DOI: https://doi.org/10.33103/uot.ijccce.24.1.4

Handwritten Signature Identification Based on Hybrid Features and Machine Learning Algorithms

Zainab Hashim[1], Hanaa Mohsin[2], Ahmed Alkhayyat[3]

[1,2] *Department of Computer Sciences, University of Technology, Baghdad, Iraq*

[3] *Qaultiy Assurance Department, The Islamic University, Najaf, Iraq*

[1] *cs.20.16@grad.uotechnology.edu.iq*, [2] *110113@uotechnology.edu.iq*, [3] *ahmedalkhayyat85@gmail.com*

Abstract— *Handwritten signature identification is a process that determines an individual's true identity by analyzing their signature. This is an important task in various applications such as financial transactions, legal document verification, and biometric systems. Various techniques have been developed for signature identification, including feature-based methods and machine learning-based methods. However, verifying handwritten signatures in digital transactions and remote document authentication is still challenging. The inherent variety in people's signatures, which may occur due to factors such as mood, exhaustion, or even the writing tool used, contributes to the problem. Furthermore, the proliferation of sophisticated forgery methods, such as freehand mimicking and sophisticated picture manipulation, necessitates the development of reliable and precise tools for identifying authentic signatures from fake ones.The present paper suggests a method for identifying signatures based on integrating static (off-line) handwritten signature data. This is done by fusing three types of signature features: Linear Discriminant Analysis (LDA) as appearance-based features, Fast Fourier Transform (FFT) as frequency-features, and Gray-Level Co-occurrence Matrix (GLCM) as texture-features. Then, these fused features are inputted into four types of machine learning algorithms: Naive Bayes, K-Nearest Neighbor, Decision Tree, and AdaBoost classifiers, to identify each person and to find the most robust algorithm in terms of accuracy and precision and recall. For experiments, we have used two famous datasets: SigComp2011 and CEDAR. After training datasets, the highest accuracy achieved was 100% on the CEDAR dataset and 94.43% on the SigComp2011 dataset using a Naive Bayes classifier.*

Index Terms— *Fast Fourier Transform, Gray-Level Co-occurrence Matrix, Handwritten Signature, Linear Discriminant Analysis, Machine Learning.*

I. INTRODUCTION

The automatic recognition and verification of handwritten signatures is the subject of the field of handwritten signature identification. In many contexts, such as banking, legal documents, and official records, signatures are commonly employed as a form of personal authentication and authorisation. The integrity and security of these procedures depend on the capacity to precisely identify and validate signatures [1].

There are now two different methods of identifying signatures: offline and online. Offline handwriting recognition technologies use regular writing instruments for writing down handwriting signatures on paper, which is then pictured or scanned as an image [2]. The features that can be retrieved from offline images can be combined to provide a wide range of useful features that are distinctive enough to be noticed. Online signatures can be created

Received 18/July/2023; Accepted 03/December/2023

 ISSN (Print) 1811-9212 *ISSN (Online) 2617-3352*

DOI: https://doi.org/10.33103/uot.ijccce.24.1.4

by signing on touch-screen devices like tablets and smartphones, and many features can be gained by utilizing a specific pen and tablet together with a scanned image of the signer's signature. By gathering detailed data, such as writing speed, angle, strength utilized by writers, and stroke order online, online handwriting recognition can be accomplished [3]. The main problem behind handwritten signature identification is summarized as follows: Signatures can have a wide range of formats. Because of that, it is difficult to create an all-encompassing model that can precisely validate and categorize signatures from various people. Also, the most exclusive data gathered from handwritten signatures may not be captured by conventional feature extraction techniques. Furthermore, Conventional deep learning models might need a lot of training and struggle to adapt to changes in signature data. Our proposed work extracts meaningful signature features based on the fusion of appearance-based features, texture-based features, and frequency-based features, then various machine learning algorithms are used to classify the extracted features. The findings of this study hold practical importance for various customers, especially financial institutions, courts, and security organizations. Enhancing the feature extraction and classification methods, lowering the chance of deception, can all be accomplished with a more reliable and accurate signature identification system. The following is how the paper is set up: Section II describes the works related to this study. Our methodology is described in Section III. The study's findings are presented in Section IV. Conclusions are in Section V.

II. RELATED WORKS

Twenty years ago, a wide variety of biometric systems [4, 5, 6, 7, 8], of which signature identification systems are one have been proposed for use in confirming a person's identity. This section provides an overview of various recent studies that tackle the issue of identifying writers of signatures. **Ganapathi, G., & Nadarajan, R. (2014) [9]** proposed that combines inference rules for image augmentation, fuzzy rough reduction for feature selection, and simplified fuzzy ARTMAP for verification. In that work, they presented a writer-dependent system that achieved accuracy from 88.95% to 93.99% on the CEDAR dataset. **Zois et al. (2016) [10]** suggested a unique grid-based template-matching approach for offline signature evaluation and verification. They transformed the initial gray-level image into a binary using a standard preprocessing procedure. The feature extraction stage then applied each signature to a family of six groups of grid lattices (GoG's). A writer-dependent model was subsequently created during training using SVMs. On the CEDAR dataset, they achieved an accuracy of 96.52% in that work. **Chen et al. (2018) [11]** developed a method for investigative handwriting analysis to verify signatures. The method utilized online characteristics including the likelihood ratio (LR) and width, gray-scale, and radian paired with writing sequence information to distinguish authentic signatures from forgeries. The SigComp2011 dataset served as the foundation for comparisons of the system performances. The strategy described in this study (based on pertinent online characteristics) produced 96.17% accuracy on the Dutch and Chinese SigComp2011 dataset, demonstrating that online characteristics are more accurate than offline characteristics. **Maergner et al. (2019) [12]** recommended combining a statistical method based on deep triplet networks with a current structural technique based on graph edit distance. Performance on four standard datasets that are available to the public is significantly improved by using both the structural and statistical techniques.

Hafemann et al. (2016) [13] utilized a combined writer-independent/dependent strategy. They trained a writer-dependent classifier (also utilizing DCNN) to determine whether test signatures were forgery or not after training a writer-independent Deep Convolutional Neural Network (DCNN) to construct a feature representation of the signatures. **Yilmaz et al.(2018) [14]** devised a method akin to this.

Received 18/July/2023; Accepted 03/December/2023

DOI: https://doi.org/10.33103/uot.ijccce.24.1.4

Through the use of a Convolutional Neural Network (CNN), where signature pairs are fed into the network as a two-channel image, they were able to extract the features in a writer-independent context. A writer-independent strategy, a writer-dependent approach, and a mixture of both approaches were created, with respective EERs of 4.13%, 3.66%, and 1.76%. In this study, skilled forgeries are solely employed during tests (30 for each skilled forgery of a user). By **A. Rexit [15] (2022)** a method for recognizing handwritten signatures was suggested which utilized local maximum frequency characteristics and histogram of oriented gradients (HOG) characteristics. Principal component analysis (PCA) spatially reduced the high-dimensional characteristics produced by each of these techniques. The k-nearest neighbors (k-NN) algorithm was used for classification, and it was compared to the random forest approach. In comparison to other approaches, the suggested approach had an identification percentage of 98.4% using a wide-ranging signature dataset.

III. METHODOLOGY

We used a reliable method to identify a person based on their signature. Our approach can be divided into three phases, starting with the pre-processing of signature images, moving on to feature extraction, and concluding with a classification procedure. Our research was based on two publicly available datasets, SigComp2011 [16] and CEDAR [17].

First, we pre-process all training and test images in order to get the signatures ready for the identification stage. The images are converted to gray-scale, histogram equalized, blurred, and resized. Then, we proposed an approach based on the fusion of appearance-based features (Linear Discriminative Analysis (LDA)), texture-features (Gray-Level Co-occurrence Matrix), and frequency-features (Fast Fourier Transform) in order to build a hybrid feature vector for each image for identification. Finally, these features were classified using machine learning algorithms to identify each person. *Fig. 1* shows the proposed signature identification method.

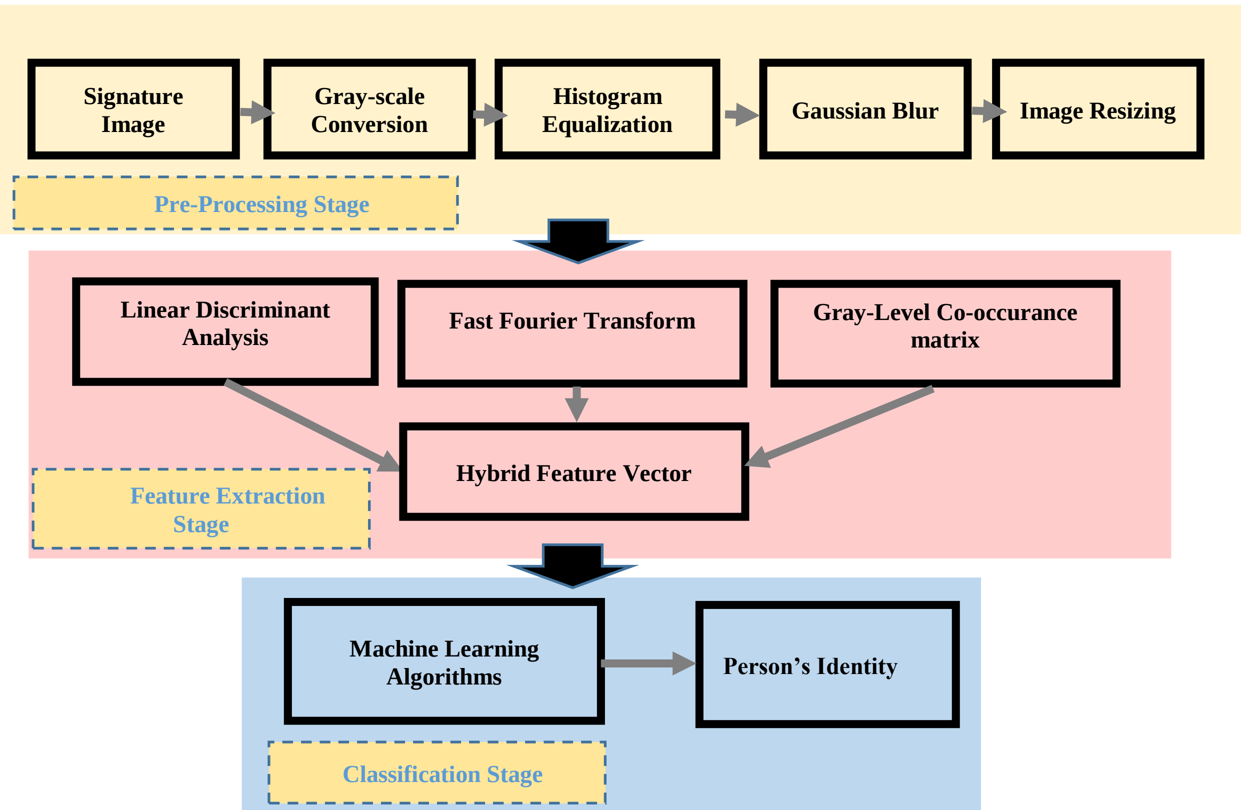


FIG. 1. THE PROPOSED SYSTEM ARCHITECTURE.

Received 18/July/2023; Accepted 03/December/2023

DOI: https://doi.org/10.33103/uot.ijccce.24.1.4

A. Pre-processing Stage

The procedure that is done before the extraction of features is known as "pre-processing". This is a critical step in the procedure of handwritten signature identification because once signatures are acquired; procedures like scanning images or additional processing could introduce issues like noise. This step's goal is to improve the appearance of the acquired images and have them ready for simple extraction of their unique characteristics. In our study, we used the following pre-processing steps:

- **Gray-level Conversion:** The primary stage in the initial processing of signature images is the conversion of the 24-bit gray-level and standard red, green, and blue formats of the signature images to 8-bit gray-scale images using the weighted approach. As demonstrated in Equation 1 below:

$$Gray = (0.21 \times R) + (0.72 \times G) + (0.07 \times B) \quad (1)$$

  Gray-level images require fewer details in order to be presented for every single pixel, so using this format will make the feature extraction process simpler and accelerate overall processing. Each gray-level brightness is recorded as an 8-bit integer with 256 possible levels of gray, which vary from black to white, as shown in Table I.

- **Signature Image Enhancement:** Sometimes signature images can be affected by poor contrast, which can be brought down due to illumination problems such as an uneven distribution of image brightness. As a result, we first processed particular areas of the signature images by Histogram Equalization (HE) this process is done after the conversion of signature images to gray-level. The main goals of Histogram Equalization are to increase the general brightness of the image and to smooth out the range of brightness. Thus, areas that lack local brightness can acquire additional brightness. This can be accomplished via HE by spreading the greatest frequency brightness levels equally [18]. When performing Histogram Equalization, a significant amount of noisy background is revealed in the signature images for the two dataset types we employed (SigComp2011 and CEDAR), as can be seen in Table I Because sensor wavelengths possess a limited dynamic range, signature images captured by digital cameras in poor lighting conditions show minimal brightness in both dark and bright areas. Secondly, we employed a Gaussian smoothing filter with dimensions of 7 * 7 to soften the Histogram Equalized image to distinguish the signature from the noisy backgrounds. This filter will reduce several details that were distinct in the original image, enhancing the resultant signature image so that the signature stands out more than the background, as shown in Table I. The Gaussian filter is a sort of visible softening that employs the weighted principle to calculate the mean of each of the pixels. This can be done by giving every pixel a weight according to its normal distribution [19], Thus Equation 2 of the Gaussian filter is defined as follows:

$$GB_p = \sum_{q \in S} G_\sigma \left||p - q|\right| I_q \quad (2)$$

  Where GB is the Gaussian function at position p, $G_\sigma \left||p - q|\right|$ represents the Gaussian function applied to the distance between the current pixel p and the neighboring pixel q, σ is the standard deviation of the Gaussian distribution and I_q is the intensity or value of the pixel at position q.

- **Image Resizing: Finally, because some of the signature images in both datasets** (SigComp2011 and CEDAR) are significantly larger or smaller than others, they were resized to 50*50 as shown in Table I. Our machine learning algorithms and some of the feature extraction algorithms we used require signature images to be the same size LDA involves computing covariance matrices and eigenvectors, and having images of the same size can make these computations more efficient. We use Bicubic interpolation as a resizing method.

Received 18/July/2023; Accepted 03/December/2023

DOI: https://doi.org/10.33103/uot.ijccce.24.1.4

The bicubic interpolation estimates the pixels in the (i, j) positions using a sampling distance of 16 adjacent pixels (4x4) as shown in Equation 3.

$$f i,j = [W_{-1}(S_Y)W_0(S_Y)W_1(S_Y)\ W_2(S_Y)] \begin{pmatrix} f_{i-1,j-1} & f_{i,j-1} & f_{i+1,j-1} & f_{i+2,j-1} \\ f_{i-1,j} & f_{i,j} & f_{i+1,j} & f_{i+2,j} \\ f_{i-1,j+1} & f_{i,j+1} & f_{i+1,j+1} & f_{i+2,j+1} \\ f_{i-1,j+2} & f_{i,j+2} & f_{i+1,j+2} & f_{i+2,j+2} \end{pmatrix} \begin{bmatrix} W_{-1} & (S_X) \\ W_0 & (S_X) \\ W_1 & (S_X) \\ W_2 & (S_X) \end{bmatrix} \quad (3)$$

Where:

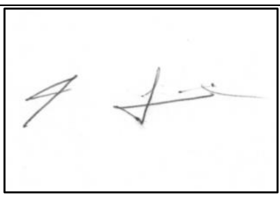
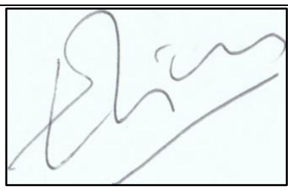
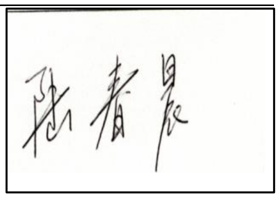
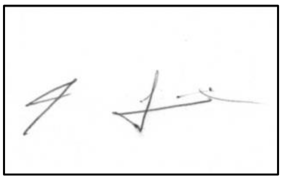
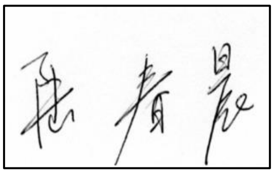
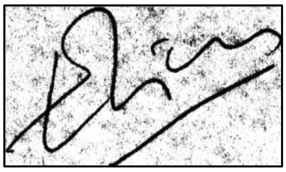
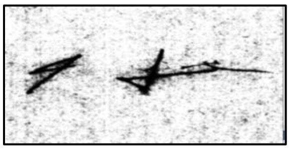
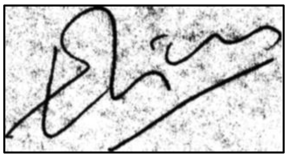
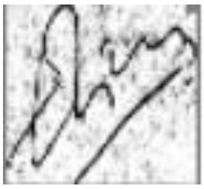
$$W-1(S_X) = \frac{-S^3+2S^2-S}{2}$$

$$W0(S_X) = \frac{3S^3-5S^2+2}{2}$$

$$W1(S_X) = \frac{-3S^3+4S^2+2}{2}$$

$$W2(S_X) = \frac{S^3-S^2}{2}$$

TABLE I. PREPROCESSING HANDWRITTEN SIGNATURE IN CEDAR, AND SIGCOMP2011 DUTCH AND CHINESE

Original Signature Images			
Gray Scale Conversion			
Histogram Equalized			
Gaussian Blurred			
Resized			

Received 18/July/2023; Accepted 03/December/2023

 ISSN (Print) 1811-9212 *ISSN (Online) 2617-3352*

DOI: https://doi.org/10.33103/uot.ijccce.24.1.4

B. Feature Extraction Stage

In the process of identifying signatures, extracting features is regarded as crucial. In our paper, a fusion of three types of features is presented to build a hybrid feature vector for each signature image. These are: Linear Discriminant Analysis (LDA) as appearance-based features, Fast Fourier Transform (FFT) as frequency-features, and Gray-Level Co-occurrence Matrix (GLCM) as texture-features as shown in *Fig. 2*. By gradually transforming the initial data set into a low-dimensional characteristic with an emphasis on the most significant characteristics, the LDA of signature images generates an effective approximation. FFT represents the signature features by reflecting their frequency spectrum energy. Through the GLCM of signature images, comprehensive information about the direction, the adjacent interval, and the change of gray scale can be obtained, which is the basis for analyzing the local pattern and arrangement rules of the image.

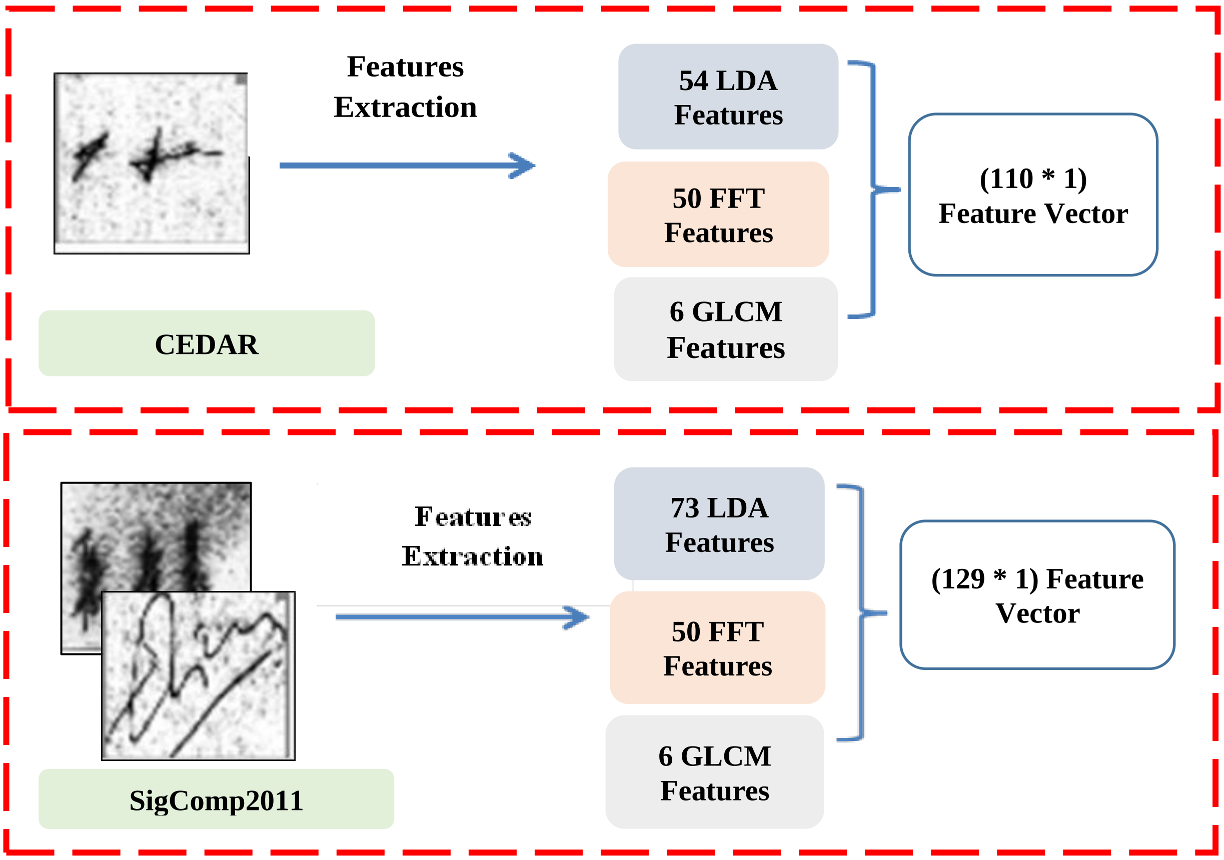


FIG. 2. THE PROPOSED FEATURE EXTRACTION STAGE.

- **Linear Discriminant Analysis (LDA)**

  R.A. Fisher created Fisher's Linear Discriminant (FLD) in 1936, which is employed for feature extraction [20]. In this paper, we apply the LDA technique to the preprocessed signature images, in order to extract signature features. LDA extracted 73 features from each image in SigComp2011 dataset, 54 features from CEDAR dataset. Because the number of non-zero eigenvalues obtained during the computation of LDA is equal to the number of classes minus one. So if we have C classes, the maximum number of meaningful LDA features you can obtain is C−1. In the case the Sigcomp2011 dataset has 74 classes so the number of LDA features is 73 and in case of CEDAR dataset the number of classes is 55 so the number of LDA features is 54. First, the inta-personal scatter matrix S_I and the between-class scatter matrix S_B is calculated as shown in Equation 4 and 5:

Received 18/July/2023; Accepted 03/December/2023

 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.24.1.4

$$S_I = \sum_{i=1}^{c} (X_i - U_{K_i})(X_i - U_{Ki})^T \quad (4)$$

$$S_B = \sum_{i=1}^{c} n_i (U_i - U)(U_i - U)^T \quad (5)$$

Where, c is the total number of samples in whole image set, X_i is the feature vector of a sample, and U_{K_i} is vector of image class that X_i belongs to. and n_i is number of samples in image class i. Then, the eigenvectors of the projection matrix W_P is calculated as shown in Equation 6:

$$E = eig(S_T^{-1}\, S_B) \quad (6)$$

Where the projection matrix is: $W_P = argmax\, \frac{|W^T S_B W|}{|W^T S_w W|}$, $S_T = S_B + S_I$

The projection matrix of each training image is contrasted with the projection matrix of the examined image using a metric for similarity. The training image, that is the most similar to the test image, is the outcome [21].

- **Gray-Level Co-occurrence Matrix (GLCM)**

We have utilized 6 GLCM features namely Contrast, Energy, Homogeneity, Entropy, Mean and Inverse. The GLCM matrix was created with 256 levels, radius=1 and in the horizontal direction. Table II shows samples of the extracted GLCM features.

1) **Contrast:** is a measurement of local differences or intensity at the grayscale level. Over the entire image, it measures the variations between the pixel point and its neighbors. According to one theory, a high-contrast image has more tones at either end of the spectrum than a low-contrast image, which has a smoother range of gray tones (black and white). Equation 7 displays the key formula utilized in contrast calculations:

$$\text{Contrast} = \sum_{i,j=0}^{N-1} \widetilde{P_{ij}}\, (i-j)^2 \quad (7)$$

Where $\tilde{P}$ is the estimated probability of the groups of pairs of surrounding gray-levels in the image and N is the overall number of gray-levels employed (the GLCM dimensions) [22].

2) **Energy:** is a metric of similarity or Angular Second Momentum (ASM), which evaluates the consistency of the textural representation, or the repeating of pixel pairs, as illustrated in Equation 8. It is in charge of identifying texture disorders. Energy can reach a maximum value of 1[23].

$$\text{Energy} = \sum_{i,j=0}^{N-1} \widetilde{(p_{ij})}^2 \quad (8)$$

3) **Entropy:** which is typically categorized as an initial measure of the level of chaos in an image, is another crucial GLCM property to distinguish an image texture. Equation 9 can be used to quickly calculate the GLCM derived entropy from the GLCM elements [24], which is inversely proportional to GLCM energy.

$$\text{Entropy} = -\sum_{i,j=0}^{N-1} P_{i\,j} \log P_{i\,j} \quad (9)$$

Received 18/July/2023; Accepted 03/December/2023

DOI: https://doi.org/10.33103/uot.ijccce.24.1.4

4) **Homogeneity:** is also called Inverse Difference Moment (IDM). Large values of the diagonal elements of the GLCM represent the large homogeneity in the visual texture. The homogeneity is greatest when the image pixel values are the same [24]. Homogeneity decreases when contrast increases but energy stays fixed in the GLCM due to the large but adverse link between the two quantities. The IDM is shown in Equation 10.

$$IDM = \sum_{i,j=0}^{N-1} \frac{\tilde{P}_{i\,j}}{1+(i-j)^2} \tag{10}$$

5) **Mean:** Compared to other GLCM textural features, it seems to be the best way to measure GLCM texture. The GLCM Mean is not simply the total of all the original values of pixels in the image window; instead, it is numerically equal to the GLCM dissimilarity, in which each pixels is valued by its frequency of appearance and specific neighboring pixels as shown in Equation 11.

$$u_i = \sum_{i,j=0}^{N-1} i\,\tilde{P}_{i\,j} \qquad u_j = \sum_{i,j=0}^{N-1} j\,\tilde{P}_{i\,j} \tag{11}$$

6) **Inverse:** the last feature we use is inverse and is shown in Equation 12.

$$Inverse = \sum_{i,j=0}^{N-1} \frac{\tilde{P}_{i\,j}}{(i-j)^2} \tag{12}$$

TABLE II. SAMPLES OF GLCM FEATURES VALUES OF THREE IMAGES

	Energy	Contrast	Homogeneity	Inverse	Entropy	Mean
Image1	0.003863	1637.6	0.699045	0.01055	2.76925	177.786
Image2	0.00596	1137.48	0.703653	0.01426	2.70134	191.692
Image3	0.003622	1103.54	0.699323	0.015479	2.78311	179.504

- **Fast Fourier Transform (FFT)**

The FFT method, considered a fast representation of the Discrete Fourier Transform (DFT), works by transforming information from the time to the frequency range. The spatial frequency of each object in the remote sensing image is unique. The shape, structure, texture, and other properties of various things can be efficiently reflected in their frequency spectrum energy [25] [26]. In this paper, we will use FFT as feature extractor to extract frequency features from signature images. First, the FFT of a 2D signature image is calculated using Equations 13 and 14.

$$\mathrm{f(x,y)} = \sum_{M=0}^{M-1} \sum_{N=0}^{N-1} f(m,n) e^{-\left(i\times x\times\pi\left(x\frac{m}{M}+\frac{n}{N}\right)\right)} \tag{13}$$

$$\mathrm{f(x,y)} = \frac{1}{M.N} \sum_{M=0}^{M-1} \sum_{N=0}^{N-1} f(m,n) e^{-\left(i\times x\times\pi\left(x\frac{m}{M}+\frac{n}{N}\right)\right)} \tag{14}$$

The pixel at location (m, n) is represented by f(m, n), while f(x, y) is the function to represent the image in the frequency domain with respect to position x and y, M x N indicates the

Received 18/July/2023; Accepted 03/December/2023

DOI: https://doi.org/10.33103/uot.ijccce.24.1.4

image's dimension, and i is sqrt (-1). Then, we apply vector quantization on spectral signature images to convert it to feature vector.

C. Classification Stage

In our work we used four types of classifiers to indicate each person identity based on its hybrid feature vector, these are Naive Bayes, K-Nearest Neighbor, Descision Tree and AdaBoost classifiers.

- **Naive Bayes**

  The Naive Bayes classifier is a straightforward probabilistic classification algorithm. It is based on the Bayes' theorem and makes a claim that every combination of features is independent [27], as illustrated in Algorithm 1.

Algorithm 1. Naive Bayes

Input: Training data $TR_{Signature}$ and Predictors $P_{Signature}$ (F_1, F_2,...F_n)
Output: Decision: Person's identity (label of test signature)
Begin
Step1: Read $TR_{Signature}$
Step2: Compute the Mean and Standard Deviation of $P_{Signatures}$ for each label
Step3: Repeat
Step4: Compute the probability of $P_{Signature}$
using this Equation $P(C|F_1, \ldots \ldots F_n) = \frac{P(C).P(F_1 \ldots\ldots F_n|C)}{P(F_1 \ldots\ldots F_n)}$ until all the probability of $P_{Signatures}$ is computed
Step5: Compute the likelihood for each label
Step6: Take the maximum likelihood
End

- **D-Nearest Neighbor**

  For categorization tasks, the automated machine learning method k-nearest neighbors (KNN) is used. KNN categorizes unlabeled data by figuring out how far apart each unlabeled datapoint is from every other point in the dataset. Then, using patterns in the dataset, assign each unlabeled datapoint to the class with the most similarly labeled data. The most popular way to calculate distance in KNN is using the Euclidean distance formula [28]. KNN steps are declared in Algorithm 2.

Algorithm 2. K-Nearest Neighbor

Input: Training Data $TR_{Signature}$, Labels $L_{Signature}$, number of nearest neighbors K =2
Output: Decision: Person's identity (label of test signature)
Begin
Classify ($TR_{Signature}$, $L_{Signature}$, $T_{Signature}$)
Step1: For each test signature $T_{Signature}$ compute distance:
d ($T_{Signature}$, $TR_{Signature}$)=$\sqrt{\sum_{i=1}^{n}\left(T_{\text{Signature}_i} - TR_{\text{Signature}_i}\right)^2}$
End for
Step2: Compute $C\left(T_{\text{Signature}_i}\right) = argmax_k \sum C(\ TR_{\text{Signature}_j}, L_{\text{Signature}_K})$
End

Received 18/July/2023; Accepted 03/December/2023
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.24.1.4

- **Decision Tree**

A Decision Tree is produced by the ID3 algorithm, which is a commonly used classification method, using the training examples from the dataset. A group of unknown examples is used to assess the categorization efficiency of the generated Decision Tree [29], as shown in Algorithm 3.

Algorithm 3. Decision Tree

Input: attributes, instances, parent_instances, depth= nodes are expanded until all are pure
Output: Decision: Person's identity (label of test signature)
Begin
Step 1: start constructing Decision Tree with the root node R.
Step 2: Compute the information Gain in order to select the most suitable attribute in the set of data.
Step 3: Make specific R sets that contain possible outcomes for the finest characteristics.
Step 4: Create the best attribute-containing Decision Tree node.
Step 5: The selected portions of the dataset produced in step 3 will be used to iteratively form newer decision trees. Keep going through this procedure until reaching a point where we can no longer categorize the nodes, at which point we will refer to the last node as a leaf node.
End

- **Adaboost**

Adaboost is a well-known ensemble learning approach for categorization. The core principle of AdaBoost is to mix the results of numerous weak classifiers, in which each weak classifier is an approach that outperforms at random somewhat better. AdaBoost produces a powerful classifier that excels at capturing complicated patterns in data by continuously boosting these weak classifiers and providing greater weight to misclassified samples in each iteration as illustrated in Algorithm 4.

Algorithm 4. Adaboost

Input: Hybrid signature feature vector
Output: Decision: Person's identity (label of test signature)
Begin
Step1: Give each sample in the training dataset a comparable weight.
Step2: For each iteration:
    2.1: Using the existing sample, train an ineffective learner on the training dataset.
    2.2: Determine the ineffective learner's weighted error.
    End for
End

IV. RESULTS AND DISCUSSION

The experimental results are presented in this section from using our technique on two different datasets of handwritten signatures. Additionally, the system's outcomes are compared with state-of-the-art methods that utilize and incorporate the same datasets.

The two datasets are used separately to compare the effectiveness and performance of our suggested approach with different approaches: SigComp2011, which is considered more challenging as it contains Chinese and Dutch signatures, and the popular CEDAR dataset. We divide the datasets

Received 18/July/2023; Accepted 03/December/2023

ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.24.1.4

randomly into training (70%) and test (30%) partitions. The accuracy was used as a performance metric to evaluate the efficacy of our method, as shown in Equation 15.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{15}$$

Where TP, TN, FP, and FN stand for the corresponding True Positive, True Negative, False Positive, and False Negative results. Table III shows the results that we obtained when training the proposed hybrid handwritten signature features resulting from the combination of three types of feature extraction methods and using four types of classifiers. The highest accuracy percentages that we obtained were when using the Naive Bayes classifier, and it was equal to 94% on the SigComp2011 dataset and 100% on the SigComp2011 dataset.

TABLE III. ACCURACY VALUES FOR EACH CLASSIFIER BASED ON SIGCOMP2011 AND CEDAR DATASETS

Method	Accuracy on SigComp2011 Dataset	Accuracy on CEDAR Dataset
Hybrid Features+Naive Bayes	94.43%	100%
Hybrid Features +K-Nearest Neighbor	90.78%	100%
Hybrid Features + Decision Tree	87.94%	96.21%
Hybrid Features +AdaBoost	67.37%	77.50%

When our method is contrasted with new methods utilizing accuracy on the CEDAR dataset, it is evident that our method outperforms them, with an accuracy value of 100% when using Naive Bayes and K-Nearest Neighbor classifier. As for the SigComp2011 dataset, it excelled in state-of-the-art methods when using the Naive Bayes classifier with an accuracy of 94.43% as shown in Table III. By combining multiple feature types into a hybrid feature vector, we are providing the classifiers (Naive Bayes and K-NN) with a richer and more informative representation of the data. K-NN achieves higher accuracy than Decision Tree because K-NN ignores the ineffective signature features, whereas a decision tree might be affected by irrelevant features. In general, the CEDAR dataset outperforms the Sig Comp2011 dataset due to the complexity of the Sig Comp2011 dataset, which contains different writing styles and characters because it contains Chinese and Dutch signatures, whereas the CEDAR dataset only contains English signatures, as a result, differences in script and language can complicate the classification process. Some state-of-the-art methods struggle with lower scores, others manage to obtain very high accuracy. The effectiveness of signature identification systems is significantly influenced by the selection of features extraction methods, classifications algorithms, and datasets. Using this hybrid features give as more comprehensive representation of signatures as well as reducing the sensitivity of system to noise that appear in signature images. For instant, if one feature type is sensitive to noise, the other types may compensate and provide more reliable information. R. Ghosh (2020) [29] achieves a near-perfect accuracy of 99.94% on the CEDAR dataset this concluding that deep learning methods, appear to hold promise for obtaining high accuracy rates. Our method outperform the deep learning method of (Ghosh (2020) [29]) since machine learning algorithms (K-NN and Naive Bayes) perform well with smaller datasets, which may not be enough for a deep learning method (that often require a huge quantities of data) to learn these handwritten signatures.

I. Hadjadj (2019) [31] reaches less than that of other state-of-art methods, suggesting that the classification strategy which is SVM classifier and the textural features might not be as successful for identification task on SigComp2011signature dataset as shown in Table IV.

Received 18/July/2023; Accepted 03/December/2023
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.24.1.4

TABLE IV. COMPARISON BETWEEN OUR SIGNATURE VERIFICATION METHOD AND OTHER METHODS

Method	Classifier	Features	Accuracy	Dataset
A. Rexit (2022) [15]	RF Classifier	LomoHOG	96.66%	CEDAR
M. Varol Arisoy (2021) [31]	Siamese Neural Network	Siamese Neural Network	92.11%	SigComp2011
R. Ghosh (2020) [30]	RNN	Structural and directional	99.94%	CEDAR
I. Hadjadj (2019) Chinese [32]	SVM classifiers	Textural features	75.98%	SigComp2011
B. Cozzens (2018) [33]	CNN	CNN	84.74%	SigComp2011
Our method	Naive Bayes	The proposed hybrid features	100%	CEDAR
Our method	K-Nearest Neighbor	The proposed hybrid features	100%	CEDAR
Our method	Decision Tree	The proposed hybrid features	96.21%	CEDAR
Our method	Naive Bayes	The proposed hybrid features	94.43%	SigComp2011

V. CONCLUSIONS

This study provides a hybrid feature-based and machine-learning method for handwritten signature identification. On two distinct datasets (CEDAR and SigComp2011), it appears that the hybrid feature method obtained differing degrees of accuracy using various machine learning algorithms. The highest accuracy percentages that we obtained were when using the Naive Bayes classifier, which was equal to 94% on the Sigcomp2011 dataset and 100% on the Sigcomp2011 dataset, while the lowest accuracy that we obtained was when using the AdaBoost classifier and it was equal to 67.37% on SigComp2011 dataset and 77.50% on CEDAR dataset. Naive Bayes and K-Nearest Neighbor achieve flawless accuracy on the CEDAR dataset, whereas the suggested hybrid features show encouraging outcomes in enhancing accuracy for handwritten signature identification tasks. On the SigComp2011 dataset, there is a modest decrease in accuracy, indicating that the efficiency of these features may differ depending on the dataset and the machine learning technique selected. It was concluded that the hybrid features that were used to distinguish signatures gave better results than the traditional features because they gave a more comprehensive description of the signatures, also the use of the machine learning algorithms gave higher results compared to traditional deep learning algorithms that were used on the same datasets, which may not be enough for a deep learning method. It might be necessary to conduct more analysis and testing to refine the method for various datasets and applications.

CONFLICT OF INTEREST

There are no conflicts of interest to declare.

Received 18/July/2023; Accepted 03/December/2023

DOI: https://doi.org/10.33103/uot.ijccce.24.1.4

REFERENCES

[1] Y. Zhou, J. Zheng, H. Hu, and Y. Wang, "Handwritten Signature Verification Method Based on Improved Combined Features," Applied Sciences, vol. 11, no. 13, p. 5867, Jun. 2021, doi: https://doi.org/10.3390/app11135867.

[2] M. Saleem and B. Kovari, "Online signature verification using signature down-sampling and signer-dependent sampling frequency," Neural Computing and Applications, Oct. 2021, doi: https://doi.org/10.1007/s00521-021-06536-z.

[3] Z. Hashim, H. M. Ahmed, and A. H. Alkhayyat, "A Comparative Study among Handwritten Signature Verification Methods Using Machine Learning Techniques," Scientific Programming, vol. 2022, pp. 1–17, Oct. 2022, doi: https://doi.org/10.1155/2022/8170424.

[4] H. M. Ahmad and S. R. Hameed, "Eye Diseases Classification Using Hierarchical MultiLabel Artificial Neural Network," 2020 1st. Information Technology To Enhance e-learning and Other Application (IT-ELA, Baghdad, Iraq, 2020, pp. 93-98, doi: 10.1109/IT-ELA50150.2020.9253120.

[5] M. A. Taha and H. M. Ahmed, "Iris Features Extraction and Recognition based on the Local Binary Pattern Technique," 2021 International Conference on Advanced Computer Applications (ACA), Maysan, Iraq, 2021, pp. 16-21, doi: 10.1109/ACA52198.2021.9626827.

[6] M. A. Taha and H. M. Ahmed, "A fuzzy vault development based on iris images," EUREKA: Physics and Engineering, no. 5, pp. 3–12, Sep. 2021, doi: https://doi.org/10.21303/2461-4262.2021.001997.

[7] A. Et al., "Eye Detection using Helmholtz Principle," Baghdad Science Journal, vol. 16, no. 4(Suppl.), p. 1087, Dec. 2019, doi: https://doi.org/10.21123/bsj.2019.16.4(suppl.).1087.

[8] H. M. Ahmad and S. R. Hameed, "Eye Diseases Classification Using Hierarchical MultiLabel Artificial Neural Network," 2020 1st. Information Technology To Enhance e-learning and Other Application (IT-ELA, Baghdad, Iraq, 2020, pp. 93-98, doi: 10.1109/IT-ELA50150.2020.9253120.

[9] G. Ganapathi and N. Rethinaswamy, "A fuzzy framework for offline signature verification," 2014 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), 2014. doi:10.1109/conecct.2014.6740344.

[10] E. N. Zois, L. Alewijnse, and G. Economou, "Offline signature verification and quality characterization using poset-oriented grid features," Pattern Recognition, vol. 54, pp. 162–177, Jun. 2016, doi: https://doi.org/10.1016/j.patcog.2016.01.009.

[11] X. Chen et al., "Assessment of signature handwriting evidence via score-based likelihood ratio based on comparative measurement of relevant dynamic features," vol. 282, pp. 101–110, Jan. 2018, doi: https://doi.org/10.1016/j.forsciint.2017.11.022.

[12] P. Maergner et al., "Combining graph edit distance and triplet networks for offline signature verification," vol. 125, pp. 527–533, Jul. 2019, doi: https://doi.org/10.1016/j.patrec.2019.06.024.

[13] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, "Writer-independent feature learning for Offline Signature Verification using Deep Convolutional Neural Networks," Apr. 2016, doi: https://doi.org/10.1109/ijcnn.2016.7727521.

[14] M. Yilmaz and K. Ozturk, "Hybrid User-Independent and User-Dependent Offline Signature Verification with a Two-Channel CNN," Jun. 2018, doi: https://doi.org/10.1109/cvprw.2018.00094.

[15] A. Rexit, M. Muhammat, X. Xu, W. Kang, A. Aysa, and K. Ubul, "Multilingual Handwritten Signature Recognition Based on High-Dimensional Feature Fusion," Information, vol. 13, no. 10, p. 496, Oct. 2022, doi: 10.3390/info13100496.

[16] M. Liwicki et al., "Signature Verification Competition for Online and Offline Skilled Forgeries (SigComp2011)," 2011 International Conference on Document Analysis and Recognition, 2011, pp. 1480-1484, doi: 10.1109/ICDAR.2011.294.

[17] M. A. Shaikh, T. Duan, M. Chauhan and S. N. Srihari, "Attention based Writer Independent Verification," 2020 17th International Conference on Frontiers in Handwriting Recognition (ICFHR), Dortmund, Germany, 2020, pp. 373-379, doi: 10.1109/ICFHR2020.2020.00074.

[18] R. R. M. Dorothy, Joany, J. Rathish, and S. Prabha, "Image enhancement by Histogram equalization," International Journal of Nano Corrosion Science and Engineering, vol. 2, pp. 21–30, 2015.

[19] P. Singhal, A. Verma and A. Garg, "A study in finding effectiveness of Gaussian blur filter over bilateral filter in natural scenes for graph based image segmentation," 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2017, pp. 1-6, doi: 10.1109/ICACCS.2017.8014612.

[20] J. Rasheed, "Analyzing the Effect of Filtering and Feature-Extraction Techniques in a Machine Learning Model for Identification of Infectious Disease Using Radiography Imaging," Symmetry, vol. 14, no. 7, p. 1398, Jul. 2022, doi: https://doi.org/10.3390/sym14071398.

Received 18/July/2023; Accepted 03/December/2023

DOI: https://doi.org/10.33103/uot.ijccce.24.1.4

[21] Erwin Yudi Hidayat, N. A. Fajrian, Azah Kamilah Muda, Choo Yun Huoy, and S. Ahmad, “A comparative study of feature extraction using PCA and LDA for face recognition,” Dec. 2011, doi: https://doi.org/10.1109/isias.2011.6122779.

[22] H. A. Dwaich and H. A. Abdulbaqi, “Signature Texture Features Extraction Using GLCM Approach in Android Studio,” Journal of Physics: Conference Series, vol. 1804, no. 1, p. 012043, Feb. 2021, doi: https://doi.org/10.1088/1742-6596/1804/1/012043.

[23] N. Iqbal, R. Mumtaz, U. Shafi, and S. M. H. Zaidi, “Gray level co-occurrence matrix (GLCM) texture based crop classification using low altitude remote sensing platforms,” PeerJ Computer Science, vol. 7, p. e536, May 2021, doi: https://doi.org/10.7717/peerj-cs.536.

[24] S. Bakheet and A. Al-Hamadi, “Automatic detection of COVID-19 using pruned GLCM-Based texture features and LDCRF classification,” Computers in Biology and Medicine, vol. 137, p. 104781, Oct. 2021, doi: https://doi.org/10.1016/j.compbiomed.2021.104781.

[25] D. Yanqing, Y. Guoqing, and Z. Yanjie, “Remote Sensing Image Content Retrieval Based on Frequency Spectral Energy,” Procedia Computer Science, vol. 107, pp. 448–453, 2017, doi: https://doi.org/10.1016/j.procs.2017.03.088.

[26] R. A. Abtan, A. H. Al-Saleh, H. J. Mohamed, H. K. Abbas, and A. Alzuky, “Texture Features of Grey Level Size Zone Matrix for Breast Cancer Detection”, Iraqi Journal of Science, vol. 64, no. 1, pp. 492–502, Jan. 2023.

[27] D. Prabha, J. Aswini, B. Maheswari, R. S. Subramanian, R. Nithyanandhan and P. Girija, "A Survey on Alleviating the Naive Bayes Conditional Independence Assumption," 2022 International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), Trichy, India, 2022, pp. 654-657, doi: 10.1109/ICAISS55157.2022.10011103.

[28] A. Almomany, W. R. Ayyad, and A. Jarrah, “Optimized implementation of an improved KNN classification algorithm using Intel FPGA platform: Covid-19 case study,” Journal of King Saud University - Computer and Information Sciences, Apr. 2022, doi: https://doi.org/10.1016/j.jksuci.2022.04.006.

[29] M. Al Hamad and A. M. Zeki, "Accuracy vs. Cost in Decision Trees: A Survey," 2018 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), Sakhier, Bahrain, 2018, pp. 1-4, doi: 10.1109/3ICT.2018.8855780.

[30] R. Ghosh, “A Recurrent Neural Network based deep learning model for offline signature verification and recognition system,” Expert Systems with Applications, p. 114249, Nov. 2020, doi: https://doi.org/10.1016/j.eswa.2020.114249.

[31] M. VAROL ARISOY, “SIGNATURE VERIFICATION USING SIAMESE NEURAL NETWORK ONE-SHOT LEARNING,” International Journal of Engineering and Innovative Research, Aug. 2021, doi: https://doi.org/10.47933/ijeir.972796.

[32] I. Hadjadj, A. Gattal, C. Djeddi, M. Ayad, I. Siddiqi, and F. Abass, “Offline signature verification using textural descriptors,” in Pattern Recognition and Image Analysis, Cham: Springer International Publishing, 2019, pp. 177–188, doi: 10.1007/978-3-030-31321-0_16.

[33] B. Cozzens et al., (2018). Signature verification using a convolutional neural network: AEOP/STEM/REAP/RET programs technical report.

Received 18/July/2023; Accepted 03/December/2023