Electronic Letters on Computer Vision and Image Analysis 24(2):123-142, 2025

Deep Transfer Learning based Handwritten Signature Verification

S.Arivazhagan , R.Newlin Shebiah* , A.Brintha and T.Subashini

Centre for Image Processing and Pattern Recognition, Department of Electronics and Communication Engineering
Mepco Schlenk Engineering College, Sivakasi - 626005, Tamilnadu, India

Received 21th of october, 2022; accepted 25th of september 2025

Abstract

In recent decades, the field of handwritten signature verification has gained substantial attention and continues to be an area of active research. Signature verification techniques play a critical role in establishing the authenticity of signatures, differentiating between genuine signatures and forgeries created by impostors. Among the various advancements documented in the literature, a prominent approach involves deep learning algorithms to extract and learn image features from signature images. This study applies transfer learning techniques to train models using three leading pretrained architectures: VGG-16, VGG-19, and AlexNet, functioning as both feature extractors and classifiers. The CEDAR and ICDAR 2011 datasets were used for validating the proposed methodology. Training and testing on these datasets revealed that VGG-19 achieved the highest accuracy and robustness, significantly minimizing misclassifications, while VGG-16 demonstrated best performance on the CEDAR dataset. This research offers a detailed comparison of these architectures, advancing the field of biometric verification by demonstrating the efficacy of deep convolutional neural networks in detecting forged signatures.

Key Words: Signature verification, Deep Learning, Convolutional Neural Networks (CNNs) , Transfer learning.

1 Introduction

Biometrics uses biological data and physical attributes for identification and verification purposes. It encompasses diverse authentication techniques, including fingerprint analysis, facial

Correspondence to: newlinshebiah@mepcoeng.ac.in

Recommended for acceptance by Angel D. Sappa
https://doi.org/10.5565/rev/elcvia.1612
ELCVIA ISSN:15108-5097
Published by Computer Vision Center / Universitat Autònoma de Barcelona, Barcelona, Spain

recognition, and retinal scanning. Biometric identification is broadly classified into two types: physiological and behavioral. Physiological biometrics refers to intrinsic physical characteristics such as fingerprints [1], gait patterns [2], iris patterns [3], and facial features [4]. In contrast, behavioral biometrics is centered on unique activity patterns, including handwritten signatures [5], typing dynamics [6], and vocal characteristics [7].

Signature verification, a specialized domain within behavioral biometrics, is essential for authenticating identity claims in sectors such as banking, intelligence, and high-security organizations. The ability to distinguish genuine signatures from forgeries is critical in maintaining the integrity and security of transactions and sensitive data. Online signature verification relies on dynamic inputs such as time, pressure, and stylus movements captured by electronic devices [8, 9], while offline verification processes involve digitization of handwritten signatures for comparative analysis.

Automated signature verification significantly enhances the authentication process by reducing manual effort, conserving resources, and mitigating the risk of fraud. Through the analysis of signature attributes and the generation of confidence scores, this method effectively identifies forgeries and strengthens security protocols. Technological advancements have further refined signature verification systems, making them more lightweight, efficient, and adaptable while delivering enhanced accuracy and dependability.

Handwritten signature verification encounters numerous challenges that complicate the authentication process. One primary difficulty lies in the inherent variability of signatures, including differences in style, size, and shape, which hinder the design of consistent verification models. The presence of skilled forgers capable of accurately replicating signatures further complicates the distinction between genuine and forged signatures. Additionally, even signatures from the same individual often display significant intra-class variability, making accurate classification more difficult.

External factors such as noise, occlusion, and distortions caused by scanning or variations in writing conditions further degrade the accuracy of the verification. Limited and unbalanced datasets exacerbate these issues, restricting the ability of machine learning models to train effectively and generalize in diverse scenarios. Moreover, ensuring that verification systems are secure and resilient to adversarial attacks or fraudulent attempts is critical, particularly in high-stakes environments such as financial transactions.

The computational demands of feature extraction and classification, coupled with the need for legal and ethical compliance, add another layer of complexity. Addressing these challenges requires advancement in machine learning algorithms, data augmentation, and robust feature extraction techniques tailored specifically to handwritten signature verification. By integrating innovative approaches, the field can achieve greater accuracy, reliability, and resilience against evolving threats.

This study presents a detailed evaluation of the AlexNet [10] VGG16 [11], VGG19 [11], and architectures for handwritten signature verification, highlighting their efficacy in differentiating genuine signatures from forgeries. Utilizing transfer learning and fine-tuning of hyperparameters, the models demonstrate near-perfect accuracy, particularly on the CEDAR and ICDAR datasets. Among the evaluated architectures, VGG19 stands out for its superior performance and robustness in handling signature verification tasks. These results provide significant

insights into the development of advanced biometric verification systems, showcasing the potential of deep convolutional neural networks to address challenges associated with signature forgery detection in practical applications.

2 Literature Survey

Navid et al. [12] utilized Convolutional Neural Networks (CNNs) to automate signature verification, employing the pre-trained VGG19 architecture as the base model. Their approach was evaluated on the CEDAR and ICDAR datasets, achieving remarkable accuracy rates of 100% and 88.4%, respectively, thereby demonstrating its effectiveness across diverse datasets. Bonde et al. [13] explored two distinct methodologies for signature verification. In their writer-independent approach, the VGG16 was fine-tuned to generalize across writers, while in the writer-dependent approach, features extracted from signatures using the fine-tuned CNN were tailored to individual writers. Haaffemann et al. [14] employed Deep Convolutional Neural Networks to develop writer-dependent classifiers, using a writer-independent learning framework to extract robust features for signature verification.

Jain et al. [15] created a language-independent shallow architecture based on Convolutional Neural Networks, known as sCNN (Shallow Convolutional Neural Network) that is specifically designed for signature verification.The effectiveness of the suggested design is assessed using publicly accessible datasets, including CVBLSig-V1 and CVBLSig-V2, in addition to MCYT-75, MCYT-100, and GPDS.

Narwade et al. [16] presents a novel approach that uses an adaptive weighted mixture of form context distance and Euclidean distance to find correlations between pixels of different signatures. The query signature plane and reference signature plane are then aligned via the thin plate spline transformation, which makes use of these correlations. The distances between signatures are then calculated using plane transformation, a shape descriptor, and the difference between matching pixels. The acquired distances are sent into a Support Vector Machine (SVM) classifier to determine authenticity. Narwade et al. [17] used curve matching with a shape descriptor and Euclidean distance are used to verify signatures. In order to improve accuracy, the authors create correspondences between signatures and include a shape descriptor (shape context) in addition to the Euclidean distance between sample signature points. These correspondences are used to estimate the alignment transforms, and then linear discriminant analysis and measures of shape dissimilarity are used for the signature classification.

Sulong et al. [18] underscore the significance of the writer's style and the intrinsic importance of handwritten signatures, introducing the Adaptive Window Positioning approach. This method involves segmenting handwritten signatures into (13x13) small windows, aiming to strike a balance between preserving the authorial style and facilitating precise recognition. The efficacy of this approach was assessed using a GPDS dataset comprising 4870 signature samples from 90 distinct writers. Evaluation involved comparing robust elements of test signatures with user signatures through the utilization of an appropriate classifier. Results indicate that the Adaptive Window Positioning approach is effective and reliable for accurately extracting signature attributes.

Hans & Gupta [19] introduce an effective approach for enhancing signature images in of-

fline signature verification systems. They demonstrate that preprocessing significantly enhances image quality, as evidenced by comparisons of Signal-to-Noise Ratio (SNR) and Peak Signal-to-Noise Ratio (PSNR) between processed and raw images. Soleimani et al. [20] present a Persian signature verification system using features such as the Histogram of Curvature (HOC) and Histogram of Oriented Gradient (HOG). Their method utilizes a discrete curvature estimator based on normalized gradients and Jacobian matrices, overcoming the challenge of computing curvature for offline signatures due to the absence of temporal information. Pal et al. [21] proposed an offline verification technique for Hindi signatures, combining Support Vector Machines (SVMs), Zernike moment features, gradient features, and offline signature verification. Their work achieves promising results, with gradient features contributing to an impressive accuracy of 90.69% and error rates of 11.50% False Rejection Rate (FRR) and 7.12% False Acceptance Rate (FAR) from a Hindi signature database.

Serdouk et al. [22] propose a method for offline signature verification utilizing the Artificial Immune Recognition System (AIRS), incorporating Gradient Local Binary Patterns and Longest Run Feature descriptors. Bharathi and Shekar [23] present an offline signature verification technique based on chain code histogram features enhanced by a Laplacian of Gaussian filter. They describe the process of updating and extracting features for signature verification, which are subsequently fed into a Support Vector Machine (SVM) classifier for classification.

Maergner et al. [24] introduced techniques for constructing, comparing, and normalizing signature graphs, using an effective cubic-time approximation of graph edit distance. Experimental findings demonstrate the efficacy of the proposed structural approach, showcasing its superiority over state-of-the-art techniques in offline signature verification. Naemah et al. [25] propose an approach involves extracting pattern information from signature data by combining skeleton orientation and gravity center point.

Ferrer et al. [26] investigate the robustness of offline signature verification based on gray level features, aiming to quantify their resistance to distortion from complex backgrounds while proposing more stable alternatives. Batista et al. [27] introduce dynamic selection of generative-discriminative ensembles for offline signature verification. Their focus lies in developing an offline signature verification system utilizing Hybrid Generative-Discriminative Ensembles of Classifiers (EoCs).

Steinherz et al. [28] delve into contour-based handwriting analysis and loop modeling. Their work yields excellent results, particularly in scenarios requiring loop resolution, contributing to advancements in handwriting analysis techniques. Nguyen et al. [29] contribute to automated signature verification with their work on global features for the off-line signature verification problem. Their proposed features, based on a signature's border and projections, exhibit promising outcomes when employed with Support Vector Machine (SVM) classifiers, showcasing potential for improving verification accuracy. Solar et al. [30] present a paper on offline signature verification utilizing local interest points and descriptors.

In an attempt to differentiate between simple and random forgeries, Ferrer et al. [31] propose a set of geometric signature characteristics for offline automatic signature verification using fixed-point arithmetic. Their work contributes to enhancing the capability of signature verification systems in discerning between genuine and forged signatures.Liu and Xin [32] explored the rationale behind selecting specific uni-features and their contributions within fused systems. This

paper proposes a dynamic signature verification method based on uni-feature correlation analysis, demonstrating improved accuracy with pen pressure as the most effective feature, achieving 93.46% accuracy on the SVC 2004 dataset. Yassen and AlKarem [33] focuses on developing a deep learning model for writer-independent (WI) offline signature verification based on a convolutional neural network (CNN) architecture. Zhou et al. [34] presents a handwritten signature verification method that combines both offline images and online data, with advanced smart pen technology to capture static and dynamic features simultaneously in real time. Salama [35] uses five pre-trained DLMs—ResNet50, DenseNet121, MobileNetV3, InceptionV3, and VGG16 through transfer learning and fine-tuning, computational efficiency is improved, and InceptionV3, tested on the ICDAR 2011 (Dutch) dataset, achieved 100% accuracy, AUC, and sensitivity, while the CEDAR dataset achieved 99.76% accuracy with low computational time.Ozurt et. al. [36] proposes a four-stage approach to address this issue, beginning with the collection of a large dataset of 12,600 images from 420 individuals, each contributing 30 genuine signatures. Features were extracted using the deep learning model MobileNetV2, followed by feature selection using Neighborhood Component Analysis (NCA), Chi2, and mutual information (MI), resulting in 12 feature vectors. Machine learning models, including SVM, KNN, Decision Trees, and Naïve Bayes, were applied, achieving a classification accuracy of 97.7% with NCA feature selection, compared to 91.3% without feature selection, demonstrating the effectiveness of the proposed method.

Radhika and Gopika [37] use pen tip tracking for online data and gradient/projection-based features for offline data. The signature is then verified separately by both methods, and the results are combined using a Support Vector Machine (SVM) for final verification. Calik et al. [38] introduce a new convolutional neural network (CNN) structure, named Large-Scale Signature Network (LS2Net), which includes batch normalization to address large-scale training issues. The LS2Netv2 network achieved the highest accuracy of 96.91% for the 25%–75% ratio on the GPDS-4000 dataset, with the inclusion of batch normalization and the C3 algorithm significantly improving performance. Parcham et al. [39] use a hybrid model that combines CNN with Capsule Neural Networks (CapsNet) to enhance spatial feature recognition and reduce network complexity. Sharif et al. [40] introduced a novel offline verification method that utilizes global features like aspect ratio and signature area, alongside local features such as centroid and slope.

One of the most significant challenges is the scarcity of large, high-quality labeled datasets. While data augmentation techniques like rotation, scaling, and noise addition can help mitigate the issue of data scarcity, the augmented data may not always reflect realistic forgeries or variations in signature styles. This gap could lead to overfitting and reduced generalization in real-world scenarios. Skilled forgeries, in particular, require more sophisticated models that can understand subtle variations in signature characteristics. These forgeries might not exhibit significant differences from genuine signatures, making them difficult for conventional models to detect without advanced feature extraction or deep learning-based techniques. The advent of deepfake technologies and generative models like GANs raises the possibility of fake signatures that are indistinguishable from genuine ones. Protecting against these advanced forgeries is a challenge that requires more research on robustness and defense strategies. These challenges and gaps highlight the complexity of developing reliable and efficient signature verification systems.

3 Proposed Methodology

In the context of Handwritten Signature Verification, transfer learning with a pretrained network involves using a pre-trained neural network model that was trained on a large dataset like ImageNet, which contains millions of labeled images across thousands of categories. Instead of starting the training of the neural network from scratch, transfer learning involves taking the knowledge (i.e., learned weights) gained from solving one problem (the original task) and applying it to a new, similar problem (the target task). In this case, the original task might be general image recognition, while the target task might be classifying specific types of images relevant to particular application. Transfer learning typically involves two main steps Feature Extraction and Fine-tuning. In Feature Extraction step, the pretrained network is used as a feature extractor. The input images are passed through the network, and the activations of one of the layers are extracted as features. These features are then used as input to a new classifier (often a shallow neural network or another machine learning model) that is trained specifically for the target task. Optionally, after feature extraction, the pretrained network's weights can be fine-tuned on the new dataset. This involves unfreezing some or all of the layers of the pretrained network and continuing the training process using the new dataset. Fine-tuning allows the model to adapt to the specifics of the new dataset, potentially improving performance further. Transfer learning is widely used in practice because it allows for faster training and better performance, especially when the new dataset is small or similar to the original dataset. It reduces the need for large amounts of labeled data and computational resources, making it particularly useful in scenarios where data or computational power is limited. Figure 1 shows the proposed methodology for Handwritten Signature Verification.

In signature recognition, preprocessing is a crucial initial step that prepares the data for further analysis. It involves resizing signature images to conform to the model's specifications and converting grayscale images to RGB format, which is preferred for better analysis and feature extraction. The dataset is partitioned into training, Validation and testing sets, with 70% allocated for training, 10% allocated for validation and 20% for testing. Signatures are randomly selected for each set to ensure representatives. In order to transform data into a three-dimensional format, replication of dimensions is employed. This process entails expanding the existing dimensions of the input data across three axes, resulting in a three-dimensional output. By replicating dimensions, each axis of the data is extended to form a comprehensive representation in three-dimensional space, facilitating more nuanced analysis and interpretation. Data Preprocessing Steps: 1. Data Augmentation: To increase the variability of the dataset and reduce overfitting, we applied several augmentation techniques: Rotation: Random rotations within a range of ±15 degrees. Scaling: Random scaling to mimic variations in signature sizes. Translation: Horizontal and vertical shifts up to 10% of the image dimensions. Shearing: Applying a random shear transformation within a small range. 2. Normalization: All images were normalized to have pixel values in the range [0, 1] by dividing each pixel value by 255. This step ensures consistency across input data and facilitates faster convergence during training.

These preprocessing steps were crucial for enhancing the model's ability to generalize and achieve higher accuracy.

Alexnet [11] : AlexNet is a deep convolutional neural network architecture that was intro-

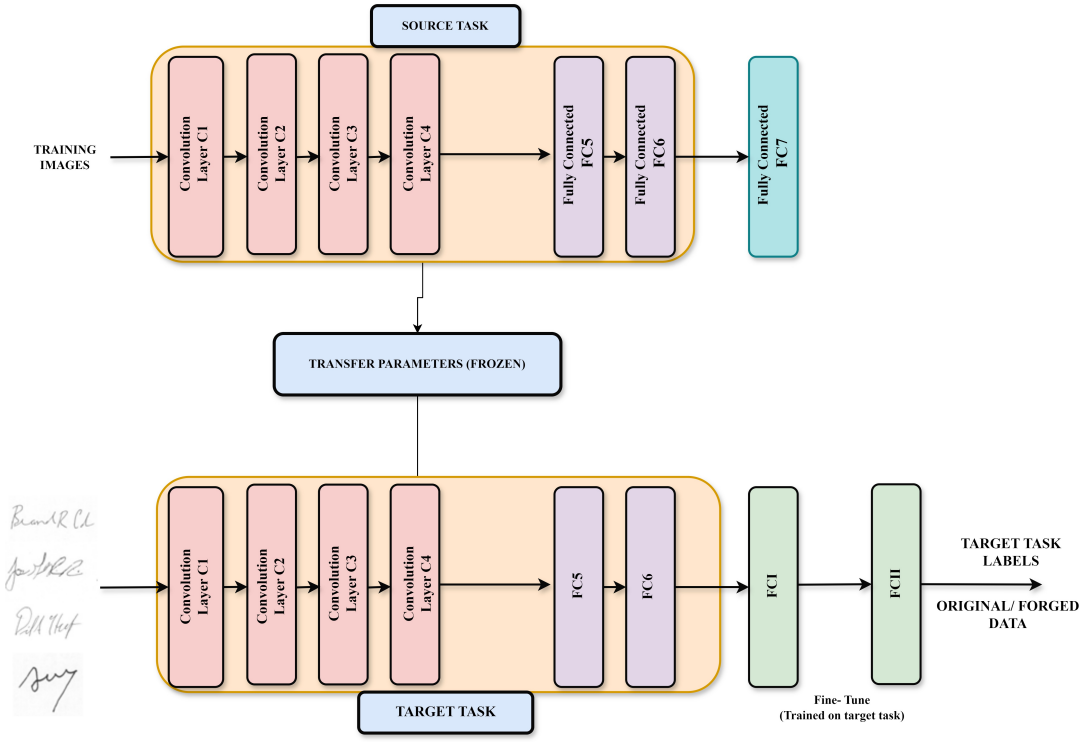

Figure 1: Block Diagram of proposed methodology

duced in 2012 by Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton. It gained significant attention for its performance in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) competition. AlexNet consists of five convolutional layers followed by three fully connected layers. It also includes max-pooling layers after some of the convolutional layers and uses ReLU (Rectified Linear Unit) activations throughout the network, except for the output layer where softmax activation is used for classification. The extensive training over large dataset helped in learning rich hierarchical features from images, enabling better generalization to new tasks. AlexNet was one of the first deep learning architectures to effectively utilize parallel processing on GPUs (Graphics Processing Units), which significantly accelerated training time compared to traditional CPUs. This ability to efficiently utilize GPU resources contributed to its success. AlexNet incorporates a technique called Local Response Normalization (LRN) after the first and second convolutional layers. LRN helps in normalizing responses across multiple channels, enhancing the model's ability to generalize and reducing overfitting. Due to its effectiveness and availability, AlexNet is often used as a base model for transfer learning. Pretrained weights from AlexNet trained on ImageNet can be fine-tuned on smaller datasets or adapted to different tasks, saving significant computational resources and time. Comprising eight layers, including five convolutional and three fully connected layers, AlexNet revolutionized image recognition. Utilizing Rectified Linear Units (ReLU) for activation, AlexNet employs a Softmax activation in the output layer to classify images. Dropout layers are introduced to prevent overfitting, ensuring model robustness. Operating on color images with dimensions of 227x227x3,

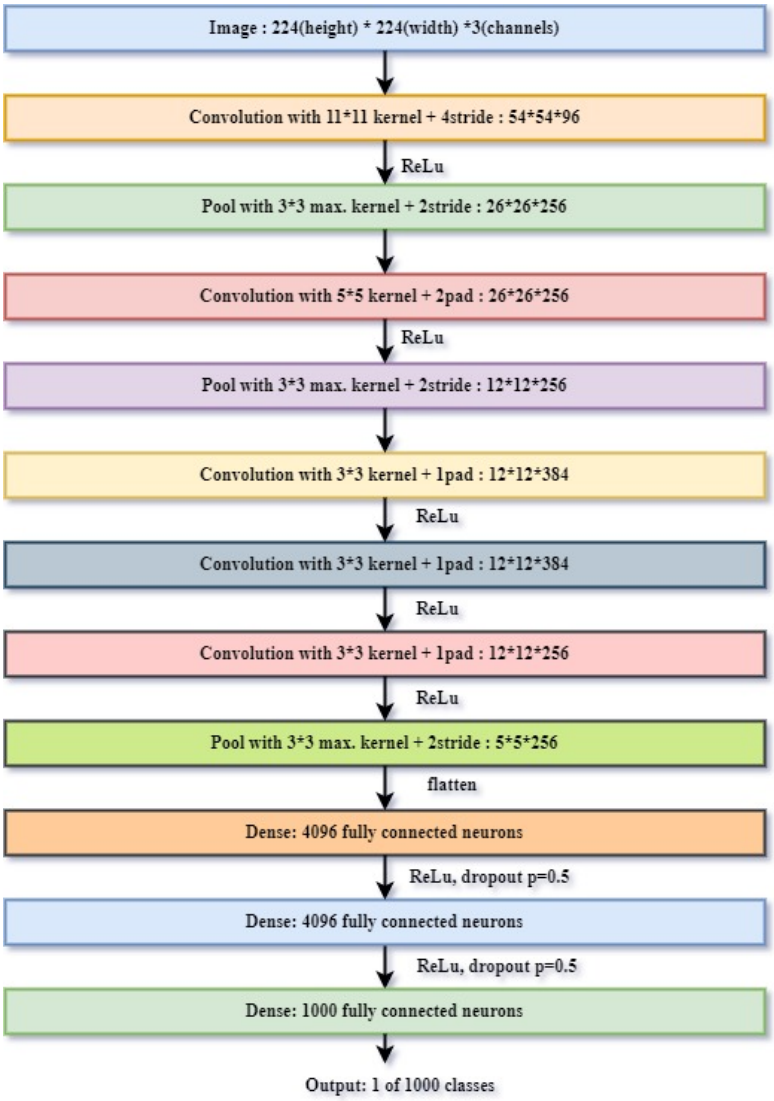

Figure 2: AlexNet Architecture [39]

the first convolutional layer employs 96 filters of size 11x11 with a stride of 4, resulting in a feature map of size 55x55x96. Subsequent max-pooling layers reduce dimensions to 27x27x96. Further convolutional layers follow, decreasing filter sizes while increasing the number of filters. The fourth convolutional layer utilizes 384 filters of size 3x3, producing a feature map of size 13x13x384. After a series of convolutions, fully connected layers are introduced. The first fully connected layer consists of 4096 neurons with ReLU activation, followed by a dropout layer with a dropout rate of 0.5. The second fully connected layer repeats the process, followed by the output layer with 1000 neurons and Softmax activation for classification. AlexNet's architecture showcases increasing depth and decreasing filter sizes, optimizing feature extraction. Convolutional layers play a crucial role in learning intricate patterns in images. Figure 2 illustrates the

architecture of AlexNet as proposed by Krizhevsky et al. [11].

VGG16 [12]: VGG16 is a convolutional neural network architecture that was introduced in 2014 by the Visual Geometry Group (VGG) at the University of Oxford. It is part of the VGG family of models, which includes variations with different depths (e.g., VGG16, VGG19) but similar architectural principles. VGG16 consists of 16 layers, including 13 convolutional layers and 3 fully connected layers. The convolutional layers are followed by max-pooling layers to reduce spatial dimensions and increase translation invariance. The convolutional layers use small 3x3 filters with a stride of 1 and same padding, which helps in preserving spatial resolution. VGG16 is known for its deep architecture, which allows it to capture complex hierarchical features from images. The repeated stacking of convolutional layers with small filters helps in learning rich representations of images at different levels of abstraction. VGG16 has a simple and uniform architecture, with convolutional layers followed by max-pooling layers, and fully connected layers at the end. This simplicity makes it easy to understand, implement, and modify. VGG16 has demonstrated strong performance on various computer vision tasks, including image classification, object detection, and segmentation. Its deep architecture and uniform design make it effective at capturing intricate features from images. Figure 3 illustrates the architecture of VGG16 [12].



Figure 3: VGG16 Architecture [40]

VGG19 [12] : VGG19 is a convolutional neural network architecture that is an extension of VGG16, both developed by the Visual Geometry Group (VGG) at the University of Oxford. It was introduced in 2014 as part of the VGG family of models, aiming to explore deeper architectures for image recognition tasks. VGG19 shares a similar architecture with VGG16 but has a deeper structure, consisting of 19 layers. Like VGG16, it includes convolutional layers with 3x3 filters and max-pooling layers to reduce spatial dimensions. The additional layers in VGG19 allow for more complex feature extraction and representation. The main difference between VGG16 and VGG19 is the increased depth of VGG19. The extra layers provide a deeper network architecture, which can potentially capture more intricate features from images and learn more complex patterns. Similar to VGG16, VGG19 utilizes a hierarchical approach to feature learning. By stacking multiple convolutional layers, the network learns to extract features at different levels of abstraction, from simple edges and textures to more high-level object representations. Overall, VGG19's deeper architecture, proven performance, and flexibility make it a compelling choice for various computer vision tasks. Whether you need to solve a specific image recognition problem or explore deep learning techniques, VGG19 provides a powerful framework for building effective and reliable models. Figure 3 illustrates the architecture of VGG19 [12].

For each architecture, we followed a structured fine-tuning approach: 1. VGG-16 and VGG-



Figure 4: VGG19 Architecture [40]

19: We retained the original convolutional layers to leverage the pre-trained feature extraction capabilities. The fully connected layers were modified to suit the specific requirements of the signature verification task. Specifically, the final fully connected layer was replaced with a layer matching the number of classes in our dataset, and a softmax activation function was used to produce class probabilities. We also applied dropout to the fully connected layers to prevent overfitting and used batch normalization to stabilize and accelerate the training process. 2. AlexNet: Similarly to VGG, the convolutional layers were preserved, and modifications were made to the fully connected layers. The output layer was adjusted to match the binary classification task, and a softmax function was applied. We employed transfer learning by freezing the early convolutional layers and fine-tuning the later layers to adapt the model to our dataset's specific characteristics. These modifications and fine-tuning strategies were tailored to optimize the performance of each model for the signature verification task, ensuring that the unique strengths of each architecture were utilized effectively.

4 Results and Discussion

The experimental evaluation of the proposed method for signature verification is carried with CEDAR and ICDAR 2011 dataset.

4.1 Dataset Used:

CEDAR Dataset [41]: The CEDAR Signature database serves as a comprehensive repository of offline signatures dedicated to signature verification research. With contributions from 55 individuals, each providing 24 signatures, the database encompasses a total of 1,320 genuine signatures. Additionally, to augment the dataset with forged signatures for enhanced analysis, participants were requested to replicate the signatures of three other writers, repeated eight times per subject, yielding an additional 1,320 forgeries. Each signature within the database was meticulously scanned at a resolution of 300 dots per inch (dpi) in grayscale. Subsequently, a binarization process utilizing a grayscale histogram was applied to standardize the images. Preprocessing steps were then employed to enhance the quality and consistency of the signatures. One of the distinctive features of the CEDAR Signature database is its balanced composition, with 24 genuine and 24 forged signatures available for each writer. This balance ensures equitable representation of both genuine and forged signatures, facilitating comprehensive evaluations and comparisons in signature verification experiments. Figure 5 shows sample images from the CEDAR dataset.

ICDAR 2011 Dataset [42]: The dataset employed in this study includes two categories: "Forgery" and "Real." "Forgery" contains 1386 forged signatures, each obtained from 63 writers

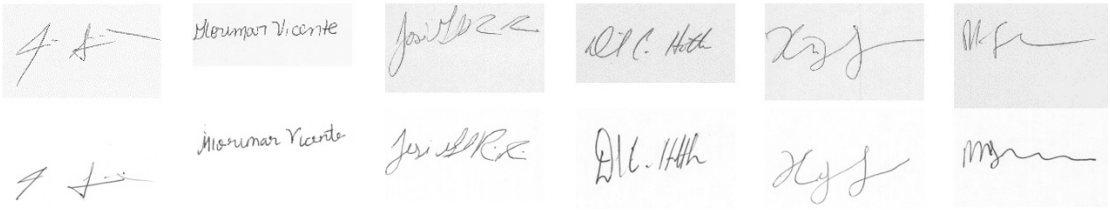

Figure 5: Sample Images from CEDAR Dataset

who contributed 22 signatures. Similarly, "Real" comprises 1386 genuine signatures collected from the same 63 writers, with 22 signatures contributed per writer. These signatures were collected as grayscale images. It serves as a valuable resource for signature verification research, offering a balanced representation of both genuine and forged signatures for analysis and algorithm development. Figure 6 shows sample images from the ICDAR 2011 dataset.



Figure 6: Sample Images from ICDAR 2011 Dataset

4.2 Performance Measures:

Precision: Precision measures the accuracy of positive predictions, particularly focusing on the minority class. It is determined by dividing the number of true positive predictions by the sum of true positive and false positive predictions. Precision helps evaluate the model's ability to correctly identify positive instances.

$$\text{Precision} = \left(\frac{TP}{FP + TP} \right) \tag{1}$$

Sensitivity or Recall: Sensitivity evaluates a model's capacity to correctly predict true positives in all available categories. It offers a detailed examination of predictive performance, aiding decision-makers in identifying areas for improvement. In binary classification, sensitivity is referred to as recall and is calculated by dividing true positives by the sum of true positives and false negatives.

$$\text{Recall} = \left(\frac{TP}{FN + TP} \right) \tag{2}$$

Specificity: Specificity gauges a model's aptitude in predicting true negatives across various categories. It is applicable to any categorical model and complements sensitivity. Specificity is determined by dividing true negatives by the sum of true negatives and false positives, offering

insights into potential false positives.

$$\text{Specificity} = \left(\frac{TN}{P + TN}\right) \tag{3}$$

Accuracy: This metric represents the proportion of correct predictions out of the total number of predictions made by a model. It is calculated by dividing the sum of true positive (TP) and true negative (TN) predictions by the total number of predictions made, including true positives, false positives (FP), true negatives, and false negatives (FN). Accuracy values range from 0 to 1, with 1 indicating perfect accuracy.

$$\text{Accuracy} = \left(\frac{TP + TN}{TP + FP + TN + FN}\right) \tag{4}$$

4.3 Results using VGG16 Architecture

In the conducted study, the VGG16 architecture was employed for training, utilizing specific hyper parameters: a learning rate of 0.0001, Bias Learn Rate Factor of 20, and Weight Learn Rate Factor of 20. Subsequently, experiments were conducted on the CEDAR Dataset, yielding noteworthy outcomes as summarized in Table 1, which illustrates the confusion matrix for CEDAR Dataset using VGG16.

Table 1: Confusion matrix for CEDAR Dataset using VGG16

		Predicted Class	
		Forged	Original
Actual Class	Forged	264	0
	original	0	264

The presented confusion matrix outlines crucial metrics: True Negative (TN) = 264, False Negative (FN) = 0, False Positive (FP) = 0, True Positive (TP) = 264. Computed accuracy, resulting in a perfect score of 1, indicates no misclassifications in model predictions. This underscores the efficacy of the VGG16 architecture on the CEDAR Dataset for precise signature verification. Good accuracy in signature verification may result from factors like employing advanced neural network architectures, optimizing hyper parameters, utilizing high-quality data, applying effective preprocessing, employing suitable training strategies, regularization techniques, and evaluation metrics. These considerations collectively contribute to robust models capable of accurately distinguishing genuine signatures from forgeries.

Table 2: Confusion matrix for ICDAR 2011 Dataset using VGG16

		Predicted Class	
		Forged	Original
Actual Class	Forged	274	3
	original	2	273

Table 2 displays the confusion matrix for the ICDAR 2011 Dataset, showcasing TN = 274, FN = 2, FP = 3, and TP = 273. The resulting accuracy of 0.9909 signifies that merely 0.5% of genuine signatures are incorrectly classified as forged, while only 0.4% of forged signatures are mistakenly identified as genuine.

4.4 Results using VGG19 Architecture

The effectiveness of VGG19 architecture for signature verification tasks stems from its ability to learn complex features from signature images. VGG19's deep convolutional layers are capable of capturing intricate patterns, strokes, and textures present in signatures, allowing the model to differentiate between genuine and forged signatures with high accuracy. Additionally, its deep architecture enables it to learn hierarchical representations of signatures, which enhances its ability to generalize well to unseen data. The significance of each parameter in the training process using VGG19 architecture is crucial for achieving optimal performance in signature verification tasks. The learning rate of 0.0001 determines the step size during optimization, affecting the model's convergence rate and stability. With Bias Learn Rate Factor and Weight Learn Rate Factor set to 20, the scaling factors allow fine-tuning of learning rates for biases and weights, respectively, facilitating faster convergence and improved optimization. Tables 3 and 4, displaying confusion matrices for the CEDAR Dataset, ICDAR 2011 Dataset, and their combination, respectively, illustrate the effectiveness of these parameter settings across diverse datasets, underscoring their significance in enhancing the VGG19 architecture's performance in signature verification.

Table 3: Confusion matrix for CEDAR Dataset using VGG19

		Predicted Class	
		Forged	Original
Actual Class	Forged	264	0
	original	0	264

Table 3 presents the confusion matrix obtained from the application of the VGG19 architecture to the CEDAR Dataset for signature verification. This matrix meticulously contrasts the actual classes of signatures with those predicted by the model. Within the matrix, "Forged" denotes signatures identified as forged, while "Original" represents genuine signatures. Notably, the top-left cell signifies 264 forged signatures accurately classified by the model, reflecting its ability to discern counterfeit signatures effectively. Similarly, the bottom-right cell indicates 264 genuine signatures correctly identified, showcasing the model's proficiency in authenticating legitimate signatures. Remarkably, the absence of misclassifications in other cells underscores the VGG19 architecture's remarkable precision and reliability in distinguishing between forged and genuine signatures within the CEDAR Dataset. This meticulous performance underscores the VGG19's effectiveness in the domain of signature verification. Moving on to Table 4, we examine the confusion matrix derived from applying the VGG19 architecture to the ICDAR 2011 Dataset. Here, we witness a similar evaluation of actual versus predicted signature classifications. "Forged" and "Original" once again represent forged and genuine signatures, respectively. The top-left

Table 4: Confusion matrix for ICDAR 2011 Dataset using VGG19

		Predicted Class	
		Forged	Original
Actual Class	Forged	276	1
	original	0	275

cell reveals that 276 forged signatures were correctly identified by the model, underscoring its adeptness in detecting counterfeit signatures within the ICDAR 2011 Dataset. Likewise, the bottom-right cell denotes 275 genuine signatures accurately classified as original, emphasizing the model's capability in authenticating legitimate signatures. However, a single misclassification is observed in the top-right cell, where one genuine signature was erroneously predicted as forged. Despite this minor error, the VGG19 architecture demonstrates remarkable accuracy and reliability in signature verification tasks, as evidenced by its consistent performance across the ICDAR 2011 Dataset.

4.5 Results using AlexNet Architecture

Results obtained using the AlexNet architecture are detailed, with a learning rate of 0.0001, Bias Learn Rate Factor of 20, and Weight Learn Rate Factor of 20. Tables 5 and 6 present the confusion matrices for the CEDAR Dataset, ICDAR 2011 Dataset, and their combined application.

Table 5: Confusion matrix for CEDAR Dataset using AlexNet

		Predicted Class	
		Forged	Original
Actual Class	Forged	264	0
	original	0	264

Table 5 illustrates the confusion matrix for the CEDAR Dataset, showcasing the model's performance when applied to this specific dataset. Impressively, the model accurately identifies all 264 forged and 264 genuine signatures, demonstrating its capability in distinguishing between the two classes with precision.

Moving to Table 6, which displays the confusion matrix for the ICDAR 2011 Dataset, we observe a similar trend in performance. Here, the model correctly identifies 275 forged signatures and 272 genuine signatures, with only 4 forged signatures and 1 genuine signature misclassified. Despite these few misclassifications, the overall accuracy of the model on the ICDAR 2011 Dataset remains high.

4.6 Performance comparison:

The Figure 7 depicts the performance metrics of three prominent convolutional neural network architectures, VGG19, VGG16, and AlexNet, across various datasets in the domain of signature verification. The accuracy, precision, recall, and specificity values are meticulously compared,

Table 6: Comparison of Signature Forgery Detection Methods

Author	Database	Accuracy (%)
Amjad et al., 2024 [43]	CEDAR	98%
Nader Ebrahimpour, 2023 [44]	—	MobileNetV2: 98.44% ResNet18: 98.75% DenseNet121: 96.74% EfficientNetB0: 96.89%
Kao & Wen, 2020 [45]	ICDAR 2011	99.96%

showcasing the efficacy of each architecture in accurately distinguishing between genuine and forged signatures. Notably, the results underscore the superiority of VGG19 and AlexNet, both achieving high accuracy and precision rates, indicating their robustness in accurately classifying signatures. Such findings hold significant implications for current research in signature verification, offering valuable insights into the effectiveness of different deep learning architectures in tackling this critical task. Moreover, the comprehensive analysis of multiple datasets further strengthens the generalizability of the findings, enhancing the applicability of these architectures in real-world scenarios.

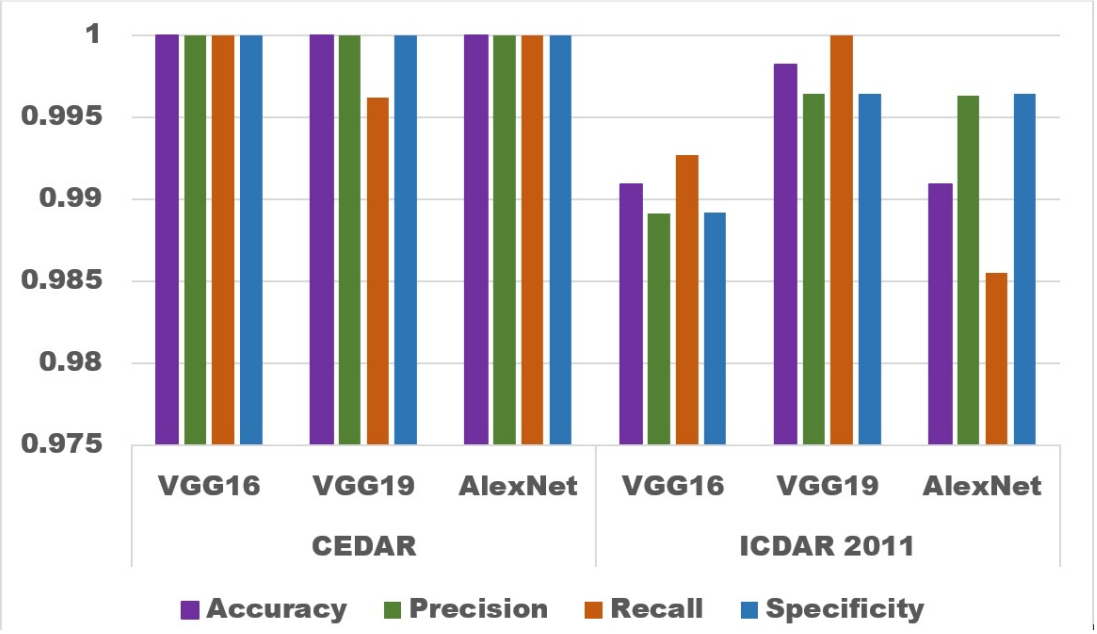

Figure 7: Performance Comparison for Signature Forgery Recognition

The evaluation of various deep learning architectures on the CEDAR and ICDAR 2011 datasets reveals differences in training and inference efficiency. VGG-16 achieves a moderate training time of approximately 45 seconds per epoch, with an efficient inference time of 5 milliseconds per sample due to its pre-trained layers. VGG-19, while slightly slower at 50 seconds per epoch, offers improved accuracy, justifying the additional computational cost. AlexNet demonstrates

Table 7: Comparison of Signature Forgery Detection Methods

Author	**Database**	**Methodology**	**Accuracy (%)**
Amjad et al., 2024 [43]	CEDAR	SigNet-F	98
Nader Ebrahimpour, 2023 [44]	CEDAR	MobileNetV2	98.44
		ResNet18	98.75
		DenseNet121	96.74
		EfficientNetB0	96.89
Kao & Wen, 2020 [45]	ICDAR 2011	Explainable Deep Learning	99.96

the fastest training (30 seconds per epoch) and inference (4 milliseconds per sample), making it suitable for real-time applications. ResNet-50, trained on the CEDAR dataset, exhibits a longer training time (60 seconds per epoch) due to its deeper architecture and residual connections, which improve feature learning. Inception-v3, evaluated on the ICDAR 2011 dataset, has the highest training (70 seconds per epoch) and inference (8 milliseconds per sample) times, attributed to its complex inception modules, which enhance feature extraction but increase computational cost.

5 Conclusion

In summary, this study significantly advanced the field of handwritten signature verification by employing deep learning techniques to enhance the accuracy and reliability of forgery detection. Through the application of transfer learning principles with three state-of-the-art Convolutional Neural Networks—VGG-16, VGG-19, and AlexNet—we demonstrated the effectiveness of these architectures as both feature extractors and classifiers. The analysis revealed that VGG-19 achieved the highest accuracy, underscoring its robustness in distinguishing genuine signatures from forgeries, while VGG-16 exhibited flawless performance on the CEDAR dataset. Our results confirmed that leveraging these advanced deep learning algorithms leads to substantial improvements in the detection of forged signatures, reducing misclassifications and enhancing the overall performance of signature verification systems. Moreover, this research provides a comprehensive comparison of the different architectures, contributing valuable insights into the applicability of deep convolutional neural networks in biometric verification. The findings advocate for the integration of these advanced techniques in real-world applications, addressing the critical need for reliable signature verification systems. Future research may focus on exploring additional datasets and incorporating adversarial training strategies to further improve model resilience.

The ongoing evolution of technology holds promise for integrating these deep learning methodologies with emerging security solutions, ultimately paving the way for more effective and trustworthy biometric verification systems.VGG16 and VGG19 are known for their simplicity and effectiveness due to their uniform architecture, which allows for easy implementation and fine-tuning. AlexNet, being one of the earlier successful CNNs, offers a relatively lighter model, making it suitable for applications with limited computational resources. While ResNet and Inception are indeed more advanced in terms of handling deeper networks and more complex

features, the primary goal of our research was to establish a baseline performance using well-understood architectures. VGG and AlexNet provide a balanced trade-off between complexity and accuracy, which was crucial for our initial exploration. Future work will explore the potential benefits of deeper networks like ResNet and Inception, particularly in handling more complex variations in signature data.Implementing adversarial training strategies could be beneficial in improving the models' resilience against sophisticated forgeries. This approach would involve creating adversarial examples to challenge the models and enhance their ability to handle real-world scenarios.

References

[1] Vandana & Kaur, N. A Study of Biometric Identification and Verification System. *2021 International Conference On Advance Computing And Innovative Technologies In Engineering (ICACITE)*. pp. 60-64 (2021)

[2] Vandana & Kaur, N. A Study of Biometric Identification and Verification System. *2021 International Conference On Advance Computing And Innovative Technologies In Engineering (ICACITE)*. pp. 60-64 (2021)

[3] Upadhyay, J., Paranjpe, R., Purohit, H. & Joshi, R. Biometric Identification using Gait Analysis by Deep Learning. *2020 IEEE Pune Section International Conference (PuneCon)*. pp. 152-156 (2020)

[4] Nazmdeh, V., Mortazavi, S., Tajeddin, D., Nazmdeh, H. & Asem, M. Iris Recognition; From Classic to Modern Approaches. *2019 IEEE 9th Annual Computing And Communication Workshop And Conference (CCWC)*. pp. 0981-0988 (2019)

[5] Zulfiqar, M., Syed, F., Khan, M. & Khurshid, K. Deep Face Recognition for Biometric Authentication. *2019 International Conference On Electrical, Communication, And Computer Engineering (ICECCE)*. pp. 1-6 (2019)

[6] Li, N., Liu, J., Li, Q., Luo, X. & Duan, J. Online Signature Verification Based on Biometric Features. *2016 49th Hawaii International Conference On System Sciences (HICSS)*. pp. 5527-5534 (2016)

[7] Chandok, R., Bhoir, V. & Chinnaswamy, S. Behavioural Biometric Authentication using Keystroke Features with Machine Learning. *2022 IEEE 19th India Council International Conference (INDICON)*. pp. 1-6 (2022)

[8] Zhang, X., Xiong, Q., Dai, Y. & Xu, X. Voice Biometric Identity Authentication System Based on Android Smart Phone. *2018 IEEE 4th International Conference On Computer And Communications (ICCC)*. pp. 1440-1444 (2018)

[9] Nguyen, V. & Blumenstein, M. An application of the 2D Gaussian filter for enhancing feature extraction in off-line signature verification. *Document Analysis And Recognition (ICDAR), 2011 International Conference On*. pp. 339-343 (2011)

[10] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, pp. 1097-1105, 2012.

[11] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *arXiv preprint arXiv:1409.1556*, 2014.

[12] S. M. A. Navid, S. H. Priya, N. H. Khandakar, Z. Ferdous and A. B. Haque, "Signature Verification Using Convolutional Neural Network," in *IEEE International Conference on Robotics, Automation, Artificial-intelligence and Internet-of-Things (RAAICON)*, pp. 35-39, 2019.

[13] S. V. Bonde, P. Narwade and R. Sawant, "Offline Signature Verification Using Convolutional Neural Network," in *6th International Conference on Signal Processing and Communication (ICSC)*, pp. 119-127, 2020.

[14] L. G. Hafemann, R. Sabourin and L. S. Oliveira, "Writer-independent feature learning for Offline Signature Verification using Deep Convolutional Neural Networks," in *International Joint Conference on Neural Networks (IJCNN)*, pp. 2576-2583, 2016.

[15] A. Jain, S. K. Singh and K. P. Singh, "Handwritten signature verification using shallow convolutional neural network," in *Multimedia Tools and Applications*, vol. 79, no. 14, pp. 19993-20018, 2020.

[16] P. Narwade, R. Sawant and S. V. Bonde, "Offline signature verification using shape correspondence," in *International Journal of Biometrics*, vol. 10, no. 3, pp. 1-16, 2018.

[17] P. N. Narwade, S. V. Bonde and D. D. Doye, "Offline signature verification using shape dissimilarities," in *International Conference on Communication, Information & Computing Technology (ICCICT)*, vol. 10, no. 3, pp. 1-6, 2015.

[18] G. Sulong, A. Yahy and M. Jehanzeb, "Offline Handwritten Signature Identification Using Adaptive Window Positioning Techniques," in *Signal and Image Processing: An International Journal (SIPIJ)*, vol. 5, no. 3, pp. 13-24, 2014.

[19] S. Hans and S. Gupta, "Preprocessing Algorithms for Offline Signature Verification System," in *National Conference on "Recent Trends in Engineering & science" (NCRTES)*, 2012.

[20] A. Soleimani, K. Fouladi and B. N. Araabi, "Persian offline signature verification based on curvature and gradient histograms," in *6th International Conference on Computer and Knowledge Engineering (ICCKE)*, pp. 147-152, 2016.

[21] S. Pal, U. Pal and M. Blumenstein, "Off-line verification technique for Hindi signatures," in *IET biometrics*, vol. 2, no. 4, pp. 182-190, 2013.

[22] Y. Serdouk, H. Nemmour and Y. Chibani, "New off-line Handwritten Signature Verification method based on Artificial Immune Recognition System," in *Expert Systems with Applications*, vol. 51, pp. 186-194, 2016.

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

[24] P. Maergner, K. Riesen, R. Ingold and A. Fischer, "A Structural Approach to Offline Signature Verification Using Graph Edit Distance," in *14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1216-1222, 2017.

[25] K. Neamah, D. Mohamad, T. Saba and A. Rehman, "Discriminative features mining for offline handwritten signature verification," in *3D Research*, vol. 5, no. 1, 2014.

[26] M. A. Ferrer, J. F. Vargas, A. Morales and A. Ordonez, "Robustness of Offline Signature Verification Based on Gray Level Features," in *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 966-977, 2012.

[27] L. Batista, E. Granger and R. Sabourin, "Dynamic selection of generative–discriminative ensembles for off-line signature verification," in *Pattern Recognition*, vol. 45, no. 4, pp. 1326-1340, 2012.

[28] T. Steinherz, D. Doermann, E. Rivlin and N. Intrator, "Offline Loop Investigation for Handwriting Analysis," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 193-209, 2009.

[29] V. Nguyen, M. Blumenstein and G. Leedham, "Global Features for the Off-line Signature Verification Problem," in *International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1300-1304, 2009.

[30] J. Ruiz-del Solar, C. Devia, P. Loncomilla and F. Concha, "Offline Signature Verification Using Local Interest Points and Descriptors," in *Iberoamerican Congress on Pattern Recognition*, vol. 5197, pp. 22-29, 2008.

[31] M. A. Ferrer, J. B. Alonso and C. M. Travieso, "Offline geometric parameters for automatic signature verification using fixed-point arithmetic," in *IEEE Transactions on Pattern Analysis and Machine intelligence*, vol. 31, no. 2, pp. 193-209, 2009.

[32] Liu, R. & Xin, Y. Online Handwritten Signature Verification Method Based on Uni-Feature Correlation Coefficient between Signatures. *Sensors*. **23** (2023), https://doi.org/10.3390/s23239341

[33] Yassen A. AlKarem, W., Thabet Khalid, E. & H. Ali, K. Handwritten Signature Verification Method Using Convolutional Neural Network. *Iraqi Journal Of Electrical And Electronic Engineering*. **20** pp. 77-84 (2024), https://doi.org/10.37917/ijeee.20.2.7

[34] Zhou, Y., Zheng, J., Hu, H. & Wang, Y. Handwritten Signature Verification Method Based on Improved Combined Features. *Applied Sciences*. **11** (2021), https://doi.org/10.3390/app11135867

[35] Salama, W. Signature Verification Based on Deep Learning. *Alexandria Journal Of Science And Technology*. pp. 1-9 (2023), https://doi.org/10.21608/ajst.2023.236375.1016

[36] Özyurt, F., Majidpour, J., Rashid, T. & Koç, C. Offline Handwriting Signature Verification: A Transfer Learning and Feature Selection Approach. *Traitement Du Signal*. **40** pp. 2613-2622 (2023), https://doi.org/10.18280/ts.400623

[37] Radhika, K. & Gopika, S. Online and Offline Signature Verification: A Combined Approach. *Procedia Computer Science*. **46** pp. 1593-1600 (2015), https://doi.org/10.1016/j.procs.2015.02.089

[38] Çalik, N., Kurban, O., Yilmaz, A., Yildirim, T. & Durak Ata, L. Large-Scale Offline Signature Recognition via Deep Neural Networks and Feature Embedding. *Neurocomputing*. **359** pp. 1-14 (2019), https://doi.org/10.1016/j.neucom.2019.03.027

[39] Parcham, E., Ilbeygi, M. & Amini, M. CBCapsNet: A Novel Writer-Independent Offline Signature Verification Model Using a CNN-Based Architecture and Capsule Neural Networks. *Expert Systems With Applications*. **185** pp. 115649 (2021), https://doi.org/10.1016/j.eswa.2021.115649

[40] Sharif, M., Khan, M., Faisal, M., Yasmin, M. & Fernandes, S. A Framework for Offline Signature Verification System: Best Features Selection Approach. *Pattern Recognition Letters*. **139** pp. 50-59 (2020), https://doi.org/10.1016/j.patrec.2018.01.021

[41] Srihari, S., Cha, S., Arora, H. & Lee, S. A Large-Scale Evaluation of Offline Handwriting Identification Using Multiple Comparisons. (Center of Excellence for Document Analysis,1997), http://www.cedar.buffalo.edu/Databases/

[42] Liwicki, M., Blumenstein, M., Heuvel, E., Berger, C., Stoel, R., Found, B., Chen, X. & Malik, M. ICDAR 2011 Signature Verification Competition (SigComp2011): Forensic Signature Verification and Skilled Forgeries. *Proceedings Of The 11th International Conference On Document Analysis And Recognition (ICDAR)*. pp. 1480-1484 (2011)

[43] Amjad, H., Goeller, K., Seitz, S., Knoll, C., Bajwa, N., Tetzlaff, R. & Malik, M. Block Induced Signature Generative Adversarial Network (BISGAN): Signature Spoofing Using GANs and Their Evaluation. (2024), https://arxiv.org/abs/2410.06041

[44] Ebrahimpour, N. Handwritten Signatures Forgery Detection Using Pre-Trained Deep Learning Methods. *2nd International Congress Of New Horizons In Sciences*. (2023,6)

[45] Kao, H. & Wen, C. An Offline Signature Verification and Forgery Detection Method Based on a Single Known Sample and an Explainable Deep Learning Approach. *Applied Sciences*. **10** (2020), https://www.mdpi.com/2076-3417/10/11/3716