Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.047852

ARTICLE

Improving the Transmission Security of Vein Images Using a Bezier Curve and Long Short-Term Memory

Ahmed H. Alhadethi[1,*], Ikram Smaoui[2], Ahmed Fakhfakh[3] and Saad M. Darwish[4]

[1]Computer Science and School of Electronics and Telecommunications, University of Sfax, Sfax, 3029, Tunisia

[2]LETI Laboratory, University of Sfax, Sfax, 3029, Tunisia

[3]Digital Research Center of Sfax, Technopark of Sfax, Sfax, 3029, Tunisia

[4]Department of Information Technology, Institute of Graduate Studies and Research, Alexandria University, Alexandria, Egypt

*Corresponding Author: Ahmed H. Alhadethi. Email: ahmadhusham1985@gmail.com

Received: 20 November 2023 Accepted: 22 April 2024 Published: 20 June 2024

ABSTRACT

The act of transmitting photos via the Internet has become a routine and significant activity. Enhancing the security measures to safeguard these images from counterfeiting and modifications is a critical domain that can still be further enhanced. This study presents a system that employs a range of approaches and algorithms to ensure the security of transmitted venous images. The main goal of this work is to create a very effective system for compressing individual biometrics in order to improve the overall accuracy and security of digital photographs by means of image compression. This paper introduces a content-based image authentication mechanism that is suitable for usage across an untrusted network and resistant to data loss during transmission. By employing scale attributes and a key-dependent parametric Long Short-Term Memory (LSTM), it is feasible to improve the resilience of digital signatures against image deterioration and strengthen their security against malicious actions. Furthermore, the successful implementation of transmitting biometric data in a compressed format over a wireless network has been accomplished. For applications involving the transmission and sharing of images across a network. The suggested technique utilizes the scalability of a structural digital signature to attain a satisfactory equilibrium between security and picture transfer. An effective adaptive compression strategy was created to lengthen the overall lifetime of the network by sharing the processing of responsibilities. This scheme ensures a large reduction in computational and energy requirements while minimizing image quality loss. This approach employs multi-scale characteristics to improve the resistance of signatures against image deterioration. The proposed system attained a Gaussian noise value of 98% and a rotation accuracy surpassing 99%.

KEYWORDS

Image transmission; image compression; text hiding; Bezier curve; Histogram of Oriented Gradients (HOG); LSTM; image enhancement; Gaussian noise; rotation

1 Introduction

Communication and digital technologies have changed society's daily activities and had major economic and social impacts on information used in all relevant domains [1]. Due to the rapid growth of the Internet, Mobile Networks, and social media, the development of more secure authentication systems with higher performance on mobile devices has increased dramatically [2]. Newly developed technologies make accessing, processing, storing, and transmitting information simpler and more affordable [3]. In this ever-changing and evolving environment, secure communications must be established using an authentication system. It is currently an important target for researchers to develop secure authentication systems using certain authentication techniques [4]. At present, sharing images via the Internet is commonplace, requiring strong authentication techniques to ensure secure transmission. Authentication refers to a service that ensures whether a given block of data has integrity (i.e., that the associated content has not been modified) and was provided by a legitimate sender [5]. Authentication is traditionally ensured through mechanisms that involve message authentication codes (MACs) and digital signatures [6], known as “hard authenticators”. In hard authentication, a MAC (also known as a message digest) or a digital signature of the data to protect called an authenticator is created at the source and transmitted with the data. At the receiver, the authenticator is verified using the received data to deduce whether the received information is, in fact, unmodified and from the alleged sender.

When the data represent an image that can travel through a set of diverse distribution chains, they are susceptible to content-preserving operations such as compression, transcoding, and other standard-format conversions that severely impede the usefulness of hard authentication mechanisms. Any processing of an image that changes its bit representation yet still maintains the validity of the perceptual content may be inaccurately categorized as “inauthentic.” Thus, more recently, there has been a movement toward developing schemes that provide “soft authentication,” in which content-preserving processing is distinguished from unlawful content-changing manipulations. One toolset that was recently applied to soft-authentication, and which is a partial focus of this work, is called semi-fragile digital watermarking. In this method, an authenticator which may consist of a MAC or digital signature for salient parts of an image is used to form a watermark. This watermark is imperceptibly embedded within the original image (commonly called the host). The integration of the authenticator within the image to be secured simplifies the logistical problems associated with the MAC and digital signature data handling during image transmission. Moreover, semi-fragile watermarking can provide information on the degree and location of tampering within an image, allowing for more application-relevant decisions based on credibility [7]. The methodology begins with the utilization of the k-means technique to compress the sending image efficiently. Subsequently, the paper extracts distinctive features from the compressed image using the Histogram of Oriented Gradients (HOG) method. To further enhance the intricacy of the model, the extracted features are incorporated into the LSTM deep learning algorithm. The results obtained from the LSTM are then utilized to generate a sequence of Bezier curves. These curves collectively form the proposed signature, providing a unique and robust representation. In the final step of the process, the outcomes of the LSTM are employed as spatial indicators within the original image, facilitating the concealment of the generated signature.

This article proposes an effective adaptive compression algorithm that achieves notable reductions in computational and energy requirements while maintaining minimal deterioration in image quality throughout the communication. This approach employs multi-scale characteristics to enhance the resilience of signatures against image degradation. This innovative approach not only showcases the effectiveness of combining compression, feature extraction, deep learning, and curve generation but also underscores the practical application of LSTM results in the context of image-based signature hiding. This paper makes a significant contribution by employing a series of sophisticated algorithms and techniques in the domain of image processing. Furthermore, this article is organized as follows. A comprehensive introduction is presented in Section 1. Section 2 presents and discusses the related works. Moreover, the methodology and proposed method are presented in detail in Sections 3 and 4. Section 5 presents and discusses the results of the proposed system. Finally, the conclusion of the research is presented in Section 6.

2 Related Work

Paper data transmission requires compression and decompression, especially in scenarios with restricted bandwidth or storage. For example, compression using an upgraded AI-powered model can reduce the fingerprint image size for biometric data encryption, thus increasing the speed of wireless transmission. Data are compressed by deleting extraneous information while maintaining their key qualities. Fingerprints are self-similar, with repeated patterns and structures, enabling compression algorithms to simplify them, thus reducing the fingerprint image size without sacrificing crucial information by quickly recognizing and encoding these patterns. Zhan et al. [8] have employed a deep-learning self-encoder framework to improve image reconstruction and reduce RoI distortion. Many compression strategies, however, have positives and negatives. Mital et al. [9] have proposed compressing an information source utilizing only decoder-side correlated information on stereo image pairs from synchronized and calibrated cameras. While previous methods based on deep neural networks convert the input image into a latent representation, compressing the image with less loss of features through entropy coding, these authors used a cross-attention module in the decoder to align the feature mappings from the input image's latent representation and the decoder-side information's latent representation, thus improving the use of correlated patches. Li et al. [10] have proposed a compression domain processing-based method for efficient analysis of Whole-Slide Images (WSIs), which are increasingly significant for modern pathologists. Due to the vast size of WSIs, current methods for analyzing such images involve picture decompression and are computationally costly. Pyramidal magnification structures and compression domain characteristics can be used to assign different decompression depths to WSI patches depending on features directly retained from compressed or partially decompressed patches. Attention-based clustering screens for low-magnification patches, resulting in variable decompression depths for high-magnification patches at different locations. When dealing with sensitive information, such as biometrics, authentication plays a key role in any safe communication system. For example, authentication ensures that only authorized individuals can access and modify encrypted biometric data within the context of the use of enhanced AI-powered models.

Several methods exist for authenticating data, such as the method proposed by Li et al. [11], which makes use of K-Means clustering and the side match methodology to recover and restore mosaiced images. To implement Image Secret Sharing (ISS) without pixel enlargement, Yan et al. [12] presented a new public-key-based bi-directional shadow image authentication mechanism, in which the shadow picture is authenticated only during the decoding phase using conventional dealer-participatory methods, which suffer from substantial pixel expansion and require the storage of supplementary information; however, the approach suggested by the authors permits shadow image authentication in both the encoding and decoding stages, unlike previous methods. Watson's Visual Model and LLE form the basis of the image-hashing technique reported by Yan et al. [13]. To build hashing sequences that accurately reflect how humans perceive images, the suggested system for such images must be able to reliably differentiate between copies and similar images. First, Watson's Visual Model is used to modify the weights of the DCT coefficients for non-overlapping image blocks; then, the Hu-invariant moments of each block are combined to build an intermediate feature matrix that is subsequently encrypted using chaotic hashing.

Guarino et al. [14] introduced a TGSB approach for recognizing gender and age groups. This approach employs a transfer learning strategy that is applied to Convolutional Neural Networks (CNNs). These CNNs are trained using image-based representations of touch gestures executed by users on mobile devices. In their study, Guarino et al. [15] investigated the influence of noise on the efficacy of a biometric system by employing pattern recognition techniques for the identification of individuals based on diverse characteristics such as facial features and fingerprints. Abuturb et al. [16] introduced an innovative optical single-channel security system designed for the protection of multiple color images. This system utilizes 3D logistic map biometric keys as a means of authentication, employing a chaotic random phase mask to modulate the biometric key and subsequently encrypting each channel of the original color images individually. Grizheboskya et al. [17] focused on correcting geometric distortions and incorrect scales in fingerprint images. These corrections play a vital role in ensuring precise recognition and identification in biometric systems. The authors of [18] investigated the application of deep learning algorithms for text feature extraction and classification. Researchers have explored the use of Recurrent Neural Networks (RNNs) and LSTM networks to capture sequential information and learn meaningful representations from textual data, leading to improved performance in tasks such as sentiment analysis and text categorization. While reference [19] focused on deep learning-based feature extraction for speech emotion recognition. and propose a model that combines CNNs and LSTM networks to extract discriminative features from speech signals. These learned features are then used for accurately classifying emotional states in speech data.

In the consideration of this paper, the fingerprint data might be signed before compression to guarantee its legitimacy. The fingerprint data itself or some other mechanism (e.g., hash function or a digital signature) may be used to construct the signature, which is a unique identifier. Before employing the Least Significant Bit (LSB) method to embed data in an image, the signature must be generated. After the signature is constructed, it can be included in the compressed fingerprint data by employing the same LSB method. This feature makes it extremely difficult to detect or tamper with the signature during transmission, as it is embedded in the image. The LSB method allows the signature to be recovered from the received picture and compared to the original signature at the receiving end. If the two signatures are consistent, the recipient knows that the data is genuine and that the image was not altered during transmission. It is also crucial to safeguard images against any transmission errors or noise that may be introduced. The use of error-correcting codes allows for the detection and correction of any transmission faults. These codes, along with the compressed fingerprint data and signature, can be embedded in the image using the same LSB method. Transmission faults can be detected and remedied at the receiving end using error-correcting codes. These measures ensure that the data has not been tampered with in transit and can be relied upon when analyzed.

3 Methodology

3.1 K-Means Clustering

Clustering is a process that involves dividing data into groups that share certain characteristics, according to patterns in the data. K-Means clustering is a clustering technique used to identify clusters of data objects, as illustrated in Fig. 1. Due to the nature of clustering, it is considered an unsupervised learning approach, as we do not need to label the data given that this method recognizes similar patterns between the data [20]. K-Means clustering is a centroid-based algorithm, whereby central points are chosen and data points close to each central point are grouped according to certain properties in the data. Minimizing the sum of distances between the points and their respective cluster centroids is the main process of K-Means clustering [21]. Various distance metrics can be used in the K-Means clustering algorithm to calculate the distance between the data points and centroids [22,23].

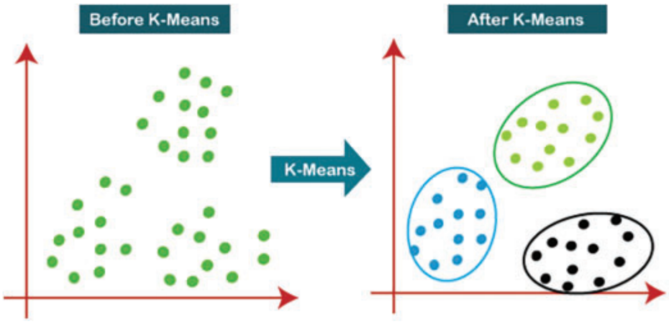


Figure 1: K-Means clustering technique

3.2 Histogram of Oriented Gradients (HOG)

HOG is a feature descriptor of a grayscale image that is formed of a group of normalized gradient histograms, which is obtained as follows: The scan window's pixels are organized into r × r rectangular cells. The magnitude of the gradient in each cell determines the orientation of the gradient. A histogram is then used to represent all the orientations as votes. For optimal results, fine quantization is used for the gradient orientations. For testing, the range $(0, \pi)$ is divided into 8 histogram bins and, as the contrast between an object and its backdrop is typically uncertain, $(\pi, 2\pi)$ is mirrored [24–26]. The values of the 8c2 histogram bins for each cell are concatenated into a vector B, which is then normalized. This process builds a HOG feature vector with high dimensionality. HOG can be thought of as taking a non-linear function of an image's edge orientations, concentrating on small spatial areas to eliminate the sensitivity to exact edge localization [27]. HOG is resistant to changes in illumination and can achieve a high degree of computational precision when detecting objects with a variety of textures. Recent high-performance general-purpose processors have the computational capability to perform real-time object detection. However, as such processors consume large amounts of power, they are inappropriate for use in mobile devices with limited battery life. Therefore, low-power, high-performance HOG feature extraction processors are needed, such that the associated range of applications can be expanded. A detection window scans the input images, and each cell's gradient orientations are combined into a histogram. For increased illumination invariance, histogram normalization can be achieved by combining the local histogram energy over blocks and applying the results, allowing for excellent computational precision in the detection of various textured objects [28]. Each cell in a block is massive, and the histograms (HOG features) are normalized and collected over the detection window [27]. The HOG feature extraction is presented in Fig. 2.

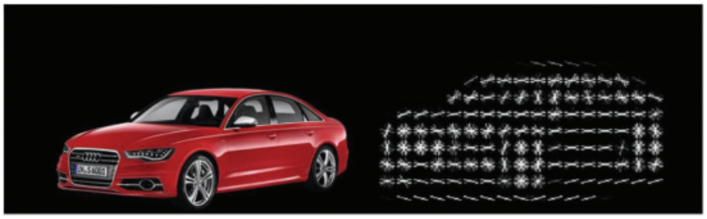


Figure 2: HOG feature extraction

3.3 Long Short-Term Memory (LSTM)

As a subset of the neural network family, RNNs are designed to handle serial data. There are two types of RNNs: Discrete-time RNNs and continuous-time RNNs [29]. These networks are designed with a periodic connection structure, allowing them to update their current state based on both previous and current input data. RNNs are generally artificial neural networks consisting of standard repetitive cells. These types of neural networks are known to be able to identify potential conditions very accurately. RNNs specialize in processing a series of values (e.g., time-series data). Fig. 3 illustrates the general structure of an RNN algorithm [30].

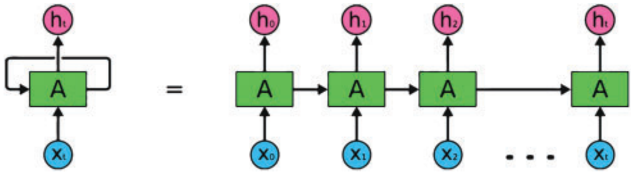


Figure 3: General structure of an RNN algorithm

Most RNNs can process variable-length sequences. However, they cannot typically learn long-term dependencies. To resolve this issue, in 1997, Hochreiter and Schmidhuber proposed a solution called LSTM [31]. LSTM is a type of RNN that is capable of learning and preserving long-term dependencies and is distinguished by the presence of a memory cell. In an LSTM, gates control the insertion or deletion of information within a cell. Three types of gates are used to regulate information flow within the cell: Forget gates, input gates, and output gates. The forget gate determines how much information from the prior cell state is removed. The input gate controls the information that enters the cell state. Then, a vector for a filter layer is generated and attached to the cell state. According to the previous two lines, the old cell state is subsequently updated to a new state. The output layer then selects the cell state information for use as the output [32]. The final LSTM layer at the final time step (***n***) transmits the output (***hn***) to a dense layer with a single-output neuron, which calculates the final flow (***y***). Fig. 4 illustrates the overall structure of the LSTM algorithm [33].

3.4 Bezier Curve

The Bezier curve can be described as a parametric curve denoted by P(t), which is a polynomial function of the parameter t. The number of points employed to identify this curve affects the degree of the polynomial. This mechanism generates an approximate curve utilizing certain control points. The curve does not necessarily pass through any of these points; rather, it is attracted toward them [33–36]. This phenomenon can be understood as the points “pulling” on the curve. Each point influences the curve’s orientation by pulling it closer, and this pull is stronger when the curve is closer to the point. An illustration of a cubic Bezier curve is provided in Fig. 5. This curve is defined by four points and a cubic polynomial, making it simple to edit, modify, and/or re-shape. Consequently, this type of curve is popular among researchers. In addition to adding or removing points, the curve may also be edited [35].

The parametric formula for a cubic Bezier curve when considering four control points is as follows:

$$P\left(x,y\right)=\begin{pmatrix}f_x\left(t\right)\\ f_y\left(t\right)\end{pmatrix},\ \text{where } (0\le \mathrm{t}\le 1), \tag{1}$$

$$\mathrm{f_x\left(t\right)=t^3x_0+3\left(1-t\right)t^2x_1+3(1-t)^2tx_2+(1-t)^3x_3}, \tag{2}$$

$$f_y(\mathrm{t}) = \mathrm{t}^3\mathrm{y}_0 + 3(1-\mathrm{t})\mathrm{t}^2\mathrm{y}_1 + 3(1-\mathrm{t})^2\mathrm{t}\mathrm{y}_2 + (1-\mathrm{t})^3\mathrm{y}_3 \tag{3}$$

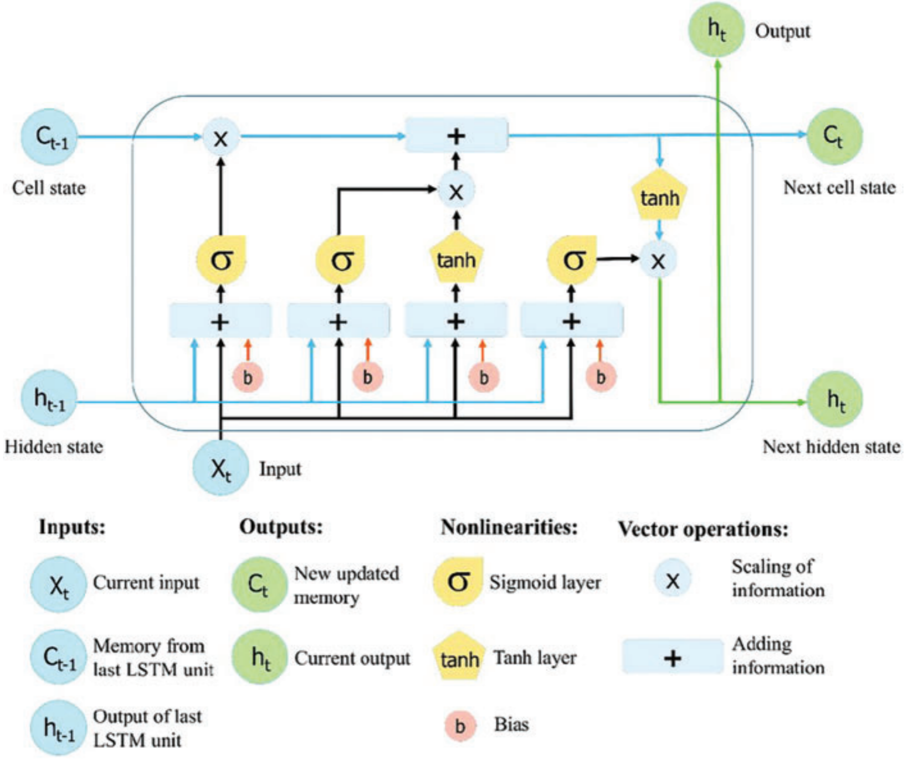


Figure 4: Long short-term memory node structure

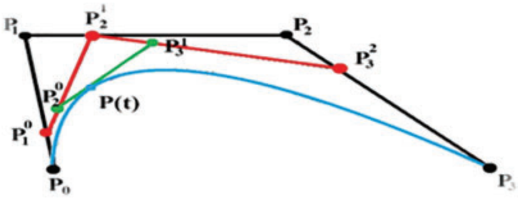


Figure 5: Bezier curve

Eq. (1) for a Bezier curve can be transformed into matrix form as follows [35]:

$$P(t) = [t^3 \quad t^2 \quad t \quad 1]\begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{pmatrix} \tag{4}$$

For example, the cubic Bezier curve is defined by its four control points, P_0, P_1, P_2, and P_3. The following explains the impact of the control points on the Bezier curve: P_0 and P_3 are the starting and ending points of the curve, while P_1 and P_2 determine the direction of the initial curve [36].

Case 1: When t = 0:

$$(0) = [0 \quad 0 \quad 0 \quad 1] \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{pmatrix} = P_0 \tag{5}$$

Case 2: When t = 1:

$$P(t) = [1 \quad 1 \quad 1 \quad 1] \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{pmatrix} = P_3 \tag{6}$$

Bezier curves are fitted at the first and last points. Fig. 6 shows several examples of Bezier curves and shows how the curves fit the first and last points, but not the control points [33].

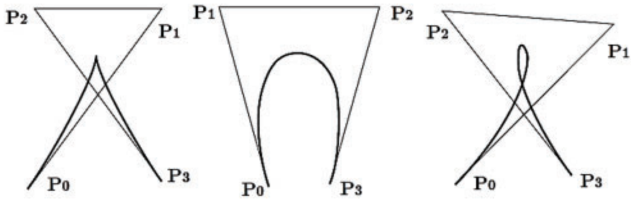


Figure 6: Examples of Bezier curves

3.5 Simulation Setup

The finger vein database was used as an example for implementing the proposed system, and different types of medical images can be used. The proposed system was implemented using the 64-bit Python 3.9 language, relying on many libraries, the most important of which are image processing libraries, deep learning libraries, and 2D drawing libraries.

4 Proposed Method

The proposed system consists of two steps: Signature generation and signature matching.

4.1 Signature Generation

In this step, the sender generates a signature for the vein image to be sent. Fig. 7 presents a block diagram of the proposed system for signature generation. The system first compresses the image using K-Means clustering. The image is composed of pixels, each of which has a three-dimensional representation indicating the intensity values for the red, green, and blue (RGB) colors, which can vary from 0 to 255. The following formula can be used to determine the amount of storage space required for an image with dimensions of 320 × 240 pixels: 320 × 240 × 3 × 8 = 1,843,200 bits. When compressing RGB images via the K-Means clustering approach, k colors associated with the centroids are selected.

The remaining colors are then associated with these centroid points, according to the degree of resemblance between them. Each pixel's value is replaced with the value of the associated centroid point. The image compression size can be determined as follows: If K (centroid points) = 64 is specified, the image size will be 320 × 240 × 6 + 64 × 3 × 8, where the third operand (number 6) indicates the number of bits that can express K values between 0 and 63. If the value of K is increased, the image's overall quality and size will also increase. The utilized K-Means clustering algorithm is outlined in Algorithm 1.

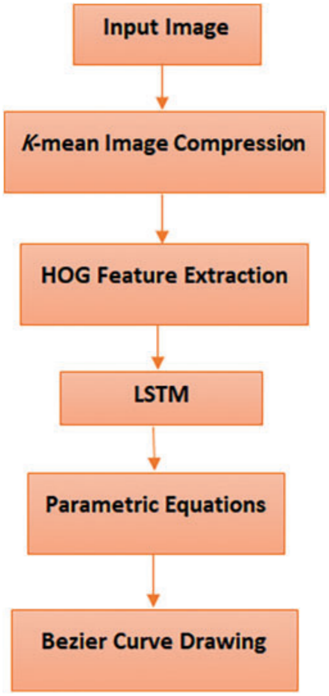


Figure 7: Signature generation block diagram

Algorithm 1: Image Compression (K-Means clustering)

```
Input: Vein Image
Output: Compressed Image
Process:
    Step 1: Load the image of the vein from a disk.
    Step 2: Re-shape the input image, which requires that the input image be transformed from (Rows, cols, 3) to (rows × cols, 3).
    Step 3: Clustering occurs using the K-Means clustering algorithm to locate $k$ centroid points that reflect the color combination of their surroundings.
    Step 4: Replace the values of each pixel with those of its respective centroid point.
    Step 5: Re-shape the compressed image back to its original format (rows, cols, 3).
    Step 6: Return the compressed image.
End
```

Feature extraction is the second step. The HOG method describes features by calculating the number of times a vein image has a gradient orientation within a cell. The key steps followed to calculate HOG features are summarized in Algorithm 2.

```
Algorithm 2: HOG feature extraction
Input: Compressed Vein Image
Output: Extracted features
Process:
    Step 1: Load the compressed vein image.
    Step 2: Gradient computation. In this step, the gradient angles and magnitudes are calculated in the vertical and horizontal directions after calculating spatial gradients in these two directions.
    Step 3: Orientation binning. The image is split into tiny connected pieces called cells the gradient magnitude of every pixel in a cell is divided into multiple orientation bins, based on the gradient angle.
    Step 4: Feature description. Nearby cells are gathered into blocks in this step. Each block is normalized. A descriptor is created by concatenating the normalized block histograms in a detection window.
    Step 5: Return the extracted features.
End
```

In the next step, the feature matrix is used as input to a further algorithm to increase the randomness of the numbers, as outlined in Algorithm 3.

```
Algorithm 3: Increase Randomness using LSTM
Input: HOG Features
Output: Randomness Vector
Process:
    Step 1: Use HOG features as the input X to the LSTM
    Step 2: Calculate the following LSTM equations:
$f_t = \sigma_g \left( W_{fx_t} + U_f h_{t-1} + b_t \right)$
$i_t = \sigma_g \left( W_{ix_t} + U_i h_{t-1} + b_i \right)$
$o_t = \sigma_g \left( W_{ox_t} + U_o h_{t-1} + b_o \right)$
$\overline{c}_t = \sigma_c \left( W_{cx_t} + U_c h_{t-1} + b_c \right)$
$c_t = f_t \bigodot C_{t-1} + i_t \bigodot \overline{c}_t$
$h_t = o_t \bigodot \sigma_h \left( C_t \right)$
where
$\mathrm{x}_t$ : the input vector of the LSTM unit
$f_t$ : the forget gate activation vector
$i_t$ : the input/update activation vector
$c'_t$ : the cell input activation vector
$c_t$ : the cell input vector
$h_t$ : the hidden vector of the LSTM unit
    Step 3: Return Output Vector.
End
```

These outputs are then used as inputs for the parametric equations to generate Bezier curves, as outlined in Algorithm 4.

Algorithm 4: Signature Generation

```
Input: LSTM output vector
Output: Signature
Process:
    Step 1: Read the input vector value by value.
    Step 2: Use each value to generate four points using the following parametric equations:
$P_0 = (x_t + x_t, x_t \times cos\,(t))$
$P_1 = (x_t, x_t \times t)$
$P_2 = (cos\,(x_t) \times t, \sin\,(x_t) \times t)$
$P_3 = (x_t \times t, x_t)$
where $x_t$ is the input value and t is the index of the value
    Step 3: Draw a Bezier curve for each value.
    Step 4: Return the signature.
End
```

4.2 Signature Hiding

In this step, the generated signature is hidden within the compressed image. The locations for this process are selected based on the numbers generated by the LSTM algorithm. Algorithm 5 describes the steps followed to hide the signature. LSB is a technique used in steganography to hide data within an image. This technique involves replacing the least significant bit of each pixel value in the image with the bits of the data to be hidden. As the least significant bit represents the smallest amount of information in a pixel value, modifying it does not significantly affect the quality of the image. When the image is transmitted, the receiver can extract the hidden signature data by extracting the least significant bit of each pixel value to reconstruct the original signature.

Algorithm 5: Signature Hiding

```
Input: LSTM output vector, Signature, Vein Image
Output: Steganographic  Image
Process:
    Step 1: Generate the location of hidden pixels from the LSTM output vector.
    Step 2: Convert signature points into binary (Signature vector).
    Step 3: Extract selected pixels from the vein image and convert into binary.
    Step 4: Hide the signature vector in the LSB of selected pixels.
    Step 5: Return the steganographic  vein image.
End
```

4.3 Signature Matching

In this step, the image including the signature is received by the recipient, which then performs the same processes used in the previous step to generate a signature from the received image. This signature is then matched with the sent signature. Fig. 8 presents the signature matching process.

5 Results and Discussion

Several experiments were conducted on the proposed system. In this paper, the FV-USM was used as a database. The images in the database were collected from 123 classes of 83 males and 40 females, every subject provided four fingers: Left index, left middle, right index, and right middle fingers, resulting in a total of 492 finger classes obtained. Each finger was captured six times in one session individual participated in two sessions. In the first session, a total of 2952 (123 × 4 × 6) images were collected, Fig. 9 illustrates a sample of vein image [37].

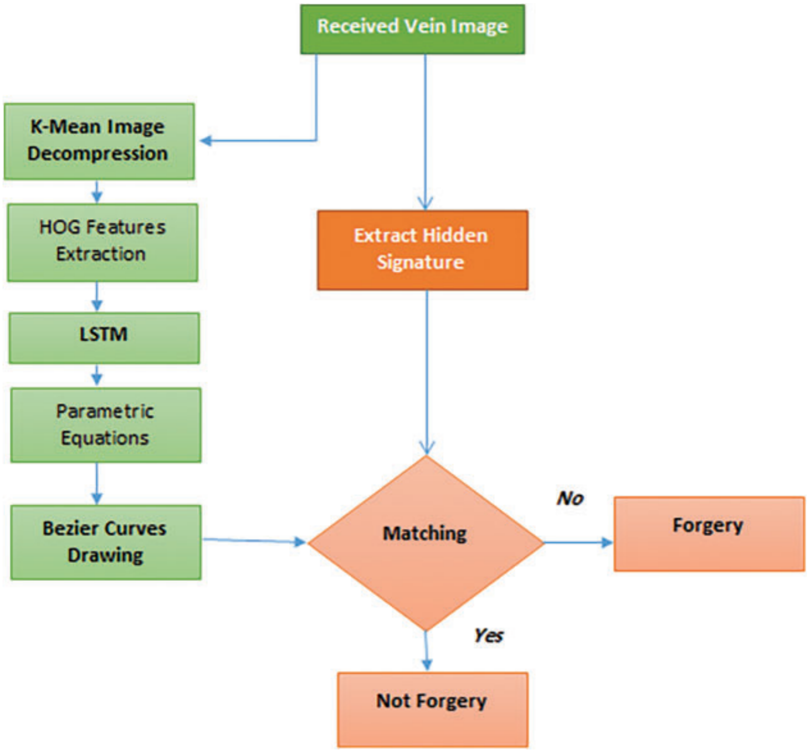


Figure 8: Signature matching block diagram

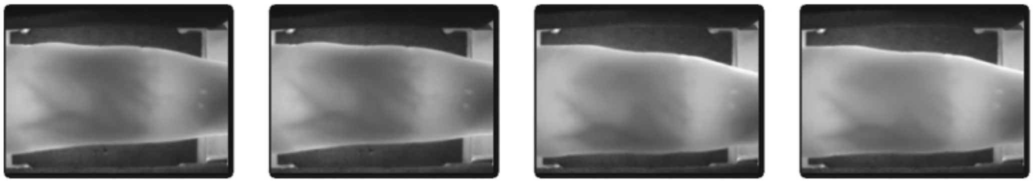
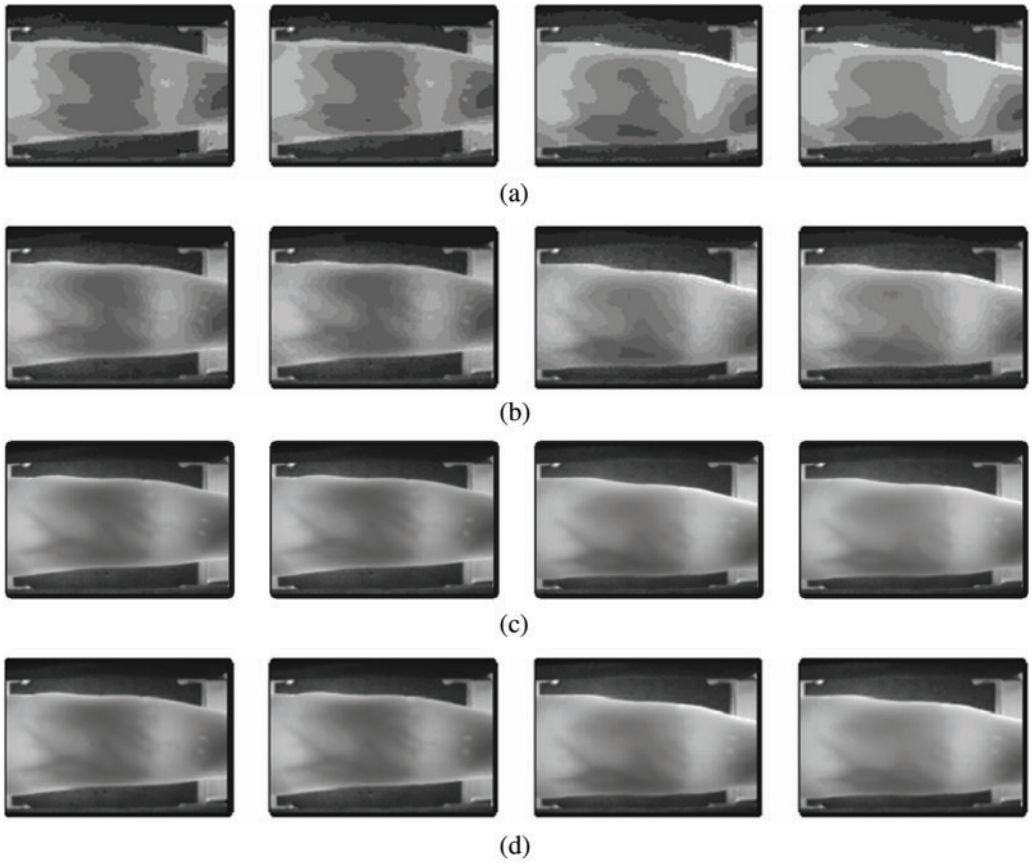


Figure 9: Sample vein images from the used data set. Reprinted with permission from [37]. © 2024 Dr. Bakhtiar Affendi Rosdi

To generate a signature, the vein image was chosen, and applied the following steps:

Step 1: The most important practical results of the proposed system in image compression will be explained. The results of image compression were excellent using the K-Mean algorithm. The experiment was carried out using several cases to test the compression algorithm by changing the K value through which the number of colors that should remain in the image is determined. It can also be seen that the values of affect the size of the image after compression. Compress the image using Algorithm 1. Fig. 10 shows the image compression results using different K values.



(a)

(b)

(c)

(d)

Figure 10: Image compression results (a) K value 16 (b) K value 32 (c) K value 64 (d) K value 128

Note that the clarity of the image and the important parts and the value of their compression were good when using the value of K is 64. Table 1 illustrates compressed results with K = 64 for different images.

Table 1: Compress results for different image sizes

No.	Original image size	Compressed image size
1	225 KB	51.1 KB
2	520 KB	119.6 KB
3	1 MB	235.62 KB
4	2 MB	465.5 KB
5	4 MB	931 KB
6	8 MB	1.8 MB

Step 2: Extract image features using Algorithm 2. Fig. 11 shows the HOG feature extraction results.

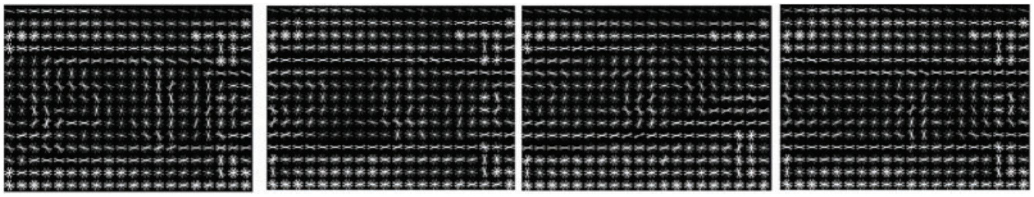


Figure 11: HOG feature extraction results

Step 3: Calculate the LSTM output from the HOG features using Algorithm 3. Table 2 shows sample output values from the LSTM network after processing the HOG features extracted from the vein images. These numerical output values from the LSTM network after processing the HOG features extracted from the vein images. The values in the range of 0 to 1, which is the outputs of LSTM networks when using functions. These values represent the learned representations or encodings of the input data (HOG features) by the LSTM network.

Table 2: LSTM output results

Sample	Sample 1	Sample 2	Sample 3	Sample 4
LSTM results	0.12411547	0.17926845	0.31956965	0.12048425
	0.05998326	0.15905383	0.01884169	0.04905463
	0.24121725	0.21212033	0.05873325	0.24756899
	0.14955842	0.11671609	0.25024057	0.08627665
	0.24121725	0.24121725	0.31956965	0.25014656
	0.13717021	0.09800187	0.17420914	0.12582455
	0.15773029	0.24121725	0.14221221	0.17239371
	0.01789027	0.18367915......	0.31956925	0.02699074 ......

Step 4: Generate a signature using Algorithm 4. Table 3 shows the signatures generated for each vein image. The coordinates $(x_0.y_0)$. $(x_1.y_1)$. $(x_2.y_2)$. $(x_3.y_3)$ of four points used to generate the Bezier curve signature for each vein image, calculated using the parametric equations from Algorithm 4:

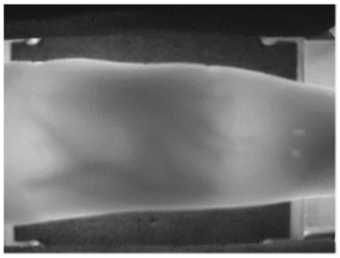
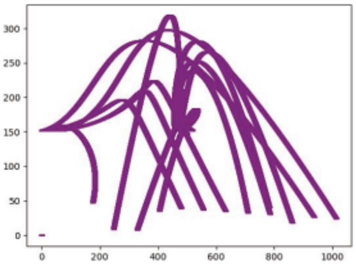
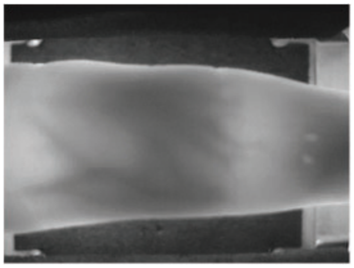
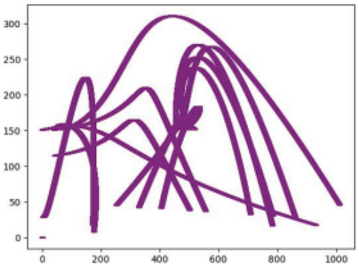
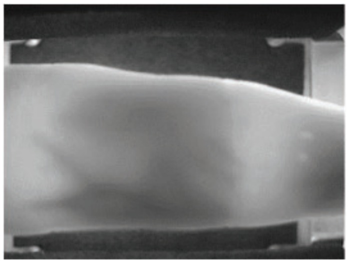
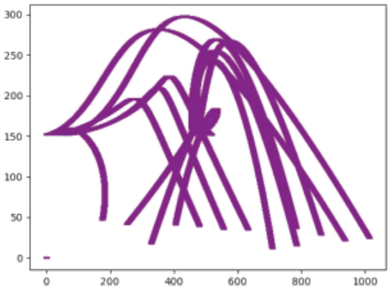
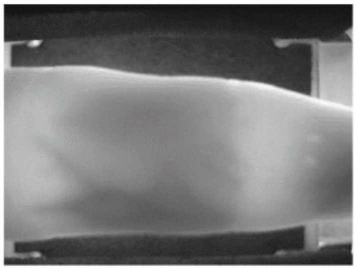
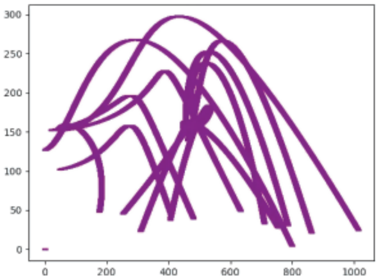
$$P_0 = (x_1 + xt\cos(t) \cdot xt \times \cos(t))$$

$$P_1 = (xt \cdot xt \times t)$$

$$P_2 = (\cos(xt) \times \mathrm{t}.\sin(xt) \times t)$$

$$P_3 = (xt \times t.xt)$$

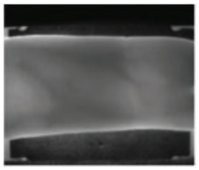
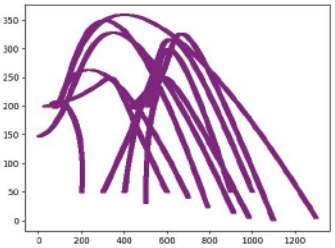
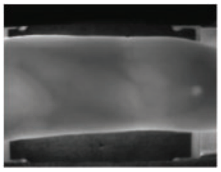
Table 3: Signatures generated for different images

Vein Image	Signature
	
	
	
	

Where *xt* is the LSTM output value and t is the index of that value.

Step 5: There are many factors of an image to be considered when determining the strength of a system in extracting hidden data from the image, including the addition of noise or rotation to the image. Table 4 illustrates the results of the hiding step.

Table 4: Signature hiding and retrieval results

Vein image	Signature	Steganographic vein image	Parameter type	Value	Extracted signature
			Gaussian noise	0.1	True
				0.2	True
				0.3	False
			Rotate	0.5°	True
				1°	True
				1.5°	True
				2°	False

The proposed system was able to retrieve the hidden signature when adding two types of attacks: Gaussian noise and rotation. The intensity of Gaussian noise ranged up to 0.3, while the image was rotated up to 2°. This result indicated a higher percentage than that reported in previous work. The results are detailed in Table 5 which compares the robustness evaluations by scrutinizing the normalized cross-correlation coefficient (NCC).

Table 5: Comparison with previous works

Reference	Type of attack	Metric value	Image name	NCC
Ref. [38]	Gaussian noise	0 to 0.002	Image 1	0.9673
			Image 2	0.9453
			Image 3	0.9613
			Image 4	0.9754
	Rotation	0° to 2°	Image 1	0.9956
			Image 2	0.9936
			Image 3	0.9928
			Image 4	0.9934
Ref. [39]	Gaussian noise	0 to 0.002	Image 1	0.9567
			Image 2	0.9368
			Image 3	0.9614
			Image 4	0.9544
	Rotation	0° to 2°	Image 1	0.9874
			Image 2	0.9747
			Image 3	0.9567
			Image 4	0.9774
Ref. [40]	Gaussian noise	0 to 0.002	Image 1	0.9831
			Image 2	0.9748
			Image 3	0.9735
			Image 4	0.9788
	Rotation	0° to 2°	Image 1	0.9834
			Image 2	0.9829

(Continued)

Table 5 (continued)

Reference	Type of attack	Metric value	Image name	NCC
			Image 3	0.9811
			Image 4	0.9847
Proposed work	Gaussian noise	0 to 0.29	Image 1	0.9884
			Image 2	0.9879
			Image 3	0.9914
			Image 4	0.9902
	Rotation	0° to 2°	Image 1	0.9985
			Image 2	0.9974
			Image 3	0.9998
			Image 4	0.9967

The proposed system effectively combines adaptive image compression using K-Means clustering, discriminative feature extraction via HOG, increased feature randomness through LSTM networks, unique signature generation using parametric Bezier curves, and robust signature hiding by replacing the least significant bits in the compressed image at LSTM-selected locations. The deep learning-based approach shows high resilience against common image distortions like Gaussian noise up to 0.29 intensity and rotation up to 2 degrees, outperforming some previous methods. Using HOG for extracting distinctive features and LSTM for sequential modeling helps generate highly complex yet content-dependent signatures that are resistant to content-preserving modifications.

6 Conclusions

Correspondence systems are one of the most important types of systems at present, considering the rapid development of the Internet and electronic correspondence. In this paper, a new system to secure images sent through the Internet was assessed, which was constructed using a series of algorithms and techniques. Image processing algorithms, deep learning, and curve drawing approaches were combined to build a security system. Ultimately, the archived results demonstrated that the use of the HOG technique for feature extraction was effective and that the use of a deep learning technique increased the model's randomness. However, the proposed system secures a limited type of images, including biometric images and medical images. This requires the use of other algorithms specifically for more feature-extracting image characteristics and algorithms that achieve the best features. An image can also remain safe if it is exposed to Gaussian-type noise in the range of 0–0.29 or rotation 0 to 2 degrees. If the image is exposed to noise greater than this range, the hidden signature will be lost. Moreover, the power to generate a signature was further enhanced using a Bezier curve, which is already considered to be highly random. The system was applied to 2952 vein images, achieving an error rate of 0.001 and yielding an error in only four pictures.

Acknowledgement: The authors would like to extend their appreciation to the University of Sfax for supporting this research.

Funding Statement: The authors confirm that this study did not receive any funding.

Author Contributions: Draft manuscript preparation, design, data collection, analysis, and interpretation of results: Ahmed H. Alhadethi; Ikram Smaoui and Saad M. Darwish reviewed the results and Ahmed Fakhfakh approved the final version of the manuscript.

Availability of Data and Materials: The dataset used in this article is available at http://drfendi.com/fv_usm_database/ (Last Visit 20/7/2018). Additional materials can be requested from the first author.

Conflicts of Interest: The authors declare that they have no conflicts of interest in this manuscript.

References

[1] S. N. Atluri and S. Shen, "Global weak forms, weighted residuals, finite elements, boundary elements & local weak forms," *The Meshless Local Petrov-Galerkin (MLPG) Method*, vol. 1, pp. 15–64, 2004.

[2] S. Ilan, "Telecommunications and travel relationships: A review," *Transp. Res. Part A: General*, vol. 20, no. 3, pp. 223–238, 1986. doi: 10.1016/0191-2607(86)90096-8.

[3] S. Thakur, "Fifth generation (5G) wireless technology," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 9, no. 4, pp. 909–912, 2021. doi: 10.22214/ijraset.2021.33792.

[4] B. H. Taher, H. Liu, F. Abedi, H. Lu, A. A. Yassin and A. J. Mohammed, "A secure and lightweight three-factor remote user authentication protocol for future IoT applications," *J. Sens.*, vol. 2021, no. 7, pp. 1–18, 2021. doi: 10.1155/2021/8871204.

[5] N. F. Abdulsattar *et al.*, "Botnet detection employing a dilated convolutional autoencoder classifier with the aid of hybrid shark and bear smell optimization algorithm-based feature selection in FANETs," *Big Data Cogn. Comput.*, vol. 6, no. 4, pp. 112, 2022. doi: 10.3390/bdcc6040112.

[6] Z. A. Zukarnain, A. Muneer, and M. K. Ab Aziz, "Authentication securing methods for mobile identity: Issues, solutions and challenges," *Symmetry*, vol. 14, no. 4, pp. 821, Apr. 2022. doi: 10.3390/sym14040821.

[7] A. Mahmood, T. Hamed, C. Obimbo, and R. Dony, "Improving the security of the medical images," *Int. J. Adv. Comput. Sci. Appl.*, vol. 4, no. 9, 2013. doi: 10.14569/issn.2156-5570.

[8] G. Alvarez, S. Li, and L. Hernandez, "Analysis of security problems in a medical image encryption system," *Comput. Biol. Med.*, vol. 37, no. 3, pp. 424–427, 2007. doi: 10.1016/j.compbiomed.2006.04.002.

[9] J. Zhang, S. Zhang, H. Wang, Y. Li, and R. Lu, "Image compression network structure based on multiscale region of interest attention network," *Remote Sens.*, vol. 15, no. 2, pp. 522, 2023. doi: 10.3390/rs15020522.

[10] N. Mital, E. Ozyilkan, A. Garjani, and D. Gunduz, "Neural distributed image compression with cross-attention feature alignment," in *Proc. 2023 IEEE Winter Conf. Appl. Comput. Vis.*, Waikoloa, HI, USA, 2023, pp. 2497–2506. doi: 10.1109/WACV56688.2023.00253.

[11] Z. Li, B. Li, K. W. Eliceiri, and V. Narayanan, "Computationally efficient adaptive decompression for whole slide image processing," *Biomed. Opt. Express*, vol. 14, no. 2, pp. 667, 2023. doi: 10.1364/BOE.477515.

[12] X. Zhou, W. Hong, T. S. Chen, and G. Yang, "Reversible demosaiced image authentication scheme with recoverability using clustering and matching techniques," *J. Inf. Secur. Appl.*, vol. 73, no. 8–9, pp. 103425, 2023. doi: 10.1016/j.jisa.2023.103425.

[13] X. Yan, L. Li, J. Chen, and L. Sun, "Public key based bidirectional shadow image authentication without pixel expansion in image secret sharing," *Front. Inform. Technol. Electron. Eng.*, vol. 24, no. 1, pp. 88–103, 2023. doi: 10.1631/FITEE.2200118.

[14] H. Xing, H. Che, Q. Wu, and H. Wang, "Image perceptual hashing for content authentication based on Watson's visual model and LLE," *J. Real Time Image Process.*, vol. 20, no. 1, pp. 507, 2023. doi: 10.1007/s11554-023-01269-9.

[15] A. Guarino, D. Malandrino, R. Zaccagnino, C. Capo, and N. Lettieri, "Touchscreen gestures as images. A transfer learning approach for soft biometric traits recognition," *Expert. Syst. Appl.*, vol. 219, no. 9, pp. 119614, 2023. doi: 10.1016/j.eswa.2023.119614.

[16] S. Arora, R. Mittal, H. Kukreja, and M. P. S. Bhatia, “An evaluation of denoising techniques and classification of biometric images based on deep learning,” *Multimed. Tools Appl.*, vol. 82, no. 6, pp. 8287–8302, 2023. doi: 10.1007/s11042-021-11573-w.

[17] M. R. Abuturab, “Optical single-channel security system using 3D-logistic map biometric keys for multiple color images,” *Opt. Quantum Electron.*, vol. 55, no. 3, pp. 152, 2023. doi: 10.1007/s11082-022-04493-y.

[18] M. B. Er, E. Isik, and I. Isik, “Parkinson’s detection based on combined CNN and LSTM using enhanced speech signals with variational mode decomposition,” *Biomed. Signal Process. Control*, vol. 70, no. 1, pp. 103006, 2021. doi: 10.1016/j.bspc.2021.103006.

[19] M. Rayhan Ahmed, S. Islam, A. K. M. Muzahidul Islam, and S. Shatabda, “An ensemble 1D-CNN-LSTM-GRU model with data augmentation for speech emotion recognition,” *Expert. Syst. Appl.*, vol. 218, no. 4, pp. 119633, May 2023. doi: 10.1016/j.eswa.2023.119633.

[20] A. G. Grizhebovskaya, A. V. Mikhalev, and L. P. Dmitrieva, “Geometric distortion correction of images received from biometric fingerprint devices,” *J. Math. Sci.*, vol. 269, no. 3, pp. 317–321, 2023. doi: 10.1007/s10958-023-06283-7.

[21] X. Wan, “Application of K-means algorithm in image compression,” *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 563, no. 5, pp. 052042, 2019. doi: 10.1088/1757-899X/563/5/052042.

[22] M. K. Islam, M. S. Ali, M. S. Miah, M. M. Rahman, M. S. Alam and M. A. Hossain, “Brain tumor detection in MR image using superpixels, principal component analysis and template based K-means clustering algorithm,” *Mach. Learn. Appl.*, vol. 5, no. 11, pp. 100044, 2021. doi: 10.1016/j.mlwa.2021.100044.

[23] G. Verma and A. Kumar, “Image compression using deep learning based multi-structure feature map and K-means clustering,” *Machine Learning, Image Processing, Network Security and Data Sciences*, 2020, vol. 1240, pp. 365–374. doi: 10.1007/978-981-15-6315-7_30.

[24] T. Kobayashi and N. Otsu, “Image feature extraction using gradient local auto-correlations,” *Computer Vision–ECCV 2008*, 2008, vol. 5302, pp. 346–358. doi: 10.1007/978-3-540-88682-2_27.

[25] T. Kobayashi, “BFO meets HOG: Feature extraction based on histograms of oriented p.d.f. gradients for image classification,” in *2013 IEEE Conf. Comput. Vis. Pattern Recognit.*, Portland, OR, USA, IEEE, Jun. 2013, pp. 747–754. doi: 10.1109/CVPR.2013.102.

[26] S. Routray, A. K. Ray, and C. Mishra, “Analysis of various image feature extraction methods against noisy image: SIFT, SURF and HOG,” in *Proc. 2017 2nd IEEE Int. Conf. Electr., Comput. Commun. Technol (ICECCT)*, Coimbatore, India, 2017. doi: 10.1109/ICECCT.2017.8117846.

[27] A. Choudhury, H. S. Rana, and T. Bhowmik, “Handwritten Bengali numeral recognition using hog based feature extraction algorithm,” in *2018 5th Int. Conf. Signal Process. Integr. Netw. (SPIN)*, Noida, India, 2018, pp. 687–690. doi: 10.1109/SPIN.2018.8474215.

[28] K. Cho *et al.*, “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” in *Proc. 2014 Conf. Empir. Methods Nat. Lang. Process. (EMNLP)*, Stroudsburg, PA, USA, Association for Computational Linguistics, 2014, pp. 1724–1734. doi: 10.3115/v1/D14-1179.

[29] M. Sundermeyer, R. Schlüter, and H. Ney, “LSTM neural networks for language modeling,” in *13th Annu. Conf. Int. Speech Commun. Assoc. 2012, (INTERSPEECH 2012)*, Portland, OR, USA, 2012, vol. 1, pp. 194–197. doi: 10.21437/interspeech.2012-65.

[30] Y. Yu, X. Si, C. Hu, and J. Zhang, “A review of recurrent neural networks: LSTM cells and network architectures,” *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, 2019. doi: 10.1162/neco_a_01199.

[31] R. C. Staudemeyer and E. R. Morris, “Understanding LSTM—A tutorial into long short-term memory recurrent neural networks,” arXiv:1909.09586, 2019.

[32] M. Sundermeyer, H. Ney, and R. Schluter, “From feedforward to recurrent LSTM neural networks for language modeling,” *IEEE Trans. Audio, Speech and Lang. Process.*, vol. 23, no. 3, pp. 517–529, 2015. doi: 10.1109/TASLP.2015.2400218.

[33] S. Maqsood, M. Abbas, K. T. Miura, A. Majeed, and A. Iqbal, “Geometric modeling and applications of generalized blended trigonometric Bézier curves with shape parameters,” *Adv. Differ. Equ.*, vol. 2020, pp. 550, 2020. doi: 10.1186/s13662-020-03001-4.

[34] X. A. Han, Y. Ma, and X. Huang, "A novel generalization of Bézier curve and surface," *J. Comput. Appl. Math.*, vol. 217, no. 1, pp. 180–193, Jul. 2008. doi: 10.1016/j.cam.2007.06.027.

[35] R. Lattarulo, L. González, E. Martí, J. Matute, M. Marcano and J. Pérez, "Urban motion planning framework based on N-Bézier curves considering comfort and safety," *J. Adv. Transport.*, vol. 2018, no. 3, pp. 1–13, 2018. doi: 10.1155/2018/6060924.

[36] Ü. Dinçer and M. Çevik, "Improved trajectory planning of an industrial parallel mechanism by a composite polynomial consisting of Bézier curves and cubic polynomials," *Mech. Mach. Theory*, vol. 132, no. 5, pp. 248–263, 2019. doi: 10.1016/j.mechmachtheory.2018.11.009.

[37] M. S. Mohd Asaari, S. A. Suandi, and B. A. Rosdi, "Fusion of band limited phase only correlation and width centroid contour distance for finger-based biometrics," *Expert. Syst. Appl.*, vol. 41, no. 7, pp. 3367–3382, Jun. 2014. doi: 10.1016/j.eswa.2013.11.033.

[38] S. P. Vaidya, "Fingerprint-based robust medical image watermarking in hybrid transform," *Vis. Comput.*, vol. 39, no. 6, pp. 2245–2260, 2023. doi: 10.1007/s00371-022-02406-4.

[39] B. Huang, Y. Dai, R. Li, D. Tang, and W. Li, "Finger-vein authentication based on wide line detector and pattern normalization," in *Proc. Int. Conf. Pattern Recognit.*, Istanbul, Turkey, 2010, pp. 1269–1272. doi: 10.1109/ICPR.2010.316.

[40] N. Miura, A. Nagasaka, and T. Miyatake, "Extraction of finger-vein patterns using maximum curvature points in image profiles," in *Proc. 9th IAPR Conf. Mach. Vis. Appl.*, Tsukuba Science City, Japan, 2005, pp. 347–350.