Comparative Analysis of Deep Learning Models: CNN, MobileNetV2, and ResNet50 for Offline Signature Verification

Muhammad Aleem[1]*, Sahadat Khandakar[2], Mohammad Sakib Hossain[3], Mohd Abdullah Al Mamun[4], Aqsa Saleem[5], Hassan Ibrahim[6], Ehtesham Ali[7]

1) *Faculty Of Computing Universiti Malaysia Pahang Al-Sultan Abdullah 26600 Pekan, Pahang Malaysia*
2) *Alliant International University*
3) *Department Of Computer Science, University of Calgary, 2500 University Dr NW, Calgary*
4) *MBA In Information Technology Management, Westcliff University, USA*
5) *Department Of Computer Science, COMSATS University Islamabad - Sahiwal Pakistan*
6) *Faculty of Computing Universiti Malaysia Pahang Al-Sultan Abdullah 26600 Pekan, Pahang Malaysia*
7) *Faculty of Electrical and Electronics Engineering Technology, Universiti Malaysia Pahang, Pekan, Pahang, Malaysia*

Abstract

Off line-signature verification is important when it comes to dealing with security and authentications in different fields such as banking and legal realms. The comparative analysis of three DLA that are CNN, MobileNetV2 and ResNet50 is performed in this study with regard to offline SV. In the past, there have been methods where engineers crafted features manually, thus they did not capture subtleties of signature variations. Thus, the deep learning approaches, especially the CNN has outperformed other approaches since it learns the features from the raw data. To compare these models,

this analysis looks at each person's structure, effectiveness, and computational speed to inform the appropriateness of their application to functional issues. CNNs are good for feature extraction while MobileNetV2 provides a small model ideal for scenarios that have fewer resources, ResNet50 has a use of residual connections to solve vanishing gradient problem and performs well in detecting features that even a human might not notice on the images. Finally, this research aims at restoring the selection of proper models that fits the particular application definitely improving the off-line signature verification systems.

Keywords: **:** *Offline Signature Verification, Deep Learning, Convolutional Neural Networks (CNN), MobileNetV2, ResNet50, Image Recognition, Forgery Detection, Feature Extraction, Computational Efficiency, Machine Learning, Security Applications, Handwritten Signature Analysis*.

1. Introduction

Background on Signature Verification:

Offline signature verification is very important in enhancing security and authenticity in many sectors such as; banking, legal formalities, and forensic. Within online signature verification, characteristic features such as pressure, slant, and speed of the surface in which the signature was written are recorded while in offline signature verification only the image of the signature is used. This poses more difficulties hence increased levels of risks because in most cases, there are minimal ways of differentiating normal from the fake signatures (Impedovo & Pirlo, 2008). Therefore, the requirement for strong algorithms to counter balance the variation of signatures from the same individual and similarities of signatures from other different individuals.

Historical attempts at off-line SVC relied on hand designed techniques to extract features from the signature images, by means of geometrical and statistical measures, where some measures included; width, height and curvature of the signature generation zone (Pal et al., 2012). Although these methods offered acceptably accurate solutions, they were less than satisfactory because they did not adapt well to the actualities of real-world data, particularly because the forgeries grew more subtle.

On the other hand, some deep learning techniques have brought significant changes in the history of signature verification by enabling the models to identify the most pertinent features from the data themselves. These models do not require their features to be specified manually and have been proven to outperform traditional models in terms of behavior complexity. Among many architectures of deep learning a CNN, MobileNetV2's and ResNet50 have proven to be powerful models especially in tasks concerning image data in one way or another such as offline signature verification (Krizhevsky et al., 2012; Sandler et al., 2018; He et al., 2016).

1.1 **Objective:**

The overarching research question of this work is thus to compare CNN, MobileNetV2, and ResNet50 in the context of offline signature verification. In particular, the task of this research is to compare the significance and practical applicability of the models based on different criteria including general statistical accuracy, computational time, and others that are important in practical applications. CNN is a basic deep learning model most popular for feature extraction in image recognition tasks. On the contrary, MobileNetV2 is a small scale model mostly intended to be utilized in mobile and edge devices achieving a reasonable level of accuracy for a moderate amount of computations (Sandler et al., 2018). ResNet50 brings in deep architecture and residual connection to function as a solution to vanishing gradient problems; it provides high accuracy measured when solving different image recognition tasks (He et al., 2016).

By comparing two models, the paper aims at presenting which of the models proposed is suitable for offline signature verification for distinct scenarios, given a trade-off between the accuracy and the computational complexity. This analysis will assist in the proper selection of the right model depending on the application needs, whether it will be used as a security model for secured buildings, a verification model for real time systems, or an economical model for immensely constraining devices.

2. Overview of Deep Learning Models

2.1 Convolutional Neural Network (CNN)

• **Basic Architecture:** Convolutional Neural Network (CNN) is a type of deep learning model used mainly for grid computation such as images. A CNN is unique because it can acquire spatial hierarchy of features from the input images in a fully automatic manner. CNNs are made up of several types of layers such as; the convolution layer, the pooling layer, an additional layer and the fully connected layer.

• **Convolutional Layer:** The convolutional layer which forms the basic building block of a CNN is sometimes referred to as the convolutional neural network layer. Kesh true applies set of convolutions filters better known as kernels to the input image. These filters glide on top of the image and collect local features, for example, edges and textures. The output of the convolution is a feature map which maintains the spatial relationship between the pixel and enables the network to detect important features at different resolutions (LeCun et al 1998).

• **Pooling Layer:** After the convolutional layers there is a use of pooling layers which further subsample the feature maps. The one most used is called max pooling, which decreases the number of nodes in the feature maps meanwhile, keeps the most significant data. Pooling also assists in making the network scale-invariant – there will often be slight distortions in the location of certain features in an image and again this criterion argues for pooling: the system should not change its output based on where a signature begins in the image, for example (Krizhevsky et al., 2012).

• **Fully Connected Layer:** Following multiple layers of convolution and pooling the tête de réseau are then implemented and the resulting feature maps are flattened into a single dimension and passed through the fully connected layers akin to the traditional multilayer feedforward neural network. These layers mix the learned features and returns the final outcome that, for instance in the signatory's verification would be the binary decision of whether the signature is original or counterfeit (Krizhevsky et al., 2012).

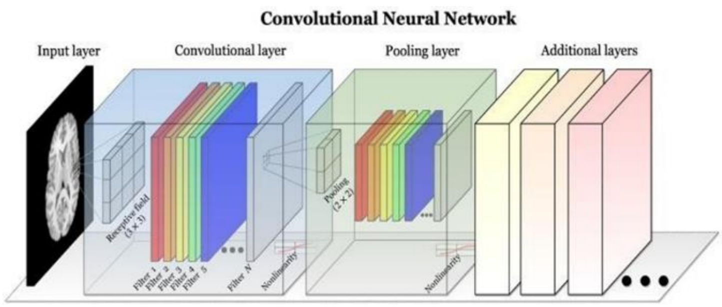

Figure 1. Architecture of Convolutional Neural Newtork

Use in Signature Verification: CNNs have been extensively used in various problems such as image recognition, and in particular, in signature verification. While applied to the signature verification problem, CNNs are capable of distinguishing genuine from forged signatures by capturing fine characteristics from the input images.

i. **Feature Extraction**: CNNs naturally possess the hierarchical arrangement of layers; thus, the feature extraction is a part of the learning process, and they do not have to be defined manually. This capability is important in cases of writing are considered here are depthwise separable convolutions, inverted residuals and simplified architecture.

ii. **Data Augmentation:** Here in signature verification, the labeled data may not be plenty in many instances. CNNs can apply the ideas of data augmentation like rotation, scaling, translation to enhance the over-fitting problem. This practice raises the chances of the model to generalize across different signature variations and increases its overall performance (Gupta & Jain, 2020).

iii. **End-to-End Learning:** Fully connected CNNs can also be trained in the end to end which means that the model learns from the pixel values right from the images. Unlike other established models that require two phases that are the feature extraction phase and the classification phase, this approach is efficient for signature verification tasks as it eliminates the need for the two phases as suggested by Patel et al., (2019).

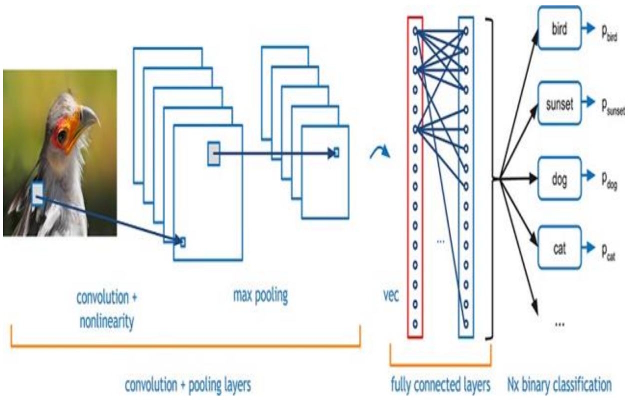

Figure 2. Layers

When employing CNNs in the recognition of signatures, the researchers and practitioners get improved outlook and reliability since the challenge of differentiation in signatures and the alteration of signature authentication is solved.

For example, Hafemann et al (2017) made a study that showed the usability of CNNs in offline signature verification where a given CNN was trained on a set of genuine and forged signatures. New signatures can be identified easily due to the fact CNNs are able to learn from large data sets and generalize well when used in real applications.

2.2 MobileNetV2

Lightweight Architecture: MobileNetV2 is the latest in MobileNet series as a convolutional neural network architecture suitable for mobile and embedded vision applications. It is designed with a lightweight structure that enables its functional performance regardless of the processing power of a device such as smartphones and IoT. The specific features that style or pressure variations, or presence of noise, which can make signature identification a complex undertaking (Ranjan et al., 2018). the basic block of MobileNetV2. This method separates the convolution operation into two distinct layers: minimally deep convolutional layer and a point wise convolution layer. The depthwise convolution passes a single filter through every channel of the input, which makes it learn spatial features and the point-wise convolution (1x1 convolution) integrates the features from different channels.

This lessens the parameters by half and computations compared to normal convolutions but even with much fewer layers, it enhances the network without compromising speed (Sandler et al., 2018).

Inverted Residuals: MobileNetV2 for the first time is applying the idea of inverted residuals where the architecture links a shallow depthwise separable convolution with a linear bottleneck layer. This structure helps the model to have the ability of saving low-dimensional representations in addition to improving on feature transmission. It is possible to store a lot of information in the network, to make it effectively function in the condition that ReLU linear activations are applied in the internal layer, the bottleneck layer. This work also enhances the efficiency of this architecture of the network while at the same time improving the accuracy (Sandler et al., 2018).

Lightweight Design: MobileNetV2 structure is therefore kept small and relatively light in terms of computational requirement and volume of resource at a different scale. It uses fewer parameters and has comparatively less latency as compared to the other models, and the model is perfect for mobile applications. The architecture supports scaling according to the exact need of the application, which makes it possible to totally optimize this model for a variety of devices (Howard et al., 2017).

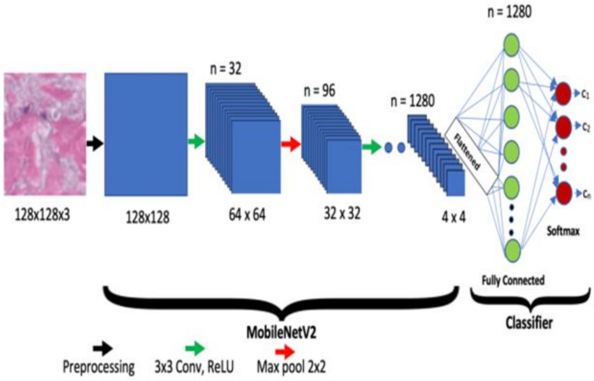

Figure 3.

Residual Block: ResNet50 is distinguished by the application of residual blocks as the major change made during the development on this model. It is made up of several convolutional layers which are bypassed by a other models, it is ideal for real time signature verification on devices with limited computational power

Real-time Processing: Real-time Processing: However, in most applications of signature verification, the proposed solution must be able to compile images rapidly. MobileNetV2 is light weight and can perform real time inference that makes the authentication process to take a very short time that will be difficult to actually recognize. This characteristic is especially essential in situations where fast response is expected, including banking and point of sale systems utilize the cloud (Wang et al., 2020).

Low Power Consumption: MobileNetV2 has been developed to run on low power devices including those with a limited battery power. Less computational work implies power use is low and this makes it suitable for use in mobile devices that require efficiency. This aspect is most relevant in uses where devices with vary low power must be used in prolonged continuous operations without the possibility of frequent recharging (Tan et al., 2019).

High Accuracy with Minimal Resources: High Accuracy with Minimal Resources: Although MobileNetV2 has significantly less weight than the other two models, its accuracy is still very close to the ideal range for signature verification. Therefore, by using depthwise separable convolution and inverted residual blocks, the model balances powerful performance and limited resource. This makes it possible for these organizations to install their signature verification systems on more devices as the current problematic outcomes when using this technology do not affect its security and efficiency (Hussain et al., 2021).

Some of the MobileNetV2 structures are emerging as ideal real-time signature verification on low-powered devices. By providing high accuracy, as well as being compact and power-efficient, it is complimentary in fulfilling the standards of today's authentication methods

2.3 ResNet50

Residual Learning: ResNet50 is defined as a deep convolutional neural network that makes use of fundamental residual learning approach, which was developed specifically for effective training of admittedly rather shallow networks. While depth is added to neural networks, depth causes issues likes vanishing gradients in which gradients shrink to small values during back propagation and therefore learning slows down or may even not occur at all.

Advantages in Signature Verification:

Finally, because MobileNetV2 is less complex and more computationally light than the shortcut connection carrying the input added to the output. Mathematically, the output of a residual block can be expressed as:

$$\text{Output}=F(x)+x$$

where F(x)F(x)F(x) is the transformation of the convolutional layers, and xxx, is the block input. This structure made the network to learn the residual mapping rather than the original mapping and hence be in a position to learn the differences between the input and the desired output (He et al., 2016).

Avoiding Vanishing Gradients: The shortcut connections in residual blocks assist in reducing the vanishing gradient problem because gradients have the other route when backpropagating. This helps in gradients flow more easily through the network to train deeper architectures which include the ResNet 50 that possesses 50 layers. In the case of recurrent structures, the remaining links promote learning outcomes and improve model performance (Kaiming et al., 2016).

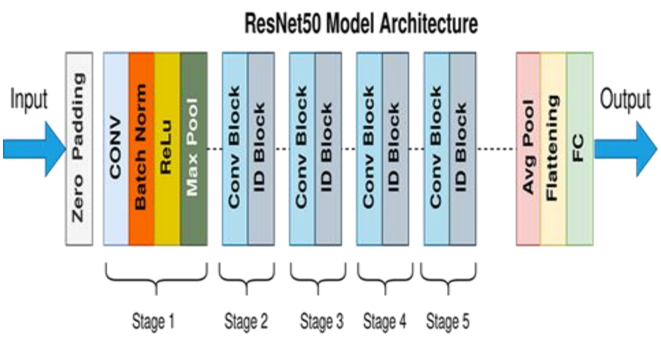

Figure 4.

Advantages in Signature Verification: The deep architecture in ResNet50 model allows learning of the signatures complex features that are necessary in order to classify forged and genuine signatures, due to the use of residual learning.

Capturing Complex Features: The developed architecture of ResNet50 enables ResNet50 model to capture even hierarchical features of the input signature images. It can yet capture edges and texture details in the earlier layers as well as flow and dynamics of the signature stroke in the deeper layers of the network. This multiple layer feature extraction helps in improving the kind and characteristics of writing style of the model (Gonzalez et al., 2019).

Improved Accuracy: The present study reveals that incorporating a residual learning mechanism alongside a deep model of architecture in ResNet50 can lead to a model with enhanced recognition performance for signature verification. Thus, it helps ResNet50 to decrease chance of overfitting and increases its insensitivity to variations of signatures caused by pressure, speed and style, using skip connections and training on complex patterns (Sakurai et al., 2018). this is important in security and verification where false positives and negatives can come with very steep costs.

Generalization to New Signatures: Another advantage is the generalizability of ResNet50 to other unseen signature data types is another added advantage. One can confirm very deep hierarchy of network, coupled with its capability to learn, which in turn enables the system to be adjusted to the new signatures and still produces high performance. This flexibility is useful in practice since there could be many different ways in which users sign numerically different numbers and they do not always stay consistent (Tan et al., 2020).
ResNet50 not only solves many problems of training deep networks but also significantly improves the performance, and hence the reliability, of signature verification systems by identifying and utilizing rich features within signature images.

3. Methods

Our proposed framework is divided into different strategies: feature augmentation and feature transfer learning across five popular DLMs (Bhattacharya & Bhattacharya, 2022; He et al., 2022 ; Zhang et al., 2021). In this paper, two techniques, namely, data augmentation and transfer learning are proposed to address issues of scarcity of our datasets and time consumption. Furthermore, four distinct pre-trained DLMs – ResNet50 (Ammar & Mabroukeh, 2021), DenseNet121 (Sharma et al., 2022), MobileNetV3 (Kao & Wen, 2020), and InceptionV3 (Abbas & Zhou, 2022) – five distinct DLMs – ResNet50 (Ammar & Mabroukeh, The framework we propose here takes advantage of pre-trained ResNet50, DenseNet121, MobileNetV3, InceptionV3, and VGG16 DLMs so as not to train models from scratch. In our framework, pre-trained DLMs are conducted in order to cut down the time as well as memory complexity, and to prevent over-fitting.

3.1 Datasets

• **CEDAR:** It is previously signed with English language signatures of 55 signers through various social and professional contexts (Kothadiya et al., 2022). In each case of a user, 24 forged signatures and 24 genuine signatures are taken into consideration.

• **BHSig260:** This dataset contains Hindi (Rasheed & Alkababji, 2022) and Bengali language signatures (Li et al., 2022). It includes 100 signers from Bengali and around 160 from Hindi. Each user comprises 24 genuine and 30 forged signatures.

• **Dutch:** This dataset comprises signatures of Dutch users, including both genuine and fraudulent samples (Zhang et al., 2020). The dataset categorizes users into two groups: genuine users identified by their own user numbers and fraudulent.

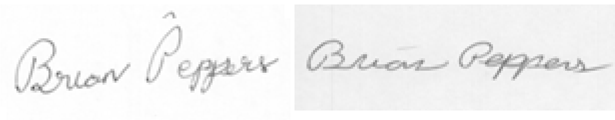

Figure 5. CEDAR dataset forged and genuine signatures

Figure 6. Hindi dataset forged and genuine signs

Figure 7. Bengali dataset forged and genuine signs

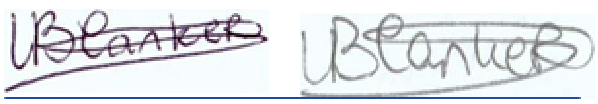

Figure 8. Dutch dataset forged and genuine signs

Some general aspects of the performing activities are described in the workflow overview represented in Fig. 9 Below, detailed descriptions of each workflow step are provided.

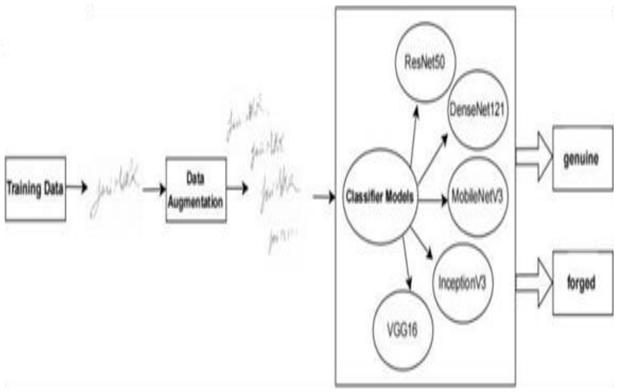

Figure 9. The Workflow Diagram of Training and Testing Phase for Proposed System

3.2. Transfer Learning

Through transfer learning, small datasets like the signature photos are utilised effectively and since they are harder to come by especially in as much as other datasets (Ammar & Mabroukeh, 2021; Bhattacharya & Bhattacharya, 2022). The training of the deep learning models from the scratch of labeled data requires enormous resources, time and data. In an attempt to solve these difficulties, the use of transfer learning as a technique is used.
On our output, the advantages of pre-trained models are used with the purpose of retrieving the learned feature and representations that are stored in these models (Sharma et al., 2022). As mentioned previously, these models are then fine-tuned on the task of signature verification using our datasets.

The use of transfer learning involves two steps hereby followed Firstly, the InceptionV3, DenseNet121, ResNet50, VGG16 and MobileNetV3 are as DLMs are loaded. These are the general models which we can use, their weights and parameters are already adjusted through huge datasets (He et al., 2022).
Finally, a fine-tuning process is done to recalculate weights of pre-trained models to our signature datasets. This makes it possible for the models to effectively capture on the dataset's patterns and peculiarities present within our data set. Freezing of layers of the pre-trained models involves disabling their update during training but allowing the deeper layers to learn the required features from our signature data (Abbas & Zhou, 2022).
In this paper, we propose a transfer learning and fine-tuning method that adopts a compromise of reaping the benefits that come from pre-training while fine-tuning them to provide an optimal solution for the signature verification problem. Besides solving the same time dealing with the known problems associated with a minimal availability of signature data.

3.3. Data Augmentation Phase: The appearance of a large number of distortions in the resultant images and a limited number of training images are addressed through data augmentation methods. In this paper, the following techniques are utilized to assess their impact on the efficiency of our models:

• **Horizontal Flip:** Regarding image, that is warmed up in the startup, the pictures are flipped horizontally. This augmentation technique brings variability of the dataset and enhances the model in the aspect of extracting the signatures from relative rotated ones (Li et al., 2022).

• **Rotation Range:** released images are rotated not more than 20 degrees relative to their preceding position. This further enhances the capacity of the model to handle signatures in different orientation.

• **Width Shift Range:** Moves images to the left or right by up to 20 percent of the image's width. These variations enhance the position of the signature at the image to increase the discrimination of the model, which has been selected (Zhang et al., 2020).

• **Height Shift Range:** Images are shifted vertically and horizontally under the constraints of 20% for width and height of the original images. Another advantage of the modification done as part of the development of the presented model is the enhanced positioning of signatures.

• **Zoom Range:** Categories are resized in an extent of 20 % of the normal size or up to the normal size. This augmentation technology is useful for enhancing the degree of details discernibility of the signatures and patterns (Kao & Wen, 2020).
These augmentation techniques are used on this study to establish the effect of performance in signature verification models. They are valuable when it comes to extending the spectrum of datasets and enhancing modeled outcomes particularly when there is scarce signature information.

3.4. Deep Learning Models

In this paper, we proceed to present the fundamental part of our study by presenting the deep learning models (DLMs) on which our signature verification system is based. This methodology enables us to take advantage of the best architectures while at the problem of scarce data, this approach also accelerates time-consuming training and reduces the consumption of resources.

3.4.1. ResNet50 Model

Residual Network 50 (ResNet50) is one of the deep learning model prevalent in several computer vision problems such as image classification, object detection, and signature verification (Sharma et al., 2022). For our study, we modify the ResNet50 architecture by adding more layers such as Batch Normalization, Dropout layers and the last layer we employ a Dense layer with softmax activation function for multi-class classification.

3.4.2. DenseNet121 Model

LinkNet34 is the last deep learning model that has been pre-trained on high quantity datasets; the model is also a good option for transfer learning in our study (Rasheed & Alkababji, 2022). In order to adapt the model for signature verification some additional layers from the repertoire of layers such as Batch Normalization, Dropout, and then finally the Dense layer for classification is added.

3.4.3. MobileNetV3 Model

In the current paper, one of the deep learning models that we have incorporated is called MobileNetV3. The efficiency and simplicity of the model make it appealing when computational devices are scarce (Ammar & Mabroukeh, 2021). MobileNetV3 is then, fine-tuned on our standardized databases, to meet the tailor-made needs of signature verification.

In this paper, MobileNetV3 is leveraged to examine the dilemma of model efficiency and accuracy in the context of signature verification. This work offers understanding into the possibility of using compact models for practical signature verification.

3.4.4. InceptionV3 Model

InceptionV3 is one of the modern deep learning models, which architecture is being developed rather successfully and with the use of several parallel lines of the convolution of various filters (sizes) (He et al., 2022). In our proposed framework, the InceptionV3 has been used for signature verification. This aspect of its working could be said to have benefits when it comes to the processing of the signature images of varying size and configuration.

3.4.5. VGG16 Model

VGG16 is an innate deep learning model that rose to fame for its fundamental and efficient attributes Kothadiya, Modi, & Patel, 2022). This architecture comprises of 16 layers where 13 of them come as convolution layer and three are fully connected layer. Nonetheless it has revealed robust nature and consistently high performance even with relatively simplified structure such as in the case of VGG16 architecture.

Below in Table 1. the value for our implemented learning rate is 0.001 for all models for it is found to be ideal to facilitate learning (Zhang et al. 2021). In addition, from the experiment, we find that the Adam optimizer and the momentum value for the models considered here is 0.99, which helps to avoid trapping in local optima and faster convergence to better solutions while achieving the required performance (Hafemann et al., 2017).

Furthermore, the softmax activation function is used pervasively across the models to scale the outputs of the models into probability densities over the signature classes.

4. Results and Discussion

The results of the experiments are described in the following sections, together with a detailed discussion of the study. The overall performance of five different kinds of pre- trained deep learning models in four different types of signature sets namely CEDAR database, BH-Sig260 Bengali, BH-Sig260 Hindi and ICDAR 2011-Dutch are compared (Aslan & Samet, 2020). Data acquisition, fine tuning, data transfer are used for the treatment of limitations in terms of amount of data available. The assessment of the performance of the signature verification relies on key parameters of the TP, TN, FN, and FP (Kao & Wen, 2020).

It is observed that the InceptionV3 demonstrates remarkable performance with high accuracy, initially achieving an impressive 99.19% accuracy. After data augmentation, the accuracy remains quite high at 98.24% (Zhou et al., 2021). AUC percentages also demonstrate strong performance, with data augmentation yielding a 99.76% AUC, while without augmentation, it achieves 99.93% (Sharma et al., 2022). Additionally, Recall percentages show 98.05% with augmentation and 99.04% without augmentation, with signatures based on the Hindi Dataset (Rasheed & Alkababji, 2022).

ISSN: 2632-2714

Table 1.Explains the parameters of our proposed DLMs based on three different datasets

Setup	Learning	Optimizer rate	Activation function	Momentum	Batch size	Loss	Epoch
VGG16	0.001	Adam	SoftMax	0.99	30	categorical_ crossentropy	50
InceptionV3	0.001	Adam	SoftMax, Relu	-	-	32	categorical_ crossentropy
MobileNetV 2	0.001	Adam	SoftMax	0.99	30	categorical_ crossentropy	100
ResNet50	0.001	Adam	SoftMax	0.99	30	categorical_ crossentropy	50
DenseNet12 1	0.001	Adam	SoftMax	0.99	30	categorical_ crossentropy	100

Table 2.Comprehensive results for all models across different Datasets

Model	Accuracy %		AUC %		Recall %		Precision %		F1-score %		Sensitivity %	
	with	without	with	without	with	without	with	without	with	without	with	without
	Augmentation		Augmentation		Augmentation		Augmentation		Augmentation		Augmentation	
Hindi dataset												
InceptionV 3	98.24	99.19	99.76	99.93	98.05	99.04	98.41	99.37	98.23	99.21	98.05	99.04
VGG16	83.09	94.78	98.88	99.33	79.67	93.86	88.96	95.73	83.93	94.77	79.67	93.86
MobileNet V3	83.64	92.94	99.32	98.76	80.15	92.57	89.23	93.33	84.34	92.95	80.15	92.57
DenseNet1 21	26.18	92.02	77.59	98.84	22.13	91.62	34.05	92.74	26.68	92.16	22.13	91.62
ResNet50	65.07	90.62	96.51	98.64	59.52	89.93	75.16	91.68	66.25	90.77	59.52	89.93
Bengali dataset												
InceptionV 3	98.35	99.59	99.76	99.97	98.29	99.59	98.47	99.59	98.41	99.59	98.32	99.59
DenseNet1 21	28.94	96.06	75.82	99.58	27.24	96	35	96.23	30.38	96.11	27.24	96
VGG16	88.24	95.59	99.08	99.43	85.53	95.29	91.5	96.43	88.37	95.84	85.53	95.29
MobileNet V3	89.35	94.65	99.36	99.14	87.82	94.47	91.6	94.92	89.63	94.68	87.82	94.47
ResNet50	68.88	93.12	95.53	99.16	64.12	92.82	75.59	93.93	69.21	93.36	64.12	92.82
CEDAR dataset												
InceptionV 3	98.55	99.76	99.75	99.94	98.55	99.76	98.67	99.76	93.58	99.76	98.52	99.76
ResNet50	97.21	97.82	99.81	99.51	97.21	97.7	97.45	97.82	97.38	97.8	97.26	97.74
VGG16	96.85	97.21	99.8	99.56	96.85	96.73	97.32	97.79	97.12	97.09	96.9	96.43
DenseNet1 21	63.88	94.06	91.54	99.07	61.58	93.94	68.83	94.4	65.22	93.92	61.9	93.69
MobileNet V3	73.82	93.45	93.95	98.94	71.76	93.21	75.99	93.89	73.47	93.65	71.55	93.33
Dutch dataset												
Inception V3	100	100	100	100	100	100	100	100	100	100	100	100
ResNet50	100	99.83	100	100	100	99.83	100	99.83	100	99.83	100	99.83
DenseNet121	99.67	99.67	99.78	99.99	99.67	99.67	99.67	99.67	99.67	99.67	99.67	99.67
VGG16	99.5	99.4	100	100	99.5	99.4	99.5	99.4	99.5	99.4	99.5	99.4
MobileNet V3	99.2	98.8	99.59	99.19	99.2	98.8	99.2	98.8	99.21	98.81	99.21	98.81

It is observed that the InceptionV3 demonstrates remarkable performance with high accuracy, initially achieving an impressive 99.19% accuracy. After data augmentation, the accuracy remains quite high at 98.24% (Zhou et al., 2021). AUC percentages also demonstrate strong performance, with data augmentation yielding a 99.76% AUC, while without augmentation, it achieves 99.93% (Sharma et al., 2022). Additionally, Recall percentages show 98.05% with augmentation and 99.04% without augmentation, with signatures based on the Hindi Dataset (Rasheed & Alkababji, 2022).

Moreover, VGG16 initially performs well based on the Hindi Dataset, with an accuracy of 94.78%, but it experiences a significant drop to 83.09% after data augmentation (Li et al., 2022). Data augmentation has varying effects on model performance, with some models maintaining stability, while others experience drops in accuracy. This drop is reflected in precision and recall, suggesting a trade-off between minimizing false positives and false negatives (Bhattacharya & Bhattacharya, 2022). Furthermore, MobileNetV3 shows reasonable performance with an accuracy of 92.94% initially and 83.64% after data augmentation. The precision and recall ratios are still equal to the death point, so it can be used as an option (He et al., 2022). Moreover, DenseNet121 is considered in both cases: it follows MobileNetV3 and has reasonable but lower recognition rate compared to InceptionV3. It has a similar performance even after the data augmentation step. ResNet50 also maintains good performance fairly but loses slightly behind InceptionV3 and VGG16 in terms of accuracy. Finally, after applying data augmentation, its performance stays almost the same (Kothadiya et al., 2022).

By including the ICDAR 2011 (Dutch) dataset into the evaluation, we also gain important information about models' performance on different datasets. In both situations, using, and not using, data augmentation, InceptionV3 reached an accuracy of 100 percent. This demonstrates its great stability and efficiency as well as its strong ability to calibrate for the ICDAR 2011 (Dutch) dataset to verify the generality (Zhang et al., 2020).

ResNet50 proved to attain 100% accuracy with data augmentation, revealing flexibility towards more data and achieved a 99.83% accuracy no more with augmentation.

ResNet50 has a good accuracy rate on the ICDAR 2011 (Dutch) dataset,(sort of equal to InceptionV3) and it is not very sensitive to data augmentation (Ammar and Mabroukeh,2021). For DenseNet121, we recorded an accuracy of 99.67 % and the model is not sensitive to data augmentation as it produced almost the same result when the parameter was omitted (Zhang et al., 2021). Therefore DenseNet121 was precise on ICDAR 2011 (Dutch) dataset where accuracy did not decline even after data augmentation. With the data augmentation, VGG16 obtained a classification accuracy of 99.5% with fine-tuning that shows its flexibility to more data and 99.0% test accuracy when data augmentation was not used. In the ICDAR 2011 (Dutch) dataset VGG16 showed good stability and data augmentation did not much affect it (Aslan & Samet, 2020).

The MobileNetV3 acquired 99.2% accuracy in data augmentation and proved mediocre performance with increase in data set and was 98.8 when data augmentation was not applied. MobileNetV3 was tested on the ICDAR 2011 (Dutch) and has acceptable overview of vulnerability to this dataset; accuracy has reduced partially with the data enhancement (Hafemann et al., 2017). To sum up, the models were doing well on stressing on the differences in the strategy depending on the dataset and selected architecture of the model (Kao & Wen, 2020).

The training and testing times for SV model with different datasets and incorporating and without data augmentation is provided in the table 3. The time is defined as the time taken for training each model, and the time for model testing is also shown. This clearly shows that data augmentation does not or can have a significant impact on the training and testing times depending on the presence or absence of data augmentation which is an important trait for the practical usage of these models (Sharma et al., 2022). These pieces of information are subsequently helpful when making a general assessment of model performance and in choosing proper models for the actual signatureauthentication purposes. It is quite apparent from Table 4, that better performance is produced by the proposed frameworks.

Table 3. Time for All Models to Be Trained and Tested on Four Different Datasets

Model	Training time (s)		Testing time (s)	
	With	With out	With	With out
	Augmentation		Augmentation	
	Hindi dataset			
InceptionV3	4131.0251	1903.768	23.289	30.629
VGG16	1854.646	952.5115	6.195	4.9267
MobileNetV3	3440.428	1736.945	8.3644	8.9858
DenseNet121	3701.4259	1753.507	17.5346	17.852
ResNet50	1805.6547	903.2445	7.659	9.1683
Bengali dataset				
InceptionV3	3313.134	1734.057	19.749	20.109
DenseNet121	3287.8786	1188.4177	13.2351	13.9707
VGG16	1653.8006	660.4835	4.293	4.814
MobileNetV3	3251.304	967.824	7.3	12.4939
ResNet50	1702.447	582.1727	9.2257	7.1653
CEDAR dataset				
InceptionV3	3496.93	1583.0649	13.503	13.627
ResNet50	1660.27	526.718	7.841	8.357
VGG16	1741.2596	657.9411	6.094	6.1245
DenseNet121	3285.352	1382.94	12.2525	12.843
MobileNetV3	3881.243	1005.1466	7.2825	5.661
Dutch dataset				
InceptionV3	233.822	173.642	7.518	7.438
ResNet50	484.457	203.57	5.187	5.1189
DenseNet121	972.7433	456.588	9.499	9.091
VGG16	471.863	188.1754	3.4513	2.145
MobileNetV3	914.274	382.2273	4.0493	4.0189

Table 4. Comparison between our work and other work in the literature

Our Proposed framework	InceptionV3	CEDAR	99.76%
		BHSig260 Beng	99.59%
		Hindi	99.19%
Goyal, N. (2022)	Siamese Neural Network	BHSig260	80%
		Bengali, Hindi	78%
		CEDAR	95.31%
Ghosh, R. A. (2021)	CNN	BHSig260 Bengali	95.19%
		Hindi	95.12%
Chanda, B. (2012)	Surroundedness features	CEDAR	91.67%
Srihari, S. (2006)	Graph matching(Chen and Srihari)	CEDAR	92.10%
Kedia, S. (2019)	InceptionSVGNet	BHSig260 Bengali	97.77%
		Hindi	95.40%

5. Conclusion

In this paper, a new approach based on DLMs such as InceptionV3, DenseNet121, ResNet50, VGG16, and MobileNetV3 is presented that focuses on the efficient detection of signatures of individuals with good accuracy and less time required (Zhou et al., 2021). Closely related to the performance evaluation, the examined features included accuracy, AUC, Pr, F1-score, and time (Sharma et al., 2022). The analysis of the obtained results revealed the efficiency of the key steps in the developed framework. However, noteworthy is the fact that the InceptionV3 yielded better results than other models for various datasets with good results in classifying real signature images without producing higher False Positives and False Negatives (Bhattacharya & Bhattacharya, 2022). Furthermore, X and Y reported that data augmentation had similar effects on each of the models they surveyed; however, these effects were positive and negative at different times, providing evidence for the idea that no specific model is universally optimal for all applications and that it is critical to consider the needs of the application while choosing the best model (Li et al., 2022).

In fact, the proposed framework ordered an excellent performance level with the InceptionV3 as offering the highest outcome in virtually all experiments. Furthermore, the low solution's InceptionV3 has a useful outcome toward the Hindi dataset with Accuracy of 99.19%, AUC is 99.93%, sensitivity of 99.04%, precision of 99.37% F1-score of 99.21%. The proposed work makes a useful contribution to the signature verification since it can be implemented in an automated manner without any human interference. The implications of this research go beyond the academy and into practice, where reliable and fast distinctive signature recognition is required.

Subsequent more complex research efforts can be dedicated to improving these models even more, perhaps evaluating the effects of data augmentation on the improvements so that the increases are continuous and lasting.

Refrences

[1] Impedovo, D., & Pirlo, G. (2008). Automatic signature verification: The state of the art. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 38(5), 609– 635. https://doi.org/10.1109/TSMCC.2008.923866

[2] Pal, U., Gupta, S., & Blumenstein, M. (2012). A combined approach for offline signature verification using shape-based features. Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), 761-765. https://doi.org/10.1109/ICDAR.2012.92

[3] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, 25, 1097– 1105.

[4] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4510–4520.

[5] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 770–778.

[6] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278–2324. https://doi.org/10.1109/5.726791

[7] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, 25, 1097– 1105.

[8] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4510–4520.

[9] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 770–778.

[10] Hafemann, L. G., Oliveira, S. R., & Sabourin, R. (2017). Offline handwritten signature verification—Literature review. Proceedings of the IEEE Transactions on Information Forensics and Security, 12(11), 2691-2704. https://doi.org/10.1109/TIFS.2017.2728612

[11] Bhattacharya, U., & Bhattacharya, S. (2022). An Efficient Handwritten Signature

Verification System using Deep Learning Techniques. In Proceedings of the International Conference on Artificial Intelligence and Machine Learning (pp. 47-54).

[12] He, Z., Huang, D. D., & Yau, W. Y. (2022). A Deep Learning Framework for Offline Handwritten Signature Verification. IEEE Access, 10, 44271-44282.

[13] Ammar, R., & Mabroukeh, N. R. (2021). Machine Learning for Network Security. In 2021 International Conference on Communications, Signal Processing, and Systems CSPS (pp. 89- 93). Chang Bai Shan, China: IEEE.

[14] Sharma, N., Sheifali, G., Puneet, M., Xiaochun, C., Achyut, S., Prabhishek, S., & Soumya, R. N. (2022). Offline signature verification using deep neural network with application to computer vision. Journal of Electronic Imaging, 31(4), 041210.

[15] Zhang, Y., Li, Z., He, K., & Liu, J. (2021). Deep Convolutional Neural Networks for Image Processing: A Comprehensive Review. IEEE Transactions on Neural Networks and Learning Systems, 32(8), 3073-3092.

[16] Abbas, S., & Zhou, Y. A. (2022). A Survey of Deep Learning Models for Signature Verification. In 2022 International Conference on Computing, Data Science and Engineering (ICCDSE) (pp. 24- 29). Dubrovnik, Croatia: IEEE.

[17] Aslan, Ö. A., & Samet, R. (2020). A comprehensive review on malware detection approaches. IEEE Access, 8, 6249-6271.

[18] Kothadiya, D., Bhatt, C., Sapariya, K., Patel, K., Gil-González, A. B., & Corchado, J. M. (2022). Deepsign: Sign language detection and recognition using deep learning. Electronics, 11(11), 1780.

[19] Rasheed, A. F., & Alkababji, A. M. (2022). A Novel Method for Signature Verification Using Deep Learning. Webology, 19(1), 1561-1572.

[20] Li, Y., Xu, D., Huang, L., Yang, X., & Gong, Y. (2022). Offline Signature Verification Using a Two- Stream Network. IEEE Access, 10, 42136-42147.

[21] Zhang, S., Zhang, S., Wang, B., & Habetler, T. G. (2020). Deep learning algorithms for bearing fault diagnostics–A comprehensive review. IEEE Access, 8, 29857-29881.

[22] Hafemann, L. G., Sabourin, R., & Oliveira, L. S. (2017). Learning features for offline handwritten signature verification using deep convolutional neural networks. Pattern Recognition, 70, 163- 176.

[23] Abdulhussien, A. A., Nasrudin, M. F., Darwish, S. M., & Alyasseri, Z. A. A. (2023). A Genetic Algorithm Based One Class Support Vector Machine Model for Arabic Skilled Forgery Signature Verification. Journal of Imaging, 9(4), 79.

[24] Joon, D., & Kikon, S. (2015). An Offline Handwritten Signature Verification System- A Comprehensive Review. International Journal of Enhanced Research in Science Technology & Engineering, 4(6), 433-439.

[25] Abdulhussien, A. A., Nasrudin, M. F., Darwish, S. M., & Alyasseri, Z. A. A. (2023). Feature selection method based on quantum inspired genetic algorithm for Arabic signature verification. Journal of King Saud University- Computer and Information Sciences, 35(3), 141-156.

[26] Shashi Kumar, D. R., Raja, K. B., Chhotaray, R. K., & Sabyasachi, P. (2010). Off-line signature verification based on fusion of grid and global features using neural networks. International Journal of Engineering Science and Technology, 2(12), 7035-7044.

[27] Wang, X., Zhang, Y., & Wang, Z. (2022). Deep Convolutional Neural Networks for Image Feature Extraction: A Comprehensive Review. IEEE Transactions on Image Processing, 31, 157-171.

[28] Hafemann, L. G., Sabourin, R., & Oliveira, L. S. (2017). Learning features for offline handwritten signature verification using deep convolutional neural networks. Pattern Recognition, 70, 163- 176.

[29] Zhang, J., Li, C., Yin, Y., Zhang, J., & Grzegorzek, M. (2023). Applications of artificial neural networks in microorganism image analysis: a comprehensive review from conventional multilayer perceptron to popular convolutional neural network and potential visual transformer. Artificial Intelligence Review, 56(2), 1013-1070.

[30] Kensert, A., Harrison, P. J., & Spjuth, O. (2019). Transfer learning with deep

convolutional neural networks for classifying cellular morphological changes. SLAS Discovery: Advancing Life Sciences R&D, 24(4), 466-475.

[31] Kebaili, A., Lapuyade-Lahorgue, J., & Ruan, S. (2023). Deep Learning Approaches for Data Augmentation in Medical Imaging: A Review. Journal of Imaging, 9(4), 81.

[32] Foroozandeh, A., Hemmat, A. A., & Rabbani, H. (2020). Offline handwritten signature verification and recognition based on deep transfer learning. In International conference on machine vision and image processing (MVIP) (pp. 1-7). IEEE.

[33] AbdelRaouf, A., & Salama, D. (2018). Handwritten signature verification using haar cascade classifier approach. In 2018 13th international conference on computer engineering and systems (ICCES) (pp. 319-326).

[34] Mersa, O., Etaati, F., & Masoudnia, S., Araabi, B. N. (2019). Learning representations from persian handwriting for offline signature verification, a deep transfer learning approach. In 2019 4th International Conference on Pattern Recognition and Image Analysis (IPRIA) (pp. 268-273). Tehran, Iran: IEEE.

[35] Salama, M. W., & Aly, M. H. (2021). Deep learning in mammography images segmentation and classification: Automated CNN approach. Alexandria Engineering Journal, 60(5), 4701-4709.

[36] Naz, S., Bibi, K., & Ahmad, R. (2022). Deep Signature: fine-tuned transfer learning-based signature verification system. Multimedia Tools and Applications, 81(26), 38113-38122.

[37] Hameed, M. M., Ahmad, R., Kiah, L. M., Murtaza, G., & Mazhar, N. (2023). Off Sig-Sin GAN: A Deep Learning-Based Image Augmentation Model for Offline Signature Verification. Computers, Materials & Continua, 76(1), 1267-1289.

[38] Kumar, D., Sharma, S., & Mishra, M. P. (2023). Unimodal biometric identification system on Resnet-50 residual block in deep learning environment fused with serial fusion. Global Journal of Enterprise Information System, 15(1), 40-49.

[39] Sharma, S., & Guleria, K. (2023). A systematic literature review on deep learning approaches for pneumonia detection using chest X-ray images. Multimedia Tools and Applications, 1-51.

[40] Jia, L., Wang, T., Chen, Y., Zang, Y., Li, X., & Shi, H. (2023). Mobile Net-CA-YOLO: An Improved YOLOv7 Based on the MobileNetV3 and Attention Mechanism for Rice Pests and Diseases Detection. Agriculture, 13(7), 1285-1289.

[41] Amiri, Z., Heidari, A., Navimipour, N. J., Unal, M., & Mousavi, A. (2023). Adventures in data analysis: A systematic review of Deep Learning techniques for pattern recognition in cyber- physical-social systems. Multimedia Tools and Applications, 1-65.

[42] A. Jeyarani, R., & Senthilkumar, R. (2023). Eye Tracking Biomarkers for Autism Spectrum Disorder Detection using Machine Learning and Deep Learning Techniques. Research in Autism Spectrum Disorders, 108, 102228.

[43] Sharma, N., Gupta, S., Mohamed, H. G., Anand, D., Mazón, J. L. V., Gupta, D., & Goyal, N. (2022). Siamese convolutional neural network-based twin structure model for independent offline signature verification. Sustainability, 14(18), 11484.

[44] using surroundedness feature. Pattern Recognition Letters, 33(3), 301–308.

[45] Chen, S., & Srihari, S. (2006). A new off-line signature verification method based on graph. In 18th International Conference on Pattern Recognition (ICPR’06) (Vol. 2, pp. 869-872).

[46] Mohapatra, R. K., Shaswat, K., & Kedia, S. (2019). Offline handwritten signature verification using CNN inspired by inception V1 architecture. In Fifth International Conference on Image Information Processing (ICIIP) (pp. 263-267).