

Offline signature verification using long short-term memory and histogram orientation gradient

Fadi Mohammad Alsuhiat, Fatma Susilawati Mohamad

Faculty of Informatics and Computing, Computer Science, Universiti Sultan Zainal Abidin, Kuala Terengganu, Malaysia

Article Info

Article history:

Received Apr 30, 2022

Revised Sep 5, 2022

Accepted Oct 11, 2022

Keywords:

CEDAR

HOG

LSTM

Offline signature verification

USTig

ABSTRACT

The signing process is a critical step that organizations take to ensure the confidentiality of their data and to safeguard it against unauthorized penetration or access. Within the last decade, offline handwritten signature research has grown in popularity as a common method for human authentication via biometric features. It is not an easy task, despite the importance of this method; the struggle in such a system stem from the inability of any individual to sign the same signature each and every time. Additionally, we are indeed interested in the dataset's features that could affect the model's performance; thus, from extracted features from the signature images using the histogram orientation gradient (HOG) technique. In this paper, we suggested a long short-term memory (LSTM) neural network model for signature verification, with input data from the USTig and CEDAR datasets. Our model's predictive ability is quite outstanding: The classification accuracy efficiency LSTM for USTig was 92.4% with a run-time of 1.67 seconds and 87.7% for CEDAR with a run-time of 2.98 seconds. Our proposed method outperforms other offline signature verification approaches such as K-nearest neighbour (KNN), support vector machine (SVM), convolution neural network (CNN), speeded-up robust features (SURF), and Harris in terms of accuracy.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Fadi Mohammad Alsuhiat

Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin

Besut Campus, 22200 Besut, Terengganu, Malaysia

Email: karaklove@yahoo.com

1. INTRODUCTION

At the present time and with the technical means of developing innovation, data protection is considered one of the most critical topics in order to maintain a strategic distance from fraud and extortion within the information. Despite the fact that there are numerous systems for securing data, such as facial recognition, iris recognition, and speech, in compared to other models, signature verification is regarded as one of the most important biometric techniques for information verification [1]. Signature verification is regarded as an important process for safeguarding systems and information against unauthorized access or penetration. Furthermore, signature verification is appropriate for all types of institutions in order to ensure the confidentiality of information because it is simple to implement, does not require a high cost, and has the ability to protect systems and information by distinguishing between original and fraudulent signatures [2].

A handwritten signature is a unique talent made up of symbols and letters written in a certain language. Signature is one of the techniques used to provide individuals with authentication. A handwritten signature is a unique skill comprised of symbols and characters written in a certain language. Signing is one of the methods used to provide authentication to individuals in order for them to perform a variety of

activities, such as bank transactions and school attendance, where a signature may ensure the permitted validity of people and distinguish a fake signature from an original signature [3].

The problem with signature verification systems is that the signature is not a specific shape or picture, but rather an unusual drawing that an individual creates as a reference indicating his personality, and the signature may include letters, numbers, symbols, and shapes. Individuals use signatures because they want to perform certain transactions that require them to prove their personalities through their signatures, such as banking transactions or a legal instrument, so signature issues arise as a result of someone attempting to copy or counterfeit someone else's signature [4]. Signature verification systems employ handwritten signatures to validate an individual's identification, and signature verification systems are the most socially and legally acceptable form of identifying persons and the degree of authority entrusted to them [5]. Additionally, Alsuhiat and Mohamad [6] noted that the signature verification technique is a simple and effective means of identifying between a genuine and a counterfeit mark.

The process of identifying signatures or other biometric features are critical in all aspects of life. Given that security and ambiguity in general, and information security in particular, are the primary concerns of individuals and nations, the use of signature verification systems significantly aids in identifying individuals and authorizing them to perform specific tasks. Furthermore, signature verification has various advantages since it is a socially acceptable approach of ensuring information security, as well as the most secure method used in credit card and bank transactions. Additionally, when compared to other biometric and non-biometric systems, the signature verification system is considered to be more efficient, as the user can easily change his/her signature, whereas the face or iris patterns cannot be changed [7]. The Figures 1(a) and 1(b) depicts various signature patterns for the same individual.

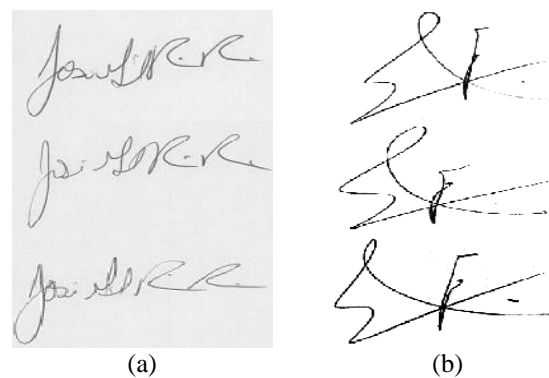


Figure 1. An example of several signature patterns (a) example of CEDAR dataset and (b) example of UTSig dataset)

A handwritten signature is one of an individual's biometric features. This signature consists of a collection of letters, shapes, or symbols, or all of them, drawn in a specific order. As a result, a system for identifying and verifying signatures is required in order to distinguish genuine from forged signatures [8]. Signature verification may be a difficult design recognizable proof with inadequacy due to the fact that no two people's signatures can be identical. As a consequence, ensuring signature authenticity contributes to the protection of users and information from harm or loss, making signature verification a critical and efficient system in all fields [9].

Signature verification systems need the execution of several processes, the most essential of which are feature extraction and classification, since these two stages are vital for validating signatures and distinguishing between authentic and faked signatures [10]. In order to boost the process of identifying between genuine and fraudulent markings, the features extraction stage focuses on recognising picture highlights with exceptional precision by decreasing the measurements of the initial image and then extricating a collection of hidden characteristics inside the image [11]. The signature verification stage in the signature verification system determines if the signature is fake or authentic by comparing the signature attributes provided in the database with anybody who desires to verify his/her signature [5]. To identify the true signature, the classification phase analyses the enrolled and authenticated signature attributes. Based on the threshold, the decision-maker determines whether the signature must be accepted or denied [12].

Although the handwriting signature feature is considered one of the most important types of biometrics, and it is used widely and in many areas of life as one of the most common and safe methods of

protecting systems and information, this system still faces some challenges and obstacles that require further studies and research on it, in order to develop a signature verification system that is able to distinguish between original and fraudulent signatures efficiently and effectively. Furthermore, the signature is a behavioural characteristic of people that is used in the field of biometric structures to confirm people's identities, and with the growing use of biometric highlights in the field of security, the signature appears as a biometric include that provides a secure means of designating people and ensuring their personality in legitimate reports. Furthermore, in the field of biometric systems, people are more tolerant of this property than other biometric qualities, such as (hand geometry, iris scan, or deoxyribonucleic acid (DNA)). All of these factors have contributed to an increase in the number of signature verification systems available on the market, as well as the need for future improvement.

The purpose of this research is to look at the feature extraction and classification phases of signature image processing. As a consequence, we suggested a new model that uses the UTSig and CEDAR datasets to combine histogram orientation gradient (HOG) as a features extraction approach with the long short-term memory (LSTM) neural network model. In this study, the first most significant advance is to define the best block size of the HOG algorithm, where the block size affects the representation of the signature image, thus affecting the features extraction process and achieving non-accurate results. In addition, the second most significant advance of this study is to use recurrent neural network (RNN) nodes to classify the signature images by using HOG results as input for the LSTM algorithm. Finally, the third most significant advance of this study is to evaluate the proposed method with two different datasets to ensure its validity.

2. METHOD

This section provides an explanation of the methodology and processes used in this investigation. Section 1 goes into great depth on the feature extraction approach. The second portion depicts the categorization method, and the third section discusses the database that was used. This article describes how the proposed method was carried out in a series of steps as seen in Figure 2.

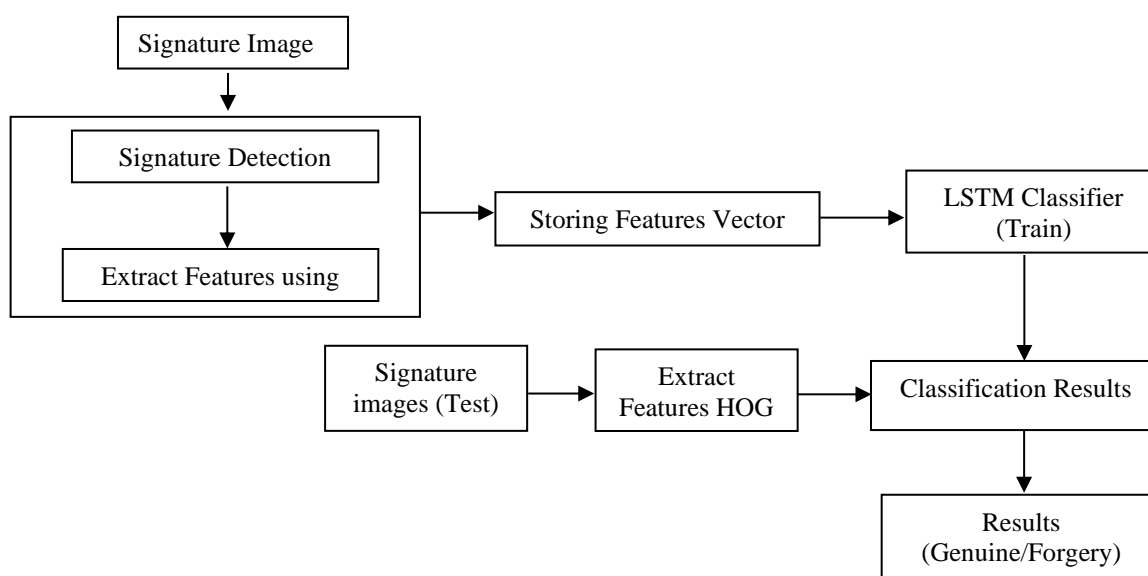


Figure 2. Flow chart of the suggested signature verification model

2.1. Features extraction stage

In this work, a HOG approach was employed for offline signature verification. "Dalal and Triggs [13] proposed the HOG for trait shape representation at the computer vision and pattern recognition (CVPR) conference in 2005." The HOG is mostly used for human detection. In this work, HOG was employed as a feature extraction technique to detect and recognise the signature image. The following Figure 3 shows how the HOG algorithm works.

In theory, the HOG descriptor technique counts angle introduction occurrences in localised chunks of an image or region of interest (ROI). The main use of the HOG descriptor, as seen in Figure 2, is as follows: to begin, the image is split into small-related areas (cells), and for each locale, a histogram of angle

directions or edge orientations for the pixels within the cell is constructed, and the resultant gradient orientation is used. Subsequently, each cell is discretized into precise containers; each cell's pixel then supplies a weighted angle to its appropriate precise canister; and last, neighbouring cells are grouped into pieces inside the spatial region. This establishes the framework for histogram collection and normalisation; finally, the normalised collection of histograms becomes the piece histogram, and the collection of these square histograms represents the descriptor [14].

The HOG is specified in this study as having a block size of $[4 \times 4]$ pixels. As a result, the total length of the feature vector utilised to describe each signature picture sample is 34,596. Figure 3 depicts two offline handwritten signature datasets with varying cell sizes that were utilised in this study and were examined to showcase the HOG implementation on the offline signature. According to Abbas *et al.* [14], the number of depicted gradients and directions is more visible when the cell size is small than when the cell size is big. The directions and gradient will be lowered gradually when the cell size number of the HOG parameter is increased. Figure 4 shows the effects of HOG on offline signature images with 4-cell sizes.

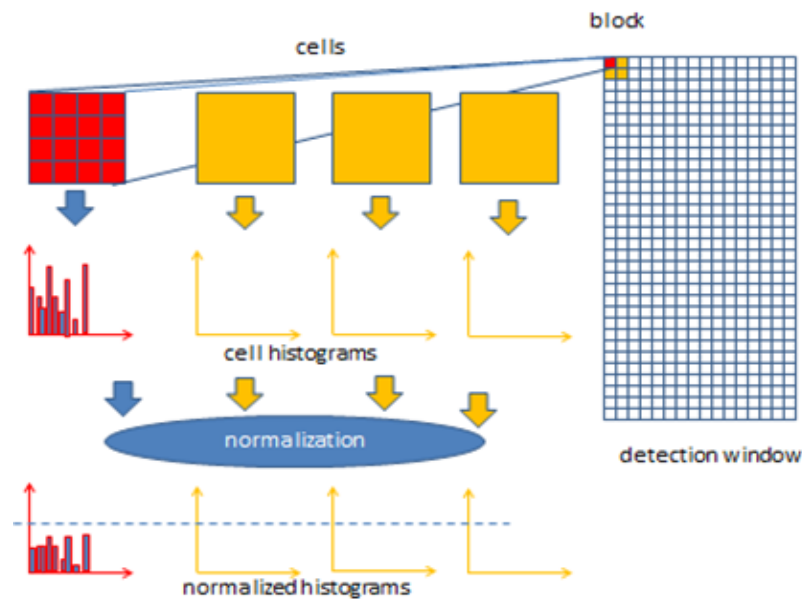


Figure 3. Demonstrates the implementation of the HOG algorithm

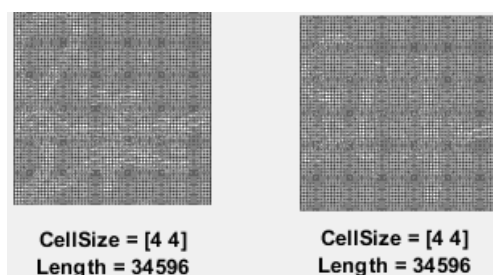


Figure 4. 4-set cell size HOG implementation

2.1.1. Preprocess the data

Most researchers are already familiar with this phase. Preprocessing data is a critical component in any machine learning research, especially when working with images. To reduce the width-to-height ratio to 1:2, the image must be preprocessed. Ideally, the image should be 64×128 pixels in size.

2.1.2. Calculating gradients

The gradient is then determined for each pixel in the image. Gradients are minor x and y axis variations. Subtract the value on the left from the pixel value on the right to determine the gradient (or

change) in the x-direction. To compute the gradient in the y direction, subtract the pixel value immediately below the selected pixel from the pixel value immediately above it. As a consequence, we utilised the below to determine the gradients of the pixels in the x and y axes:

$$G_x = LV - RV \quad (1)$$

$$G_y = BV - AV \quad (2)$$

This method will produce two new matrices, one holding gradients in the x direction and the other in the y direction. This is equivalent to utilising a size 1 sobel kernel. Where G_x is change in X direction, G_y is change in Y direction, LV is value on the left from the pixel, RV is value on the right from the pixel, BV is value below the chosen pixel, and AV is value above the chosen pixel.

2.1.3. Calculate the magnitude and orientation

We will now compute the magnitude and direction of each pixel value using the gradients we determined in the previous step. For this stage, we shall apply Pythagoras' theorem. Here, the gradients are the base and perpendicular. So, using Pythagoras' theorem, we can compute the entire gradient magnitude as (3):

$$TGM = \sqrt{[(G_x)^2 + (G_y)^2]} \quad (3)$$

where TGM is total gradient magnitude, G_x is change in X direction, and G_y is change in Y direction. Then, for the same pixel, compute the orientation (or direction). We already know how to write the tan for the angles:

$$\tan(\Phi) = G_y / G_x \quad (4)$$

$$\Phi = \text{atan}(G_y / G_x) \quad (5)$$

2.2. Signature image classification

In this study, we use a LSTM network to categorise a signature dataset. The LSTM structure is a deep learning artificial RNN. Furthermore, unlike ordinary feed forward neural networks, LSTM incorporates feedback connections, allowing it to prepare both single bits of information (such as images) and whole information groupings (such as speech or video) [15].

Since its inception in 1995, a multi variant of the LSTM structure for RNN has been proposed. Over time, these systems have evolved into state-of-the-art models for a variety of machine learning issues. This has rekindled interest in determining the utility and role of various computational components of LSTM typical variants [16].

LSTMs are RNN nodes that are specially designed to maintain long-term conditions. They are made up of a self-connected memory cell, similar to a classical RNN node, and three gates that control the hub's yield and input. Each gate could be a sigmoid function of the LSTM hub's input. The primary door is an input door that controls whether new input for the hub is available. The moment door could be a disregard entryway, allowing the hub to reset the memory cell's activation values. The final entryway is a yield entryway that controls which parts of the cell yield are accessible to the other nodes [17]. Figure 5 show the structure of the LSTM.

As a result, we employed LSTM layers in our study to understand the long-term interdependence of signature strokes. Three gates and a cell are included in LSTM units. The LSTM operational are as follows:

$$f_t = \text{sig}(W_f \cdot [\text{hidden}_{t-1}, X_t] + \text{bias}_f) \quad (6)$$

$$i_t = \text{sig}(W_i \cdot [\text{hidden}_{t-1}, X_t] + \text{bias}_i) \quad (7)$$

$$\tilde{C}_t = \tanh(W_c \cdot [\text{hidden}_{t-1}, X_t] + \text{bias}_c) \quad (8)$$

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \quad (10)$$

$$O_t = \text{sig}(W_o \cdot [\text{hidden}_{t-1}, X_t] + \text{bias}_o) \quad (11)$$

$$h_t = O_t \times \tan h(C_t) \quad (12)$$

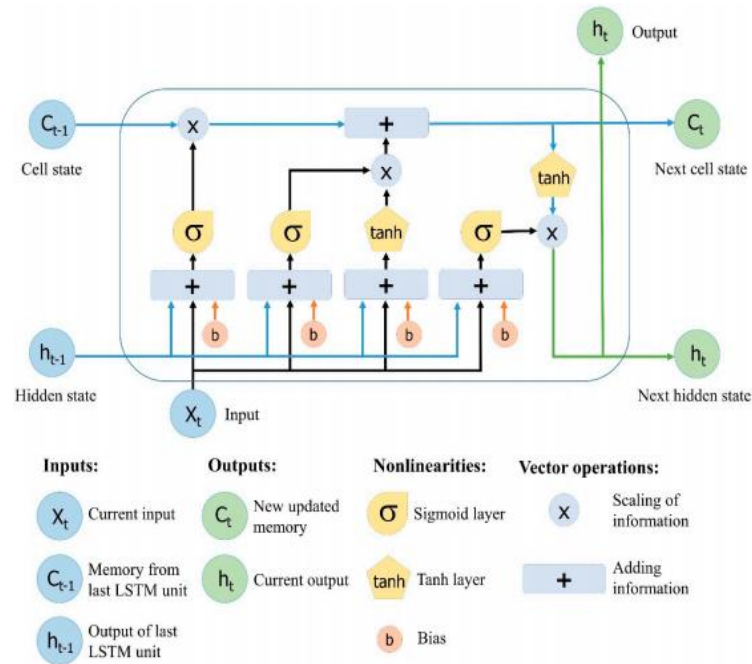


Figure 5. The structure of the LSTM neural network. Reproduced from Yan [18]

2.3. Signature dataset

The comparison of four algorithms using images of signatures from the (UTSig) and CEDAR datasets. As demonstrated in Figure 6, the UTSig dataset includes “(115) classes containing: (27) genuine signatures; (3) opposite-hand forgeries; (36) simple forgeries; and (6) skill forgeries. Each class is associated with a single authentic individual. UTSig contains (8,280) images of signatures collected from undergraduate and graduate students at the University of Tehran and Sharif University of Technology; signatures were scanned at a resolution of 600 dpi and stored as 8-bit Tiff files” [19].

In this paper, for the UTSig dataset, a total of (1,350) signature images were chosen to train the set, which included 50 people for each person (27) genuine signature and (6) skilled forgery signature, we prefer skilled forgery signature because it is more difficult than other forgery types, and (300) signature images were chosen to test our classification algorithm. CEDAR data set “consists of the signatures of 55 signers from various professional and cultural backgrounds. Each of these signers signed 24 genuine signatures 20 minutes apart. Each forger attempted to imitate the signatures of 3 people 8 times in order to produce 24 forged signatures for each genuine signer. As a result, the dataset contains $55 \times 24 = 1,320$ genuine signatures and 1,320 forged signatures” [20].

In this paper, we used the CEDAR dataset to train our classification algorithms on a total of (1,200) signature images and (400) signature images to test our classification algorithms. Figure 6 shows examples of forger and genuine signature. Figure 6(a) is genuine signature image from both dataset CEDAR and UTSig and Figure 6(b) is forged signature image from both dataset CEDAR and UTSig. In this study, the original and forged signatures of the first 50 people were selected from UTSig database, while the original signatures of the first 50 people, and 8 forged signatures for each of them, were selected from CEDAR database, and Table 1 shows the number of images of signatures selected from each database.



Figure 6. Forger and genuine signature examples from UTSig and CEDAR dataset (a) is genuine signature image from both dataset CEDAR and UTSig and (b) is forged signature image from both dataset CEDAR and UTSig

Table 1. The statistical differences between testing and training sets

Sets	UTSig	CEDAR
Training	1,350	1,200
Test	300	400
Total	1,650	1,600

3. RESULTS AND DISCUSSION

This section presents the classification results of our classifiers. This section consists of two parts, part (3.1) explains the implementation process, begin with selected the number of signature images, then features extraction process, and finally the classification process. While section (3.2) summarizes the classifiers' accuracy, performance, and compare it with other classifiers.

3.1. Experimental setup

A HOG technique was used to extract features, which were subsequently categorised in original-forgeries using an LSTM classifier. The first model for the UTSig dataset was trained using a set of signatures for (50) individuals, each with 33 signatures, 27 genuine and 6 forgeries, whereas the second model for the CEDAR dataset was trained using a set of signatures for (50) individuals, each with 32 signatures, 24 genuine, and 8 forgeries. We recorded the results of extracting features for each individual signature picture using the HOG technique in a vector that comprised both features and labels.

In the LSTM stage we load the vector with both features and labels and identify them as input for LSTM, then we get the sequence length for each observation and sort the data by their sequence length, after that we divide the training data evenly by using a mini-batch size of 27 in order to reduce the amount of padding in the mini-batch. In addition, we define the LSTM network architecture by identifying the input size, number of hidden units, and class number. Figure 7 depicts the training of an LSTM network for the UTSig dataset, whereas Figure 8 illustrates the training of an LSTM network for the CEDAR dataset.

3.2. Efficiency

The efficiency was determined by the time required to run each algorithm and the accuracy obtained when each algorithm was run on (300) signature images from the UTSig dataset and (400) signature images from the CEDAR dataset. Table 2 displays the run-time and accuracy of classification algorithms. The results of our experiment are summarized in Table 2, which includes the run-time and accuracy of each classifier. We discovered that our proposed model achieves a high level of accuracy and run-time on both UTSig and CEDAR datasets, with an LSTM accuracy of 92% and a run-time of 1.67 seconds for UTSig dataset and 76% and a run-time of 20.3 seconds for CEDAR dataset. Table 3 shows the result of compression process between our proposed method and some other methods for offline signature verification based on accuracy result for each method.

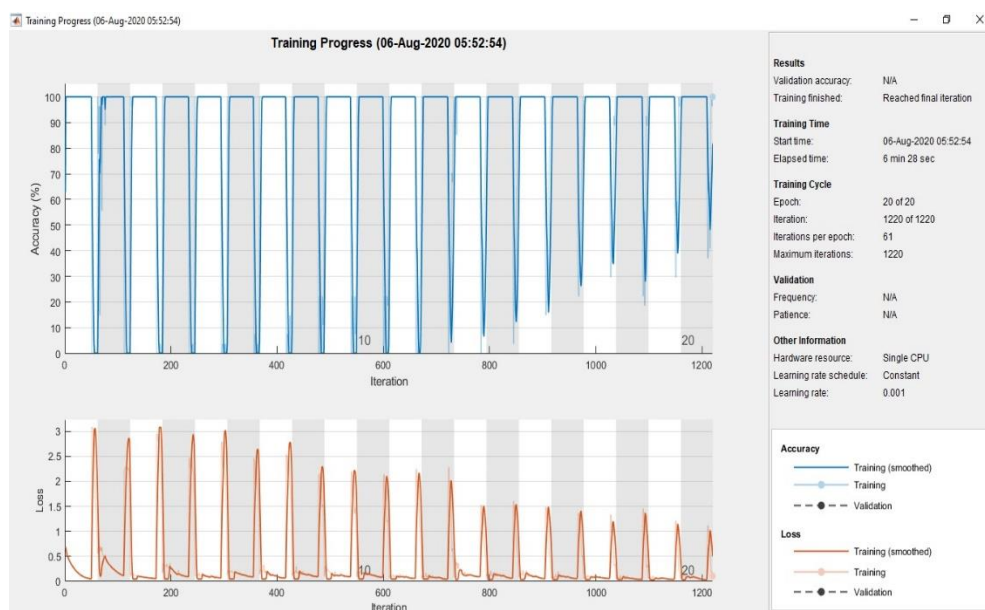


Figure 7. Train LSTM network for UTSig dataset

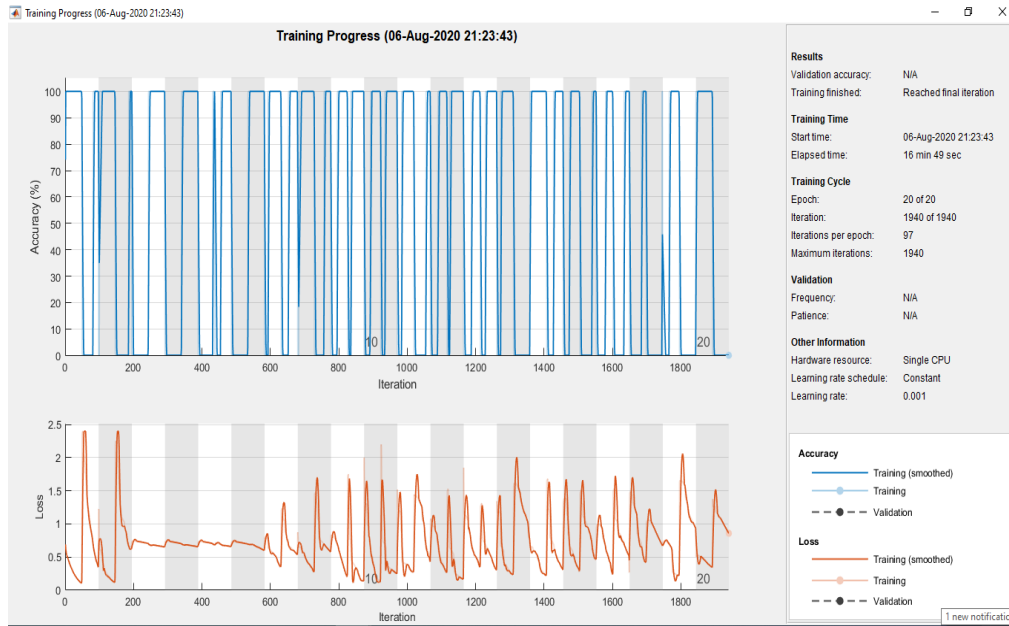


Figure 8. Train LSTM network for CEDAR dataset

Table 2. Run-time values and accuracy for each classifier

Method	Run-time		Accuracy	
	UTSig dataset	CEDAR dataset	UTSig dataset (%)	CEDAR dataset (%)
LSTM	1.67	2.98	92.4	87.7

Table 3. Results of comparing our proposed method with other methods

Methods	Algorithms used	Accuracy (%)
[21]	Convolution neural network (CNN), speeded-up robust features (SURF), and Harris	89
[22]	K-nearest neighbour (KNN), Support vector machine (SVM)	78.5
[23]	Gaussian empirical rule	91.2
[24]	Probabilistic neural network	92.06
[25]	Multilayer perceptron and SVN	91.67
	Proposed methods	92.4

Also, we use false acceptance rate (FAR), false rejection rate (FRR), and equal error rate (EER) to evaluate the performance of the proposed system. EER is calculated as the value at which FAR and FRR are equal. The EER is the best and most accepted single explanation of a verification algorithm's error rate, and the lower the EER, the lower the algorithm's mistake rate. The strategy with the lowest ERR is thought to be the most exact. As a consequence, the findings reported in Table 4 reveal that our method proved to be the most efficient way for accurately verifying signature features of offline handwritten signatures.

Table 4. Results of comparing our proposed method with other methods

Methods	Algorithms used	FAR (%)	FRR (%)	ERR (%)
[26]	Discrete radon transforms (DRT) features and HOG	15.08	22.76	20.94
[24]	WP entropy neural network system (WPENN)	16.1	16.2	16.5
[27]	Global features for offline systems	17.25	17.26	17.25
	Proposed methods	12.68	10.12	11.40

4. CONCLUSION

This research proposed a new model for signature verification by using HOG algorithm for features extraction from signature images, then the extracted features save into vector and classified into two classes genuine or forgery using LSTM, the UTSig dataset has (8,280) signature images divided into (115) classes where and each class refers to one person, and each person has four types of signature, (27) genuine

signatures; (3) opposite-hand forgeries, (36) simple forgeries and (6) skill forgeries. Additionally, we select the optimal cell size (4×4) for the HOG feature extraction algorithm by comparing four different cell sizes in order to determine the optimal number of extracted features. The experimental results indicated that our proposed model performed quite well in terms of performance and predictive ability, achieving an accuracy of 92%, which is considered a high value, especially given that we tested skilled forged signatures, which are harder to identify than other types of forged signatures such as (simple or opposite-hand), as skilled forged signatures are frequently quite similar to the original signatures. In the future, we expect that optimizing the feature extraction phase will increase the performance and predictive ability of signature verification. As a result, utilizing a deep learning algorithm such as CNN for feature extraction and combining it with LSTM will improve the signature verification model.

REFERENCES




- [1] M. Narayana, L. B. Annapurna, and K. Mounika, "Offline signature verification," *International Journal of Electronics and Communication Engineering and Technology (IJECET)*, vol. 8, no. 2, pp. 120–128, 2017.
- [2] F. M. Alsuhimat, F. S. Mohamad, and M. Iqtait, "Detection and extraction features for signatures images via different techniques," *Journal of Physics: Conference Series*, vol. 1179, no. 1, pp. 1–6, Jul. 2019, doi: 10.1088/1742-6596/1179/1/012087.
- [3] S. Chandra and S. Maheskar, "Offline signature verification based on geometric feature extraction using artificial neural network," in *2016 3rd International Conference on Recent Advances in Information Technology (RAIT)*, Mar. 2016, pp. 410–414, doi: 10.1109/RAIT.2016.7507937.
- [4] F. S. Mohamad, F. M. Alsuhimat, M. A. Mohamed, M. Mohamad, and A. A. Jamal, "Detection and feature extraction for images signatures," *International Journal of Engineering & Technology*, vol. 7, no. 3, pp. 44–48, Aug. 2018, doi: 10.14419/ijet.v7i3.28.20963.
- [5] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, "Learning features for offline handwritten signature verification using deep convolutional neural networks," *Pattern Recognition*, vol. 70, pp. 163–176, Oct. 2017, doi: 10.1016/j.patcog.2017.05.012.
- [6] F. M. Alsuhimat and F. S. Mohamad, "Offline signature recognition via convolutional neural network and multiple classifiers," *International Journal of Network Security & Its Applications*, vol. 14, no. 1, pp. 43–52, Jan. 2022, doi: 10.5121/ijnsa.2022.14103.
- [7] B. Thakare and P. Mahalle, "Handwritten signatures: An understanding," *International Journal of Computer Applications*, vol. 139, no. 4, pp. 21–26, Apr. 2016, doi: 10.5120/ijca2016909143.
- [8] F. M. Alsuhimat and F. S. Mohamad, "Histogram orientation gradient for offline signature verification via multiple classifiers," *Nveo-Natural Volatiles & Essential OILS Journal| NVEO*, vol. 8, no. 6, pp. 3895–3903, 2021.
- [9] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, "Offline handwritten signature verification—Literature review," in *2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA)*, Nov. 2017, pp. 1–8, doi: 10.1109/IPTA.2017.8310112.
- [10] M. Hanmandlu, A. B. Sronothara, and S. Vasikarla, "Deep learning based offline signature verification," in *2018 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, Nov. 2018, pp. 732–737, doi: 10.1109/UEMCON.2018.8796678.
- [11] T. Jahan, A. Shahriar, and S. M. A. Al-Mamun, "A study on preprocessing and feature extraction in offline handwritten signatures," *Global Journal of Computer Science and Technology*, vol. 15, no. 2, pp. 21–25, 2015.
- [12] S. N. Gunjal, B. J. Dange, and A. V. Brahmane, "Offline signature verification using feature point extraction," *International Journal of Computer Applications*, vol. 141, no. 14, pp. 6–12, May 2016, doi: 10.5120/ijca2016909852.
- [13] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005, vol. 1, pp. 886–893, doi: 10.1109/CVPR.2005.177.
- [14] N. H. Abbas, K. N. Yasen, K. H. A. Faraj, L. F. A. Razak, and F. L. Malallah, "Offline handwritten signature recognition using histogram orientation gradient and support vector machine," *Journal of Theoretical and Applied Information Technology*, vol. 96, no. 8, pp. 2075–2084, 2018.
- [15] X.-H. Le, H. V. Ho, G. Lee, and S. Jung, "Application of long short-term memory (LSTM) neural network for flood forecasting," *Water*, vol. 11, no. 7, pp. 1–19, Jul. 2019, doi: 10.3390/w11071387.
- [16] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017, doi: 10.1109/TNNLS.2016.2582924.
- [17] A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 855–868, May 2009, doi: 10.1109/TPAMI.2008.137.
- [18] S. Yan, "Understanding LSTM and its diagrams," *Medium*, 2016. <https://medium.com/mlreview/understanding-lstm-and-its-diagrams-37e2f46f1714> (accessed Apr. 10, 2022).
- [19] A. Soleimani, K. Fouladi, and B. N. Araabi, "UTSig: A Persian offline signature dataset," *IET Biometrics*, vol. 6, no. 1, pp. 1–8, Jan. 2017, doi: 10.1049/iet-bmt.2015.0058.
- [20] S. Dey, A. Dutta, J. I. Toledo, S. K. Ghosh, J. Lladós, and U. Pal, "SigNet: Convolutional siamese network for writer independent offline signature verification," Jul. 2017, [Online]. Available: <http://arxiv.org/abs/1707.02131>
- [21] J. Poddar, V. Parikh, and S. K. Bharti, "Offline signature recognition and forgery detection using deep learning," *Procedia Computer Science*, vol. 170, pp. 610–617, 2020, doi: 10.1016/j.procs.2020.03.133.
- [22] S. Rana, A. Sharma, and K. Kumari, "Performance analysis of off-line signature verification," in *International Conference on Innovative Computing and Communications*, 2020, pp. 161–171, doi: 10.1007/978-981-15-1286-5_14.
- [23] D. R. Kisku, P. Gupta, and J. K. Sing, "Fusion of multiple matchers using SVM for offline signature identification," in *International Conference on Security Technology*, 2009, pp. 201–208, doi: 10.1007/978-3-642-10847-1_25.
- [24] K. Daqrouq, H. Sweidan, A. Balamesh, and M. Ajour, "Off-line handwritten signature recognition by wavelet entropy and neural network," *Entropy*, vol. 19, no. 6, p. 252, May 2017, doi: 10.3390/e19060252.
- [25] R. Kumar, J. D. Sharma, and B. Chanda, "Writer-independent off-line signature verification using surroundedness feature," *Pattern Recognition Letters*, vol. 33, no. 3, pp. 301–308, Feb. 2012, doi: 10.1016/j.patrec.2011.10.009.
- [26] A. Soleimani, B. N. Araabi, and K. Fouladi, "Deep multitask metric learning for offline signature verification," *Pattern*

Recognition Letters, vol. 80, pp. 84–90, Sep. 2016, doi: 10.1016/j.patrec.2016.05.023.




- [27] G. Sulong, A. Y. Ebrahim, and M. Jehanzeb, "Offline handwritten signature identification using adaptive window positioning techniques," *Signal & Image Processing: An International Journal*, vol. 5, no. 3, pp. 13–24, Jun. 2014, doi: 10.5121/sipij.2014.5302.

BIOGRAPHIES OF AUTHORS



Fadi Mohammad Alsuhiat    received the B.S. degree in computer information system from Alhussien Bin Talal University, Ma'an, Jordan, in 2007, the M.S. degree in computer science from Utara University Malaysia (UUM), Kedah, Malaysia, and now Ph.D. student in pattern recognition, deep learning at University Sultan Zainal Abidin, Kuala Terengganu (UNISZA), interesting in machine and deep learning, data science, and artificial intelligence. He can be contacted at email: karaklove@yahoo.com.



Fatma Susilawati Mohamad    is an Associate Professor in School of Information Technology, Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Terengganu, Malaysia. She obtained her B.Sc in Information System from Oklahoma City University, USA in 1997. Then she pursued her master's degree in computer science from Universiti Kebangsaan Malaysia in 2004. In 2012, she obtained her Ph.D in Computer Science specializing in Pattern Recognition from Universiti Teknologi Malaysia. Currently, she is appointed as Deputy Director of Research Institute for Islamic Product and Malay Civilization. Dr. Fatma is acted as Head of Image Processing and Pattern Recognition Research Group at the faculty. At present, she involved in research in Deep Learning apart of human biometric identification. Dr Fatma is an experienced lecturer. She has more than 20 years experiences in teaching and research. She has taught various courses both in undergraduate and postgraduate level over the past 20 years. Dr Fatma has published more than 100 journal articles, proceedings, and academic books to date. She has graduated more than 20 Ph.D and Master students at present. Due to her vast experience and knowledge, Dr. Fatma has been invited to deliver her keynote and invited speech at several conferences and academic seminars local or internationally. She is also being invited to give talk in academic and research for several institutions local and international. Besides, Dr. Fatma also being appointed as an Academic Advisor and evaluator for postgraduate program local and internationally. She can be contacted at email: farie999@gmail.com.