

Intelligent Signature Verification System Using Hybrid Deep Learning with WOA-GGO Driven Feature Selection

D.N. Vasundhara

Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology

Y. Ramadevi

Chaitanya Bharathi Institute of Technology

Bibhuprasad Sahu

bibhuprasad@sithyd.siu.edu.in

Symbiosis Institute of Technology, Hyderabad Campus, Symbiosis International University

CH V K N S N Moorthy

Vasavi College of Engineering

Shilpa S. Harnale

Bheemanna Khandre Institute of Technology

Abdul Aleem

Galgotias University

Davesh Kumar Sharma

Krishna Institute of Engineering & Technology (KIET)

Mukesh Kumar Tripathi

Krishna Institute of Engineering & Technology (KIET)

Article

Keywords: Signature Verification, Median Filtering, ResNet features, Bi-LSTM, RNN

Posted Date: October 22nd, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10575571/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Intelligent Signature Verification System Using Hybrid Deep Learning with WOA-GGO Driven Feature Selection

D.N. Vasundhara¹

¹Department of Computer Science and Engineering,
Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology,
Hyderabad, India
vasundhara_d@vnrvjiet.in

Y. Ramadevi²

Department of Artificial Intelligence & Machine Learning,
Chaitanya Bharathi Institute of Technology, Hyderabad, India

yramadevi_cseaiml@cbit.ac.in

Bibhuprasad Sahu ^{*3},

Symbiosis Institute of Technology, Hyderabad Campus, Symbiosis International University, Hyderabad, India
bibhuprasad@sithyd.siu.edu.in

CH V K N S N Moorthy⁴

Department of Mechanical Engineering
Vasavi College of Engineering, Hyderabad, India
krishna.turbo@gmail.com

Shilpa S. Harnale⁵

⁵Department of Computer Science & Engineering
Bheemanna Khandre Institute of Technology, Bidar, India
shilpasharnale@gmail.com

Abdul Aleem⁶

⁶Department of Computer Science & Engineering

Galgotias University, Greater Noida, India
abdul.aleem@galgotiasuniversity.edu.in

Davesh Kumar Sharma⁷
Department of Computer Science & Engineering (AIML),
Krishna Institute of Engineering & Technology (KIET), Delhi-NCR, India
daves.h.sharma@kiet.edu

Mukesh Kumar Tripathi*⁸
Department of Computer Science & Engineering (Artificial intelligence),
Krishna Institute of Engineering & Technology (KIET), Delhi-NCR, India
Mukesh.kumar@kiet.edu

Abstract- The paper-based signature verification continues to be important despite increasing adoption of digital repositories, creating a demand for accurate and reliable signature verification systems. Biometric authentication provides a secure and efficient solution compared to conventional verification methods such as identity cards and magnetic strip cards. This paper proposed a hybrid signature verification framework to improve the verification accuracy through effective feature selection and score level fusion. The framework consists of four main stages. First, preprocessing is performed using an improved median filtering process and image resizing technique to obtain the high quality signature images. Next ResNet, global and local features are extracted from the preprocessed images. In the feature selection stage, a hybrid Walrus Optimization Algorithm integrated with Greylag Goose Optimization (WOA-GGO) is used to identify the optimal features. Finally, signature verification was performed using the hybrid deep learning model combining RNN and Bi-LSTM. An improved score-level fusion technique integrates the outputs of both classifiers to produce the final verification result. The proposed framework was evaluated using the GPDS 1-150 and CEDAR datasets. Experimental result shows

that the RNN + Bi-LSTM model achieved maximum accuracy values of 0.973 and 0.971 performing better than LinkNet, LSTM, EfficientNet, Bi-GRU, LeNet and Bi-LSTM. The effectiveness of the proposed framework was also validated through various performance and statistical analyses.

Keywords: Signature Verification, Median Filtering, ResNet features, Bi-LSTM, RNN.

1. Introduction

Signature identification aims to determine the identity of the signer by matching an unknown signature to a database of known signatures. Signature verification, on the other hand, seeks to confirm the authenticity of the signature, comparing it to a reference signature of the same individual to check if it's genuine. Identification is used to find out who signed, while verification ensures the signature is genuine. Both processes are essential in different contexts, such as identity verification and fraud prevention. In many financial firms, biometrics was essential for a person's identity verification, and until recently, signatures were the most used method [1]. Over the past several decades, biometric-based authentication has become a growing field of research [2]. Body measurements were the primary emphasis of biometrics, although as the field evolved and became more necessary for authentication, it also began to incorporate other biometric features associated with the human body [3]. Despite the availability of numerous biometric authentication mechanisms, handwritten signatures remain the most popular in today's society [4]. Verifying the authenticity of the signatures is essential for differentiating between authentic and fraudulent ones. As a result, one of the many active study areas is the signature verification challenge [5] [6].

Governmental and private financial firms have been using signature verification systems to confirm a person's identity based on the handwriting on various accounts, records, and documents [7] [8]. The majority of the research on offline signature verification has been focused on

developing a reliable depiction of signatures [9]. The conventional methods depend on handcrafted features, and yet no extractors of handcrafted features are especially relevant for signature validation that is currently being developed [10] [11]. Offline signature verification, however, presents a number of difficulties because of various factors, such as a lack of data related to online signature verification, the absence of dynamic features (such as pressure, the altitude of the pen, etc.) in online signatures, as well as variability in the signature of the identical person due to multiple reasons, such as ageing, illness, or normal variation following a specific number of repetitions [12] [13].

For offline signature verification, a variety of approaches have been put forth; the majority of them revolve around feature extraction, depiction, and comparison of similarities and distance metrics [14] [15]. Numerous identifiers relying on graphology, computer vision, signal processing, and so forth have been proposed for feature extraction, including BoVW, directional characteristics of signature contours, VLAD, surrounded features, local granulometric size distributions, Fisher vector, as well as energy of Curvelet coefficient [16] [17]. Furthermore, DTW and RNN are the major foundations of the currently used online signature techniques; however, for the majority of these techniques, a significant amount of training data is needed to operate as expected [18] [19].

Secured authentication is essential since signatures are the biometric attribute and are most commonly accepted by many law enforcement officials and agencies for verification purposes, particularly in financial institutions, legal transactions, etc [20] [21]. These are some of the applications where signature verification works out [22]. Moreover, many transactions are happening in the digital age where organizations, like banks etc., demand handwritten signature verification [23]. In these circumstances, the signature verification procedure should not only be

precise and safe but also quick, allowing us to do real-time verification [24]. Additionally, document authentication, credit card transactions, check verification, attendance recording, and forensics are the most pertinent applications of signature verification [25]. Because of its vast importance, many researchers have done research on signature verification to aid humanity and society [26]. However, their efforts in implementing an efficient signature verification framework may not resemble their thoughts on supporting humanity and society [27]. Existing signature verification systems often struggle with intra-class variability, such as differences in pressure, stroke order, or style across genuine signatures from the same individual. Many rely heavily on handcrafted features, which lack robustness and generalizability across datasets. Deep learning approaches have improved performance but frequently overlook the complementary value of combining local and global features. Additionally, existing models tend to use simple score-level fusion strategies that may not effectively leverage the strengths of multiple classifiers. These limitations lead to reduced accuracy, especially when verifying skilled forgeries. The motivation for this work stems from the need to build a more robust and discriminative verification framework. By integrating deep features (via ResNet), local-global feature fusion and an improved hybrid model (RNN + Bi-LSTM), the proposed system addresses these gaps. Enhanced score-level fusion further ensures more reliable verification outcomes. Thereby, the role of improvements proposed in this deep hybrid signature verification framework proves its efficiency significantly over state-of-the-art approaches. Those improvements that are part of ensuring the proposed framework's efficiency are listed below.

The main contributions are

- Introduces a novel signature identification framework in which preprocessing is carried out by an improved median filtering process and image resizing technique to obtain noise-free and quality preprocessed images.
- Integrates ResNet for deep feature extraction with both global and local feature analysis. Our framework provides a richer and more nuanced representation of signatures, which enhances the accuracy of the identification process.
- A hybrid feature selection method based on the integration of the Walrus Optimization Algorithm and Greylag Goose Optimization (WOA-GGO) is proposed to select the optimal features.
- Proposes a hybrid model which comprises Deep Learning (DL) approaches, including improved Recurrent Neural Network (RNN) and Bidirectional Long Short-Term Memory (Bi-LSTM) for the identification of novel signatures and an improved score level fusion is employed to fuse intermediate scores resulting from the classifiers. This model leverages sequential learning and bidirectional context to effectively capture the intricate patterns in signatures.

Further, the process involved in the proposed deep hybrid signature verification is categorized into five sections. The existing works are reviewed in Section 2, the overall implementation is described in Section 3, experimental analyses are discussed in Section 4 and the conclusion is described in Section 5.

2. Literature Review

In 2024, Zheng *et al.*, [40] has proposed an online handwritten signature verification algorithm, which mainly uses the CAE-MV-based feature dimensionality reduction method to

compress and select features of the original signature data to construct a signature feature set. Through the analysis of the verification results on two publicly available signature database, MCYT-100 and SVC-2004, the proposed algorithm improved the accuracy and reduced the False Acceptance Rate (FAR) and False Rejection Rate (FRR). The average verification accuracy reached 98.92% which was superior to the current mainstream online signature verification frameworks, demonstrated the effectiveness and efficiency of the proposed framework.

In 2023, Alsuhimat *et al.*, [7] has introduced a hybrid method for extracting features from signature images, wherein Convolutional Neural Network (CNN) and Histogram of Oriented Gradients (HOG) were used, followed by the feature selection algorithm (Decision Trees) to identify the key features and finally the CNN and HOG methods were combined. The experimental findings indicated that suggested model executed quite satisfactorily in terms of efficiency and predictive ability obtaining accuracy of (95.4%, 95.2% and 92.7%) with UTSig dataset and (93.7%, 94.1% and 91.3%) with CEDAR dataset.

In 2025, Sanchez-Nielsen *et al.*, [41] has introduced Dual-AngleNet, an interpretable and lightweight CNN-LSTM model that authenticates of pen-tip tilt dynamics. The model integrated azimuth and altitude angles for robust spatiotemporal feature learning. Experimental results on a dataset of 600 signature samples confirmed the model's superior accuracy (97.60%) and resilience (2.18% EER) against forgery compared to existing methods, while maintaining efficient inference suitable for edge deployment.

In 2026, Hao *et al.*, [42] has proposed Dual-AngleNet, an interpretable lightweight framework for online signature verification that leverages pen-inclination dynamics as a discriminative biometric feature. The model achieved state- of-the-art performance (97.69% accuracy, 2.18% EER) through the integration of azimuth and altitude angles within a hierarchical CNN-LSTM

architecture. Ablation studies confirmed a 5.38% accuracy gain from angular features, while SHAP analysis revealed their dominant contribution (77.7%) to verification decisions, ensuring transparency.

In 2025, Diaz *et al.*, [43] has proposed a new set of features based on the dynamics of online signatures. These new features were inferred through a Lagrangian formulation, which obtained the sequences of generalized coordinates torques for 2D and 3D robotic arm models. By combining kinematic and dynamic robotic features, the results demonstrated their significant effectiveness for online automatic signature verification and achieved state-of-the-art performance when integrated into deep learning models.

In 2025, Moises *et al.*, [44] has introduced a neural network model to estimate the kinematic and dynamic features of a robotic arm. The first approach involved using a physical UR5e robotic arm to reproduce a signature while capturing those parameters over time. The second method, a cost-effective approach, used a neural network to estimate the same parameters. The findings demonstrated a simple neural network model can extract effective parameters for signature verification. Training the neural network with the MCYT300 dataset and OnOffSigBengali-75 confirm the models generalization capability.

In 2024, Tehsin *et al.*, [14] designed to enhance the discrimination of inter-class and intra-class differences. CNN architectures proposed framework consistently achieved superior performance, particularly attributed to the triplet loss function ability to optimize inter-class disparities while minimizing intra-class variations. ResNet marginally outperformed methodology on one dataset, this approach demonstrated remarkable accuracy, surpassing 85% in discriminating between genuine and forged signatures across multiple datasets.

In 2026, De Moura *et al.*, [45] introduced ProtoSig, a novel training data generation strategy for handwritten signature verification. The method applied a summarization process to the development data to generate prototypical signatures that served as negative samples, enabling a deterministic selection that replaces arbitrary random forgery sampling. It achieved comparable or higher accuracy with a smaller training set, yielding substantial scalability benefits including over 98% reduction in training time and up to two orders of magnitude lower computational cost while strengthening data privacy and promoting fairness in offline HSV systems.

In 2023, Rokade *et al.*, [28] has developed a framework called CNN for signature verification. This work evaluates the impact of data augmentation on CNN performance for handwritten signature verification, demonstrating how augmentation techniques like rotation, scaling, and flipping can improve model robustness and generalization when working with limited signature data. The findings indicate that there is an improvement in all evaluated metrics when data augmentation is applied. Particularly, the model's test accuracy increased by 10.68%, and the F1 score increased from 73.08% to 88%. While data augmentation enhances performance, it may not fully address challenges posed by intra-class signature variations and CNNs' lightweight architecture could still struggle with highly complex or noisy signature patterns.

In 2020, Alajrami *et al.*, [29] has established a framework for multiscript recognition and handwritten signature verification. The aim of this work is to create and evaluate a quantitative experimental method for handwritten signature and multiscript pattern verification. To ascertain the authenticity of the signature, the system trains a model utilizing input data and output labels using a supervised learning methodology. The system uses machine learning methods, such as support vector machines (SVM) and neural networks, for training. In order to create a reliable system that can adapt well to various handwriting inputs, data collection is essential. Overall, the

system's accuracy rates in multiscript verification and signature verification were 92% and 89%, respectively. However, the model's complexity may lead to slower processing times, and its performance might be limited by the availability of diverse training datasets, potentially struggling with signatures or scripts not well-represented in the training phase.

In 2024, Ishfaq *et al.*, [30] has developed hybrid method for verifying people's signatures that combines a Vision Transformer with pre-trained deep learning models MobileNetV2 and ResNet-18 to accommodate different writing styles. A fixed-size representation of the retrieved features is then obtained by applying adaptive pooling and flattening algorithms, which makes it easier for later layers to analyze the data. For higher-level feature processing and classification, the vision transformer architecture which consists of linear layers with relu activation functions, dropout layers, and batch normalization levels is then used. Predictions for signature verification are produced by the model's output layer, which uses softmax activation to transform raw output scores into probability distributions across signature classes.

In 2025, Zheng *et al.*, [31] has established the hybrid Transformer and Convolution Signature Network (HTCSigNet) to extract multi-scale features from signatures. Specifically, HTCSigNet is an innovative framework consisting of two parts: a transformer and CNN-based blocks. Global and local features are extracted from signatures using these components, respectively. The CNN-based block comprises three modules: spatial and channel reconstruction convolution (SCConv) that improves model generalization by focusing on more distinctive micro-deformation features while reducing attention to common features; space-to-depth convolution (SPD-Conv) that improves feature learning capability by precisely focusing on signature strokes; and a convolution module that extracts the shape, morphology of specific strokes, and other local features from signatures. In the transformer-based block, global properties including general direction, layout,

and overall shape are extracted from signatures using the Vision Transformer (ViT). After the feature learning stage, Writer-Dependent (WD) and Writer-Independent (WI) verification systems are constructed to evaluate the performance of the created HTCSigNet.

The summary of features, limitations, datasets used, training and testing samples of existing research are presented in Table 1.

Table 1: Features, Limitations, Datasets, Training and Testing samples Found in Conventional Signature Verification Systems

Author	Approaches	Features	Limitations	Dataset Used	Training and Testing Data
Zheng <i>et al.</i> , [40]	DWSCNN	It reduces the number of neural network parameters	Performance is sensitive to hyperparameters settings	MCYT-100 and SVC-2004	-
Alsuhimat <i>et al.</i> , [7]	Hybrid deep learning	Hybrid deep learning model with Decision	May require longer training time because	UTSig and CEDAR	50.77% is used for training

		Tree enhances the signature verification	of the hybrid feature extraction process		and 49.23% is used for testing
Sanchez-Nielsen <i>et al.</i> , [41]	TrOCR	It provides a scalable and privacy-preserving signature verification framework	Requires large datasets for effective fine-tuning and high accuracy	ID and FD	90.48% is used for training and 98.91% is used for testing
Hao <i>et al.</i> , [42]	Dual-AngleNet	It provides lightweight and interpretable signature authentication for edge devices.	Validated on a limited dataset of 600 signature samples	SVC2004 Task2	-
Diaz <i>et al.</i> , [43]	Deep Learning models	It improves verification using kinematic and dynamic robotic features	Requires complex feature extraction using robotic arm models	DeepSignDB and SVC2021 EvalDB	-

Moises <i>et al.</i> , [44]	Neural Network Model	It improves signature verification using dynamic features	Requires neural network training for parameter estimation	MCYT300	-
Tehsin <i>et al</i> [14]	tSSN	Triplet loss Siamese similarity network with Manhattan distance for offline signature verification	Performance may decrease when applied to unseen signature styles and real-world datasets.	4NSigComp2012, SigComp2011, 4NSigComp2010 and BHsig260	-
De Moura <i>et al.</i> , [45]	ProtoSig	Generates informative prototypical negative samples for efficient and accurate offline signature verification	Its performance in writer-independent verification remains unverified	GPDS Synthetic, CEDAR and MCYT-75.	-
Rokade <i>et al.</i> [28]	CNN	It provides empirical evidence that augmentation	It lacks a detailed analysis of other key aspects, such as model explainability	GPDS dataset	240 samples for training

		helps improve the generalization ability of models.	and real-time execution under constrained computational environments.		and 300 for testing.
Alajrami, <i>et al.</i> [29]	SVM	Because of its strong performance on signature recognition tasks, real-time applications with less processing power may be possible.	While the system aims to provide high precision, the potential for false positives or false negatives may still exist, especially when signatures exhibit significant variability or are forged using advanced techniques.	GPDS-960 Dataset	400 is taken for training, and 100 is used for testing
Ishfaq <i>et al.</i> [30]	Hybrid vision transformer	It utilizes the self-attention mechanism of ViTs to effectively capture global contextual	The model's computational complexity and training time are relatively high due to the transformer	Bhsig-Bengali, Bhsig-Hindi, Cedar, UTsig-Persian datasets	520 is taken for training and 280 is for testing.

		information from signature images.	architecture, which may hinder deployment on resource-constrained devices.		
Zheng <i>et al.</i> [31]	HTCSigNet	It employs multi-scale feature fusion to enhance discrimination between genuine and forged signatures.	The model's performance under varying image quality and signature distortions is not thoroughly investigated.	GPDSsynthetic, CEDAR, UTSig, and BHSig260 (Bengali and Hindi)	700 is used for training and 300 is used for testing.

2.1 Problem Statement

Despite numerous advancements in signature verification, existing methods still face several critical challenges that hinder their practical deployment and accuracy. Approaches such as DWSCNN [40] and SVM classifiers [29] offer lightweight solutions but struggle with high variability in signature styles and skilled forgeries. Deep Learning models [43] and Neural Network Model [44] rely on complex robotic arm-based feature extraction and computationally expensive neural network training, increasing system complexity and processing time. Moreover, methods like ProtoSig [45] and Dual-AngleNet [42] performance in writer-independent verification remains unverified and validated on a limited dataset of 600 signature samples. Hybrid models combining global and local features, such as HTCSigNet [31] and hybrid Vision

Transformers [30], have introduced more contextual awareness, but they are computationally intensive and less suited for deployment on low-resource devices. Additionally, many of these methods, including those using depth-wise separable tSSN [14] and TrOCR [41], either rely on handcrafted features or insufficient fusion strategies. A critical gap exists in unifying deep learning with a refined fusion mechanism that can leverage complementary strengths of multiple classifiers. Therefore, the proposed work introduces a novel hybrid framework that integrates ResNet-based deep features, local-global descriptors, and a dual RNN–BiLSTM classifier enhanced with improved score-level fusion. This design aims to address limitations in adaptability, accuracy and computational efficiency across diverse datasets.

3. Summary of Proposed Signature Verification Approach with Deep & Local Features

3.1 Taxonomy of Signature Verification

- Forgery detection can be classified into two types: Active and Passive. Active forgery detection includes watermarking and signature. Watermarking helps to hidden markers to detect forgery. Signatures are validated using biometric or behavioral analysis.
- Passive forgery detection includes Image Statistics and image content. Image statistics help to analyze features like pixel distribution, size and shape to detect anomalies. Image Content is used to focus on geometric properties, strokes and layout of the image to identify inconsistencies.
- Signature is one of the types of active forgery detection. This signature can be classified into two types: Signature Identification and Signature Verification.
- Signature identification approach helps to identify the person who signed by comparing signatures to a database of known signatures. Signature verification technique is used to

verify the authenticity of the signature (genuine or fake), focusing on its validity rather than identification.

- The signature verification includes two types: online verification and offline verification. Online signature verification uses dynamic features such as speed, pressure and stroke dynamics during the signing process. Offline signature verification focuses on local and global features of the signature image that are used to determine the authenticity of the signature.
- In our proposed work, offline signature verification is employed, where the focus is on verifying the authenticity of a signature using scanned or photographed images. Figure.1 represents the taxonomy of signature verification.

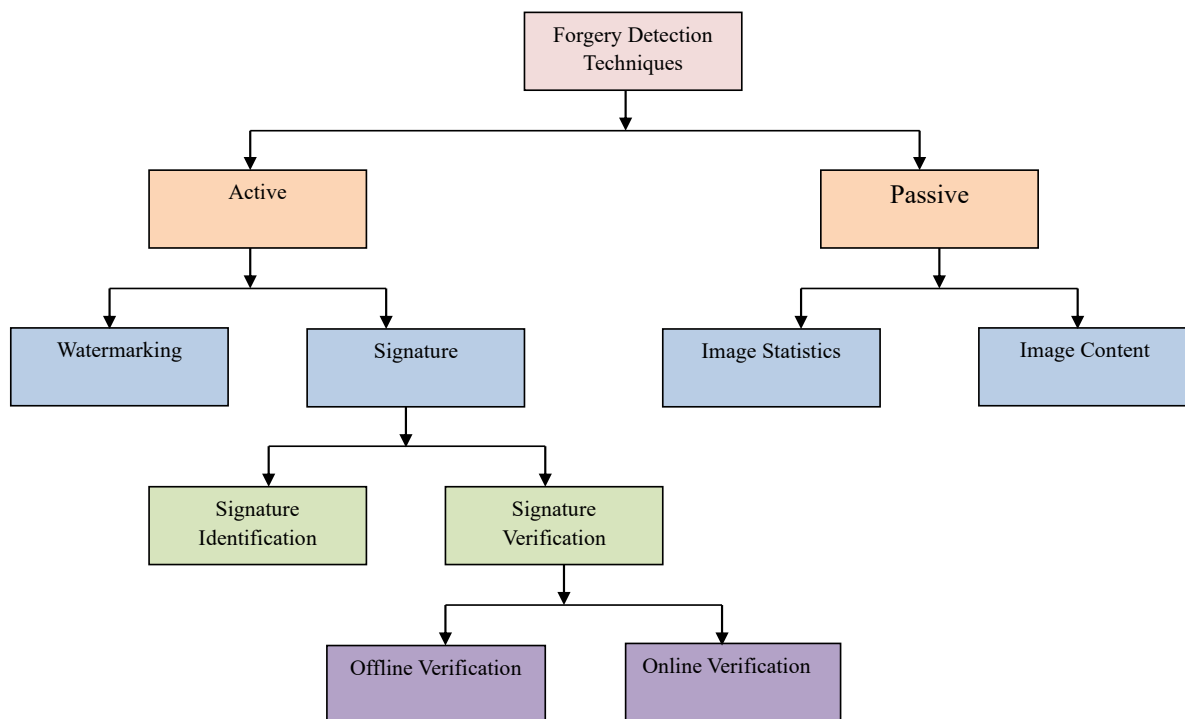


Figure 1. Taxonomy of Signature Verification

A lot of financial transactions need to be verified for authenticity because of advancements in technology. Many systems based on the signing process were therefore offered in order to increase the accuracy of signature verification [32]. These efforts can be divided into two groups based on the kinds of signature data: offline signature verification and online signature verification [33]. Online signature verification relies on the time series dynamic features of the authentication phase, whereas offline signature verification is primarily dependent on image data of the signature that is obtained just after the signing procedure has been completed [34]. It's possible for two persons to sign identically or for signatures to be similar across classes. Signatures exhibit intra-class variation, where one instance from one class differs from another sample within the same category [35]. By considering these issues, a novel deep hybrid signature verification framework is proposed by employing fundamental steps like preprocessing, feature extraction, feature selection and verification and its architecture is illustrated in **Figure.2**.

This work presents a novel deep hybrid signature verification framework that processes multiple input signature images through a preprocessing phase, which includes an improved median filtering method for noise reduction and image resizing to maintain consistency and quality. Essential features, such as ResNet features, global features (e.g., active pixel count) and local features (e.g., signature dimensions and separation between the center and center of mass), are extracted from the preprocessed images. A hybrid feature selection approach integrating the Walrus Optimization Algorithm and Greylag Goose Optimization (WOA-GGO) is employed to identify and select the features for improving the signature verification performance.

The framework then employs a hybrid verification model combining deep learning classifiers like Recurrent Neural Networks (RNN) and Bidirectional Long Short-Term Memory (Bi-LSTM) networks. An enhanced score-level fusion technique is used to merge the outputs of these

classifiers, utilizing a weighted sum in the fusion process and tanh normalization in the score normalization step. This hybrid approach leverages the strengths of sequential learning (RNN) and bidirectional context (Bi-LSTM), yielding a final binary identification result (genuine or forgery). A corresponding detailed explanation of this proposed deep hybrid signature verification model is discussed in the sections below.

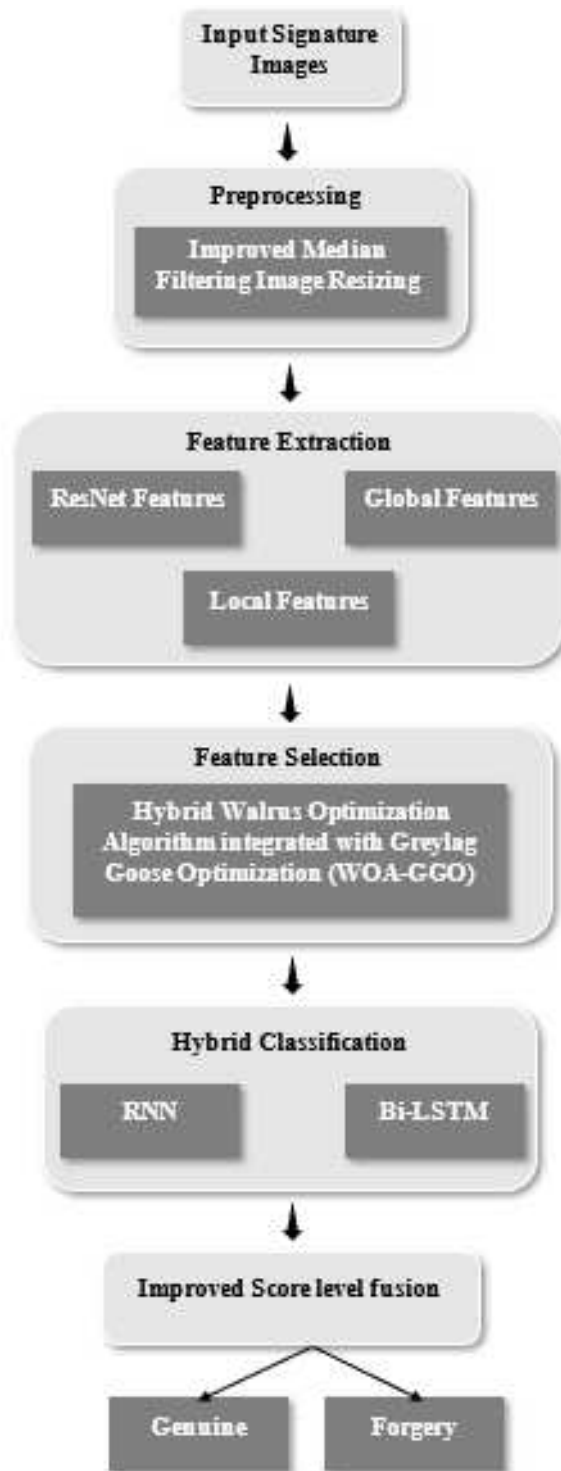


Figure.2. Architectural framework of the proposed deep hybrid verification model

3.2. Improved Median Filtering-based Preprocessing

In this section, preprocessing on the signature image takes place, in which the input signature is denoted as $s_i = \{s_1, s_2, s_3, \dots, s_n\}$. Generally, preprocessing raw data involves processes such as cleaning, converting and sorting to obtain data in a suitable format for further analysis. By enhancing raw data in a suitable format, the efficiency and reliability of the system are also improved. By the way, in this research work, the process of preprocessing data, s_i not only performs median filtering [17] but also does image resizing. Originally, the size of the input signature image was 640x480; through image resizing, its size is reduced to 256x256. Such processes are introduced for noise reduction, data consistency and computational efficiency. Instead of using the conventional median filtering process, an improved process is proposed because the conventional median filter exhibits some limitations, is ineffective in the removal of several noise types, has increased computational cost, is inefficient in the selection of kernel size, etc. However, the improved median filtering process proposed in this preprocessing stage has the ability to handle these issues.

Further, this input signature image is preprocessed by an improved median filtering approach to attain the resultant outcome in an enhanced manner. Typically, a median filter is a filtering technique which filters noise in a non-linear image and results in a noise-free image. This process is done by masking over each pixel and replacing the value of the pixel at each portion with the neighbor pixel's median value, which is defined by a kernel. This proposed framework employs an improved median filtering process. The rationale for choosing Improved Median Filtering (IMF) in the proposed research lies in its superior ability to preserve essential edge details while effectively removing impulse noise, which is common in scanned or photographed signature images. Traditional median filtering, though effective at denoising, can lead to the loss of fine structural information critical for accurate feature extraction. IMF enhances this by adaptively

modifying the filtering window based on local pixel variation, allowing better retention of signature strokes and curves. This ensures that key signature characteristics are maintained, which is vital for downstream processing. Its robustness against both salt-and-pepper noise and background artifacts makes it ideal for preprocessing diverse datasets. By using IMF, the proposed framework ensures high-quality, noise-free inputs that lead to more accurate and reliable verification outcomes. The steps for this improved MF approach given as follows:

- i. The given input signature image's center pixel $c(i, j)$ is determined.
- ii. The values of the respective pixels of the mask are determined.
- iii. These pixel values are sorted and then the median value is identified. In addition to this, the masked pixel's Weighted Power Mean was also calculated (Eq. (1)).

$$WPM_p = \left[\frac{\sum_{i=1}^m w_i p_i^p}{\sum_{i=1}^m w_i} \right]^{1/p} \quad (1)$$

Where, the number of pixel values and the i th value in the set are denoted as m and p_i^p , respectively. The term w_i represents the weight assigned to the i th pixel value. An exponent p is used which determines the type of mean, for instance, if $p=1$ then it is the weighted arithmetic mean and if $p=2$, then it is the weighted root mean square. The weighted becomes maximum if p approaches infinity.

The integration of the weighted power mean into the median filtering process enhances the filter's adaptability by allowing different emphasis on pixel values based on their weights, which improves noise suppression while preserving important signature details. Unlike traditional median filtering that treats all pixels equally, the weighted approach better handles varying noise levels and local variations in the image. This leads to more effective

removal of impulsive noise without blurring critical edges or fine features. Consequently, it results in higher-quality preprocessed images that improve the accuracy of subsequent feature extraction and verification stages.

- iv. Then, the adaptive values of masked pixel 1 mpi_1 and masked pixel 2 mpi_2 are calculated using Eq. (2) and (3). These values are determined by resizing the masked pixels by considering the level of noise that exists in the masked pixel. In which min and max represent the minimum and maximum grey level pixels and md is the median value of the grey level pixels.

$$mpi_1 = md - min \quad (2)$$

$$mpi_2 = md - max \quad (3)$$

- v. Subsequently, each pixel's adaptive masked pixel and weighted power mean are compared by the following conditions.

$$\begin{aligned}
 & \text{if } (mpi_1 > 0 \text{ and } mpi_2 < 0) \text{ then search median value} \\
 & \quad c(i, j) = md \\
 & \text{else if } (c(i, j) > WPM) \text{ then search median value} \\
 & \quad c(i, j) = md \\
 & \text{else} \\
 & \quad \text{return the pixel's original value}
 \end{aligned} \quad (4)$$

- vi. Repeat the above step (v) until $i = j = m$.

Thus, the noise-free signature image is obtained from the preprocessing stage using an improved median filtering process and this resultant preprocessed signature image is symbolized as Pre_{s_i} .

3.3. Extraction of Relevant Features: ResNet features, Global Features and Improved Local Features

The outcome obtained from the preprocessing stage is then input into the feature extraction stage. Here, the resultant preprocessed signature image Pre_{s_i} is applied to retrieve essential features to execute the signature verification stage. Thereby, the essential features to be extracted from Pre_{s_i} are ResNet features, global features and local features. By extracting these features, the signature verification process on the training signature dataset and the testing signature dataset is simple and effective. Incorporating ResNet features leverages deep convolutional layers to capture complex, high-level patterns and subtle variations in signature images, which are essential for distinguishing genuine signatures from forgeries. Global features provide an overall representation of the signature's shape, size and spatial distribution, offering a broad context that aids in recognizing consistent structural traits. Local features focus on fine-grained details such as pen strokes, curves and texture variations, which are critical for identifying unique, individual characteristics within the signature. Combining these three types of features ensures a comprehensive and multi-scale understanding of the signature, enhancing robustness against intra-class variability and forgery attempts. This hybrid approach effectively balances the macro and micro perspectives of the signature, improving verification accuracy and reliability across diverse signature styles and conditions.

3.3.1. Extraction of ResNet Features from Preprocessed Signature Image

The advancement of ResNet [6] makes revolutionary improvements in fields such as computer vision and image classification. It is a DNN design featuring residual blocks, which solve the gradient vanishing issue and enable the network to efficiently train very deep network

architectures. Layer by layer, the network automatically learns visual features with the use of deep learning. SoftMax classification is used in the final layer of deep learning networks, such as ResNet50, to forecast class output. High-level attributes can be obtained prior to FC layers. High-level features are extracted from the fifth residual block (Conv5_x) in the "avg_pool" layer of this investigation using ResNet50. Deep picture features are extracted from this output. 2048-dimensional deep feature vectors are predicted by ResNet50.

With this advantage, the learnable features are extracted from preprocessed signature images Pre_{s_i} at various layers of the network. These extractable features are referred to as ResNet features. ResNet architecture comprises multiple layers in which each layer is built with residual blocks. These blocks have shortcut connections to allow preprocessed signature images, Pre_{s_i} directly input into the network without receiving any heavy alterations. This is the reason why the ResNet has the ability to train very depth network structures. The mathematical expression of the residual block to calculate the output is formulated in Eq. (5).

$$ResNet^O = F\left(Pre_{s_i} w_{res} + Pre_{s_i}\right) \quad (5)$$

Here, $F\left(Pre_{s_i}\right)$ is the transformation function which is performed by each layer in the block. Pre_{s_i} is the input layer. As in Eq. (5), the addition of the input preprocessed signature image Pre_{s_i} to the transformed output is done by a shortcut connection in the residual block, which prevents the network from degradation issues that can happen when networks are too deep.

The activations at several layers of the network represent the features extracted by a ResNet model. The different levels of abstraction are captured from the preprocessed signature image Pre_{s_i} by these activations.

The version of ResNet is the ResNet 50 pretrained model. To extract the features by using 50 layers. The ResNetF feature vector is 1 x 1000 after being extracted from the ResNet model. When you further apply a histogram transformation, the resulting feature vector becomes 1 x 100. As a result, the extracted ResNet features from the hierarchical patterns of the preprocessed signature image are fully identified as *ResNetF*

3.3.2. Extraction of Global Features from Preprocessed Signature Image

The representations which represent the entire input image are known as global features or global representations. In this research work, the input preprocessed signature image Pre_{s_i} is thought to be used to extract the global information, such as the active pixel count and related features. By capturing these features, the input preprocessed signature image's holistic information is revealed frequently by considering the whole context rather than concentrating on local regions. This information gives a clear-cut understanding of the whole structure and content of the preprocessed signature image Pre_{s_i} .

1. Number of Active Pixels

Every person has a unique signing style. Some of them do their sign while some others do it in an elaborate manner. As per this statement, one of the global features known as the number of active pixels is calculated [1]. This feature is computed using Eq. (6) by considering active pixels in each preprocessed signature image Pre_{s_i} .

$$A_{pi}(Pre_{s_i}) = \sum_{i=1}^b \sum_{j=1}^a Pre_{s_i}(i, j) = 0 \quad (6)$$

Where, the preprocessed signature image and its size are represented as $Pre_{s_i}(i, j)$ and (b, a) , respectively. Thus, the resultant number of active pixels featured in global features is indicated as $GF^{A_{pi}}$.

2. Number of Connected Components

It is a graph theory algorithm which defines various connected parts in the image that are characterized by a specific label. The preprocessed signature image Pre_{s_i} is given as an input to extract the number of connected components feature, which this feature will also be considered as the preprocessed signature image's differentiating feature [1]. This feature is considered to be extracted because some people are doing their sign-in pieces. Thereby, the number of connected components featured in global features is denoted as GF^{cc} .

Thus, the resultant global features, including the active pixel count and related characteristics count, are represented as $GF^{A_{pi}}$ and GF^{cc} , respectively. Further, these extracted global features are collectively represented as GF .

3.3.3. Extraction of Local Features from Preprocessed Signature Image

The preprocessed signature image Pre_{s_i} is provided as input to the local feature extraction phase. Generally, local features are extracted from small regions of the image as distinctive representations. From the preprocessed signature image Pre_{s_i} , among the important local features that are recovered are the distance, height and length between the effective center and the center of mass.

The separation between the center of mass and the effective center captures unique spatial distribution patterns of the signature ink, reflecting how the signature's strokes are concentrated or dispersed. This distance varies significantly between individuals because each person's writing style affects the density and layout of their signature differently. Unlike simple geometric features, this measure encodes subtle variations in stroke placement and pressure, which are difficult to

imitate. Therefore, it serves as a robust distinguishing feature by capturing intrinsic personal traits embedded in the signature's structure. Incorporating this separation enhances the model's ability to differentiate genuine signatures from forgeries, improving overall verification accuracy.

1. Improved Distance between the Effective Center and Center of Mass

The factors, such as the effective center and the center of mass, are evaluated from the input preprocessed signature image Pre_{s_i} . The preprocessed signature image's effective center is altered on the basis of the density and content of the signature. The center of mass is considered the selective feature of the preprocessed signature image, as it is based on the user [1]. Mathematically, the center of mass of each block in the preprocessed signature image is computed in Eq. (7) and (8).

$$\forall p C_P^M = \left(\frac{pi \in Pre_{s_i}(i, j) p_{pi}}{2} \right) \quad (7)$$

$$\forall q C_Q^M = \left(\frac{pi \in Pre_{s_i}(i, j) q_{pi}}{2} \right) \quad (8)$$

Here, the preprocessed signature image is Pre_{s_i} and the pixel value is pi . The pixels in the x and y directions are denoted as p_{pi} and q_{pi} , respectively. Then, the effective center is evaluated in Eq. (9) and (10) where, $i = 1, 2, 3, \dots, b$ and $j = 1, 2, 3, \dots, a$.

$$C_P^{EC} = \left(\frac{\max(\sum p_i = 0)}{2} \right) \quad (9)$$

$$C_Q^{EC} = \left(\frac{\max(\sum q_i = 0)}{2} \right) \quad (10)$$

Using Eqs. (7), (8), (9) and (10), the essential distance between the effective center and the center of mass feature is expressed in Eq. (11).

$$D = \sqrt{(C_P^{EC} - C_P^M) + (C_Q^{EC} - C_Q^M)} \quad (11)$$

The above-mentioned 11th equation is proposed using a sparse weight vector and the proposed distance formula is formulated in Eq. (12) where, the sum of weights, w_k is denoted as W in which those weights are calculated using a sparse weight vector. The variable z must be positive i.e. $z > 0$ here, $z = 2$. The applied sparse weight vector in weights is randomly generated by the generation of a function which generates an ensemble of random vectors. Then, some of its elements are randomly set to zero to make it asparse weight vector, i.e. $|C_P - C_Q|$ which is obtained from $D(C_P, C_Q) = \sqrt{(C_P - C_Q)^2}$. Thereby, the resultant distance between the effective center and the center of mass feature in local features is denoted as LF^D .

$$D(C_P, C_Q) = \left[\frac{1}{W} \sum_{k=1}^g w_k \cdot |C_P - C_Q|^z \right]^{1/z} \quad (12)$$

2. Length and Height of the Signature

The length and height of the signature vary from person to person. For instance, some of them use their initials, which makes their signs longer, while others just use their names, which makes their signs a little shorter than those that combine names and initials. Even this small piece of information will be effectively used as a feature to train the datasets effectively [1]. Thereby, the signature's length and height in the preprocessed signature image Pre_{s_i} is evaluated by considering the pixels in x and y directions and it is formulated in Eq. (13) and Eq. (14), which is the sum of

active pixels in the i th row and j th column, respectively. The resultant length and height of the signature feature in local features are denoted as LF^{LH} .

$$H = \max(\sum p_i = 0); \quad i = 1, 2, 3, \dots, b \quad (13)$$

$$L = \max(\sum q_i = 0); \quad j = 1, 2, 3, \dots, a \quad (14)$$

The distance between the center of mass and the effective center captures the unique spatial distribution of ink in a signature, which varies naturally between individuals. Genuine signatures tend to have consistent patterns in stroke density and placement, resulting in a characteristic separation between these two points. Forged signatures, however, often fail to replicate this subtle spatial arrangement accurately, leading to noticeable differences in this distance. Thus, this feature effectively encodes personal writing habits and stroke dynamics that are difficult to imitate. Including this measure enhances the system's ability to distinguish genuine signatures from forgeries, improving overall verification reliability.

Thus, the resulting local characteristics, including the signature's length and height as well as the separation between the effective center and the center of mass, are indicated as LF^D and LF^{LH} , correspondingly. Further, these extracted local features are described as LF .

Finally, the extracted features like ResNet features ($ResNetF$), global features (GF) and local features (LF) are concatenated as a whole into a feature set F_{s_i} .

3.4 Selection of Relevant Features: WOA-GGO Hybrid Optimization Algorithm

Feature Selection is the process of selecting the features from the datasets. In this stage, the optimal features are selected from the extracted features by using the WOA-GGO hybrid optimization algorithm. The workflow of feature selection is shown in the figure 3.

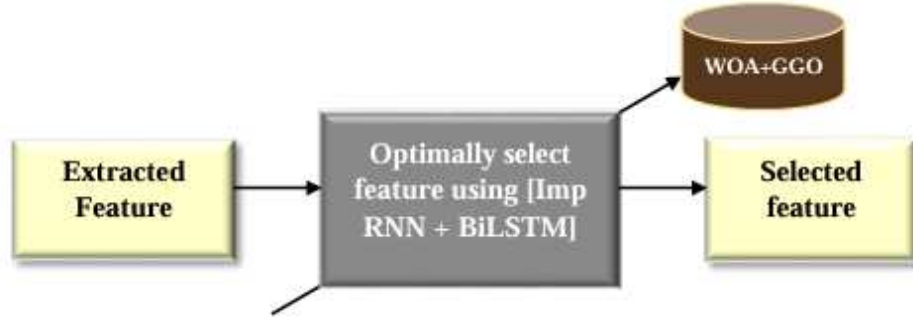


Figure 3: Workflow of Feature Selection

3.4.1 Walrus Optimization Algorithm Integrated Greylag Goose Optimization (WOA-GGO)

The WOA-GGO algorithm combines the Walrus Optimization Algorithm (WOA) and the Greylag Goose Optimization (GGO) Algorithm. GGO Algorithm is a nature inspired optimization algorithm based on the migratory and cooperative behavior of greylag geese to solve complex optimization problems effectively.

Walrus Optimization Algorithm

WOA [39] is an optimization algorithm modeled after the feeding, predator avoidance and migration behaviors of walruses.

$$y_{i,j}^{H2} = \begin{cases} y_{i,j} + rand_{i,j} \cdot (y_{k,j} - J_{i,j} \cdot y_{i,j}), & Fn_k < Fn_i \\ y_{i,j} + rand_{i,j} \cdot (y_{i,j} - y_{k,j}), & else \end{cases} \quad (15)$$

Where $y_{i,j}^{H2}$ is denoted as the j th dimension based on the 2nd phase, Fn_k and Fn_i refers to the objective function values. Converting the Eq. (15) as one and $rand_{i,j}$ is the random selection as shown in the Eq. (16).

$$= 2y_{i,j} + rand_{i,j} \cdot y_{i,j} - rand_{i,j} \cdot J_{i,j} \cdot y_{i,j} \quad (16)$$

$$= y_{i,j} \left(2 + rand_{i,j} \cdot y_{i,j} (1 - J_{i,j}) \right) \quad (17)$$

$$y_{i,j} = \frac{y_{i,j}^{H2}}{\left[2 + rand_{i,j} (1 - J_{i,j}) \right]} \quad (18)$$

Here the values got cancelled and it got the $y_{i,j}$ value which will be substituted in the greylag goose optimization.

Greylag Goose Optimization

GGO [38] algorithm belongs to the class of swarm-based algorithms and is inspired by the Greylag Goose. The optimization process of the GGO algorithm is explained using the following equations.

$$Y(a+1) = Y^*(a) - B \cdot |D \cdot Y^*(a) - Y(a)| \quad (19)$$

Where $Y(a)$ is an agent at an iteration a . The $Y^*(a)$ represents the best solution position and $Y(a+1)$ is the updated position of the agent as shown in the Eq. (19). Hybrid these equations (18) and (19). The first term is derived from the social migration learning of WOA, whereas the second term is derived from the exploitation of GGO.

$$Y_{new} = \left[\frac{y_{i,j}^{H2}}{\left[2 + rand_{i,j} (1 - J_{i,j}) \right]} \right] - \left[B \cdot |D \cdot Y^*(a) - Y(a)| \right] \quad (20)$$

where $y_{i,j}^{H2}$ sets to the $Y^*(a)$ in the below Eq. (20)

$$Y_{new} = \left[\frac{y^*(a)}{\left[2 + rand_{i,j} (1 - J_{i,j}) \right]} \right] - \left[B \cdot |D \cdot Y^*(a) - Y(a)| \right] \quad (21)$$

In Greylag Goose Eq. (19) is replaced by the Eq. (21), where $rand_{i,j}$ is calculated using

$$rand_{i,j} = 2 \cos\left(\frac{\pi}{2} * \frac{a}{A_{\max}}\right) - 1 \quad (22)$$

$$Y(a+1) = Y(a) + E(1+m) * r * (Y - Y_{Flock1}) \quad (23)$$

Where individual search near the best solution to identify the improved solutions denoted as Y_{Flock1} as shown in the Eq. (23). r is calculated using

$$r = r_{\max} \cdot \left[1 - \ln\left(1 + \frac{a}{a_{\max}}\right) \right] + \left[\frac{r_3}{r_{\max}} \cdot \ln\left(1 + \frac{t}{t_{\max}}\right) \right] \quad (24)$$

Where $r_3 = 1 - r_1 - r_2$, $r_1 = \cos \theta$, $r_2 = \frac{1}{2} \tan \theta \cdot \cot \phi$, $\theta = 2 \arccos \frac{1}{3} \cdot \arcsin(ia)$, $\phi = \frac{1}{2} \arccos(ia)$,

ia refers to the ∞ , $r_{\max}$ is equal to 0.9, $r_{\min}$ is equal to 0.4, a refers to the current iteration and

$a_{\max}$ refers to the maximum number of iteration as shown in the Eq. (24).

Traditional GGO suffers from several limitations. One major disadvantage is premature convergence, where the algorithm easily gets trapped in local optima, particularly in complex or high dimensional optimization problems. Additionally, the fixed division of the population into exploration and exploitation groups often results in an imbalanced search process, reducing the algorithms search efficiency and overall optimization performance. Modified GGO effectively overcomes the limitations of the traditional GGO by introducing the adaptive population partitioning, which dynamically balances the exploration and exploitation based on the fitness distribution, thereby improving the search robustness. The integration of random calculations enhances the population diversity and prevents stagnation during the optimization process. Furthermore, the updated position updates the equations, normalize the search process and include the oscillatory terms, resulting in the faster convergence, higher accuracy and improved performance for solving the complex real world optimization problems.

3.4.2 Objective Function

The objective function is used to minimize the error in the signature verification as shown in the following Eq. (25).

$$fit = \frac{1}{acc} \quad (25)$$

Where *fit* refers to the fitness, *acc* refers to the accuracy, lower bound is set to 0, the upper bound is set to 1, population size is fixed at 10 and the problem size is based on the number of attributes in the extracted feature set as shown in the above equation.

3.5. Signature Verification using Improved Score Level Fusion

For many years, security software has been used by numerous financial and educational institutions. These days, a lot of financial transactions need to be verified for authenticity because of advancements in technology. Therefore, a number of signing-process-based solutions were made available to improve the precision of signature verification. Many studies have already been conducted on handwritten signature authenticity. However, they don't provide efficient results in signature verification processes. Thereby, this research work introduces a novel deep hybrid signature verification framework in which the verification process on signature images is effectively performed by the hybrid verification model.

Figure.4 represents the structure of the proposed hybrid model for signature verification, which is formed by the hybridization of two DL classifiers: RNN and Bi-LSTM. This hybridization is involved to train and test the signature images in a time-efficient and accurate manner. Also, both the RNN and Bi-LSTM classifiers exhibit a few limitations which will have a negative impact on the final verification results by providing inaccurate results. Such limitations exhibited by RNN

classifiers are gradient vanishing, training complexity, contextual understanding difficulty, etc., while these RNN limitations are viable advantages of Bi-LSTM. As well as limitations of Bi-LSTM, including overfitting, long-term independence, losing causality, etc. are the manageable benefits of RNN. By considering these facts, this proposed deep hybrid signature verification framework becomes more effective by hybridizing these classifiers. These classifiers receive the feature set that is produced as an output from the feature extraction stage independently. The intermediate scores generated by RNN and Bi-LSTM classifiers are then fused using the enhanced score-level fusion.

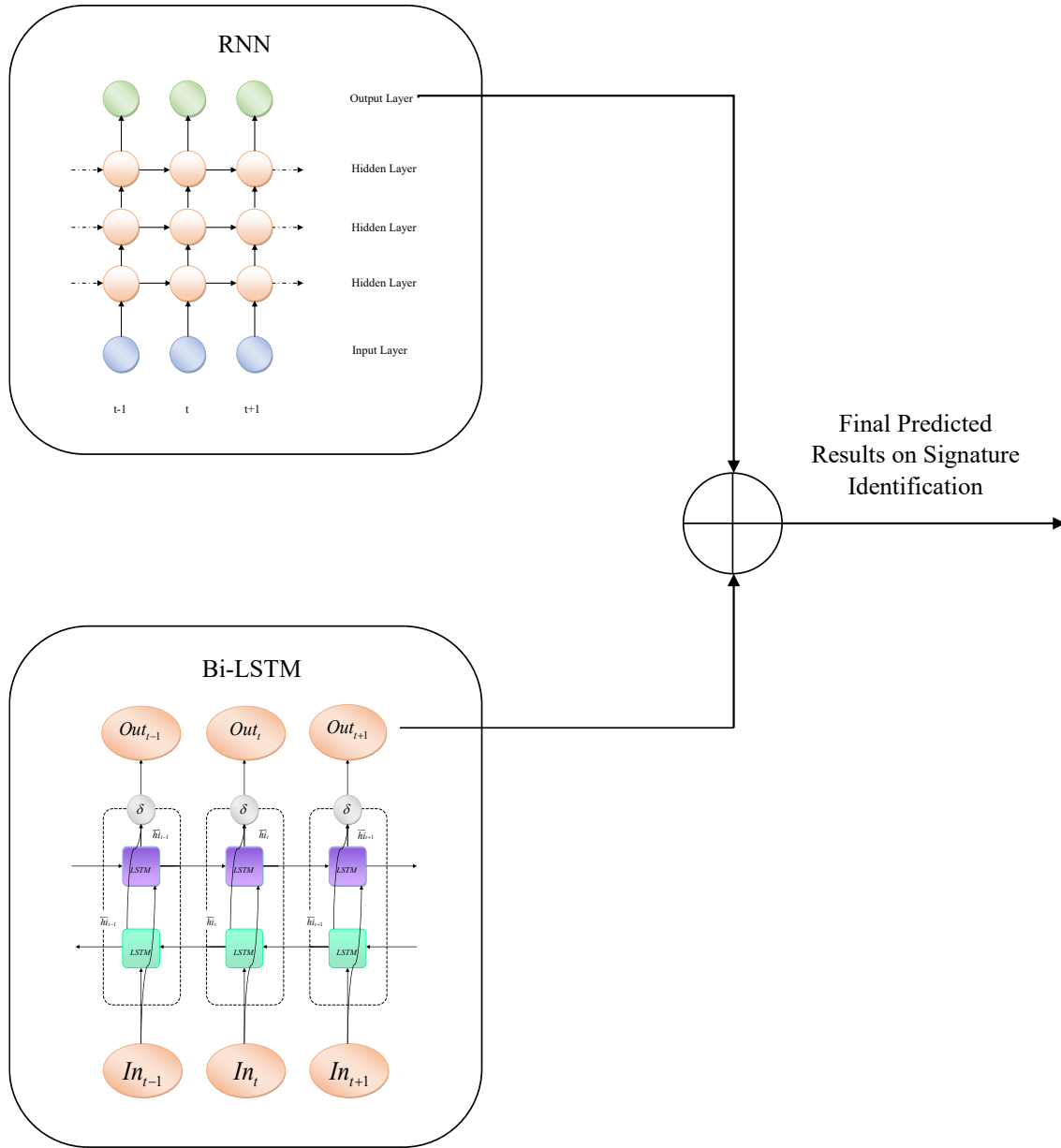


Figure.4. Structural diagram of the proposed hybrid Verification model

3.5.1. RNN Classifier in Signature Verification

RNN classifier [13] is one of the effective DL architectures because it can even recall past information. In RNN, the current step is processed by an input, which is nothing but the previous step's output. It even ensures that each and every output generated from the RNN is processed on

the basis of the previous step and stores calculated results in the memory of the previous steps. The network structure of RNN is replicated in **Figure.5**, in which each layer has its specific determined threshold values and weights to result in more accurate outcomes.

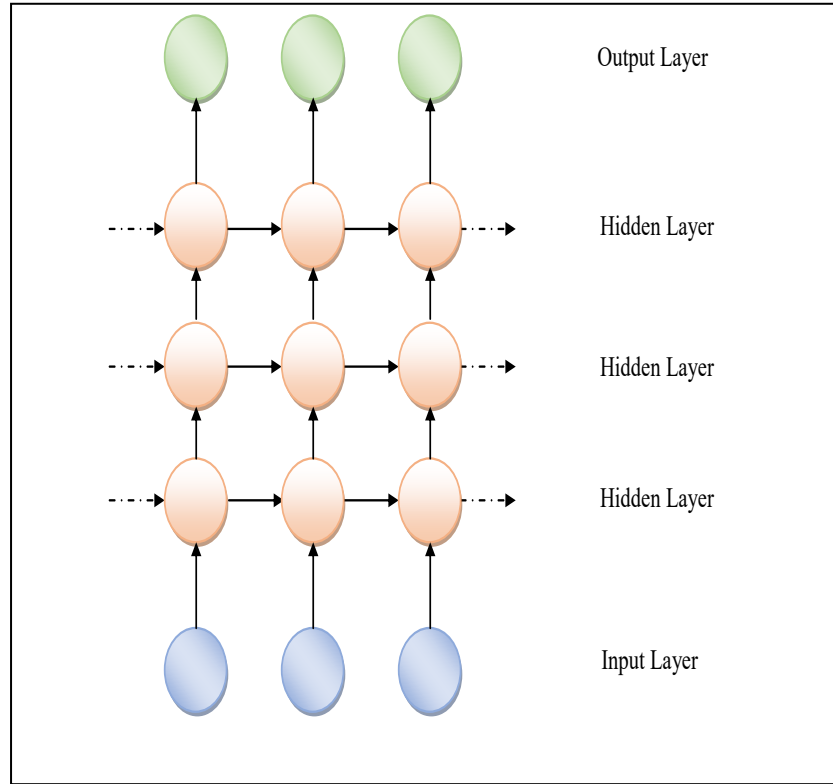


Figure.5. Architecture of RNN

The newly attained input value in a step is combined with the already stored previous input state, where this process is associated with the newly attained input with the previous input state. The general Equations which mathematically express the RNN architecture are as follows.

$$hi = \sigma(ZF_{s_i} + Rhi_{t-1} + B) \quad (26)$$

$$Out = O(Rhi + Y) \quad (27)$$

$$Out = f(In, hi, w_{RNN}, Z, R, B, Y) \quad (28)$$

Here, the parameters w_{RNN}, Z, R, B and Y denoted in the above equations are shared across time steps in the network. The mathematical formulation that describes the current state is formulated in Eq. (29). Then, the activation function applied current state is expressed in Eq. (30), where the weights of recurrent and input neurons are indicated as $w_{r_{hihi}}$ and $w_{r_{Inhi}}$, respectively.

$$hi_t = f(hi_{t-1}, In_t) \quad (29)$$

$$hi_t = \tanh(w_{r_{hihi}} hi_{t-1} + w_{r_{Inhi}} In_t) \quad (30)$$

Then, the output layer in RNN is represented as Eq. (20).

$$Out_t = w_{r_{Outhi}} hi_t \quad (31)$$

In this classifier, the resultant feature set, F_{s_i} obtained by concatenating the extracted essential features from the preprocessed signature image, is provided as input. This feature set is processed by this RNN classifier as per its functioning, and the resultant intermediate score attained is symbolized as Sc_{RNN} .

3.5.2. Bi-LSTM Classifier in Signature Verification

An improvement over the LSTM classifier is the Bi-LSTM classifier [15] because it addresses some of the limitations of conventional LSTM by processing input in the forward direction as well as in the backwards direction. The Bi-LSTM classifier involved in the hybrid verification model is modeled in **Figure.6**. This self-regulated classifier is formed by integrating two RNNs, and it comprises a cell state and three gates: input gate (In^t), output gate (Out^t) and forget gate (for^t). It is advantageous because it is free from vanishing gradient issues while training on long-term dependencies. This is the reason why this classifier is employed in the deep hybrid signature verification framework. In which data transmission is done by two parallel layers to transmit data

in a fourth direction. In the Bi-LSTM architecture, the memory cells are self-connected, and it is used to store the processed data in the current state at each time step. The resulting feature set F_{s_i} is sent to the output gate after being supplied as an input to this Bi-LSTM classifier. The forget gate simultaneously identifies the memory cell's internal state and combines it with the input feature set F_{s_i} . Then, according to the self-recurrence process, this internal state combined feature set is sent to the following cell, and it forgets the information stored in the memory cell. The feature set's input sequence and output sequence are $F_{s_i} = \{F_{s_{i_1}}, F_{s_{i_2}}, \dots, F_{s_{i_T}}\}$ and $oF_{s_i} = \{oF_{s_{i_1}}, oF_{s_{i_2}}, \dots, oF_{s_{i_T}}\}$, respectively. The output timestep of the Bi-LSTM model is represented as $t = 1 \text{ to } T$. The mathematical expressions involved in the Bi-LSTM model are formulated from Eq. (32)- Eq. (37).

$$In_t = \delta \left(Z_{In} F_{s_{i_t}} + R_{In} hi_{t-1} + G_{In} cu_{t-1} + B_{In} \right) \quad (32)$$

$$Out_t = \delta \left(Z_{Out} F_{s_{i_t}} + R_{Out} hi_{t-1} + G_{Out} cu_{t-1} + B_{Out} \right) \quad (33)$$

$$for_t = \delta \left(Z_{for} F_{s_{i_t}} + R_{for} hi_{t-1} + G_{for} cu_{t-1} + B_{for} \right) \quad (34)$$

$$u_t = \delta \left(Z_u F_{s_{i_t}} + R_u hi_{t-1} + G_u cu_{t-1} + B_u \right) \quad (35)$$

$$cu_t = for_t * cu_{t-1} + In_t * u_t \quad (36)$$

$$hi_t = Out_t * \tanh(cu_t) \quad (37)$$

At each time step t , the current input, previous hidden state and previous cell output state are represented as $F_{s_i}^t, hi_{t-1}$ and cu_{t-1} , respectively. The terms Z^*, R^* and G^* are the weights and B^* denotes biases. The overall intermediate score of Bi-LSTM is calculated as Eq. (38) and it is denoted as $Sc_{Bi-LSTM}$.

$$hi_t = \delta \left(Z_{hi} \begin{bmatrix} \overrightarrow{hi^t} \\ \overleftarrow{hi^t} \end{bmatrix} + B_{hi} \right) \quad (38)$$

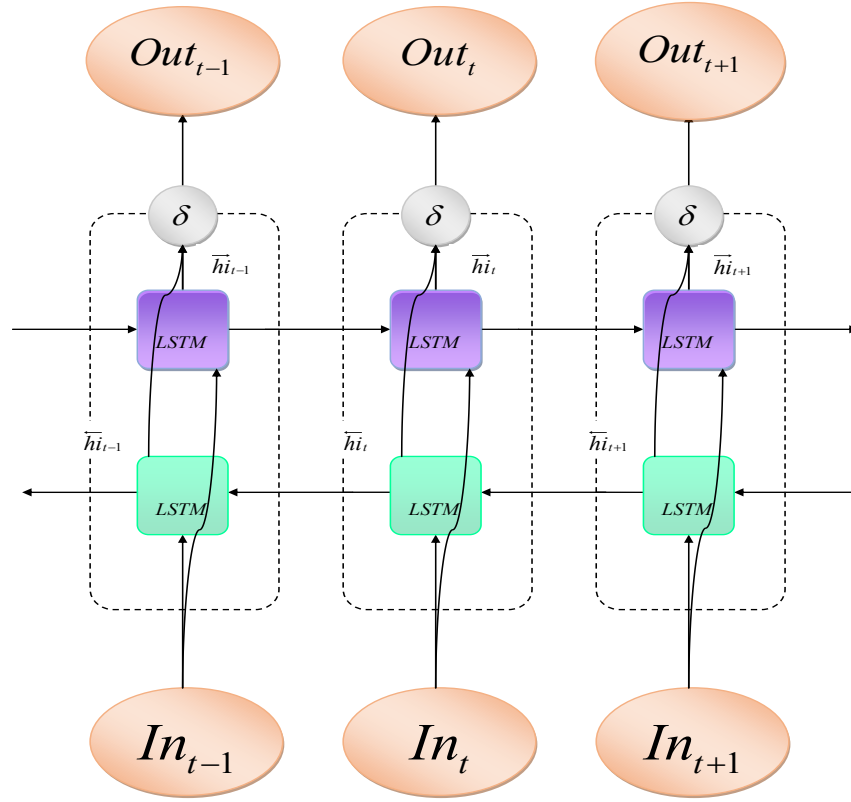


Figure.6. Architecture of Bi-LSTM

Table 2 describes the parameters of the classifiers, like Bi-LSTM and Improved RNN. The parameters of the Improved RNN and Bi-LSTM models significantly impact the proposed signature verification framework. The Improved RNN with three hidden layers (40, 20 and 10 neurons) using sigmoid activation functions captures hierarchical features from the input signature data, with context layers allowing dynamic learning of inter-feature dependencies. The use of ReLU in the output layer enhances convergence, while the softmax function ensures proper classification. The Adam optimizer and binary cross-entropy loss facilitate efficient learning. The Bi-LSTM model, with its bidirectional architecture and 64 LSTM units, captures both forward and backwards dependencies, essential for recognizing subtle variations in signature patterns. The inclusion of dropout (0.5 and 0.2) in the Bi-LSTM layers prevents overfitting and improves

generalization. The SGD optimizer and categorical cross-entropy loss are used for classification tasks, ensuring the model effectively distinguishes between genuine and forged signatures. The combination of these parameters ensures a balance between accuracy and computational efficiency, making the model robust for signature verification applications.

Table 2: Parameters of the Classifiers

Models	Layers and Parameter Values
Improved RNN	Number of input layers: 1 Number of RNN hidden layers: 3, i)First hidden layer neuron=40, activation='sigmoid' context layer1: initializer='random normal', trainable=True, name='context weights' ii)second hidden layer neuron=20, activation='sigmoid' context layer2: initializer='random normal', trainable=True, name='context weights' iii)third hidden layer neuron=10, activation='sigmoid' context layer3: initializer='random normal', trainable=True, name='context weights' activation: 'relu' Number of output layer: 1 activation function: softmax batch size:32 optimizer='adam', loss='binary cross entropy, epoch=50

	validation split=0.2
Bi-LSTM	input layer:1 Bidirectional LSTM layer1: lstm unit:64 activation='relu', return sequences=True Dropout=0.5 Bidirectional LSTM layer2: lstm unit:64 Dropout=0.2 output layer ->1 activation=sigmoid loss='categorical cross entropy, optimizer='sgd', metrics=['accuracy']

Finally, the outputs attained from RNN and Bi-LSTM are denoted as Sc_{RNN} and $Sc_{Bi-LSTM}$, respectively. An enhanced score-level fusion method is employed to combine these scores.

3.5.3. Improved Score Level Fusion Technique

In this proposed deep hybrid signature verification framework, an improved score-level fusion technique is employed to fuse the resultant intermediate scores provided by RNN and Bi-LSTM classifiers. The score-level fusion in this work combines the intermediate verification scores generated independently by the RNN and Bi-LSTM classifiers. Rather than relying on a single classifier, this fusion method leverages the complementary strengths of both models to improve decision reliability. The fusion is performed through a weighted averaging scheme, where the contribution of each classifier's score is adjusted based on validation performance. This approach reduces the impact of individual classifier errors and balances their weaknesses, such as RNN's

difficulty with long-term dependencies and Bi-LSTM's potential overfitting. By integrating the scores at this intermediate level, the system achieves more robust and accurate verification outcomes. Typically, score level fusion [21] is a technique which is used to integrate multiple individual scores from various models to make a final result. This technique is broadly utilized in several fields, including ML, information retrieval, pattern recognition, etc. This primarily involved enhancing the entire robustness, performance and reliability of a model. Most commonly, the score level fusion technique is processed by techniques such as Min-Max normalization, weighted averaging, majority voting, etc. In this research work, the Min-Max normalization-based score level fusion is considered a conventional technique, and it is replaced by the hyperbolic tangent normalization technique [2]. The improved score-level fusion method is chosen in the proposed work to enhance the reliability and accuracy of signature verification by effectively combining the strengths of both RNN and Bi-LSTM classifiers. Unlike traditional fusion methods such as simple averaging, this approach uses adaptive weighting based on each classifier's performance, allowing more accurate models to have greater influence on the final decision. This helps reduce false acceptance and rejection rates. Additionally, by integrating hyperbolic tangent normalization, it handles outliers more robustly and ensures normalized outputs in the $[0,1]$ range. This results in a more stable and discriminative verification outcome. Overall, the method increases robustness without adding significant computational complexity.

As mentioned above, the RNN classifier is separately processed by providing a feature set F_{s_i} as input, and it results in an intermediate score Sc_{RNN} . Likewise, the Bi-LSTM classifier is separately processed by F_{s_i} and generated $Sc_{Bi-LSTM}$ as an intermediate score. These intermediate scores are integrated by employing an improved score level fusion technique to generate efficient

results on signature verification as genuine or forged. This technique follows two key steps: score normalization and score fusion.

1. Score Normalization

Conventionally, the normalization process on scores is done by Min-Max normalization. However, this improved score-level fusion technique utilizes the hyperbolic tangent normalization technique. Because it can provide sufficient performance and high robustness while handling outliers. And the saw score is always mapped in the interval (0, 1). The mathematical expression of intermediate scores by applying hyperbolic tangent normalization is formulated in Eq. (39) and (40).

$$Sc'_{RCNN} = 0.5 \times \tanh \left[\frac{0.01 (Sc_{RCNN} - Sc_{RCNN}^{\mu})}{Sc_{RCNN}^{\sigma}} \right] + 1 \quad (39)$$

$$Sc'_{Bi-LSTM} = 0.5 \times \tanh \left[\frac{0.01 (Sc_{Bi-LSTM} - Sc_{Bi-LSTM}^{\mu})}{Sc_{Bi-LSTM}^{\sigma}} \right] + 1 \quad (40)$$

Where, the intermediate scores of RNN and Bi-LSTM classifiers are represented as Sc_{RCNN} and $Sc_{Bi-LSTM}$, respectively. The mean estimation of RNN and Bi-LSTM intermediate score distribution is denoted as Sc_{RCNN}^{μ} and $Sc_{Bi-LSTM}^{\mu}$, respectively. In addition, the standard deviation estimation of RNN and Bi-LSTM intermediate score distribution are indicated as Sc_{RCNN}^{σ} and $Sc_{Bi-LSTM}^{\sigma}$, accordingly.

2. Score Fusion

After score normalization, a score fusion process takes place, which combines the intermediate scores exhibited by RNN and Bi-LSTM classifiers and generates a new predicted fusion score.

Conventionally, this process is carried out as Eq. (41). However, the improved score level fusion takes place by using weighted sum methodology and its formulated expression is shown in Eq. (42).

$$SLF = \sum_{i=1}^x (Sc_{RNN} + Sc_{Bi-LSTM}) \quad (41)$$

$$ISLF_{w_{sum}} = \sum_{(i=1 \text{ to } N)} W_i * Sc_i \quad (42)$$

Where the term W_i are user-specific weights associated with each classifier. These weights are set according to the accuracy of each classifier. It is set in accordance with the accuracy of the RNN and Bi-LSTM classifiers. The probability of false rejection is reduced by the lower accuracy (weight) of the classifier and the higher accuracy (weight) of the classifier. In conclusion, the fused score achieved from enhanced score level fusion is the final verification result. The fused score will also fall within this range is [0,1].

4 Results and Discussion

4.1 Experimental Setup

Python was used to simulate the suggested signature verification framework. Further, the processor used was an AMD Ryzen 5 3450U with Radeon Vega Mobile Gfx 2.10 GHz, and the installed RAM capacity was "16.0 GB (13.9 GB usable)". Additionally, the Python version was "PYTHON 3.7." Here, we utilized two different datasets for signature verification analysis: a) GPDS 1-150 dataset 1 [36] and Cedar dataset 2 [37].

4.2 Dataset 1 Description

The GPDS 1-150 dataset consists of 8,100 signature samples, with 3,600 genuine signatures and 4,500 forged signatures, collected from 150 distinct writers. Each writer contributed 24 genuine and 30 forged signatures, with both online and offline signatures. This dataset is widely used for signature verification tasks, offering a challenging scenario due to the variability in signature styles and the inclusion of different types of forgeries. It provides a rich set of dynamic and static features, making it suitable for developing and testing robust signature verification models using machine learning and deep learning techniques.

4.3 Dataset 2 Description

The CEDAR Signature Dataset contains 2,640 signature samples, with 1,320 genuine signatures and 1,320 forged signatures, collected from 55 different writers. Each writer contributed 24 genuine and 24 forged signatures, with the dataset providing both online (capturing dynamic writing features such as pressure and velocity) and offline (scanned) signature data. This dataset is commonly used for signature verification research, offering a diverse set of samples with different writing styles, dynamic characteristics and forgery techniques. It is particularly useful for testing models focused on distinguishing between genuine and forged signatures using both static and dynamic features. The training and testing data samples for datasets 1 and 2 are described in Table 3.

Table 3: Testing and training data of both Datasets

Dataset 1				
Percentage of Data	60	70	80	90

Data Used	Training Data (4860, 105); Testing Data (3240, 105)	Training Data (5670, 105); Testing Data (2430, 105)	Training Data (6480, 105); Testing Data (1620, 105)	Training Data (7290, 105); Testing Data (810, 105)
Dataset 2				
Percentage of Data	60	70	80	90
Data Used	Training Data (1584, 105); Testing Data (1056, 105)	Training Data (1847, 105); Testing Data (793, 105)	Training Data (2112, 105); Testing Data (528, 105)	Training Data (2376, 105); Testing Data (264, 105)

4.4 Performance Analysis

4.4.1. Evaluation Metrics

The estimation on RNN+Bi-LSTM and traditional methods was done for metrics such as Precision, FNR, EER F-measure, Sensitivity, NPV, FPR, Specificity, MCC and Accuracy.

Accuracy:

Accuracy measures the overall correctness of the signature verification system. It is the ratio of correctly classified instances (both genuine and forgery signatures) to the total number of instances. Eq (32) represents the accuracy, in which TP represents true positive, TN signifies true negative, FP implies false positive and FN signifies false negative.

$$Accuracy = \frac{TP + TN}{TotalSamples} \quad (32)$$

Precision

A precision metric is the proportion of correctly identified cases. Precision is represented by equation (33).

$$\text{Precision} = \frac{TP}{FP + TP} \quad (33)$$

Sensitivity

The probability of a positive test result is known as sensitivity, which solely considers positive true values. Sensitivity can be expressed using equation (34).

$$\text{Sensitivity} = \frac{TP}{FN + TP} \times 100 \quad (34)$$

Specificity

By examining the probability of only having actual negative values, specificity can be quantified.

The specificity is provided by equation (35).

$$\text{Specificity} = \frac{TN}{FP + TN} \times 100 \quad (35)$$

FNR

The false negative rate is the likelihood that the test may miss a true positive. The FNR formulas are defined by equation (36).

$$\text{FNR} = \frac{FN}{FN + TP} \quad (36)$$

FPR

The percentage of negative events that are mistakenly classified as positive (false positives) relative to the total number of negative events that actually occur is known as the false positive rate. In Eq. (37), the FPR is expressed mathematically.

$$\text{FPR} = \frac{FP}{FP + TN} \quad (37)$$

F-measure

The correctness of the results is evaluated using the F-measure. Equation (38), which expresses the formula.

$$\text{F-measure} = \frac{TP}{\frac{1}{2}(FP + FN) + TP} \quad (38)$$

Equal Error Rate (EER)

The Equal Error Rate (EER) is the point at which the False Acceptance Rate (FAR) and False Rejection Rate (FRR) are equal. It is a common metric for evaluating biometric verification systems, including signature verification. The Eq (39) refers the formula of EER.

$$EER = FAR = FRR \quad (39)$$

Where FAR denotes the False Acceptance Rate; FRR indicates the False Rejection Rate.

Matthews Correlation Coefficient (MCC):

MCC is a metric that takes into account all four confusion matrix values (TP, TN, FP, FN). It provides a balanced measure even if the data is imbalanced. MCC ranges from -1 (perfectly incorrect prediction) to +1 (perfectly correct prediction), with 0 indicating random prediction. The formula of MCC is represented in Eq. (40).

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (40)$$

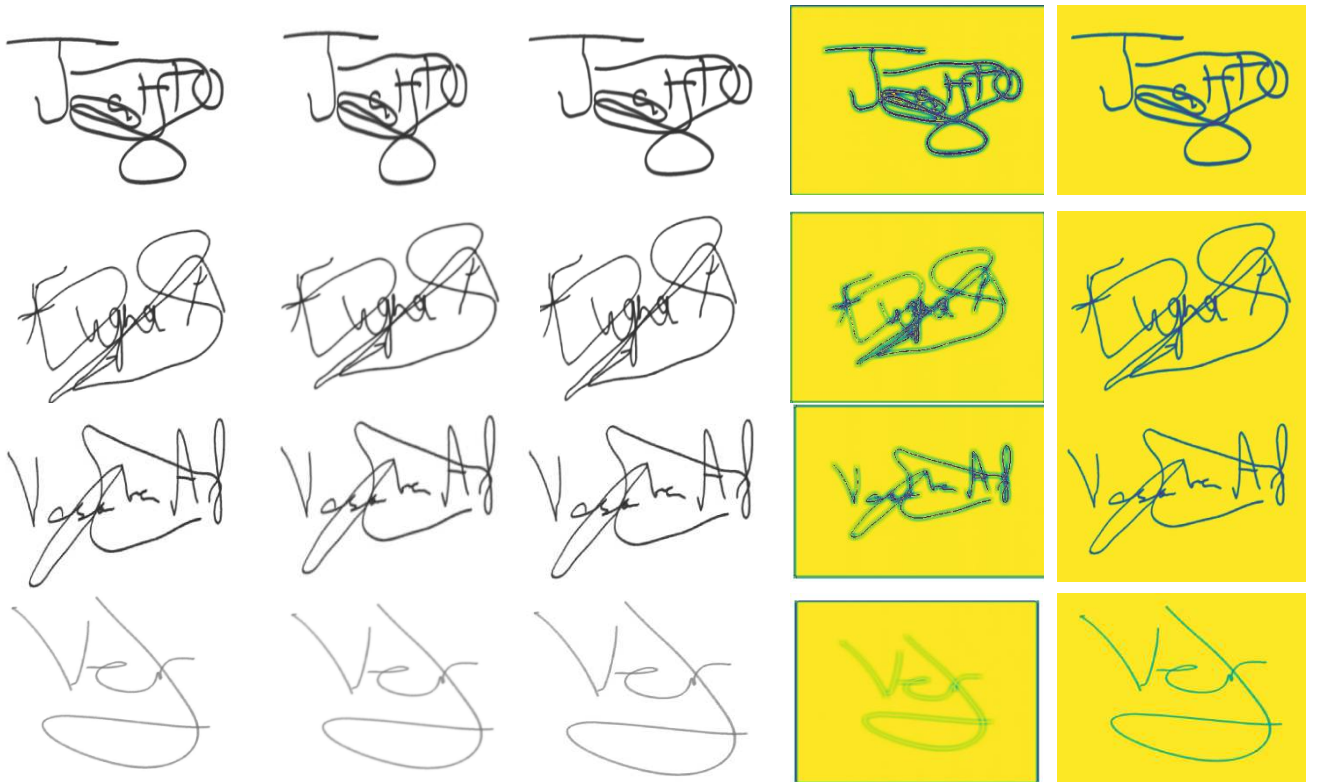
Where, TP indicates the True positive and TN denotes the True negative; FP represents false positive; FN denotes False Negative.

NPV

NPV measures the proportion of negative predictions (forgeries) that are actually correct. It is particularly useful for evaluating the system's ability to correctly identify forgeries. Eq. (41) represents the formulation of NPV.

$$NPV = \frac{TN}{TN + FN} \quad (41)$$

The metrics, especially accuracy, EER, MCC and NPV, depend heavily on the distribution of genuine signatures and forgeries in the test set. Furthermore, the RNN+Bi-LSTM was contrasted with the state-of-the-art methods, like, hybrid ViT [29], HTCSigNet [30], hybrid deeplearning [7] and tSSN [14]. In addition, the RNN+Bi-LSTM was evaluated over the prior classifiers like Linknet, LSTM, EfficientNet, Bi-GRU, LeNet and Bi-LSTM. Similarly, images for the signature verification framework for Dataset1 and Dataset2 were shown in Figure. 7 and Figure. 8.



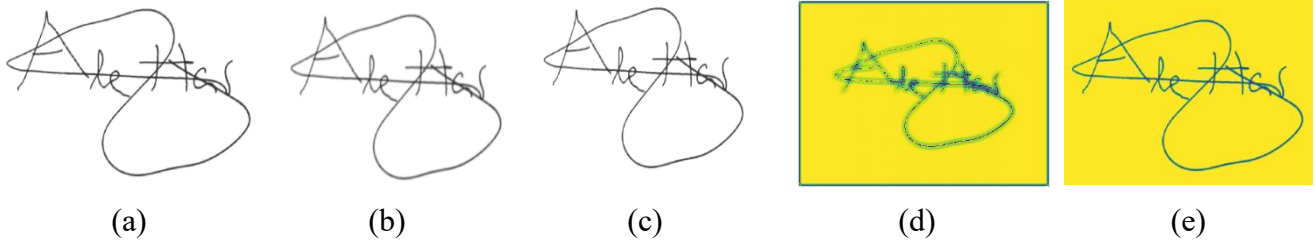


Figure.7. Signature Verification images for dataset1: a) Original images, b) Gaussian Filtering images, c) Conventional Median Filtering images, d) Wiener Filtering images and e) Improved Median Filtering images

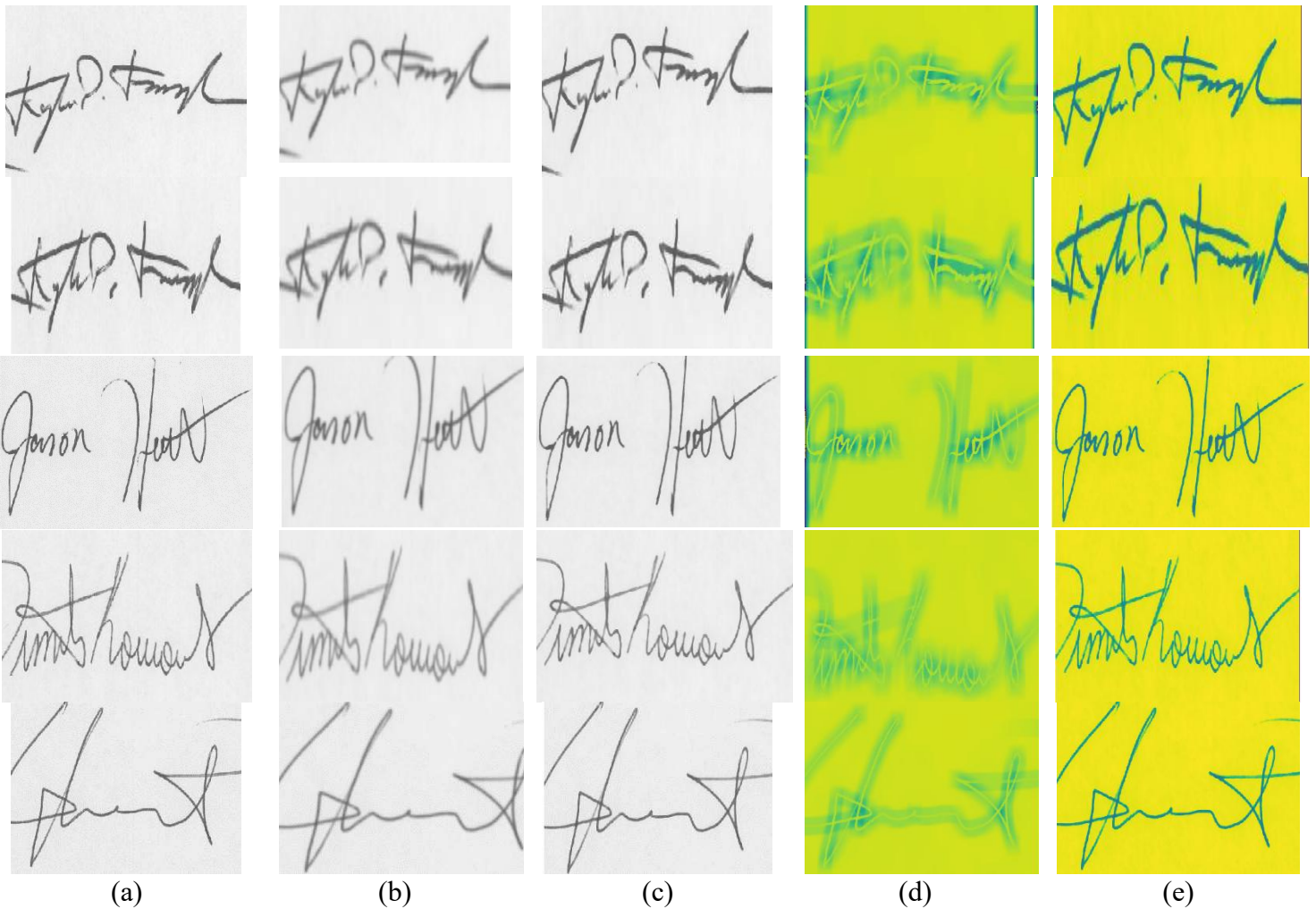


Figure.8. Signature Verification images for dataset2: a) Original images, b) Gaussian Filtering images, c) Conventional Median Filtering images, d) Wiener Filtering images and e) Improved Median Filtering images

4.5 Analysis of PSNR and SSIM

The PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural Similarity Index) analysis presented in Tables 4 and 5 illustrates the superior performance of the Improved Median Filtering technique for signature verification, especially when compared to other commonly used filtering methods like Gaussian Filtering, Weiner Filtering and Conventional Median Filtering. These two metrics, PSNR and SSIM, are widely used to assess the quality of image processing, and higher values in both metrics typically indicate better image quality and less distortion, which is crucial for tasks like signature verification. For a better filtering process, the model should achieve higher PSNR and SSIM values. Mainly, for dataset1, the enhanced median filtering's PSNR is 36.091, even though the Gaussian Filtering is 29.669, the Weiner Filtering is 25.291 and conventional Median filtering is 26.406, respectively. The significantly higher PSNR for the Improved Median Filtering indicates that it is more effective at preserving the integrity of the signature images while removing noise, thus providing cleaner and more accurate data for the subsequent feature extraction and signature verification stages. In addition, while the Gaussian, Weiner and traditional median filtering obtained low SSIM values, the enhanced median filtering's SSIM is 0.905 (dataset 2). Therefore, it is shown that improved median filtering is better. As seen by its higher PSNR and SSIM scores when compared to Gaussian Filtering, Weiner Filtering and Conventional Median Filtering, the results validate that the Improved Median Filtering approach produces images of superior quality. This enhancement in image quality translates into better preprocessing for the signature verification framework, where noise reduction and preservation of critical features are essential for accurate verification. The Improved Median Filtering directly increases the overall efficacy of the signature verification system by enhancing the quality of the input signature images.

Table 4: PSNR and SSIM analysis for Dataset 1

Methods	PSNR (decibel)	SSIM
Gaussian	29.669	0.822
Weiner	25.291	0.730
Conventional Median Filtering	26.406	0.789
Improved Median Filtering	36.091	0.913

Table 5: PSNR and SSIM analysis for dataset 2

Methods	PSNR	SSIM
Gaussian	26.931	0.794
Weiner	19.945	0.679
Conventional Median Filtering	30.079	0.838
Improved Median Filtering	35.993	0.905

4.6 Comparative Analysis of Positive, Negative and Other Metrics for Dataset 1

Figure. 9 provides the performance comparison of various signature verification models, such as hybrid ViT [29], HTCSigNet [30], hybrid deeplearning [7], tSSN [14], Linknet, LSTM, EfficientNet, Bi-GRU, LeNet, Bi-LSTM and RNN + Bi-LSTM are evaluated using four key metrics: Accuracy, Precision, Sensitivity and Specificity, with a training rate of 90%. RNN+Bi-

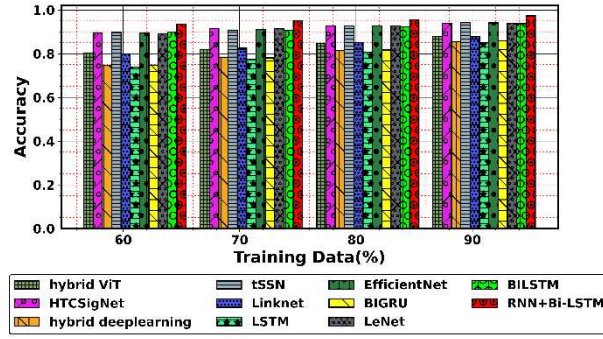
LSTM outperforms all other models across all metrics, achieving the highest Accuracy (0.973), Precision (0.965), Sensitivity (0.975) and Specificity (0.972). This indicates that the hybrid RNN+ Bi-LSTM model is the most robust in terms of correctly identifying both genuine and forged signatures, with a balanced and high overall performance. BiGRU follows closely, achieving the second-best results, with an accuracy of 0.860, precision of 0.833, sensitivity of 0.856 and specificity of 0.862. While it performs well overall, it does not match the performance of the RNN + Bi-LSTM model in terms of accuracy and specificity. The proposed RNN + Bi-LSTM hybrid model offers better results in comparison to traditional methods like hybrid ViT [29], HTCSigNet [30], hybrid deeplearning [7], tSSN [14], Linknet, LSTM, EfficientNet, Bi-GRU, LeNet, Bi-LSTM, primarily due to its ability to capture both temporal dependencies and bidirectional context. The RNN component excels at processing sequential data, which is crucial for analyzing the dynamics of signature images, while the Bi-LSTM captures both past and future context, providing a more holistic view of the signature's structure. This bidirectional approach is particularly valuable in signature verification, where the order and flow of strokes are critical for accurate verification. Additionally, the model leverages ResNet features, which offer deep, residual-based feature learning that improves the accuracy of the model in distinguishing subtle patterns, along with global and local feature extraction, which adds a layer of granularity to the signature representation. The improved preprocessing techniques, including median filtering and image resizing, contribute to noise reduction and better-quality data for feature extraction, reducing the chance of erroneous verification.

Figure 10 depicts the negative metric estimation on RNN + Bi-LSTM over traditional models for the analysis of the signature verification framework. For the exact verification of the signature, the approach must generate fewer ratings in the negative measure. Moreover, the FNR of the RNN

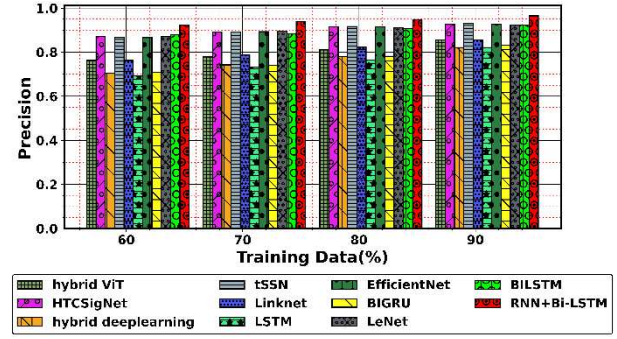
+ Bi-LSTM scheme is 0.049 (training rate =80), though the hybrid ViT [29] (0.145), HTCSigNet [30] (0.081), hybrid deeplearning [7] (0.184), tSSN [14] (0.081), Linknet (0.155), LSTM (0.184), EfficientNet (0.078), BIGRU (0.183), LeNet (0.074) and BILSTM (0.078) correspondingly. Additionally, the greatest FPR is acquired by employing the RNN approach accompanied by LeNet, whereas our RNN + Bi-LSTM methodology recorded the lowest FPR of 0.028. The FNR and FPR values are also significantly better for the RNN + Bi-LSTM model, ensuring that fewer valid signatures are rejected and fewer false positives are generated. This is a critical advantage in signature verification systems, where both false negatives and false positives must be minimized.

The analysis on other measures for RNN + Bi-LSTM is contrasted with the traditional strategies for the signature verification framework, as depicted in Figure. 11. Moreover, the other measure rate must be maximized for a better signature verification framework. Mainly, the F-measure of the RNN + Bi-LSTM scheme is 0.970 at the training rate 90, although the hybrid ViT [29] (0.866), HTCSigNet [30] (0.932), hybrid deeplearning [7] (0.839), tSSN [14] (0.935), Linknet (0.866), LSTM (0.835), EfficientNet (0.934), BIGRU (0.844), LeNet (0.931) and BILSTM (0.930) scored lesser F-measure values. The RNN + Bi-LSTM methodology's MCC and NPV are significantly higher than those of the earlier approaches across all learning percentages. Overall, the RNN + Bi-LSTM methodology's exceptional performance guarantees that it can precisely identify the signatures using enhanced distance measurement, improved median filtering, the hybrid classifier (RNN and Bi-LSTM) and improved score level fusion for the final precise verification. Thus, the hybrid RNN + Bi-LSTM approach performs better over traditional methods in signature verification due to its ability to capture both sequential and bidirectional information, its deep feature extraction capabilities and its effective preprocessing and score-level fusion strategies. These advantages enable the model to achieve higher accuracy, better sensitivity and

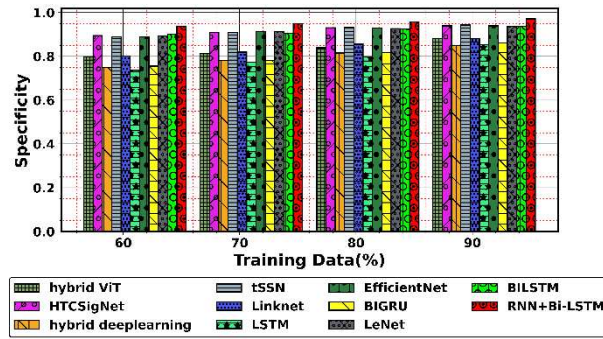
lower error rates, making it a more reliable and robust solution for signature verification tasks compared to existing methods like hybrid ViT [29], HTCSigNet [30], hybrid deeplearning [7], tSSN [14], Linknet, LSTM, EfficientNet, Bi-GRU, LeNet, Bi-LSTM and RNN + Bi-LSTM.



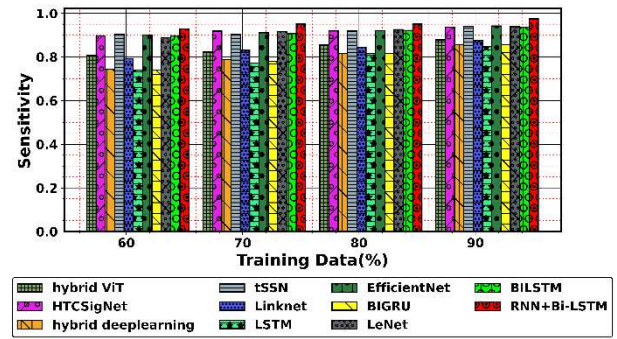
(a)



(b)

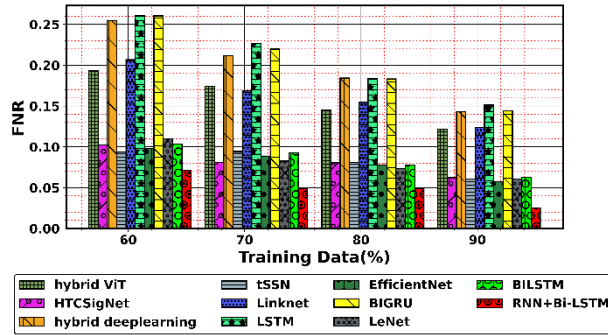


(c)

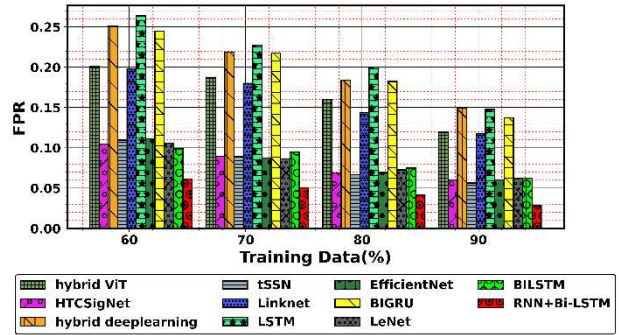


(d)

Figure.9. Examination of RNN + Bi-LSTM over traditional schemes in terms of positive measure for dataset 1



(a)



(b)

Figure.10. Examination of RNN + Bi-LSTM over traditional schemes in terms of negative measure for dataset 1

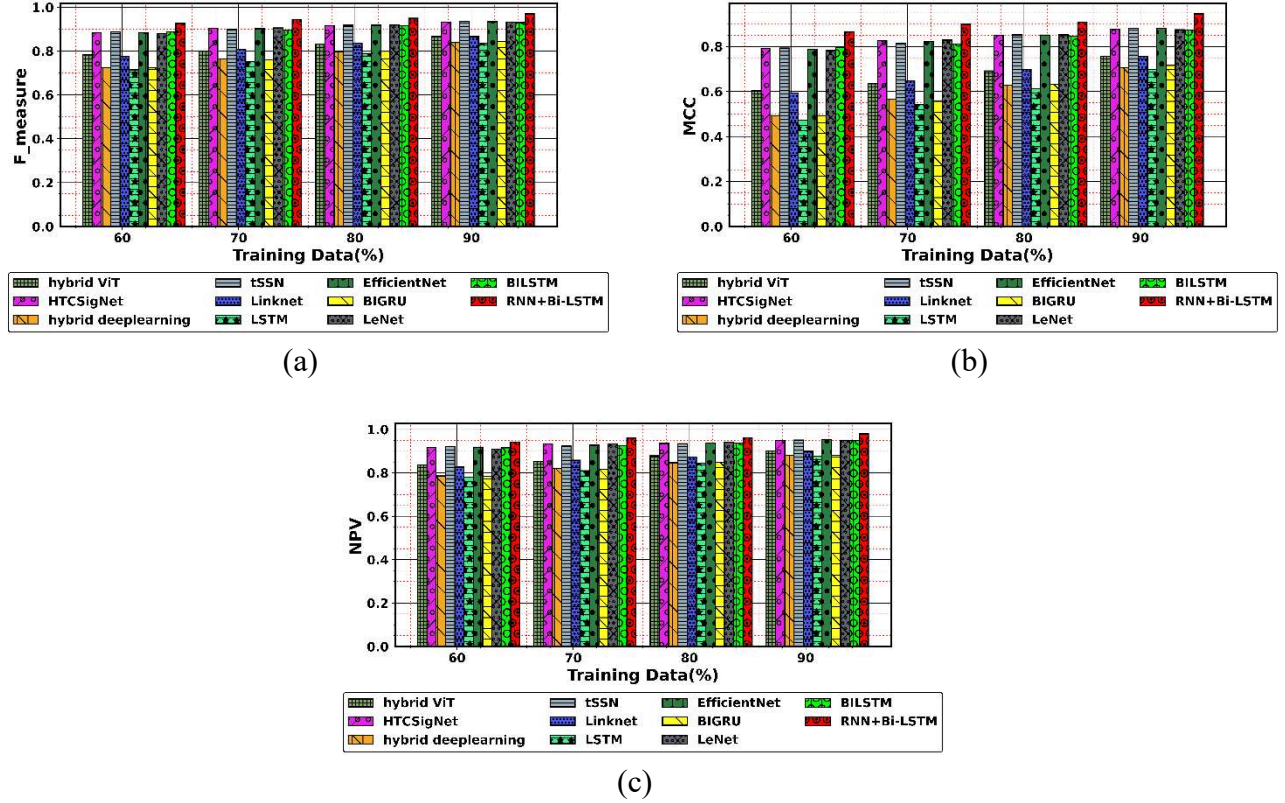


Figure.11. Examination of RNN + Bi-LSTM over traditional schemes in terms of other measures for dataset 1

4.7 Performance analysis on positive, negative and other measures for dataset2

Figure. 12-14 presents the positive metric, negative metric and other metric analysis for Dataset 2.

The models compared include hybrid ViT [29], HTCSigNet [30], hybrid deeplearning [7], tSSN [14], Linknet, LSTM, EfficientNet, Bi-GRU, LeNet, Bi-LSTM and RNN + Bi-LSTM and the hybrid RNN + Bi-LSTM model. Among the models, the RNN+ Bi-LSTM hybrid approach achieves the highest accuracy (0.971), precision (0.976), sensitivity (0.966) and specificity (0.976), demonstrating its superior performance compared to the other models. tSSN [14] ranks second

with notable performance across all metrics, particularly excelling in sensitivity (0.942). tSSN [14] achieves high precision (0.940) and specificity (0.939), while models like hybrid ViT [29], HTCSigNet [30], hybrid deeplearning [7], Linknet, LSTM, EfficientNet, Bi-GRU, LeNet, Bi-LSTM show slightly lower values across all metrics. The tSSN [14] and Bi-LSTM models also provide strong results, but still fall behind the hybrid RNN + Bi-LSTM model in terms of overall performance. The results suggest that the hybrid model combining tSSN [14] and Bi-LSTM is the most effective approach for signature verification tasks in terms of both classification accuracy and the balance of precision, sensitivity and specificity. The RNN + Bi-LSTM hybrid model outperforms all other models in both FNR (0.034) and FPR (0.024) for dataset 2, demonstrating its capacity to reduce false positives and false negatives, which translates to better overall reliability. The LSTM and Bi-GRU models also have relatively higher error rates compared to the hybrid approach, with RNN showing the highest FNR (0.153) and FPR (0.164). Thus, the RNN+ Bi-LSTM model is notable for its exceptional capacity to reduce false positives and false negatives, making it the most efficient model for signature verification.

As a result of enhanced median filtering and an improved distance measure using a hybrid classifier technique (RNN and Bi-LSTM), the RNN + Bi-LSTM scheme is unquestionably more successful at recognizing the signature. The findings demonstrate that the RNN + Bi-LSTM model is significantly more effective for signature verification, as it achieves higher accuracy, sensitivity and F-measure while minimizing false positives and false negatives. The combination of improved preprocessing, hybrid classification and score-level fusion techniques contributes to its robustness, making it a superior choice for signature verification tasks. Compared to other techniques, which either struggle with sequential data or fail to capture sufficient context, the RNN + Bi-LSTM framework delivers a more reliable and accurate solution. These improvements justify the analysis

outcomes, showing that the hybrid approach is well-suited for the complex nature of signature verification.

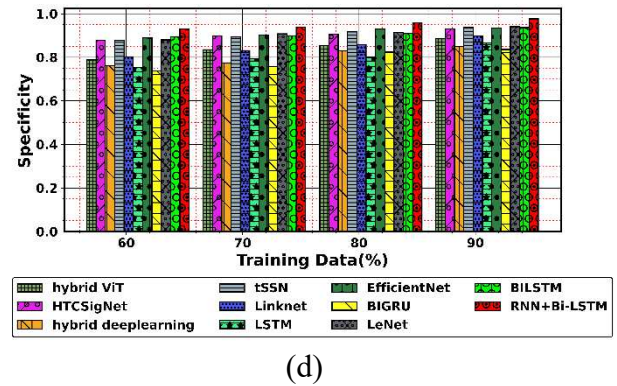
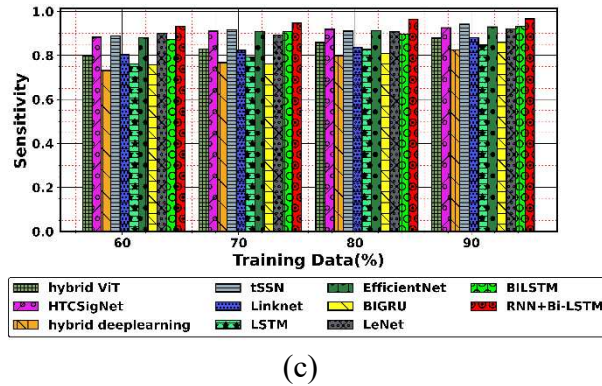
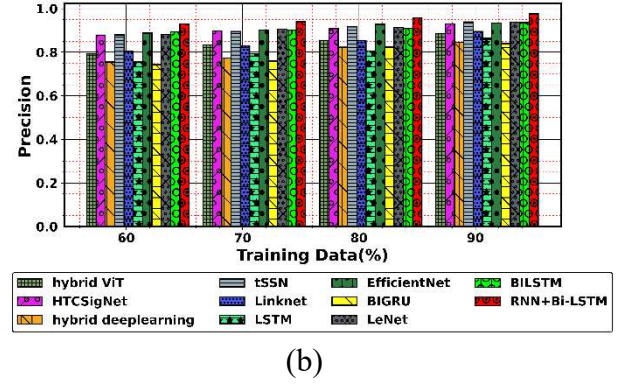
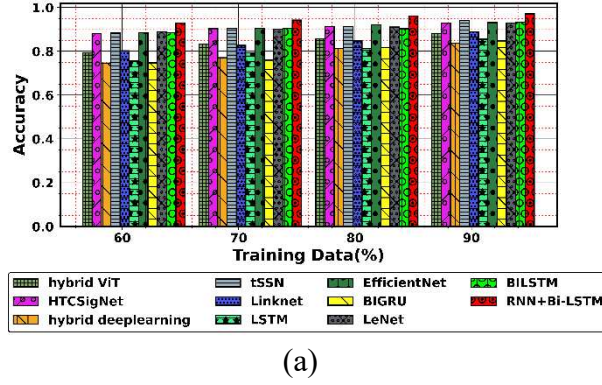


Figure.12. Examination of RNN + Bi-LSTM over conventional models with regards to positive metric for dataset 2

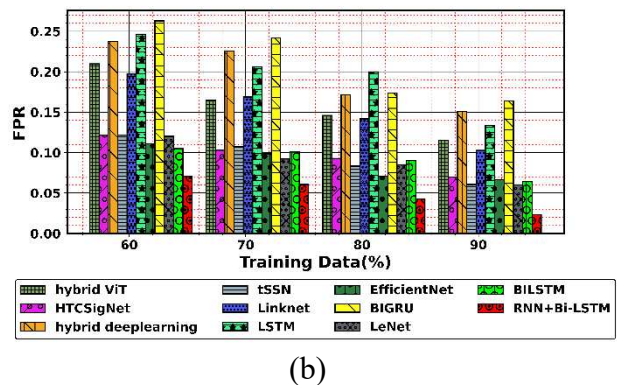
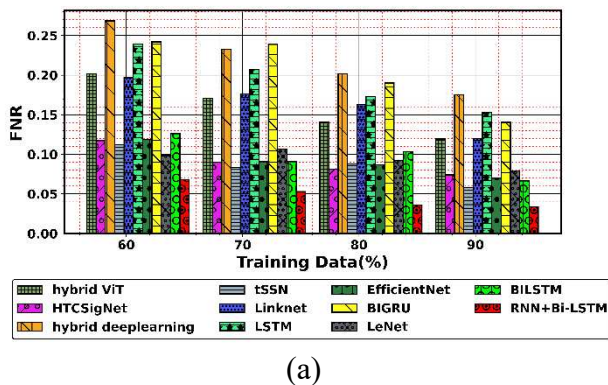


Figure.13. Examination of RNN + Bi-LSTM over conventional models with regards to negative metric for dataset 2

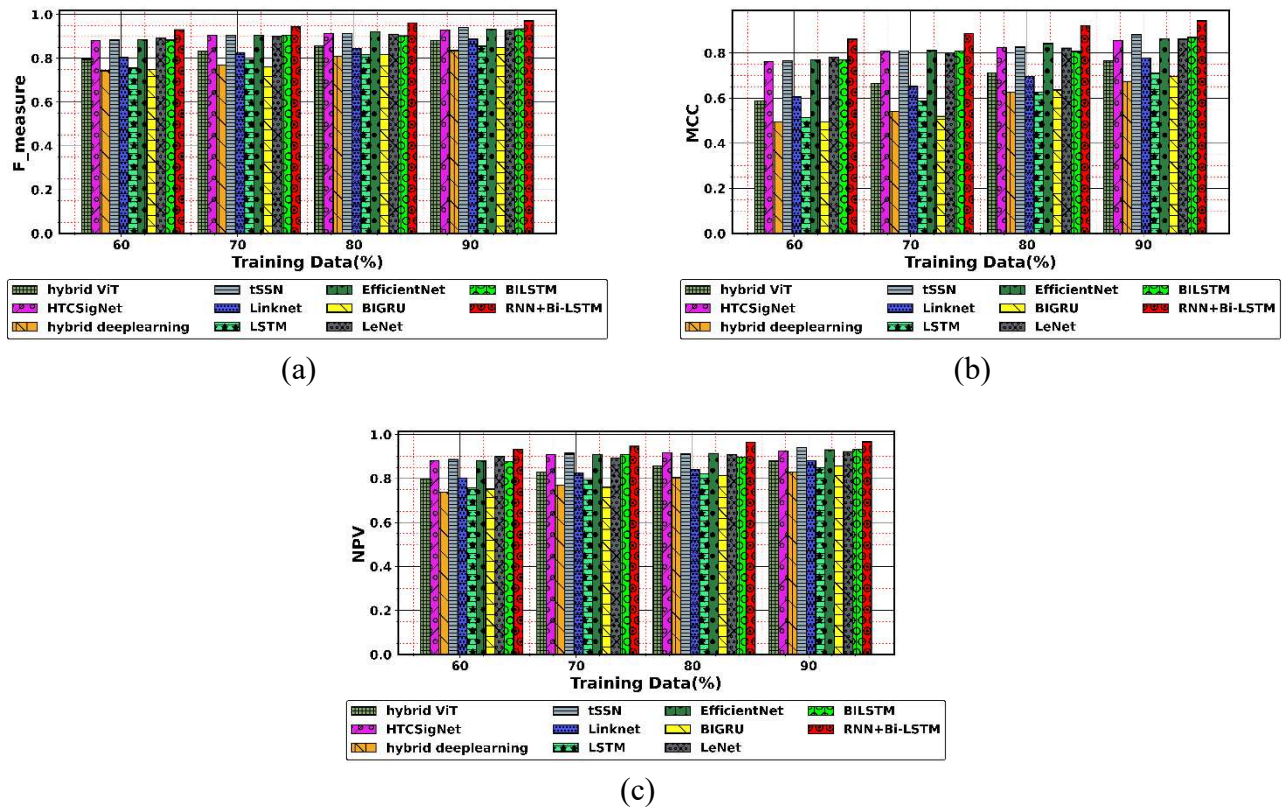


Figure.14. Examination of RNN + Bi-LSTM over existing schemes with regards to other measures for dataset 2

4.8 Ablation study on RNN + Bi-LSTM

Table 6 describes the efficiency of several signature verification methods, such as conventional preprocessed, conventional score fusion, conventional local feature, RESNET feature, global feature and the proposed model across various metrics. The proposed model consistently outperforms the others, achieving the highest accuracy (0.955), sensitivity (0.951), specificity (0.959), precision (0.948), F-measure (0.950) and MCC (0.909), while also minimizing both false positive rate (0.041) and false negative rate (0.049). Other models, such as conv score fusion, also show strong performance, particularly in specificity and precision, but still fall short of the proposed model in terms of overall accuracy and error rates. RESNET feat and global feature

models offer decent results but are not as robust as the proposed model, which remains the most balanced and accurate for signature verification tasks.

Table 6: RNN + Bi-LSTM ablation evaluation, model with traditional preprocessing, model with traditional local feature and model with traditional score fusion for dataset 1

Measures	Conventional Preprocessed	Conventional Local Feature	Conventional Score Fusion	RESNET Feature	Global Feature	RNN+Bi- LSTM
Accuracy	0.902	0.917	0.927	0.924	0.930	0.955
Sensitivity	0.900	0.916	0.927	0.924	0.931	0.951
Specificity	0.903	0.917	0.924	0.921	0.926	0.959
Precision	0.887	0.906	0.920	0.915	0.925	0.948
F-measure	0.893	0.911	0.924	0.919	0.928	0.950
MCC	0.806	0.838	0.863	0.855	0.872	0.909
NPV	0.914	0.926	0.931	0.929	0.932	0.961
FPR	0.097	0.083	0.076	0.079	0.074	0.041
FNR	0.100	0.084	0.073	0.076	0.069	0.049

Table 7 compares the performance of several signature verification models. The proposed model outperforms all other models, achieving the highest accuracy (0.961), sensitivity (0.964), precision (0.958), F-measure (0.961) and MCC (0.921), demonstrating superior classification capability. It also achieves the lowest FPR (0.043) and FNR (0.036), indicating fewer false positives and false negatives. The conv score fusion model performs strongly, with a high sensitivity (0.927) and precision (0.923), while RESNET feat and conv local feature models also show solid results, especially in sensitivity and F-measure. However, the proposed model remains the most balanced

and effective, with better overall performance across all metrics. The global feature model provides good results in precision and specificity, but it still lags behind in terms of the other key performance indicators compared to the proposed approach.

Table 7: Ablation evaluation on RNN + Bi-LSTM, model with conventional preprocessing, model with conventional Local feature and model with conventional score fusion for dataset 2

Measures	Conventional Preprocessed	Conventional Local Feature	Conventional Score Fusion	RESNET Feature	Global Feature	RNN+Bi- LSTM
Accuracy	0.889	0.926	0.926	0.922	0.939	0.961
Sensitivity	0.887	0.928	0.927	0.923	0.942	0.964
Specificity	0.892	0.924	0.924	0.920	0.935	0.957
Precision	0.891	0.921	0.923	0.918	0.932	0.958
F-measure	0.889	0.925	0.925	0.921	0.937	0.961
MCC	0.779	0.856	0.854	0.847	0.882	0.921
NPV	0.888	0.930	0.929	0.925	0.944	0.964
FPR	0.108	0.076	0.076	0.080	0.065	0.043
FNR	0.113	0.072	0.073	0.077	0.058	0.036

4.9 Statistical evaluation of accuracy

Table 8 presents statistical performance measures (mean, median, standard deviation, minimum and maximum) for various signature verification models, including hybrid ViT [29], HTCSigNet [30], hybrid deeplearning [7], tSSN [14], Linknet, LSTM, EfficientNet, Bi-GRU, LeNet, Bi-LSTM and the hybrid RNN + Bi-LSTM. Superior overall performance is indicated by the RNN + Bi-

LSTM model, which regularly exhibits the highest mean (0.953) and median (0.953) accuracy. Additionally, its standard deviation (0.014) is the lowest, indicating higher consistency across assessments. In contrast, the other models exhibit slightly lower means and medians, with tSSN [14], EfficientNet and Bi-LSTM showing the next best performance around 0.91. The LSTM model has the lowest mean (0.792) and median (0.790), reflecting its relatively poorer performance. The maximum accuracy for the RNN + Bi-LSTM model (0.973) is notably higher than all other models, highlighting its superior ability to handle varying signature verification tasks.

Table 8: Statistical assessment of accuracy for dataset 1

Methods	Mean	Median	Standard Deviation	Minimum	Maximum
hybrid ViT [29]	0.837	0.832	0.029	0.802	0.880
HTCSigNet [30]	0.919	0.920	0.016	0.896	0.939
Hybrid deeplearning [7]	0.800	0.800	0.039	0.747	0.854
tSSN [14]	0.919	0.918	0.017	0.897	0.942
Linknet	0.838	0.838	0.030	0.798	0.880
LSTM	0.792	0.790	0.042	0.738	0.851
EfficientNet	0.919	0.919	0.017	0.894	0.941
BIGRU	0.802	0.799	0.041	0.748	0.860
LeNet	0.918	0.921	0.017	0.892	0.938
BILSTM	0.917	0.915	0.015	0.899	0.937
RNN+Bi-LSTM	0.953	0.953	0.014	0.934	0.973

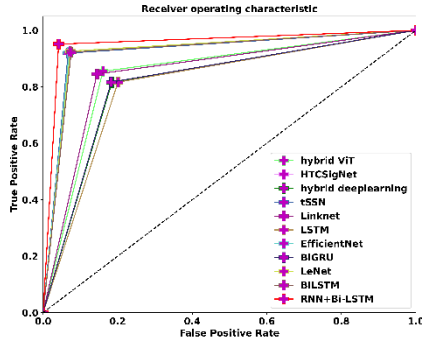
Table 9 represents the statistical performance metrics for dataset 2. The RNN + Bi-LSTM model performs better overall, as seen by its highest mean accuracy (0.951) and median accuracy (0.952). Additionally, it has the lowest standard deviation (0.016), indicating that it consistently produces results from various runs. EfficientNet ranks second in mean (0.911) and median (0.913), performing relatively well with a moderate standard deviation (0.018). In summary, the RNN + Bi-LSTM model offers the best combination of high accuracy, consistency and performance across different metrics.

Table 9: Statistical assessment of accuracy for dataset 2

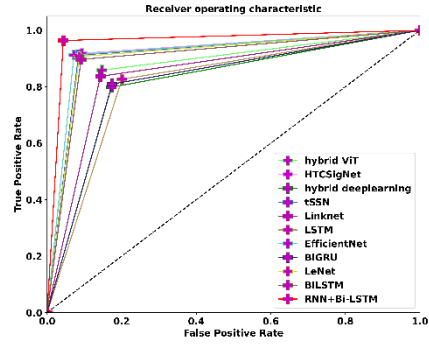
Methods	Mean	Median	Standard Deviation	Minimum	Maximum
hybrid ViT [29]	0.841	0.844	0.033	0.794	0.883
HTCSigNet [30]	0.906	0.908	0.017	0.881	0.928
Hybrid deeplearning [7]	0.792	0.792	0.035	0.747	0.837
tSNN [14]	0.911	0.910	0.021	0.883	0.941
Linknet	0.842	0.837	0.031	0.803	0.888
LSTM	0.805	0.803	0.036	0.757	0.856
EfficientNet	0.911	0.913	0.018	0.885	0.932
BIGRU	0.793	0.789	0.041	0.747	0.848
LeNet	0.908	0.906	0.015	0.891	0.931
BILSTM	0.907	0.904	0.018	0.884	0.935
RNN+Bi-LSTM	0.951	0.952	0.016	0.931	0.971

4.10 Analysis of ROC curve and EER

A graphic representation of a matching or classification system's performance is called a ROC curve. It sheds light on how well a system can discriminate between positive and negative occurrences. The ROC curve of both datasets is illustrated in Figure. 15. An important statistic for biometric authentication and other classification tasks is the EER. In other words, the EER is the threshold at which both acceptance and rejection errors are equal. The analysis of EER on both datasets is tabulated in Table 10. The Equal Error Rate (EER) analysis in Table 10 highlights the performance of the proposed RNN+Bi-LSTM model across two datasets (Dataset 1 and Dataset 2) with varying training data percentages (60%, 70%, 80% and 90%). For Dataset 1, the proposed model consistently achieves the lowest or near-lowest EER values at all training data splits, demonstrating its robustness and efficiency. Even with comparatively less data, the model can learn and generalize well, as evidenced by its best EER of 1.013 at 60% training data. The EER slightly increases at higher training splits but remains well below competing models, confirming strong performance in reducing both false acceptance and false rejection errors. Similarly, after 70% training data, the RNN+Bi-LSTM model for Dataset 2 obtains the best EER of 0.770, outperforming other architectures like hybrid ViT [29], HTCSigNet [30] and Bi-LSTM models. The slightly higher EER values at other splits still compare favorably with alternative methods, demonstrating consistent and reliable signature verification performance. Overall, the EER values across different training data percentages confirm that the RNN+Bi-LSTM framework provides superior generalization and robustness for signature verification across diverse datasets, even with varying amounts of training data.



(a) Dataset 1



(b) Dataset 2

Figure. 15. ROC curve of a) Dataset 1 and b) Dataset 2

Table 10: EER Analysis on two datasets

Dataset 1				
Methods	60	70	80	90
hybrid ViT [29]	0.810	0.613	0.501	0.401
HTCSigNet [30]	0.409	1.320	1.153	0.409
hybrid deeplearning [7]	3.093	0.708	2.995	2.414
tSSN [14]	0.913	2.402	0.457	0.263
Linknet	0.050	0.720	2.054	1.749
LSTM	0.693	0.172	2.687	1.912
EfficientNet	0.729	0.778	1.595	0.325
BIGRU	2.084	0.268	1.670	2.383
LeNet	2.174	1.494	0.741	1.924
BiLSTM	2.117	0.995	1.307	0.242
RNN+Bi-LSTM	0.270	0.770	0.670	1.005

Dataset 2				
hybrid ViT [29]	0.810	0.613	0.501	0.401
HTCSigNet [30]	0.409	1.320	1.153	0.409
hybrid deeplearning [7]	3.093	0.708	2.995	2.414
tSSN [14]	0.913	2.402	0.457	0.263
Linknet	0.050	0.720	2.054	1.749
LSTM	0.693	0.172	2.687	1.912
EfficientNet	0.729	0.778	1.595	0.325
BIGRU	2.084	0.268	1.670	2.383
LeNet	2.174	1.494	0.741	1.924
BILSTM	2.117	0.995	1.307	0.242
RNN+Bi-LSTM	0.270	0.770	0.670	1.005

4.11 Feature extraction analysis

A thorough comparison of various feature extraction methods is provided in Table 11, which highlights the suggested feature set's superior performance. The suggested approach obtains the highest accuracy of 0.950 on Dataset 1, outperforming SIFT (0.927), HOG (0.921) and individual deep (0.914) or global (0.918) features. Similar results are observed on Dataset 2, where the proposed approach reaches 0.945 accuracy, compared to SIFT (0.925) and HOG (0.917). Sensitivity and precision also peak with the proposed method, achieving 0.952 and 0.941 on Dataset 1. The F-measure values further support its effectiveness, with 0.947 and 0.948 for Datasets 1 and 2, respectively. Classification decisions are strongly correlated, as confirmed by

MCC values of 0.900 and 0.890. Lower FPR and FNR values with the proposed method indicate fewer false rejections and acceptances. These results substantiate the benefit of combining deep, global and local features over using traditional methods alone, validating the selection of feature sets in the proposed framework.

Table 11: Comparative Analysis of Feature Extraction on Dataset 1 and 2

Dataset 1						
Metrics	Conventional Local Feature	ResNet Feature	Global Feature	Shift Feature	HOG Feature	Proposed Feature Set
Accuracy	89.4%	91.4%	91.8%	92.7%	92.1%	95.0%
Sensitivity	89.7%	91.8%	92.7%	93.3%	92.5%	95.2%
Specificity	88.9%	90.7%	90.2%	91.9%	91.3%	94.8%
Precision	88.2%	90.6%	92.1%	92.3%	91.4%	94.1%
F-measure	88.9%	91.2%	92.4%	92.8%	92.0%	94.7%
MCC	79.9%	84.1%	86.7%	86.9%	85.5%	90.0%
NPV	90.3%	91.9%	91.0%	92.9%	92.4%	95.8%
FPR	11.1%	9.3%	9.8%	8.1%	8.7%	5.2%
FNR	10.3%	8.2%	7.3%	6.7%	7.5%	4.8%
Dataset 2						
Accuracy	91.7%	92.0%	90.9%	92.5%	91.7%	94.5%
Sensitivity	92.0%	92.4%	91.4%	93.0%	92.2%	95.3%
Specificity	91.2%	91.5%	89.9%	91.7%	90.8%	93.7%
Precision	90.5%	90.9%	90.3%	91.8%	91.1%	94.2%

F-measure	91.3%	91.6%	90.8%	92.4%	91.6%	94.8%
MCC	84.2%	84.9%	83.6%	85.8%	84.7%	89.0%
NPV	92.5%	92.8%	91.1%	92.9%	92.0%	94.8%
FPR	8.8%	8.5%	10.1%	8.3%	9.2%	6.3%
FNR	8.0%	7.6%	8.6%	7.0%	7.8%	4.7%

4.12 Confusion Matrix

Figure 16 shows the confusion matrix results for the proposed RNN+Bi-LSTM model evaluated on Dataset 1 and Dataset 2. For Dataset 1, the model correctly classifies 820 genuine signatures and 718 forged signatures. It misclassifies 46 forged signatures as genuine and 36 genuine signatures as forged. In Dataset 2, 237 genuine and 261 forged signatures are correctly identified, while 17 forged signatures are wrongly accepted and 13 genuine signatures are incorrectly rejected. These results demonstrate that the model performs well in distinguishing genuine from forged signatures in both datasets. The number of false acceptances and false rejections is relatively low, indicating strong classification reliability. The confusion matrices provide valuable insights into model performance by illustrating the balance between true positives and false positives. Overall, the proposed model achieves effective verification, maintaining a good trade-off between security and usability. This qualitative analysis supports the robustness of the proposed framework across different datasets.

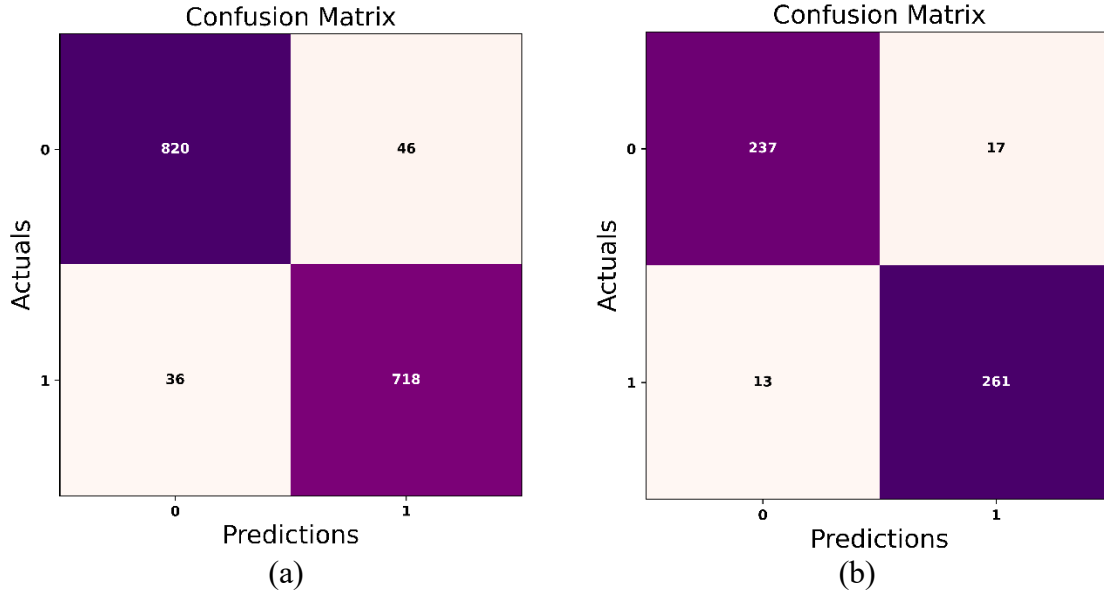


Figure. 16: Analysis on Confusion Matrix (a) Dataset 1 and (b) Dataset 2

4.13 Computational Time Analysis

Table 12 presents the execution times for various signature verification methods across Dataset 1 (GPDS-150) and Dataset 2 (CEDAR) to assess the computational efficiency of the proposed framework. The proposed hybrid RNN+Bi-LSTM model achieves the lowest processing time, with 57.454 seconds for Dataset 1 and 37.300 seconds for Dataset 2. Compared to complex models such as LeNet (85.039s, 42.542s) and BIGRU (96.165s, 70.518s), the proposed method demonstrates a noticeable improvement in efficiency. Traditional models like LSTM and Linknet also show higher time consumption, especially on Dataset 1. The hybrid RNN+Bi-LSTM model outperforms most deep learning baselines in terms of speed and accuracy, according to the results. These findings highlight the model's potential for real-time or near-real-time signature verification applications, particularly in systems requiring prompt authentication. While deep models typically raise concerns over latency, the proposed architecture balances accuracy and speed effectively, making it practical for deployment in real-world verification scenarios.

Table 12: Computational Time Analysis of proposed signature verification method over existing models on Dataset 1 and Dataset 2

Methods	Computing Time (seconds)	
	Dataset 1	Dataset 2
hybrid ViT [29]	81.935	61.794
HTCSigNet [30]	76.815	63.097
hybrid deeplearning [7]	81.646	59.574
tSSN [14]	78.979	79.288
Linknet	75.744	62.368
LSTM	79.709	50.787
EfficientNet	68.393	59.698
BIGRU	96.288	70.518
LeNet	85.039	42.542
BILSTM	60.397	48.902
RNN+Bi-LSTM	57.454	37.300

4.14 Noise Variation analysis

The noise variation analysis is described in Table 13 to evaluate the impact of image quality and noise on verification performance by applying different levels of Gaussian noise (0.2 to 0.8) on both Dataset 1 (GPDS-150) and Dataset 2 (CEDAR). Table 13 demonstrates that the suggested model continuously maintains good accuracy in spite of noise, with values ranging from 0.937 to 0.922 for Dataset 1 and 0.934 to 0.925 for Dataset 2 under increasing noise levels. Sensitivity and

specificity slightly decline with stronger noise, yet remain above 0.918 in all cases, confirming the model's resilience. F-measure and precision values also stay robust, with Dataset 1 achieving 0.934–0.918 and Dataset 2 maintaining 0.934–0.923. MCC values above 0.849 (Dataset 1) and 0.857 (Dataset 2) further support stable classification. Although both FPR and FNR increase slightly with heavier noise, the degradation is minimal, demonstrating strong noise tolerance. The best performance is observed when no noise is added, with accuracies of 0.950 (Dataset 1) and 0.945 (Dataset 2). These findings show that the framework is not only effective with clean images but also robust under poor-quality or noisy signature conditions.

Table 13: Noise Variation Analysis under Different Levels of Gaussian Noise

Dataset 1					
Measures	Gaussian noise (0.2)	Gaussian noise (0.4)	Gaussian noise (0.6)	Gaussian noise (0.8)	proposed without noise
Accuracy	93.7%	93.4%	92.5%	92.2%	95.0%
Sensitivity	94.0%	94.0%	92.8%	92.5%	95.2%
Specificity	93.2%	92.5%	92.1%	91.8%	94.8%
Precision	92.9%	93.1%	91.4%	91.1%	94.1%
F-measure	93.4%	93.5%	92.1%	91.8%	94.7%
MCC	87.9%	88.4%	85.5%	84.9%	90.0%
NPV	94.3%	93.4%	93.3%	93.1%	95.8%
FPR	6.8%	7.5%	7.9%	8.2%	5.2%
FNR	6.0%	6.0%	7.2%	7.5%	4.8%
Dataset 2					

Accuracy	93.4%	92.8%	92.7%	92.5%	94.5%
Sensitivity	94.0%	93.2%	93.1%	93.0%	95.3%
Specificity	92.7%	92.3%	92.2%	92.0%	93.7%
Precision	92.9%	91.9%	91.9%	91.7%	94.2%
F-measure	93.4%	92.5%	92.5%	92.3%	94.8%
MCC	87.8%	86.0%	85.9%	85.7%	89.0%
NPV	93.7%	93.5%	93.4%	93.2%	94.8%
FPR	7.3%	7.7%	7.8%	8.0%	6.3%
FNR	6.0%	6.8%	6.9%	7.0%	4.7%

4.15 K-fold validation analysis

Table 14 provides a k-fold validation analysis to evaluate model stability and mitigate overfitting concerns. The suggested RNN+Bi-LSTM model obtains the best and most reliable accuracy on Dataset 1, ranging from 0.959 to 0.971, followed by BiGRU (0.892–0.930) and LeNet (0.930–0.946). Hybrid ViT [29] also performs strongly with scores up to 0.928, while older models like Linknet show weaker accuracy, ranging from 0.875 to 0.904. Similar trends are observed in Dataset 2, where RNN+Bi-LSTM again outperforms others, reaching 0.969 accuracy across multiple folds. HTCSigNet [30] and tSSN [14] show more variance, indicating less generalization. The consistent strong performance of the proposed model across all folds and both datasets confirm its strong generalization ability and resistance to overfitting. This k-fold validation analysis strengthens the reliability of the proposed approach and directly addresses concerns related to overfitting.

Table 14: Analysis on K-fold validation

Dataset 1					
Models	fold-1	fold-2	fold-3	fold-4	fold-5
hybrid ViT [29]	0.882	0.903	0.928	0.911	0.941
HTCSigNet [30]	0.928	0.906	0.932	0.918	0.916
hybrid deeplearning [7]	0.895	0.905	0.895	0.874	0.874
tSSN [14]	0.823	0.914	0.815	0.929	0.910
Linknet	0.896	0.904	0.890	0.875	0.878
LSTM	0.898	0.928	0.900	0.885	0.889
EfficientNet	0.943	0.944	0.930	0.934	0.928
BIGRU	0.902	0.892	0.914	0.930	0.910
LeNet	0.945	0.946	0.932	0.936	0.930
BILSTM	0.938	0.939	0.925	0.929	0.923
RNN+Bi-LSTM	0.960	0.960	0.971	0.959	0.965
Dataset 2					
hybrid ViT [29]	0.871	0.831	0.867	0.880	0.899
HTCSigNet [30]	0.931	0.895	0.871	0.927	0.915
hybrid deeplearning [7]	0.877	0.906	0.876	0.856	0.856
tSSN [14]	0.906	0.919	0.928	0.906	0.936
Linknet	0.877	0.885	0.872	0.869	0.859
LSTM	0.879	0.858	0.882	0.866	0.851
EfficientNet	0.910	0.911	0.898	0.901	0.896

BIGRU	0.938	0.929	0.911	0.907	0.913
LeNet	0.926	0.927	0.933	0.917	0.911
BILSTM	0.919	0.929	0.906	0.940	0.904
RNN+Bi-LSTM	0.961	0.951	0.953	0.969	0.960

4.16 Cross Validation Analysis

To assess the suggested signature verification framework's capacity for generalization, a cross-dataset validation is carried out in Table 15. In the first experiment, the model was trained using Dataset 1 (GPDS-150) and tested on Dataset 2 (CEDAR). In the second experiment, the training and testing datasets were reversed, Dataset 2 (CEDAR) was used for training and Dataset 1 (GPDS-150) for testing. As shown in Table 15, high performance was consistently observed in both configurations, with accuracy values of 0.966 and 0.953, indicating strong generalization across different signature styles and populations. Sensitivity and specificity remained balanced (above 0.95), suggesting reliable detection of both genuine and forged signatures. Precision and F-measure scores above 0.96 demonstrate consistent prediction quality, while MCC values near 0.90 further confirm robust classification performance. Notably, low FPR (≤ 0.033) and FNR (≤ 0.035) reflect minimal error rates during cross-dataset evaluation. Thus, the framework is well-suited for other datasets as well, clearly demonstrating its strong generalizability across varying signature sources.

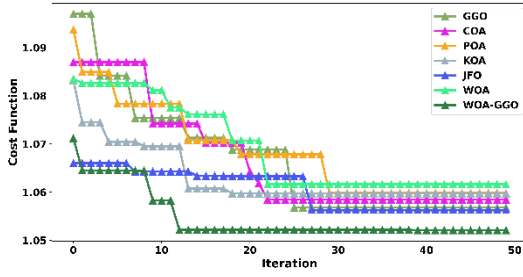
Table 15: Analysis on Cross-Dataset Validation

Measures	Training: Dataset 1 and	Training: Dataset 2 and
	Testing: Dataset 2	Testing: Dataset 1
Accuracy	0.966	0.953

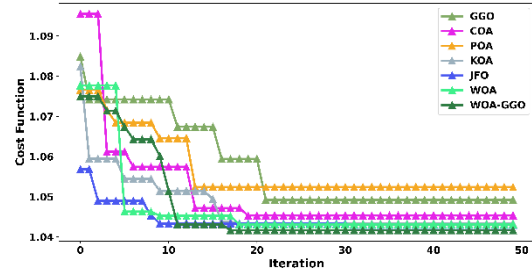
Sensitivity	0.965	0.951
Specificity	0.967	0.954
Precision	0.967	0.943
F-measure	0.966	0.947
MCC	0.932	0.904
NPV	0.965	0.960
FPR	0.033	0.046
FNR	0.035	0.049

4.17 Convergence Study

This section describes the convergence study for both the proposed WOA-GGO method and GGO, COA, POA, KOA, JFO and WOA as shown in the Figure 17. Similarly the iteration has been modified from 10 to 50. In the accordance with the developed method, the suggested model attained greater error rate of 1.010, although it gained the least error rate of 1.052 at the final (50th) iteration. Furthermore, during the final iteration, the traditional methods such as and GGO, COA, POA, KOA, JFO and WOA acquired the error value ranges from 1.055 to 1.063 in the dataset 1. Similarly in the dataset 2, the suggested model attained greater error rate of 1.010, although it gained the least error rate of 1.042 at the final (50th) iteration. Moreover, during the final iteration the traditional methods such as and GGO, COA, POA, KOA, JFO and WOA acquired the error value ranges from 1.042 to 1.055 in the dataset 2. As a result, the proposed WOA-GGO algorithm performed better than the other methods.



(a)



(b)

Figure. 17: Convergence Analysis on (a) Dataset 1 and (b) Dataset 2

4.18 PR-Curve Analysis

The Precision-Recall (PR) curve is a graph that shows the relationship between the precision and recall to evaluate the performance of a classification model. The performance of the proposed model was compared with the conventional models including hybrid ViT [29], HTCSigNet [30], Hybrid deeplearning [7], tSNN [14], Linknet, LSTM, EfficientNet, BIGRU, LeNet and BiLSTM as shown in the Figure 18, for dataset 1 and dataset 2. The comparison was performed to evaluate the precision and recall performance of each model. For Dataset 1, the proposed RNN+Bi-LSTM achieved the highest PR curve value of 0.998. The proposed model maintained a better balance between precision and recall than the existing models. This indicates that the proposed model provides better classification performance than the existing models. The result shows the improved classification performance of the proposed RNN + Bi-LSTM model. Overall. The proposed RNN + Bi-LSTM performed better than the conventional model on both datasets.

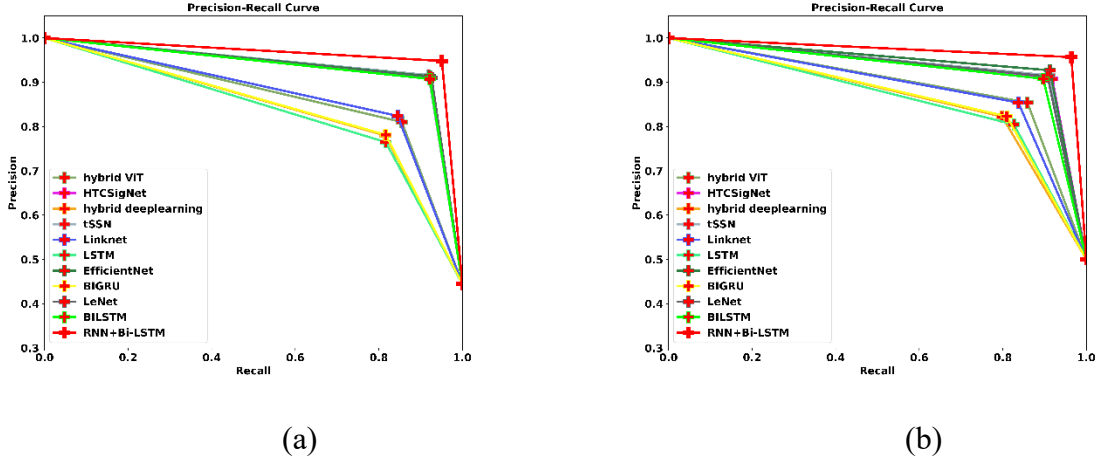


Figure 18: PR Curves for (a) Dataset 1 and (b) Dataset 2

4.19 Feature selection analysis performance

Feature selection performance indicates the effectiveness of a feature selection method in selecting the important features to improve the prediction accuracy and reducing the computation time. The performance of the WOA-GGO was evaluated using dataset 1 and dataset 2, compared with the conventional methods including GGO, WOA, COA, POA, KOA and JFO as shown in the figure 19. The comparison is carried out based on the selected features and their impact on prediction performance. For Dataset 1. The WOA-GGO selected the optimal features, which improved the prediction performance. The WOA-GGO achieved the highest accuracy of 0.951 and the lowest MCC of 0.900. Similarly, for Dataset 2, the WOA-GGO obtained the highest accuracy of 0.951 and the lowest MCC of 0.903. These results show the effectiveness of the proposed feature selection approach in identifying the features while maintaining the high prediction performance. Overall, the experimental results show that the proposed feature selection method effectively improves the prediction performance on both datasets.

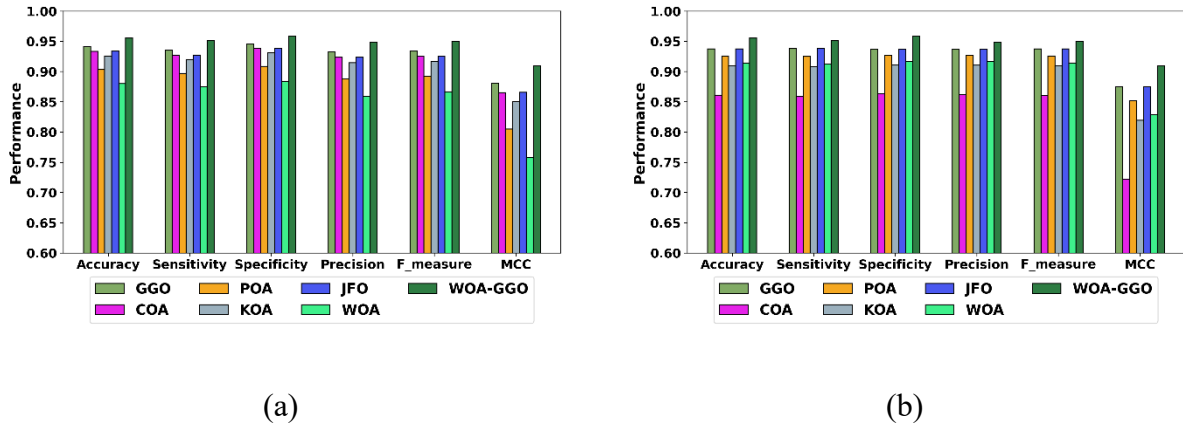


Figure 19: Feature Selection Performance for dataset 1 and dataset 2

4.20 Analysis on Friedman chi-square Test

The Friedman Chi-Square test is a non-parametric statistical test used to compare the performance of multiple paired methods assuming a normal distribution. The Friedman test was performed to compare the proposed model with hybrid ViT [29], HTCSigNet [30], Hybrid deep learning [7], tSSN [14], Linknet, LSTM, Efficientnet, BIGRU, LeNet and BiLSTM using Dataset 1 and Dataset 2 as shown in the Table 16. The Friedman Chi-square test was performed at a significance level of $\alpha = 0.05$ to determine whether the observed performance differences among the compared methods, whereas a p-value greater than or equal to 0.05 indicates that the observed differences are not statistically significant. For Dataset 1, the obtained p-value ranges from 0.0095 to 0.0267 and for Dataset 2, they range from 0.0093 to 0.0265. As all the obtained p-values are less than the specified significance level of 0.05, the results indicate the statistically significant differences among the evaluated models for both datasets. This indicates that the performance differences among the evaluated models are statistically significant. The results show that the proposed model shows statistically significant performance compared with the existing approaches across both datasets.

Table 16: Friedman chi-square Test analysis

Methods	Dataset 1($\alpha = 0.05$)	Dataset 2 ($\alpha = 0.05$)
hybrid ViT [29]	0.0096	0.0101
HTCSigNet [30]	0.0237	0.0168
hybrid deep learning [7]	0.0095	0.0093
tSSN [14]	0.0261	0.0265
Linknet	0.0097	0.0100
LSTM	0.0095	0.0095
EfficientNet	0.0267	0.0213
BIGRU	0.0096	0.0096
LeNet	0.0254	0.0158
BILSTM	0.0196	0.0180

4.21 Analysis of the T-test Based Accuracy

A T-test is utilized to assess the mean of two groups and a p-value is a measure of indication against a null hypothesis. Table 17 presents and T-test analysis on RNN + Bi-LSTM and traditional methodologies for Signature Verification. As shown in Table 17, the T-Test of the RNN + Bi-LSTM approach, showcasing statistically significant difference than the compared models with hybrid ViT [29] at 0.0008, HTCSigNet [30] at 0.0290, hybrid deep learning [7] at 0.0007, tSSN [14] at 0.0340, Linknet at 0.0010, LSTM at 0.0007, EfficientNet at 0.0350, BIGRU at 0.0009, LeNet at 0.0325 and BILSTM at 0.0208 respectively. A T-values less than 0.05 indicates that the result is statistically significant, whereas a t-value greater than 0.05 indicates that the result is not

statistically significant. Also, it helps to make normal distribution and the improve the performance. The proposed model also exhibits the stable performance while effectively handling the large-scale genuine and forged signature samples, thereby improving the robustness, scalability and overall effectiveness of the signature verification.

Table 17: Comparative T-Test Analysis of Model Accuracy for dataset 1 and dataset 2

Methods	Dataset 1($\alpha = 0.05$)	Dataset 2($\alpha = 0.05$)
hybrid ViT [29]	0.0008	0.0019
HTCSigNet [30]	0.0290	0.0152
hybrid deep learning [7]	0.0007	0.0004
tSSN [14]	0.0340	0.0346
Linknet	0.0010	0.0016
LSTM	0.0007	0.0007
EfficientNet	0.0350	0.0244
BIGRU	0.0009	0.0008
LeNet	0.0325	0.0132
BILSTM	0.0208	0.0177

4.22 Feature Selection analysis

Feature Selection analysis evaluates the ability of a feature selection method to select the features that improve the prediction performance while reducing the computational complexity. The performance of the feature selection method was evaluated using the metrics of accuracy, sensitivity, specificity, precision, F-measure, MCC, NPV, FPR and FNR. These metrics were used to assess the effectiveness of various feature selection methods, including Mutual Information

(MI), Correlation, Chisquare and Principal Component Analysis (PCA). The comparative analysis of dataset 1 and dataset 2 as shown in the Tables 18 and 19. For dataset 1, the MI method ranges from 0.0869 to 0.9263, the correlation method ranges from 0.0642 to 0.9435, the Chi-Square method ranges from 0.1076 to 0.9100 and the PCA method ranges from 0.0755 to 0.9349. For dataset 2, the MI method ranges from 0.0836 to 0.9164, the correlation method ranges from 0.0940 to 0.9060, the Chi-square method ranges from 0.1112 to 0.8888 and the PCA method ranges from 0.1043 to 0.8957. The obtained results indicate that each feature selection method shows different level of performance for the evaluated metrics. The analysis shows the performance of the evaluated feature selection methods for both datasets.

Table 18: Comparative Analysis of Feature Selection Performance for Dataset 1

Metrics	MI	Correlation	Chisquare	PCA
Accuracy	0.9077	0.9315	0.8890	0.9196
Sensitivity	0.8964	0.9237	0.8815	0.9100
Specificity	0.9131	0.9358	0.8924	0.9245
Precision	0.8811	0.9147	0.8612	0.8979
F-measure	0.8886	0.9192	0.8711	0.9039
MCC	0.8084	0.8589	0.7725	0.8337
NPV	0.9263	0.9435	0.9100	0.9349
FPR	0.0869	0.0642	0.1076	0.0755
FNR	0.1036	0.0763	0.1185	0.0900

Table 19: Comparative Analysis of Feature Selection Performance for Dataset 2

Metrics	MI	Correlation	Chisquare	PCA
----------------	-----------	--------------------	------------------	------------

Accuracy	0.9154	0.9043	0.8847	0.8933
Sensitivity	0.9144	0.9026	0.8888	0.8908
Specificity	0.9164	0.9060	0.8806	0.8957
Precision	0.9159	0.9053	0.8817	0.8948
F-measure	0.9151	0.9039	0.8852	0.8928
MCC	0.8308	0.8088	0.7694	0.7867
NPV	0.9150	0.9035	0.8878	0.8920
FPR	0.0836	0.0940	0.1194	0.1043
FNR	0.0856	0.0974	0.1112	0.1092

4.23 Feature Selection-Based Optimization

Feature selection-based optimization is the process of using an optimization algorithm to identify the features, thereby improving the prediction performance while reducing the computation time. The performance of the WOA-GGO was evaluated on Dataset 1 and Dataset 2 and compared with conventional optimization algorithms, including GGO, WOA, COA, POA, KOA and JFO as shown in the Tables 20 and 21. The comparison was carried out using the performance metrics of accuracy, sensitivity, specificity, precision, F-measure, MCC, NPV, FPR and FNR to evaluate the effectiveness of the feature selection process. For Dataset 1, the WOA-GGO achieved the highest NPV of 0.961 and the lowest FPR of 0.041, indicating the better classification performance. Among the conventional optimization algorithms, GGO obtained the highest NPV of 0.948 and the lowest FPR of 0.054. Similarly, for Dataset 2, the WOA-GGO achieved the highest NPV of 0.961 and the lowest FPR of 0.041. In comparison, GGO obtained the highest NPV of 0.948 and the lowest FPR of 0.054 among the existing methods. These result shows that the WOA-GGO

optimization-based feature selection method effectively identifies the features while minimizing the false positive predictions. Overall, the WOA-GGO performed better than the conventional optimization algorithms on both datasets, showing its effectiveness and reliability for the feature selection.

Table 20: Comparative Analysis of Feature Selection-Based Optimization for Dataset 1

Metrics	GGO	WOA	COA	POA	KOA	JFO	WOA-GGO
Accuracy	0.941	0.933	0.903	0.926	0.934	0.880	0.955
Sensitivity	0.935	0.927	0.897	0.920	0.927	0.875	0.951
Specificity	0.946	0.938	0.908	0.931	0.938	0.884	0.959
Precision	0.933	0.923	0.888	0.915	0.924	0.859	0.948
F-measure	0.934	0.925	0.892	0.917	0.925	0.867	0.950
MCC	0.881	0.865	0.805	0.850	0.865	0.758	0.909
NPV	0.948	0.942	0.916	0.935	0.942	0.898	0.961
FPR	0.054	0.062	0.092	0.069	0.062	0.116	0.041
FNR	0.065	0.073	0.103	0.080	0.073	0.125	0.049

Table 21: Comparative Analysis of Feature Selection-Based Optimization for Dataset 2

Metrics	GGO	WOA	COA	POA	KOA	JFO	WOA-GGO
Accuracy	0.941	0.933	0.903	0.926	0.934	0.880	0.955
Sensitivity	0.935	0.927	0.897	0.920	0.927	0.875	0.951
Specificity	0.946	0.938	0.908	0.931	0.938	0.884	0.959

Precision	0.933	0.923	0.888	0.915	0.924	0.859	0.948
Fmeasure	0.934	0.925	0.892	0.917	0.925	0.867	0.950
MCC	0.881	0.865	0.805	0.850	0.865	0.758	0.909
NPV	0.948	0.942	0.916	0.935	0.942	0.898	0.961
FPR	0.054	0.062	0.092	0.069	0.062	0.116	0.041
FNR	0.065	0.073	0.103	0.080	0.073	0.125	0.049

4.24 Discussion

The experimental findings validate the superiority of the proposed RNN + Bi-LSTM hybrid signature verification framework. Across both GPDS and CEDAR datasets, the model consistently achieved top-tier performance: accuracy of 0.955 (dataset 1) and 0.961 (dataset 2), precision of 0.948 and 0.958, sensitivity of 0.951 and 0.964, specificity of 0.959 and 0.957, respectively. The model's high capacity to accurately validate authentic signatures and reject forgeries is reflected in these metrics. The false positive rate (FPR) and false negative rate (FNR) were minimized to 0.041 and 0.049 on dataset 1, and 0.043 and 0.036 on dataset 2, highlighting the model's low error tendency. The RNN + Bi-LSTM model had the lowest accuracy standard deviation (0.014 for dataset 1 and 0.016 for dataset 2), showing strong consistency, according to statistical analysis, which validated the approach's stability. The mean and median accuracy also remained highest among all tested methods (0.953 and 0.953 for dataset 1; 0.951 and 0.952 for dataset 2). These findings confirm that the proposed hybrid framework not only surpasses existing methods such as hybrid ViT [29], HTCSigNet [30], hybrid deep learning [7] and tSSN [14], but also provides a generalizable and reliable solution for signature verification tasks, capable of handling variability in signature style, forgery type and image quality.

5. Conclusion

This research presented a deep hybrid signature verification framework consisting of four stages: preprocessing, feature extraction, feature selection and verification. Improved median filtering and image resizing were used during preprocessing to enhance the image quality. ResNet, global and local features were extracted from the preprocessed signature verification images, while a hybrid RNN and Bi-LSTM model was for the verification process. The Walrus Optimization Algorithm integrated with Graylag Goose Optimization (WOA-GGO) was utilized for feature selection and an improved score-level fusion technique was applied to combine intermediate scores and improve verification accuracy. Experimental results showed the effectiveness of the proposed framework, achieving an accuracy of 0.950 at a training rate of 70 % on Dataset 1, achieving higher performance than the conventional methods such as hybrid ViT [29], HTCSigNet [30], hybrid deep learning [7], tSSN [14], Linknet, LSTM, EfficientNet, Bi-GRU, LeNet and Bi-LSTM. However, the framework showed limitations in cross-domain generalization, model interpretability and robustness against spoofing attacks. The integration of the framework with paper-based verification systems also required reliable digital conversion, effective preprocessing and appropriate computational resources. Future work will focus on integrating advanced deep learning architectures, improving secure biometric data handling and developing real-time signature verification systems with enhanced efficiency and reliability for high-security applications.

Data Availability Statement:

The datasets generated and/or analysed during the current study are available in the GPDS 1-150 Dataset repository, <https://www.kaggle.com/datasets/adeelajmal/gpds-1150> and CEDAR-Dataset <https://www.kaggle.com/datasets/shreelakshmigp/cedardataset>.

Declarations:

Conflict of Interest: The authors declare that they have no conflict of interest.

Funding: No funding was received for conducting this study or for the preparation of this manuscript.

Clinical Trial Registration Requirement: We confirm that this study is not a clinical trial and does not involve the registration of a clinical trial. Therefore, clinical trial registration details, including the registry, trial registration number, and date of registration, are not applicable to this manuscript.

Ethics, Consent to Participate, and Consent to Publish Declarations: We confirm that the study does not involve human participants, human subjects, or identifiable personal data. Therefore, ethics approval, consent to participate, and consent to publish are not applicable to this study.

References

- [1] Jain A, Singh SK, Singh KP. Signature verification using geometrical features and artificial neural network classifier. *Neural Computing and Applications*. 2021;33(12):6999-7010. <https://doi.org/10.48550/arXiv.2108.02029>.
- [2] Bhanja S, Das A. Impact of data normalization on deep neural network for time series forecasting. *arXiv preprint arXiv:1812.05519*. 2018. 10.48550/arXiv.1812.05519.
- [3] Jain A, Singh SK, Pratap Singh K. Multi-task learning using GNet features and SVM classifier for signature identification. *IET Biometrics*. 2021;10(2):117-26. <https://doi.org/10.1049/bme2.12007>.
- [4] Zheng Y, Iwana BK, Malik MI, Ahmed S, Ohyama W, Uchida S. Learning the micro deformations by max-pooling for offline signature verification. *Pattern Recognition*. 2021;118:108008. <https://doi.org/10.1016/j.patcog.2021.108008>.
- [5] Yapıcı MM, Tekerek A, Topaloğlu N. Deep learning-based data augmentation method and signature verification system for offline handwritten signature. *Pattern Analysis and*

Applications. 2021;24(1):165-79. <https://link.springer.com/article/10.1007/s10044-020-00912-6>.

- [6] Sarwinda D, Paradisa RH, Bustamam A, Anggia P. Deep learning in image classification using residual network (ResNet) variants for detection of colorectal cancer. *Procedia Computer Science*. 2021;179:423-31. <https://doi.org/10.1016/j.procs.2021.01.025>.
- [7] Alsuhiat, Fadi Mohammad and Fatma Susilawati Mohamad. "A hybrid method of feature extraction for signatures verification using CNN and HOG a multi-classification approach." *IEEE Access* 11 (2023): 21873-21882.
- [8] Okawa M. Online signature verification using single-template matching with time-series averaging and gradient boosting. *Pattern Recognition*. 2020;102:107227. <https://doi.org/10.1016/j.patcog.2020.107227>.
- [9] Parcham E, Ilbeygi M, Amini M. CBCapsNet: A novel writer-independent offline signature verification model using a CNN-based architecture and capsule neural networks. *Expert Systems with Applications*. 2021;185:115649. <https://doi.org/10.1016/j.eswa.2021.115649>.
- [10] Li H, Li H, Zhang H, Yuan W. Black-box attack against handwritten signature verification with region-restricted adversarial perturbations. *Pattern Recognition*. 2021;111:107689. <https://doi.org/10.1016/j.patcog.2020.107689>.
- [11] Tolosana R, Vera-Rodriguez R, Gonzalez-Garcia C, Fierrez J, Morales A, Ortega-Garcia J, Ruiz-Garcia JC, Romero-Tapiador S, Rengifo S, Caruana M, Jiang J. SVC-onGoing: Signature verification competition. *Pattern Recognition*. 2022;127:108609. <https://doi.org/10.1016/j.patcog.2022.108609>.

- [12] Banerjee D, Chatterjee B, Bhowal P, Bhattacharyya T, Malakar S, Sarkar R. A new wrapper feature selection method for language-invariant offline signature verification. *Expert Systems with Applications*. 2021;186:115756. <https://doi.org/10.1016/j.eswa.2021.115756>.
- [13] Başarslan MS, Kayaalp F. MBI-GRUMCONV: A novel Multi Bi-GRU and Multi CNN-Based deep learning model for social media sentiment analysis. *Journal of Cloud Computing*. 2023;12(1):5. <https://doi.org/10.1186/s13677-022-00386-3>.
- [14] Tehsin, Sara, Ali Hassan, Farhan Riaz, Inzamam Mashood Nasir, Norma Latif Fitriyani, and Muhammad Syafrudin. "Enhancing signature verification using triplet siamese similarity networks in digital documents." *Mathematics* 12, no. 17 (2024): 2757.
- [15] Zheng X, Chen W. An attention-based bi-LSTM method for visual object classification via EEG. *Biomedical Signal Processing and Control*. 2021;63:102174. <https://doi.org/10.1016/j.bspc.2020.102174>.
- [16] Liu L, Huang L, Yin F, Chen Y. Offline signature verification using a region based deep metric learning network. *Pattern Recognition*. 2021;118:108009. <https://doi.org/10.1016/j.patcog.2021.108009>.
- [17] Kalaivani K, Asnath Vicky Phamila Y. Modified Wiener Filter for Restoring Landsat Images in Remote Sensing Applications. *Pertanika Journal of Science & Technology*. 2018;26(3). http://dx.doi.org/10.1007/978-981-33-4102-9_33.
- [18] Wei Z, Yang S, Xie Y, Li F, Zhao B. SVSV: Online handwritten signature verification based on sound and vibration. *Information Sciences*. 2021;572:109-25. <https://doi.org/10.1016/j.ins.2021.04.099>.

- [19] Ponce-Hernandez W, Blanco-Gonzalo R, Liu-Jimenez J, Sanchez-Reillo R. Fuzzy vault scheme based on fixed-length templates applied to dynamic signature verification. *IEEE Access*. 2020;8:11152-64. <https://doi.org/10.1109/ACCESS.2020.2965165>.
- [20] Ren Y, Wang C, Chen Y, Chuah MC, Yang J. Signature verification using critical segments for securing mobile transactions. *IEEE Transactions on Mobile Computing*. 2019;19(3):724-39. <https://doi.org/10.1109/TMC.2019.2897657>.
- [21] Vishi K, Mavroeidis V. An evaluation of score level fusion approaches for fingerprint and finger-vein biometrics. *arXiv preprint arXiv:1805.10666*. 2018. <https://doi.org/10.48550/arXiv.1805.10666>.
- [22] Hafemann LG, Sabourin R, Oliveira LS. Meta-learning for fast classifier adaptation to new users of signature verification systems. *IEEE Transactions on Information Forensics and Security*. 2019;15:1735-45. <https://doi.org/10.1109/TIFS.2019.2949425>.
- [23] Tolosana R, Vera-Rodriguez R, Fierrez J, Ortega-Garcia J. DeepSign: Deep on-line signature verification. *IEEE Transactions on Biometrics, Behavior, and Identity Science*. 2021;3(2):229-39. <https://doi.org/10.1109/TBIOM.2021.3054533>.
- [24] Souza VL, Oliveira AL, Cruz RM, Sabourin R. A white-box analysis on the writer-independent dichotomy transformation applied to offline handwritten signature verification. *Expert Systems with Applications*. 2020;154:113397. <https://doi.org/10.1016/j.eswa.2020.113397>.
- [25] Malik J, Elhayek A, Guha S, Ahmed S, Gillani A, Stricker D. DeepAirSig: End-to-end deep learning based in-air signature verification. *IEEE Access*. 2020;8:195832-43. <https://doi.org/10.1109/ACCESS.2020.3033848>.

- [26] Başarslan MS, Kayaalp F. MBI-GRUMCONV: A novel Multi Bi-GRU and Multi CNN-Based deep learning model for social media sentiment analysis. *Journal of Cloud Computing*. 2023;12(1):5. <https://doi.org/10.1186/s13677-022-00386-3>.
- [27] Zhou Y, Zheng J, Hu H, Wang Y. Handwritten signature verification method based on improved combined features. *Applied Sciences*. 2021;11(13):5867. <https://doi.org/10.3390/app11135867>.
- [28] Rokade, S., Singh, S. K., Bansod, S., & Pal, P. (2023, January). An Offline Signature Verification Using Deep Convolutional Neural Networks. In *2023 Third International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)* (pp. 1-4). IEEE.
- [29] Alajrami, Eman, Belal AM Ashqar, Bassem S. Abu-Nasser, Ahmed J. Khalil, Musleh M. Musleh, Alaa M. Barhoom, and Samy S. Abu-Naser. "Handwritten signature verification using deep learning." (2020).
- [30] Ishfaq M, Saadia A, Alserhani FM, Gul A. Enhancing Security: Infused Hybrid Vision Transformer for Signature Verification. *IEEE Access*. 2024.
- [31] Zheng L, Wu D, Xu S, Zheng Y. HTCSigNet: A Hybrid Transformer and Convolution Signature Network for offline signature verification. *Pattern Recognition*. 2025;159:111146.
- [32] Diaz M, Fischer A, Ferrer MA, Plamondon R. Dynamic signature verification system based on one real signature. *IEEE transactions on cybernetics*. 2016;48(1):228-39. <https://doi.org/10.1109/TCYB.2016.2630419>.
- [33] Lu J, Qi H, Wu X, Zhang C, Tang Q. Research on authentic signature identification method integrating dynamic and static features. *Applied Sciences*. 2022;12(19):9904. <https://doi.org/10.3390/app12199904>.

- [34] Sanchez-Reillo R, Liu-Jimenez J, Blanco-Gonzalo R. Forensic validation of biometrics using dynamic handwritten signatures. *IEEE Access*. 2018; 6:34149-57. <https://doi.org/10.1109/ACCESS.2018.2849503>.
- [35] Herbst NM, Liu CN. Automatic signature verification based on accelerometry. *IBM Journal of Research and Development*. 1977;21(3):245-53. <https://doi.org/10.1147/rd.213.0245>.
- [36] <https://www.kaggle.com/datasets/adeelajmal/gpds-1150>
- [37] <https://www.kaggle.com/datasets/shreelakshmigp/cedardataset>
- [38] El-Kenawy, El-Sayed M., Nima Khodadadi, Seyedali Mirjalili, Abdelaziz A. Abdelhamid, Marwa M. Eid and Abdelhameed Ibrahim. "Greylag goose optimization: nature-inspired optimization algorithm." *Expert Systems with Applications* 238 (2024): 122147.
- [39] Trojovský, Pavel and Mohammad Dehghani. "Walrus optimization algorithm: a new bio-inspired metaheuristic algorithm." (2022).
- [40] Zheng, Jianbin, Ziyao Chen, Liping Huang, Yifan Gao, Xiangxiang Yu, Hui Wang, Jiamei Yang and Yu Wang. "DWSCNN online signature verification algorithm based on CAE-MV feature dimensionality reduction." *IEEE Access* 12 (2024): 22144-22157.
- [41] Sanchez-Nielsen, Elena and Ismael Martín-Herrera. "Transformer-Based OCR System for Handwritten Signature Verification in Lawmaking Petitions: A Proof of Concept." *IEEE Access* (2025).
- [42] Hao, Yanli, Dahao Xu, Weiyi Jiang and Weihao Du. "Dual-AngleNet: an interpretable lightweight model for online signature verification using pen-tip tilt dynamics with application to power system security." *Journal of Engineering and Applied Science* 73, no. 1 (2026): 67.

- [43] Diaz, Moises, Miguel A. Ferrer, Juan M. Gil, Rafael Rodriguez, Peirong Zhang and Lianwen Jin. "Online Signature Verification based on the Lagrange formulation with 2D and 3D robotic models." *Pattern Recognition* 164 (2025): 111581.
- [44] Diaz, Moises, Miguel A. Ferrer, Jose Juan Quintana, Adam Wolniakowski, Roman Trochimczuk, Kanstantsin Miatliuk, Giovanna Castellano and Gennaro Vessio. "Neural network modelling of kinematic and dynamic features for signature verification." *Pattern Recognition Letters* 187 (2025): 130-136.
- [45] De Moura Kecia G., Robert Sabourin and Rafael MO Cruz. "ProtoSig: Enhancing training data for offline handwritten signature verification using prototypical signatures." *Pattern Recognition* (2026): 114258.