International Journal on Document Analysis and Recognition (IJDAR) (2018) 21:187–198
https://doi.org/10.1007/s10032-018-0308-z

SPECIAL ISSUE PAPER

Integrating scattering feature maps with convolutional neural networks for Malayalam handwritten character recognition

K. Manjusha[1] · M. Anand Kumar[1] · K. P. Soman[1]

Received: 30 September 2017 / Revised: 25 May 2018 / Accepted: 19 June 2018 / Published online: 2 July 2018

Abstract
Convolutional neural network (CNN)-based deep learning architectures are the state-of-the-art in image-based pattern recognition applications. The receptive filter fields in convolutional layers are learned from training data patterns automatically during classifier learning. There are number of well-defined, well-studied and proven filters in the literature that can extract informative content from the input patterns. This paper focuses on utilizing scattering transform-based wavelet filters as the first-layer convolutional filters in CNN architecture. The scattering networks are generated by a series of scattering transform operations. The scattering coefficients generated in first few layers are effective in capturing the dominant energy contained in the input data patterns. The present work aims at replacing the first-layer convolutional feature maps in CNN architecture with scattering feature maps. This architecture is equivalent to utilizing scattering wavelet filters as the first-layer receptive fields in CNN architecture. The proposed hybrid CNN architecture experiments the Malayalam handwritten character recognition which is one of the challenging multi-class classification problems. The initial studies confirm that the proposed hybrid CNN architecture based on scattering feature maps could perform better than the equivalent self-learning architecture of CNN on handwritten character recognition problems.

Keywords Handwritten recognition · Convolutional neural networks · Deep learning models · Scattering convolutional network · Malayalam character recognition · Scattering transform

1 Introduction

Handwritten character recognition is one of some popular and well-studied research areas under pattern recognition. Still the handwriting text recognition is a challenging machine learning problem due to the varying writing styles of human beings. The feature descriptor extracted from data patterns greatly affects the performance of the underlying recognition system [1]. Despite the variabilities, recognition system should be capable of generating invariant features for achieving stable recognition performance. Initial works on handwritten character recognition focused on transformation and statistical techniques for extracting the informative feature representation [2]. Deep learning architectures based on convolutional neural networks (CNN) have got attention during past few years and are the state-of-the-art method in most of the handwritten text recognition applications [3,4]. Self-learned features in CNN perform better than other handcrafted feature descriptors in most of the image-related pattern recognition applications [5–7]. CNN is designed in such a way that it can deal with the variabilities in two-dimensional patterns present in images. CNN-based architectures have set a benchmark recognition performance in MNIST, one of the most widely used handwritten digit database [5,8–10]. The receptive fields or filters banks present in convolutional layers in CNN architectures are capable of generating invariant feature representations from image patterns with the support of subsampling operations. Initialization values for convolutional filter fields considerably affect the model convergence, and thus the recognition performance of CNN classifier [6]. The self-learned filters initialized with pre-trained model

✉ K. Manjusha
k_manjusha@cb.amrita.edu

M. Anand Kumar
m_anandkumar@cb.amrita.edu

K. P. Soman
kp_soman@amrita.edu

1 Center for Computational Engineering and Networking (CEN), Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India

parameters could achieve better recognition values and faster convergence.

In the image recognition literature, there are well-defined, well-studied and predefined filters that are guaranteed to produce benchmark performances. Gradient detectors, Gabor filters, wavelets are some of the traditional well-defined filters and have proven their effectiveness in most of the pattern recognition applications. The first-layer filters in CNN architecture mostly resemble orientation bar detectors and Gabor filters [12]. Deep networks sometimes may not be able to learn vital domain-specific features. In [13], for online Chinese handwritten character recognition, domain-specific information is investigated and integrated with deep CNN to achieve improved recognition performance. The directional feature maps integrated with GoogLeNet architecture, for improving offline character recognition performance of Chinese script [14]. Another work in the same area of handwritten Chinese character recognition incorporated direction-decomposed feature maps with deep CNN and could obtain state-of-the-art recognition performance in both offline and online ICDAR-2013 competition database [15]. Unsupervised Fisher vector and supervised fully connected layers are utilized to form a hybrid classification architecture [16]. Besides this, learning free data-independent DCTNet [17] is inspired from the hybrid architecture PCANet [18]. Apart from image recognition applications, hybrid architectures are experimented in speech recognition domain by incorporating Gabor functions into CNN filter fields, giving rise to Gabor convolutional neural network [19]. For image super-resolution, the sparse coding model is combined with the CNN and could obtain improved results [20]. All these hybrid architectures, which make use of both predefined filters and self-learning filters in deep neural networks, describe the advantages of using domain-specific information in CNN classifier for further improving the recognition performance. Scattering network is a recently evolved, predefined scattering wavelet filter bank that has achieved comparable performance as that of deep learning architectures [21]. Scattering transform is employed in different pattern recognition applications [22,23]. Instead of the first-layer self-learning filters in CNN architecture, utilizing the above-mentioned predefined filters can help to achieve better feature representation from the input images.

In this paper, the convolutional feature maps in the first layer of CNN classifier are replaced by scattering feature maps extracted from scattering network for Malayalam handwritten character recognition. Malayalam is one of the classical languages in India and is the official language of the state Kerala in India [24]. Compared to other Indian language scripts, the high similarity among character shapes, a large number of character classes and the usage of old as well new Malayalam script in writing make Malayalam character recognition difficult [25]. The history of document recognition research in Malayalam language script dates back to the 1950s.

Even though so many isolated works have been reported in recognition of Malayalam characters [26–30], the similarly shaped characters still lead to misclassification. Grouping of character classes may improve the recognition accuracy in case of similarly shaped character classes [31]. Robust invariant feature descriptors or multistage classification techniques can solve this problem to a certain extent. The proposed CNN architecture is evaluated in the context of Malayalam handwritten character recognition.

Section 2 discusses scattering representations and CNN. The proposed architecture is described in Sect. 2.3. The Malayalam handwritten character database utilized for conducting experiments is described in Sect. 2.4. Finally, Sect. 3 discusses the various experiments conducted toward implementing the proposed architecture.

2 Materials and methods

This section describes the algorithms, methodologies and Malayalam character image database employed to implement the recognition system.

2.1 Scattering network

Scattering convolutional network employs the scattering transform which can generate stable invariant feature representation. Scattering transform computes invariant image representation with the help of wavelet decomposition, modulus and averaging operations. Scattering network produces invariant image descriptors through sequential wavelet decomposition over multiple layers. Instead of trainable filter banks in CNN, scattering network makes use of predefined complex directional wavelet filters over spatial and angular filters to generate convolutional feature maps.

Let r belong to the rotation group of $\mathbb{R}^2$; then, the 2D directional wavelets with $\lambda = 2^j r$ can be defined as shown in Eq. 1, where $j < J$. J represents the spatial scale of wavelet function.

$$\psi_\lambda(u) = 2^{-2j}\psi(2^{-j}r^{-1}u). \tag{1}$$

The wavelet decomposition coefficients $\{x * \psi_\lambda\}$ are given to modulus and averaging operators to produce translation-invariant feature descriptors. The resultant invariant feature representation or the scattering transform coefficients are mathematically represented as shown in Eq. 2.

$$\|x * \psi_\lambda\|_1 = \int |x * \psi_\lambda(u)|\mathrm{d}u. \tag{2}$$

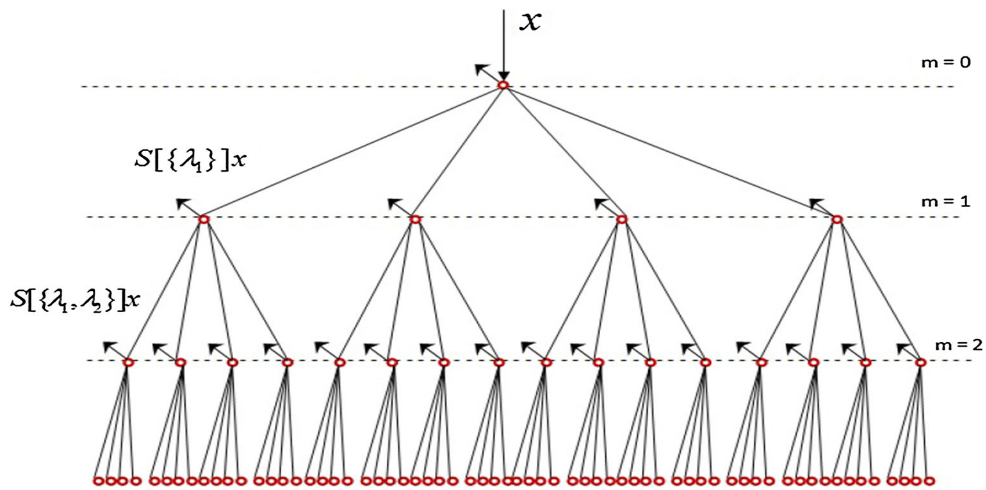

Fig. 1 Scattering convolution network [11]

In classification tasks, localized invariants are computed through windowed scattering transform. Low-pass averaging at scale 2^J using $\phi_{2^J}(u) = 2^{-2J}\phi(2^{-J}u)$ function can produce localized invariant feature descriptors. The information loss happening through the averaging function can be recovered by applying wavelet transform again on $|x * \psi_\lambda|$ with ψ_{λ_i}. A large set of translation invariants are generated by repeating wavelet decomposition operation iteratively. A scattering convolutional network is formed by computing scattering coefficients through all possible combinations of space and angular variables and iterating the same operation on the modulus of wavelet coefficients again. Figure 1 shows the visualization of scattering convolutional network. The generalized equation for scattering coefficient S calculated through the path $p = \{\lambda_1, \lambda_2 \ldots \lambda_m\}$ is in Eq. 3.

$$S[p]x = \left|\left|\left|\left|x * \psi_{\lambda_1}\right| * \psi_{\lambda_2}\right| \ldots\right| * \psi_{\lambda_m}\right| * \phi_{2^J}. \tag{3}$$

Each node in the scattering network produces scattering feature maps. The root node is $S[\emptyset]x = x * \phi_{2^J}$. $S[p]x$ is of order m and is computed at mth layer of the network. The scattering feature maps are equivalent to subsampled convolutional feature maps in CNN. But scattering feature maps make use of predefined wavelet filters instead of self-learned filters in CNN.

The mathematical properties of scattering transform explained in [32] proves the effectiveness of the feature representation. Fourier-transform modulus is also capable of producing invariants, but the Fourier power spectrum depends only on the second-order moments. Scattering transform provides information on higher-order moments, which gives more discrimination capability for scattering representations among textures having same power spectrum [21,32]. Scattering operators avoids high-frequency instabilities and preserves the Lipschitz stability of wavelets to the effect of deformations [32]. Stability to deformations is an important concern in most of the recognition applications. If appropriate wavelets are employed for scattering transform, then it preserves the signal norm and the energy of scattering coefficients decreases quickly as the length of path in network increases [21]. As scattering energy is concentrated on low-frequency paths, fast computation of scattering representations is possible [21]. Scattering transform is non-expansive and complete in the sense that signal reconstruction is possible [21]. Employing scattering networks for feature learning has the advantage that very few parameters have to be tuned according to the nature of dataset.

2.2 Convolutional neural network (CNN)

CNN is a specific class of deep neural network architecture which produces invariant signal representations through trainable filter banks, learned by utilizing nonlinear and pooling operations. In contrast with general neural networks, CNN considerably reduces the number of learning parameters through weight sharing. Weight sharing and the translation invariance achieved through trainable filters make CNN suitable for applying in image-related applications [4,33]. Neural networks which use fully connected layers to classify images do not take into account the spatial structure or the localized information in the images. CNN architecture is particularly suitable for classifying the images. CNN usually comprises mainly three types of layers: convolutional, pooling and fully connected. Figure 2 shows general classification architecture of CNN. Convolutional layers generate convolutional feature maps with the help of trainable receptive fields (or convolutional filters). Weight sharing through filters reduces training parameters that need to be learned. Pooling layers (subsampling layers) reduce the spatial size of the feature representation and thereby control overfitting

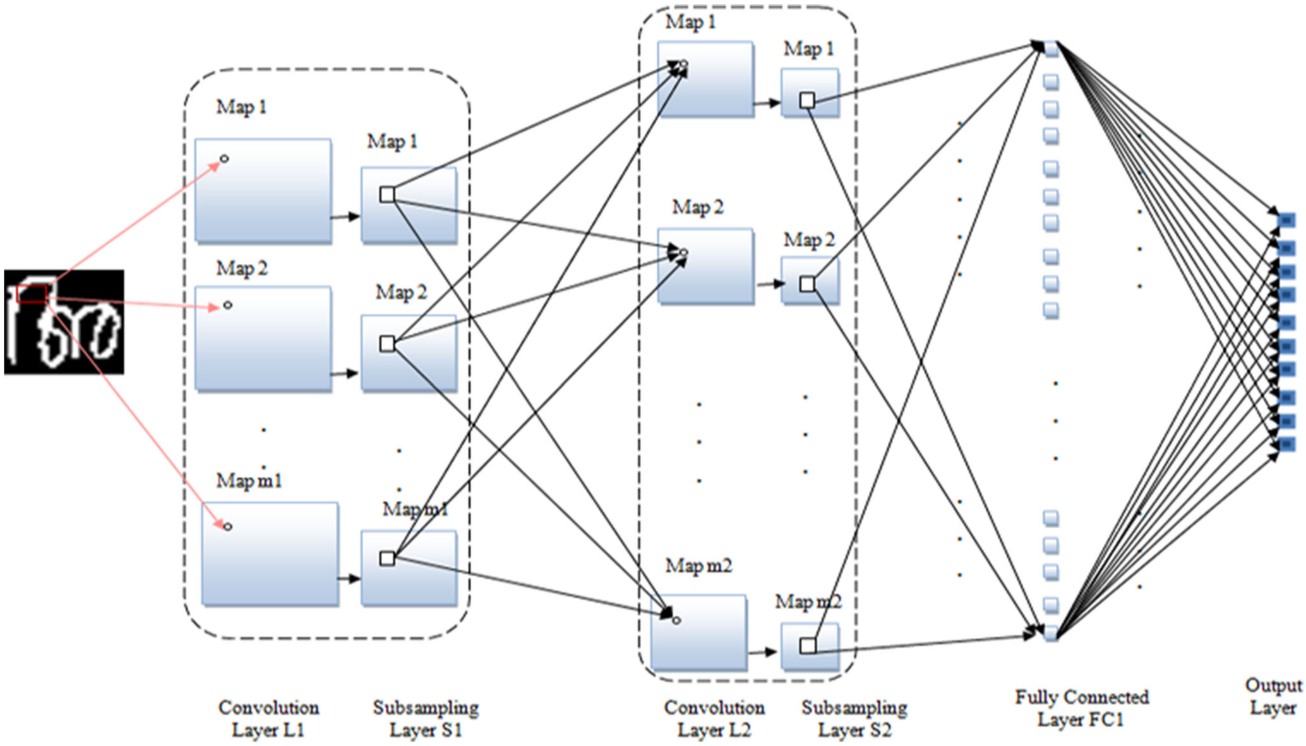

Fig. 2 Convolutional neural network architecture

of network. In most of the network architectures after a convolutional layer, a subsampling layer follows as shown in Fig. 2. Several such combinations of convolutional and subsampling layer can generate invariant feature maps. Finally, these generated features are classified through proceeding fully connected layers in CNN architectures. The output layer in Fig. 2 has number of nodes equivalent to that of target classes. Softmax function on the output layer gives class-wise probabilities for the input pattern. Backpropagation algorithm learns the connecting weights and the bias terms, which in turn minimize the loss value calculated between the target and actual probabilities of training data patterns.

2.3 Proposed architecture

The convolutional filters inside CNN capture informative description from the input patterns which in turn act as the self-learned features. The filter coefficients of the training parameters are learned during the training phase by minimizing the loss function. This paper focuses on utilizing predefined scattering wavelet filters as the first-layer convolutional filters inside CNN. The scattering feature maps are capable of capturing discriminating information through the cascade of wavelet decompositions over input patterns [21]. The proposed architecture is equivalent to replacing the first layer in Fig. 2 with scattering feature maps. The proposed architecture is shown in Fig. 3. Scattering feature maps replace layers $L1$ and $S1$ in Fig. 2. Inside scattering network, each node generates image invariants, which can act as subsampled feature maps of the first layer inside CNN. The proposed architecture is a hybrid framework, which makes use of the strength of both scattering transform and CNN. The effective and stable feature representations extracted through series of scattering transform inside scattering network are integrated with CNN architecture.

Traditional CNN architecture requires to learn the feature representations from the scratch. This hybrid architecture tries to fill the gap between the unsupervised scattering transform representation suitable to the particular dataset with the help of supervised CNN architecture. The hybrid architecture helps to reach the convergence region faster and provides faster learning. The hybrid approach combines the power of proven predefined filter strengths and the self-learned feature representations. Utilizing predefined features on the deep network provides the facility to interpret and understand that deep network better, and help the designer to apply better techniques to get stable feature representations [34]. At the same time, utilizing the unsupervised representations in other layers using deep network helps to learn variabilities in the data patterns and learns information which can fill up the performance gap of the predefined representations with the self-learned deep network. Another advantage is that usual deep networks require large dataset to achieve the state-of-the-art performance. Utilizing the scattering transform filter

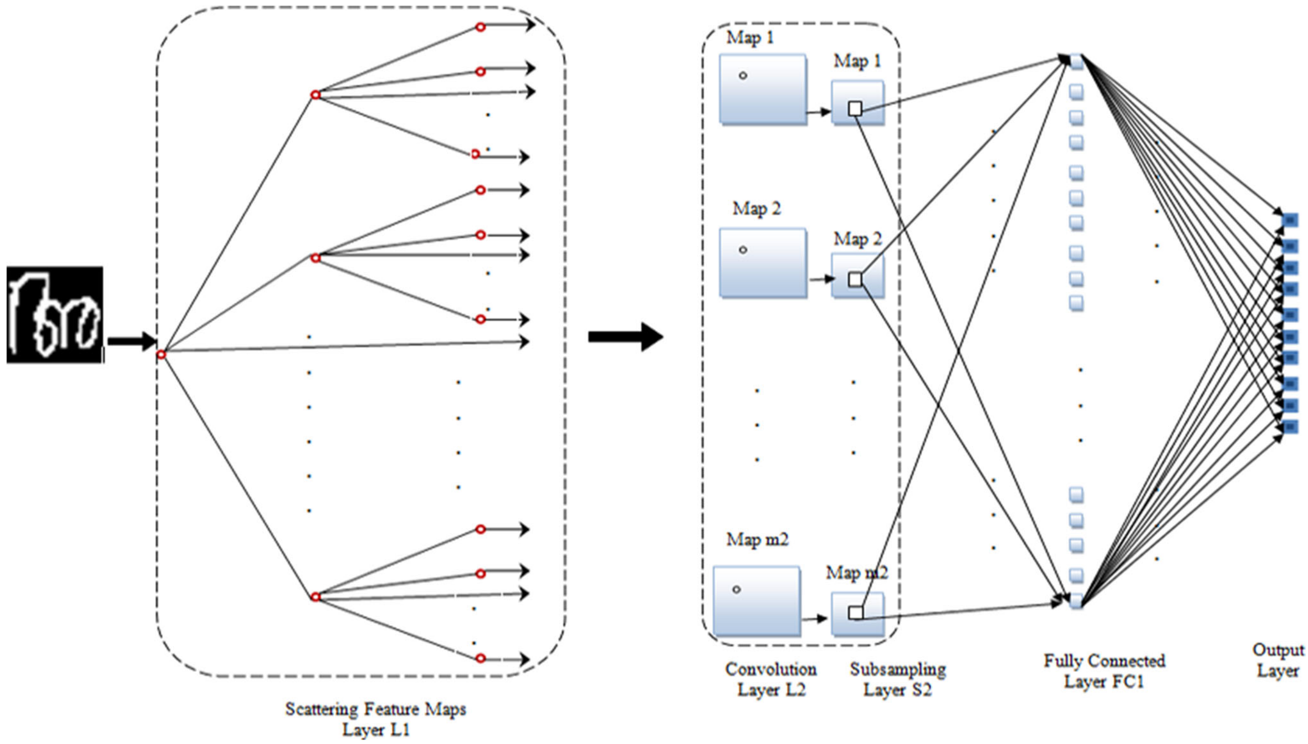

Fig. 3 Proposed architecture

values as initialization values for the first-layer convolutional filters in deep networks has proven better generalization on small datasets [34].

Applying scattering transform for building hybrid architecture can reduce the instabilities in the first layers as the scattering transform is stable and non-expansive. Another advantage is that very few parameters need to be adapted with according to the dataset for the scattering representation which reduces the burden of tuning the parameters suitable for the dataset for the first layer of hybrid architecture. As scattering convolutional network is based on cascade of nonlinear operations which have the capability to reduce the variability, the proposed hybrid architecture can generate discriminative feature information in the first layer of deep network. Hybrid architectures have the capability to reduce the required depth and spatial dimension of the deep learning networks [34].

2.4 Malayalam handwritten database (Malayalam_DB)

Malayalam language script is syllabic in nature. The individual graphic element in the script is termed as *akshara*. Malayalam language contains 16 vowels and 36 consonants. The list of unique character shapes used in writing Malayalam script is more than three hundred. This list consists of vowels, vowel modifiers, consonants, half consonants, allographs (of *ra*, *va*, *ya*), diacritics, compound characters (combination of consonants), numerals and some special symbols. Malayalam language has the largest number of character set among Indian languages [25]. For implementing Malayalam handwritten character recognition system, a character-level image database has been created for the Malayalam script.

In this paper, 85 different Malayalam character classes are considered. Handwritten data sheets collected from 77 persons belong to different age-groups. Fast global minimization algorithm for active contour models (ACM-FGM) [35] applied for detecting the character object in the handwritten document image [36,37]. Character images extracted from the handwritten data sheets collected from 13 persons are considered for testing, and images from rest of the writers are utilized for training classifier models. Training dataset contains 22,942 character image samples, and test dataset has 6360 character image samples. Randomly selected 25% of training images are considered for validation purpose. Figure 4 shows character samples from five different Malayalam character classes in the database. All the character images are converted to binary representation using Otsu's global image threshold algorithm [38], and the size of each image is normalized to 32×32 dimension.

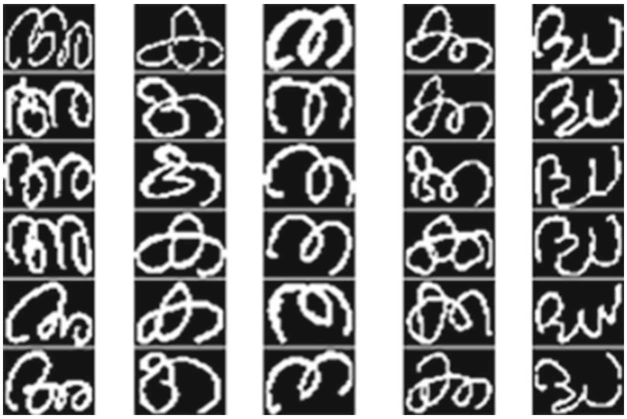

Fig. 4 Sample images from Malayalam handwritten character database

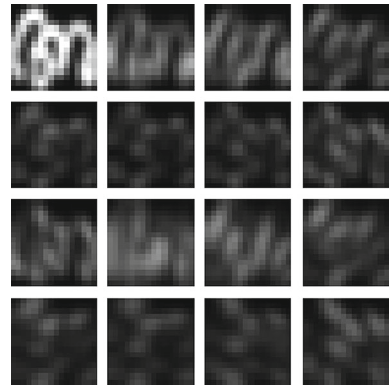

Fig. 5 Scattering feature map for Malayalam letter *a*

3 Results and discussion

This section describes the different experiments conducted toward implementing Malayalam handwritten recognition using CNN and the proposed approach.

3.1 Creation of baseline CNN classifier model for Malayalam_DB

A baseline CNN classifier model for Malayalam handwritten character recognition is built on the training dataset described in Sect. 2.4. The baseline CNN architecture considered for the experiments has four convolutional layers and a fully connected layer. We fixed upon this architecture by conducting some random trials through increasing the number of convolutional layers and number of convolutional maps in each layer. We tried to keep the baseline architecture simple, as the datasets considered are comparatively small when compared to large image recognition tasks. Besides this, the focus of this paper is on evaluating scattering feature maps on the first layer of CNN architecture rather than obtaining state-of-the-art results on the considered dataset. In the considered baseline CNN architecture, the convolutional feature maps in each layer are subjected to max-pooling operation followed Relu activation function. The number of feature maps employed in the four convolutional layers are 100, 200, 300 and 400, respectively. The fourth convolutional layer connected to one fully connected layer. The fully connected layer has 200 hidden neurons which in turn connected to the output layer. Stochastic gradient descent optimization algorithm is utilized for learning weights through backpropagation. Dropout of 0.25 is applied after two consecutive convolutional layers, and in fully connected layer dropout of 0.3 is applied. The loss value is calculated by computing categorical cross-entropy measure between network prediction probabilities and target label vectors. Validation loss value is monitored after each epoch to check the convergence condition. The model learning stopped when validation loss is not reduced for consecutive 30 epochs, and the optimized model is the one for which minimum validation loss value is obtained. The input character images for all the experiments are converted such that the background pixels are zero and the nonzero foreground pixels represent the character information. For implementing the deep learning classifier models, we utilized Tensorflow [39] machine learning framework with Keras [40] as the application programming interface.

The CNN architecture is trained for different learning rates starting from 0.01 to 0.0001, and the learning rate for which maximum validation accuracy obtained is chosen to create the final baseline CNN classifier model. The optimal learning rate obtained for Malayalam_DB is 0.0009. During CNN model learning, as the number of epochs increases both training and validation loss values decrease exponentially. After fifteenth epoch the pace of decrease in loss values is slow. The weight and bias values for which minimum validation loss obtained are saved and used as the final CNN model parameters. For the optimized baseline CNN classifier model, 89.29% recognition accuracy is obtained on the test dataset.

3.2 Scattering feature maps

For creating scattering convolutional network, ScatNet toolbox [41] is utilized in our experiments. Morlet wavelet function is used as ψ and ϕ is chosen as Gaussian. The angular rotations are considered in the range $[0, \pi)$. The scattering feature maps are generated with eight different orientations in rotation group of directional wavelets. Figure 5 shows some randomly selected scattering feature maps generated for Malayalam vowel sign *a* at different nodes in the scattering network for the maximum scale $J = 2$.

Table 1 Recognition performance on scattering network having different depths or maximum layers (m)

Depth of network	Validation accuracy (%)	Testing accuracy (%)
$m = 0$	93.13	82.17
$m = 1$	98.23	93.24
$m = 2$	98.27	93.99
$m = 3$	98.25	93.65

3.3 Proposed ScatCNN architecture on Malayalam_DB

For the same configuration of CNN classifier model described in Sect. 3.1, the first convolutional layer is replaced with the scattering feature maps obtained from the scattering network. It is equivalent to using scattering wavelet filter banks as the fixed filter coefficients in the first convolutional layer of CNN. The proposed architecture is termed as ScatCNN further in this paper. The same learning configuration of CNN and the same parameter tuning in Sect. 3.1 are employed for ScatCNN classifier model creation. ScatCNN has less number of trainable parameters compared to equivalent CNN architecture as the convolutional filters in the first layer of ScatCNN are predefined scattering wavelet filters.

In scattering convolutional network as the depth increases the energy captured among the scattering coefficients converges to zero [21]. Higher-order scattering coefficients have negligible energy, and considering those scattering coefficients as such may decrease the performance of the underlying recognition system [11]. The depth or the maximum number of layers of scattering convolutional network that should be considered for feature extraction is an important decision parameter. Scattering networks having the different numbers of maximum layers are considered for the handwritten Malayalam character recognition by setting $J = 3$ and rotation angles $K = 8$. The performance of the ScatCNN for different numbers of maximum layers m is tabulated in Table 1.

Both validation and testing set recognition accuracy improves with the increase in depth of the scattering network initially. When $m = 2$, maximum recognition accuracy is obtained, but again increasing the depth of scattering network results in performance degradation. After the second layer, the energy of scattering coefficients is negligible and including the second-layer feature directly may degrade the performance [21]. Applying dimension reduction techniques on higher layer scattering representations may be able to capture the prominent discriminant representation [21,42].

3.3.1 Comparison of proposed ScatCNN with baseline CNN

The ScatCNN architecture, which is equivalent to the baseline CNN architecture, is obtained through replacing the first convolutional layer and the proceeding max-pooling layer of baseline CNN with scattering feature maps obtained from the scattering convolutional network. For that, scattering network with $J = 2$, $K = 8$ and depth $m = 2$ is utilized. The optimal learning rate obtained for the ScatCNN learning through the same procedure as that of baseline CNN is 0.007. Figure 6 shows the training loss and validation loss values obtained for optimal baseline CNN (mentioned in Sect. 3.1) and ScatCNN models on all training epochs, till the convergence condition reached. Compared to CNN, ScatCNN obtains less initial training and validation loss values, as the first layer of ScatCNN has well-predefined wavelet filters for creating feature maps. The ScatCNN model converges faster compared to CNN with comparatively less training and validation loss values. In Fig. 6a and b, the sudden decrease in loss values for ScatCNN compared to CNN can be observed. This sudden decrease in loss values helped ScatCNN to converge faster. The invariant features captured through scattering feature maps in the first convolutional layer of proposed architecture could accelerate the learning process of proceeding layers. Compared to CNN, ScatCNN models have stable and better performance for comparatively high learning rate values.

The recognition accuracy obtained for CNN and ScatCNN over the Malayalam_DB test dataset is tabulated in Table 2. The performance of these recognition systems is compared with image pixel values (IMG), histogram of oriented gradients (HOG) [43] and reduced scattering (ScatSVD) [42] features, classified using support vector machine (SVM) classifier. IMG features are the vector representation of image pixel values, while HOG features are obtained by calculating normalized histograms of image gradient directions. ScatSVD features are obtained from two-layer scattering convolutional network, by applying SVD on the second-layer scattering features and concatenating those SVD features with the zeroth- and first-layer scattering coefficients [42]. These features are classified using radial basis function (RBF)-based nonlinear support vector machine classifier. LibSVM toolkit [44] is utilized for experimenting with SVM classifier which makes use of one-against-all multi-class classification approach. The optimal parameters for the SVM classifier models are estimated by using the grid search facility provided in LibSVM toolkit. CNN and scattering-based features have better recognition performance compared with IMG and HOG feature descriptors. ScatSVD in SVM classifier acquired slightly better performance than CNN. The recognition accuracy of ScatSVD improved when the scattering feature maps are applied to deep learning framework, ScatCNN.

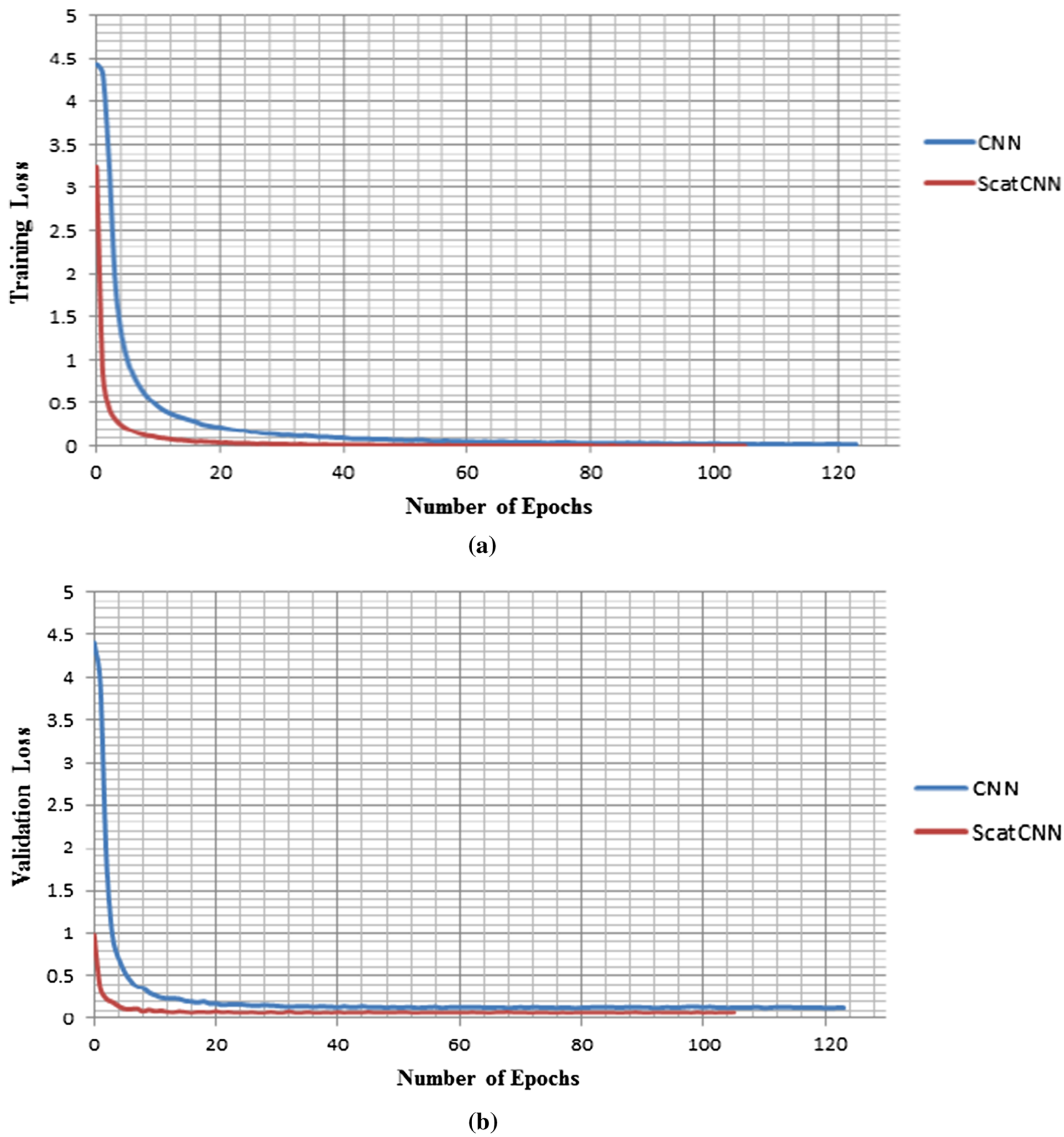

Fig. 6 Variation in loss values for CNN and ScatCNN on Malayalam_DB. **a** training loss. **b** Validation loss

Table 2 Recognition accuracy obtained on Malayalam_DB for different recognizers

Recognizer	Recognition accuracy (%)
ScatCNN	93.77
CNN	89.29
ScatSVD in SVM	90.96
HOG in SVM	86.72
IMG in SVM	77.22

Among the 10.71% of misclassification in CNN architecture, 5.99% of misclassifications got correctly classified through ScatCNN. Four Malayalam character classes along with their misclassification rate in CNN and ScatCNN are listed in Table 3, for which the misclassification in CNN classifier got corrected through ScatCNN architecture.

3.3.2 Proposed ScatCNN in big deep neural networks

The proposed approach of replacing the first convolutional layer of the CNN architecture with scattering feature maps is also experimented in residual networks (Resnet) [45]. Going deeper or increasing the layers in the neural network

Table 3 Class-wise listing of misclassifications in CNN corrected through ScatCNN

Malayalam character	Misclassification rate (%)	
	CNN	ScatCNN
 Vowel character *aa*	0.38	0.16
 Compound character *nna*	0.20	0.00
 Vowel character *u*	0.41	0.20
 Combined character *ngnga*	0.31	0.17

Table 4 Evaluating resnet architecture by applying proposed approach

Architectures	Recognition accuracy (%)
SmallResnet	91.79
ScatSmallResnet	92.85
SmallResnet + image augmentation	95.27

may sometimes degrade or worse the performance than the shallower networks if the architectural design is not proper. Resnet architecture won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC'15) by employing very deep networks using residual connections [45]. For experimenting on Malayalam_DB, SmallResnet [46] is considered which has obtained 99.60% recognition accuracy in MNIST. ScatSmallResnet is obtained by replacing the first five layers of ZeroPadding, Conv2D, BatchNormalization, PReLu and MaxPooling layers in SmallResnet with scattering feature maps obtained from scattering network with $J = 3$, $K = 8$ and $m = 2$. The number of first convolutional layer feature maps in SmallResnet is fixed as the number of scattering feature maps, to make both ScatSmallResnet and SmallResnet equivalent in architecture. Table 4 lists the recognition performance of these architectures on Malayalam_DB. ScatSmallResnet has obtained better recognition accuracy compared to that of SmallResnet.

As deep network models require more data for better optimization, image augmentation techniques are applied on the existing dataset. SmallResnet classifier model is built on augmented image dataset created by applying rotation, shear, zoom, width shift and height shift operations on the training dataset of Malayalam_DB. The SmallResnet classifier model created on augmented character image dataset obtains higher recognition performance than ScatSmallResnet classifier model on the original character image dataset. That indicates the need for more training instances to acquire better performance on deeper networks. As ScatResnet has better recognition performance compared to the SmallResnet on original Malayalam character dataset, if we apply ScatSmallResnet on extended augmented image dataset, it may improve the recognition accuracy further.

For an image of N pixels, the scattering features from the scattering convolutional network are calculated in $O(N \log N)$ [21]. For the proposed architecture when compared with the equivalent CNN architecture, no learning is needed for the first-layer filter coefficients (receptive fields). That considerably reduces the initial training and validation loss. Compared to CNN, ScatCNN obtains higher recognition performance in comparatively higher learning rates. Besides that, the proposed ScatCNN converges faster compared to the equivalent CNN. The number of learning parameters in the proposed ScatCNN is less compared to that of CNN. The scattering feature extraction is an extra overhead during the testing phase. But faster convergence and less number of learning parameters make ScatCNN usable for classification problems. In small datasets where data augmentation or other synthetic image data creation is not possible, ScatCNN can be a good alternative.

3.4 Performance of proposed ScatCNN on other handwritten character databases

The ScatCNN and CNN architectures are experimented on another three handwritten benchmark database. One of the database is MNIST handwritten digit character database [4], which contains 60,000 training and 10,000 testing digit images. The images have size normalized to 28×28. The other handwritten character database considered for the experiment is ISI handwritten character database for Bangla numerals [47]. The ISI handwritten character database contains 23,392 Bangla numeral character samples collected from mail pieces and job applications. Our focus is on the comparative analysis of scattering feature maps with CNN, rather than improving the benchmark recognition accuracy on the considered databases. Because of this, the same CNN architecture mentioned in Sect. 3.1 is experimented with self-learned convolutional filters and scattering featuremaps on the considered databases. The other handwritten database utilized for experimentation is CASIA handwritten Chinese offline character image database, HWDB1.1 [48]. The HWDB1.1 database contains handwritten samples of 3755 character classes collected from 300 writers. In this paper,

Table 5 Recognition performance of CNN and ScatCNN on other handwritten databases

Dataset	Recognition accuracy (%)	
	CNN	ScatCNN
MNIST	99.18	99.31
ISI Bangla numeral	99.12	99.22
CASIA HWDB1.1 (500 Classes)	90.84	92.09

only 500 character classes from all the writers considered and experimented on CNN and ScatCNN architecture.

The maximum recognition accuracy obtained on the test dataset of both the database using CNN and ScatCNN architectures using the same training configuration as that of Malayalam handwritten database is listed in Table 5. The learning rate for which minimum loss value obtained is selected, for creating the final CNN and ScatCNN model. From Table 5, it is clear that ScatCNN performs better than the baseline CNN technique. Comparing with the current state-of-the-art technique on MNIST, the recognition performance of ScatCNN listed in Table 5 is comparatively less. The CNN models which have acquired the state-of-the-art performance on MNIST have utilized data augmentation techniques. Standard dataset distortion techniques are applied for creating new training instances which in effect help the deep network models to generate best performing classifiers. In this paper, we have not made use of data augmentation techniques on MNIST dataset. Without any data generators and preprocessing techniques, on the standard dataset of MNIST ScatCNN could achieve 99.31%. Applying data augmentation may improve recognition accuracy of ScatCNN on MNIST further.

From the CASIA Chinese handwritten testing dataset, images from the considered 500 classes are tested. For the tested 29,859 images, the ScatCNN obtained greater recognition accuracy compared to the CNN architecture. The experimental evaluation of baseline CNN and ScatCNN on the considered handwritten character datasets proves that introducing scattering wavelet filter banks as the first layer of CNN architecture does improve the recognition performance of the system.

4 Conclusion and future works

Handwritten character recognition is a well-studied research area under pattern recognition. CNN-based deep learning architectures are the state-of-the-art techniques in handwritten character recognition, which learns the informative features with the help of convolutional filter fields. Well-defined and proven filters in the area of image recognition can replace self-learning convolutional filters for speeding up the learning process and improving the recognition performance. This paper deals with replacing the convolutional feature maps in the first layer of CNN architecture with scattering transform-based feature maps. Scattering transform is capable of computing stable invariant description of input patterns by applying a series of wavelet decomposition, modulus and averaging operations. This replacement is equivalent to using scattering wavelet filter bank as the first-layer convolutional filter fields.

Malayalam handwritten character recognition is challenging, due to the strong structural resemblance among character classes and due to the presence of a large number of character classes. The proposed hybrid CNN architecture (ScatCNN) is evaluated in the context of Malayalam handwritten character recognition. Eighty-five Malayalam character classes collected from 77 different persons are employed for the experimental analysis. ScatCNN could achieve better performance compared to the equivalent CNN architecture in validation loss and recognition accuracy. The initial and converging loss values obtained for ScatCNN are less compared to CNN. Experiments on other benchmark handwritten databases confirm that ScatCNN can achieve better recognition performance in handwritten character recognition. The predefined scattering wavelet filters perform better than the self-learned filters in generating invariant feature representation for character images.

The proposed ScatCNN architecture is experimented on Resnet, one among the big deep neural network architecture. For the dataset without applying data augmentation technique, proposed ScatCNN concept applied on Resnet could obtain better recognition accuracy compared to equivalent Resnet architecture. When data augmentation applied, Resnet shows much better performance compared to ScatCNN. One of the possible future works on ScatCNN can be the evaluation of applying data augmentation techniques over the proposed hybrid architecture.

The self-learning features in deep learning architectures are the key-point of their success in most of the pattern recognition application. Our present work is a small attempt toward utilizing one of the proven filter banks inside self-learning CNN architecture. The initial results prove that well-defined filters can improve the recognition performance through hybrid architecture. Another future work of the paper aims at incorporating both self-learning and proven filter in initial layers of CNN for improving recognition accuracy further.

References

1. Trier, Ø.D., Jain, A.K., Taxt, T.: Feature extraction methods for character recognition—a survey. Pattern Recognit. **29**(4), 641–662 (1996)

2. Arica, N., Yarman-Vural, F.T.: An overview of character recognition focused on off-line handwriting. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) **31**(2), 216–233 (2001)
3. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)
4. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
5. Simard, P.Y., Steinkraus, D., Platt, J.C.: Best practices for convolutional neural networks applied to visual document analysis. In: Seventh International Conference on Document Analysis and Recognition, pp. 958–963. IEEE (2003)
6. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: 25th International Conference on Neural Information Processing Systems—Volume 1. NIPS'12, pp. 1097–1105 (2012)
7. LeCun, Y., Kavukcuoglu, K., Farabet, C.: Convolutional networks and applications in vision. In: 2010 IEEE International Symposium on Circuits and Systems, pp. 253–256 (2010)
8. Ciresan, D.C., Meier, U., Gambardella, L.M., Schmidhuber, J.: Convolutional neural network committees for handwritten character classification. In: 2011 International Conference on Document Analysis and Recognition, pp. 1135–1139. IEEE (2011)
9. Ciresan, D.C., Meier, U., Masci, J., Maria Gambardella, L., Schmidhuber, J.: Flexible, high performance convolutional neural networks for image classification. In: International Joint Conference on Artificial Intelligence, pp. 1237–1242 (2011)
10. Ciregan, D., Meier, U., Schmidhuber, J.: Multi-column deep neural networks for image classification. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3642–3649. IEEE (2012)
11. Manjusha, K., Anand Kumar, M., Soman, K.P.: Scattering representation in Malayalam character recognition. In: 2017 Twenty-third National Conference on Communications (NCC), pp. 1–5 (2017)
12. Luan, S., Zhang, B., Chen, C., Cao, X., Ye, Q., Han, J., Liu, J.: Gabor convolutional networks. arXiv preprint arXiv:1705.01450 (2017)
13. Yang, W., Jin, L., Xie, Z., Feng, Z.: Improved deep convolutional neural network for online handwritten Chinese character recognition using domain-specific knowledge. In: 13th International Conference on Document Analysis and Recognition, pp. 551–555. IEEE (2015)
14. Zhong, Z., Jin, L., Xie, Z.: High performance offline handwritten Chinese character recognition using GoogLeNet and directional feature maps. In: 13th International Conference on Document Analysis and Recognition, pp. 846–850. IEEE (2015)
15. Zhang, X.-Y., Bengio, Y., Liu, C.-L.: Online and offline handwritten Chinese character recognition: a comprehensive study and new benchmark. Pattern Recognit. **61**, 348–360 (2017)
16. Perronnin, F., Larlus, D.: Fisher vectors meet neural networks: a hybrid classification architecture. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3743–3752 (2015)
17. Ng, C.J., Teoh, A.B.J.: DCTNet: A simple learning-free approach for face recognition. In: 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, pp. 761–768. IEEE (2015)
18. Chan, T.-H., Jia, K., Gao, S., Lu, J., Zeng, Z., Ma, Y.: PCANet: a simple deep learning baseline for image classification? IEEE Trans. Image Process. **24**(12), 5017–5032 (2015)
19. Chang, S.-Y., Morgan, N.: Robust CNN-based speech recognition with Gabor filter kernels. In: Fifteenth Annual Conference of the International Speech Communication Association. INTERSPEECH, vol. 2014, pp. 905–909 (2014)
20. Wang, Z., Liu, D., Yang, J., Han, W., Huang, T.: Deep networks for image super-resolution with sparse prior. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 370–378 (2015)
21. Bruna, J., Mallat, S.: Invariant scattering convolution networks. IEEE Trans. Pattern Anal. Mach. Intell. **35**(8), 1872–1886 (2013)
22. Nikhila, Haridas, Sowmya, V., Soman, K.P.: Comparative analysis of scattering and random features in hyperspectral image classification. Proc. Comput. Sci. **58**, 307–314 (2015)
23. Sifre, L., Mallat, S.: Rotation, scaling and deformation invariant scattering for texture discrimination. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1233–1240 (2013)
24. Pal, U., Chaudhuri, B.B.: Indian script character recognition: a survey. Pattern Recognit. **37**(9), 1887–1899 (2004)
25. Govindaraju, V., Setlur, S.: Guide to OCR for Indic Scripts. Springer, London (2009)
26. Rahiman, M.A., Rajasree, M.S.: Printed Malayalam character recognition using back-propagation neural networks. In: 2009 IEEE International Advance Computing Conference, pp. 197–201. IEEE (2009)
27. John, J., Pramod, K.V., Balakrishnan, K.: Unconstrained handwritten Malayalam character recognition using wavelet transform and support vector machine classifier. Proc. Eng. **30**, 598–605 (2012)
28. Manjusha, K., Anand Kumar, M., Soman, K.P.: Experimental analysis on character recognition using singular value decomposition and random projection. Int. J. Eng. Technol. **7**(4), 1246–1255 (2015)
29. Kumar, S.S., Manjusha, K., Soman, K.P.: Novel SVD Based Character Recognition Approach for Malayalam Language Script. In: Thampi, S., Abraham, A., Pal, S., Rodriguez, J. (eds) Recent Advances in Intelligent Informatics. Advances in Intelligent Systems and Computing, vol 235. Springer, Cham (2014)
30. Neeba, N.V., Jawahar, C.V.: Empirical evaluation of character classification schemes. In: 2009 Seventh International Conference on Advances in Pattern Recognition, pp. 310–313. IEEE Computer Society (2009)
31. Shivsubramani, K., Loganathan, R., Srinivasan, C.J., Ajay, V., Soman, K.P.: Multiclass hierarchical SVM for recognition of printed Tamil characters. In: International Joint Conference on Artificial Intelligence 2007 Workshop on Analytics for Noisy Unstructured Text Data (2007)
32. Mallat, S.: Group invariant scattering. Commun. Pure Appl. Math. **65**(10), 1331–1398 (2012)
33. LeCun, Y., Bengio, Y.: Convolutional networks for images, speech, and time series. In: Arbib, M.A. (ed.) The Handbook of Brain Theory and Neural Networks, pp. 255–258. MIT Press, Cambridge (1998)
34. Oyallon, E., Belilovsky, E., Zagoruyko, S.: Scaling the scattering transform: deep hybrid networks. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 5619–5628 (2017)
35. Goldstein, T., Bresson, X., Osher, S.: Geometric applications of the split Bregman method: segmentation and surface reconstruction. J. Sci. Comput. **45**(1–3), 272–293 (2010)
36. Soman, K .P., Ramanathan, R.: Digital Signal and Image Processing—The Sparse Way, 1st edn. Elsevier India, Gurgaon (2012)
37. Syama, K., George, N., Swathy Sekhar, C.S., Neethu ,M., Manikandan, S., Soman, K.P.: Performance study of active contour model based character segmentation with nonlinear diffusion. In: International Conference on Advances in Computing and Communications (ICACC), pp. 118–121. IEEE (2012)
38. Otsu, N.: A threshold selection method from gray-level histograms. IEEE Trans. Syst. Man Cybern. **9**(1), 62–66 (1979)
39. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S. Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner,

B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X.: TensorFlow: large-scale machine learning on heterogeneous systems (2015). Software available from tensorflow.org
40. Chollet, F., et al.: Keras. https://keras.io (2015). Accessed 15 Dec 2017
41. Sifre, L., Anden, J., Kapoko, M., Oyallon, E., Lostanlen, V.: ScatNet: a MATLAB toolbox for scattering networks (2013)
42. Manjusha, K., Anand Kumar, M., Soman, K.P.: Reduced scattering representation for Malayalam character recognition. Arab. J. Sci. Eng. (2017). https://doi.org/10.1007/s13369-017-2945-9
43. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol 1, pp. 886–893. IEEE (2005)
44. Chang, C.-C., Lin, C.-J.: Libsvm a library for support vector machines. ACM Trans. Intell. Syst. Technol. **3**(2), 27:1–27:27 (2011)
45. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
46. MNIST competition tensorflow KR group. https://github.com/kkweon/mnist-competition. Accessed On 20 Jan (2018)
47. Bhattacharya, U., Chaudhuri, B.B.: Databases for research on recognition of handwritten characters of Indian scripts. In Eighth International Conference on Document Analysis and Recognition, vol. 2, pp 789–793. IEEE (2005)
48. Liu, C.-L., Yin, F., Wang, D.-H., Wang, Q.-F.,: CASIA online and offline Chinese handwriting databases. In: 2011 International Conference on Document Analysis and Recognition, pp. 37–41. IEEE (2011)