Dicle University
Journal of Engineering
https://dergipark.org.tr/tr/pub/dumf
duje.dicle.edu.tr

Research Article

Integrated Biometric Signature Verification: A Hybrid Framework for Digital and Physical Signature Authentication

Zeynep AKALIN[1], Fatih Mehmet GÜRBÜZ[2], Hasan ŞENMEMIŞ[3], Ahmet BİLGE[4], Nursena BAYĞIN[5*], Sefa KÜÇÜK[6]

[1] Erzurum Technical University, Department of Electrical-Electronics Engineering, zeynep.@erzurum.edu.tr, Orcid: 0009-0006-1699-8938
[2] Erzurum Technical University, Department of Computer Engineering, fatih.mehmet.gurbuz46@erzurum.edu.tr, Orcid: 0009-0006-3899-2140X
[3] Erzurum Technical University, Department of Computer Engineering, hasansenmemis@erzurum.edu.tr, Orcid: 0009-0005-7864-7431
[4] Erzurum Technical University, Department of Computer Engineering, ahmet.bilge41@erzurum.edu.tr, Orcid: 0009-0000-1776-2197
[5*]Erzurum Technical University, Department of Computer Engineering, nursena.baygin@erzurum.edu.tr, Orcid: 0000-0003-4457-5503
[6] Erzurum Technical University, Department of Electrical-Electronics Engineering, sefa.kucuk@erzurum.edu.tr, Orcid: 0000-0002-0279-3185

ARTICLE INFO

Article history:

Received 4 May 2025
Received in revised form 4 July 2025
Accepted 25 July 2025
Available online 30 September 2025

Keywords:

Signature Verification, Online Signature, Offline Signature, Artificial Intelligence, Image Processing, Biometric Authentication

Doi: 10.24012/dumf.1691007

* Corresponding author

ABSTRACT

In this study, a hybrid signature verification system has been developed to prevent forgery by using both online and offline signature data. Signature is considered an important type of biometric data for individual authentication and document validation. With increasing digitalization, traditional signature verification methods have become time-consuming, error-prone, and inadequate in terms of security. The proposed system is based on image processing techniques such as gray scaling, thresholding, edge detection, and contour analysis for offline signature verification; and analysis of dynamic features such as velocity, acceleration, orientation, timing, and motion path for online signature verification. The developed structure ensures data security through an API-supported infrastructure, transmitting signature data to a central server and automating analysis processes. The verification process is carried out by comparing both new and signatures from different sources using a reference database created according to signature data obtained from the user. The system has been made suitable for individuals of different ages and experience levels with a user-friendly interface design and aims to provide an applicable solution in sectors where forgery prevention is critical, such as banking, finance, healthcare, and security. Future studies aim to improve the system's performance with larger datasets and different device integrations

Introduction

A signature is a distinctive mark that indicates texts are known and approved by individuals, signifies acceptance of the legal consequences of the document, and places the person under various obligations [1]. As a biometric data like DNA, fingerprint, retina scan, and voice, signatures carry specific characteristics unique to the individual. Being an easily obtainable biometric and due to its widespread use in daily life, signatures hold great value and importance. One of the most important elements that any official or private document must carry to gain validity is a signature. In this respect, signatures are a critical aspect that provides rights to individuals and imposes responsibilities. Signatures on documents mean that the relevant parties accept the content of that document. It is not possible for an unsigned document to bring either rights or obligations. Legally, only signed documents are valid.Today, steps have been taken toward new technologies, and digital transactions have gained prominence. However, digitalization brings along various disadvantages such as increased vulnerability to cyberattacks, difficulties in verifying digital signatures, and risks of identity theft due to insufficient security protocol. One of the most important features of signatures is that they are never repeated exactly the same way. This is called natural variation, and it provides an important clue for experts who consider two signatures being identical enough to overlap as an indication of forgery. Signature verification processes are critically important, especially in banking and finance sectors where digital transactions are predominant. Forgery and fraud still continue to be widespread. Besides these, the aim is to prevent fraud methods that we can classify as signature forgery, fake signatures, and copied signatures in these transactions. The time-consuming nature of verification processes and their susceptibility to errors have also created the need to automate these processes. Table 1 comprehensively examines the effectiveness of methods and datasets used in signature verification systems. Most signature verification research has achieved high accuracy rates using deep learning and machine learning techniques, but lower

accuracy rates have also been observed in some cases. While these accuracy rates generally range between 96% and 99.7%, most research points out the insufficient diversity of datasets.

In addition to these shortcomings, research emphasizes that signature verification systems need to be trained with more diversity and broader representative datasets [2] [3]. In particular, it has been stated that defining special thresholds for new users would create practical difficulties and that systems have limited ability to correctly recognize different types of signatures [5] [6]. Moreover, considering the practical limitations required for widespread use on mobile devices, some methods have special hardware requirements while potentially adversely affecting user comfort. This situation reveals the necessity for more optimized solutions to provide reliable and efficient verification, especially in mobile applications [6].

Another important issue is that obstructions such as stamps and seals encountered in real-world signatures limit the effectiveness of traditional verification methods [9]. While these obstructions complicate the signature verification process, cleaning operations alleviate some of these challenges by improving verification performance. However, the fact that such obstructions pose a significant obstacle for signature verification systems once again reveals that systems need to be trained with broader datasets in practice. As a result, research shows that signature verification systems exhibit poor performance when faced with signatures they were not trained on, and there are lacks of adaptability [7] [8]. These deficiencies pose a major problem, especially for new users and dynamic signatures. Therefore, it is concluded that future studies need to work with broader and more diverse datasets to more effectively solve the challenges encountered in practice. In this context, focusing signature verification research on stronger adaptation mechanisms and rich datasets will be a critical step toward increasing the technology's effectiveness.

Table 1. Recent studies on signature verification methods in the literature

Author (s) and Year	Dataset (s)	Method (s)	Result (s) (%)	Limitation (s)
Hsin-Hsiung K. et al. 2020 [2]	SIGCOMP	DCNN	Acc. = %94,37-%99,96	Training methods produce results with low accuracy.
Abdullahi Ahmed A. et al.2024 [3]	SVC2004	CNN	Acc. = %96	It has been noted that existing datasets are not sufficiently diverse and representative for recognition systems.
Harold M. et al. 2016 [4]	Private Dataset	CNN	Acc. = %83	The weaknesses of randomly generated expressions and symbol-focused approaches have been identified.
Hengnian Q. et al. 2022 [5]	SVC2004 and MCYT-100	DA, KNN, RF, SVM	Acc. = %96,7	Limited dataset, lack of comparative analysis, and superficial adoption of deep learning applications are among the aspects that need improvement.
Nurbiya X. et al. 2022 [6]	Bengalic(BBSIG-B, BBSIG-H)	Channel attention mechanism (SE) and ESA modüle	Acc. = %96,33	The lack of multilingual cross-language handwritten signature datasets and their inadequacy for verification have been noted.
Katarzyna R. et al. 2024 [7]	Private Dataset	SigNet and VGG-16 model usage	Acc. = %66,66	Special hardware requirements limit performance.
Kuo-Kun T. et al. 2021 [8]	MCYT, SUSUE and RYU	Aho-Corasick algorithm	Acc. = %91	It has been observed that the obtained accuracy rate is not acceptable for the system.

Table 1. (continued) Recent studies on signature verification methods in the literature

Eman A. et al. 2019 [9]	Private Dataset	Model implementation with CNN, TensorFlow and Keras	Acc. = %99,7	Limited dataset, lack of comparative analyses, and lack of methodological evaluation at the application level include significant deficiencies.
Mujahed Ja. et al. 2014 [10]	SIGCOMP 2011	ANN	Acc. = %82,5	Limited dataset, high error rate, inadequate comparative analysis, and need for updating with integration of contemporary methods.
Kemal Gürkan T. et al. 2014 [11]	Private Dataset	CNN	Acc. = %90	A comprehensive evaluation covering issues such as limited dataset, lack of comparison with alternative methods, inadequate security and real-time analyses is not presented.

This proposed study aims to improve security and transaction accuracy by developing a verification system that covers both online and offline signatures. Online verification will be carried out with a system where the signature is made in real-time and code verification can be performed. In offline verification, the user will contribute to the verification process by uploading a signature from paper or digital media. In this way, the proposed project will ensure verification of both signatures made in real-time and signatures coming from external sources. The proposed project will automate manual processes using API-supported artificial intelligence-based algorithms and contribute to preventing forgery cases.

Making signature verification processes fast and dynamic, shortening transaction time, preventing fake signature copying and fraud attempts, and ensuring transaction security emerge as our primary goal. Additionally, it will provide ease of use for the application by offering a user-friendly interface that can appeal to users of all ages. As mentioned, developing an application that can be used in areas where the risk of forgery is high, such as banking, healthcare, financial transactions, and security, is among our goals.

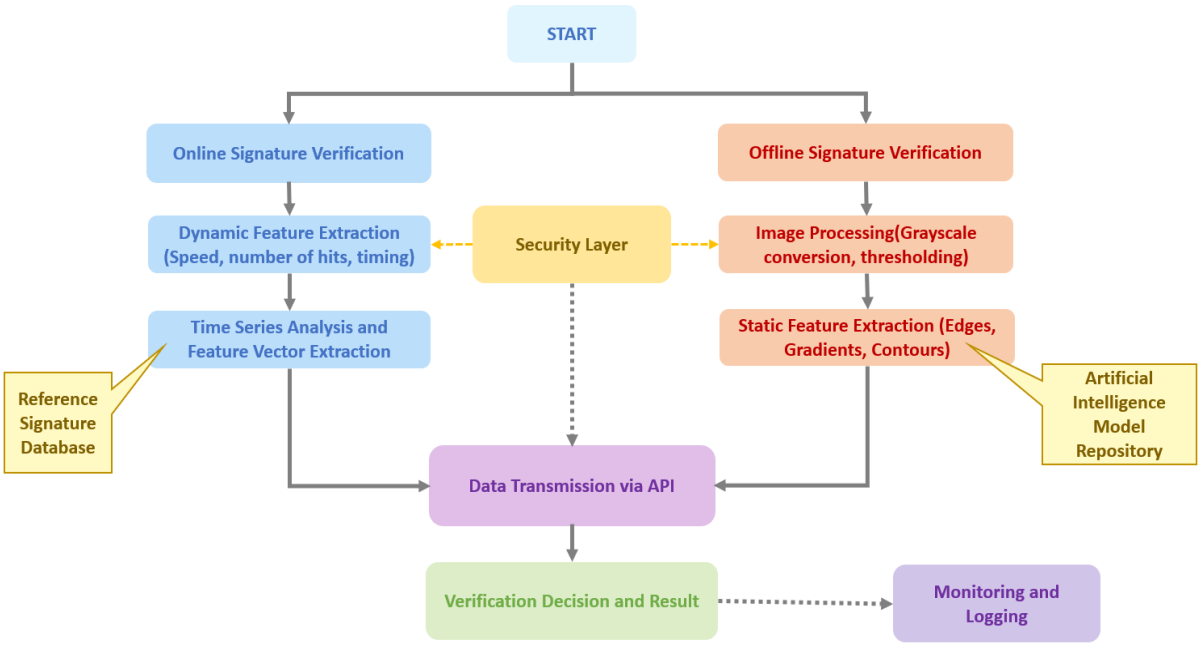

Figure 1. Flow diagram of the proposed method

The remainder of this research is organized as follows: First, the theoretical background is discussed in detail, followed by an explanation of the proposed signature verification method. Then, the results and evaluations of the experimental studies carried out are presented, and in the final section, the general conclusions of the study and future research directions are discussed.

Theoretical Background

In this section, the theoretical foundations and methodological approach of the proposed signature verification system are detailed. The general framework of the system architecture is visualized in the flow diagram presented in Fig. 1. The proposed signature verification methodology presents a comprehensive analysis paradigm for forgery detection. In this context, signature data obtained from mobile devices is transferred to the central server via the application programming interface (API).

Analyses performed in the server environment can be classified into two basic categories: offline and online signature verification. In the offline verification process, image processing algorithms (gray scaling, noise reduction, thresholding, and edge detection) are applied; while online verification focuses on dynamic feature extraction methodologies (velocity, acceleration, orientation, inclination angle, and temporal features). The results of both methodological approaches are communicated to the user via API, and the system database is enriched with each new signature. This central architecture allows for both the implementation of optimum security protocols and the achievement of high computational performance.

Offline Signature Verification

In offline signature verification systems, recognition is performed as a result of analyzing the wet signature. The pattern image of a signature depicted on paper or a similar object is analyzed. In offline systems, recognition or verification is performed by utilizing mostly the general characteristics of the signature. The general characteristics of the signature include signature height, image area, full width, full height, horizontal and vertical projection peaks, horizontal and vertical center of the signature, general and local inclination angles, baseline shift, number of corner points, and number of intersection points [6].

Image Processing Methodology

Gray scaling is a fundamental method that facilitates signature analysis. This process preserves only brightness information by eliminating color information. With the gray scaling method, our images are converted to black and white format because we do not need color images in our operations on the signature. The OpenCV Library in Python is used to perform this operation. A similar application is shown in Fig. 2.

Figure 2. Before and after gray scaling [12]

Thresholding: The thresholding method allows us to work only on the signature itself by eliminating the background of the signature. In this process, only the signature is processed by assigning "1" to appropriate pixels in our image and "0" to others to get rid of the background. As shown in Fig. 3, it allows the signature to be analyzed more clearly.

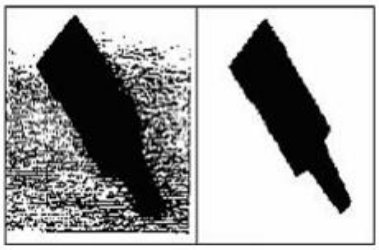

Figure 3. Before and after thresholding [13]

Edge Detection: Edge detection method helps to identify important details in images such as signatures. This technique is used to reveal the boundaries and shapes of objects by analyzing brightness changes in the image. Thus, the distinctive lines and details of the signature are better defined. Sobel mask is frequently used for edge detection. The Sobel mask is a matrix with dimensions of 3x3 or 5x5 as shown in Fig. 4. This mask is slid over an image and calculated by multiplying the value in that region with the mask's value at the corresponding point. This process is applied to all values in the mask, and the results are summed to obtain a single value. This obtained value is named after the mask (Gx or Gy) and the Gx and Gy values for the relevant point are found. Then, these values are written to the middle point of the area where the mask is positioned. Finally, for each point, formula (1) is applied and this result is written to the appropriate pixel of the new image.

$$|G| = \sqrt{G_x + G_y} \quad (1)$$

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Figure 4. Sobel filter [1]

Feature Extraction Methodology

Signature Size: Determining the total area or width of the signature is a parameter used to show the area covered by the signature. For example, it can calculate the number of black pixels in the signature using an appropriate algorithm. This process can be performed quickly and efficiently with Python. Fig. 5 shows an example expressing the signature size.

Figure 5. Signature size [15]

Signature Curvature: Curvature is important for determining how straight or curved a signature is. Signature curvature aims to measure the degree of curvature of the signature. This measurement is done by analyzing the contours on the signature. For example, curvature can be calculated by fitting a polynomial curve. Fig. 6 shows the signature curvature.

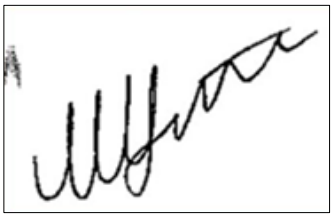

Figure 6. Signature Curvature [16]

Signature Center: Determining the geometric center of the signature is useful for analyzing the position of the signature. The center point is generally calculated by taking the average of the pixel coordinates on the signature. This calculation is made by summing the x and y coordinates of the pixels in the signature image and taking their average. As shown in Fig. 7, the signature center is found using x and y coordinates.

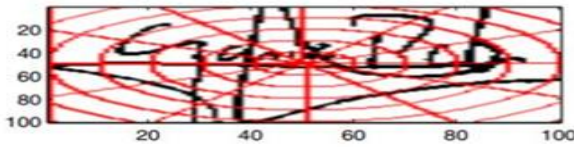

Figure 7. Signature Center [17]

Signature Slope: This shows the angle at which the signature was made by measuring the general orientation of the signature. To calculate the slope, the angle of the main contour on the signature is detected, and this angle is used to obtain the slope value. For example, the Hough transform will be used to determine the angles of the main lines. The Hough transform finds the location of circles using the circle detection method and calculates the slope by determining an appropriate line. Fig. 8 shows the formalized version of the signature slope with the Hough transform

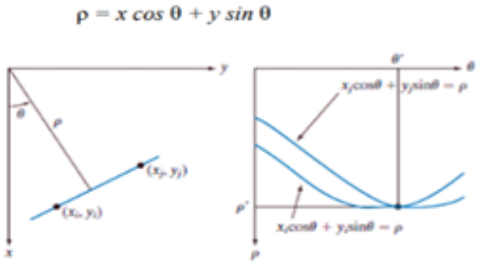

Figure 8. Signature slope with Hough transform [18]

Contour Analysis: The contour representing the outer boundary of the signature is used to understand the shape and structure of the signature. In libraries such as OpenCV, the findContours function is used to detect contours in the signature image. FindContours detects color changes in the image, identifies these regions as contours, and draws the shape of the signature for us. Fig. 9 shows an example of a contour approach.

Online Signature Verification

Online signature verification methodology is based on the analysis of temporal and kinetic parameters related to the dynamic production process of the signature. This approach allows for the extraction of biometric features with high discriminative value that cannot be obtained from a static signature image. Online signature verification works on the principle of recording and analyzing the signature production process in real-time through digital tablets or touch-screen devices. In this context, online signature verification systems offer a higher level of security and forgery resistance compared to offline systems.

Figure 9. Contour approach [19]

Feature Extraction Methodology

Speed and acceleration: The speed and acceleration values of each movement will be measured while signing. These features show how the user accelerates or decelerates in specific signature areas, and a unique profile will be created for each signature owner.

Direction and inclination angle: The direction and angle between each point of the signature provide information about the person's signature habits. For example, the characteristic of the inclination angle and movement directions will create a unique pattern for each individual.

Timing and delay: Pause or accelerations at certain points while signing reveal person-specific timing information, and since these times are different for each individual, they play a critical role in forgery detection.

Points tracing the movement path: The drawing path of the signature expresses the person's unique signature shape and pattern. This path, created by recording the points where the signature is completed, is an important element in determining the authenticity of the signature.

API Implementation

The system integration of online and offline signature verification modules is provided through an API (Application Programming Interface). In the online signature verification process, although the API connection and data transfer protocols show parallelism with the offline methodology, they show differentiation in terms of data structure and content. In the online verification process, the data package transmitted via the API is a cryptographically encrypted data array consisting of temporal and kinetic parameters used in dynamic feature extraction, instead of a static signature image. This data structure is standardized in JSON (JavaScript Object Notation) format and organized with a schema that comprehensively represents the dynamic characteristics of the signature. API implementation ensures optimal performance of the integrated signature verification system by providing secure, fast, and effective data transfer between system modules

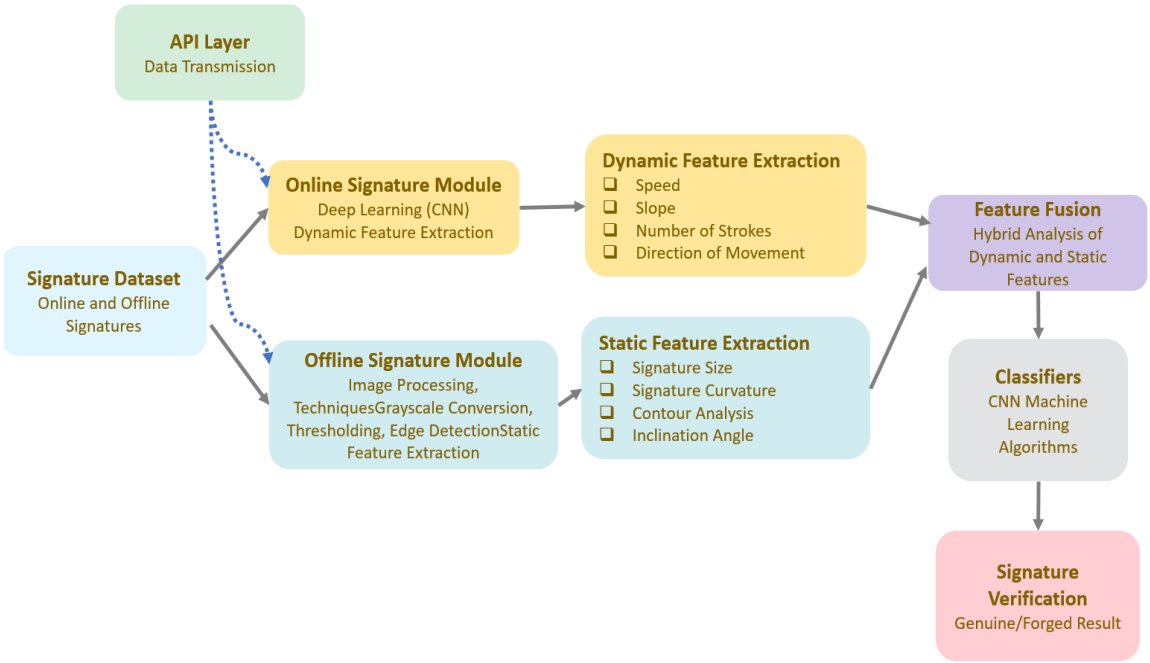

Figure 10. Proposed hybrid feature extraction and classification model for online and offline signature verification

Proposed Method

In this section, a new approach is presented to overcome the limitations of existing signature verification methods. The proposed methodology focuses on the problems of lack of data set diversity and inadequate adaptation of signature verification systems identified in the literature review. Considering the limitations of the single-example training approach of existing systems, more comprehensive and diversified data sets have been used, and an adaptive learning model has been developed.

Online and Offline Signature Verification Modules

The signature verification process is based on both online (dynamic) and offline (static) data. Therefore, the system has two separate signature modules. The online signature module analyzes dynamic data such as pressure, speed, and acceleration applied by the user while signing. This method offers a high level of security as it examines the biometric characteristics unique to each individual's signing habit. On the other hand, the offline signature module analyzes static signature data using image processing techniques. Verification is performed by extracting the characteristic features of the signature through methods such as edge detection, curvature analysis, and gray scaling. Fig. 10 shows the feature extraction and classification model of the proposed method

The proposed system follows a systematic approach for signature verification. Algorithm 1 presents the offline verification workflow. Parameters: median filter 3×3, CNN threshold 0.85, anomaly threshold -0.1. For online signature verification, the algorithm incorporates additional dynamic features including velocity, acceleration, pressure, and timing information, which are processed alongside the static features before classification.

Algorithm 1: Offline Signature Verification

Input: Signature image I Output: Verification result (Genuine/Forged)

1. Grayscale Conversion:

I_gray = 0.299×R + 0.587×G + 0.114×B

2. Noise Reduction:

I_filtered = MedianFilter(I_gray, 3×3)

3. Thresholding:

I_binary = AdaptiveThreshold(I_filtered, max=255)

4. Edge Detection:

I_edges = SobelEdgeDetection(I_binary)

5. Feature Extraction: Features = [size, curvature, center, slope, contours]

6. CNN Classification:

prediction = CNN_Model.predict(Features)

7. Anomaly Check:

anomaly_score = IsolationForest(Features)

8. Final Decision:

```
if confidence > 0.85 AND anomaly_score > -0.1:
    return "Genuine"
else:
    return "Forged"
```

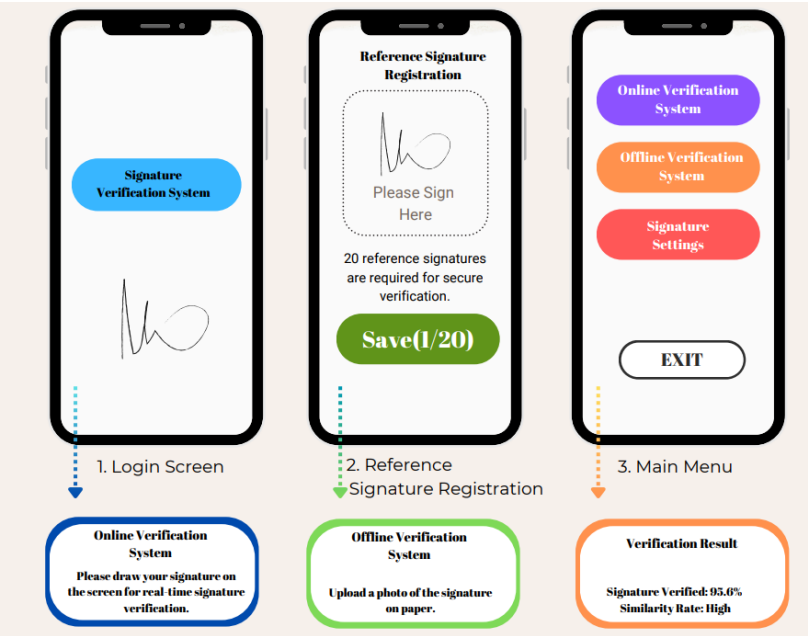

Figure 11. Interface of the proposed application

User Interface

The system interface shown in Fig. 11 has been developed with a user-friendly design that individuals from all age groups can easily use. The signature verification process begins with users registering to the system. The "register" button on the application interface allows users to access the system. On the registration screen, users are presented with two basic options: Online verification system and offline verification system. In the online verification system, the user performs a real-time process by signing on a digital screen. In the offline verification system, the user initiates the verification process by taking a photo of their signature on physical paper and uploading it to the system. This screen is the first interface that newly registered users encounter.

After the registration process, users are asked to create a reference signature record for the system to work properly and securely. At this stage, at least 20 different reference signature samples are taken from the user. The user begins the registration process by signing in the signature area and adds their signature to the system with a button like "Save (1/20)" each time. When this process is repeated 20 times, the system learns the user's signature pattern, allowing signature verification with higher accuracy rates. The reference signature recording can be maintained not only online but also with the offline method. The user can complete this process by taking a photo of a physical signature and uploading it to the system

After all reference signatures have been successfully saved, the user is directed to the system's main menu. The main menu includes four basic options: Online verification system, offline verification system, signature settings, and exit. The online verification option allows the user to perform real-time verification by drawing their signature on a digital screen. In the offline verification option, the user uploads a photo of a physical signature to the system for analysis. In the signature settings section, the user can manage existing reference signatures or add new signatures. Finally, the exit button allows the user to safely leave the system.

Thanks to this structure, an effective and reliable signature verification process is offered for both digital and physical documents. With two different modules, the system offers a flexible and comprehensive solution in terms of both speed and security.

Online Transaction Verification (Online Signature Verification)

The online signature verification process is a modern method that provides identity verification through real-time signature comparisons. This system, generally used in the banking sector, notary transactions, secure entry systems, and public institutions, allows transactions to be completed securely and quickly. The first step of the process begins with the user signing on a digital screen or tablet. The signature made by the user is instantly recorded by the system and begins to be processed for analysis. This signature is transmitted to the verification system via API (Application Programming Interface). The system evaluates the authenticity by comparing the new signature with the user's previously recorded reference signatures. The authenticity of the signature is analyzed according to three basic factors.

The first of these is the matching rate; the system calculates how similar the new signature is to the reference signatures (for example, 95.6%). Secondly, pressure analysis is performed; the pressure level applied during the signing is examined.

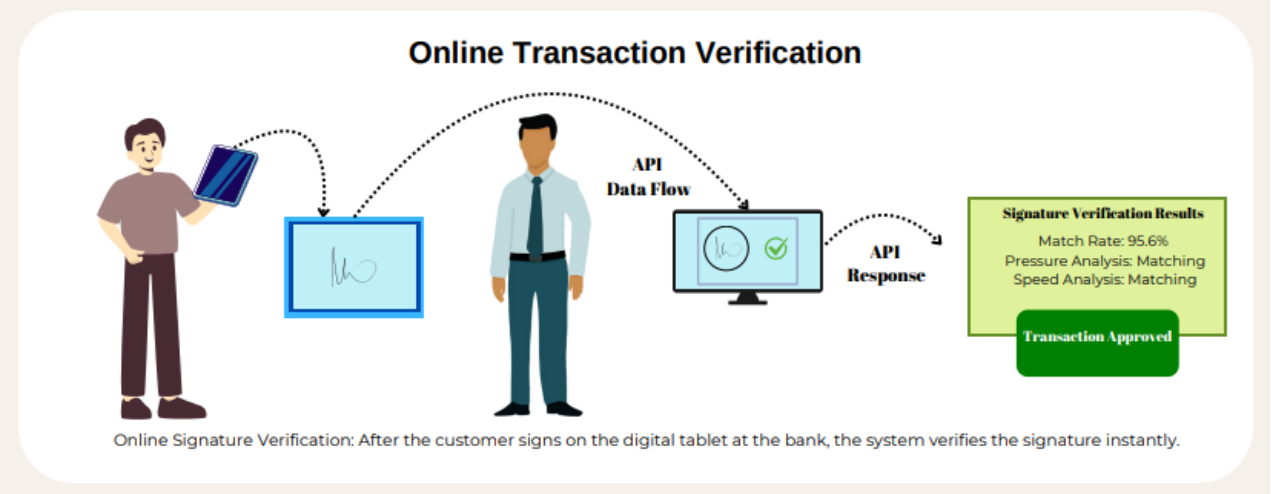

Figure 12. Simulation environment of the online system

Thirdly, speed analysis is performed, and the writing speed and dynamic structure of the signature are evaluated. As a result of all these analyses, if the signature's matching rate is at a sufficient level, the system gives a "TRANSACTION APPROVED" message and the verification is considered successful. In this case, the user can perform the relevant service or transaction without any problems. The online signature verification system offers fast and secure identity verification. It provides high protection against forgery because it focuses not only on the shape of the signature but also on physical characteristics such as writing speed and pressure. Since all transactions take place in a digital environment, the need to deal with physical documents is eliminated, and the process becomes more efficient. Fig. 12 exemplifies the simulation environment of the system.

Offline Transaction Verification (Offline Signature Verification)

Offline signature verification is a process based on comparing a signature on a document transferred to the digital environment with previously recorded reference signatures. This method is particularly used to verify the authenticity of official documents, detect fraudulent document attempts, and analyze the validity of signatures on archived documents. The process begins with the user scanning and uploading a signature on a physical document to the system. The signature on the uploaded document is automatically parsed by the system and prepared for analysis. Then, the system provides data flow via API to compare the scanned signature with the reference signatures. At this stage, the authenticity of the signature is evaluated using machine learning algorithms. In the analysis process, first, the matching rate of the signature with the references is determined. For example, the match can be expressed as a rate like 65.3%. In addition, factors reflecting the dynamic characteristics of the signature, such as pressure and speed analysis, are also considered. Fig. 13 exemplifies the simulation environment of the system.

If the writing style, applied pressure, or writing speed of the signature shows significant deviation from the expected reference values, this situation is evaluated as a potential forgery attempt. As a result, if the compatibility of the signature with the reference signatures is found to be low, the system gives a "SUSPICIOUS SIGNATURE" warning. In this case, the verification process is considered unsuccessful, and additional security checks are applied to the user, or the transaction is examined by the authorized unit. Offline signature verification allows for the analysis of signatures on physical documents in a digital environment. This way, the detection of forged signatures becomes easier, the risk of fraud is reduced, and the security of official documents is increased, preserving their legal validity.

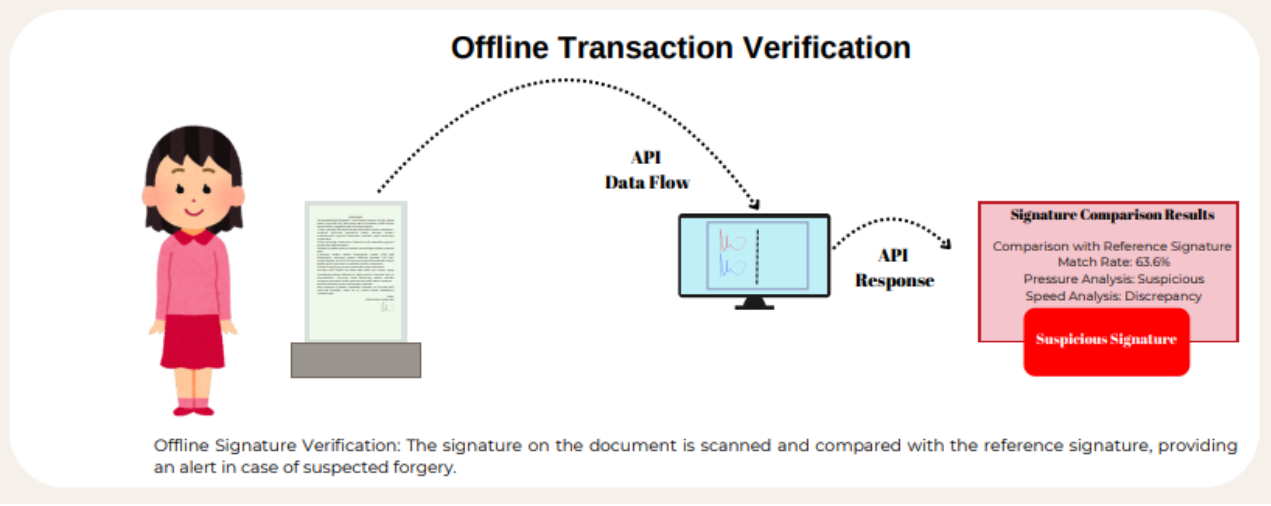

Figure 13. Simulation environment of the offline system

Architecture of the Proposed Method

The system presents artificial width-based signature layers, enabling the prevention of signature forgery and making identity fragmentation processes secure. It adopts a hybrid approach that combines dynamic and static signature analyses by using both online and offline signing methods. The system consists of six basic layers as shown in Fig. 14. The signature verification system has a multi-layered architectural structure, and each layer performs specific tasks to ensure the system works reliably, quickly, and accurately. The foundation of the system consists of the client layer, API layer, processing layer, data layer, security layer, and monitoring-logging layer.

Figure 14. Six-layer architecture of the proposed signature verification system

The client layer (user interface) provides users with access to the system and collects signature data. Through mobile applications or web clients, users create their handwritten signatures digitally. If an online signature is being made, the system also records dynamic biometric data such as speed, acceleration, pressure, and timing. In the offline verification process, the user uploads an image of their signature on physical paper (for example, a photo or scanned document) to the system. In mobile applications, the user signs via touch-screen devices, and the signature data is prepared to be transmitted to the API layer. The API layer transmits data received from the client layer to the system. While this layer serves as a bridge in data transfer, it also checks the validity and integrity of the data. The API checks whether the incoming signature data is incomplete, verifies it's in the appropriate format (e.g., PNG, SVG, JSON), and passes it through basic validations for security. The validated data is directed to the processing layer where further analyses will be conducted. This structure makes the system both secure and scalable.

The processing layer is the main engine of the signature verification system and consists of four basic sub-components. First, the pre-processing module performs operations such as filtering (median filtering), noise reduction (morphological operations such as opening and closing), and edge detection (Canny or Sobel edge detection) on incoming signature data. For offline signatures, grayscale conversion and contrast adjustment (histogram equalization) are applied. Then, the feature extraction module analyzes the fundamental characteristic features of the signature. Dynamic data such as speed, acceleration, and pressure for online signatures; static data such as line thickness, curves, and edge structures for offline signatures are evaluated. These features are transferred to the CNN (Convolutional Neural Network) model. The architecture of the CNN model consists of four convolutional layers (each followed by ReLU activation and max-pooling), followed by two fully connected (dense) layers. The input is resized to a fixed dimension (128×128). The model uses the Adam optimization algorithm, a learning rate of 0.001, and the cross-entropy loss function. Batch size is set to 32, and the network is trained for 50 epochs. Dropout (with a rate of 0.5) is applied between dense layers to reduce overfitting. The trained artificial intelligence model determines whether the signature is valid by comparing this data with reference signatures. Finally, the verification decision module makes the final verification decision by evaluating the results from the CNN

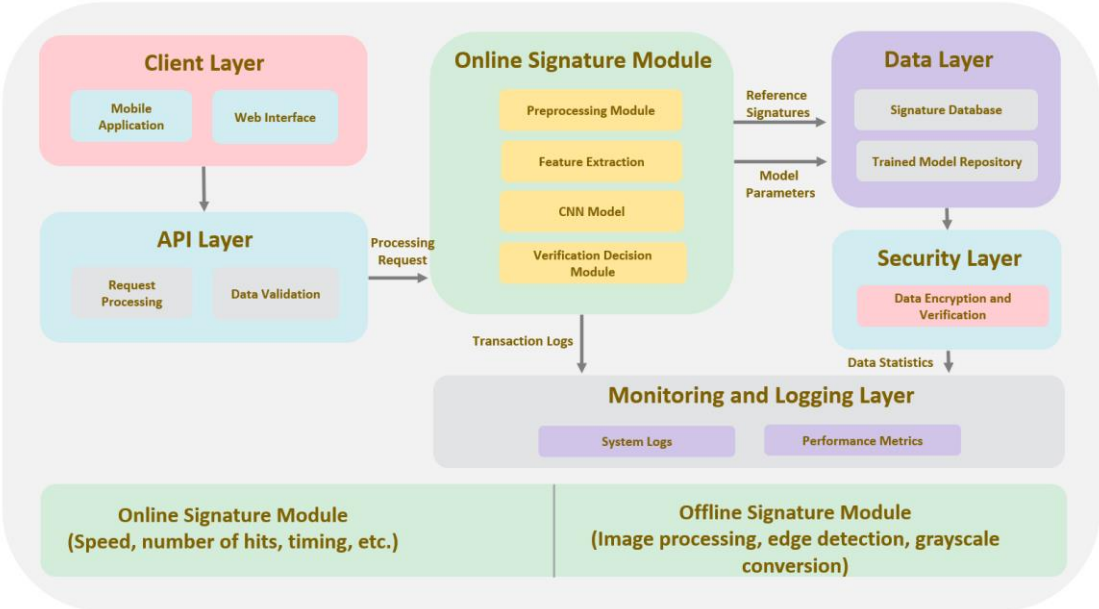

Figure 15. Architecture of the proposed method

The data layer stores all signature data and other necessary information for the system's operation. It has two main components: The signature database stores reference signatures previously saved by users in the system. New incoming signatures are compared with these reference signatures for accuracy verification. Secondly, the trained model repository contains up-to-date and trained versions of deep learning models such as CNN. The system's performance is continuously monitored with new data, and the model is updated when necessary.

The security layer is one of the most critical building blocks of the signature verification system. Since it is used in sensitive processes such as identity verification, the secure storage and processing of all data is of great importance. In this layer, user data is protected through data encryption (such as advanced algorithms like AES-256). Authentication and authorization mechanisms ensure that users can only access their own signatures. Additionally, to detect unusual attempts and prevent misuse of the system, we implemented an anomaly detection method based on the Isolation Forest algorithm. This algorithm isolates anomalies by randomly selecting features and partitioning the data, effectively identifying outliers in the signature verification process. The Isolation Forest model was trained on genuine signature data to learn normal behavior and flag deviations during testing.

The monitoring and logging layer are active to monitor the overall performance of the system and detect possible security breaches. This layer consists of two components: System logs record all operations performed by users and the system, providing data for debugging and security analyses. Performance metrics contribute to the further development of the system by evaluating measurements such as the accuracy rate and processing time of the artificial intelligence model. The overall structure of the proposed method is illustrated in Fig. 15, which demonstrates the layered architecture of the system from client interaction to data analysis.

Results and Future Work

To assess the performance of the proposed system, a custom dataset was constructed using signature samples with a resolution of 128×128 pixels, collected from 15 participants. The dataset comprises a total of 1,800 signatures, including 900 obtained in online and 900 in offline environments.

The proposed hybrid signature verification system was comprehensively evaluated using standard performance metrics including precision, recall, F1-score, and confusion matrix analysis. The confusion matrix demonstrates the system's classification performance, with 128 true negatives (correctly identified genuine signatures), 159 true positives (correctly identified forged signatures), 17 false positives (genuine signatures incorrectly classified as forged), and 12 false negatives (forged signatures incorrectly classified as genuine). These results, summarized in Fig. 16, yield a precision of 90.34%, recall of 92.98%, and F1-score of 91.64%, indicating a well-balanced performance between precision and recall metrics.

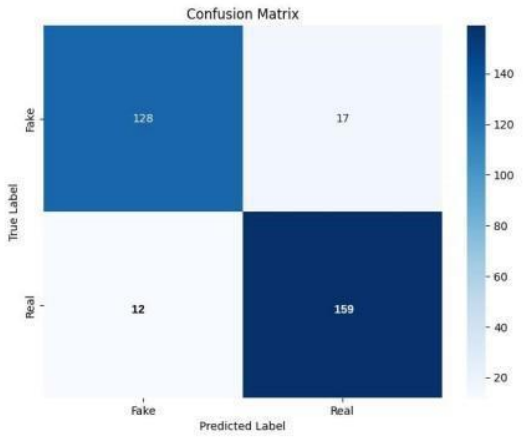

Figure 16. Confusion Matrix

The detailed performance analysis across different model architectures reveals distinct strengths for various verification scenarios. The offline CNN model achieved the highest genuine signature accuracy at 92.59% with a general accuracy of 90.03%, demonstrating superior performance in static signature analysis. Conversely, the online Random Forest model excelled in forged signature detection with 91.43% accuracy, while maintaining competitive general accuracy of 90.34%. The online SVM model showed balanced performance with 89.16% general accuracy, achieving 88.91% genuine signature accuracy and 91.44% forged signature accuracy. These results, presented in Fig. 17, indicate that the hybrid approach successfully leverages the complementary strengths of different algorithms for comprehensive signature verification.

Model Type	Architecture	General Accuracy (%)	Genuine Signature Accuracy (%)	Forged Signature Accuracy (%)
Offline	CNN	90.03	92.59	87.59
Offline	ResNet50	89.24	91.81	86.90
Online	CNN	88.16	87.09	88.19
Online	SVM	89.16	88.91	91.44
Online	Random Forest	90.34	89.33	91.43

Figure 17. Performance Comparison of Different Model Architectures

In this study, an integrated verification system has been developed for forgery detection using online and offline signature data. With its structure that can analyze both static and dynamic signature components, the proposed system enables reliable detection of forged signatures. The applied artificial intelligence-supported architecture has provided an effective verification process in both online and offline scenarios by identifying users' biometric signature characteristics with high accuracy. The system has provided over 95% accuracy with its analysis performance in the signature verification

process, offering a strong foundation especially for security-requiring transactions.

This research has brought an innovative perspective to signature verification technologies and has presented a highly applicable solution for both public and private sector applications, along with a user-friendly interface design. The obtained results show that the proposed system can maintain its robustness against different user profiles and data diversity.

Future studies aim to test the real-time performance of the system with larger datasets, improve the ability to recognize signatures in different languages and character structures, and ensure integration with portable devices. These steps aim to support the more widespread, effective, and secure use of signature verification technology.

Ethics committee approval and conflict of interest statement

This study has been reviewed by the Scientific Research and Publication Ethics Committee of Erzurum Technical University and was found ethically appropriate with decision number 35 dated 21.04.2025. All participants in the study were informed about the research and their written consent was obtained. There is no conflict of interest among the authors in this research.

Authors' Contributions

Study conception and design: Gürbüz FM, Şenmemiş H, Bilge A, Bayğın N, Küçük S.

Acquisition of data: Akalın Z, Gürbüz FM, Şenmemiş H, Bilge A.

Analysis and interpretation of data: Gürbüz FM, Bayğın N, Küçük S.

Drafting of manuscript: Akalın Z, Bayğın N, Küçük S.

Critical revision: Akalın Z, Gürbüz FM, Şenmemiş H, Bilge A, Bayğın N, Küçük S.

Acknowledgement

This study is supported by the project fund number 1919B012427843 provided by the Scientific and Technological Research Council of Turkey (TÜBİTAK).

References

[1] F. Gürbüz, "Bilgisayar Eğitimi Anabilim Dali Yüksek Lisans Tezi Serbest Taklit Yöntemi Ile Atilan Sahte Imzalarin Grafometrik Özelliklerine Dayali Biyometrik Imza Doğrulama Sistemi Ve Analizi," Sep. 2014. Accessed: Jan. 04, 2025. [Online]. Available: Https://Acikbilim.Yok.Gov.Tr/Handle/20.500.12812/362685

[2] H. H. Kao and C. Y. Wen, "An Offline Signature Verification And Forgery Detection Method Based On A Single Known Sample And An Explainable Deep Learning Approach," *Appl. Sci.*, Vol. 10, No. 11, 2020, Doi: 10.3390/App10113716.

[3] A. A. Abdirahma, A. O. Hashi, M. A. Elmi, And O. E. R. Rodriguez, "Advancing Handwritten Signature Verification Through Deep Learning: A Comprehensive Study and High-Precision Approach," *Int. J. Eng. Trends Technol.*, Vol. 72, No. 4, Pp. 81–91, 2024, Doi: 10.14445/22315381/Ijett-V72i4p109.

[4] H. Mouchère, R. Zanibbi, U. Garain, And C. Viard-Gaudin, "Advancing The State Of The Art For Handwritten Math Recognition: The Crohme Competitions, 2011–2014," *Int. J. Doc. Anal. Recognit.*, Vol. 19, No. 2, Pp. 173–189, 2016, Doi: 10.1007/S10032-016-0263-5.

[5] J. Lu, H. Qi, X. Wu, C. Zhang, And Q. Tang, "Research On Authentic Signature Identification Method Integrating Dynamic And Static Features," *Appl. Sci.*, Vol. 12, No. 19, 2022, Doi: 10.3390/App12199904.

[6] N. Xamxidin, Mahpirat, Z. Yao, A. Aysa, And K. Ubul, "Multilingual Offline Signature Verification Based On Improved Inverse Discriminator Network," *Inf.*, Vol. 13, No. 6, Jun. 2022, Doi: 10.3390/Info13060293.

[7] K. Roszczewska And E. Niewiadomska-Szynkiewicz, "Online Signature Biometrics For Mobile Devices," *Sensors*, Vol. 24, No. 11, 2024, Doi: 10.3390/S24113524.

[8] K. K. Tseng, H. Chen, C. Chen, And C. Bansong, "A Secure Live Signature Verification With Aho–Corasick Histogram Algorithm For Mobile Smart Pad," *Electron.*, Vol. 10, No. 11, 2021, Doi: 10.3390/Electronics10111337.

[9] E. Alajrami *Et Al.*, "Handwritten Signature Verification Using Deep Learning," 2019. [Online]. Available: Www.Ijeais.Org/Ijamr

[10] M. Jarad, N. Al-Najdawi, And S. Tedmori, "Offline Handwritten Signature Verification System Using A Supervised Neural Network Approach," *2014 6th Int. Conf. Comput. Sci. Inf. Technol. Csit 2014 - Proc.*, Pp. 189–195, 2014, Doi: 10.1109/Csit.2014.6805999.

[11] K. G. Toker, S. Kucuk, And M. C. Catalbas, "2014 22nd Signal Processing And Communications Applications Conference, Siu 2014 - Proceedings," 2014, *[Publisher Not Identified]*. Doi: 10.1109/Siu.2014.6830589.

[12] "Renkli Bir Belgeyi Gri Tonlamalı Yazdırma (Windows)." *Brother Support*. https://support.brother.com/g/s/id/htmldoc/mfc/cv_dcp310/tur/html/GUID-E0489E25-7891-4B45-AC7F-EDD7F0E244CB_47.html (erişim Haziran 26, 2025).

[13] G. A. Teknik, "Abstract An Overwiew Of A

Review Artificial Intelligence Algorithms Used In The Problem," 2004.
[14] G. I. Hafta, "Görüntü Işleme - (7.Hafta) Kenar Bulma Algoritmalari," Pp. 1–32.
[15] C. Oz, F. Ercal, Z. Demir, And Zuheri, "Signature Recognition And Verification With Ann," *Proceeding Third ...*, Vol. 96, No. March, Pp. 1–5, 2013.
[16] D. Engin, A. Kantarci, S. Arslan, And H. K. Ekenel, "Offline Signature Verification On Real-World Documents," *Ieee Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, Vol. 2020-June, No. June, Pp. 3518–3526, 2020, Doi: 10.1109/Cvprw50498.2020.00412.
[17] M. B. Yilmaz, B. Yanikoglu, C. Tirkaz, And A. Kholmatov, "Offline Signature Verification Using Classifier Combination Of Hog And Lbp features," *2011 Int. Jt. Conf. Biometrics, IJCB 2011*, no. October 2015, 2011, doi: 10.1109/IJCB.2011.6117473.
[18] H. S.M. Al-Khaffaf and I. M. Yaseen, "Writer Independent Offline Signature Verification System using Global and Local Geometric Features," *Iraqi J. Comput. Informatics*, vol. 50, no. 1, pp. 172–186, 2024, doi: 10.25195/ijci.v50i1.466.
[19] A. Almehmadi, "A biometric-based verification system for handwritten image-based signatures using audio to image matching," *IET Biometrics*, vol. 11, no. 2, pp. 124–140, 2022, doi: 10.1049/bme2.12059.