Indian Journal of Engineering, Management and Technology

https://doi.org/XX.XXXX/ijems.XXXX

Research Article

Exploring Various Feature Extraction Methods for Offline Signature Verification

Bhavani S.D.[a], Bharathi R.K.[b]

[a, b]*Department of Computer Applications, JSS Science and Technology University, Mysuru, Karnataka, 570006, India,*

Received: 20-05-2025, *Accepted*: 20-09-2025, *Published*: 28-09-2025

Abstract

Offline signature verification is a difficult task in the biometric authentication domain, with particular importance in legal documents, banking activities, and identity management systems. The performance of offline signature verification systems is heavily reliant on the quality of features extracted. Feature extraction forms the cornerstone upon which sophisticated models perceive, analyze, and make sense of digital data. This paper delves into the intricacies of feature extraction, examining its significance and associated challenges. In this paper, three feature extraction methods, namely Histogram of Oriented Gradients (HOG), which captures directional and structural local edges, GLCM (capturing statistical texture information), and Local Binary Pattern (LBP), have been exploited. Instead of using a single classifier, a Voting Classifier is used, which includes Support Vector Machines (SVM), Random Forests (RF), and K-nearest neighbor (KNN) for discriminating genuine and forged signatures. The proposed models are evaluated on five benchmark datasets: CEDAR, BHSig260 (Bengali and Hindi), UTSig, and MCYT-75.

Keywords: Offline Signature Verification, GLCM, HOG, LBP, SVM, Random Forest, K-nearest neighbor, Voting Classifier.

***Corresponding Author: Bharathi R.K.**

Email: *rkbharathi@jssstuniv.in*

1. Introduction

In the context of rapidly evolving technological landscapes and innovation-driven digital systems, data protection has emerged as a critical priority for preventing fraud, unauthorized access, and data breaches. Although numerous biometric authentication techniques exist—such as facial recognition, iris scanning, and voice-based identification—signature verification is considered one of the most practical and widely accepted methods for validating identity and securing sensitive information. Signature verification serves as an effective mechanism for preventing unauthorized access by distinguishing genuine signatures from forgeries. Its widespread applicability across various institutions is attributed to its ease of implementation, cost-effectiveness, and high level of social and legal acceptance. Unlike more complex biometric systems, signature-based authentication does not require specialized hardware and remains user-friendly, making it suitable for both digital and physical verification scenarios.

A handwritten signature is a unique behavioral biometric composed of distinct symbols, letters, and sometimes numbers, written in a specific language and personalized by the signer. It represents a learned motor skill and serves as a commonly used method for authenticating individuals across diverse applications, including financial transactions, legal documentation, and institutional processes like attendance and authorization. One of the key challenges in signature verification arises from its variability: a signature is not a fixed image or structure but rather a complex and dynamic graphical pattern that reflects the individuality of the signer. Signatures can consist of different shapes, strokes, characters, or stylistic elements, making them susceptible to forgery. For this reason, signature verification systems are essential for distinguishing between authentic and fraudulent signatures [6,8,9]. This paper proposes three different models for solving the signature verification problem. The first model is based on HOG features, the second model is based on GLCM features, and the third model is based on LBP features. Once the relevant features are extracted, the features are fed to Voting classifiers that include SVM, RF, and kNN for the verification task. The classifiers are trained and tested using stratified k-fold cross-validation.

The rest of the paper is organized as follows. Section 2 narrates the methodologies proposed in the literature, Section 3 describes the proposed model, the results are drawn in Section 5, and finally conclusion is given in Section 6.

2. Literature Survey

Offline signature verification is a tricky problem, and many researchers in the literature have tried to solve this problem. The process of signature verification involves extraction of useful features from signature images before passing it to classifiers for verification. In paper [5], the authors have used HOG method for extracting features and then features were fed to LSTM. The proposed model was evaluated on CEDAR and UTSig Datasets. In [7] authors proposed a writer-dependent (WD) offline signature verification system. The useful features were extracted using GLCM and Improved Local Binary Pattern (ILBP), respectively, and these features are fused with geometric features, and finally, SVM was used for classification. In paper [4] authors extracted features using CNN and HOG; these features were fused to form hybrid feature vector. From fused feature vector relevant features were selected using decision tree. The selected features were fed to classifiers such as SVM, LSTM and KNN for the verification task. The authors in paper [2] extracted GLCM features and used Graph Neural Network for verification. The proposed model has performed well compared CNN. Three different types of features namely principal component analysis (PCA) , gray-level co-occurrence matrix (GLCM), and fast Fourier transform (FFT) were extracted [3] from signature images in order to build a hybrid feature vector for each image. Finally, to classify signature features, we have designed a proposed fast hyper deep neural network (FHDNN) architecture. In paper [1], the authors used a Gaussian Denoising Filter, GLCM, Principal Component Analysis (PCA), Kernal Principal Component Analysis (KPCA) for feature extraction on the Kaggle dataset. The authors in [15] used pre-trained deep neural network called VGG16 for solving offline signature verification task. They evaluated their method using CEDAR dataset and showed the significance of hyper-parameters used to tune the model. The authors in [11] used Convolution Neural Network (CNN), Crest-Trough method and SURF algorithm & Harris corner detection algorithm for solving offline signature verification task. The authors in [12] used combinations of four features, i.e., Average object area, mean, Euler number and area of signature image to verify the signature. They evaluated their model on BHSig260 Bengali and Hindi dataset. In paper [13] authors collected signatures at Yildiz Technical University, from 15 people, 40 samples from each. They extracted HOG features from signature images and feature reduction using PCA was done before feeding to ANN for verification. The authors in [14] used a probabilistic neural network (PNN) and wavelet transform average framing entropy (AFE). The system was tested with a wavelet packet (WP) entropy and with a discrete wavelet transform (DWT) entropy. The authors in [10] proposed a graph neural network-based architecture for offline signature verification. In this work, the features in the signature images are extracted by the SIFT algorithm and sent to the graph-based neural network as a graph structure. The authors in [16], used chain code histogram to extract features and fed the features to SVM for classification. They evaluated their model on English dataset and the Kannada dataset called MUKOS. In paper [17], Discrete Cosine Transform is applied to obtain feature vector and feature reduction was done using Linear Discriminant Analysis. Then the reduced feature vector is fed to Multi-layer Perceptron for classification. The authors in [18] used a pre-trained deep neural network called VGG16 for solving the offline signature verification task. They evaluated their method using the CEDAR dataset and also showed the significance of the hyperparameters used to tune the model.

3. Methodology

This section explains the methodology used in this paper to solve the signature verification problem. Here we have proposed two models like Model 1 based on HOG features and Model 2 based on GLCM features. The Fig. 1 shows the proposed method in this paper.

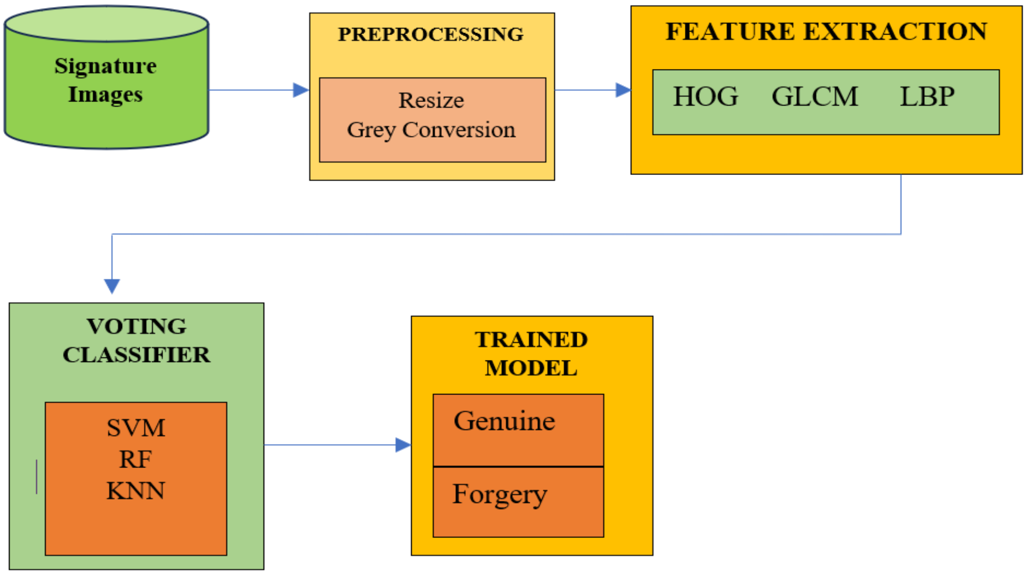

Fig. 1*:* Proposed Models

3.1. Feature Extraction Stage

In this study, the Histogram of Oriented Gradients (HOG) method and GLCM is used for offline signature verification. In this work, HOG and GLCM features are extracted from signature images. Then the extracted features are fed to voting classifier to discriminate signatures as authentic or forgery. Figure 2 illustrates the workings of the HOG algorithm.

Model 1: Histogram of Oriented Gradients based Signature Verification (HOG_SV)

The Histogram of Oriented Gradients (HOG) descriptor is based on the idea that the appearance and shape of objects in an image can be effectively characterized by the distribution of edge directions or intensity gradients. Initially, images are converted to grey scale and resized to (380 X 962). To compute HOG features, the image is split into small, connected regions called cells (128 X 128). Within each cell, the algorithm computes a histogram that captures the directions of gradients (edges) present.

These local histograms are then combined to form a single feature vector that describes the entire image or a region of interest. To enhance accuracy, especially under varying lighting conditions or shadows, the method normalizes these histograms. This is done using larger overlapping regions called blocks(2 X 2), which provide a measure of local contrast. By normalizing the histograms within these blocks, the descriptor gains robustness to illumination changes. Then finally flatten all block histograms into a 1D feature vector. The image size , cell size and block size , all determine the feature vector dimension. Here we have obtained 216 distinct features for each image by setting above mentioned specifications.

One of the main strengths of the HOG descriptor is its resistance to small geometric and lighting variations. Because it works on localized patches of the image, it remains reliable even if there are changes in pose or brightness—except for large orientation shifts of the object. As found by Dalal and Triggs, using coarse spatial grids, finely divided orientation bins, and robust contrast normalization makes HOG especially effective at detecting humans, since it can overlook minor variations in body posture, provided the person is generally upright.

HOG is a powerful tool for detecting objects—especially pedestrians—because it focuses on local edge structure, which remains relatively stable across many real-world image variations.

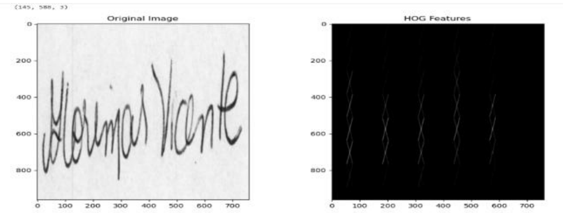

Fig. 2: HOG feature extracted for signature image

Model 2: Gray Level Co-occurrence Matrix based Signature Verification (GLCM_SV)

The Gray Level Co-occurrence Matrix (GLCM) is a texture analysis method used in image processing, including biometric applications like signature verification. GLCM is a statistical method that examines the spatial relationship of pixels in a grayscale image. It calculates how often pairs of pixels with specific values and in a specified spatial relationship occur in an image. Signatures are complex patterns with unique textures and structures. GLCM helps extract texture features that can distinguish between different individuals' signatures, making it useful for: Offline (static) signature verification, where images of signatures are analyzed. Feature extraction, where patterns in pixel intensity are used to identify or authenticate.

It begins with converting the images to a grey scale to reduce computation complexity. The co-occurrence matrix is computed for four different angles, such as 0, 45, 90, and 135 degrees. Once the co-occurrence matrix is computed, various features like contrast, homogeneity, dissimilarity, energy, and entropy are calculated. The resultant feature vector consists of 24 features for each signature image.

Model 3: Local Binary Pattern-based Signature Verification (LBP_SV)

Local Binary Pattern is a texture descriptor used to summarize the local spatial structure of signature images. For a given pixel, the intensities of 8 surrounding neighbors are compared. Then, binary values are assigned based on whether the neighbor's intensity is greater than or equal to the center pixel. This forms an 8-bit binary number, which is then converted to a decimal number indicating the LBP code for the center pixel. Finally, histogram of LBP is computed and this histogram if used as feature vector. Here, uniform LBP is used to reduce the feature vector dimension. In this work, 18 LBP features are extracted for each image.

3.2 Voting Classifier

The extracted features are fed to a Voting classifier, which includes classifiers such as SVM, RF, and KNN. A Voting Classifier is an ensemble method that combines predictions from multiple different machine learning models to improve overall accuracy and robustness. Each model has its strengths and weaknesses. A voting classifier combines them to achieve better performance than any single model. In this work, soft voting is used, which averages the predicted probabilities and then picks the highest.

Cross Validation: Instead of using a static split, we have used cross validation, which better generalizes the model. The k-fold cross-validation splits the dataset into K equal parts (folds). Then the model is trained K times, each time K–1 fold is used for training, and 1-fold for testing (a different one each time). The average of the performance over the K tests is taken as the final result.

4. Dataset Description

In this study, we evaluate and compare the performance of four signature verification algorithms using benchmark datasets: UTSig, CEDAR, MCYT-75, and BHSig260. Each dataset offers unique challenges and characteristics related to the types of signatures, collection protocols, and demographic diversity.

UTSig Dataset: UTSig is a Persian offline signature dataset comprising 8,280 scanned signature images from 115 individuals. Each person provided 27 genuine signatures, 3 opposite-hand forgeries, 36 simple forgeries, and 6 skilled forgeries. Signatures were collected from students at the University of Tehran and Sharif University of Technology, scanned at 600 dpi resolution, and stored as 8-bit TIFF images

CEDAR Dataset: The CEDAR dataset includes handwritten signatures from 55 individuals of varying professional and cultural backgrounds. Each signer provided 24 genuine signatures (collected in separate sessions, 20 minutes apart), 24 skilled forgeries (produced by three forgers, each attempting eight forgeries). Total signature count: 1,320 genuine signatures and 1,320 forged signatures. This dataset is widely used due to its balanced structure and challenging forged samples.

MCYT-75 Dataset: The MCYT-75 dataset is a subset of the full MCYT biometric signature dataset, containing data from 75 individuals. Each individual contributed 15 genuine signatures, 15 skilled forgeries (produced by other participants imitating the original signature). Total signatures: 1,125 genuine signatures and 1,125 skilled forgeries. The signatures were collected using a WACOM graphic tablet, making this dataset suitable for both online and offline verification systems (we use the static images in this paper).

*BHsig260 Bengali Dataset:*The BHsig dataset includes signature samples from Indian writers, covering multiple scripts including Hindi and Bengali. There are two subsets: BHsig-H (Hindi): 100 writers, 40 signatures per writer (24 genuine, 16 forged). BHsig-B (Bengali): 100 writers, 40 signatures per writer (24 genuine, 16 forged). Total: 4,800 Hindi signatures and 4,800 Bengali signatures. BHsig is particularly useful for studying script-influenced variations in signature style. We have used BHSig-Bengali in this paper.

5. Results and Discussion

This section presents the verification results of the classifiers used in this paper. Different metrics, such as Voting Classifier Accuracy (VCA), False Acceptance Rate (FAR), False Rejection Rate (FRR), and Average Error Rate (AER), are used to evaluate the proposed model. Ultimately, the proposed model's results are compared with state-of-the-art work. Table 1, 2 and 3 shows the accuracies obtained by HOG_SV, GLCM_SV, and LBP_SV models.

Table 1: Results of HOG_SV model

Dataset	VCA	FAR	FRR	AER
CEDAR	97.08	5.37	0.45	2.91
BHSig_B	96.38	2.40	5.12	3.76
BHSig_H	91.67	7.04	10.02	8.53
MCYT_75	87.37	15.02	10.22	12.62
UTSig	89.64	53.47	0.77	27.12

Table 2: Results of GLCM_SV model

Dataset	VCA	FAR	FRR	AER
CEDAR	100	0.00	0.00	0.00
BHSig_B	90.81	4.46	15.08	9.77
BHSig_H	79.15	12.79	30.91	21.85
MCYT_75	76.48	24.08	22.93	23.51
UTSig	82.26	95.65	0.41	48.03

Table 3: Results of LBP_SV model

Dataset	VCA	FAR	FRR	AER
CEDAR	100	0.00	0.00	0.00
BHSig_B	83.61	4.10	31.75	17.92
BHSig_H	74.42	11.14	43.61	27.38
MCYT_75	79.28	16.44	24.97	20.71
UTSig	95.73	17.68	1.28	9.48

Among the three models proposed in this work, HOG_SV model outperforms. The HOG_SV model performs well on BHSig260 Hindi, BHSig260 Bengali and MCYT_75 datasets whereas LBP_SV model performs well on CEDAR and UTSig datasets. The Fig. 3 illustrates the results of all three models proposed in this work.

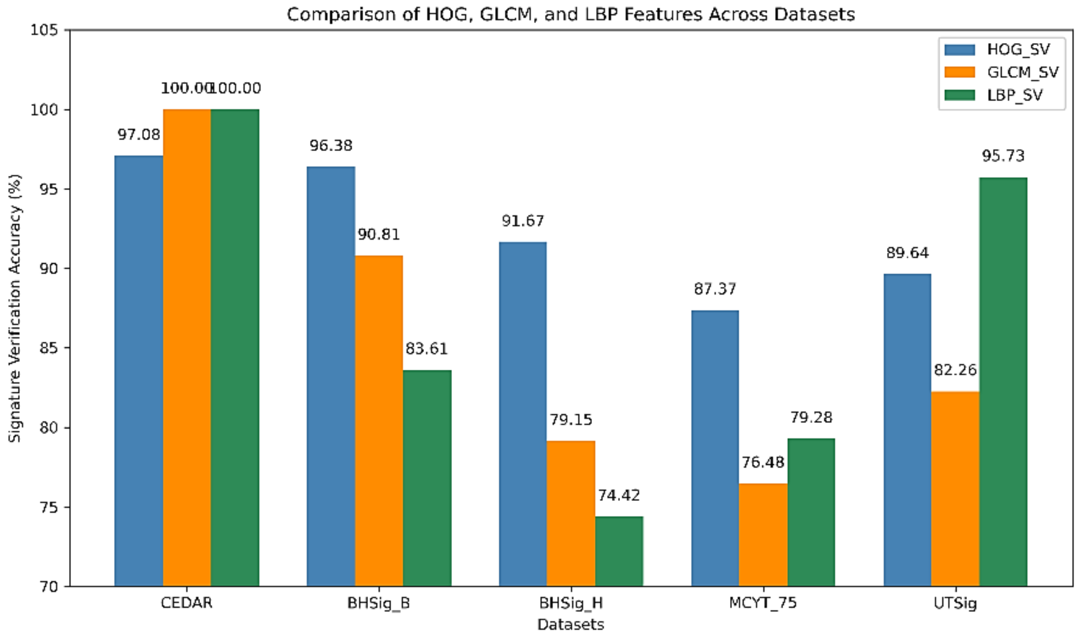

Fig. 3: Comparison of results of HOG_SV, GLCM_SV, and LBP_SV models

The proposed model is compared with state-of-the-art work to show the robustness of the proposed model. Table 4 shows the results of models proposed in the literature along with the results of the proposed methods in this paper.

Table 4: Comparison of proposed model with state-of-the-art work

Ref.	Methods used	Accuracy
[5]	LSTM and HOG	UTSig: 92.4, CEDAR: 87.7
[4]	CNN and HOG	CEDAR: 95.4
[7]	GLCM and SVM	CEDAR:97.0, MCYT:94.90
	Proposed HOG_SV model **LBP_SV model**	BHSig_B: 96.38, BHSig_H: 91.67, MCYT_75: 87.37 CEDAR: 100, UTSig : 95.73

6. Conclusion

This research proposes three models for signature verification by using HOG and GLCM algorithms for feature extraction from signature images. Then the extracted features were saved as a vector and classified into two classes, genuine and forgery, using a voting classifier (including SVM, RF, and KNN). The HOG_SV model has achieved good results on BHSig260 Bengali, BHSig260 Hindi, and MCYT_75. The LBP_SV model has achieved good results on the CEDAR and UTSig datasets.

As the results showed HOG method gives better results compared to GLCM and LBP. The LBP_SV model used a lesser number of features compared to the HOG_SV and GLCM_SV models. Our future work concentrates on selecting relevant features and removing redundant features so as to reduce computation complexity yet achieve good verification results.

Declarations

Ethical Approval

This research did not involve any studies with human participants. Therefore, ethical approval was not required.

Consent to Participate

Not applicable. The study used publicly available datasets and did not involve direct interaction with human participants.

Data Availability Statement

The datasets analysed in this study (CEDAR, BHSig260, MCYT_75, UTSig) are publicly available from their respective official sources.

Authors Contributions

Bhavani S. D. has designed the study, conducted experiments, and written the paper.

Bharathi R. K. is the research supervisor, has offered essential guidance at the inception of the study, and has continuously provided valuable insights and suggestions throughout its progression.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Competing Interest

All authors declare that they have no competing interests related to this work. The authors declare that this manuscript is original, has not been published before, and is not currently being considered for publication elsewhere.

References

[1] Lokare, C., Patil, R., Rane, S., Kathirasen, D., & Mistry, Y. (2021). Offline handwritten signature verification using various Machine Learning Algorithms. In ITM Web of Conferences (Vol. 40, p. 03010). EDP Sciences.

[2] Roy, S., Sarkar, D., Malakar, S., & Sarkar, R. (2023). Offline signature verification system: a graph neural network based approach. Journal of Ambient Intelligence and Humanized Computing, 1-11.

[3] Hashim, Z., Mohsin, H., & Alkhayyat, A. (2024). Signature verification based on proposed fast hyper deep neural network. IAES Int J Artif Intell, 13(1), 961-73.

[4] Alsuhimat, F. M., & Mohamad, F. S. (2023). A hybrid method of feature extraction for signatures verification using CNN and HOG a multi-classification approach. IEEE Access, 11, 21873-21882.

[5] Alsuhimat, F. M., & Mohamad, F. S. (2023). Offline signature verification using long short-term memory and histogram orientation gradient. Bulletin of Electrical Engineering and Informatics, 12(1), 283-292.

[6] Bhavani, S. D., & Bharathi, R. K. (2024). A multi-dimensional review on handwritten signature verification: strengths and gaps. Multimedia Tools and Applications, 83(1), 2853-2894.

[7] Wang, Y., Zheng, J., & Zhou, Y. (2022, September). An Efficient Offline Signature Verification Method Based on Improved Feature Extraction. In 2022 2nd International Conference on Computer Science, Electronic Information Engineering and Intelligent Control Technology (CEI) (pp. 609-612). IEEE.

[8] Diaz, M., Ferrer, M. A., Impe dovo, D., Malik, M. I., Pirlo, G., & Plamondon, R. (2019). A perspective analysis of handwritten signature technology. A cm Computing Surveys (C sur), 51(6), 1-39.

[9] Swamy, M. R., Vijayalakshmi, P., & Rajendran, V. (2025, February). Signature verification based on machine learning and deep learning techniques: A review. In AIP Conference Proceedings (Vol. 3162, No. 1). AIP Publishing.

[10] Badie, A., & Sajedi, H. (2024). Offline handwritten signature authentication using Graph Neural Network methods. International Journal of Information Technology, 1-11.

[11] 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.

[12] 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.

[13] Taşkiran, M., & Çam, Z. G. (2017, January). Offline signature identification via HOG features and artificial neural networks. In 2017 IEEE 15th International Symposium on Applied Machine Intelligence and Informatics (SAMI) (pp. 000083-000086). IEEE.

[14] 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.

[15] Bhavani, S. D., Bharathi, R. K., & Kumar, R. J. (2024, March). Offline signature verification using pre-trained deep convolutional neural network. In AIP Conference Proceedings (Vol. 2966, No. 1). AIP Publishing.

[16] Bharathi, R. K., & Shekar, B. H. (2013, August). Off-line signature verification based on chain code histogram and support vector machine. In 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI) (pp. 2063-2068). IEEE.

[17] Bharathi, R. K., & Shekar, B. H. (2014, September). Discriminative dct-mlp based approach for off-line signature verification. In 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI) (pp. 2309-2315). IEEE.

[18] Bhavani, S. D., Bharathi, R. K., & Kumar, R. J. (2024, March). Offline signature verification using pre-trained deep convolutional neural network. In AIP Conference Proceedings (Vol. 2966, No. 1). AIP Publishing.