PAPER

Deep learning approaches for online signature authentication: a comparative study of pre-trained CNN models

To cite this article: M Ranga Swamy *et al* 2025 *Eng. Res. Express* **7** 015230

Engineering Research Express

PAPER

Deep learning approaches for online signature authentication: a comparative study of pre-trained CNN models

RECEIVED
27 September 2024

REVISED
17 December 2024

ACCEPTED FOR PUBLICATION
9 January 2025

PUBLISHED
21 January 2025

M Ranga Swamy, Vijayalakshmi P* and V Rajendran

Department of Electronics and Communication Engineering, Vels Institute of Science, Technology and Advanced Studies, Chennai, Tamil Nadu, India

* Author to whom any correspondence should be addressed.

E-mail: ranga1310@gmail.com, viji.se@velsuniv.ac.in and drvrajen@gmail.com

Keywords: convolutional neural network (CNN) model, VGG16, xception, ResNet50, optimizers

Abstract

Authorization is essential for handling document assurance and security. Nowadays, it constitutes one of the top responsibilities for securing information and effectiveness in every domain. Technological advances have made interactions with machinery more effortless. As a result, the demand for authentication for various legitimate causes is growing rapidly. Therefore, biometric-based identification has dramatically accelerated. This is an improvement over the other approaches. The present work is intended to apply convolutional neural networks for mining features and supervised machine-learning techniques to verify handwritten signatures. Raw images of signatures were used to train the CNN models for feature extraction and data augmentation. In the present work, pre-trained CNN models, such as VGG16, Inception-v3, ResNet50, and Xception, were used to separate authentic from fake signatures. Supervised learning methods, including Logistic Regression and SVM, were used to classify features. The test data were obtained from the ICDAR 2011 Signature Dataset. The results obtained from the present work showed a clear improvement over traditional methods over 69 different signatures. VGG-16 with RMSProp achieved an impressive validation accuracy of 83%, demonstrating robustness with minimal overfitting. Compared with existing techniques, the proposed deep learning approach proved to be more accurate and reliable for signature verification.

1. Introduction

Signature verification and forgery detection are crucial steps for authenticating an individual's identity and confirming the legitimacy of their signatures [1]. The authenticity of signatures is a prominent issue in various sectors, such as finance, legal documents, and contracts. It is accepted as one of the most widely used biometric authentication approaches. There are two primary types of signature verification: static, offline, dynamic, and online [2], which are often used. In static verification, signatures are written on paper and scanned for further testing, capturing only the image of the signature without considering any dynamic characteristic features [2]. In contrast, dynamic or online verification occurs when a person signs digitally on an electronic gadget, such as an iPad or tablet [3]. This method captures not only the visual signature but also dynamic traits, such as pressure, speed, and stroke patterns, adding an extra layer of security and reliability [4] to increase the accuracy of the signature verification procedure.

Although static verification is one of the basic verification methods and can be sufficient for initial identity validation, online verification offers a supplementary comprehensive approach, making it ideal for applications requiring high-level security. The present study focuses on an online signature verification method with the primary objective of developing a robust real-time signature verification system using advanced deep learning techniques, leveraging pre-trained Convolutional Neural Networks (CNNs) such as VGG16, Inception-v3, ResNet50, and Xception, which are optimized to extract unique features from signatures for enhanced classification [5].

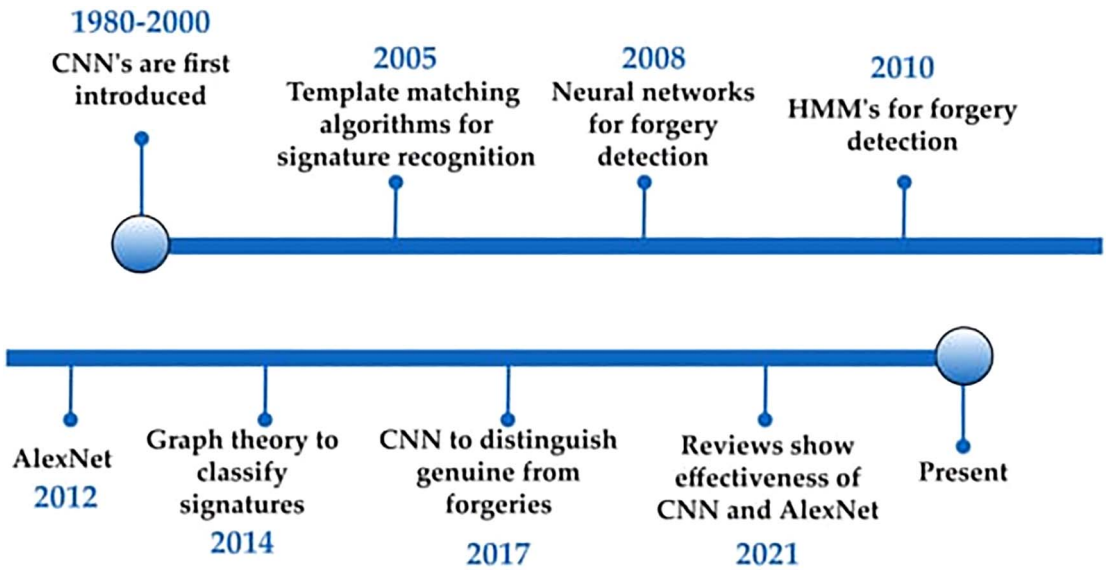

Figure 1. Overall signature detection survey during the year 1980-present [7].

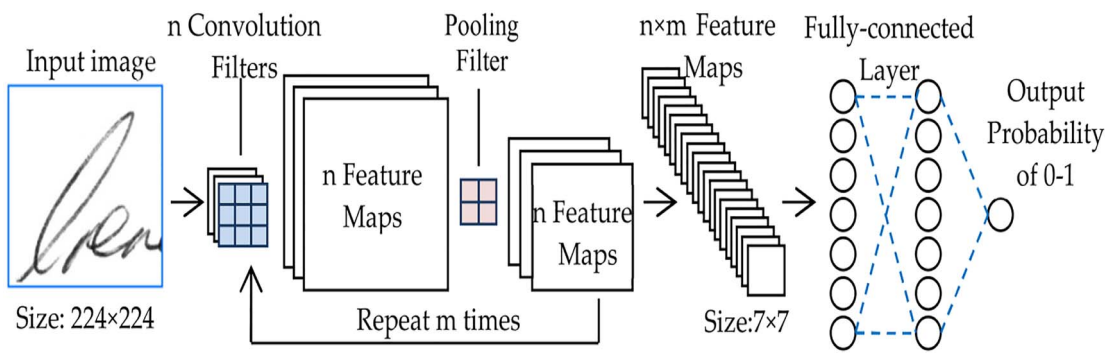

Figure 2. Framework of convolutional neural network [9].

The present work also uses optimization algorithms such as Adam, RMSProp, and SGD to improve the model performance further, ensuring high accuracy in distinguishing between genuine and forged signatures. These innovations enable the system to adapt to varying signature styles, overcoming the challenges posed by intrapersonal variations in signing behavior [6]. The present work can solve the problems of online signature verification in numerous real-world applications in which quick, reliable, and secure identity verification is paramount, including banking, e-commerce, and legal documentation. By utilizing cutting-edge CNN models and optimization techniques, this study aims to offer a scalable and efficient solution for real-time online signature authentication, ensuring both security and ease of use in biometric-based identity systems.

1.1. Problem statement

Since the early 1990 s, significant research has been conducted to address the challenges of offsite handwritten signature identification using the available techniques. Different methods have been investigated, with notable contributions made by Jose Lopes *et al* [7], who established multiple approaches to enhance the accuracy of signature verification. Despite these advancements in techniques for verifying signatures, many challenges remain in achieving high accuracy and reliability for detecting forgeries [8]. Figure 1 shows an overview of important milestones and breakthroughs in signature verification practices over the years.

Moreover, Hsin *et al* [9] used a CNN approach to verify offline signatures. They identified forgery signatures appropriate for various business circumstances, such as bank check payment sign verification procedures based on human assessment. The authors developed a CNN framework as shown in figure 2.

These layers, known as Convolutional Girshick [10], use numerous convolution filtering methods (or convolution kernels) to separate more advanced data from lower-level data, including identifying boundaries, angles, connecting scores, and numerous other image characteristics. Here, the author employs a pooling layer

in the order of Schener *et al* [11] to lower the characteristic map dimensions, which results in faster convergence rates for connections, because using numerous filters for convolution significantly increases the overall dimension of the characteristic image and must be accompanied by tiresome computations [12]. FCC layers then receive all multifaceted characteristic mappings as feeds in the form of a single-dimensional vector of features to produce predicted classes for subsequent categorization assignments. An adequately connected layer, an ordinary perceptron with multiple layers (MLP), was employed in a previous study [13].

Moreover, the Inception v1 and Inception v3 models, along with a CNN, were utilized by Jahandada *et al* [14] to verify the offline signatures. In this study, we propose the VGG-16, Inception V3, ResNet 50, and Xception models in the present experiments because they are well designed, exposing the immense capability of VGG16 to identify online forgery signatures. Moreover, optimization algorithms include SGD, RMS Prop, Adagrad, and Adam to obtain an optimal solution for detecting and verifying whether a signature is forgery.

The main intention of this research work is mentioned as follows

- Establishing a signature authentication approach using the most recent advances in deep-based approaches, particularly the CNN model.
- The novel signature dataset is sufficient for training the neural network-based approach for signature-based authentication.
- The system accepts a combination of identical fingerprints in the PNG appearance and returns a Boolean expression of either 1 or 0.

1.2. Feature extraction

In this study, CNNs automated feature extraction by capturing complex hierarchical patterns within signature images, such as stroke thickness, angles, and curvatures. Unlike traditional manual methods, which require human-defined features (e.g., texture or shape descriptors), CNNs dynamically learn relevant features directly from data, enhancing effectiveness by adapting to subtle variations in signatures. This automation reduces preprocessing time and improves efficiency, especially with complex datasets, by eliminating the need for handcrafted features. CNN-extracted features generally yield higher accuracy and robustness, particularly in tasks requiring fine details, making them superior for signature verification [15].

1.3. Research background

Several online and offline investigations have been conducted on signature verification using various techniques [16]. This survey explains signature verification using deep networks. Alajrami *et al* [17] applied a CNN approach for detecting offline signatures with a test accuracy of 99.7%. In contrast, Fayyaz *et al* [18] used a Gaussian distribution for finger vein detection by extracting features based on autoencoders. Fayyaz [19] showed that reducing the error rate also enhanced the accuracy range in online signature verification. Ghosh [20] compared the evaluation of signature verification using a CNN and Recurrent Neural Network approach. In 2016, Kim *et al* [21] found that verifying signatures using a CNN was significantly better. In 2018, signature verification was performed using a Hidden Markov Model [22]. Signature verification via handwritten/offline was performed by Soelistio *et al* using deep learning [23] and Poddar *et al* [24]. Menotti *et al* [25] applied CNN for signature spoofing verification, ANN by Adewole [26], CNN by Zhang *et al* [27]. Sadkhan *et al* analyzed various signatures [28], biometric recognition by sherin *et al* [29], ISRSAC was determined by yang *et al* [30], and Deep Air Segmentation by Malik *et al* [31], Fully Connected layers are appropriate for detecting signatures using an FCNN.

2. Proposed method

A scribbled signatory is a cognitive fingerprint because it depends on the behavioral rather than specific physiological aspects of the person's signatures. The examination and approval of an autograph might require quite a while; the signature of someone changes with duration, leading mistakes to occasionally increase. Increased incorrect rejection percentages resulted from mismatched signatures for signers who were not doing so consistently.

The selection of VGG16, Inception-v3, ResNet50, and Xception was based on their distinct architecture, proven performance in feature extraction, and generalizability in image-classification tasks. VGG16 offers simplicity and depth, capturing intricate details, while Inception-v3's mixed convolutions improve the efficiency in handling varied signature features [32]. ResNet50's skip connections allow deeper learning without gradient issues, thereby enhancing the complex feature recognition [33]. Xception's depth-wise separable convolutions capture finer patterns, which are ideal for distinguishing subtle variations in signatures [34]. The

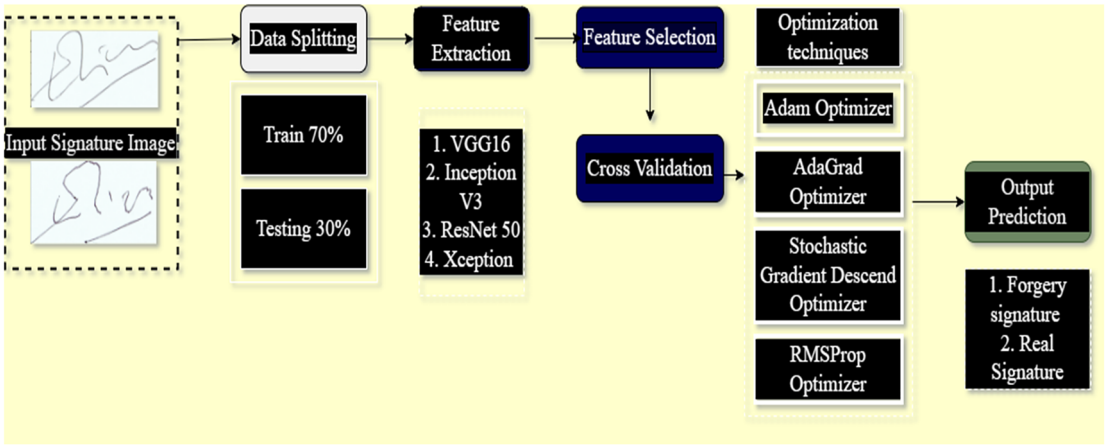

Figure 3. Proposed architecture for signature authentication [35].

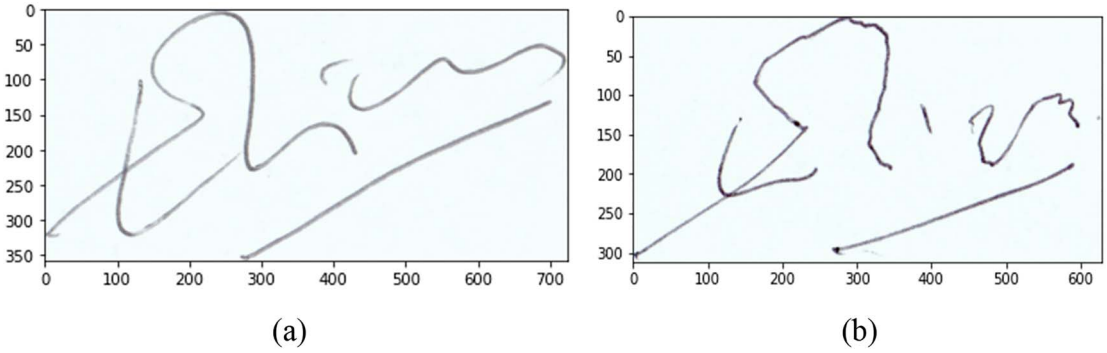

Figure 4. (a) Raw input image, (b) forgery signature image [35].

framework of our proposed model is depicted in figure 3, where the signatures are verified using deep-based optimization techniques.

2.1. Dataset

To build an understanding database for all individual's, written signatures were gathered, and distinctive elements were retrieved. A standard database of each person's signature is required to assess the effectiveness of the confirmation of the signature system and to compare the results of other approaches to the same database. Figures 4(a) and (b) illustrate examples of indivisible genuine and fake signatures, respectively. In this study, the author utilized approximately 600 signature images gathered from 69 subjects, 420 real and 180 forged signatures per person. These signature image datasets were collected from ICDAR 2011 [36] and are described in the RGB format.

The image on the left represents the raw input image and original signature signed by an individual. In contrast, the image on the right corresponds to a forgery signature image signed by unauthorized users. By comparing the images in figures 4 (a) and (b), the fake/forgery signature can be quickly identified [36], which is illegally supported in several domains such as cash withdrawals from banks, land registration, and field-based documents.

2.2. Feature extraction

The crucial phase in the digital signature authentication procedure is feature extraction, which is typically divided into two categories: manually created characteristic extraction, and pattern systems for learning Hafemann *et al* [37]People have developed tools for manually extracting feature strategies according to their perceptions. Various review publications have examined the handmade extraction and classification of feature approaches for verifying signatures [38]. Deng *et al* [39] used a wavelet-based feature extractor to ascertain the bending features of the fingerprints. Pal *et al* [40] selected uniform local binary patterns (ULBP) and local binary patterns (LBP) as their method for texture-based feature extraction.

Architecture	VGG16	Inception-v3	ResNet-50	Xception
Parameters	138M	24M	23M	23M
Features	512	2048	2048	2048

Figure 5. Parameters & features used for detecting signature [35].

However, feature-learning techniques may obtain characteristics that are devoid of human manipulation. Compared to manually created characteristics, this approach, also known as CNN, along with other deep learning approaches, has demonstrated outstanding efficacy across a wide range of applications related to computer vision. To learn the features for author categorization from signature picture pixels, Khalajzadeh *et al* presented an extensive CNN approach. A CNN-based technique that can acquire reliable characteristics using variable-sized signatures was proposed by Hafemann *et al* [37].

A Conventional Neural Network (CNN) is the most significant architecture applicable for functioning behind image-based input data. In this study, 16 models were trained to compare the accuracy of the signature authentication dataset in identifying forgery images. Among the 16 models, four pre-trained models were used for feature extraction.

- VGG16 model
- Inception-v3
- ResNet-50
- Xception

Moreover, Optimizers used to compile the models are mentioned as

- Stochastic gradient descent (SGD)
- Root Mean Square Propagation (RMSprop)
- Adaptive Gradient Algorithm (Adagrad)
- Active Design and Analysis Modelling (Adam)

2.3. Parameters selection

Here, the parameters for all models, such as VGG16, Inception V3, ResNet-50, and Xception architecture as 138M, 24M, 23M, and 23M, along with the features 512, 2048, 2048, and 2048, as depicted in figure 5.

2.4. Accuracy score in 3-folds

Here, 3-fold Cross-validation was utilized directly to perform model selection using deep learning-based pre-trained models, and optimizers such as SGD, RMSProp, Adagrad, and Adam were used to obtain the optimal solution for signature authentication, as shown in figure 6.

As shown in figures 7, 3-fold cross-validation was performed to evaluate the model performance based on metrics such as validation accuracy and loss, in which VGG16 + RMSProp optimizers had a training accuracy of 96.4%. In contrast, the validation accuracy was 97.17%, and the VGG16 + Adam optimizer achieved a training accuracy of 95.8%; however, the validation accuracy reached 95.56%. Similarly, we compare the training and validation loss among various models in which the ResNet 50 + RMSProp minimum loss during training is 0.005, whereas losses during validation of signature images are 0.67. The minimum loss during validation of the signature image dataset in which the VGG16+ RMSProp model reaches 0.07 by evaluating the models.

Based on this evaluation, the VGG-16 model and RMSrop optimizer attained a maximum accuracy of approximately 97% with the least loss of 0.07 in verifying signature images and classifying them as authentic or forgery.

In the present work. The deep-learning model employed for training was used to cross the estimator parameters of the validation function. The actual value is considered parameter X. The target variable is passed as parameter y. Thus, metrics such as validation accuracy and loss were evaluated and entered into the parameter score. Finally, the author must launch a set of measures, such as accuracy and loss, that are appropriate for validating our model.

	Optimizers			
	SGD	RMSprop	Adagrad	Adam
VGG16	0.8648	**0.9645**	0.8821	**0.9584**
Inception-v3	0.8042	**0.9827**	**0.9567**	**0.9922**
ResNet50	**0.9515**	**0.9991**	**0.9991**	**0.9974**
Xception	0.7730	**0.9835**	0.8215	**0.9939**

Training Accuracy (3-Fold)

	Optimizers			
	SGD	RMSprop	Adagrad	Adam
VGG16	**0.7091**	**0.9717**	0.5111	**0.9556**
Inception-v3	0.5818	0.4202	**0.6020**	**0.6323**
ResNet50	0.4182	0.5879	0.5818	0.4182
Xception	0.5697	0.5818	0.5657	0.5899

Validation Accuracy (3-Fold)

	Optimizers			
	SGD	RMSprop	Adagrad	Adam
VGG16	**0.4497**	**0.0918**	0.3716	**0.1069**
Inception-v3	0.4485	**0.0448**	0.2218	**0.0176**
ResNet50	0.1561	**0.0050**	**0.0324**	**0.0084**
Xception	0.5424	**0.0642**	0.4889	**0.0221**

Training loss (3-Fold)

	Optimizers			
	SGD	RMSprop	Adagrad	Adam
VGG16	**0.5971**	**0.0793**	**0.9206**	**0.1127**
Inception-v3	**0.7371**	8.5688	**0.7872**	2.3959
ResNet50	1.2646	**0.6738**	1.4782	**0.7494**
Xception	**0.7339**	7.0186	**0.7754**	3.2455

Validation loss (3-Fold)

Figure 6. 3-Fold cross-validation using deep learning [35].

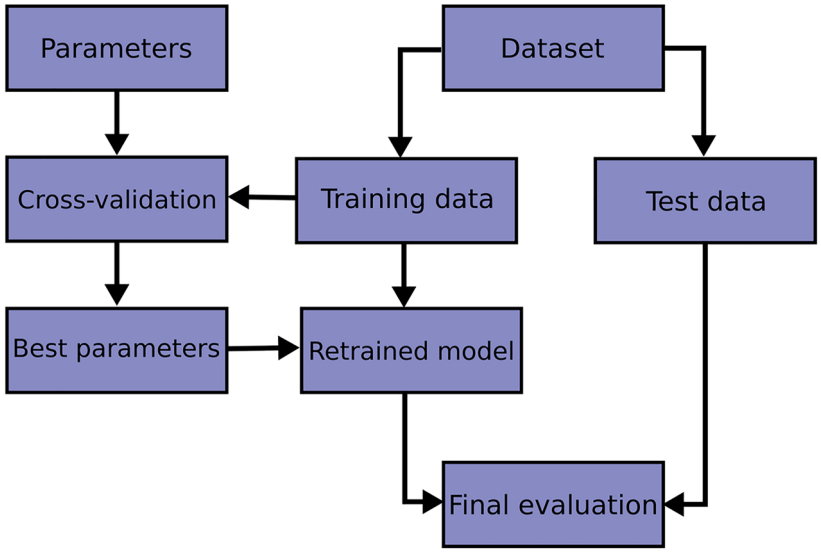

Figure 7. Signature verification based on feature selection via cross-validation [41].

2.5. Feature selection

In this phase, feature selection or dimensionality reduction was performed on a given signature image dataset. This selection of features helps enhance the accuracy score and improve the performance of the high-dimensional signature image dataset. The initial insight obtained from the preceding statistics shows that the VGG16 layout surpassed any additional designs plus characteristics extracted from algorithms for classification, with a minimum of 95% precision for training and 60% evaluation performance. Four different designs were selected to apply our categorization methods to the challenge.

2.5.1. VGG-16 model

The VGG-16 model has 16 layers of a deep convolutional Neural Network (CNN), a pre-trained model in which signature-based images are trained from the ImageNet database. This pretrained network categorizes the images into pixels and is fed into various neural network layers to predict the outcome as a single output layer. The network had an image input size of 224 × 224 pixels.

Here, optimization techniques, such as Adam, RMSProp, Adagrad, and SGD, are used to obtain the optimal solution to verify the signature and identify whether the verified signature is forgery or real.

The architecture of the VGG-16 layers is illustrated in figure 8, and the layers in the VGG-16 model are described as follows:

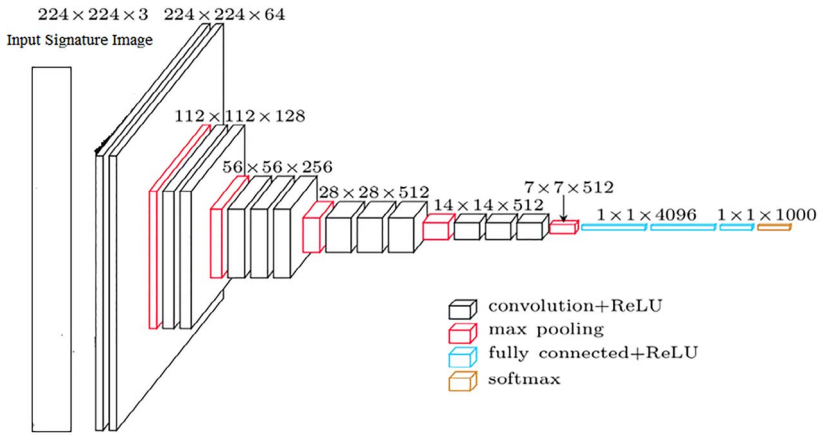

Figure 8. Architecture of VGG-16 layers [42].

Input layer: A 224 × 224 signature image was fed into the VGGNet model. The creators of the model kept the input image size consistent by removing a 224 × 24 square from the middle of every image submitted for the ImageNet concurrence.

Convolutional Layer: The VGG algorithm convolutional filters the smallest 3 × 3 reactive surfaces. In addition, VGG uses a 1 × 1 convolution filter to translate the signature picture data linearly.

Activation layer: This layer contains the function as a Rectified Linear Unit (ReLU), which reduces the learning time of the network. Moreover, this function is linear, which presents the corresponding outcome for a positive input image and provides zero for negative input images.

Hidden Stage: Rectified Linear Unit is employed to maintain AlexNet Simultaneous Data Standardization across the concealed phases of the entire VGG network. The final strategy extends workouts and consumes more mental capacity but does not result in total efficiency.

Pool Stage: This layer reduces the dimensionality and quantity of features in feature maps built by every stage of the convolutional layer. Pooling methods are critical, given the sudden rise in the total amount of viable filtering through 64–128, 256, and 512 in the last three stages.

FCC layers: VGGNet comprises three interlinked tiers. The first and second phases contained 4096 routes, whereas the third phase contained 1000 channels, one for each type. Finally, the output layer determines whether the input raw signature image is authenticated as a forgery sign or signed by an authorized person.

2.5.2. Inception V3 model

A complexity-separated convolution layer is used in inception as an addition to the Xception architecture instead of conventional convolution layers. A neural network called the inception model was used to classify objects in the signature images. Google Nets is an alternative to inception. The ImageNet dataset was used in the training phase. The resolution of the inception was 299 × 299 × 3. Inception convolutional neural networks can produce more efficient computing and deep connections by reducing dimensionality with a stacked 1 × 1 convolution layer. The components were created to address problems such as generalization and computing complexity [43].

2.5.3. ResNet 50 model

ResNet features multiple parts and sub-module configurations compared to other architectures, setting it apart from standard subsequent communication networks like VGGNet and Alex Net. Moving to the lowest level and disregarding level changes could be better. ResNet's architecture addresses this problem, which increases the network's success rate by making it easier to recall the system. The 177-layer neural network used was ResNet. This model was trained with signature images of dimensions 224 × 224 × 3.

2.5.4. Xception model

The Xception network has gradually replaced the inception network. Extreme inception is referred to as xception. Instead of the conventional fully connected layers, the Xception architecture uses larger values with discrete convolutional sections. Numerous spaces, along with parametric connections in which CNN-extracted features may be completely detached, are accessed by Xception. Convolution in the Xception architecture can be divided into 14 different alternative paths; however, the fundamental architecture of inception has been maintained for approximately 36 more years than in Xception. A continuous residue link encircles each level

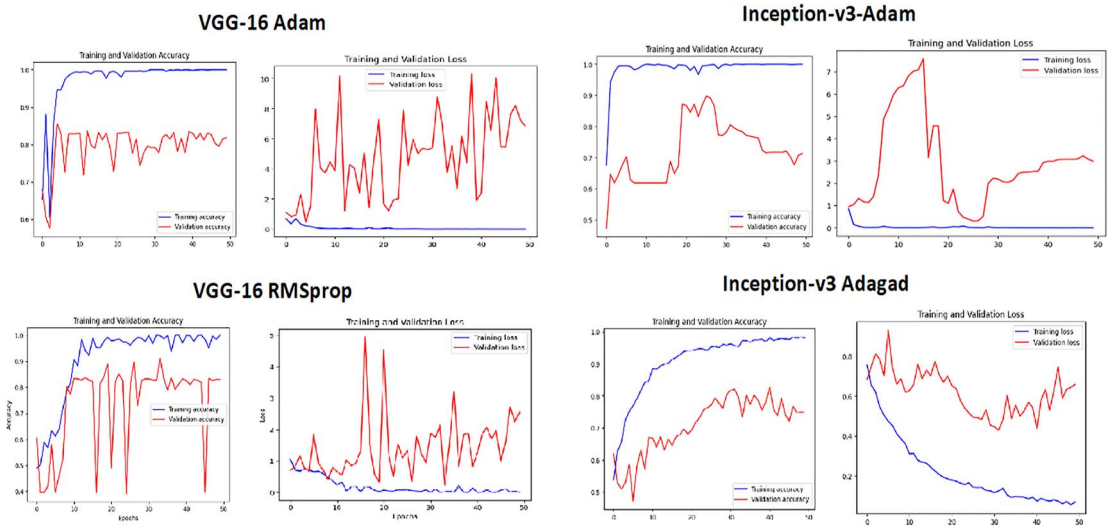

Figure 9. Validation accuracy versus loss using optimization approaches.

after deleting the first and last levels. The input image was transformed to determine the chance of collecting cross-channel correlations across every outcome. Subsequently, a depth-wise 11 convolution method was used. The interconnections can be depicted as a 2D + 1D projection instead of three-dimensional projections. A two-dimensional sector correlation sets the stage for emergence, whereas a one-dimensional space correlates first.

3. Introduced methodologies and classification

This section provides the proposed methodology by importing the necessary modules from the Keras API that binds the TensorFlow backend. Our model was constructed using backend TensorFlow. Initially, Python was used to train the Neural Network using a distinctive class of genuine and forged signatures. In this study, various deep learning models were proposed to train the network by splitting the dataset into a train-test ratio of 70:30.

3.1. Network training and validation

Here, the author evaluates the difference between the expected value and the true value of the label throughout the network training stage using the loss function task, otherwise called the cost function, and the network is trained to reduce this difference. The anticipated outcome is more closely related to the actual label: the lower the loss value. As shown in equation (1), the output layer is a sigmoid function that manages binary issues and produces an S-shaped curve with values between 0 and 1. In addition, the cost function is selected as the binary cross-entropy (BCE), as illustrated in equation (2), where y denotes the true signing and is the projected likelihood that the objective is a real identity.

$$Sigmoid\,(x) = \frac{1}{1 + e^{-x}} \tag{1}$$

$$Binary\ Cross\ Entropy = -y \log(\hat{y}) - (1 - y) \log(1 - \hat{y}) \tag{2}$$

Subsequently, the BSE parameter was used to optimize performance through elimination techniques, we improved our neural networks using the widely used stochastic gradient descent (SGD) strategy [30]. We chose a meager amount for e^-4 with the velocity factor of 0.9, which is most frequently utilized in SGD because an excessive learning speed could prevent convergence; 48 photos are in our collection, with two real authors and eight imposters.

3.2. Results and analysis

In this study, images were binarized and stored appropriately. Then, the images were split in a ratio of 70:30; subsequently, file handling and management procedures were performed to divide the batches of signature-based images. Following the construction of the deep learning models, plots of accuracy and loss were created.

Additional deep learning pre-trained models were built for various data splits, and the training and validation accuracies were plotted to determine whether there was any overfitting or underfitting. In the validation part, an optimal resolution of 83% was achieved in detecting and verifying the forgery signature, as shown in figure 9. Because the accuracies of training and testing are nearly comparable, there is quite a bit of

Table 1. Obtained results from proposed method and state of the art techniques.

Model + Optimizer	Training accuracy (%)	Validation accuracy (%)	training loss	validation loss	Remark
VGG-16 + Adam	100	82	~0.0	6.85	Slight overfitting observed
VGG-16 + RMSProp	100	83	~0.0	2.55	High generalization performance
Inception-v3 + Adam	100	72.29	~0.00	2.99	Stable performance
Inception-v3 + Adagrad	98	74.7	~0.07	~0.66	Stable performance
Local features [44]	NA	Equal Error Rate 20	NA	NA	Lower performance on global features
Global features [44]	NA	Equal Error Rate 36	NA	NA	Struggles with disguised signatures
SVM with geometric features [45]	NA	67.08	NA	NA	Lower accuracy
SVM with HOG features [45]	NA	76.67	NA	NA	Lower accuracy

excess fitting. The evaluation of the training and validation loss, training accuracy, and validation accuracy for VGG-16, Inception V3, ResNet 50, and Xception, along with four optimizers (Adam, Adagrad, RMSProp, and SGD) for forgery signature authentication, are depicted in figure 9.

3.2.1. Comparision of results and discussion

In the present comparative analysis of various models and optimizers, VGG-16 with RMSProp emerged as an efficient modelling technique, achieving a training accuracy of 100% and validation accuracy of 83%, with a relatively low validation loss of 2.55. This balance between high accuracy and reduced loss indicates an effective generalization and minimal overfitting. VGG-16 paired with Adam also showed a strong training accuracy of 100%, but a slightly lower validation accuracy of 82% and a higher validation loss of 6.85, suggesting overfitting despite the model's high training performance.

By contrast, the Inception-v3 model performed less effectively. Inception-v3 with Adam achieved a validation accuracy of 72.29% with a validation loss of 2.99, highlighting some issues in generalization. However, Inception-v3 with Adagrad improved the validation accuracy to 74.7% and maintained a significantly lower validation loss of ∼0.66, indicating greater stability. Overall, VGG-16 with RMSProp demonstrated the most balanced and reliable performance in forgery signature detection. From figure 9, it is also observed that slight fluctuations in the validation curves may indicate minor overfitting, likely due to the limited size of the dataset. These issues may be addressed with applications of advanced data augmentation and further hyperparameter optimization, providing a clear direction for future work.

The results obtained were also compared with those of traditional methods such as localized and global features. The SVM methods and their accuracies are listed in table 1. The comparison clearly indicates that the proposed method outperforms traditional approaches. Table 1 highlights the higher accuracies achieved by our models, demonstrating their superior effectiveness for signature verification.

4. Conclusion

This study summarizes the verification of online signatures using the ICDAR 2011 Signature Dataset. Several existing studies have introduced deep learning-based convolutional neural networks and multi-layer perceptrons to verify digital/handwritten signatures that provide security for land, payments, etc In this study, datasets of various signatures were collected from an open-source website. Feature extraction was performed to extract relevant features, and feature selection was executed by building multiple models, such as the VGG-16, ResNet 50, Inception V3, and Xception models, to identify forgery signatures. To obtain the optimal solution for signature authentication and classification into either real or forgery, four deep-based optimization methods, RMSProp, Adam optimizer, SGD, and AdaGrad, were used, and the validation accuracy was approximately 83% with minimum loss. The proposed method also produced superior results compared with traditional techniques. This demonstrates the reliability and efficiency of our approach, making it well suited for real-world applications that require robust signature verification. In addition, the proposed research findings highlight the potential of deep-learning models to outperform traditional methods in signature authentication, paving the way for further advancements in secure biometric systems.

5. Limitations and future scope of work

5.1. Limitations

The proposed models in the present work showed strong validation accuracy, but signs of overfitting were present, particularly with the VGG-16 configuration using the Adam optimizer. This indicates the need for additional regularization or a more diverse training dataset to improve generalization. Additionally, our study's reliance on a relatively small sample size from the ICDAR 2011 Signature Dataset might not fully reflect the variety found in real-world signature scenarios.

5.2. Future directions

The present work may be extended using attention mechanisms or hybrid models that blend CNNs with RNNs, which can better capture both the spatial and temporal aspects of signatures. Expanding datasets with more diverse samples is likely to enhance model robustness by applying advanced data augmentation techniques and performing further hyperparameter optimization.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: https://doi.org/10.1109/ICDAR.2011.294.

ORCID iDs

M Ranga Swamy https://orcid.org/0009-0004-3741-2268

References

[1] Roszczewska K and Niewiadomska-Szynkiewicz E 2024 Online signature biometrics for mobile devices *Sensors* **24** 3524
[2] M A S, Suvarna S and K T R 2023 Online digital cheque signature verification using deep learning approach *2023 2nd Int. Conf. on Edge Computing and Applications (ICECAA)* 866–71
[3] Agrawal R *et al* 2023 Classification and comparison of ad hoc networks: a review *Egypt. Informatics J.* **24** 1–25
[4] Schorlemmer T R *et al* 2024 Signing in four public software package registries: quantity, quality, and influencing factors *2024 IEEE Symposium on Security and Privacy (SP)* 1160–78
[5] Bhavani S D and Bharathi R K 2024 A multi-dimensional review on handwritten signature verification: strengths and gaps *Multimed. Tools Appl.* **83** 2853–94
[6] Leghari M, Memon S, Das Dhomeja L, Jalbani A H and Chandio A A 2023 *Online Signature Verification Using Deep Learning Approach BT - Soft Computing Applications* ed V E Balas *et al* (Springer International Publishing) 465–76
[7] Jose APL M M, Bernardo B and Nuno L 2022 Offline handwritten signature verification using deep neural networks *Energies* **2022** 7611
[8] Salturk S and Kahraman N 2024 Deep learning-powered multimodal biometric authentication: integrating dynamic signatures and facial data for enhanced online security *Neural Comput. Appl.* **36** 11311–22
[9] Kao H H and Wen C Y 2020 An offline signature verification and forgery detection method based on a single known sample and an explainable deep learning approach *Appl. Sci.* **10** 11
[10] Girshick R, Donahue J, Darrell T and Malik J 2014 Rich feature hierarchies for accurate object detection and semantic segmentation *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* 580–7
[11] Scherer D, Müller A and Behnke S 2010 Evaluation of pooling operations in convolutional architectures for object recognition *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)* **6354** LNCS, no. PART 3 92–101
[12] He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2016-Decem* 770–8
[13] Zhang Z, Liu X and Cui Y 2016 Multi-phase offline signature verification system using deep convolutional generative adversarial networks *Proc. - 2016 9th Int. Symp. Comput. Intell. Des. Isc. 2016* 2, 103–7
[14] Jahandad, Sam S M, Kamardin K, Amir Sjarif N N and Mohamed N 2019 Offline signature verification using deep learning convolutional neural network (CNN) architectures GoogLeNet inception-v1 and inception-v3 *Procedia Comput. Sci.* **161** 475–83
[15] Rustam F, Ishaq A, Munir K, Almutairi M, Aslam N and Ashraf I 2022 Incorporating CNN features for optimizing performance of ensemble classifier for cardiovascular disease prediction *Diagnostics* **12** 1–17
[16] Sharma N, Gupta S and Mehta P 2021 A comprehensive study on offline signature verification *J. Phys. Conf. Ser.* **1969** 012044
[17] Alajrami E, Ashqar B A M, Abu-Nasser B S, Khalil A J, Musleh M M and Barhoom A M 2019 Verification of handwritten signature using deep learning *Int. J. Acad. Multidiscip. Res.* **3** 39–44 (www.ijeais.org/ijamr)
[18] Mohsen F, Mohammad H S, Mohammad S, Mojtaba H and Mahmood F 2015 A novel approach for finger vein verification based on self-taught learning *9th Iranian Conference on Machine Vision and Image Processing* (Shahid Beheshti University) 88–91
[19] Fayyaz M F M, Saffar M H, Sabokrou M and Hoseini M 2015 Online signature verification based on feature representation *International Symposium on Artificial Intelligence and Signal Processing* **2015** 211–6
[20] Ghosh R 2021 A recurrent neural network based deep learning model for offline signature verification and recognition system *Expert Syst. Appl.* **168** 114249
[21] Kim H L J and Kim H 2016 Online signature verification using deep convolutional neural network *International Joint Conference on Neural Networks (IJCNN)*
[22] Kim J K H and Lee H 2018 Online signature verification using deep neural network with hidden Markov model *J. Ambient Intell. Humaniz. Comput.* **9** 191–202
[23] Sudharshan D P and Vismaya R N 2022 Handwritten signature verification system using deep learning *IEEE Int. Conf. Data Sci. Inf. Syst. ICDSIS* 2022, 39–44
[24] Poddar J, Parikh V and Bharti S K 2020 Offline Signature recognition and forgery detection using deep learning *Procedia Comput. Sci.* **170** 610–7
[25] Menotti D *et al* 2015 Deep representations for iris, face, and fingerprint spoofing detection *IEEE Trans. Inf. Forensics Secur.* **10** 864–79
[26] Oladele T O, Adewole K S and Oyelami A O 2014 Forged signature detection using artificial neural network *African J. Comput. ICT* **7** 11–20
[27] Zhang X Z L and Li X 2016 Signature verification using convolutional neural network *Int. Conf. on Pattern Recognition (ICPR)*
[28] Sadkhan S B and Sadkhan R S B 2022 Analysis of different types of digital signature *8th Int. Engineering Conf. on Sustainable Technology and Development (IEC)* 241–6
[29] Minaee S, Abdolrashidi A, Su H, Bennamoun M and Zhang D 2023 Biometrics recognition using deep learning: a survey *Artif. Intell. Rev.* **56** 8647–95
[30] Yang T, Zhang Y, Xiao S and Zhao Y 2021 Digital signature based on ISRSAC *in China Communications* **18** 161–8
[31] Malik J, Elhayek A, Guha S, Ahmed S, Gillani A and Stricker D 2020 Deepairsig: end-to-end deep learning based in-air signature verification *IEEE Access* **8** 195832–43
[32] Sharma N *et al* 2022 Offline signature verification using deep neural network with application to computer vision *J. Electron. Imaging* **31** 41210
[33] Hoang H H and Trinh H H 2021 Improvement for convolutional neural networks in image classification using long skip connection *Appl. Sci.* **11** 1–15

[34] Khan M A and Park H 2024 FireXplainNet: optimizing convolution block architecture for enhanced wildfire detection and interpretability *Electron* **13** 10
[35] Swamy M R, Vijayalakshmi P and Rajendran V 2024 Intelligent systems and applications in engineering online signature authentication using pre-trained optimization techniques *Int. J. Intell. Syst. Appl. Eng.* **12** 3928–35
[36] Liwicki M *et al* 2011 Signature verification competition for online and offline skilled forgeries (SigComp2011) *2011 Int. Conf. on Document Analysis and Recognition* 1480–4
[37] Hafemann L G, Sabourin R and Oliveira L S 2017 Learning features for offline handwritten signature verification using deep convolutional neural networks *Pattern Recognit.* **70** 163–76
[38] Diaz G P, Ferrer M, Impedovo M A, Malik D and Pirlo M I 2019 A perspective analysis of handwritten signature technology *ACM Comput. Surv.* **51** 1–39
[39] Deng P S 1999 Wavelet-based off-line handwritten signature verification *Comput. Vis. Image Underst.* **76** 173–90
[40] Pal S, Alaei A, Pal U and Blumenstein M 2016 Performance of an off-line signature verification method based on texture features on a large indic-script signature dataset *2016 12th IAPR Workshop on Document Analysis Systems (DAS)* 72–7
[41] https://scikit-learn.org/1.5/modules/cross_validation.html
[42] Rum S N M and Nawawi F A Z 2021 FishDeTec: a fish identification application using image recognition approach *Int. J. Adv. Comput. Sci. Appl.* **12** 102–6
[43] Fathimathul R P P *et al* 2022 A novel method for the classification of butterfly species using pre-trained CNN models *Electron* **11** 1–20
[44] Malik M I, Liwicki M and Dengel A 2011 Evaluation of local and global features for offline signature verification *CEUR Workshop Proc.* **768** 26–30
[45] Zhou Y, Zheng J, Hu H and Wang Y 2021 Handwritten signature verification method based on improved combined features *Appl. Sci.* **11** 5867