Iraqi Journal for Computer Science and Mathematics

Journal Homepage: http://journal.esj.edu.iq/index.php/IJCM
e-ISSN: 2788-7421 p-ISSN: 2958-0544

# Offline Handwritten Signature Identification based on Hybrid Features and Proposed Deep Model

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

[1] Department of Computer Sciences, University of Technology, Baghdad, IRAQ

[2] Department of Computer Technical Engineering, College of Technical Engineering, Islamic University, Najaf, IRAQ

*Corresponding Author: Zainab Hashim

DOI: https://doi.org/10.52866/ijcsm.2024.05.01.016
R eceived Augest 2023 ; Accepted November 2023 ; Available online February 2024

**ABSTRACT:** Handwritten signature identification is the process of determining 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. However, verifying handwritten signatures is challenging even in the age of digital transactions and remote document authentication. 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. Various techniques have been developed for signature identification, including feature-based and machine learning-based methods. This paper proposes an authentic signature identification method based on integrating static (offline) signature data and a deep-based model, which fuses three types of signature features—Linear Discriminant Analysis as appearance-based features, Fast Fourier Transform as frequency-based features, and Grey-Level Co-occurrence Matrix as texture-based features. The fused features are then fed into the proposed deep-based model of 25 layers to identify each person. For experiments, we employed three datasets: our private collected dataset, called SigArab, and two public datasets called SigComp2011 and CEDAR. The proposed deep model achieved 99.23%, 100%, and 100% accuracy on the SigArab, CEDAR, and SigComp2011 datasets, respectively. In terms of precision, recall, and F-score, the findings revealed positive results for both datasets and exceeded 1.00, 0.487, and 0.655, respectively, on Sigcomp2011 dataset and 1.00, 0.507, and 0.672, respectively, on CEDAR dataset.

**Keywords:** Deep Learning, Fast Fourier Transform, Grey-Level Co-occurrence Matrix, Handwritten Signature, Linear Discriminant Analysis.

## 1. INTRODUCTION

Handwritten signatures are derived from a person's distinctive handwriting as a behavioural biometric to verify their identity. They are among the most widely used biometric traits since they are based on human qualities and are employed in a variety of applications, including banking, credit cards, passports, check processing, and financial records [1].

The two fundamental issues in processing human signatures are signature identification and verification. Identification involves determining the signer of a document. In verification, a test signature is compared with a model signature to check whether the test signature is genuine or counterfeit. The intrapersonal and interpersonal variabilities in signing induce changes depending on factors such as the space available for signing, the type of pen used, and the psychophysical condition of the signer, a key challenge in both signature identification and verification [2]. Signatures can now be identified in two ways: 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 [3]. The features retrieved from offline signatures can be combined to create a variety of effective features with recognisable individuality. Online signatures can be acquired by signing on devices with touchscreens such as tablets and mobile phones, and many features can be recorded through the use of a specific pen and tablet, as well as a scanned signature image [4]. Features from online signatures can be retrieved by gathering rich data such as typing velocity, typing direction, writer strength, and stroke order [5]. As it is more popular and its structural information is more intuitive when reflecting a writer's characteristics, offline signature identification is more practical than online signature identification.


*Corresponding author: author@organization.edu.co

However, because it captures the dynamic information of the signature in real-time and is difficult for impersonators to copy, the online signature verification mode is more robust than the offline signature verification mode [6].

Signature forgeries are classified into three types: simple, random, and sophisticated. In simple forgery, the forger is aware of the signer's name and other personal information but not their actual signature. When a signature is forged randomly, the forger is aware of either the signer's name or one of their genuine signatures. In sophisticated forgery, the forger is aware of both the signer's name and the true signature, and they frequently practice copying the signer's signature [7].

Several techniques are available for handwritten signature identification, including those based on features, templates, and machine learning. Features such as stroke width, direction, and pressure are extracted from signatures using feature-based approaches. Template-based approaches compare a known template of the signature with a template of an unidentified signature. Machine learning-based approaches use neural networks to learn the characteristics of the signature and then recognise it using the discovered patterns [8].

The main challenge in handwritten signature identification is summarised as follows: Signatures can have a wide range of formats. Hence, it is difficult to create an all-encompassing model that can precisely validate and categorise signatures from various people. Conventional feature extraction techniques may also not capture the most exclusive data gathered from handwritten signatures. Furthermore, conventional deep learning models might need a lot of training and struggle to adapt to changes in signature data.

The findings of this study hold practical importance for various customers, especially financial institutions, courts, and security organisations. A more reliable and accurate signature identification system can enhance feature extraction and classification methods in handwritten signature identification and lower the chance of deception.

The main contribution of this paper is that the proposed work extracts meaningful and perfect signature features based on the fusion of appearance-based, texture-based, and frequency-based features. Using new deep neural network architecture to classify the extracted features, the proposed deep model demonstrated improved results in comparison with previous approaches. The system also achieved great performance by using two types of datasets from the more popular challenge handwritten signature datasets, without any data augmentation techniques. This paper is organised as follows: Section 2 describes the works related to this study. Section 3 describes the materials and methods of the current study. Section 4 describes the study's results. Lastly, Section 5 presents the discussion.

Motivations: Many studies and research papers have explored the field of handwritten signature identification due to the many challenges in this task, such as the diversity in handwritten signature styles. Many of the systems proposed in these papers and studies use pre-trained deep learning models and machine learning algorithms that have yet to yield results in terms of accuracy or generalisability.

## 2. RELATED WORKS

Over the past 20 years, a wide variety of biometric systems [9, 10, 11, 12] have been proposed for confirming a person's identity, signature identification systems being one of them [1]. This section presents an overview of various recent studies that have tackled the issue of identifying signatures. Ganapathi, G., & Nadarajan, R. [13] proposed a method that combines inference rules for image augmentation, fuzzy rough reduction for feature selection, and simplified fuzzy ARTMAP for verification. They presented a writer-dependent system achieving accuracy from 88.95% to 93.99% on the CEDAR dataset. Zois et al. [14] suggested a unique grid-based template-matching approach for offline signature evaluation and verification. They transformed an initial grey-level image into a binary using a standard pre-processing procedure. Then, through feature extraction, they assigned each signature to a family of six groups of grid lattices. A writer-dependent model was subsequently created during training using SVMs. On the CEDAR dataset, they achieved an accuracy of 96.52%. Chen et al. [15] developed a method for investigative handwriting analysis to verify signatures. The method utilised online characteristics including likelihood ratio and width, greyscale, 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 datasets, demonstrating that online characteristics are more accurate than offline characteristics. Maergner et al. [16] 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 available to the public is significantly improved by using both structural and statistical techniques.

Hafemann et al. [17] utilised a combined writer-independent/dependent strategy. First, they used a deep convolutional neural network to train their writer-independent model that was used to extract signature features. Then, a writer-dependent model was trained using the same deep model for verifying handwritten signatures. Yilmaz et al. [18] devised a method akin to this. They used a convolutional neural network (CNN) as a writer-independent model to extract signature features by feeding two signatures into the CNN at one time. The writer-independent strategy, the writer-dependent approach, and a mixture of both were created, with respective EERs of 4.13%, 3.66%, and 1.76%. In that study, skilled forgeries were employed solely during experiments (30 for each skilled forgery person). Aliya Rexit suggested [19] a method for recognising a handwritten signature, which utilised local maximum frequency characteristics and histogram of oriented gradients (HOG) characteristics. A principal component analysis (PCA) spatially reduced the high-dimensional characteristics produced by each of these techniques. The k-nearest neighbours (k-NN) were used for classification and were compared to the random forest approach. In comparison to other approaches, this approach had an identification percentage of 98.4% using a wide-ranging signature dataset.

## 3. OFFLINE HANDWRITTEN SIGNATURE IDENTIFICATION

We developed a reliable method to identify a person based on their signature and built a dataset of Arabic-language signatures from 31 individuals (consisting of 372 signature images). Our research was based on this dataset as well as two publicly available datasets, SigComp2011 [20] and CEDAR [21]. First, we pre-processed all training and test images to get the signatures ready for the identification stage. The images were then converted to greyscale, histogram equalised, blurred, and resized. Then, we proposed an approach based on the fusion of appearance-based features (Linear Discriminative Analysis [LDA]), texture-based features (Grey-Level Co-occurrence Matrix [GLCM]), and frequency-based features (Fast Fourier Transform [FFT]) to build a hybrid feature vector for each image for identification. Finally, we developed a Deep Neural Network model to identify each person. The resultant hybrid feature vectors were fed into the proposed deep neural network architecture to be trained to use them later to classify the new features. Figure 1 shows the proposed signature identification method.

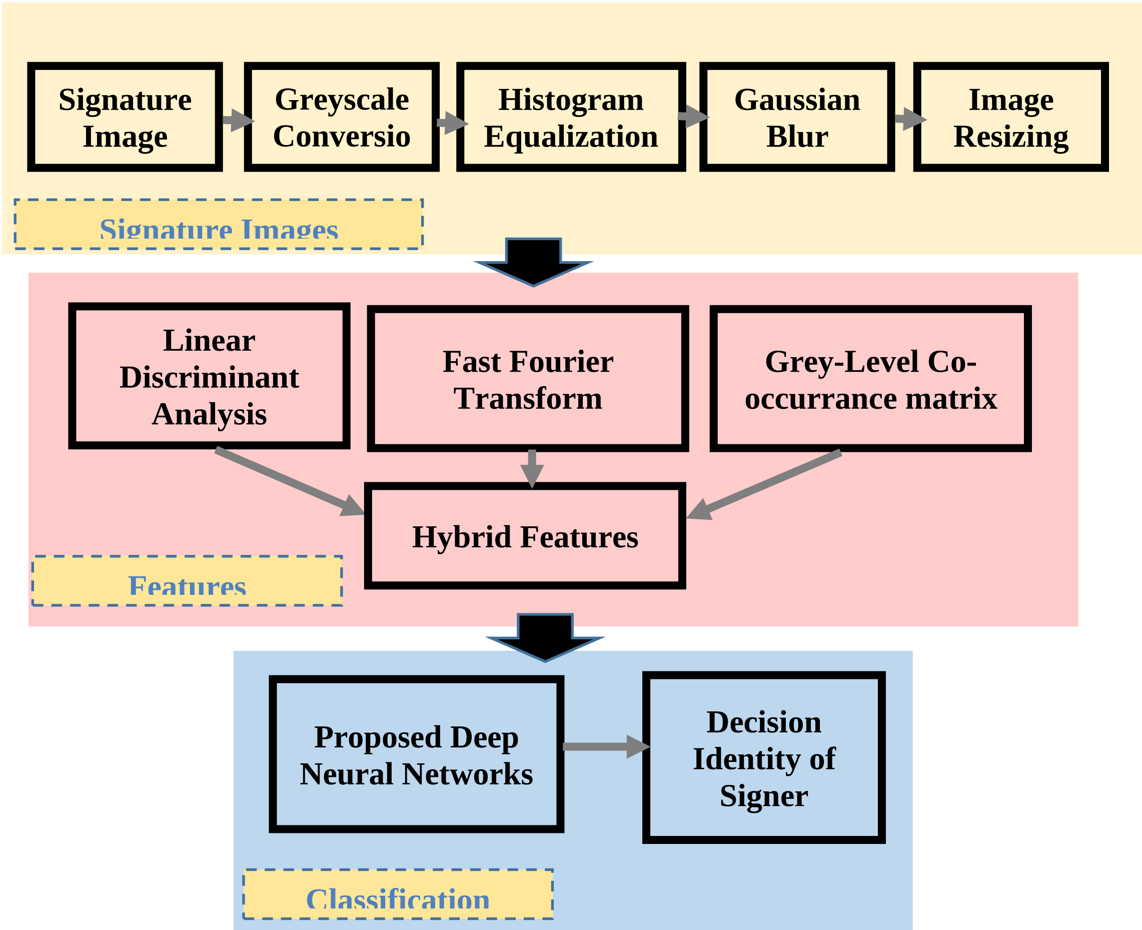



**FIGURE 1. - The proposed signature identification method**

- **SigArab**

  The SigArab dataset was collected from 31 signers with 12 Arabic genuine signature images for each signer. It includes 372 signatures. Each image in it contains a signature produced with either a black, blue, or red pen. All files are PNG signature images scanned at 1200 DPI as shown in Table 1.
- **SigComp2011**

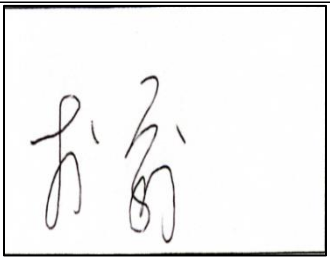
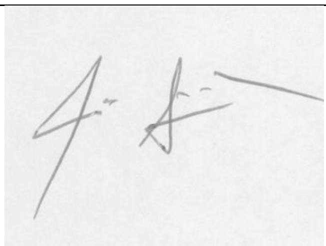
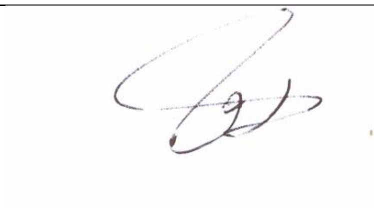
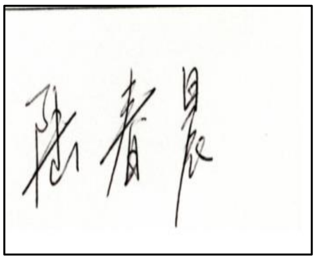
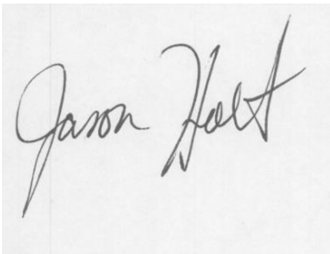
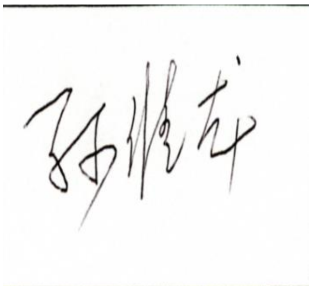
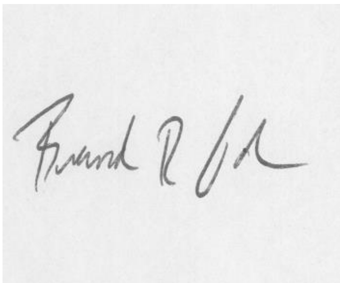
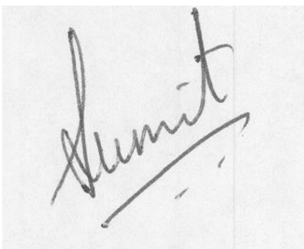
  The public dataset SigComp2011 is available [20]. This dataset was made available for the ICDAR 2011 conference's International Signature Verification Competition. As demonstrated in Table 1, the dataset contains both online and offline signatures from Chinese and Dutch authors. This dataset consists of 74 authors with 12 real signatures and 12 forgeries per author, and it includes PNG signature images scanned at 400 DPI. We used only real offline (Chinese and Dutch signature) subsets to train and test our model.

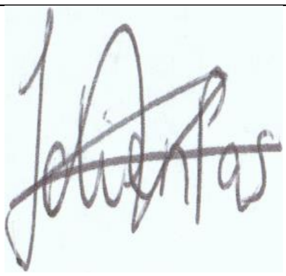
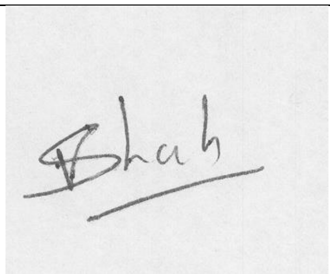
The number of offline Chinese signature images is 575, while the number of Dutch signatures is 362. This makes the total number of images in the SigComp2011 dataset 937.

- **CEDAR**

  CEDAR is an offline signature dataset available from the Centre of Excellence for Document Analysis and Recognition (CEDAR) [21]. The dataset has 55 authors in total, 24 real signatures and 24 expert forgeries per author. It contains signature images written in black pen, and all images are scanned at 300DPI/8bit. The CEDAR dataset has 2640 signature images written in the English language. Similar to previous datasets, only real signatures were used to train and test the model. An illustration of a CEDAR dataset can be found in Table 1.

**Table 1. - Samples of handwritten signature datasets used in the proposed system**

| SigComp2011 | CEDAR | SigArab |
|---|---|---|
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

| | | |
|---|---|---|
|  |  |  |

### 3.1 SIGNATURE PREPROCESSING

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

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

$$Gray = (0.21 \times R) + (0.72 \ \times G) + (0.07 \ \times B) \tag{1}$$

Grey-level images require fewer details to be presented for every single pixel, so using this format will simplify the feature extraction process and accelerate overall processing. Each grey-level brightness is recorded as an 8-bit integer with 256 possible levels of grey, which vary from black to white, as shown in Table 2.

**2. Signature Image Enhancement:** Sometimes poor contrast can affect signature images, which can be brought down due to illumination problems such as an uneven distribution of image brightness. As a result, particular areas of the signature images were first processed by Histogram Equalisation (HE) done after the conversion of signature images to grey-level. The main goals of HE 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 [22]. When performing HE, a significant amount of noisy background is revealed in the signature images for the three dataset types that were employed (SigComp2011, CEDAR, and SigArab), as can be seen in Table 2. 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, a Gaussian smoothing filter is employed with dimensions of 7 * 7 to soften the histogram-equalised 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 2. The Gaussian filter is a sort of visible softening that employs the weighted principle to calculate the mean of each of the pixels by giving every pixel a weight according to its normal distribution [23]. Thus Equation 2 of the Gaussian filter is defined as follows:

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

**3. 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 2. The deep learning model architecture and some of the feature extraction algorithms that were used require signature images to be the same size, whereas raw collected images can be of varying sizes. Bicubic interpolation was used as a resizing method. The bicubic interpolation estimates the pixels in the (i, j) positions using a sampling distance of 16 adjacent pixels (4x4) [24] as shown in Equation 3.

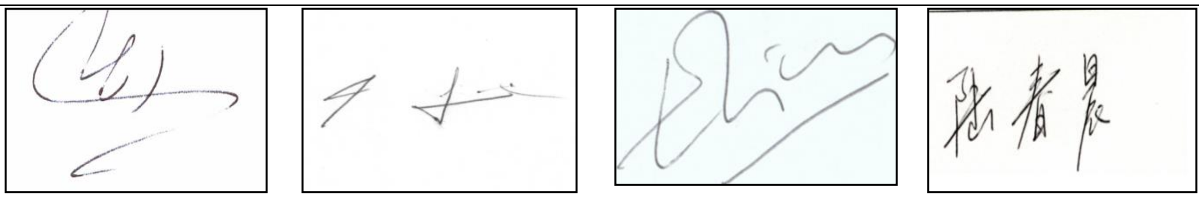
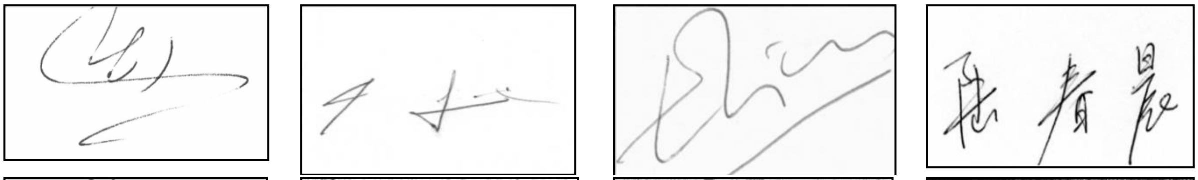
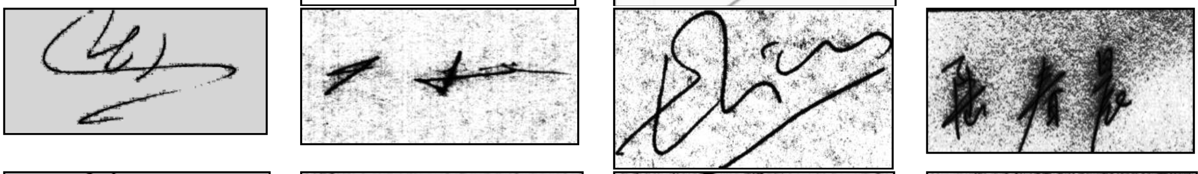
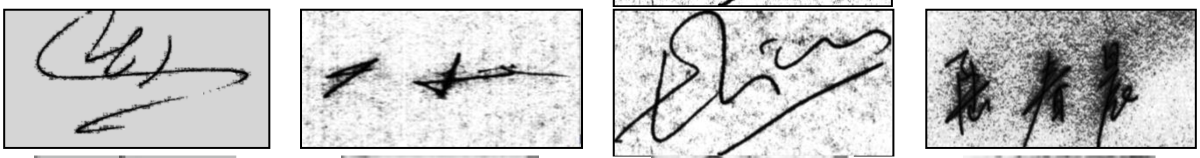
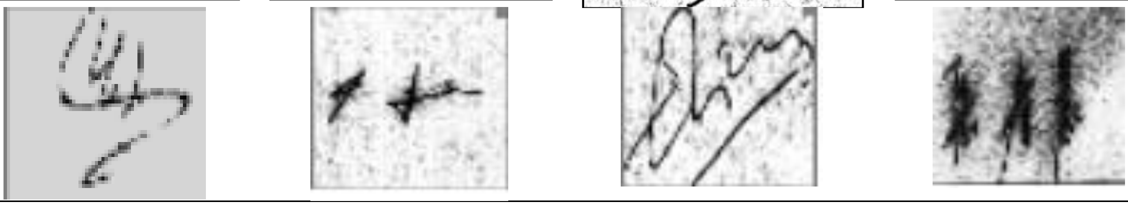
$$fi,j = [W_{-1}(S_Y) W_0(S_Y) W_1(S_Y) \ W_2(S_Y)] \begin{pmatrix} f_{i-1,j-1} & f_{i,j-1} & f_{i+1,j-1} & f_{i+2,j-1} \\ f_{i-1,j} & f_{i,j} & f_{i+1,j} & f_{i+2,j} \\ f_{i-1,j+1} & f_{i,j+1} & f_{i+1,j+1} & f_{i+2,j+1} \\ f_{i-1,j+2} & f_{i,j+2} & f_{i+1,j+2} & f_{i+2,j+2} \end{pmatrix} \begin{bmatrix}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

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 2. - Preprocessing handwritten signature in SigArab, CEDAR, and SigComp2011 Dutch and Chinese [20][21]**

| | |
|---|---|
| Original Signature Images |  |
| Greyscale Conversion |  |
| Histogram Equalised |  |
| Gaussian Blurred |  |
| Resized |  |

### 3.2 SIGNATURE FEATURES EXTRACTION

Feature extraction is considered an important stage in signature identification. In this paper, a fusion of three types of features was carried out to build a hybrid feature vector for each signature image. These were LDA, as appearance-based features; FFT, as frequency-based features, and GLCM as texture-based features as shown in Figure 2. LDA of signature images produces an efficient representation that linearly transforms the original data space into a low-dimensional feature with a focus on the most discriminant features. 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 greyscale can be obtained, which is the basis for analysing the local pattern and arrangement rules of the image.

#### 3.2.1 LINEAR DISCRIMINANT ANALYSIS

R.A. Fisher created Fisher's Linear Discriminant in 1936, which is employed for feature extraction [25]. In this paper, The LDA technique was applied to the pre-processed signature images to extract signature features. LDA extracted 73 features from each image in the SigComp2011 dataset, 54 features from the CEDAR dataset, and 30 features from SigArab. First, the intra-personal scatter matrix $S_I$ and the between-class scatter matrix $S_B$ was calculated as shown in Equation 4 and 5:

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

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

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

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

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

$W_P$ of every signature in the training dataset was contrasted with $W_P$ of the examined signature, utilising a metric for likeness. The training signature, which is the most identical to the test signature, was the outcome [26].

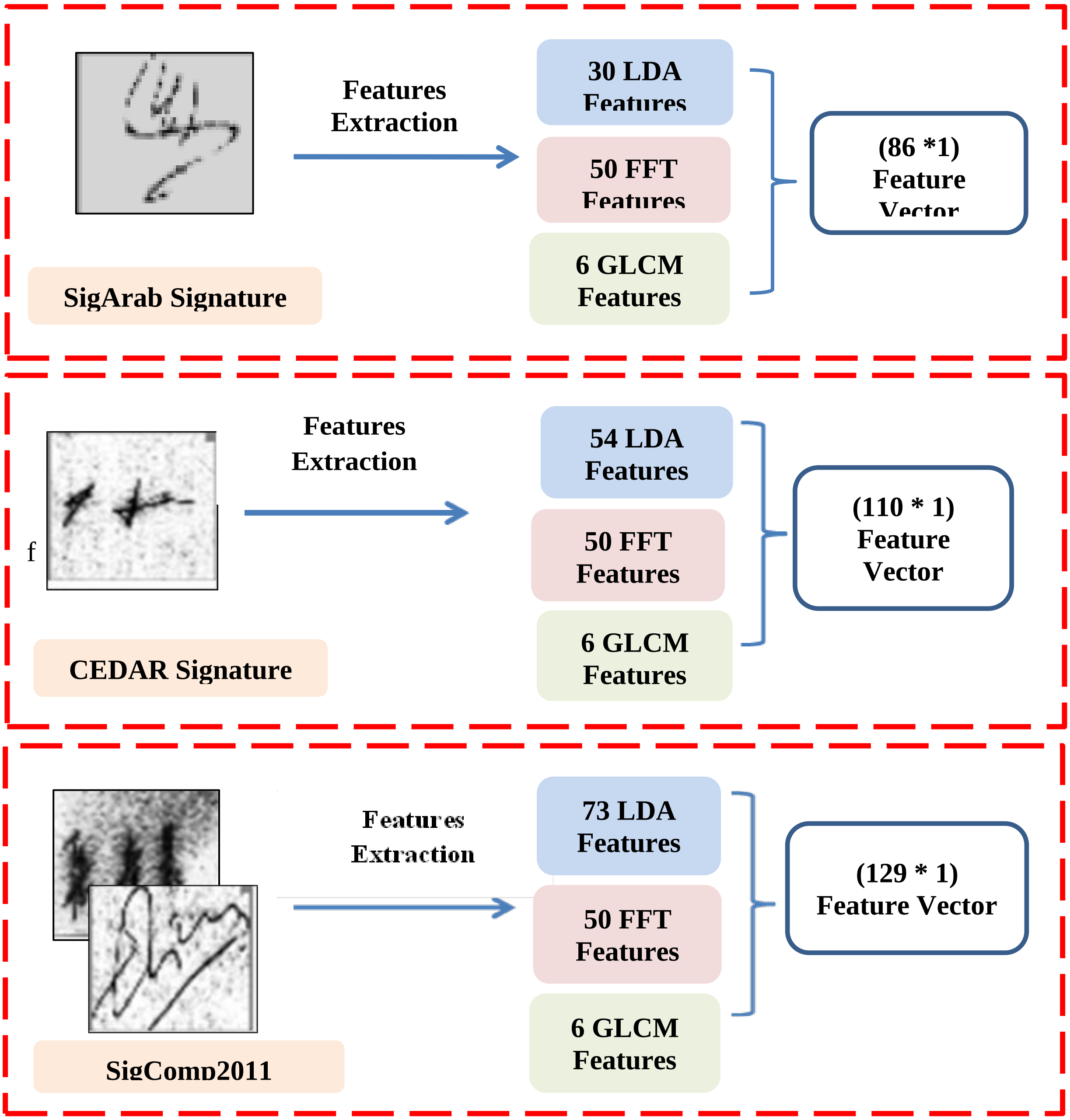



FIGURE 2. - **The proposed features extraction technique**

### 3.2.2 GREY-LEVEL CO-OCCURRENCE MATRIX

Six GLCM features were utilised in the study, namely Contrast, Energy, Homogeneity, Entropy, Mean and Inverse. The GLCM matrix was created with 256 levels, radius=1, and in the horizontal direction.

1) **Contrast:** It is a measurement of local differences or intensity at the greyscale level. Over the entire image, it measures the variations between the pixel point and its neighbours. According to one theory, a high-contrast

image has more tones at either end of the spectrum than a low-contrast image has, which has a smoother range of grey tones (black and white). Equation 7 displays the key formula utilised in contrast calculations:

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

Where $\tilde{P}$ is the estimated probability of the groups of pairs of surrounding grey-levels in the image and N is the overall number of grey-levels employed (the GLCM dimensions) [27].

2) **Energy:** It is a metric of similarity or Angular Second Momentum, 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[28].

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

3) **Entropy:** It is typically categorised as an initial measure of the level of chaos in an image and 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 [29], which is inversely proportional to GLCM energy.

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

4) **Homogeneity:** It is additionally referred to as Inverse Difference Moment (IDM). Large values of GLCM features represent a large homogeneity in the visual texture. The homogeneity is greatest when the image pixel values are the same [28]. As contrast rises, homogeneity declines but the 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 initial amounts of every pixel in the signature image are not what make up the GLCM Mean.; instead, the Mean is numerically equal to the variation in GLCM, in which the pixels will be valued by their frequency of appearance as well as their neighbouring 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 that has been used use is the inverse, which is shown in Equation 12.

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

### 3.2.3 FAST FOURIER TRANSFORM

The FFT method considered a rapid performance of the Discrete Fourier Transform, and it works by transforming data within the range of times to frequencies. The spatial frequency of each object in the distant sensing image is unique. Shape, structure, texture, and other properties of various things can be efficiently reflected in their frequency spectrum energy [30]. In this paper, FFT was used as a feature extractor to extract frequency features from signature images. First, the FFT of a 2D signature image was 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 dimension of the image, and i is sqrt (-1). Then, vector quantisation was applied to spectral signature images to convert them to feature vectors.

### 3.3 THE PROPOSED DEEP NEURAL NETWORK

In this stage, classification processes were implemented by the proposed deep neural network. The primary goal of the proposed model was to categorise the hybrid features derived from the earlier stage to determine the identity of the signer. The proposed model was built with 25 layers, **nine** of them were convolutional layers (with filters equal to 16, 32, 64, 128, 256, 512, and 395, respectively), **six** Max-pooling 1D layers, **eight** Leaky-ReLU 1D layers, **one** Dense 1D layers, **and one** Flatten layer. Since the input layer was a 1D feature vector of size (129 * 1) for the SigComp2011 dataset; (110 * 1) for CEDAR, as shown in Figure 3; and (86 * 1) for the SigArab dataset, as shown in Figure 4, all layers with their parameters were built as 1D layers instead of 2D layers. Due to the simple and compact arrangement of 1D layers, these layers had a low computing need, making real-time and inexpensive hardware implementation possible. The proposed model was trained with 64 batches, 100 epochs, and a learning rate of 0.001. This model design uses the Adam (Adaptive momentum) optimiser, which has a learning rate of 0.001.

A customised version of 2D CNNs called 1D Convolutional Neural Networks (1D CNNs) was used with a kernel of size 3, a padding value equal to 1 (to give the kernel extra space to cover the vector, padding was applied to the frame of the feature vector), and stride of 1 that modifies the amount of movement over the feature vector in which the filter would move one unit at a time, as shown in Algorithm 1. Leaky Rectified Linear Units (Leaky ReLU) function, a non-linear activation function, was applied after the CNN layers. When compared to conventional activation functions, the Leaky ReLU function can speed up the training of deep neural networks. A new pooling layer was added after the Leaky ReLU layers. The Max-Pooling method was then used to obtain the maximum output. For 1D temporal data, the maximum value over the window determined by the pool size was used to downsample the input representation. The window was moved a little. When the "valid" padding option was used, the output had the following shape: output shape = (input shape - pool size + 1) / strides. To take the output of the preceding layers, "flatten" them, and create a single vector that can be an input for the following stage, one flattened layer was added. One dense layer was used after the flattened layer with the Softmax activation function, as a classifier of the resultant vectors. The stated number of neurons in the dense layer would impact the output shape. The dense layer carried out the action: **Activation (dot [input, kernel] + bias)** was equal to output; Table 3 summarizes all the layers of the proposed hyper deep model.

**Algorithm 1: The proposed Deep Neural Network Algorithm.**

**Input: Input shape SigComp2011= (1 * 129), Input shape CEDAR (1 *110), Input shape SigArab (1 * 86).**

**Output: Decision (Signer Identity)**

**Begin**

**Step1: Compute Convolution, Maxpooling, Leaky ReLU, Dense, Flatten**

**1.2: For i=1 to N compute: (N number of features in the dataset)**

**1.2.1: Conv1D (Filters= 16, K-size= 3, Stride=1, Padding=0)**

**1.2.2: Leaky ReLU (Alpha= 0.3)**

**1.2.3: Max-pooling (P-size=1, Stride=2)**

**1.2.4: Leaky ReLU (Alpha= 0.3)**

**1.2.5: Conv1D (Filters= 32, K-size= 3, Stride=1, Padding=0)**

**1.2.6: Max-pooling (P-size=1, Stride=1)**

**1.2.7: Conv1D (Filters= 64, K-size= 3, Stride=1, Padding=0)**

**1.2.8: Leaky ReLU (Alpha= 0.3)**

**1.2.9: Max-pooling (P-size=1, Stride=1)**

**1.2.10: Conv1D (Filters= 128, K-size= 3, Stride=1, Padding=0)**

**1.2.11: Leaky ReLU (Alpha= 0.3)**

**1.2.12: Max-pooling (P-size=1, Stride=1)**

**1.2.13: Conv1D (Filters= 256, K-size= 3, Stride=1, Padding=0)**

**1.2.14: Leaky ReLU (Alpha= 0.3)**

**1.2.15: Max-pooling (P-size=1, Stride=1)**

**1.2.16: Conv1D (Filters= 512, K-size= 3, Stride=1, Padding=0)**

**1.2.17: Leaky ReLU (Alpha= 0.3)**

**1.2.18: Max-pooling (P-size=1, Stride=2)**

**1.2.19: Conv1D (Filters= 1024, K-size= 3, Stride=1, Padding=0)**

**1.2.20: Leaky ReLU (Alpha= 0.3)**

**1.2.21: Conv1D (Filters= 1024, K-size= 3, Stride=1, Padding=0)**

**1.2.22: Leaky ReLU (Alpha= 0.3)**

**1.2.23: Conv1D (Filters= 395, K-size= 3, Stride=1, Padding=0)**

**1.2.24: Flatten**

**1.2.25: Dense (D= 74 or 55 or 31, Act=" Softmax")**

**End For**

**End**

**Table 3. - Layers summary of the proposed model**

| **Layer** | **Filters** | **Parameters** |
|---|---|---|
| Conv1D | **16** | Stride=1, Kernal-size=3 |
| Max-pooling | **16** | Stride=2, Pool size=2 |
| Conv1D | **32** | Stride=1, Kernal-size=3 |
| Max-pooling | **32** | Stride=1, Pool size=2 |
| Conv1D | **64** | Stride=1, Kernal-size=3 |
| Max-pooling | **64** | Stride=1, Pool size=2 |
| Conv1D | **128** | Stride=1, Kernal-size=3 |
| Max-pooling | **128** | Stride=1, Pool size=2 |
| Conv1D | **256** | Stride=1, Kernal-size=3 |
| Max-pooling | **256** | Stride=1, Pool size=2 |
| Conv1D | **512** | Stride=1, Kernal-size=3 |
| Max-pooling | **512** | Stride=1, Pool size=2 |
| Conv1D | **1024** | Stride=1, Kernal-size=3 |
| Conv1D | **1024** | Stride=1, Kernal-size=3 |
| Conv1D | **395** | Stride=1, Kernal-size=3 |
| Flatten | | None |
| Dense | | SoftMax Activation Function |

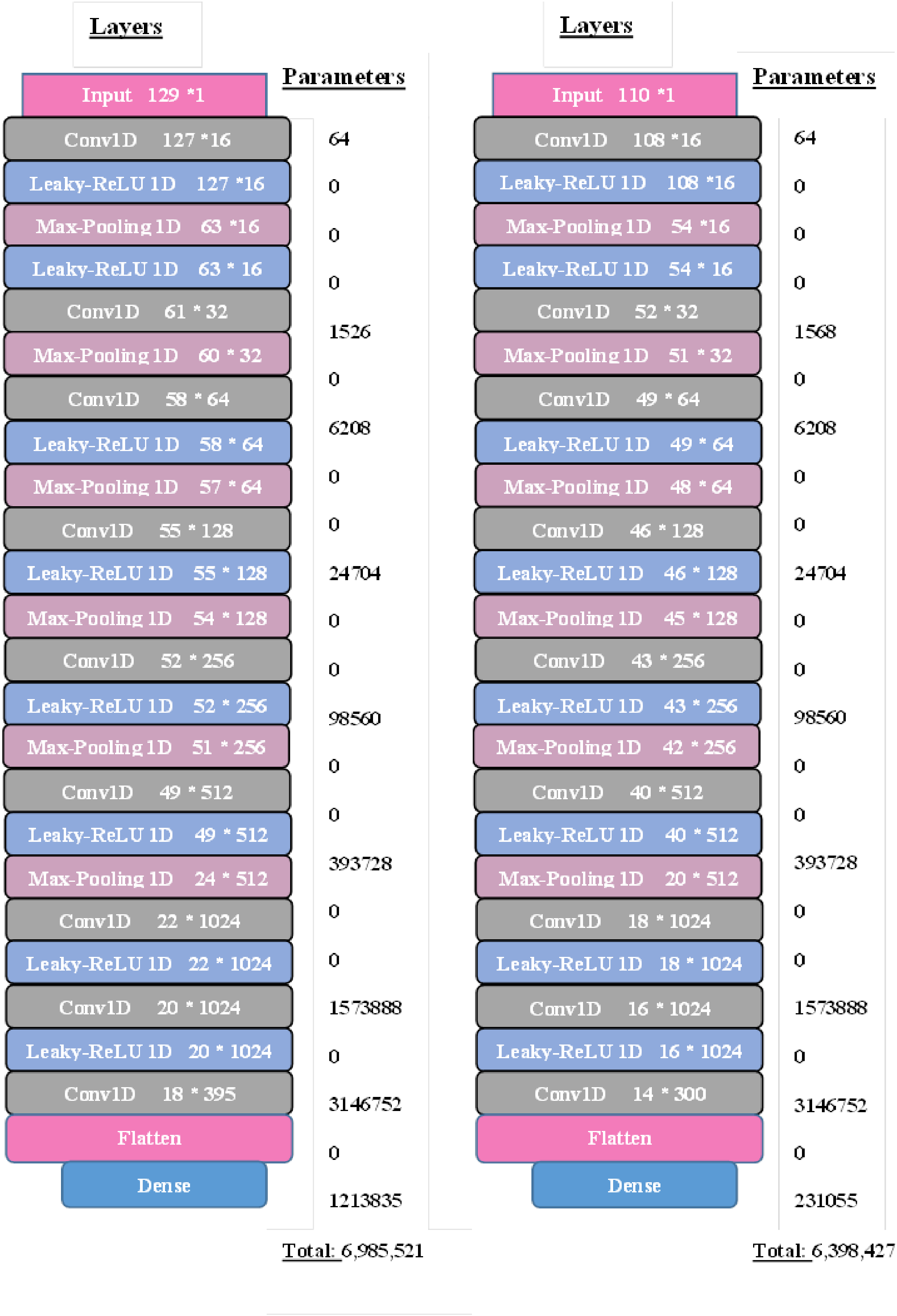



**FIGURE 3. - The proposed deep model architecture on SigComp2011 and CEDAR dataset**

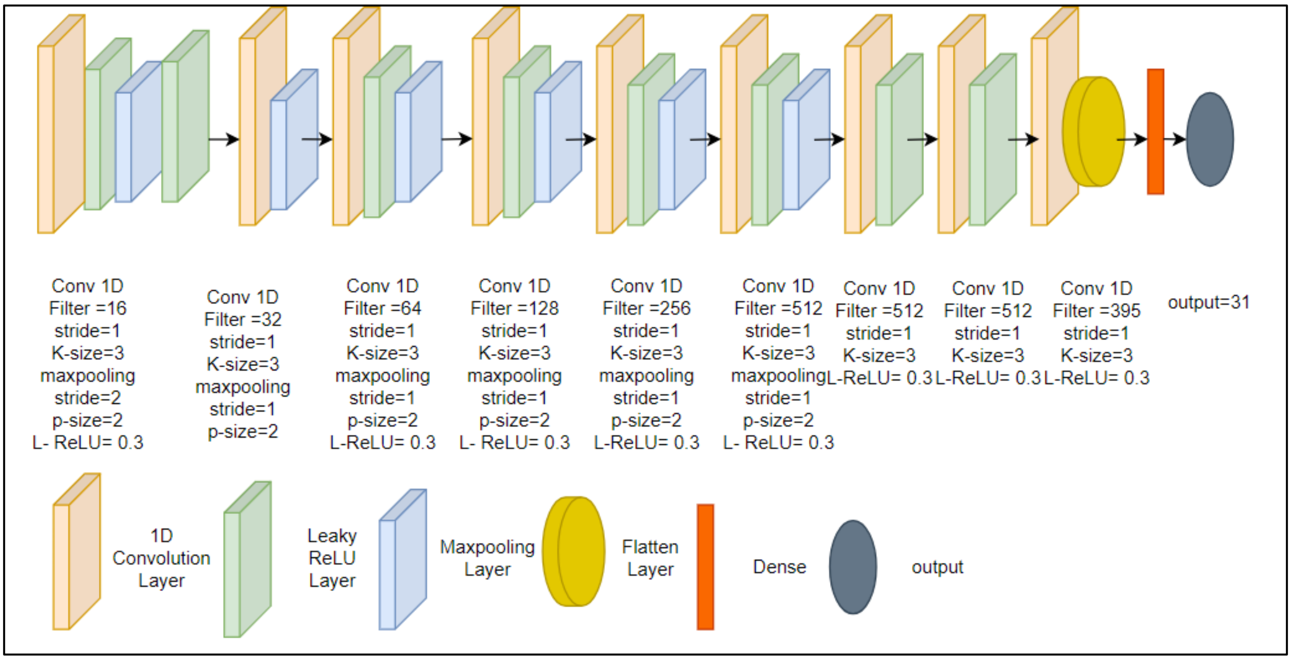



**FIGURE 4. - The proposed deep model architecture on the SigArab dataset**

## 4. RESULTS AND DISCUSSIONS

This section presents the experimental findings from using these techniques on three different datasets of handwritten signatures. Additionally, the outcomes of the system are compared with other deep neural networks such as VGG16 and VGG19 and four types of machine learning algorithms as well as state-of-art methods that utilise and incorporate the same datasets. The proposed model was trained with a learning rate of 0.001, epochs of 100, and a batch size of 64. The total number of parameters obtained from this model on the SigComp2011 dataset is equal to (6,985,521), (6,398,427) on the CEDAR dataset.

The three datasets were used separately to compare the effectiveness and performance of this proposed approach with different approaches. These are SigComp2011, which is considered more challenging as it contains Chinese and Dutch signatures. The total number of images in this dataset is 937, and the popular CEDAR dataset has 2640 images while the private dataset SigArab contains 372 Arabic signatures. The datasets were divided randomly into training (70%) and test (30%) partitions. Four performance metrics—accuracy, precision, recall, F-score, and loss metrics—were used to evaluate the efficacy of this proposed methodology. According to Equation 15, the percentage of true positive and true negative categorised points to all total points is known as accuracy.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (15),$$

where TP, TN, FP, and FN stand for the corresponding True Positive, True Negative, False Positive, and False Negative results, respectively. Equation 16 defines the F1-score as the harmonic mean of precision and recall. Equations 17 and 18 illustrate precision and recall.

$$F1Score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (16)$$

$$Precision = \frac{TP}{TP + FP} \quad (17)$$

$$Recall = \frac{TP}{TP + FN} \quad (18)$$

In the case of the SigComp2011 (Dutch and Chinese) and CEDAR datasets, when this method is compared to other state-of-the-art methods utilising accuracy, it is evident that this method outperforms them, with an accuracy value of 100%. When this system was applied to the collected Arabic dataset, the accuracy was equal to 99.23%, as presented in Table 4. The main consideration of our deep model was to build weights inside the deep layers that are different and spaced between each other; the reason was that these layers are built in a way that each person receives a completely different weight from the other person, which makes the classification process yield high results. This effect is unlike other deep neural networks that focus only on building high weights that can be equal for each person and cause a lower accuracy rate and overfitting. The suggested strategy has improved accuracy metrics with a growing range of 12.3%

compared to F. M. Alsuhimat et al. (2023) [35], which is the lowest accuracy achieved. The state-of-the-art methods achieve accuracies ranging from 87.7% to 99.94%; our system beat them, representing a significant improvement. Some state-of-the-art methods struggle with lower scores, while others manage to obtain very high accuracy. The selection of feature extraction methods, classification algorithms, and datasets significantly influences the effectiveness of signature identification systems. Additionally, some cutting-edge strategies, such as deep learning methods, appear to hold promise for obtaining high accuracy rates. Two CNN models VGG16 and VGG19 have been applied on the same handwritten signature datasets in which the inputs of these models were the original signature images without any pre-processing steps. The results in Table 3 show that both VGG16 and VGG19 achieved poor outcomes compared to that of the proposed deep model. Four classifiers (Naïve Bayes, Decision Tree, AdaBoost, and K-NN) were applied to the same datasets for comparison; the inputs of these classifiers were the proposed hybrid signature features. The highest accuracy percentages were obtained when using the K-NN and Naïve Bayes classifiers, equal to 98.20% on the CEDAR dataset and 97.46% on the Sigcomp2011 dataset, respectively. In the same case, this model outperforms known classification algorithms. The proposed model is used with features extracted from LDA, GLCM, and FFT, which lead to the best results of Precision, Recall, F-measure. Table 5 shows the proposed system results on the SigComp2011 and CEDAR datasets. The main limitation solved by our proposed system is improving the accuracy of identifying persons by signature, which was not achieved in previous systems, without reaching the state of overfitting when the system reaches 100% accuracy.

The only limitation that appeared in our proposed system is the high complexity because the proposed identification system did not rely only on the deep learning model to extract the signature features and classify, but rather the features of the signature images were extracted using the proposed hybrid features and then entered into the deep model to extract another features from the hybrid features and classify them.

**Table 4. - Compares the outcomes of this model using the SigComp2011 and CEDAR datasets with those of other systems**

| Method | Dataset | Features Extraction | Classification | Accuracy | EER |
|---|---|---|---|---|---|
| G. Ganapathi and N. Rethinaswamy (2013) [13] | CEDAR | Histogram based | Fuzzy Method | 93.99% | - |
| Zois et al. (2016) [14] | CEDAR | Grid-based template | SVM | - | 3.02 |
| Chen et al. (2018) [15] | SigComp2011 Dutch | Correlation Coefficient | Score based likelihood | 96.17% | - |
| Chen et al. (2018) [15] | SigComp2011 Chinese | Correlation Coefficient | Score based likelihood | 96.17% | - |
| P. Maergner et al. (2019) [16] | CEDAR | Graph edit distance | Multiple classifier system | - | 0.30 |
| Aliya Rexit (2022) [19] | CEDAR | LomoHOG | RF Classifier | 96.66% | - |
| Aliya Rexit (2022) [19] | CEDAR | Lomo | RF Classifier | 98.4% | - |
| R. Ghosh (2020) [31] | CEDAR | Structural and directional | RNN | 99.94% | 0.01 |
| H.-H. Kao and C.-Y. Wen [32] (2020) | SigComp2011 Dutch | CNN | CNN | 98.96% | - |

| | | | | | |
|---|---|---|---|---|---|
| T. M. Maruyama et al. (2021) [33] | CEDAR | SigNet-F | SVM | - | 0.82 |
| M. Jampour et al. (2019) [34] | CEDAR | Fractal features | Chaos game theory | 98.76% | - |
| F. M. Alsuhimat et al. (2023) [35] | CEDAR | HOG | LSTM | 87.7% | 11.40 |
| M. S. Kadhm et al. (2021)[36] | CEDAR | Histogram of Sparse Codes (HSC) | ANN | 99.7% | - |
| VGG16 | SigComp2011<br>CEDAR | VGG16 | VGG16 | 50.71%<br>49.89% | - |
| VGG19 | SigComp2011<br>CEDAR | VGG19 | VGG19 | 66.80%<br>56.2% | - |
| Decision Tree | SigComp2011<br>CEDAR | The Proposed method (LDA+FFT+GLCM) | Decision Tree | 87.94%<br>96.21% | - |
| Adaboost | SigComp2011<br>CEDAR | The Proposed method (LDA+FFT+GLCM) | Adaboost | 67.37%<br>77.50% | - |
| Naïve Bayes | SigComp2011<br>CEDAR | The Proposed method (LDA+FFT+GLCM) | Naïve Bayes | 94.32%<br>97.46% | - |
| K-NN | SigComp2011<br>CEDAR | The Proposed method (LDA+FFT+GLCM) | K-NN | 90.78%<br>98.20% | - |
| The Proposed Method | SigComp2011<br>CEDAR<br>SigArab | The Proposed method (LDA+FFT+GLCM) | Proposed Deep Neural Network | 100%<br>100%<br>99.23% | - |

**Table 5. - The proposed model results in terms of precision, recall, F-score and loss**

| Dataset | Precision | Recall | F-score | Loss |
|---|---|---|---|---|
| SigComp2011 | 1.00 | 0.487 | 0.655 | 0.00001011 |
| CEDAR | 1.00 | 0.507 | 0.672 | 0.00000060849 |

## 5. CONCLUSIONS

This study introduced an authentic signature identification method based on the integration of static (offline) signature data and a proposed deep-based model. In this method, three types of signature features were fused: LDA as appearance-based features, FFT as frequency-based features, and GLCM as texture-based features. To assess the efficiency and performance of this model, two well-known datasets, SigComp2011 and CEDAR, and our private dataset SigArab were employed. First, four pre-processing stages were carried out on the datasets. Then, LDA, GLCM, and FFT were used as feature extraction methods to build a hybrid feature vector for each image. These features were then fed into the proposed model. The primary goal of the proposed model was to categorise the hybrid features derived from the earlier stage to identify each signer. This model was built with 25 layers, **nine** of which were convolutional layers (with filters equal to 16, 32, 64, 128, 256, 512, and 395, respectively), **six** Max-pooling 1D layers, **eight** Leaky-ReLU 1D layers, **one** Dense 1D layers, **and one** Flatten layer. The suggested technique enhanced the identification accuracy and outperformed other modern techniques, with an accuracy value of 100% for both public datasets and 99.23% for the private dataset. Additionally, it depicted a high precision rate reaching 1.00 for both public datasets, which can be attributed to the architecture of the model and the choice of effective signature features. This occurred due to several reasons, including the appropriate choice of a variety of features, as well as the way of building the deep model layers to create different, spaced weights for each person in the datasets used (that is, each class had weights spaced apart from each other), unlike that in previously trained models, such as VGG16 and VGG19, which produce accuracy rates lower than that of our proposed system.

In future enhancements, the offline handwritten features used in this paper can be combined with their online features such as typing velocity, direction, writer strength, and strokes to produce a comprehensive identification system. An optimisation method can also be added to the system as feature selection before the deep learning stage to improve the identification process because a high dimensional set of features may lead to overfitting. Meanwhile, more investigations about signature features will be kept to have the best identification effect with the least amount of training data.

## Funding

None

## ACKNOWLEDGEMENT

None

## CONFLICTS OF INTEREST

The author declares no conflict of interest.

## REFERENCES

[1] 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.

[2] M. Varol Arisoy, "Signature Verification Using Siamese Neural Network One-shot Learning," International Journal of Engineering and Innovative Research, 2021, doi: 10.47933/ijeir.972796.

[3] Y. Zhou, J. Zheng, H. Hu, and Y. Wang, "Handwritten signature verification method based on improved combined features," MDPI, https://www.mdpi.com/2076-3417/11/13/5867/htm (accessed Jul. 13, 2023).

[4] M. Fayyaz, Mohammad Taghi Saffar, M. Sabokrou, M. Hoseini, and M. Fathy, "Online signature verification based on feature representation," Mar. 2015, doi: https://doi.org/10.1109/aisp.2015.7123528.

[5] N. H. Al-banhawy, Al-Azhar University, Egypt, H. Mohsen, N. Ghali, Future University in Egypt, and Future University in Egypt, "Signature Identification and Verification Systems: A Comparative Study on the Online and Offline Techniques," Futur. Comput. Inform. J., vol. 5, no. 1, pp. 28–45, 2020, doi: 10.54623/fue.fcij.5.1.3.

[6] S. Rohilla, A. Sharma, and R. K. Singla, "Role of sub-trajectories in online signature verification," Array, vol. 6, p. 100028, Jul. 2020, doi: https://doi.org/10.1016/j.array.2020.100028.

[7] M. M. Hameed, R. Ahmad, M. L. M. Kiah, and G. Murtaza, "Machine learning-based offline signature verification systems: A systematic review," Signal Processing: Image Communication, vol. 93, p. 116139, Apr. 2021, doi: https://doi.org/10.1016/j.image.2021.116139.

[8] M. Saleem and B. Kovari, "Online signature verification using signature down-sampling and signer-dependent sampling frequency - neural computing and applications," SpringerLink, https://link.springer.com/article/10.1007/s00521-021-06536-z (accessed Jul. 13, 2023).

[9] 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.

[10] 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.

[11] M. A. Taha and H. M. Ahmed, "A fuzzy vault development based on iris images", Eureka: PE, no. 5, pp. 3-12, Sep. 2021, doi: 10.21303/2461-4262.2021.001997.

[12] A. et al., "Eye Detection using Helmholtz Principle", Baghdad Sci. J, vol. 16, no. 4(Suppl.), p. 1087, Dec. 2019.

[13] 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

[14] 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.

[15] 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.

[16] 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.
[17] 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.
[18] 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.
[19] 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.
[20] 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.
[21] 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.
[22] 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.
[23] 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.
[24] B. K. Triwijoyo and A. Adil, "Analysis of medical image resizing using Bicubic interpolation algorithm," Jik, vol. 14, no. 1, p. 20, 2021, doi: 10.24843/JIK.2021.v14.i01.p03.
[25] 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.
[26] 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.
[27] H. A. Dwaich and H. A. Abdulbaqi, "Signature texture features extraction using GLCM approach in android studio," J. Phys. Conf. Ser., vol. 1804, no. 1, p. 012043, 2021, doi: 10.1088/1742-6596/1804/1/012043.
[28] 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 Comput. Sci., vol. 7, no. e536, p. e536, 2021, doi: 10.7717/peerj-cs.536.
[29] S. Bakheet and A. Al-Hamadi, "Automatic detection of COVID-19 using pruned GLCM-Based texture features and LDCRF classification," Comput. Biol. Med., vol. 137, no. 104781, p. 104781, 2021, doi: 10.1016/j.compbiomed.2021.104781.
[30] D. Yanqing, Y. Guoqing, and Z. Yanjie, "Remote sensing image content retrieval based on frequency spectral energy," Procedia Comput. Sci., vol. 107, pp. 448–453, 2017, doi: 10.1016/j.procs.2017.03.088.
[31] 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.
[32] 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," Applied Sciences, vol. 10, no. 11, p. 3716, May 2020, doi: https://doi.org/10.3390/app10113716.
[33] T. M. Maruyama, L. S. Oliveira, A. S. Britto and R. Sabourin, "Intrapersonal Parameter Optimisation for Offline Handwritten Signature Augmentation," in IEEE Transactions on Information Forensics and Security, vol. 16, pp. 1335-1350, 2021, doi: 10.1109/TIFS.2020.3033442.
[34] M. Jampour and A. Naserasadi, "Chaos game theory and its application for offline signature identification," IET Biom., vol. 8, no. 5, pp. 316–324, 2019, doi: 10.1049/iet-bmt.2018.5188.
[35] F. M. Alsuhimat and F. S. Mohamad, "Offline signature verification using long short-term memory and histogram orientation gradient," Bulletin of Electrical Engineering and Informatics, vol. 12, no. 1, pp. 283–292, 2023. doi:10.11591/eei.v12i1.4024.
[36] M. S. Kadhm, M. Jassim Mohammed, and H. Ayad, "An accurate signature verification system based on proposed HSC approach and Ann Architecture," Indonesian Journal of Electrical Engineering and Computer Science, vol. 21, no. 1, p. 215, 2021. doi:10.11591/ijeecs.v21.i1.pp215-223.