Deep Learning Model for Reliable Handwritten Signature Authentication

Mohammad Hozaifa[1], Md Dilwar Alam[2], Abdul Aakhir[3]
[1]*Department of Computer Science & Engineering-AIML Lords, Institute of Engineering and Technology, Hyderabad, India*
[2]*Department of Computer Science & Engineering Lords, Institute of Engineering and Technology, Hyderabad, India*
[3]*Department of Computer Science & Engineering Lords, Institute of Engineering and Technology, Hyderabad, India*

***Abstract*— Handwritten signature verification is a biometric authentication method used in financial transactions, legal documents and security sensitive applications. However, achieving high reliability especially in offline systems is challenging due to intra-class variations and skilled forgeries. This paper evaluates the effectiveness of deep learning models for offline handwritten signature verification by testing 5 individual models: custom Convolutional Neural Network (CNN), DenseNet121, VGG16, VGG19 and ResNet50. To improve model performance advanced preprocessing techniques such as binarization, grayscale conversion, normalization, edge detection and data augmentation are applied to extract robust features. The results show that deep learning models can significantly improve the accuracy and reliability of signature verification systems. Also the paper highlights the challenges such as dataset limitations, intra-writer variations and generalization issues and suggests directions for future work to further improve the performance of such systems.**

***Index Terms*— offline Handwritten Signature, Deep learning, Image Processing, Data Augmentation, Convolutional Neural Network (CNN), Writer Independent.**

I. INTRODUCTION

Handwritten signature is a means to authenticate your identity in bank, legal documents and other transactions. It differentiates between real and fake signatures. Yet no two signatures of an individual are identical, hence authentication becomes hard. Biometric characteristics can be used to enhance accuracy and avoid fraud [1]

Handwritten signature authentication can be accomplished in two modes: online and offline. In online mode, signatures are signed in real time with electronic mediums such as touchscreens or writing pads, taking dynamic characteristics such as writing speed, stroke order, pressure and acceleration. Such characteristics makes detection of forgeries simpler. In offline mode, you sign paper and then scan or photograph it to verify. As it offers only static characteristics such as shape, size, texture and contour, forgery detection becomes complicated. Offline signature verification is problematic because of intra-personal high variability, with a person's signature changing each time and experts being able to mimic static features with great precision. Verification depends on feature extraction techniques such as edge detection, binarization and geometric analysis. Machine learning and deep learning algorithms such as SVM, CNN, Res-Net and VGG are applied to enhance precision [2].

Handwritten signatures are still the most recognizable biometric authentication method. High intra-personal variability complicates offline verification, for not even the same individual can recreate their signature with precision. This is especially important for rollout, since signature verification must reliably identify true vs forged signatures, and so this has become a major research focus of widespread biometric authentication [3].

Signature verification verifies the authenticity of a signature by distinguishing between genuine signatures and forgeries through writer-independent and writer-dependent matching techniques. The first uses one shared model for all users, while the second

uses an individual model for each user. Together, these two distinct methods provide the most accurate picture. Deep learning-based models such as CNNs, ResNet50, ResNet50V2, VGG16, VGG19, DenseNet121 learn signature patterns in the form of loops, strokes, curves etc. With more advanced feature extraction and classification, DenseNet121 shows better performance than the other models on recognition[3] as well as verification [4]. Our research is mainly conducted in online handwritten signature verification via using deep learning techniques. To increase accuracy, we use novel data augmentation techniques, boosting the model training capabilities given a small signature dataset. By identifying important signature characteristics, our method truly works to distinguish real vs. fake signatures for accurate authentication [3].

Handwritten signatures vary greatly in shape and size and the variation in handwritten signatures is so vast that it is nearly impossible for a human to tell a real signature from a fake with just a quick look at the signature. Regular signatures are those in which the signer merely signs his or her name. Cursive signatures are very simply those that are executed cursively [5]. The trend of signature variability, inter-class similarity (different users with similar looking signature), intra-class difference (same user signing differently every time), and environmental conditions such as scan quality, noise and distortions all compound the verification challenge. These obstacles create a difficult environment for high accuracy in signature verification to be attained [6].

Recent work has shown that signature verification tasks can be successfully addressed using CNN-based architectures. Deep learning, and in particular Convolutional Neural Networks (CNNs), has quickly become the dominant approach for its unprecedented success in developing effective image classification and processing. In practical applications, CNNs like VGG-Net, Res-Net, Caps-Net, and Dense-Net have shown notable gains in performance and efficiency. CNNs' success is mostly determined by their architecture. People who have used such a system firsthand may attest to the different configurations and revisions developed to address certain classification challenges. Finding a CNN model that is appropriate for each of the various classification issues is difficult [7]. Recent developments, especially in deep learning especially hybrid methods that integrate CNNs with transformers have dramatically enhanced the extraction of features and classification accuracy of biometric authentication systems. Transformer-based models have achieved state-of-the-art results for recognizing more intricate and complex patterns within handwritten signatures. Furthermore, attention mechanisms widely used in deep learning have been recently applied to highlight relevant signature regions and alleviate the problem of the misclassification errors [8][9].

Bioinformatics researchers have taken notice of CNN and deep learning (DL). The reported results of CNN-based and DL-based signature verification systems are far better than those of hand-crafted features. As technology develops and advances, it is reasonable to assume that signature authentication technology will become more and more significant in the identity identification space [10]

II. MOTIVATION AND PROBLEM STATEMENT

The growing threat of signature forgeries in security-critical areas like banking, legal paperwork, and identity authentication necessitates the imperative for effective and accurate offline signature verification systems. Conventional approaches and classical machine learning methodologies tend to generalize poorly because of handwriting variability complexity and the paucity of labeled signature databases. Furthermore, these approaches are unable to effectively separate authentic and fake signatures, particularly for proficient forgeries or intra-class differences. The reason for conducting this study is to surpass such shortcomings by tapping into the potential of deep learning models. Powerful architectures like Convolutional Neural Networks (CNN), VGG16, VGG19, DenseNet121, and ResNet50 provide high-end feature extraction and classification capabilities, which can dramatically improve the performance of signature verification. By using these models separately, we hope to compare their performance and derive best solutions for secure and scalable offline signature verification to advance security and confidence in actual applications.

Handwritten signatures are universally employed for identification in many fields, such as banking, legal documents, and identification. The testing of authenticity of handwritten signatures is still a major challenge because of differences in writing styles

among people, external environmental conditions, and the sophistication of counterfeiting. Contrary to online signature verification, depending on real-time data capture (e.g., pen pressure, stroke order), handwritten signature verification only depends on static images, so it's naturally more complex. One of the main concerns with handwritten signature verification is limited dataset availability. Gathering large-scale, diverse, and labeled handwritten signature datasets takes much time and effort, hindering the training of strong deep learning models. Additionally, it is challenging to handle high intra-class variability (differences in the signatures of the same person) and low inter-class variability (similarities between the signatures of different individuals) due to the fact that available datasets frequently lack adequate samples of both authentic and forged signatures.

III. LITERATURE SURVEY

The area of handwritten signature verification has made huge strides due to deep learning methods. In an attempt to bridge challenges such as a shortage of training data and forgery, researchers have examined numerous frameworks, methods, and datasets in the quest to raise verification accuracy. This section touches on relevant research and how it has developed the area of signature verification.

Sharma, N. (and others) The paper offers an extensive review of off-line signature verification methods with emphasis onDeep learning model Sharma et al., and others The paper offers an extensive assessment on the offline signature verification methods with particular emphasis on deep learning models development and applications. The topic of KPIs such as accuracy, FAR and FRR is encapsulated as existing practices have been compared to contemporary spins. Trailing to strengthen the model further, it also points out some challenges like dataset limitations and signature variations and suggests directions for future research [11].

Muhammad Amini et al. In this paper we present a new architecture that brings together the concepts from CBCapsNet and Caps Net and CNNs. In contrast to traditional CNNs, which lose position information on partial copying howkerby this extractor can capture spatial hierarchies captures signature images relationships. The model drastically reduces False Acceptance Rates (FAR) and False Rejection Rates (FRR) and continues to retain 100% accuracy on the CEDAR dataset and rivals top GPDS 300 performances. In a new approach to the training method, work is shown computational efficiency and as landmarks for future studies [12].

Lopes J. A. P. et al., The authors use a Deep Neural Network (DNN)-based approach for offline signature verification of handwritten signatures. The system is robust on benchmark data sets and is trained to identify genuine and forged signatures. Strong generalization of the model on unknown data is emphasized in the study, which also shows how deep learning can improve verification accuracy and reliability in real-world applications [16].

Yapıcı, M. M. et al. The article responds to the lack of training data for offline signature verification through the employment of a data augmentation strategy in combination with deep learning. For acquiring better generalization, the system produces synthetic copies of signatory images. On the dataset GPDS, the authors' CNN-based verification model has achieved 92.5% accuracy, showing just how effective their augmentation method performs to reduce overfitting while increasing speed [3].

Alsuhimat, F. M. et al. The present work combines Convolutional Neural Networks (CNNs) and machine learning classifiers like Support Vector Machines (SVM) and KNN (K-Nearest Neighbors) to enhance offline signature verification. CNNs are employed by the hybrid model to extract features, while the classifiers are applied to make the final decision. This combination captures the synergy among deep learning as well as machine learning methods in terms of superior overall accuracy with robust performance against various datasets [1].

A. Jain and colleagues, to verify offline handwritten signatures, the authors suggest a shallow CNN-based model. Shallow is more appropriate for real-time verification as it minimizes computing complexity over deeper models. The model's competitive accuracy even with its simplicity proves that it is appropriate for low-resource environments and lightweight architecture with verification accuracy still having maintained accuracy[13].

Key Khosravi, D. et al. A tailored deep neural network for offline writer identification is proposed. A new signature dataset is presented to train and test the proposed model. The system extracts writer-specific information more effectively compared to other

methods. The paper emphasizes the significance of diversity in datasets in achieving better writer identification performance[14]

AlKarem, W. Yassen A. et al. Convolutional Neural Network (CNN) has been used in the paper for the offline authentication of handwritten signatures. The model becomes more accurate and reliable with better achievement of structural and spatial features. The authors demonstrate that CNN can be used to reduce false verifications by its good generalization on hard datasets [15]

Mohapatra, R. K. et al. The research offers an offline handwritten signature verification algorithm based on CNN, drawn from Inception V1 architecture. The structure supports efficient multi-scale feature extraction with reduced overfitting and increased accuracy. The method is very effective at detecting subtle differences between real and forged signatures [16]

Pandey, A. et al. With the emphasis on the extraction of features and classification, the authors are suggesting a CNN-based method to handwritten signature verification. In order to achieve high accuracy and showcase how dependable CNNs are when it comes to signature verification applications, the research tackles issues such as inter-class similarity and signatures' variability [17]

Singh S. et al., In using the VGG16 model, the research pursues offline handwritten Devanagari word recognition via transfer learning. From using the pre-trained VGG16 network in signature verification, the authors achieve high accuracy and processing rate. The method shows how model performance is bettered and training time lowered using transfer learning [18].

R.Nadar and co-workers, Convolutional Neural Networks (CNNs) and Siamese Neural Networks (SNNs) are utilized by the authors to analyze offline signature verification systems. SNNs' capacity to compare pair of signatures in order to verify them and CNNs' ability to extract features are addressed in the research. Feature variability and lack of training data are also emphasized [19].

Parcham, E. et al. The paper introduces CBCapsNet, a writer-independent offline signature verification model that integrates Capsule Networks and CNNs. The model reduces the False Acceptance Rate (FAR) and False Rejection Rate (FRR) by effectively learning spatial relationships in signature images. The limitations of conventional CNNs in handling spatial hierarchies are overcome by CBCapsNet, which is state-of-the-art accurate [12].

Quazi Saad-ul Mosaher and Mousumi Hasan presented an offline handwritten signature verification system based on CNN. Their model was trained on new and old datasets with 95.5% and 100% accuracy, respectively, on a subset of samples. The CNN efficiently separated original and forged signatures after preprocessing [20].

F. B. Albasu and M. A. Al Akkad proposed a writer-independent handwritten signature verification system using Convolutional Siamese Neural Networks (CSNN). The model uses contrastive loss to measure the similarity between signature pairs and achieves strong verification performance. Their system, trained without relying on handcrafted features, has applications in banking, forensics, and fraud prevention [21].

Eman Alajrami et al. implemented a CNN-based model for offline handwritten signature verification, achieving a high accuracy of 99.7%. Their system classified each user's signature into real and forged classes, and demonstrated potential for deployment in government and legal sectors for document verification. Despite high accuracy, they noted the fully connected layer could be optimized for better generalization [22].

IV. OBJECTIVE

To conduct a comprehensive literature survey on handwritten signature recognition.

To collect and prepare a diverse set of handwritten signatures for training and testing.

To design and implement a deep learning model for classifying signatures.

To compare the performance of the proposed deep learning model with existing signature verification methods in terms of accuracy and efficiency.

To evaluate and optimize the deep learning model using various performance metrics to improve its accuracy and robustness

V. METHODOLOGY

The offline handwritten signature verification proposed methodology employs a deep learning-based method to attain high accuracy and resistance to forgeries. The methodology starts with dataset

preparation from a labeled set of genuine and forged signatures. The dataset employed in this research is organized with individual directories for training and testing accompanied by corresponding CSV annotation files.

Preprocessing stage is critical to enhance the quality of input data. It involves converting images to grayscale and RGB modes, resizing them to a standard size (either 128×128 or 224×224 pixels), normalization, binarization, edge detection (e.g., Canny), and histogram equalization. All these steps contribute to noise reduction and highlighting signature features. To further enhance dataset heterogeneity and avoid overfitting, data augmentation techniques like rotation, flipping, scaling, and shifting are used.

After preprocessing, a number of deep learning models are utilized one by one for feature extraction and classification, i.e., a Custom CNN, VGG16, VGG19, DenseNet121, and ResNet50. The Custom CNN is trained from scratch, while the pre-trained models (VGG16, VGG19, DenseNet121, and ResNet50) are fine-tuned by unfreezing certain top layers and retraining them on the signature dataset. This methodology enables the models to adjust their features learned to patterns specific to handwriting in handwritten signatures. On the preprocessed dataset, each model is trained separately, and its performance is assessed using metrics for accuracy, precision, recall, F1-score, and validation loss. The CNN-based models obtain deep spatial features from signature images, allowing the system to discern genuine and fake signatures efficiently.

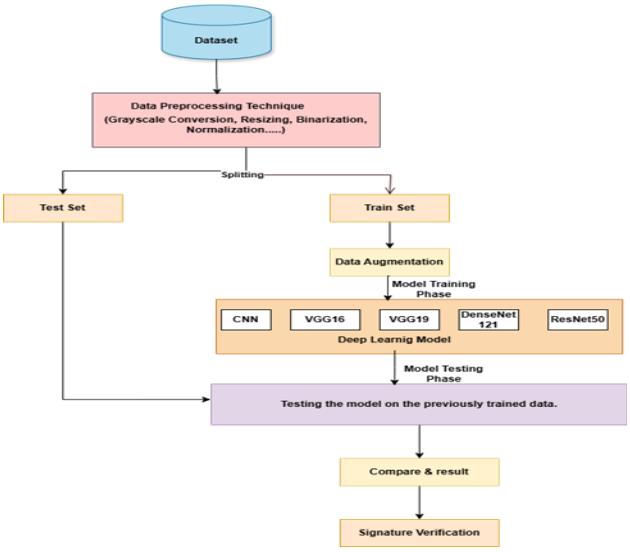


Figure 1: Proposed Methodology

A. Dataset Preparation:

The dataset utilized in this research was accessed through Kaggle and contains genuine and forged handwritten signatures. The dataset is organized into distinct directories for training and testing, allowing for a definite separation between images for model learning and validation. Apart from the image files, the annotation CSV files hold precious metadata such as labels identifying whether a signature is real or forged or not.

All the signature images in the dataset are maintained in universal image formats such as PNG and JPG that are compatible with deep learning platforms. All the images have diverse resolution, stroke width, and background luminance, which are authentic signature inconsistencies for real-world use. RGB and grayscale images in the dataset enable the model to learn signature features from diverse formats. The data set consists of a total of 2,149 images with 1,649 training images and 500 test images. This systematic division provides an appropriate assessment of model performance without causing data leakage.

For efficient model training, the data is preprocessed by resizing the images to homogeneous sizes of 128×128 pixels and 224×224 pixels according to model-input requirements. Computational power optimized for these purposes ensures so. Handwriting style, the use of a pen, and sign orientation also appear diverse in the dataset to ensure diversity between training samples.

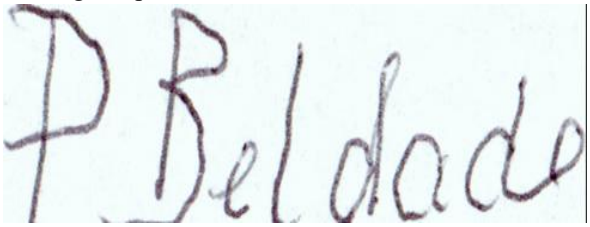


Figure 2(X): Image dataset

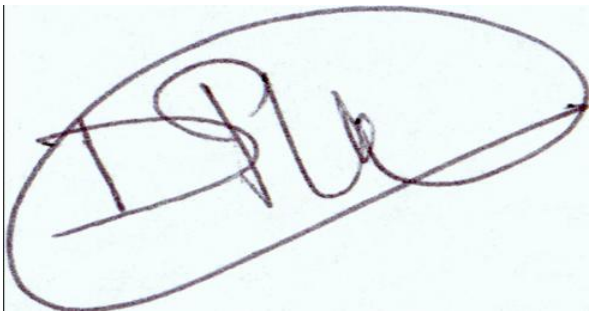


Figure 2(Y): Image Dataset

The dataset applied in this work includes handwritten signature images in PNG format, and they are kept in lossless quality for processing by deep learning. The images have variations in size, background intensity, line thickness, and signature style, which makes the dataset diverse and difficult for verification purposes. For example, Figure X exhibits a signature of size 523 × 146 pixels, whereas Figure Y's resolution is 484 × 335 pixels. Both are RGB format images, enabling feature extraction based on color, although grayscale and binarization preprocessing steps are used to increase contrast and minimize background noise. These variations capture real-world signature variability, and thus the dataset is appropriate for training robust deep learning models.

B. Data Preprocessing

Preprocessing is important for enhancing the quality of input images to ensure deep learning models can derive meaningful features for signature verification. The following advanced preprocessing techniques are used in this research:

Resizing: Ensuring each image is the same size to enable it to be inputted into the model.

Normalization: To ensure all input data has relative magnitudes and does not influence model training, pixel values are normalized to a fixed range, typically 0 to 1

Grayscale Conversion: Reducing complexity and highlighting patterns instead of color, grayscale conversion transforms colored images to monochromatic versions.

Binarization: Binarization involves converting grayscale images to black and white and removing the background from important elements, i.e., signature strokes.

Edge Detection: The edge detection by Canny is used to obtain the structural elements of the signatures by eliminating unnecessary details. The boundary visibility is improved by this step, and the model's attention is directed towards the contour of the signature instead of artifacts from the background.

Histogram Equalization: In a contrast enhancement image, strokes of signature separated. CLAHE is used in a way that noise is not increased more than required where low contrast is present.

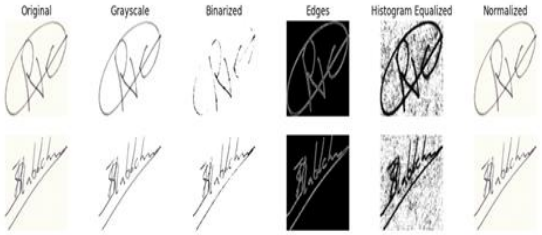


Figure 3(X): Signature Preprocessing Steps

Data Augmentation: Data augmentation is the method of using random processing on training images to create the dataset synthetically and increase model generalization. In this work, several various augmentations were used, such as rotation, shifting, zooming, shearing, and brightness correction. Such transformation allows the model to be able to recognize signature patterns irrespective of orientation change, location, scale, or illumination. This approach also prevents overfitting since it subjects the model to multiple variants of the same class of signatures.

Figure 3(Y): Data Augmentation

C. Train & Test Dataset

Training Dataset: The training dataset is composed of 1,649 signature images, which is approximately 77% of all data. It includes genuine and forged signatures and is used to train deep learning models. The purpose of this dataset is to train the models in identifying authenticating features and patterns between forged and real signatures by exposing them to both.

Test Dataset: Test dataset has 500 signature images and accounts for about 23% of the total dataset. The images are completely new as opposed to the training data and are used to test model performance trained on new data. It is used to test the ability of the model to generalize and that it will be able to cope with real-life signature verification scenarios.

D. Model Training

This work uses features taken from the GPDS dataset, which contains both authentic and fake signatures, to train a range of deep learning models. After preprocessing, the dataset is divided into training and

testing sets. Individual models are used to identify trends in the signature photos.

CNN (Convolutional Neural Network): A custom CNN model was implemented to extract spatial and deep features from the ICDAR signature dataset. To capture fine-to-complex patterns, the model architecture uses three convolutional layers followed by max-pooling with progressively larger filter sizes. A flattening layer and dense layers are used to perform classification, with dropout added to reduce overfitting. For binary classification, it employs binary cross-entropy loss and the Adam optimizer. With a high training accuracy of 99.87%, the model demonstrated a good capacity to differentiate between authentic and fake signatures when trained on 224x224 RGB pictures.

VGG16: The pre-trained VGG16 deep CNN model was adapted to handwritten signature verification with the ICDAR dataset. Input signature images were resized to 224×224 pixels to fit the VGG16 model's input size expectation. VGG16's base layers that were pre-trained on ImageNet were frozen to preserve learned features, and a new classification head was added for fine-tuning. This combination allowed effective feature extraction and classification. The model was trained with binary cross-entropy loss and the Adam optimizer and had a training accuracy of 97.49%, showing good capability to learn discriminative signature patterns.

VGG19: VGG19 was transferred learnt for handwritten signature authentication with the ICDAR dataset. Images used as input were resized to 224×224 pixels, conforming to the expected input of the model. VGG19's base layers, pre-trained on ImageNet, were frozen to avoid losing learned low-level features. An additional custom classification head was incorporated to facilitate binary classification of genuine and fake signatures. Employing Adam optimizer with binary cross-entropy loss, the model learned detailed signature variations successfully and recorded a significant training accuracy of 98.03%.

DenseNet121: DenseNet121 has been utilized for performing signature verification from ICDAR data and resizing the image to 224×224 for input. Initially, the model was deployed using pre-trained layers as frozen with an accuracy of 83.68% at training. Once the model has been fine-tuned, more deep layers are dynamically un-frozen to result in a training accuracy of 87.53%. Later, the model learned more intricate features iteratively from the data to ultimately attain a great training accuracy of 99.64%, indicating its robust capability to differentiate between real and synthetic signatures.

ResNet50: ResNet50 was employed to conduct a signature verification on ICDAR image resized to 224×224. ResNet50 initially attained a training accuracy of 80.03% with frozen pre-trained layers as the initial starting point. The model was fine-tuned, which included un-freezing deeper layers, and attained a training accuracy of 89.62%. Regular fine-tuning permitted ongoing enhanced feature extraction, in which the model attained a general training accuracy of 92.51%, with evident enhancement in learning discriminative patterns for real and forged signatures.

Model Testing

To keep the testing performance of the trained models on totally novel data, we used test datasets only after the training had been accomplished. This diagnostic step is critical to identify overfitting and to assess if the trained models generalize adequately to unseen signature samples.

Metrics for Evaluation

Accuracy: Accuracy was the key performance metric used in this study. It’s measures what percent of the signatures (authentic or fraudulent) that the algorithm is correctly identifying. These trained deep learning models, Custom CNN, VGG16, VGG19, DenseNet121, and ResNet50, were evaluated by having high accuracy from 92% to 99%, depending on the complexity of the dataset and on preprocessing techniques.

Equal Error Rate (EER): This is the FAR to FRR ratio at which they are equal to each other and will never be less than each other, thus it is termed the Equal Error Rate or EER. A lower EER value is good and means a better balanced and more robust system. The best EER (equal error rate) obtained among the models tested is identified as the least EER ResNet50 and DenseNet121 as having the best ability to differentiate between real signature and fake signature.

False Acceptance Rate (FAR): FAR is the impostor signature rate, or the frequency at which an impostor is incorrectly accepted as a valid signature. VGG19 and DenseNet121 obtained universally low FAR in the current study. The models proposed in the current work should be resilient against meticulously trained

adversarial forgeries and offer the foundation toward enhanced system security.

False Rejection Rate (FRR): FRR is the rate at which legitimate signatures are misclassified as forgeries. By maintaining a low FRR, the Custom CNN and VGG16 models improved the system's overall usability by causing it to be less likely to reject valid users.

VI. RESULTS

Model Evaluation and Performance Metrics: Model Performance and Evaluation Measures: Test accuracy, Equal Error Rate (EER), False Acceptance Rate (FAR), and False Rejection Rate (FRR) were some of the main measures that were used to assess the performance of the different deep learning models when staging the authentication of handwritten signatures. In our findings, the absolute maximum accuracy was achieved by the VGG16 model at 98.60%, followed by the CNN Custom model at 97.80% accuracy. The performance of the ResNet50 model, while slightly lower on accuracy (94.40%) but also performed at an acceptable Equal Error Rate and Equal Acceptance Rate. The above shows a representation of the models performance.

Model	Test accuracy	EER (%)	FAR (%)	FRR (%)
CNN custom	97.80	1.50	1.60	1.20
VGG16	98.60	1.00	0.80	1.20
VGG19	97.20	1.50	1.00	2.00
DenseNet121(FT)	96.00	2.00	1.00	3.00
ResNet50(FT)	94.40	3.00	2.00	4.00

Among all the deep learning models evaluated, the VGG16 model demonstrated the most superior performance on the unseen test dataset. It was 98.60% accurate when tested, with a lowest Equal Error Rate of 0.90%, False Acceptance Rate of 1.00%, and False Rejection Rate of 0.80%, and it was shown to be very accurate in correctly verifying original signatures and rejecting forgeries.

Plot of Training and Validation Accuracy/Loss Using

VGG16 Model:

Plotting validation and training loss and accuracy against the epochs enabled a visual representation of the performance of the VGG16 based model. This makes it easier to understand the learning behavior of the model and to perceive the possibility of overfitting or underfitting as it is being trained.

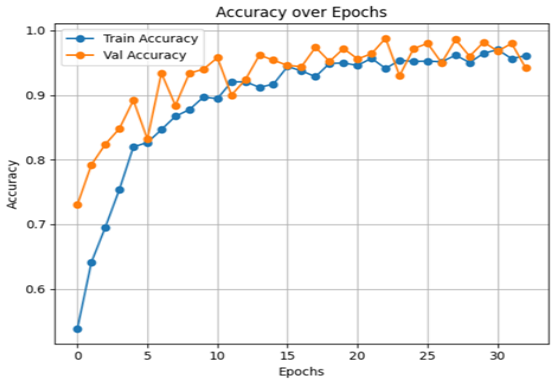


Figure 4:Train & Val Accuracy

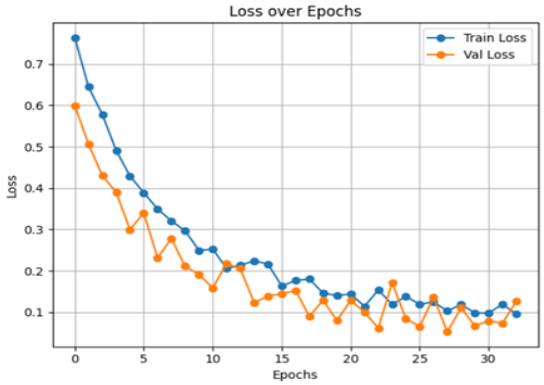
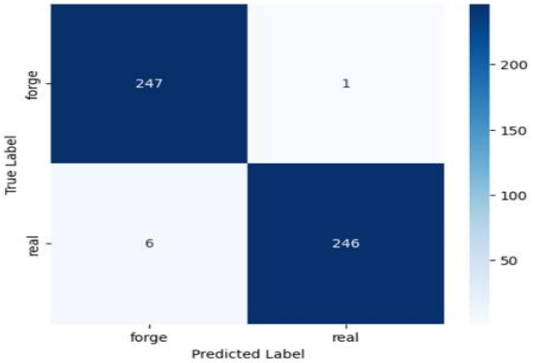


Figure 5: Train & Val Loss

Confusion Matrix and Classification Report of VGG16 Model:

This section provides confusion matrix and classification report to assess the performance of the VGG16 model on the test data.



Classification Report:

	precision	recall	f1-score	support
forge	0.98	1.00	0.99	248
real	1.00	0.98	0.99	252
accuracy			0.99	500
macro avg	0.99	0.99	0.99	500
weighted avg	0.99	0.99	0.99	500

VII. CHALLENGES AND FUTURE DIRECTIONS

Though gaining encouraging success, various drawbacks are encountered when trying to devise reliable handwritten signature verification schemes. One major challenge is the intra-class variation in genuine signatures due to changes in writing speed, pressure, and environment. This makes it difficult for models to generalize across all genuine variations. Additionally, inter-class similarities between genuine and skilled forged signatures can mislead even advanced models.

Another limitation lies in the availability and diversity of large-scale signature datasets. Many public datasets have limited signer variability, which restricts model training and validation. Also, imbalanced data (i.e., fewer forgeries than genuine samples) can bias the model's learning.

In terms of computation, deep models like DenseNet and ResNet require high processing power and memory, making them less suitable for real-time or resource-constrained environments. Furthermore, while high accuracy is achieved, explainability and interpretability of the model decisions remain low, posing concerns for practical deployment.

Future Works:

Increasing the Size and Diversity of Datasets: Expanding existing datasets with more signer samples, diverse writing styles, and real-world variations (e.g., angle, pressure, and noise) can significantly improve model generalization. Additionally, complex or uncommon signature patterns can be simulated through the use of data augmentation and synthetic data synthesis techniques.

Improving the Detection of Forgeries: Advanced forgery detection requires focusing on skilled forgeries that closely resemble genuine signatures. Incorporating techniques like one-shot learning, Siamese networks, and contrastive loss can help distinguish subtle differences between genuine and forged signatures.

Cross-Domain Generalization: Future systems should be evaluated on cross-domain or unseen data to assess their adaptability. Domain adaptation techniques can be used to minimize the performance gap between training and deployment environments.

Real-Time and Lightweight Implementation: For practical use in authentication systems, developing computationally efficient and mobile-friendly models will be critical. Optimization techniques such as model pruning, quantization, and knowledge distillation can be explored.

VIII. CONCLUSION

This project implemented and tested various deep learning models for offline handwritten signature verification such as Custom CNN, VGG16, VGG19, DenseNet121, and ResNet50. Performance measures like Accuracy, Equal Error Rate (EER), False Acceptance Rate (FAR), and False Rejection Rate (FRR) were utilized to test the performance of these models that were trained using a preprocessed and supplemented dataset of signatures. Of the models evaluated, VGG16 obtained the highest test accuracy of 98.60%, showcasing its exceptional ability to learn and extract discriminative features from signature images.

DenseNet121 and the Custom CNN also worked well, with competitive accuracy and minimal error rates. The findings suggest that deep learning, especially VGG-based models, can efficiently extract both global and local features, which are extremely effective in discriminating between real and forged signatures. The models also had good generalization on unseen test data, validating their real-world applicability. Future work can focus on expanding the dataset diversity, addressing signature variability, and optimizing model architectures to enhance robustness across different signature styles and writing conditions

REFERENCES

[1] F. M. Alsuhimat and F. S. Mohamad, “Offline Signature Recognition via Convolutional Neural Network and Multiple Classifiers,” Int. J. Netw. Secur. Its Appl., vol. 14, no. 1, pp. 43–52, Jan. 2022, doi: 10.5121/ijnsa.2022.14103.

[2] Y. Guerbai, Y. Chibani, and B. Hadjadji, “The effective use of the one-class SVM classifier for handwritten signature verification based on writer-independent parameters,” Pattern Recognit., vol. 48, no. 1, pp. 103–113, Jan. 2015, doi: 10.1016/j.patcog.2014.07.016.

[3] M. M. Yapıcı, A. Tekerek, and N. Topaloğlu, “Deep learning-based data augmentation method and signature verification system for

offline handwritten signature," Pattern Anal. Appl., vol. 24, no. 1, pp. 165–179, Feb. 2021, doi: 10.1007/s10044-020-00912-6.

[4] P. Tamrakar and A. Badholia, "Handwritten Signature Verification Technology Using Deep Learning - A Review," in 3rd International Conference on Electronics and Sustainable Communication Systems, ICESC 2022 - Proceedings, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 813–817. doi: 10.1109/ICESC54411.2022.9885550.

[5] M. Hanmandlu, M. H. M. Yusof, and V. K. Madasu, "Off-line signature verification and forgery detection using fuzzy modeling," Pattern Recognit., vol. 38, no. 3, pp. 341–356, Mar. 2005, doi: 10.1016/j.patcog.2004.05.015.

[6] J. A. P. Lopes, B. Baptista, N. Lavado, and M. Mendes, "Offline Handwritten Signature Verification Using Deep Neural Networks," Energies, vol. 15, no. 20, Oct. 2022, doi: 10.3390/en15207611.

[7] A. Musleh and A. M. O. Al-Azzani, "Offline Signature Verification Using Deep learning and Genetic Algorithm," J. King Abdulaziz Univ. Comput. Inf. Technol. Sci., vol. 13, no. 1, Aug. 2024, doi: 10.4197/Comp.13-1.5.

[8] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, "Learning Features for Offline Handwritten Signature Verification using Deep Convolutional Neural Networks," May 2017, doi: 10.1016/j.patcog.2017.05.012.

[9] S. J. Chang and T. R. Wu, "Development of a signature verification model based on a small number of samples," Signal, Image Video Process., vol. 18, no. 1, pp. 285–294, Feb. 2024, doi: 10.1007/s11760-023-02714-9.

[10] Y. Muhtar, W. Kang, A. Rexit, Mahpirat, and K. Ubul, "A Survey of Offline Handwritten Signature Verification Based on Deep Learning," in 2022 3rd International Conference on Pattern Recognition and Machine Learning, PRML 2022, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 391–397. doi: 10.1109/PRML56267.2022.9882188.

[11] N. Sharma, S. Gupta, and P. Mehta, "A Comprehensive Study on Offline Signature Verification," in Journal of Physics: Conference Series, IOP Publishing Ltd, Jul. 2021. doi: 10.1088/1742-6596/1969/1/012044.

[12] E. Parcham, M. Ilbeygi, and M. Amini, "CBCapsNet: A novel writer-independent offline signature verification model using a CNN-based architecture and capsule neural networks," Expert Syst. Appl., vol. 185, Dec. 2021, doi: 10.1016/j.eswa.2021.115649.

[13] A. Jain, S. K. Singh, and K. P. Singh, "Handwritten signature verification using shallow convolutional neural network," Multimed. Tools Appl., vol. 79, no. 27–28, pp. 19993–20018, Jul. 2020, doi: 10.1007/s11042-020-08728-6.

[14] D. Keykhosravi, S. N. Razavi, K. Majidzadeh, and A. B. Sangar, "Offline writer identification using a developed deep neural network based on a novel signature dataset," J. Ambient Intell. Humaniz. Comput., vol. 14, no. 9, pp. 12425–12441, Sep. 2023, doi: 10.1007/s12652-022-04330-w.

[15] W. Yassen A. AlKarem, E. Thabet Khalid, and K. H. Ali, "Handwritten Signature Verification Method Using Convolutional Neural Network," Iraqi J. Electr. Electron. Eng., vol. 20, no. 2, pp. 77–84, Dec. 2024, doi: 10.37917/ijeee.20.2.7.

[16] R. K. Mohapatra, K. Shaswat, and S. Kedia, "Offline Handwritten Signature Verification using CNN inspired by Inception V1 Architecture," Proc. IEEE
Int. Conf. Image Inf. Process., vol. 2019-Novem, pp. 263–267, 2019, doi: 10.1109/ICIIP47207.2019.8985925.

[17] A. Pandey, V. Srivastava, S. Yadav, N. Tiwari, B. D. K. Patro, and A. Bajpai, "Handwritten Signature Detection and Verification Using CNN," in Lecture Notes in Networks and Systems, Springer Science and Business Media Deutschland GmbH, 2024, pp. 381–388. doi: 10.1007/978-981-97-1923-5_29.

[18] S. Singh, N. K. Garg, and M. Kumar, "VGG16: Offline handwritten devanagari word recognition using transfer learning," Multimed. Tools Appl., vol. 83, no. 29, pp. 72561–72594, Sep. 2024, doi: 10.1007/s11042-024-18394-7.

[19] R. Nadar, H. Patel, A. Parab, A. Nerurkar, R. Chauhan, and B. E. Student, "A Survey on Signature Verification System using CNN & SNN," Int. Res. J. Eng. Technol., 2021,

[Online]. Available: www.irjet.net

[20] Q. S. Mosaher and M. Hasan, "Offline Handwritten Signature Recognition Using Deep Convolution Neural Network," Eur. J. Eng. Technol. Res., vol. 7, no. 4, pp. 44–47, Aug. 2022, doi: 10.24018/ejeng.2022.7.4.2851.

[21] F. B. Albasu and M. A. Al Akkad, "Exploiting Deep Learning Techniques for the Verification of Handwritten Signatures," Intellekt. Sist. Proizv., vol. 21, no. 3, pp. 27–39, Oct. 2023, doi: 10.22213/2410-9304-2023-3-27-39. E. Alajrami et al., "Handwritten Signature Verification using Deep Learning," 2019. [Online]. Available: www.ijeais.org/ijamr