Neurocomputing 409 (2020) 157–172

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

# OSVFuseNet: Online Signature Verification by feature fusion and depth-wise separable convolution based deep learning

Chandra Sekhar Vorugunti [a,*], Viswanath Pulabaigari [a], Rama Krishna Sai Subrahmanyam Gorthi [b], Prerana Mukherjee [a]

[a] Department of Computer Science and Engineering, Indian Institute of Information Technology, Sri City, India
[b] Department of Electrical Engineering, Indian Institute of Technology, Tirupati, India

ARTICLE INFO

*Article history:*
Received 20 January 2020
Revised 29 April 2020
Accepted 25 May 2020
Available online 1 June 2020
Communicated by Zhang Zhaoxiang

*Keywords:*
Online Signature Verification
Depth-wise separable convolutional neural networks
Few shot learning
Autoencoders

ABSTRACT

Online Signature Verification (OSV) techniques have been deployed in production systems for decades, yet training the model for efficient classification of the test signature from fewer training signature samples is still an open challenge. The advancements in Convolutional Neural Networks (CNNs) enormously boosted the effectiveness of OSV systems. However, learning subtle and discriminating representations from few training samples to classify the genuineness of test signature has not been explored fully. In this paper, a Convolution Autoencoder (CAE) is used to obtain high-level feature representations from the input signature and these high level features are fused with handcrafted features to constitute a hybrid feature set. The hybrid set of features is presented as an input to an Online Signature Verification framework made up of Depth-wise Separable Convolutional Neural Network (DWSCNN). DWSCNN effectively learn deep feature representations with fewer training samples and parameters than traditional CNNs resulting in a light weight OSV framework. Thorough experimental analysis on three benchmark datasets MCYT-100 (DB1), SVC-2004-Task2 and SUSIG-Visual corpus confirm that the proposed hybrid fusion of feature set and DWSCNN based OSV framework achieve higher classification accuracies and outperforms many contemporary and state-of-the art OSV models.



## 1. Introduction

Online Signature Verification (OSV) system intends to classify whether a test signature genuinely represents an individual or not by learning the exclusive features of the signature. The advances in technology resulted in the drastic use of high-end mobile computing devices like Smart Phones, Graphic Tablets, PCs, etc. in m-commerce and m-payment applications [1–5]. Due to various security pitfalls [6–8] and the unease of manual entry of password on these devices turnout a demand for auxiliary biometric authentication techniques. Owing to inherent advantages like social acceptance for personal authentication, ease of collection and complication in alteration made Online Signature Verification (OSV) a conventional and legitimate mechanism to certify a person's identity in critical digital transactions [2–5]. Online signature is defined by discrete dynamic signals and is acquired using smart devices, which facilitate reading both the spatial coordinates (x, y coordinates) and the dynamic attributes (namely stroke order, pressure, azimuth, velocity, total signature time, acceleration, etc.) [1,9–16].

In the current years, the advent of Deep Learning (DL) technologies like Convolution Neural Networks (CNN) results in higher accuracies in solving classical problems like image classification [17], object detection [18], scene text recognition [19], image segmentation [20] etc. in comparison to traditional pattern recognition techniques. Even though DL techniques perform superior, they mainly suffer from two intrinsic drawbacks: 1. Need of larger number of training samples to learn the discriminating features, 2. Need of longer training time due to the involvement of a large number of parameters (weights, biases) to be learnt by the model.

The widespread use of real time signatures on high end devices to authenticate and validate a large number of users on a day-to-day basis in critical E-commerce and M-commerce applications demands for deploying modern neural networks in memory-limited scenarios and still achieve higher classification accuracies. To adopt DL techniques in OSV, there is a need to address the two pitfalls discussed above. As depicted in Fig. 3, the OSV architecture consists of two entities: 1. Input feature set 2. Online verification

* Corresponding author.
*E-mail addresses:* chandrasekhar.v@iiits.in (C.S. Vorugunti), viswanath.p@iiits.in (V. Pulabaigari), rkg@iittp.ac.in (R.K.S.S. Gorthi), prerana.m@iiits.in (P. Mukherjee).

https://doi.org/10.1016/j.neucom.2020.05.072
0925-2312/

framework. These two entities are the scope of action to address the above requirements. To address requirement 1, a feature-level fusion [21] is implemented, where in a representation feature vector is formed using handcrafted features and deep features returned by a convolution auto encoder. The hybrid feature set resulted from lesser number of samples as minimum as one, is able to classify the test signature accurately. More about the fusion technique used is discussed in subsequent sections. To address the requirement 2, we propose a Depth-wise separable convolution neural network based OSV framework called OSVFuseNet, which uses depth-wise separable convolutions [22] instead of standard convolutions. Depth-wise separable convolution is a set of Depth-wise convolution and Point-wise convolution operations. Depth-wise separable convolution requires fewer parameters and is computationally effective and still performs equally with the standard convolution operation. Hence, Depth-wise separable CNNs are more suitable for vision and classification applications in limited memory and computational power scenarios. We have discussed in detail about depth-wise separable convolutions in subsequent sections.

Overall organization of the manuscript is as follows. Related works and contributions of the paper are provided in 2. The motivation behind selecting various constituents of our proposed OSV architecture is discussed in 3. Background on Depth-wise separable convolution operation and autoencoder are provided in 4. The methodology of the proposed OSV framework is elucidated in Section 5. A thorough discussion on implementation, results and comparative examination are explained in Section 6 and the Section 7 concludes the manuscript.

## 2. Related works and contribution

An online signature is described as an agglomeration of stroke points, where each stroke point represents a set of properties like pressure, azimuthal angle etc, which are called as local features. Global features, which represents the whole signature are extracted by applying statistical methods on all local features. In the current literature, many techniques towards automated OSV have been proposed, which are predominantly categorized into feature-based technique [1–3,9–12,23–29] considering that interpret signatures as a collection of local or global features and function-based techniques which apply diverse techniques like feature fusion [1,15], divergence [30], feature weighing [31,32], random forest [33], Hidden Markov Models [4], Gaussian Mixture Models [11,13], sequence matching [4,25], DTW [11,28,34–37], stability [38–40], writer specific classifier [41], neural networks [42], Deep learning [17,?,19,?,?,?,?]. In recent works, diversity of handcrafted features of reference signatures like sigma lognormal decomposition [46], features by means of the Relief algorithm [31], Probabilistic-DTW [14], Hausdorff distance based features [47] and stroke based [48,37] etc. has been computed and produce state-of-the-art outcomes in the domain of online signature verification. Similarly, recent deep learning-based models [34,38,43–45,49–52] has made great advancements in pattern recognition and machine learning tasks. For example, a neural network based autoencoder is utilized to extract the constrained relevant representational features from given input source like image, video and online signature samples, to minimize the reconstruction error for content representation [53]. Furthermore, the Convolution Neural Network (CNN) is capable of learning the universal representations and function approximation from the original source [44,51,54]. Compared to the handcrafted features, the deep representations learnt by the CNN enriches the rich semantic information while achieving low error rates [17–20,34,36,43]. Overview or categories of various recent frameworks are discussed below:

In literature, to evaluate the classification accuracies of OSV frameworks, the metric called Equal Error Rate (ERR) is used, which represents an intersection point of False Acceptance rate (FAR) and the False Rejection Rate (FRR) in receiver operating characteristic (ROC) curve. The lower is the EER, the higher is the efficiency of the framework [9,13].

### 2.1. Matching based OSV frameworks

Riesen et al. [7] designed an OSV framework builds upon string edit distance (SED) as an alternative to Dynamic Time Warping (DTW). SED allows removal and inclusion of symbols in addition to allowing only substitutions. Based on their cost model, Riesen *et.al* demonstrated that SED is more effective than the widely used DTW. The model realized an EER of 9.10%, while using every $10^{th}$ sample point of the underlying signatures.

Parziale et al. [55] devised a Dynamic Time Warping (DTW) based OSV model based on the concept of stability. Highly stable segments of the signature are calculated from the training samples. The matching of stable parts between the template and the test signature is computed to classify the test signature. In regard to MCYT-100, the model outcome an EER of 1.30% and 3.09% EER with respect to random and skilled forgeries respectively.

Sae-Bae et al. [56] designed three statistics based novel measures, i.e. distinctiveness, complexity and repeatability to appraise the characteristics of online signature templates. These three measures are used to classify the test signature. The Sae-Bae et al. [56] model achieved a False Rejection Rate (FRR) of 1.14%.

He et al. [47] designed a novel OSV framework wherein for each dynamic feature, the curvature and torsion features are obtained. The curvature of a signature expresses the deviation of the signature curve from a straight line and torsion illustrate the measure of distortion. An 8-dimensional feature vector is computed and stored as a matching template for each user. In the testing phase, from the test signature a 8-dimensional feature vector is calculated and Hausdorff distance between the template and the test signature is computed and classified as genuine or forgery accordingly.

### 2.2. Deep learning based OSV frameworks

Apart from the above traditional OSV techniques, the advancements in Deep learning technologies resulted in the application of advanced deep learning techniques to Online Signature Verification.

Lai et al. [43] devised an OSV framework, in which signature verification is formulated as a sequence modelling problem. A Recurrent Neural Network (RNN) is trained to learn a novel metric called the length-normalized path signature (LNPS), which represents the distance between genuine and forgeries. The model achieved scale and rotation invariance in signature verification. Evaluation with SVC dataset resulted an EER of 2.37%.

Tolosona et al. [34] designed an OSV framework, which constitutes Siamese and Long Short-Term Memory (LSTM) networks to learn the disparities between a combination of signatures. The framework is evaluated on BiosecurID dataset and resulted 6.44% of EER in the case of skilled 1 category.

Xiaomeng et al. [36] devised a dynamic time warping based OSV wherein a Siamese network is trained to learn the temporal location-wise distances between time series. The learnt feature space is provided as an input to a dynamic time warping block to learn the intra-writer variability. The model is evaluated only in skilled 5 category which records an EER of 1%. Due to inadequate experimental analysis, ratifying the Xiaomeng et al. model for real time deployment scenarios is impractical.

Chuang et al. [48] designed a Stroke-based RNN for OSV. In Chuang et al. model, based on stokes, the signature is split into numerous patches. The query signature and the writer's specific reference signature are fed as input to the RNN model for classification into either genuine or forgery. The framework realized 10.46% of EER in skilled 1 category. The main pitfall in this method is that the experimental evaluation is not elaborate. The model is not evaluated with only skilled 1 category of experimentation and the model is evaluated with only 50 users, compared to 100 users. The model performance is verified only in skilled 1 category, which doesn't gain the confidence of the model for real time deployability.

Diaz et al. [57] devised an OSV framework grounded on Robotic Arm Motion analysis. In Diaz *et.al* work, anthropomorphic features are extracted by stimulating the human upper limb bone movements with robotic arms. Similar to the above discussed methods, the main pitfall in this method is that the experimental evaluation is not elaborate. The model performance is verified only in skilled 5 category of MCYT-100.

In a more recent work, Diaz et al. [58] putforward an OSV framework wherein the features which are not captured by the digital device are computed depending on the movements of the wrist, the elbow and the shoulder joints. These features are computed by simulating real arm movements using Virtual Skeletal Arm (VSA) model. Even though it is novel attempt, the main pitfall in the proposed model is the limited experimental evaluation. The model is evaluated only for Skilled 5 and Random 5 categories. With a limited amount of evaluation, the model is not authenticated to deploy in real time environments.

### 2.3. Efficient Convolution Neural Network (CNN) architectures and feature reduction techniques

Chaoqun et al. [59] presented a novel Manifold Regularized Convolution Layers (MRCL) based deep CNN architecture (MRDL). MRDL output Manifold Regularized deep representational feature vector in low rank space.

Jun et al. [60] proposed a novel CNN architecture termed VeckerNet, in which a standard convolution layer with f × f filters are replaced with two convolution layers of f × 1 and 1 × f filters respectively. This resulted in effective reduction of CNN parameters without loss of recognition accuracy rate.

Shyam et al. [61] put forward a hybrid learning algorithm in which inner layer parameter of multilayer neural network is trained with a Random Weight Change algorithm and the output layer is trained with the delta rule. This results in elimination of back propagation operation and gives rise to better performance compared to traditional back propagation based models.

Chaoqun et al. [62] proposed a novel 3D pose reconstructing network, in which multiple set of features is fused using multimodal learning. The extracted features are passed as an input to the Multimodal Deep Autoencoder (MDA) coupled with denoising autoencoder, which extract robust features for efficient 3D representation.

Chaoqun et al. [63] proposed a novel Multimodal deep autoencoder (MDA) based framework for 3D pose recovery. A Laplacian matrix and Sparse coding technique is applied to measure the similarity between training samples and each test sample.

The above discussed optimal CNN architectures and efficient feature representation techniques can be adapted to Online Signature Verification, which results in a light weight framework in regard to the number of parameters per layer and number of features to be processed per signature.

Despite the ability to process voluminous data and yields higher classification accuracies, the prerequisite for CNN based verification frameworks is a substantial amount of training signature samples for every enrolled writer to efficaciously review the inter and intra-individual aberrations [1,9,10,23,13] to precisely categorize the genuineness of signatures [13,56]. The delicacy of applications like m-payment and m-banking, it is normally impossible to acquire sufficient amount of signature samples from each user [23,46,64]. Hence, for usage of OSV frameworks in real time scenarios, it is essential for the frameworks to learn the discriminating features from as minimum as one training signature sample and outcomes higher classification accuracies. In the literature, a very less number of works examined the feasibility of OSV systems with few shot learning, i.e., ability to learn the writer specific features with few training signature samples, specifically one sample per user. Galbally et al. [11] devised an OSV framework in which, Hidden Markov Models are used to generate the fabricated signatures from a set of genuine signature samples. The model accomplished an EER of 16.07%. Another work in congruent lines is by Diaz et al. [46], wherein fabricated signature samples are generated from original samples using sigma-lognormal parameters and applying the kinematic theory of rapid human movements. Diaz et al. [46] verification model outcome 13.56% EER. Similarly, various other hybrid methods for OSV have been developed.

Challenged by the scalability of real time OSV systems, we propose OSVFuseNet that can realize few shot learning, including one shot learning by making use of Depth-Wise Separable Convolutions (DWSC). The major contributions of our work in the paper are outlined as follows:

### 2.4. Contribution of the paper

1. We propose a novel Depth-wise Separable CNN (DWSCNN) architecture named OSVFuseNet for signature classification, where Depth-wise separable convolution operations automatically learn a latent representation to classify the input online signature. We have achieved a significant drop in the number of operations required to be performed by the framework in learning the relevant feature representations, even from the lesser number of signature samples as minimum as one and still accomplish a superior classification accuracy.
2. We have developed a modified feature ranking and feature fusion technique based on a feature ranking criteria discussed in [52] to fuse hand-crafted features with deep representative features of a convolution auto-encoder to form a hybrid feature set. As depicted in Fig. 3, top 60% of the features which are selected based on the proposed feature ranking technique and 40% of latent representations from a convolution autoencoder are fused to form joint representations. These joint representations form an input to our DWSCNN framework for classification.
3. We have thoroughly evaluated the effectiveness and generality of the proposed method for few shot learning by performing thorough experimental analysis on three extensively used OSV datasets i.e MCYT-100, SVC and SUSIG.
4. We have appraised the superiority of the proposed framework by comparing the proposed OSV framework with the latest and state of the art frameworks. The proposed approach is found to surpass the recent state of the art works. In MCYT dataset, it is found to exceed the recent state-of-the-art work [36] Stroke wise and Target wise by 0.34% in skilled 1 category, Anthropomorphic features [58] by 0.34% in skilled 5 category. In SVC dataset, it outperformed the same Stroke wise and Target wise wrt method in Ref. [36] by 12.3% in skilled 1 category. The proposed framework demonstrated similar performance in multiple categories of experimentation, which we discuss in detail in experimentation section.

## 3. Motivation

In this section, we discuss in brief about selecting various constituents of our proposed framework:

### 3.1. AutoEncoder

The widely used projection-based dimensionality reduction techniques in literature are Principal Component Analysis (PCA), and Linear Discriminant Analysis (LDA). The major pitfalls with these techniques are that they do projection linearly, which is equivalent to a linear autoencoder [65]. Hence, they are inefficient in approximating complex functions.

Autoencoders and its variants like Convolutional AutoEncoder (CAE) have recently shown excellent outcomes in challenging computer vision problems involving unsupervised feature learning [53,66,67]. An input signature device may induce noise into an online signature while signing. The global handcrafted features computed for these noise induced signatures are not efficient to represent writer signature patterns. Autoencoder learns a constricted representation of the input while retaining the most important information with reduced dimensions and minimum reconstruction error.

### 3.2. Advantages of Convolution Auto Encoder (CAE) over AutoEncoder (AE)

A Fully Connected layer based Autoencoder can produce very useful representations of the inputs. However, the main drawback with AE is that they consider each feature of a feature set is independent and cannot model the relationship between the features of a feature set [68]. Convolutional Auto Encoder (CAE) performs 1D convolution operations between 1 dimensional input signature feature vector and a set of kernels. Convolution operation extracts meaningful local patterns, hierarchical relationships among the features, which is appropriate for signature classification. In addition to it, the weights are shared between the layers, which reduces memory requirements compared to AE [69].

We have evaluated the training and testing loss in multiple experimentation categories of all the datasets wrt to AE and CAE. As depicted in Figs. 1 and 2, in case of AE, the training loss and testing loss start at higher value compared to CAE. In addition to it, CAE losses converge smoothly to much lower values as compared to AE. Similar observations are found in multiple experimentation categories. Based on these theoretical and experimental analysis, we have chosen CAE over AE.

## 4. Preliminaries of depth-wise separable convolution neural networks and convolution autoencoders

In this section, we succinctly discuss the background about important components in the proposed framework:

### 4.1. Depth-wise seperable convolution

The recent advancements [27,49,54,70,71] in deep learning techniques resulted in Depth-Wise Separable (DWS) convolutions for faster and accurate learning of representative features by the convolution layers. DWS convolution is a sequence of depth-wise convolution and $1 \times 1$ Point-wise convolution. In Depth-wise convolution, the convolution operation is implemented individually in each channel with the respective associated kernel. Subsequently, $1 \times 1$ point-wise convolution is implemented on these channel outputs, that were resulted from the depth-wise convolutions.

Mathematically, the standard 2D-2D convolution resulting from input of $c_{in}$ (2D) channels to output of $c_{out}$ (2D) channels is described as below:

$$Conv(I,K) = OP(a,b,c_o) = \sum_{c_i=1}^{c_{in}} \sum_{p=-k/2}^{k/2} \sum_{q=-k/2}^{k/2} K(p,q,c_i,c_o) \times I(a-p,b-q,c_i) \quad (1)$$

Here $1 \leqslant a \leqslant W,\ 1 \leqslant b \leqslant H$, and $1 \leqslant c_o \leqslant c_{out}$; I represents the input layers of size $W \times H \times c_{in}$, where W = width, H = height, $c_{in}$ = number of input channels. K is a convolutional kernel of dimension $k \times k \times c_{in}$ and $c_{out}$ such kernels results in output feature map OP of dimension $W \times H \times c_{out}$. For brevity, it is considered that the size of an output feature map is same as spatial size of an input map and it can be managed by zero-padding operation.

Depth-wise separable convolution is described as Depth-wise Convolution (DWC) (2) at each of the input channels followed by Point-wise Convolution (3) of these input channels to give rise to the required number of output channels.

Mathematically, the standard depth-wise convolution on input of $c_{in}$ (2D) channels with K convolutional kernels of dimension $k \times k \times c_{in}$ is described as below:

$$DC(a,b,c_i) = \sum_{p=1}^{k} \sum_{q=1}^{k} K(p,q,c_i) \times I(a-p,b-q,c_i) \quad (2)$$

Here $1 \leqslant a \leqslant W,\ 1 \leqslant b \leqslant H$. The $c_i^{th}$ filter in K is applied to the corresponding $c_i^{th}$ channel of input I to produce the $c_i^{th}$ channel of the depth-wise convolution output feature map DC.

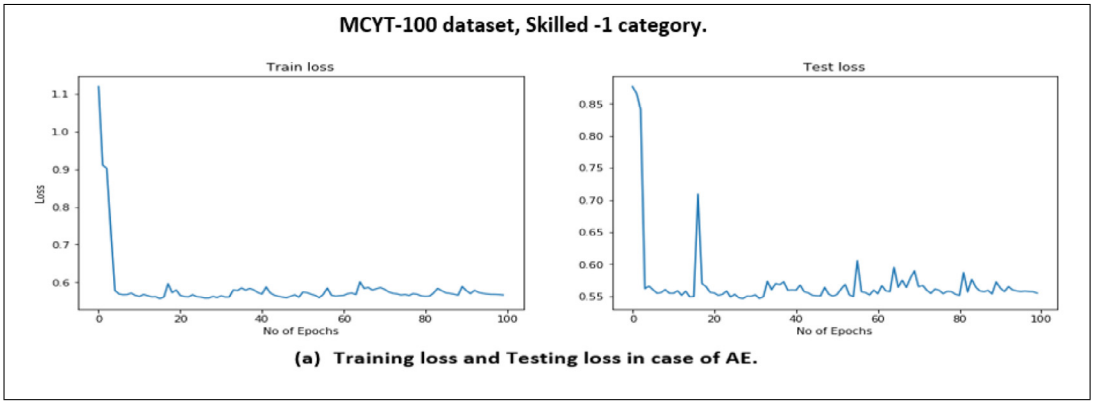


**Fig. 1.** The train and test loss in case of a fully connected layer based AE.

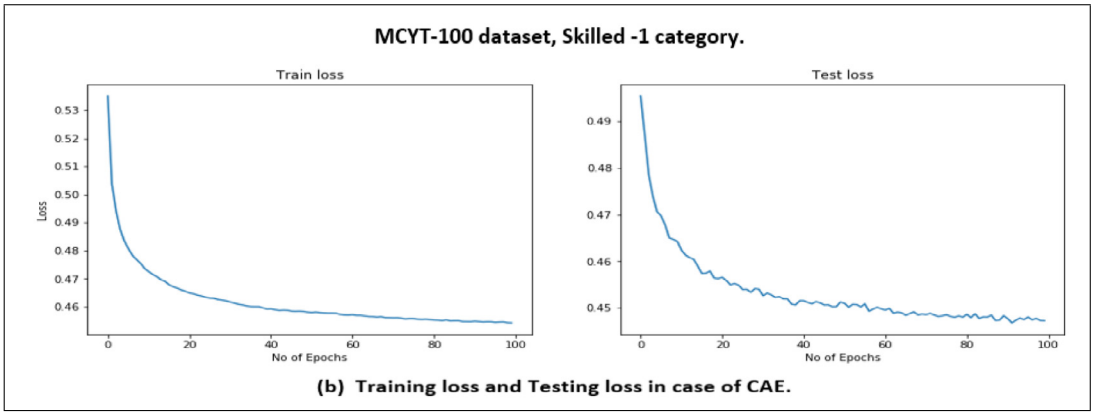


**Fig. 2.** The train and test loss in case of convolutional AutoEncoder.

This is followed by the point-wise convolutions with P ($c_{in} \times 1$) weights to give rise to $c_{out}$ final output feature maps (OP) of DWS convolutions.

$$OP(a,b,c_o) = \sum_{c_i=1}^{C_{in}} DC(a,b,c_i) \times P(c_i,c_o) \quad (3)$$

Here $1 \leqslant c_o \leqslant c_{out}$. The Pointwise convolution results in a set of features, which is a linear combination of the output from a depth-wise convolution via $1 \times 1$ convolution. As depicted in Figs. 3–5, online signature is described as a $(1 * M)$ feature vector. If we consider $W = 1$ and $H = M$ for input signature and kernel of size $1 \times k$, the above Eqs. (1)–(3) represent an online signature which are of one-dimensional.

### 4.2. AutoEncoders (AE)

As depicted in Fig. 3, Autoencoder [33,53] is a type of feed forward artificial neural network constitute of three entities: input layer, encoder, and a decoder. The encoder condenses the input into a compact knowledge representation of the original input through an unsupervised deterministic nonlinear function. Let input $\mathbf{S} = (f_1, f_2, f_3, f_4, f_5, \ldots, f_{100})$, which encodes $S$ to $\mathbf{H} = (f'_1, f'_2, f'_3, f'_4, f'_5, \ldots, f'_{80})$ through $f(W_e.\mathbf{S} + b)$, where $f(.)$ is component wise nonlinear function. The decoder reconstructs the original input from the intermediate representation $H$ through $S' = g(W_d.\mathbf{H} + c)$, $g(.)$ again being a component wise nonlinear function. The AE is trained to lessen the reconstruction loss function L,

$$L(S, W_e, W_d, b, c) = min_{W_e,W_d} \frac{1}{2} \left\| g(W_d(f(W_e.S + b)) + c) - S' \right\|^2 \quad (4)$$

which ensure that $S'$ is close to $S$, by appropriately learning $W_e$ and $W_d$. AE is trained in Unsupervised setting as per the above MSE loss.

## 5. Proposed Online Signature Verification framework

The first step in Online Signature Verification (OSV) is extracting effective, discriminating features from the acquired online signature. The second step is to build the efficient model/framework which uses the feature set to classify the genuineness of the query signature. In traditional feature extraction process, while the user is signing on a specialized device, dynamic local features include x, y-coordinates, stroke order, azimuthal angle, pressure etc. are captured at every sampling point of the signature. Based on these local features, global features which represent the entire signature are calculated. User is requested to refer Guru et al. [9] to know in detail about the global features and statistical formulae associated with them to compute 100 features of MCYT-100 dataset. However, research [21,72] concludes that the low level features com-

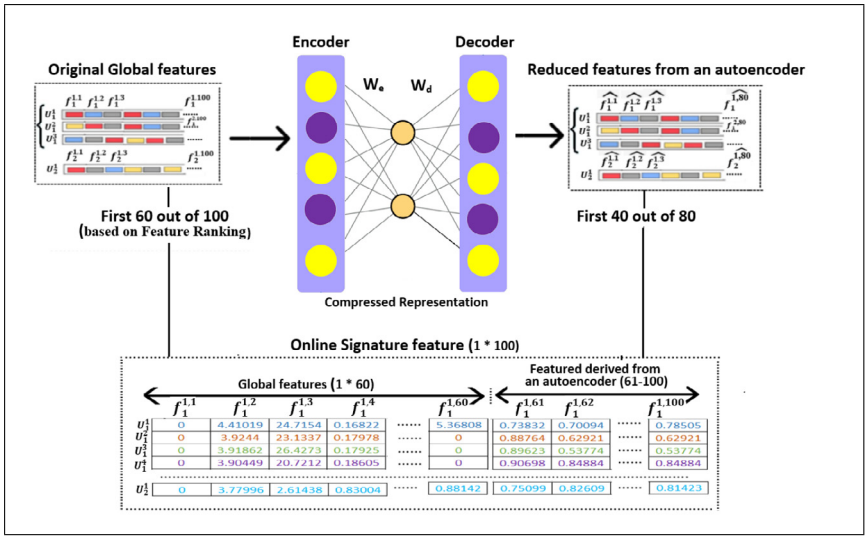


**Fig. 3.** The architecture of the proposed OSV framework.

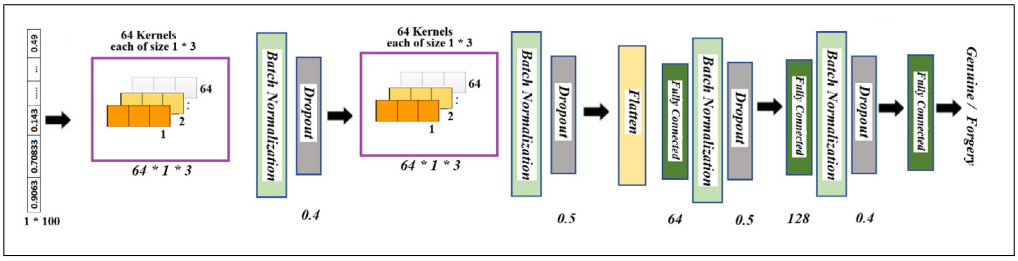


**Fig. 4.** Diagrammatic representation of separable 1D convolutional based proposed OSV framework architecture.

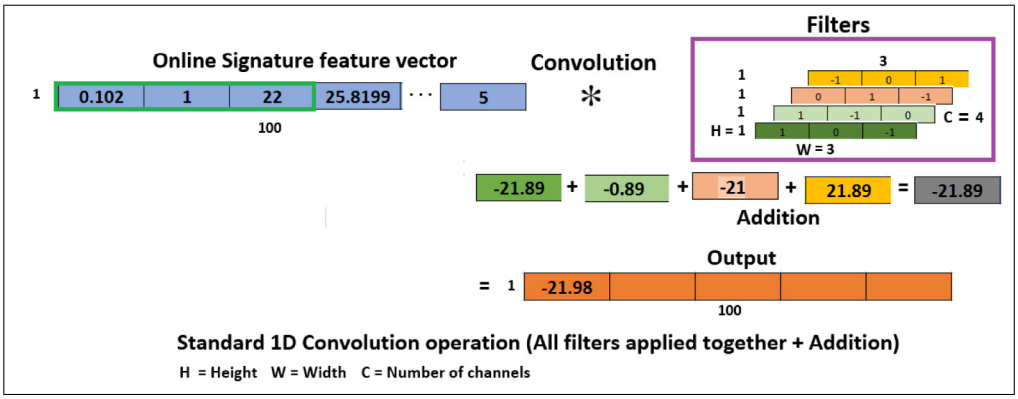


**Fig. 5.** A numerical example for standard convolution 1D operation.

puted based on these traditional techniques lack in representation and generalization competence.

### 5.1. Online signature format

As illustrated in Fig. 4 and Fig. 5, an online signature is expressed as a one-dimensional hand-crafted feature vector of dimension $1 \times 100$ with respect to MCYT-100 dataset and $1 \times 47$ for SVC and SUSIG datasets respectively. The dimensionality 100, 47 implies the number of global features [9] enumerated for each writer's signature, e.g. max velocity, signature's total duration, centripetal acceleration, standard deviation of y-coordinate, average jerk etc.

### 5.2. Proposed feature ranking algorithm

In this section, we discuss about our proposed feature ranking algorithm which assigns ranking to each individual feature based on its between class variability and within class variability is described below. Kuang et al. [52] proposed a feature ranking algorithm for selecting useful and competent blocks of a tensor. A tensor is a multidimensional data array and a block contains multiple sets of features representing a part of an image. In Kuang et al. [52] algorithm, ranking score is computed for an entire block not for each individual feature. Hence, as portrayed in Figs. 3 and 7 to select the best features from a set of $H$ features (100 for MCYT, 47 for SUSIG, SVC), we have proposed a modified version of the feature ranking algorithm proposed by Kuang et al. [52]. We discuss about their ranking scheme as follows:

$$m_c = \frac{1}{N_c} \sum_{i \in class\ c} f_i \tag{5}$$

$$d_c = \frac{1}{N_c} \sum_{i \in class\ c} (f_i - m_c)(f_i - m_c)^T \tag{6}$$

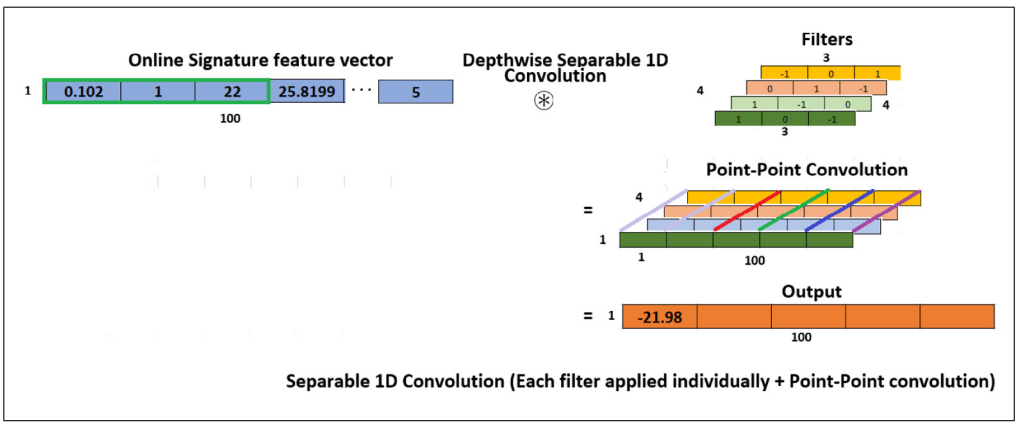


**Fig. 6.** A numerical example for depth-wise separable convolution 1D operation.

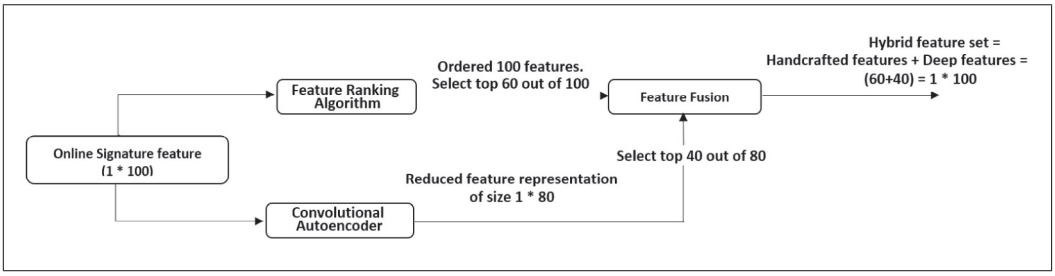


**Fig. 7.** Diagrammatic representation of feature fusion technique used in our framework.

$$S = \frac{\sum_{j=1}^{C}\sum_{m\neq j}(m_c - m_d)(m_c - m_d)^T}{\sum_{j=1}^{C} d_j} \tag{7}$$

$f_i$ is a row vector of size 1 × H, represents the feature vector of a block of an image. $m_c$ is a row vector of size 1 × H, represents the mean of all feature vectors of $c^{th}$ class. $d_c$ is a scalar quantity of size 1 × 1 representing the summation of distance between each feature vector and corresponding mean vector of class c. The numerator of (7) represents the distance between a feature vector and a mean vector of a different class. Therefore, the numerator of (7) represents the between class distance of a feature, which is a scalar quantity of size 1 × 1. Finally, the sum of between class distance is divided by the sum of within class distance to compute the feature ranking score S for an entire block.

Our proposed feature ranking algorithm which assigns ranking to each individual feature is described below:

Let $s_k$ be $k^{th}$ sample of dimension 1 × H, belonging to either genuine or forgery class (with a class label $c = 1$ or 2). Then, the mean feature vector for the respective classes can be elucidated as

$$m_c = \frac{1}{N_c}\sum_{k,s_k\in class\ c} s_k \tag{8}$$

Here $c = 1, 2$ and $m_c$ is × H. Now the sum of within class variance for each of the feature dimensions of all samples is given by

$$d_w = \sum_{c}\frac{1}{N_c}\sum_{k,s_k\in class\ c}(s_k - m_c).*(s_k - m_c) \tag{9}$$

Here '.*' represents element wise multiplication. Note that the sum of within class variance of feature dimensions represented by $d_w$ is 1 × H vector, representing the variability/spread of each feature within that particular class. Now the between class variance of feature dimensions can be given by $d_b$.

$$d_b = \sum_{c}\frac{1}{N_c}\sum_{k,s_k\notin class\ c}(s_k - m_c).*(s_k - m_c) \tag{10}$$

Now an important feature must have high between class variability and low within class variability. Thus the features can be ranked by following measure capturing the ratio of the between class variability to within class variability.

$$FRS = d_b./d_w \tag{11}$$

Here './' represents the element wise division. The feature ranking score (FRS) is a 1 × H vector containing the Feature Ranking Scores for each of the corresponding features. Note that FRS as defined above must he high for better/good feature. Thus, by sorting the scores/values in FRS in descending order, we can sort the features wrt their importance from top to bottom.

### 5.3. Intuition behind fusion of handcrafted and convolutional auto encoder features

The advances in deep learning technologies [21,36,73] enable the creation of deep convolution based neural networks for solving challenging computer vision problems. The initial and mid level convolution layers extract the low level features of an input, that are quite comprehensive and to some extent independent of any definitive classification task. The final layer extracts the higher level features, through which it learns intra-class variability of the signatures by complex non-linear transformation. Thus, on providing the hand crafted global feature set as an input to the CAE, it extracts deep representative features characterizing the spatial, hierarchical and local sensitivity relationship among the hand crafted features of a signature. As the network goes deeper, more sophisticated representations of input hand crafted feature set can be learned [74]. This inspires us if the combination of deep representative features from CAE and handcrafted features results in better performance in EER of the framework. To augment the discriminative information of a signature, we have performed a feature level fusion of handcrafted features and corresponding deep features from a Convolution Autoencoder. As summarized in latest research [75], the hybrid feature set takes advantage of both the handcrafted and deep representation features and results in higher classification accuracy.

### 5.4. The procedure followed to select the best percentage fusion of handcrafted and deep features

As depicted in Figs. 3 and 7, an online signature, which is interpreted as one dimensional feature vector of size $1 \times H$, forms an input to both the feature ranking algorithm and Convolutional AutoEncoder. 'H' represents dimension of an input feature vector, where H = 100 in case of MCYT-100, 47 in case of SVC and SUSIG datasets. We explain the feature fusion process considering the case of H = 100 (MCYT-100). Similar is the procedure with SVC and SUSIG datasets.

**Step 1:** An input signature feature vector (ISFV) of size 1 × 100, is given as input to our proposed feature ranking algorithm, which outputs a vector (FRS) of size 1 × H representing the feature ranking score of each feature. A decreasingly sorted feature vector (SFV) based on an FRS score of corresponding feature is generated from ISFV.

**Step 2:** Similarly, the same input signature feature vector (ISFV) of size 1 × 100, is passed as an input to the Convolutional AutoEncoder without label information. Hence, it is an unsupervised setting for convolutional auto-encoder feature extraction. As depicted in Fig. 3, CAE reconstructs the input at the output layers through MSE loss and outputs a latent representation feature vector (LRFV) of size 1 × 80.

**Step 3:** Initially, we select first 20% of H i.e 20 features from sorted feature vector (SFV) and 80% of H i.e 80 features from latent representation feature vector (LRFV).

**Step 4:** Similar to Ref. [21], we perform a feature-level fusion to combine the features by concatenating 20-dimensional hand crafted and 80-dimensional latent representation features. The resultant hybrid fused feature vector is of size $1 \times 100$, where first 20 represents hand crafted features and the next 80 represents CAE features. The hybrid fused vector of size $1 \times H = 1 \times 100$ is given as an input to our proposed DWSCNN framework, and the resultant EER is recorded.

**Step 5:** Likewise, step 3 to step 4 is executed with varying the percentage of hand crafted and CAE features to (30,70), (40,60), (50,50), (60,40), (70,30), (80,20) and the resultant EER outcome of the proposed framework is recorded. The percentage combinations which resulted in least EER is taken as a final percentage mix. Table 7 illustrates the EER outcome by the proposed framework for various combinations of hand crafted and deep features. As depicted in the table, (60,40) is resulting the least EER in the majority of experimentation categories. Hence, we have selected 60% of hand crafted features and 40% of deep representational features in our work.

### 5.5. Analysis of parameter settings for proposed DWSCNN based framework

To select the best combination of parameters for the proposed OSV framework, we have conducted a thorough set of experiments by performing a grid search for the parameters. The parameters to set in our OSV framework are: the number of separable convolution layers (SL = 2,3,4), number of kernels in each separable convolution layer (K = 16,32,64), kernel size (KS = 3,5), number of dense layers (DL = 2,3), number of nodes in each dense layer (DN = 16, 32,64,128), Activation function (A = 'relu','tanh'), optimizer (O = ' Adam','SGD'), the initialization values for kernel and bias. Achieving one shot learning is a critical requirement for OSV framework. Hence, to fix the parameter combination for our framework, we have evaluated the framework with Skilled_01 category of MCYT-100 dataset.

Based on recent research [35,43,45], initially, we have set A = 'Relu', optimizer = 'Adam', Kernel Initializer = 'random_ uniform', bias initializer = 'random_ uniform'. Our motivation is to find the combination of the parameters, which results in least EER outcome by the proposed framework. Few of the results of the various combinations of the above parameters are depicted below.

As illustrated in Fig. 8, we have set the number of separable convolution layers = 2. By keeping the number of layers fixed, we have altered the other parameters and plotted the EER outcome of the framework by varying the learning rate from 0.001 to 0.008.

As illustrated in Fig. 9, we have set the number of separable convolution layers = 3. By keeping the number of layers fixed, we have altered the other parameters and plotted the EER outcome of the framework by varying the learning rate from 0.001 to 0.008.

As depicted in Fig. 8c, the combination of parameters SL = 2, K = 64, KS = 3, DL = 2, DN = (64 nodes in the first layer and 128 nodes in second layer), A = Relu, O = Adam, learning rate = 0.004 results in least EER value of 13.38%. Based on these experimental analysis on Skilled_01 category of MCYT-100 dataset, we have opted final set of parameter combinations for the proposed framework. Once we fixed the parameters, we have used the same set of parameters for all categories and datasets of our experimentation.

### 5.6. Separable convolution layer

As illustrated in Fig. 4, the DWSCNN constitute a group of five layers. An illustrative example for 1D separable convolution is given in Fig. 6. The entire architectural details of the DWSCNN are given in Table 3. A $1 \times 100$ online signature is fed as an input to the first DWS convolution layer. Each DWS convolution layer consists of 64 filters of length $1 \times 3$ which outputs 3 feature maps, each of size $1 \times 100$. A $1 \times 1$ Point-wise convolution operation is performed on these feature maps, to output 64 feature maps with $100 \times 1$ features in each map. Similar operations are performed in the second DWS convolution layer which produce a next set of 64

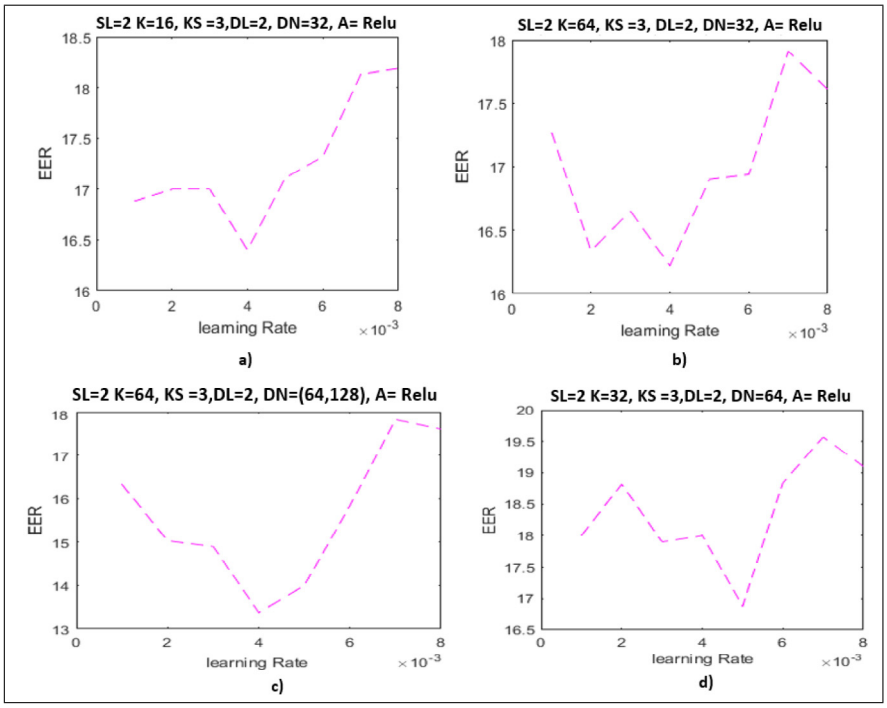


**Fig. 8.** The EER outcome of the proposed framework by fixing the number of separable convolution layers = 2 and varying other parameters of the framework.

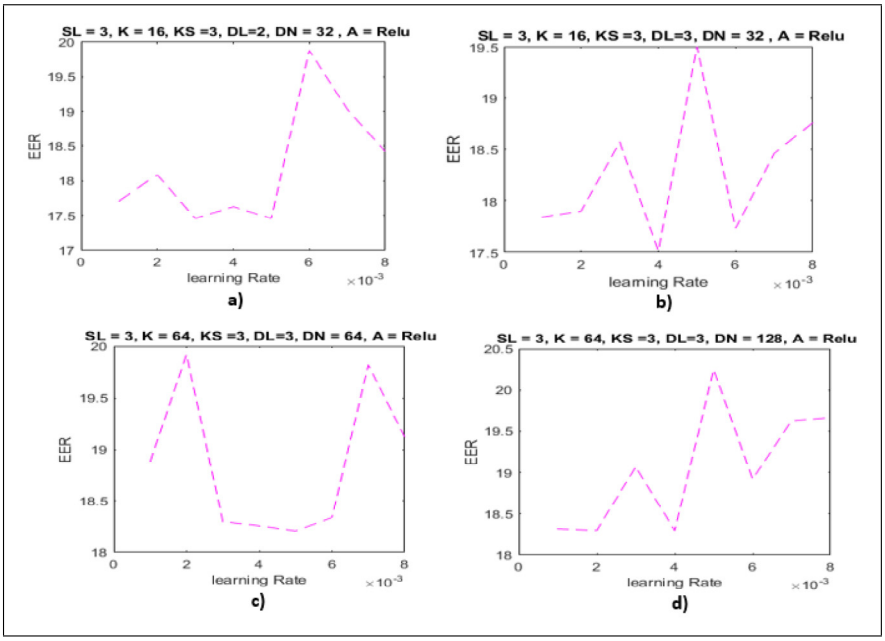


**Fig. 9.** The EER outcome of the proposed framework by fixing the number of separable convolution layers = 3 and varying other parameters of the framework.

feature maps of $100 \times 1$ dimension. To reduce overfitting of the framework, an appropriate dropout is applied at each layer, which is considered as a better convention in the literature [76]. To retain the dimensionality of the output vector from the convolution operation to be same, we have employed appropriate padding. As shown in Tables 1 and 2, the application of optimization techniques like setting initial weights to kernel and bias, batch normalization, yields reduced overfitting and greater classification accuracies. As depicted in Table 2, compared to a standard 1D convolution operation, the depth-wise separable 1D convolution requires 3.79% lesser parameters. The decreased parameter count leads to elevated representational efficiency and faster framework learning.

### 5.7. Fully connected (FC) network

The output from the last DWS convolution layer is a multidimensional deep representative feature vector of size $100 \times 64$, which is provided to a flatten layer. Flatten layers converts the feature vector into a linear vector of size $1 \times 6400$ and input these features to a single hidden layered fully connected classifier. A vector of size $1 \times 128$ forms an output from the first FC layer. The second FC layer yields a vector of length $1 \times 128$, which is provided to the batch normalization and dropout layers, which turnouts a high-level feature vector of length $1 \times 128$. The softMax layer assigns decimal probabilities to each class, i.e. genuine, forgery. As OSV is a two-class classification problem, i.e. 1. genuine 2. forgery, accordingly, to train the framework based on the error between the ground truth and the framework output, we have used supervised learning with '*binary_crossentropy*' as a loss function. To reproduce the results, we have specified all the parameters and their values in the Tables 3 and 4 (see Table 5).

**Table 1**
Number of operations and learnable weights for standard 1D convolution and separable 1D convolution operations on $1 \times H$ maps, considering all parameters as defined in text.

| Type of convolution | Number of operations | Number of weights |
|---|---|---|
| Standard Convolution | $k^2 \times C_{in} \times H \times C_{out}$ | $k^2 \times c_{in} \times c_{out}$ |
| Separable Convolution | $k^2 \times C_{in} \times H + c_{in} \times c_{out} \times H$ | $k^2 \times c_{in} + c_{in} \times c_{out}$ |
| Reduction factor | $1/k^2 + 1/c_{out}$ | $1/k^2 + 1/c_{out}$ |

**Table 2**
Correlation between the parameters needed for standard 1D convolution and DWS 1D convolution of the proposed framework.

| Convolution layer type | Trainable parameters | Non-trainable parameters | Total |
|---|---|---|---|
| Standard convolution | 214,402 | 640 | 215,042 |
| Separable convolution | 206,277 | 640 | 206917 |
| % of reduction | 3.79% | – | 3.77% |

## 6. Experimentation and results

The proposed OSV framework have been intensely assessed and confirmed by appraising experiments on three predominantly used datasets i.e. MCYT-100 dataset [36,39], SVC-Task 2 [46,47], SUSIG [50,58]. The outcomes are portrayed in Tables 9 and 10.

We have executed our experiments on Nvidia, GForce RTX 2080 4*12 GB GPU. An apprehensive explanation of each dataset with reference to the number of signatures used for training and testing of our OSV framework, the total number of genuine and forgeries available in the dataset is listed in Table 6. To evaluate our framework intensely, we have examined five categories of experimentation, i.e. Skilled_01 (S_01), Skilled_05 (S_05), Skilled_10 (S_10), Skilled_15(S_15) and Skilled_20 (S_20) in skilled forgery category and Random_01 (R_01), Random_05 (R_05), Random_10 (R_10), Random_15 (R_15) and Random_20 (R_20) in the random forgery category. Consider the dataset consists of 'W' number of users/writers, each user with 'G' genuine and 'F' forgery signature samples. In case of Skilled_N (S_N) category, where N = 01, 05, 10, . . ., (G-5), the framework is trained with 'N' genuine, 'N' forgery signature samples and tested with 'G-N' genuine samples to compute TAR and 'F-N' forgery samples to compute FAR. In case of Random_N (R_N) category, where N = 01, 05, 10,. . ., (G-5), the framework is trained with 'N' genuine and a set of 'W-1' randomly selected one genuine signature sample of each user as forgery sample and tested with '(G-N)' genuine samples and 'W-1' randomly

**Table 3**
Overview of the constituting building blocks of proposed framework.

| Layer | Size | Parameters |
|---|---|---|
| SeparableConvolution 1D + BatchNormalization + DropOut | 64x1x3 | activation='relu',kernel_initializer='random_uniform', bias_initializer='random_uniform', depthwise_initializer='random_uniform',pad='same',0.4 |
| SeparableConvolution + BatchNormalization + DropOut | 64 × 1 × 3 | ,activation='relu',kernel_initializer='random_uniform', bias_initializer='random_uniform', depthwise_initializer='random_uniform',pad='same',0.5 |
| Fully Connected + BatchNormalization + Dropout | 64 | bias_initializer='random_uniform',kernel_initializer='random_uniform', activation='relu',0.5 |
| Fully Connected + BatchNormalization + Dropout | 128 | bias_initializer='random_uniform',kernel_initializer='random_uniform', activation='relu',0.4 |

**Table 4**
Overview of the convolution AutoEncoder architecture used in our framework.

| Type of layer | Number of filters | Filter size | Activation function | Pooling size |
|---|---|---|---|---|
| 1D Convolution | 16 | 1 × 3 | Relu | N.A |
| 1D MaxPooling | N.A. | N.A | N.A | 2 |
| 1D Convolution | 16 | 1 × 3 | Relu | N.A |
| 1D MaxPooling | N.A. | N.A | N.A | 2 |
| 1D Convolution | 16 | 1 × 3 | Relu | N.A |
| 1D MaxPooling | N.A. | N.A | N.A | 2 |
| 1D Convolution | 16 | 1 × 3 | Relu | N.A |
| 1D UpSampling | N.A. | N.A | N.A | 2 |
| 1D Convolution | 16 | 1 × 3 | Relu | N.A |
| 1D UpSampling | N.A. | N.A | N.A | 2 |
| Dense | N.A. | N.A | Relu | N.A |

**Table 5**
List of various hyper parameters used in the proposed model.

| Parameter | Value |
|---|---|
| Optimizer | Adam |
| Learning rate (r) | 0.004 |
| beta_2 | 0.999 |
| beta_1 | 0.9 |
| Decay | 0.00 |
| epsilon | 1e−08 |
| Batch Size | 8 |

**Table 6**
The particulars of dataset used in the experimentation of the proposed framework.

| DataSet | MCYT-100 | SVC | SUSIG |
|---|---|---|---|
| Total number of writers | 100 | 40 | 94 |
| Set of features for each signature | 100 | 47 | 47 |
| Number of genuine signatures for each writer | 25 | 20 | 20 |
| Number of forgery signatures for each writer | 25 | 20 | 10 |
| Quantity of genuine signatures | 2500 | 800 | 1880 |
| Quantity of forgery signatures | 2500 | 800 | 940 |

selected one signature sample of each writer excluding the samples selected for training. We have adopted standard procedures as described in Refs. [9,23,29] to split the train and test signature samples for various categories and data sets. In line with the real time scenario, the OSV frameworks must satisfy the below requirements to prove to be efficient:

- **R1**: The increase in training signature samples must decrease the resultant EER value.
- **R2**: The random categories must achieve lesser EER values compared to corresponding skilled category.

Tables 9–11 demonstrates the comparison of EER values with the state of the art OSV frameworks, which are appraised on the similar datasets on which our proposed framework is evaluated. The (*) indicates the model resulted first best EER value and (**) indicates the model resulted in the second best EER value. In regard to the MCYT-100 (DB1) our proposed framework produces state-of-the-art results in Skilled 01, Skilled 10, Skilled 15, Random 10, Random 15 and Random 20 categories. With reference to SVC, the framework accomplishes the outstanding EER in all categories of experimentation excluding Random 01 and Random 05 categories. In regard to SUSIG, the framework delivers the finest EER in Random 01 and Random 10 categories. From the Tables 9–11, we summarize that the increase in training signature samples results in decreasing EER in all categories and in all the data sets. Any minor aberration of results not following the trend can be attributed to random initialization of kernel weights and biases. Also, the results summarize that, with the raise of training samples, SVC exhibits a steep decrease in EER value in both skilled and random categories followed by SUSIG and MCYT. As depicted in Tables 9–11, even though different frameworks proposed in Refs. [21,41,48,49,51,55] yields exceed EER values in comparison to the proposed framework, but, on the flip side, these models are not comprehensively adjudged with all practical categories of experimentation i.e Skilled 1, Skilled 2, Skilled 3, Skilled 4, Random 1, Random 2, Random 3, Random 4 etc. Producing higher classification accuracies with minimum number of training samples and for all possible categories of experimentation is the criterion for competence of the OSV frameworks to work in real time scenarios. Hence, we have extensively appraised our proposed model with all the possible training scenarios (1,2,3,4,5,10,15,20) in both skilled and random forgery categories and the performance is tabulated in Table 8. Table 8 contains the EER yielded by the proposed OSVFuseNet in various categories and different data sets for which there is no comparison data.

Fig. 10(a) and (b) illustrates the 2D-Histogram of EER resulted from applying the proposed approach on every user in SUSIG, skilled 5 and SVC, random 1 category. Fig. 10(a) portrays that the users from 70–90 contributes to higher EER compared to other users and the average EER varies between 2–5%. Similarly, Fig. 10 (b) illustrates that users from 25–35 contributes to higher EER and the average EER varies between 1.5–2.0%. Figs. 10(b) and 11 (b) illustrates the 2D-Histogram of EERs by the proposed approach on every user in skilled 5 and random 5 categories of MCYT dataset. Fig. 11(a) portrays that the users from 65–85 contributes to higher

**Table 7**
EER values with varying percentage of fusing handcrafted and deep features from CAE.

| Dataset | MCYT | | SUSIG | | SVC | |
|---|---|---|---|---|---|---|
| % of Handcrafted and deep features (CAE) | Skilled 1 EER | Random 1 EER | Skilled 1 EER | Random 1 EER | Skilled 1 EER | Random 1 EER |
| (20,80) | 19.23 | 7.18 | 18.34 | 4.16 | 6.16 | 5.16 |
| (30,70) | 15.10 | 6.93 | 18.27 | 2.68 | 6.96 | 4.45 |
| (40,60) | 14.01 | 6.46 | 18.20 | 2.23 | 6.78 | 4.19 |
| (50,50) | 13.93 | 5.82 | 18.13 | 1.91 | 6.12 | 4.56 |
| (60,40) | 13.38 | 5.05 | 17.96 | 1.87 | 5.95 | 3.61 |
| (70,30) | 13.97 | 5.46 | 18.68 | 3.13 | 7.28 | 3.71 |
| (80,20) | 14.98 | 6.12 | 19.08 | 3.13 | 6.98 | 3.18 |

**Table 8**
EER values with various signature samples for which comparison data is not available.

| Dataset | MCYT | | SUSIG | | SVC | |
|---|---|---|---|---|---|---|
| Number of signature samples | Skilled 1 EER | Random 1 EER | Skilled 1 EER | Random 1 EER | Skilled 1 EER | Random 1 EER |
| 02 | 12.63 | 1.13 | 10.31 | 5.19 | 2.61 | 5.56 |
| 03 | 10.01 | 0.93 | 0.93 | 3.76 | 1.59 | 3.45 |
| 04 | 6.8 | 0.46 | 1.00 | 3.13 | 0.89 | 4.19 |

**Table 9**
Comparative Analysis of EER of the proposed model against the recent models on MCYT (DB1) Database (where 'S' and 'R' represents Skilled and Random categories respectively. The number indicates the number of signature samples used for training). '–' indicates, the corresponding framework is not evaluated in that specific experimentation category.

| Method | S_01 | S_05 | S_10 | S_15 | S_20 | R_01 | R_05 | R_10 | R_15 | R_20 |
|---|---|---|---|---|---|---|---|---|---|---|
| Proposed Model: feature fusion + feature ranking + few shot learning | **13.38*** | 3.02 | **1.83*** | **1.25*** | 1.2 | **4.03**** | 0.42 | **0.1**** | **0.09**** | **0.08**** |
| Cancelable templates - HMM Protected [4] | – | 10.29 | – | – | – | – | – | – | – | – |
| Cancelable templates - HMM [4] | – | 13.30 | – | – | – | – | – | – | – | – |
| Probabilistic-DTW(case 1) [14] | – | – | – | – | 11.23 | – | **0.0118*** | – | – | – |
| Probabilistic-DTW(case 2) [14] | – | – | – | – | – | – | **0.0187**** | – | – | – |
| Writer dependent parameters (IntervalValued representation) [16] | – | 2.51 | – | – | – | – | 0.70 | – | – | **0.00*** |
| Common feature dimension and threshold (IntervalValued representation) [16] | – | 10.36 | – | – | **0.03**** | – | 10.32 | – | – | 0.74 |
| Writer dependent parameters (conventional) [16] | – | 6.79 | – | – | 5.82 | – | 1.73 | – | – | **0.00*** |
| Common feature dimension and threshold (conventional) [16] | – | 13.12 | – | – | **0.00*** | – | 5.61 | – | – | 1.66 |
| Combinational Features and Secure KNN-Global features [39] | – | 5.15 | – | – | – | – | 1.70 | – | – | – |
| Combinational Features and Secure KNN-Regional features [39] | – | 4.65 | – | – | – | – | 1.33 | – | – | – |
| Stability Modulated Dynamic Time Warping(F13) [39] | – | 13.56 | – | – | – | – | 4.31 | – | – | – |
| Dynamic Time Warping-Normalization(F13) [39] | – | 8.36 | – | – | – | – | 6.25 | – | – | – |
| Deep Learning + DTW (Tested with only 50 users) [36] | – | 2.40 | – | – | – | – | – | – | – | – |
| Angular Robotic features [57] | – | 3.44 | – | – | – | – | 0.75 | – | – | – |
| GMM + DTW with Fusion [11] | – | 3.05 | – | – | – | – | – | – | – | – |
| Stroke Based RNN [48] | 10.46 | – | – | – | – | – | – | – | | |
| Writer dependent parameters (Symbolic) [26] | – | 2.2 | – | – | 0.6 | – | 1.0 | – | – | 0.1 |
| VQ + DTW [29] | – | **1.55*** | – | – | – | – | – | – | – | – |
| Writer dependent features and classifiers [41] | – | 19.4 | – | – | 1.1 | – | 7.8 | – | – | 0.8 |
| Information Divergence-Based Matching [30] | – | 3.16 | – | – | – | – | – | – | – | – |
| Representation learning + DTW (Skilled forgery) [45] | | **1.62**** | – | – | – | – | 0.23 | – | – | – |
| Representation learning + DTW (Random forgery) [45] | | 1.81 | – | – | – | – | 0.24 | – | – | – |
| Stroke-Wise [46] | 13.72 | – | – | – | – | 5.04 | – | – | – | – |
| Target-Wise [46] | 13.56 | – | – | – | – | 4.04 | – | – | – | – |
| Curvature feature [47] | – | 10.22 | 8.25 | 6.38 | – | – | 4.12 | 3.33 | 2.58 | – |
| Torsion Feature [47] | – | 9.22 | 7.04 | 5.12 | – | – | 3.42 | 2.25 | 1.90 | – |
| Curvature feature + Torsion Feature [47] | – | 6.05 | **4.23**** | **3.10**** | – | – | 2.95 | 1.81 | 1.20 | – |
| WP + BL DTW [50] | – | 2.76 | – | – | – | – | – | – | – | – |
| Histogram + Manhattan [56] | – | 4.02 | – | – | – | – | 1.15 | – | – | – |
| Discriminative feature vector + several histograms [56] | – | 4.02 | – | – | 2.72 | – | 1.15 | – | – | 0.35 |
| VSA - DTW [58] | – | 3.24 | – | – | – | – | 0.80 | – | – | – |
| VSAr - DTW [58] | – | 2.68 | – | – | – | – | 0.75 | – | – | – |
| Fewshot learning [77] | **13.42**** | 7.03 | 5.70 | 3.95 | 2.20 | **2.00*** | 0.05 | **0.06*** | **0.01*** | **0.00*** |

The bold values are best EER values in each category.

EER of the framework compared to other users and the average EER varies between 2–5%. Similarly, Fig. 11(b) illustrates that users from 25–35 and 85–90 contributes to higher EER of the framework and the average EER varies between 0.25–0.5%.

We have implemented our OSV framework using both the normal convolutions and depth-wise separable convolutions for a comparative reason. Fig. 12 illustrates the divergence in EER with increasing number of training signature samples for skilled and random categories of MCYT dataset. It is evident that the depth-wise separable convolution results in lesser EER and converges faster compared to standard convolutions. Fig. 12 illustrates that depth-wise separable convolution based framework started reaching zero EER with 5 signature samples in both skilled and random categories. In case of normal convolutions, the framework requires a minimum of 15 signature samples to reach zero ERR. Based on the observations, we can confirm that DWS convolution outcomes lesser EER and faster convergence to zero EER in comparison to standard convolutions.

**Table 10**
Comparative analysis of EER of the proposed model against the recent models on SVC dataset. '–' indicates, the corresponding framework is not evaluated in that specific experimentation category. The bold values are best EER values in each category.

| Method | S_01 | S_05 | S_10 | S_15 | R_01 | R_05 | R_10 | R_15 |
|---|---|---|---|---|---|---|---|---|
| Proposed Model - feature fusion + feature ranking + few shot learning | **5.95**** | **3.93**** | 2.98 | **0.45**** | 3.61 | 0.39 | **0.13*** | **0.19**** |
| Stable feature and Partition [6] | – | – | 4.5 | – | – | – | – | |
| Probabilistic-DTW(case 1) [14] | – | – | – | – | – | **0.0025*** | – | – |
| Probabilistic-DTW(case 2) [14] | – | – | – | – | – | **0.0175**** | – | – |
| LCSS (User Threshold) [25] | – | – | 5.33 | – | – | – | – | – |
| Variance selection [31] | – | – | 13.75 | – | – | – | – | – |
| PCA [31] | – | – | 7.05 | – | – | – | – | – |
| Relief-1 (using the combined features set) [31] | – | – | 8.1 | – | – | – | – | – |
| Relief-2 [31] | – | – | 5.31 | – | – | – | – | – |
| Stroke Point Warping [37] | – | – | **1.00**** | – | – | – | – | – |
| SPW + mRMR + SVM(10-Samples) [37] | – | – | **1.00**** | – | – | – | – | – |
| RNN + LNPS[43] | – | – | – | – | – | 2.37 | – | – |
| Target-Wise [46] | 18.63 | – | – | – | **0.50*** | – | – | – |
| Stroke-Wise [46] | 18.25 | – | – | – | **1.90**** | – | – | – |
| Curvature feature + Torsion Feature [47] | – | 9.83 | 6.61 | 3.10 | – | 3.54 | 1.24 | 1.81 |
| DTW based (Common Threshold) [50] | – | – | 7.80 | – | – | – | – | – |
| Fewshot learning [77] | **5.83*** | **0.87*** | **0.35*** | **0.2*** | 9.08 | 1.4 | **0.15**** | **0.02*** |

The bold values are best EER values in each category.

**Table 11**
Comparative analysis of EER of the proposed model against the recent models on SUSIG dataset.'–' indicates, the corresponding framework is not evaluated in that specific experimentation category. The bold values are best EER values in each category.

| Method | S_01 | S_05 | S_10 | R_01 | R_05 | R_10 | Number of Samples for training |
|---|---|---|---|---|---|---|---|
| Proposed Model - (feature fusion + feature ranking + few shot learning) | 17.96 | 5.17 | 2.07 | **1.87**** | 1.53 | **0.3*** | 10 |
| Fewshot learning [77] | 10.41 | **0.8*** | **0.63**** | 8.7 | 2.5 | **1.26**** | 10 |
| cos angle, speed + enhanced DTW [15] | – | – | 3.06 | – | – | – | 10 |
| pole-zero models [78] | – | 2.09 | – | – | – | – | 5 |
| DCT and sparse representation [28] | – | – | **0.51*** | – | – | – | 10 |
| with all domain [79] | – | – | 3.88 | – | – | – | 10 |
| with stable domain [79] | – | – | 2.13 | – | – | – | 10 |
| Kinematic Theory of rapid human movements[20] | 7.87 | – | – | 3.61 | – | – | 01 |
| writer dependent features and classifiers [41] | – | – | 1.927 | – | – | – | 10 |
| Length Normalization + Fractional Distance [34] | – | – | 3.52 | – | – | – | 10 |
| Target-Wise [46] | **6.67*** | – | – | **1.55*** | – | – | 10 |
| Stroke-Wise [46] | **7.74**** | – | – | 2.23 | – | – | 10 |
| Information Divergence-Based Matching [30] | – | **1.6**** | 2.13 | – | – | – | 10 |
| Association of curvature feature with Hausdorff distance [47] | – | 7.05 | – | – | **1.02**** | – | 5 |
| VSA - DTW [58] | – | 3.83 | – | – | **0.78*** | – | 10 |
| VSAr - DTW [58] | – | 3.09 | – | – | **0.78*** | – | 5 |

The bold values are best EER values in each category.

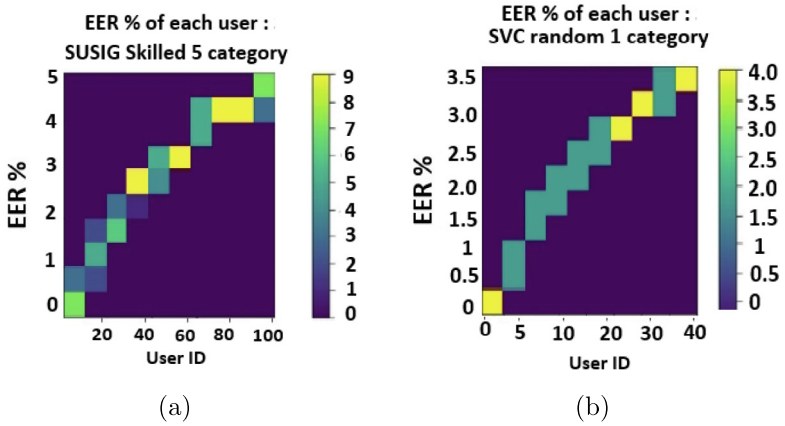


**Fig. 10.** a) The EERs of 94 users of SUSIG dataset obtained for Skilled 5. b) The EERs of 40 users of SVC dataset obtained for Random 1 categories.

### 6.1. Further experimental analysis and inferences

As OSV framework is a user/writer sensitive application, wherein both classification (True Acceptance Rate) and misclassification (False Acceptance Rate) performance are important to gauge the efficiency of the framework.

1. Taking into consideration SUSIG dataset, Fig. 13(a) portray that on an average the True Acceptance Rate of each user varies between 0.6 to 0.9 and False Acceptance Rate varies between 0.2 to 0.5.
2. Fig. 13(b) elucidates that on an average the True Acceptance Rate of each user varies between 0.4 to 0.8 and False Acceptance

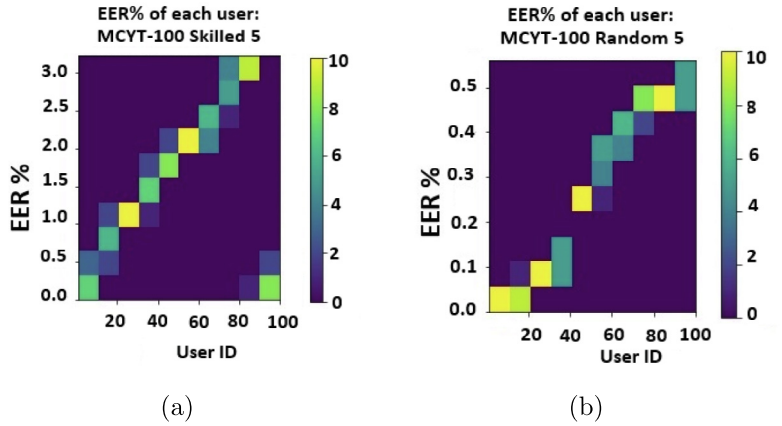


**Fig. 11.** a) The EERs of 100 users of MCYT dataset obtained for Skilled 5. b) The EERs of 100 users of MCYT dataset obtained for Random 5 categories.

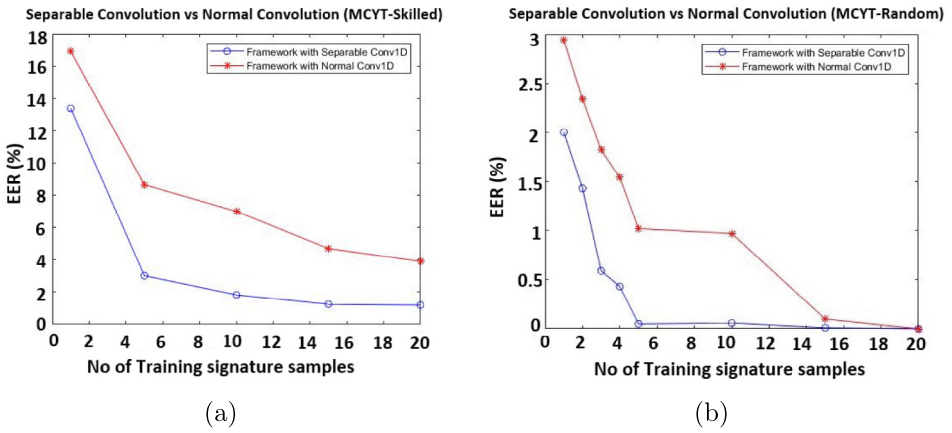


**Fig. 12.** The EER outcome by the OSV framework for MCYT-Dataset with varying signature samples (a) separable convolutions and (b) normal convolutions.

Rate varies between 0.0 to 0.2 with few outliers. The proposed framework accomplishes state of the art results in skilled 10, random 05 and 10 categories.

3. Corresponding to MCYT dataset, 14(a) illustrates that, even though the model is trained with single genuine and forgery samples, for the majority of users, the framework attained near 1 TAR. The average FAR values vary from 0.6 to 0.8.
4. Fig. 14(b) illustrates that, with increasing the training samples from 1 to 20, for 95 users, the framework delivers 100% TAR. The proposed framework accomplishes state of the art results in skilled 1, skilled 10, skilled 15, random 10, random 15 and random 20 categories.
5. Fig. With reference to 15(a) illustrates that, even though the model is trained with only one sample of genuine and forgery signature per user, except for users 12, 31, 35, the TAR varies between 0.6 to 0.8.
6. Fig. 15(b), illustrates that, the framework achieves TAR varies between 0.55 to 0.8 and recorded excellent performance in case of FAR, wherein the average values varies between 0 to 0.1. The

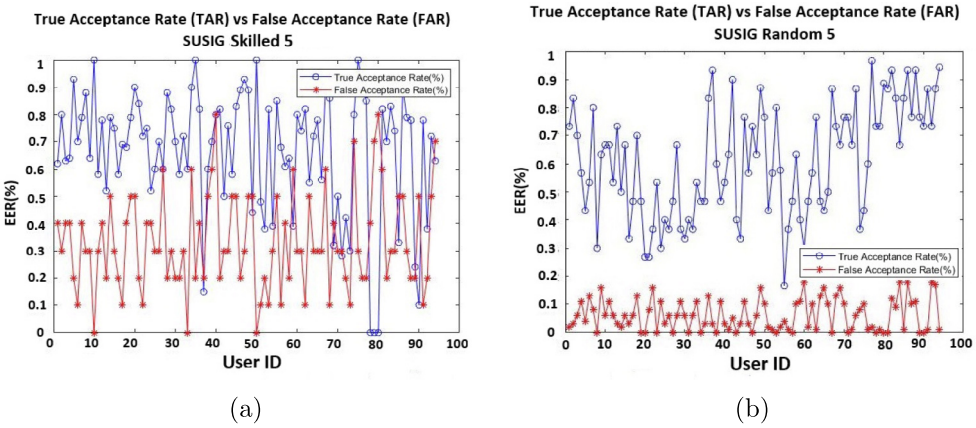


**Fig. 13.** The ROC curves resulted on SUSIG dataset. (a) The TAR and FAR values for 94 users with reference to Skilled 5 category. (b) The TAR and FAR values for 94 users with reference to Random 5 category.

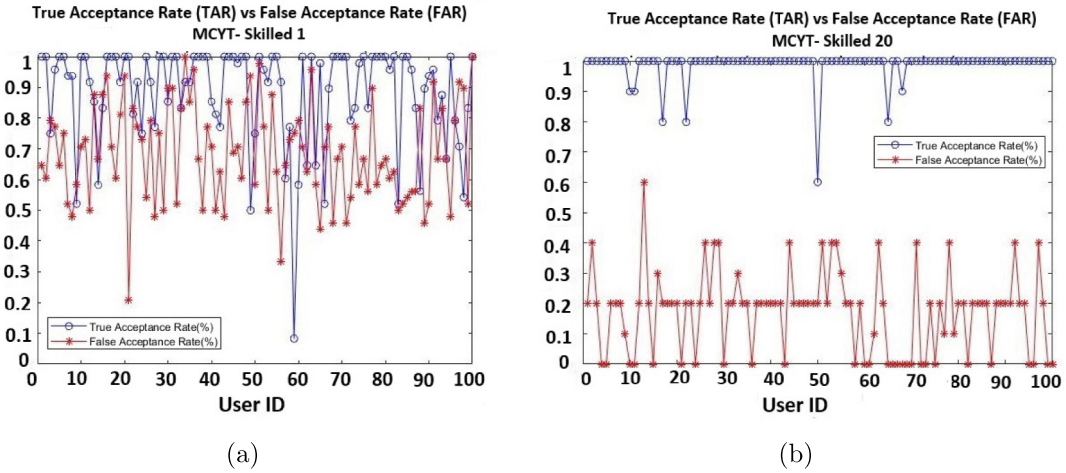


**Fig. 14.** The ROC curves resulted on MCYT-100 (DB1) database. (a) The TAR and FAR for 100 users with reference to Skilled 1 category. (b) The TAR and FAR for 100 users with reference to Skilled 20.

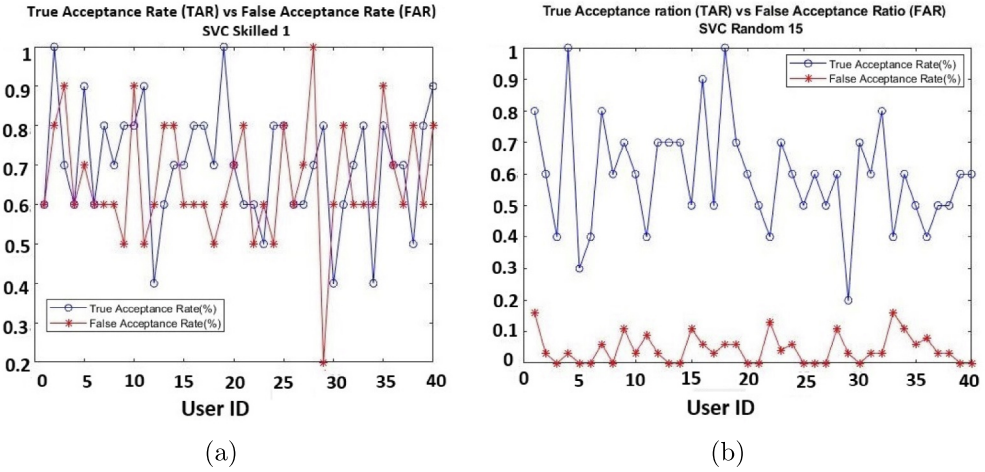


**Fig. 15.** The ROC curves resulted on SVC dataset. (a) The TAR and FAR for 40 users with reference to Skilled 1 category. (b) The TAR and FAR for 40 users with reference to Random 15 category.

proposed OSV framework surpass the state-of the-art outcomes in skilled 1, skilled 5, skilled 10, skilled 15, random 10 and random 15 categories.

7. Based on these, we can infer that, with lesser number of training signature samples, there is an overlapping of TAR and FAR values and the gap are widened with the increase of training samples. The framework is able to learn good representational features with minimum number of training samples and can achieve good TAR and FAR values, which in turn results in the state-of-the-art performance of the proposed framework.

## 7. Conclusion and future work

In this manuscript, we have introduced 'OSV FuseNet', a framework based on Depth-wise separable convolutional neural networks for online signature verification, which uses a feature ranking and feature level fusion of hand-made and deep representative features. The fusion eliminates the shortcomings of the corresponding group of features by complementing one another. The Depth-wise separable convolution operations results in a significant drop in the number of operations and parameters required by the network in learning the discriminating features. This results in the framework to learn the discriminating features from as minimum as one training signature sample and outcomes higher classification accuracies. Experiments are conducted on all possible categories of experimentation using three widely used online signature datasets. To the extent of our knowledge, this is the first work to consider hybrid fusion of features and few shot learning. We have evaluated the model thoroughly with all possible categories of experimentation like skilled 1, 2, 3, 4, 5, 10, 15, 20 and random 1, 2, 3, 4, 5, 10, 15, 20. Furthermore, the OSVFuseNet designed here has outperformed the state-of-the-art outcome on most of the standard signature datasets, which is encouraging for further research in this direction. The lightweights of the framework, ability to yield higher classification results even with one training signature sample, makes the proposed OSVFuseNet suits for real time deployment.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## CRediT authorship contribution statement

**Chandra Sekhar Vorugunti:** Conceptualization, Writing - original draft. **Viswanath Pulabaigiri:** Validation, Supervision. **Rama**

**Krishna Sai Subrahmanyam Gorthi:** Validation, Writing - review & editing. **Prerana Mukherjee:** Writing - review & editing.

## Acknowledgements

The work reported in this paper is partly funded by a DIC funded project with reference DIC Proj14 PV. Partly from IIIT Sri-City seed grant on writer recognition.

## References

[1] J. Fierrez-Aguilar, S. Krawczyk, J. Ortega-Garcia, A.K. Jain, Fusion of local and regional approaches for on-line signature verification, in: International Workshop on Biometric Person Authentication, Springer, 2005, pp. 188–196.
[2] L. Nanni, Experimental comparison of one-class classifiers for online signature verification, Neurocomputing 69 (7–9) (2006) 869–873.
[3] L. Nanni, A. Lumini, Advanced methods for two-class problem formulation for on-line signature verification, Neurocomputing 69 (7–9) (2006) 854–857.
[4] E. Maiorana, P. Campisi, J. Fierrez, J. Ortega-Garcia, A. Neri, Cancelable templates for sequence-based biometrics with application to on-line signature recognition, IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans 40 (3) (2010) 525–538.
[5] L. Songxuan, J. Lianwen, Recurrent adaptation networks for online signature verification, IEEE Transactions On Information Forensics and Security (2019) 1624–1637.
[6] D. Impedovo, P. Giuseppe, Automatic signature verification in the mobile cloud scenario:survey and way ahead, IEEE Transactions on Emerging Topics in Computing (2018) 41–54.
[7] K. Riesen, S. Roman, Online signature verification based on string edit distance, International Journal on Document Analysis and Recognition 22 (2019) 41–54.
[8] M. Zalasinski, C. Krzysztof, A new method for signature verification based on selection of the most important partitions of the dynamic signature, Neurocomputing 289 (2018) 13–22.
[9] D. Guru, H. Prakash, Symbolic representation of on-line signatures, in: International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007), vol. 2, IEEE, 2007, pp. 312–317..
[10] J. Galbally, J. Fierrez, J. Ortega-Garcia, Classification of handwritten signatures based on name legibility, in: Biometric Technology for Human Identification IV, vol. 6539, International Society for Optics and Photonics, 2007, pp. 653–660..
[11] J. Galbally, J. Fierrez, M. Martinez-Diaz, J. Ortega-Garcia, Improving the enrollment in dynamic signature verification with synthetic samples, in: 2009 10th International Conference on Document Analysis and Recognition, IEEE, 2009, pp. 1295–1299.
[12] S. Rashidi, A. Fallah, F. Towhidkhah, Authentication based on pole-zero models of signature velocity, Journal of Medical Signals and Sensors 3 (4) (2013) 195–208.
[13] A. Sharma, S. Sundaram, A novel online signature verification system based on gmm features in a dtw framework, IEEE Transactions on Information Forensics and Security 12 (3) (2016) 705–718.
[14] R. Al-Hmouz, W. Pedrycz, K. Daqrouq, A. Morfeq, A. Al-Hmouz, Quantifying dynamic time warping distance using probabilistic model in verification of dynamic signatures, Soft Computing 23 (2) (2019) 407–418.
[15] M.I. Khalil, M. Moustafa, H.M. Abbas, Enhanced dtw based on-line signature verification, in: 2009 16th IEEE International Conference on Image Processing (ICIP), IEEE, 2009, pp. 2713–2716.
[16] C.S. Vorugunti, D. Guru, P. Viswanath, An efficient online signature verification based on feature fusion and interval valued representation of writer dependent features, in: IEEE fifth International Conference on Identity, Security and Behavior Analysis (ISBA), IEEE, 2019, pp. 157–162.
[17] M.Z. Yanan Sun, G.G. Bing Xue, Yen Evolving deep convolutional neural networks for image classification, IEEE Transactions on Evolutionary Computation (2019) 1–14.
[18] Z. Zhong-Qiu, Z. Peng, X. Shou-Tao, W. Xindong, Object detection with deep learning: a review, IEEE Transactions on Neural Networks and Learning Systems (2019) 1–21.
[19] L. Minghui, Z. Jian, W. Zhaoyi, X. Fengming, L. Jiajun, L. Pengyuan, Y. Cong, X. Bai, Scene text recognition from two-dimensional perspective, in: Thirty-Third AAAI Conference on Artificial Intelligence, vol. 33, IEEE, 2019, pp. 41–54..
[20] L. Chenxi, C. Liang-Chieh, S. Florian, A. Hartwig, H. Wei, Y. Alan, F.F. Li, Auto-deeplab:hierarchical neural architecture search for semantic image segmentation, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2019, vol. 33, IEEE, 2019, pp. 82–92..
[21] Y. Daksha, K. Naman, A. Akshay, V. Mayank, S. Richa, N. Afzel, Fusion of handcrafted and deep learning features for large-scale multiple iris presentation attack detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2018, pp. 685–692.
[22] C. Francois, Xception: deep learning with depthwise separable convolutions, in: 2017 IEEE International Conference on Computer Vision Pattern Recognition (CVPR), IEEE, 2017, pp. 1251–1258.
[23] D. Guru, H. Prakash, Online signature verification and recognition: an approach based on symbolic representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (6) (2008) 1059–1073.
[24] D. Guru, K. Manjunatha, S. Manjunath, User dependent features in online signature verification, in: Multimedia Processing, Communication and Computing Applications, Springer, 2013, pp. 229–240.
[25] K. Barkoula, G. Economou, S. Fotopoulos, Online signature verification based on signatures turning angle representation using longest common subsequence matching, International Journal on Document Analysis and Recognition (IJDAR) 16 (3) (2013) 261–272.
[26] A.Q. Ansari, M. Hanmandlu, J. Kour, A.K. Singh, Online signature verification using segment-level fuzzy modelling, IET Biometrics 3 (3) (2013) 113–127.
[27] I. Sutskever, O. Vinyals, Q.V. Le, Sequence to sequence learning with neural networks, Advances in Neural Information Processing Systems (2014) 3104–3112.
[28] Y. Liu, Z. Yang, L. Yang, Online signature verification based on dct and sparse representation, IEEE Transactions on Cybernetics 45 (11) (2014) 2498–2511.
[29] A. Sharma, S. Sundaram, An enhanced contextual dtw based system for online signature verification using vector quantization, Pattern Recognition Letters 84 (2016) 22–28.
[30] L. Tang, W. Kang, Y. Fang, Information divergence-based matching strategy for online signature verification, IEEE Transactions on Information Forensics and Security 13 (4) (2017) 861–873.
[31] L. Yang, Y. Cheng, X. Wang, Q. Liu, Online handwritten signature verification using feature weighting algorithm relief, Soft Computing 22 (23) (2018) 7811–7823.
[32] P. Deepak, C. Renato, L. Peter, Feature weighting as a tool for unsupervised feature selection, Information Processing Letters 129 (2018) 44–52.
[33] D. Chakraborty, V. Narayanan, A. Ghosh, Integration of deep feature extraction and ensemble learning for outlier detection, Pattern Recognition 89 (2019) 161–171.
[34] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, J. Ortega-Garcia, Biometric signature verification using recurrent neural networks, in: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), vol. 1, IEEE, 2017, pp. 652–657..
[35] A. Alaei, S. Pal, U. Pal, M. Blumenstein, An efficient signature verification method based on an interval symbolic representation and a fuzzy similarity measure, IEEE Transactions on Information Forensics and Security 12 (10) (2017) 2360–2372.
[36] W. Xiaomeng, K. Akisato, I. Brian, U. Kenji, K. Kunio Seiichi, Deep dynamic time warping:end-to-end local representation learning for online signature verification, in: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), IEEE, 2019, pp. 1103–1110.
[37] B. Kar, A. Mukherjee, P. Dutta, Stroke point warping-based reference selection and verification of online signature, IEEE Transactions on Instrumentation and Measurement 67 (2018) 2–11.
[38] G. Pirlo, V. Cuccovillo, M. Diaz-Cabrera, D. Impedovo, P. Mignone, Multidomain verification of dynamic signatures using local stability analysis, IEEE Transactions on Human-Machine Systems 45 (6) (2015) 805–810.
[39] R. Doroz, P. Kudlacik, P. Porwik, Online signature verification modeled by stability oriented reference signatures, Information Sciences 460 (2018) 151–171.
[40] L. Yang, X. Jin, Q. Jiang, Online handwritten signature verification based on the most stable feature and partition, Cluster Computing 22 (2019) 1691–1701.
[41] K. Manjunatha, S. Manjunath, D. Guru, M. Somashekara, Online signature verification based on writer dependent features and classifiers, Pattern Recognition Letters 80 (2016) 129–136.
[42] N. Kalchbrenner, L. Espeholt, K. Simonyan, A. Oord, A. Graves, K. Kavukcuoglu, Neural machine translation in linear time, arXiv preprint arXiv:1610.10099..
[43] S. Lai, L. Jin, W. Yang, Online signature verification using recurrent neural network and length-normalized path signature descriptor, in: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), vol. 1, IEEE, 2017, pp. 400–405..
[44] F. Chollet, Xception: deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258.
[45] S. Lai, L. Jin, Recurrent adaptation networks for online signature verification, IEEE Transactions on Information Forensics and Security 14 (6) (2019) 1624–1637.
[46] M. Diaz, A. Fischer, M.A. Ferrer, R. Plamondon, Dynamic signature verification system based on one real signature, IEEE Transactions on Cybernetics 48 (1) (2016) 228–239.
[47] L. He, H. Tan, Z. Huang, Online handwritten signature verification based on association of curvature and torsion feature with hausdorff distance, Multimedia Tools and Applications (2019) 1–26.
[48] L. Chuang, Z. Xing, L. Feng, W. Zhiyong, L. Jun'E, Z. Rui, A stroke-based rnn for writer-independent online signature verification, in: International Conference on Document Analysis and Recognition (ICDAR), IEEE, 2019, pp. 526–532.
[49] L. Bai, Y. Zhao, X. Huang, A cnn accelerator on fpga using depthwise separable convolution, IEEE Transactions on Circuits and Systems II: Express Briefs 65 (10) (2018) 1415–1419.
[50] A. Sharma, S. Sundaram, On the exploration of information from the dtw cost matrix for online signature verification, IEEE Transactions on Cybernetics 48 (2) (2017) 611–624.
[51] M. Berkay Yilmaz, K. Ozturk, Hybrid user-independent and user-dependent offline signature verification with a two-channel cnn, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018, pp. 526–534.

[52] H. Kuang, L. Chen, L. Chan, R. Cheung, H. Yan, Feature selection based on tensor decomposition and object proposal for night-time multiclass vehicle detection, IEEE Transactions on Systems, Man, and Cybernetics: Systems 49 (1) (2019) 71–80.
[53] D. Alexandre, C. Chang, W. Peng, H. Hang, An autoencoder-based learned image compressor: Description of challenge proposal by nctu, in: CVPR Workshops, 2018, pp. 2539–2542..
[54] J. Gehring, M. Auli, D. Grangier, D. Yarats, Y.N. Dauphin, Convolutional sequence to sequence learning, in: Proceedings of the 34th International Conference on Machine Learning, vol. 70, JMLR.org, 2017, pp. 1243–1252..
[55] A. Parziale, D. Moises, A. Miguel, M. Angelo, Sm-dtw: stability modulated dynamic time warping for signature verification, Pattern Recognition Letters 121 (2019) 113–122.
[56] N. Sae-Bae, N. Memon, Online signature verification on mobile devices, IEEE Transactions on Information Forensics and Security 9 (6) (2014) 933–947.
[57] M. Diaz, M. Ferrer, J. Quintana, Robotic arm motion for verifying signatures, in: 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), IEEE, 2018, pp. 157–162.
[58] M. Diaz, M. Ferrer, J. Quintana, Anthropomorphic features for on-line signatures, IEEE Transactions on Pattern Analysis and Machine Intelligence 41 (2019) 2807–2819.
[59] H. Chaoqun, Y. Jun, Z. Jian, J. Xiongnan, L. Kyong-Ho, Multimodal face-pose estimation with multitask manifold deep learning, IEEE Transactions on Industrial Informatics 15 (2019) 3952–3961.
[60] O. Jun, L. Yujian, Vector-kernel convolutional neural networks, Neurocomputing 330 (2019) 253–258.
[61] A. Shyam Prasad, Y. Changju, S. Krzysztof, S. Michal, K. Hyongsuk, Hybrid no-propagation learning for multilayer neural networks, Neurocomputing 321 (2018) 28–35.
[62] H. Chaoqun, Y. Jun, W. Jian, T. Dacheng, W. Meng, Multimodal deep autoencoder for human pose recovery, IEEE Transactions on Image Processing 24 (2015) 5659–5669.
[63] H. Chaoqun, Y. Jun, T. Dacheng, W. Meng, Image-based 3d human pose recovery by multi-view locality sensitive sparse retrieval, IEEE Transactions on Industrial Electronics 62 (2015) 3742–3751.
[64] M. Diaz, M.A. Ferrer, D. Impedovo, M.I. Malik, G. Pirlo, R. Plamondon, A perspective analysis of handwritten signature technology, ACM Computing Surveys (CSUR) 51 (117) (2019) 1–39.
[65] M. Qinxue, C. Daniel, S. David, P. Kennedy, Relational autoencoder for feature extraction, in: 2017 International Joint Conference on Neural Networks (IJCNN), IEEE, 2017, pp. 364–371.
[66] R.H.X.Y.C.L.A.S. Yunchen Pu, Zhe Gan, L. Carin, Variational autoencoder for deep learning of images, in: Neural Information Processing Systems 29 (NIPS 2016), IEEE, 2016, pp. 41–54..
[67] M. Nauman, P. Jinhyun, K. Hak-Joon, S. Sung-Jin, K. Sung-Sik, Performance enhancement of convolutional neural network for ultrasonic flaw classification by adopting autoencoder, NDT & E International 111 (2020) 1–30.
[68] L. Xinchen, T. Yang, T. Huaglory, Q. Feng, Z. Weimin, A novel approach to reconstruction based saliency detection via convolutional neural network stacked with auto-encoder, Neurocomputing 349 (2019) 145–155.
[69] L. Wei, L. Jun, Y. Jian, X. Wei, Z. Jian, Convolutional sparse autoencoders for image classification, IEEE Transactions on Neural Networks and Learning Systems 29 (2018) 3289–3294.
[70] A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, Mobilenets: Efficient convolutional neural networks for mobile vision applications, arXiv preprint arXiv:1704.04861..
[71] L. Kaiser, A.N. Gomez, F. Chollet, Depthwise separable convolutions for neural machine translation, arXiv preprint arXiv:1706.03059..
[72] C. Gong, Y. Ceyuan, Y. Xiwen, G. Lei, J. Han, When deep learning meets metric learning: remote sensing image scene classification via learning discriminative cnns, IEEE Transactions on Geoscience and Remote Sensing 56 (2018) 2811–2821.
[73] N. Loris, G. Stefano, B. Sheryl, Handcrafted vs. non-handcrafted features for computer vision classification, Pattern Recognition 71 (2017) 158–172.
[74] R. Su, L. Tianling, S. Changming, J. Qiangguo, J. Rachid, W. Leyi, Fusing convolutional neural network features with hand-crafted features for osteoporosis diagnoses, Neurocomputing 385 (2020) 300–309.
[75] L. Nanni, G. Stefano, B. Sheryl, Handcrafted vs. non-handcrafted features for computer vision classification, Pattern Recognition 71 (2017) 158–172.
[76] L. Bai, Y. Zhao, X. Huang, A cnn accelerator on fpga using depthwise separable convolution, IEEE Transactions on Circuits and Systems II: Express Briefs 65 (10) (2018) 1415–1419.
[77] C.S. Vorugunti, S.G. Rama Krishna, P. Viswanath, Online signature verification by few-shot separable convolution based deep learning, in: 15th International Conference on Document Analysis and Recognition (ICDAR 2019), IEEE, 2019, pp. 1125–1131.
[78] S. Rashidi, A. Fallah, F. Towhidkhah, Authentication based on pole-zero models of signature velocity, Journal of Medical Signals and Sensors 3 (4) (2013) 195.
[79] G. Pirlo, V. Cuccovillo, M. Diaz-Cabrera, D. Impedovo, P. Mignone, Multidomain verification of dynamic signatures using local stability analysis, IEEE Transactions on Human-Machine Systems 45 (6) (2015) 805–810.

**Chandra Sekhar Vorugunti** received his B.Tech. degree in Computer Science and Engineering from JNTU-HYD, Masters from Dhirubhai Ambani Institute of Information and Communication Technology-Gujarat. Since August 2016, he is a full time PhD student in Computer Science and Engineering department of Indian Institute of Information Technology, Sri City, Andhra Pradesh. His area of research interests includes pattern recognition, computer vision and deep learning.

**Viswanath Pulabaigari** has received his Ph.D. from Indian Institute of Science (IISc) Bangalore. He has obtained B.Tech from Sri Venkateswara University, Tirupati and M.Tech. from Indian Institute of Technology (IIT) Madras. Currently, he is working as an Associate Professor in the Computer Vision and Machine Learning group at Indian Institute of Information Technology SriCity (IIIT SriCity). He has written several peer-reviewed research papers (in reputed journals and conferences). His research area includes Machine Learning, Computer Vision, Pattern Recognition, and Data Mining.

**Dr. Rama Krishna Sai Gorthi** is currently working as an Associate Professor in IIT Tirupati in the Department of Electrical Engineering, since 2017. Prior to this, he was Associate Professor at Indian Institute of Space Science and Technology, Trivandrum, since 2011. He was a postdoctoral fellow at INRIA, France from 2009 to 2011. He was doctorate from IIT Madras, INDIA and Masters (in Research) from IISc, Bangalore, INDIA both in Computer Vision. His teaching experience spans over Signal Processing, Image Processing, Computer Vision, Machine Learning and Deep learning. He has expertise in the development and application of various image processing methods specifically for image restoration, inpainting and computer vision methods such as visual tracking, 3D reconstruction from active and passive stereo, and machine learning and deep learning approaches for many real applications like blood cell classification, satellite image analysis and segmentation, face recognition and character recognition. He has around 75 publications in highly reputed International Journals and Conferences.

**Prerana Mukherjee** received her B.Tech. degree in computer science from the Guru Gobind Singh Indraprastha University, Delhi, and M.Tech. from Delhi Technological University, Delhi. She completed her Ph. D. degree from Indian Institute of Technology, Delhi. Since November 2018, she has been with the faculty of the Department of Computer Science and Engineering, Indian Institute of Information Technology, Sri City, Andhra Pradesh, where she is currently an Assistant Professor. Her research interests include image processing, computer vision, and pattern recognition.