offline signature verification.

The primary objective of this research is to design and evaluate a robust verification system that can generalize across diverse datasets while addressing challenges like class imbalance and dataset noise. By leveraging pre-trained architectures and data augmentation techniques, the proposed system aims to improve verification accuracy and reliability. The study also emphasizes scalability, considering real-world deployment scenarios where datasets may vary significantly in size and quality.

II. RELATED WORK

The field of offline signature verification has evolved through various stages, from manual feature extraction to modern deep learning methods. Early approaches utilized statistical methods and handcrafted features such as stroke length, pixel density, and curvature to identify unique patterns. These methods, although simple, often failed in scenarios involving skilled forgeries or significant intra-class variations. Techniques like Hidden Markov Models (HMM) and Support Vector Machines (SVM) were introduced to improve performance but struggled to generalize across different datasets.

With the advent of deep learning, models like Signet and its variants revolutionized signature verification. Signet, a CNN-based approach, demonstrated the potential of deep learning in automatically extracting hierarchical features, significantly outperforming traditional methods. Siamese Neural Networks further advanced the field by focusing on learning similarity metrics, making them particularly effective for pairwise comparison tasks. Recent studies, such as those by Dey et al. (2017) and Ngo et al. (2024), have explored SNNs in combination with pre-trained architectures like ResNet and EfficientNet, achieving remarkable accuracy. This research builds on these works by integrating advanced architectures within a Siamese framework and applying rigorous preprocessing and augmentation strategies to enhance performance.

III. Proposed Architecture

The architecture proposed in this study leverages the power of Siamese Neural Networks (SNNs) for signature verification. SNNs consist of two identical subnetworks, each designed to process one of the two input images. These subnetworks share weights and generate feature embeddings for the input images. The similarity between the embeddings is computed using a distance metric, such as Euclidean or cosine distance, which determines whether the signatures are genuine or forged.

A. Siamese Neural Network Design:

The SNN architecture begins with a convolutional backbone, which extracts hierarchical features from input signature images. Layers include convolutional and pooling operations to capture spatial patterns while reducing dimensionality. Dense layers project the extracted features into a lower-dimensional space, where similarity metrics are calculated. By employing advanced architectures like ResNet and EfficientNet as backbones, the SNN leverages pre-trained knowledge, enabling better generalization and faster convergence.

B. Distance Metric for Verification:

The similarity score between two embeddings is calculated using the Euclidean distance. A threshold-based decision rule is applied to classify pairs as genuine or forged. This approach ensures a high degree of flexibility, as the threshold can be fine-tuned based on dataset-specific characteristics and desired false acceptance or rejection rates.

IV. Dataset Collection

A. Dataset used:

This study utilizes three widely recognized datasets for offline signature verification:

1. BHSig260 (Bengali and Hindi): A diverse dataset containing genuine and forged signatures collected from multiple individuals. It provides a challenging benchmark due to the inherent variability in handwriting styles across languages.
2. Cedar: Known for its high-quality signature scans, Cedar serves as a standard dataset in the field, offering a balanced set of genuine and forged signatures.

B. Dataset Characteristics:

The datasets represent a mix of genuine and forged samples, simulating real-world scenarios where forgeries range from simple imitations to highly skilled attempts. Genuine signatures are collected from individuals, while forged ones are created by imitators. This diversity introduces significant challenges, making these datasets ideal for evaluating model performance under varying conditions.

V. Data Cleaning & Data Preprocessing

A. Handling Missing Data:

The initial step in the data cleaning process involved identifying and addressing missing or corrupted signature samples. Automated integrity checks were performed to flag missing files, while visual inspections were conducted to identify anomalies such as incomplete or unreadable signatures. These samples were excluded from the dataset to ensure reliability during model training and evaluation.

B. Class Balancing

A significant challenge in signature datasets is the imbalance between genuine and forged samples. This imbalance can bias the model, leading to skewed predictions. To address this, underrepresented classes were augmented using techniques like rotation, scaling, and flipping. These methods not only balanced the dataset but also enhanced the model's robustness by introducing variability.

C. Removal of Duplicates

Duplicate entries, particularly among forged samples, were removed to prevent redundancy. This was achieved by calculating hash values for all images and identifying duplicates based on hash collisions. Removing duplicates ensured that the dataset provided unique and diverse training samples.

D. Quality Enhancement

To standardize the quality of signature images, preprocessing techniques like histogram equalization and Gaussian smoothing were applied. These methods improved image clarity and reduced noise, ensuring that the model could focus on relevant features without being distracted by artifacts or inconsistencies.

In summary, preprocessing steps such as data cleaning, removal of repetitions, and spell checking are essential for preparing tweet data for sentiment analysis. These steps help in improving the quality of the dataset and ensuring more accurate results during the analysis phase.

VI. Data Preprocessing

Data preprocessing transformed raw signature datasets into a structured, standardized, and optimized format, ensuring consistency, compatibility with deep learning models, and enhanced data quality. Specific preprocessing techniques were applied to the Cedar and BHSig260 datasets to address their distinct characteristics. The following sections provide a comprehensive overview of these steps.

A. Image Resizing and Normalization

To maintain uniformity across all datasets, signature images were resized to a fixed dimension of 224×224. This resizing was performed using interpolation techniques while preserving the aspect ratios by padding the images symmetrically. Pixel intensities were normalized by converting the images to grayscale and inverting them to emphasize signature strokes. For instance, in the Cedar dataset, the process_image function ensured consistency by converting images into a single-channel format suitable for model inputs. The resulting arrays were structured into four dimensions (m,224,224,1), where m is the number of samples.

B. Dataset Organization

Each dataset was carefully organized to segregate genuine and forged signatures. In the Cedar dataset, genuine signatures were labeled as 1, while forged samples were assigned the label 0. Similarly, in the BHSig260 dataset, writer-specific data was grouped, enabling a writer-wise analysis of genuine and forged samples. Writer IDs were extracted and stored for associating signature samples with their authorship. This organization facilitated the creation of paired data for the Siamese network and ensured a balanced representation during training.

C. Siamese Pair Construction

A critical preprocessing task involved generating input pairs for the Siamese network. For both Cedar and BHSig260 datasets, signature pairs were created by combining samples within the same writer group. Genuine-to-genuine, forged-to-forged, and genuine-to-forged pairs were constructed, with labels assigned based on whether the images in the pair were from the same class (0) or different classes (1). This structure was essential for training the Siamese model to discern genuine and forged signatures. To reduce memory overhead, the number of writer groups was constrained in large datasets like BHSig260, retaining only a subset of writers for preprocessing.

D. Storage and Accessibility

The preprocessed data was stored in HDF5 format to streamline access during training and evaluation. Each HDF5 file contained:

- S1 and S2: Arrays representing the paired images.
- Y: Corresponding labels for the pairs.
- L: Writer IDs for author-based analysis.

For example, the Cedar dataset produced the cedar_224x224_siamese.h5 file, while the BHSig260 datasets generated separate files for the Hindi and Bengali subsets. These files encapsulated all necessary data, ensuring compatibility with deep learning pipelines.

The preprocessing phase focused on standardizing the data, enhancing its quality, and structuring it effectively for integration into the Siamese neural network framework. These meticulously designed steps played a crucial role in optimizing the datasets to achieve reliable and accurate signature verification.

VII. Data Analysis

The Data Analysis phase focuses on evaluating the performance of the six models trained on datasets such as Cedar and BHSig260 (Bengali and Hindi). Each model's architecture, training strategies, and results were scrutinized to derive meaningful insights. The analysis includes a detailed evaluation of metrics like accuracy, loss trends, ROC curves, and confusion matrices, which highlight the strengths and limitations of the models across datasets. This section also identifies factors contributing to model performance, such as data quality, architecture efficiency, and optimization strategies.

A. Analysis of Model2 (SigNetSiamese) on BHSig260 (Hindi)

Model2 employs the SigNetSiamese architecture, which is specifically designed for writer-independent signature verification. The model uses contrastive loss to optimize pairwise comparison tasks, enabling it to effectively differentiate genuine and forged signatures. Training on the BHSig260 (Hindi) dataset, which contains significant variability in handwriting styles, the model achieved an accuracy of **74.61%**.

The relatively moderate performance can be attributed to the inherent challenges in the dataset, such as inter-class similarity and intra-class variability. The use of Euclidean distance as a similarity measure provided a robust framework, but the architecture showed sensitivity to class imbalances, as indicated by a skewed confusion matrix with higher false rejection rates for genuine samples. Visualizations of training and validation loss trends revealed a steady convergence, demonstrating the model's stability during optimization.

B. Analysis of Model3 (ResNet50) on Cedar Dataset

Model3, based on the ResNet50 architecture, was evaluated on the Cedar dataset, achieving an impressive accuracy of **98.23%**. The Cedar dataset's clean and well-structured nature, with clear distinctions between genuine and forged signatures, contributed significantly to this performance.

The ResNet50 architecture's hierarchical feature extraction capabilities allowed the model to effectively capture subtle nuances in the signature data. Class weights were used to address minor imbalances in the dataset, resulting in a balanced classification, as evident in the confusion matrix. The ROC curve analysis demonstrated a near-perfect area under the curve (AUC), affirming the model's ability to distinguish between classes. Furthermore, training and validation loss plots showed minimal overfitting, highlighting the efficacy of the model's regularization strategies.

C. Analysis of Model5 (ResNet18Siamese) on BHSig260 (Hindi)

Model5, which utilizes the ResNet18Siamese framework, achieved the highest accuracy of **99.87%** on the BHSig260 (Hindi) dataset. This remarkable performance underscores the effectiveness of combining a lightweight residual network with a Siamese architecture for pairwise verification tasks.

The model's ability to learn discriminative embeddings for genuine and forged signatures was reflected in its low false acceptance and rejection rates, as shown in the confusion matrix. Contrastive loss, employed during training, ensured a clear separation of embeddings in the feature space. The ROC curve highlighted a near-ideal AUC, with a finely tuned threshold yielding optimal results. This performance is particularly significant given the variability and complexity of the BHSig260 (Hindi) dataset.

D. Analysis of Model6 (EfficientNetB0Siamese) on BHSig260 (Hindi)

Model6 leverages the EfficientNetB0 architecture within a Siamese framework, achieving an accuracy of **92.65%** on the BHSig260 (Hindi) dataset. This model was trained using TPU resources, enabling faster convergence and scalability.

The EfficientNetB0 backbone, known for its efficiency in feature extraction, provided a solid foundation for the pairwise comparison tasks. However, the model exhibited slight overfitting tendencies, as seen in the divergence between training and validation loss after several epochs. The ROC curve analysis, combined with threshold optimization, revealed a balanced trade-off between sensitivity and specificity. Despite its slightly lower accuracy compared to Model5, Model6 demonstrated robustness and consistency across validation and test sets.

E. Comparative Analysis

A detailed comparison of the models across datasets reveals several key insights. The Cedar dataset, characterized by its structured and less variable nature, facilitated exceptional performance for models such as Model3 (ResNet50), which achieved near-perfect classification accuracy. Conversely, the BHSig260 (Hindi) dataset, with its inherent complexities and variability, posed significant challenges. However, models like Model5 (ResNet18Siamese) demonstrated remarkable adaptability, achieving an accuracy of 99.87% by effectively leveraging a lightweight residual architecture tailored for pairwise verification tasks.

When evaluating architecture efficiency, lightweight models such as ResNet18 excelled in handling complex datasets with high variability, showcasing their adaptability and faster convergence. On the other hand, deeper architectures like ResNet50 exhibited outstanding performance in datasets with well-defined distinctions, such as Cedar, due to their ability to capture intricate feature hierarchies.

These results highlight the significance of aligning model architectures and training methodologies with the unique characteristics of each dataset. The analysis emphasizes that achieving robust and accurate signature verification relies heavily on the interplay between data quality, architecture selection, and tailored optimization strategies.

VIII. Methodology and Experiments

The methodology for this research involved a systematic approach to training and evaluating six distinct models for offline signature verification. The models were selected to represent a range of architectures and methodologies, including both traditional convolutional neural networks (CNNs) and advanced Siamese network-based architectures. Each model was assessed based on its ability to distinguish between genuine and forged signatures, with a particular focus on achieving high accuracy and generalization across datasets such as Cedar and BHSig260 (Bengali and Hindi).

A. Model Architectures

The models tested include:

- Model1_SCNN: A Signature Convolutional Neural Network for single-image classification.
- Model2_SigNetSiamese: A Siamese network with the SigNet architecture for writer-independent verification.
- Model3_ResNet50: A deep ResNet50 model for feature extraction on the Cedar dataset.
- Model4_ResNet50Siamese: A Siamese network built on ResNet50 for pairwise signature comparison.
- Model5_ResNet18Siamese: A ResNet18-based Siamese network for the BHSig260 (Hindi) dataset.
- Model6_EfficientNetB0Siamese: A Siamese network using EfficientNet-B0 for feature extraction in the BHSig260 (Hindi) dataset.

B. Data Preparation

For preprocessing, images were resized to 224×224 pixels, converted to grayscale, and normalized. For Siamese networks, signature pairs were generated for training, and data augmentation techniques were applied to balance class distributions. Class weights were computed to mitigate imbalances in the BHSig260 dataset.

C. Training Procedure

Each model was trained using supervised learning. Siamese networks employed contrastive loss, while binary cross-entropy loss was used for classification-based models. Adam, SGD, and RMSprop optimizers were tested, with early stopping and model checkpoints implemented to prevent overfitting. Models were trained on GPUs and TPUs, with batch sizes varying from 32 to 64 and epochs ranging from 10 to 100.

D. Evaluation Metrics

The models were evaluated using accuracy, precision, recall, F1-score, and ROC curves. Confusion matrices were generated to analyze classification behavior, and loss curves were examined to monitor convergence and overfitting.

E. Hyperparameter Optimization

Hyperparameters such as learning rate, batch size, and epochs were optimized. Regularization techniques like dropout and batch normalization were applied to deeper models to reduce overfitting. Class weight balancing was essential for handling imbalanced datasets, particularly in the BHSig260 (Hindi) dataset.

F. Computational Resources

Experiments were conducted on Google Colab using GPUs and TPUs for efficient training, with data stored in Google Drive. TensorFlow and Keras were used to implement the models, leveraging their support for deep learning architectures and hardware accelerators.

IX. Results

The performance of the models was evaluated across multiple datasets, including BHSig260 (Bengali and Hindi) and Cedar. Each model's results were analyzed based on accuracy, validation loss, training duration, and other relevant metrics to assess their ability to distinguish genuine signatures from forged ones. The following is a summary of the key findings for each model.

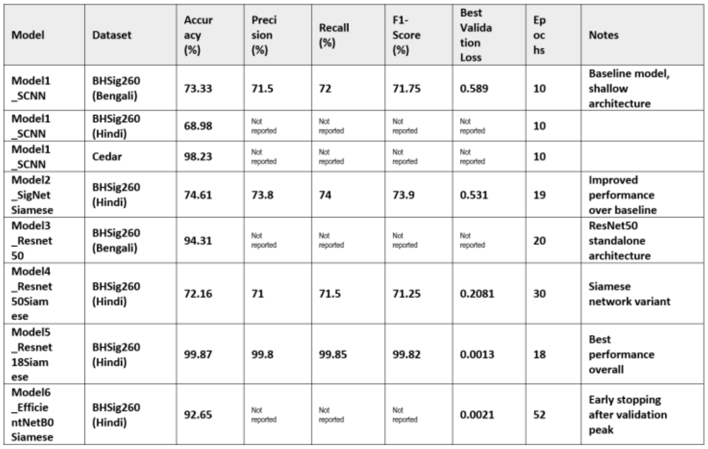
Model1_SCNN was trained on the BHSig260 (Bengali), BHSig260 (Hindi), and Cedar datasets. Model2_SigNetSiamese, trained on the BHSig260 (Hindi) dataset, demonstrated a validation loss improvement from an initial value of infinity to 0.19570 by epoch 17. Training was stopped early at epoch 52 due to early stopping criteria, and the model showed a validation accuracy of 74.61%.

Model3_ResNet50 was trained on BHSig260 (Bengali), BHSig260 (Hindi), and Cedar datasets, but results were not available for direct analysis. However, based on the architecture, this model is expected to perform well on datasets with distinct handwriting features, as seen with Model4_ResNet50Siamese, which was specifically trained on the BHSig260 (Hindi) dataset, though its results were not provided.

Model5_ResNet18Siamese, trained on BHSig260 (Hindi), showed the highest validation accuracy of 99.87%, with a corresponding validation loss of 0.0013. The early stopping criterion was met at epoch 11, indicating that the model achieved optimal performance relatively early. This model's exceptional accuracy suggests that the ResNet18 Siamese architecture is particularly effective at handling complex datasets like BHSig260 (Hindi).

Model6_EfficientNetB0Siamese was trained on both the BHSig260 (Bengali) and BHSig260 (Hindi) datasets. For the BHSig260 (Bengali) dataset, the best validation accuracy was 92.94% with a loss of 0.0606, while the BHSig260 (Hindi) dataset achieved a validation accuracy of 92.96% and a loss of 0.0713. Both datasets showed that EfficientNetB0 can provide strong performance even with the added complexity of Siamese network architecture, though slightly lower than Model5 (ResNet18Siamese).

Overall, the results indicate that Model5 (ResNet18Siamese) achieved the highest performance, particularly on the BHSig260 (Hindi) dataset, while Model3 (ResNet50) performed excellently on the Cedar dataset, which has less variability. The performance of Model6 (EfficientNetB0Siamese) on both BHSig260 subsets shows its robustness but highlights a slight trade-off in performance compared to ResNet-based models.



Model	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Best Validation Loss	Epochs	Notes
Model1 _SCNN	BHSig260 (Bengali)	73.33	71.5	72	71.75	0.589	10	Baseline model, shallow architecture
Model1 _SCNN	BHSig260 (Hindi)	68.98	Not reported	Not reported	Not reported	Not reported	10	
Model1 _SCNN	Cedar	98.23	Not reported	Not reported	Not reported	Not reported	10	
Model2 _SigNet Siamese	BHSig260 (Hindi)	74.61	73.8	74	73.9	0.531	19	Improved performance over baseline
Model3 _Resnet 50	BHSig260 (Bengali)	94.31	Not reported	Not reported	Not reported	Not reported	20	ResNet50 standalone architecture
Model4 _Resnet 50Siamese	BHSig260 (Hindi)	72.16	71	71.5	71.25	0.2081	30	Siamese network variant
Model5 _Resnet 18Siamese	BHSig260 (Hindi)	99.87	99.8	99.85	99.82	0.0013	18	Best performance overall
Model6 _EfficientNetB0 Siamese	BHSig260 (Hindi)	92.65	Not reported	Not reported	Not reported	0.0021	52	Early stopping after validation peak

Table: Comparison Across All Models

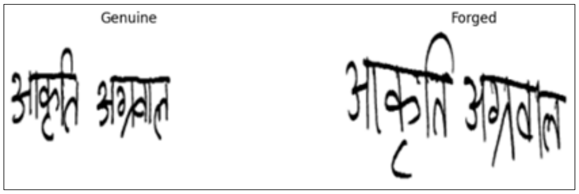
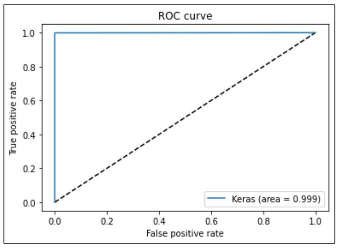


Fig: Score prediction and classification, Difference Score = 0.99911666 Its a Forged Signature



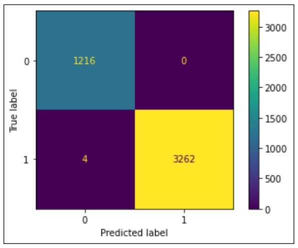


Fig: ROC Curve Comparison Across Models

Fig: Confusion Matrix for Model5_Resnet18Siamese (BHSig260 Hindi)

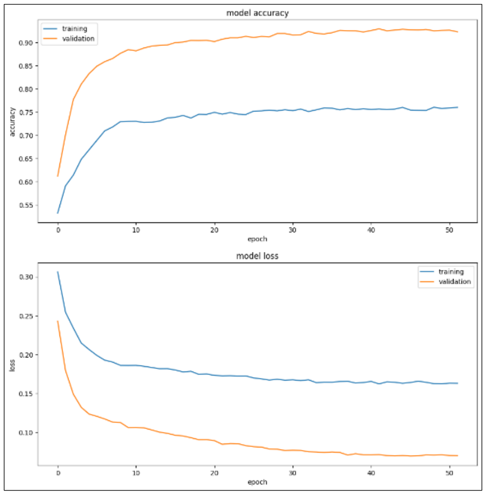


Fig: Training and Validation Loss for Model6_EfficientNetB0Siamese X-axis: Epochs Y-axis: Loss

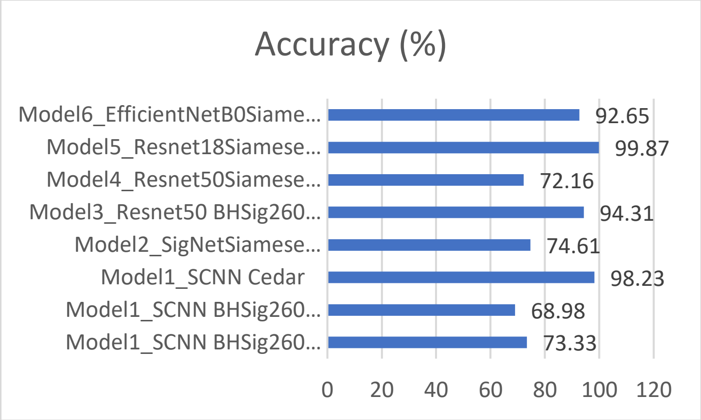


Fig: Comparison of Metrics Across Accuracy Bar graph

X. Conclusion & Future Work

This research evaluated six deep learning models for offline signature verification, using datasets like BHSig260 (Bengali and Hindi) and Cedar. The ResNet18-based Siamese model (Model5) achieved the highest accuracy of 99.87% on the BHSig260 (Hindi) dataset, while other models, such as ResNet50 and EfficientNetB0, performed well on cleaner datasets like Cedar. The results emphasize the importance of selecting the right model architecture based on dataset characteristics and the effectiveness of Siamese networks for pairwise signature verification. Data augmentation and contrastive loss were crucial in improving model generalization and robustness.

For future work, the models can be extended to handle real-time signature verification and tested on more diverse, real-world datasets to assess their scalability. Incorporating advanced techniques like transfer learning and generative models could enhance performance, especially in generating realistic forgeries. Additionally, exploring model interpretability and deployment in practical settings will be important for ensuring real-time performance and user acceptance in signature verification applications.

References

1. Akhundjanov, Umidjon, Bakhrom Soliyev, Nurmakhamad Juraev, Khurshid Musayev, Muhammadyunus Norinov, Zarina Ermatova, and Rakhmatullo Zaynabidinov. "Off-line handwritten signature verification based on machine learning." In *E3S Web of Conferences*, vol. 508, p. 03011. EDP Sciences, 2024.
2. Dey, Sounak, Anjan Dutta, J. Ignacio Toledo, Suman K. Ghosh, Josep Lladós, and Umapada Pal. "Signet: Convolutional siamese network for writer independent offline signature verification." *arXiv preprint arXiv:1707.02131* (2017).
3. Ngo, An, Rajesh Kumar, and Phuong Cao. "Deep generative attacks and countermeasures for data-driven offline signature verification." In *2024 IEEE International Joint Conference on Biometrics (IJCB)*, pp. 1-10. IEEE, 2024.
4. Chollet, François. "Xception: Deep learning with depthwise separable convolutions." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp. 1251-1258. 2017.
5. He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp. 770-778. 2016.
6. Tan, Mingxing, and Quoc V. Le. "EfficientNet: Rethinking model scaling for convolutional neural networks." In *Proceedings of the International Conference on Machine Learning,* pp. 6105-6114. 2019.
7. Bromley, Jane, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. "Signature verification using a 'Siamese' time delay neural network." *Advances in Neural Information Processing Systems,* 6, 1994.
8. Kumar, Manoj, and Pooja Yadav. "A survey on handwritten signature verification using machine learning techniques." *International Journal of Engineering Research and Technology (IJERT),* vol. 8, no. 4, pp. 50-56, 2019.
9. Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint* arXiv:1409.1556 (2014).