Computers, Materials & Continua
DOI: 10.32604/cmc.2023.032288
Article

Tech Science Press

A Novel Siamese Network for Few/Zero-Shot Handwritten Character Recognition Tasks

Nagwa Elaraby*, Sherif Barakat and Amira Rezk

Department of Information System, Faculty of Computers and Information, Mansoura University, Mansoura, P.O.35516, Egypt
*Corresponding Author: Nagwa Elaraby. Email: nagwamegahed@mans.edu.eg
Received: 13 May 2022; Accepted: 12 July 2022

Abstract: Deep metric learning is one of the recommended methods for the challenge of supporting few/zero-shot learning by deep networks. It depends on building a Siamese architecture of two homogeneous Convolutional Neural Networks (CNNs) for learning a distance function that can map input data from the input space to the feature space. Instead of determining the class of each sample, the Siamese architecture deals with the existence of a few training samples by deciding if the samples share the same class identity or not. The traditional structure for the Siamese architecture was built by forming two CNNs from scratch with randomly initialized weights and trained by binary cross-entropy loss. Building two CNNs from scratch is a trial and error and time-consuming phase. In addition, training with binary cross-entropy loss sometimes leads to poor margins. In this paper, a novel Siamese network is proposed and applied to few/zero-shot Handwritten Character Recognition (HCR) tasks. The novelties of the proposed network are in. 1) Utilizing transfer learning and using the pre-trained AlexNet as a feature extractor in the Siamese architecture. Fine-tuning a pre-trained network is typically faster and easier than building from scratch. 2) Training the Siamese architecture with contrastive loss instead of the binary cross-entropy. Contrastive loss helps the network to learn a nonlinear mapping function that enables it to map the extracted features in the vector space with an optimal way. The proposed network is evaluated on the challenging Chars74K datasets by conducting two experiments. One is for testing the proposed network in few-shot learning while the other is for testing it in zero-shot learning. The recognition accuracy of the proposed network reaches to 85.6% and 82% in few- and zero-shot learning respectively. In addition, a comparison between the performance of the proposed Siamese network and the traditional Siamese CNNs is conducted. The comparison results show that the proposed network achieves higher recognition results in less time. The proposed network reduces the training time from days to hours in both experiments.

Keywords: Handwritten character recognition (HCR); few-shot learning; zero-shot learning; deep metric learning; transfer learning; contrastive loss; Chars74K datasets

1 Introduction

Handwritten Character Recognition (HCR) is a computer-based identification of alphabets and numerals written by natural handwriting [1,2]. HCR systems have been reserved for practical applications in various disciplines like publishing houses industries, Banking sectors, and Healthcare organizations [3]. The development of automatic HCR is a motivating area in the field of pattern recognition and has gained a lot of intense effort. But building a robust HCR system is still a challenging task due to several reasons. The primary ones are the multiplicity of languages and the complex instructions of each language, the huge variations in writing styles among humans, and the unexpected noise on the scanned source images for the handwriting [4]. Dealing and learning with these constraints in machine learning algorithms requires hundreds or thousands of training samples to achieve suitable performance. The situation is getting more complicated if only a few labeled handwritten samples exist. This case can be refereed as few-shot learning in HCR.

Few-shot learning aims to make correct predictions with the restriction of only having a few training samples for each class [5,6]. It is a trail from machines for imitating human ability in acquiring new things from very little overlooking. For example, a child can generalize the shape of a "dinosaur" from a picture in a frame and also can simply identify any person from the first look or by just viewing a few photos of him. Learning from a few examples like human avoids the effort of collecting labeled data and reduces computational costs. Few-shot learning can create more robust and general models that can recognize object-based on fewer data as opposed to the highly specialized models.

Supporting few-shot learning by deep networks is a challenge. The remarkable performance expansions for deep learning occur when plenty of labeled training samples are available [7,8]. Adjusting the enormous parameters of deep networks requires sufficient labeled data. As a result, the best deep learning systems lose their ability to generalization in HCR tasks when only a few samples are available for training. Beside that training deep networks requires big training data, there is also another restriction that controls the deep network performance. This restriction is that training and test data must operate in the same feature space. This means that the deep network will be badly acted when it is tested with classes that have not been seen before in the training phase [9]. This case can be referred as zero-shot learning.

Metric learning is one of the recommended methods for the challenge of supporting few/zero-shot learning by deep networks [10]. Deep metric learning depends on building a Siamese architecture of two homogeneous CNNs that share the same architecture and parameters [11,12]. Fig. 1 represents the traditional structure of the Siamese architecture. It deals with the existence of a few training samples in each class by learning the similarity between inputs to differentiate them.

The Siamese architecture is considered a natural data augmentation technique and can create a large number of training sample pairs from only a few numbers of input characters. Fig. 2 demonstrates a simple example for the way of Siamese architecture in augmenting the training data. The main purpose of the generated training sample pairs by the Siamese architecture is to learn a nonlinear mapping function that enables it to map the extracted features in the vector space in an understandable way. This way considered that features of similar characters that share the same class identity are close to each other and conversely features of dissimilar characters that belong to different classes are far apart. Consequently, the Siamese architecture will be able to discriminate between input characters

not classify between classes [13]. This characteristic guarantees the network's generalization ability. As a result, applying the Siamese architecture will help not only in learning with the existence of few training samples but also in recognizing zero-shot classes without any extra training options.

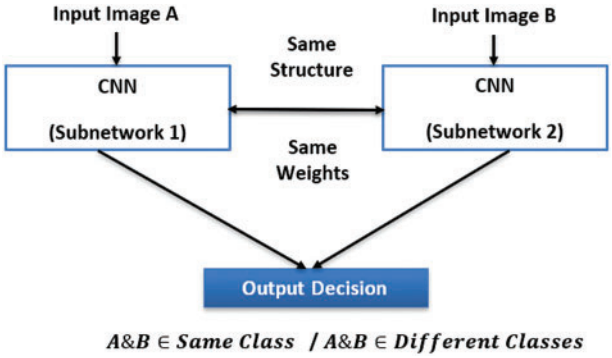

Figure 1: The traditional structure of Siamese architecture

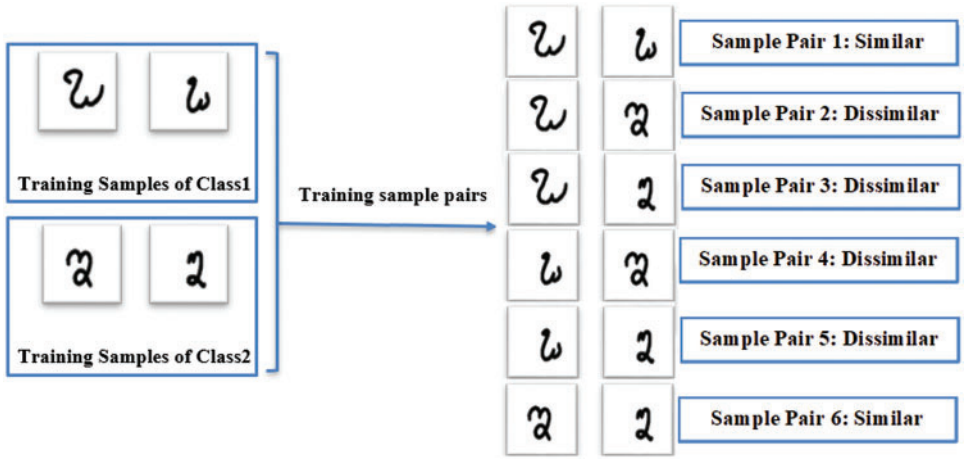

Figure 2: Number of training samples can be augmented to k times of the genuine dataset by the Siamese architecture

The traditional structure for the Siamese architecture used in deep metric learning was built by constructing twin CNNs from scratch with randomly initialized weights and trained by binary cross-entropy loss [11,14,15]. Building two networks from scratch is a trial and error, and time-consuming phase. In addition, training with binary cross-entropy loss sometimes leads to poor margins [16]. So, a novel Siamese network is proposed to avoid the limitations of traditional Siamese networks in few/zero-shot HCR tasks.

The proposed network is built by applying transfer learning instead of forming and initializing two CNNs from scratch. Applying transfer learning by fine-tuning a pre-trained network is typically much faster and easier than training a network from scratch with randomly initialized weights. The preferred pre-trained network that is used in building the proposed network is the AlexNet. AlexNet has a simple deep structure compared with other pre-trained CNNs, but it fails in achieving results and suffers from overfitting when it is trained with few training data. So, utilizing AlexNet in a Siamese architecture will be a valid solution for extending its high achieved performance to few-shot tasks. In

addition, the proposed network is trained by contrastive loss. Contrastive loss works to minimize the distance among the same class's samples and maximize it among different classes' samples. This means that the network is trained well to the extent that enables it to map features belonging to the same class close to each other in the feature space and far from features belonging to other classes.

The rest of the paper is organized as follows. Section 2 presents some existing studies in providing reliable models for few/zero-shot learning in HCR. Section 3 illustrates the steps of building and training the proposed Siamese network. Section 4 discusses the experimental results and evaluation. Finally, Section 5 poses the conclusion and suggestions for future works.

2 Related Work

The challenge of supporting few/zero-shot HCR tasks by deep networks can be reduced by applying one of three methods: data augmentation, meta-learning, and metric learning [10]. In this section, some of the state-of-the-art studies in each method is introduced.

2.1 Data Augmentation

Data augmentation is interested in providing an approach that can enlarge the number of training samples. This is committed by generating synthetic samples or edited copies of the existing samples. Han et al. [17] presented a data augmentation approach based on self-supervised learning for few-shot Oracle Character Recognition (OCR). The proposed approach was a pre-trained Orc-Bert Augmenter. The basic objective of Orc-Bert was generating sample-wise augmented samples by converting pixel format character images into vector format stroke data. The vectorization helped in highlighting the strokes and points of the character and facilitated adding noise to generate augmented samples.

XUI and JIN [18] supported few-shot learning of Korean ancient character recognition by proposing an approach that combined two augmentation methods. The first one considered applying traditional image transformations which were the random affine, elastic distortion, and noise perturbation. While the other method concentrated on generating synthetic samples by using a Conditional Deep Generative Adversarial Network (CDGAN). This combination helped in expanding the dataset and reduced the generalization error with a significant margin.

Hayashi et al. [19] suggested a statistical character structure model for preparing a large amount of image data in the field of HCR. The clue under the proposed model was extracting the strokes that represented the character structure and acquiring their probability distributions. Then, the augmented samples of that character were generated using the learned distributions. The generated character images in this way would cover various handwriting patterns similar to ones written by many people.

2.2 Meta-Learning

The basic thought of meta-learning is to borrow knowledge from formerly trained models for learning new related low data tasks. Varghese et al. [20] introduced a one-shot rule learning approach for the challenging task of Malayalam Character Recognition (MCR). The main contribution of the introduced approach was using a logic program declarative bias. As it facilitated reasoning at the meta-level and helped in reducing the search space for hypothesis derivation by allowing transfer bias from one problem to another related problem. Consequently, the rules of each character were learned in a visually acceptable way.

Zhang et al. [21] proposed an Adversarial Feature Learning (AFL) model for representing the big contrast between handwriting styles. The contribution of applying AFL was utilizing prior

knowledge of standard printed characters to develop writer-independent semantic features. This property combined the strengths of generative and discriminative models and aided in making a better classification.

Jaramillo et al. [22] applied transfer learning to borrow knowledge that aided in HCR with small-size training datasets. Parameters learned from a previously trained model with thousands of samples were borrowed in target HCR problems with a few samples. Also, Sadouk et al. [23] developed a transfer learning system by introducing a pre-trained Phoenician ConvNet and utilizing it in a series of experiments on different target alphabet datasets.

2.3 Metric Learning

This method depends on a Siamese architecture for learning a distance function that can map the input data from the input space to the feature space. Shaffi et al. [15] built a robust deep metric model based on the Siamese CNN for the problem of few-shot learning in Tamil HCR. Each CNN in the Siamese architecture was a five-layer deep network and optimized by binary cross-entropy loss. The model achieved an optimal accuracy with just 40-shots per class.

Dlamini et al. [24] introduced a one-shot learning author verification approach based on the handwritten characters by applying a Siamese architecture. A twin CNN was built, each one composed of three convolutional layers and three fully connected layers. The introduced approach was trained by the pairwise-loss.

Li et al. [14] considered that learning discriminative features in the availability of few training data for Handwritten Chinese Characters (HCC) was a template matching problem. So, a deep template matching model was introduced and built by a Siamese CNN. A collection of machine-printed images generated by a font of Microsoft YaHei were used with the available few handwritten samples for training the model. So that, the network learned a comprehensive similarity metric which enabled it to distinguish between the template and handwritten characters and generalize to zero-shot characters.

Sokar et al. [25] introduced a generic one-shot classification system for the area of Optical Character Recognition (OCR). The proposed system was based on deep Siamese CNNs and Support Vector Machines (SVMs). Firstly, the Siamese CNN was trained to learn a non-linear mapping function then one of the twin networks in the Siamese CNN was used as a feature extractor to train the SVM classifier. Evaluating the proposed system was performed on different domains of Arabic OCR tasks.

In this paper, a metric learning approach is proposed for few/zero-shot HCR tasks. Most state-of-the-art Siamese architectures used in metric learning were built by initializing two CNNs from scratch and trained by binary cross-entropy loss. Building two CNNs from scratch is a trial and error and time-consuming phase. So, the proposed approach introduces a novel Siamese network that is built by applying transfer learning and using the pre-trained AlexNet CNN to be the feature extractor in the Siamese architecture. Fine-tuning pre-trained models achieves fast and not complex training rather than training from scratch. In addition, the proposed network is trained by contrastive loss instead of the traditional binary cross-entropy.

3 The Proposed Siamese Network

This section explains the basic steps of building the proposed Siamese network. Fig. 3 summarizes these steps in a block diagram and then the detail of each step is discussed in the following subsections.

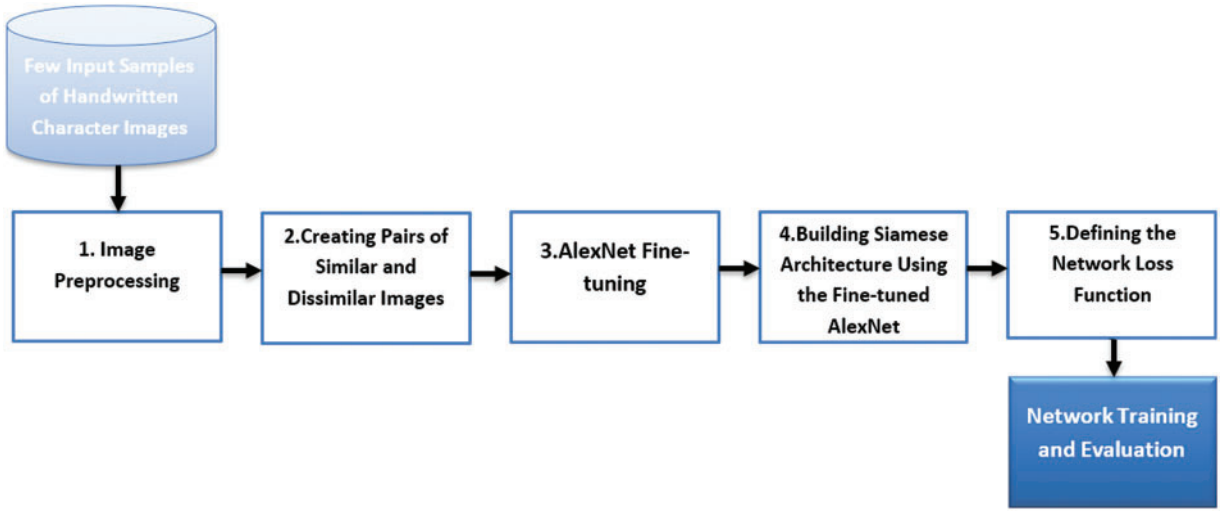

Figure 3: Block diagram for summarizing steps of building and training the proposed network

3.1 Image Preprocessing

The main purpose of preprocessing step in the proposed network is to reduce large variations of the writing styles by unique the thickness, size, and center of each character. This guarantees that the network in the training phase extracts optimal discriminative features of the input characters in a time-saving way. The main issues of preprocessing that are considered to be applied for the proposed network are resizing, conversion, noise removal by applying a Gaussian filter followed by unsharp masking, and binarization. Fig. 4 displays a sample of a handwritten character after each step of the considered preprocessing.

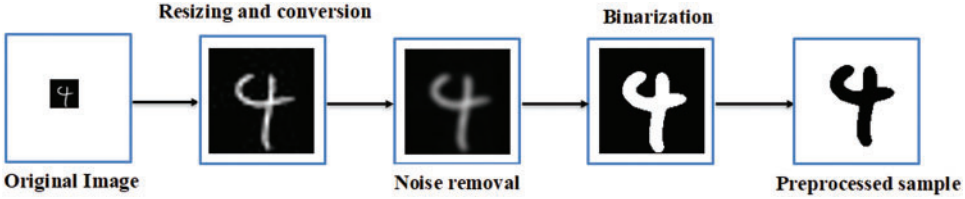

Figure 4: Preprocessing issues considered for the proposed network

3.2 Creating Pairs of Similar and Dissimilar Images

Training the proposed network entails reshaping the full set of preprocessed training samples to be a balanced set of similar and dissimilar image pairs [26]. Each similar image pair is two different handwritten samples of the same character and conversely, each dissimilar image pair is two different handwritten samples for different characters. Algorithm 1 concludes the followed steps in producing a stable set of similar and dissimilar pairs of handwritten characters in the proposed network.

Algorithm 1: Steps of generating training paired images.

1. Let the full set of training samples is $\boldsymbol{X}$ and the set of training pairs is $\boldsymbol{S}$.
2. Randomly select two samples $(\boldsymbol{X}_1, \boldsymbol{X}_2)$ from $(\boldsymbol{X})$.
3. If $\boldsymbol{X}_1, \boldsymbol{X}_2$ are the same sample, continue (2).
4. If $\boldsymbol{X}_1, \boldsymbol{X}_2$ are different samples for the same character, set label to 0; If $\boldsymbol{X}_1, \boldsymbol{X}_2$ are different samples for different characters, set label to 1.
5. Form a pair of training sample, which is $\boldsymbol{s}$:$(\boldsymbol{X}_1, \boldsymbol{X}_2, 0)$ or $(\boldsymbol{X}_1, \boldsymbol{X}_2, 1)$.
6. If $\boldsymbol{s}$ does not exist in $\boldsymbol{S}$, add it, otherwise, continue (2).
7. When the number of training sample pairs in $\boldsymbol{S}$ reaches the set-point, end.

3.3 AlexNet Fine-tuning

AlexNet is a CNN model proposed by Krizhevsky et al. [27]. The architecture of AlexNet consists of 25 layers, eight of them are learnable layers (five convolutional and three fully connected). It trained on approximately 1.2 million high-resolution images from the ImageNet database to classify 1000 different object categories. The total learned weights and biases of AlexNet exceeded 60 million parameters [28].

Several studies were successed in utilizing the previously learned parameters of AlexNet for extracting features in numerous HCR tasks [29,30]. It is observed that every time the training set increases, the AlexNet recognition rate also increases. As a result, AlexNet suffers from overfitting when it is applied for classifying few-shot HCR tasks [31]. Utilizing AlexNet in a Siamese architecture will be a valid solution for such a problem. The Siamese architecture reformulates the few input samples to be an enormous number of training input pairs. So, using AlexNet as a feature extractor in the Siamese architecture will 1) lead to easy and fast training, and 2) make AlexNet more suitable for few-shot learning.

Hua et al. [28] compared five different fine-tuning settings for AlexNet. Each setting determined which layers would be frozen and which ones would be eliminated or replaced to make AlexNet suitable for new target models. Fig. 5 summarizes these settings. We find that Setting D is an optimal fine-tuned AlexNet for the proposed network. Setting D indicates that all the AlexNet's Convolutional Layers (CLs) and the first Fully Connected (FC) layer are frozen. These layers are usually responsible for creating a feature map for the detected features in the input. But the last fully connected layers are eliminated and no SoftMax or classification layers are added.

3.4 Building Siamese Architecture Using the Fine-tuned AlexNet

Fig. 6 explains the total structure of the proposed Siamese network. Each subnetwork in the proposed Siamese architecture is formed by a fine-tuned AlexNet. As the Siamese architecture accepts a couple of images at once, each AlexNet parallelly receives an image to extract its corresponding features. The extracted features by each one is an output vector mapped in the feature space. The two output vectors will be mapped behind together if the two inputs are belonging to the same class. Conversely, if the two inputs are belonging to different classes, the two output vectors will be mapped far from each other. The distance metric between the two output vectors is the Euclidean distance and is calculated as Eq. (1).

$$\boldsymbol{d} = ||\boldsymbol{f}(\boldsymbol{x}_1) - \boldsymbol{f}(\boldsymbol{x}_2)||_2 \tag{1}$$

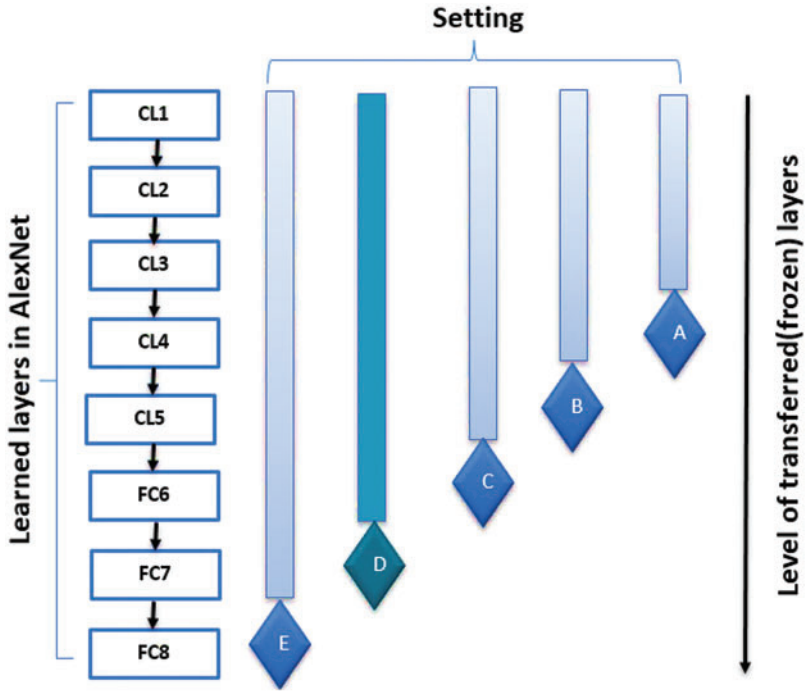

Figure 5: Different settings for fine-tuning AlexNet (Setting *D* fits the proposed network)

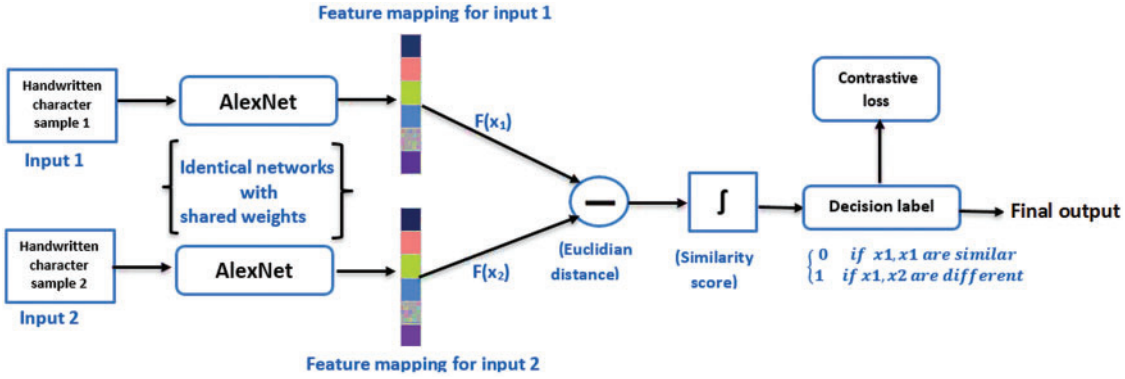

Figure 6: The proposed Siamese network

Reshaping the value of d to be a probability between 0 and 1 is needed in determining the decision label. So, a sigmoid operation is applied based on Eq. (2) to determine a meaningful similarity score. If the output vectors are sufficiently close in the feature map, then the system will decide that the input images are similar and belonging to the same class. Otherwise, they are dissimilar and belonging to different classes. The final decision (D) of the system depends on Eq. (3).

$$\boldsymbol{S}(\boldsymbol{d}) = \frac{1}{1 + \boldsymbol{e}^{-d}} \tag{2}$$

$$\boldsymbol{D} = \begin{cases} 0 & if\ d < 0.5 \\ 1 & if\ d \geq 0.5 \end{cases} \tag{3}$$

3.5 Defining the Network Loss Function

Making an adaptive estimation during training between the proposed network decisions and the ground truth is considered by applying contrastive loss function. Contrastive loss has achieved an optimized weight learning compared with the binary cross-entropy [16]. Contrastive loss function is calculated as Eq. (4).

$$\boldsymbol{loss}\,(\boldsymbol{D}, \boldsymbol{Y}) = (1 - \boldsymbol{y})\,(\boldsymbol{d})^2 + \boldsymbol{y}\,[\max\ ((\boldsymbol{m} - \boldsymbol{d})\,, 0)]^2 \tag{4}$$

where y is the true label ($y = 1$ for dissimilar pairs, $y = 0$ for similar pairs), m is a hyper-parameter called margin and has a value more than zero.

The first part of Eq. (3) is related to loss of similar pairs. As indicated in Fig. 7a, for any similar pair, the value of d is small and near to zero, and square of d is extremely small. So, the network decides that the two images are similar. As a result, the predicted label will be equal to the true label. So, the calculated loss will be near to zero, and the learned weights are not needed to be changed. In contrast, if the loss value is large this implies that the predicted distance between the two images is different from the actual distance. In this case, the network will be optimized to minimize the distance between similar image pairs.

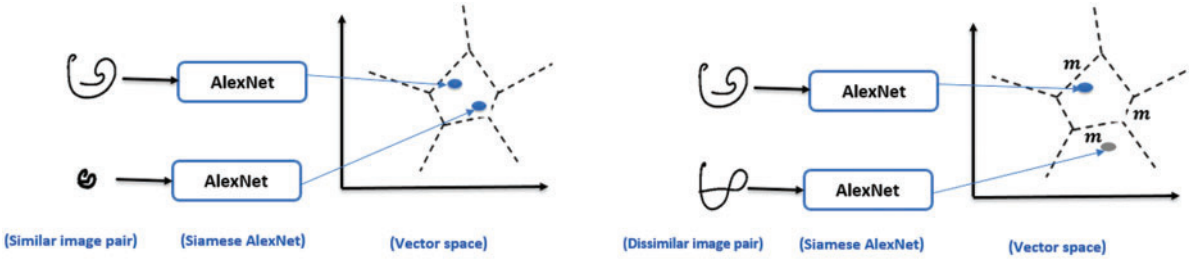

Figure 7: An illustration for the feature mapping in the proposed network

The second part of Eq. (3) is considering the loss of dissimilar pairs. For any dissimilar pair, the value of d is large as indicated in Fig. 7b, and the difference between m and d is a negative value, so 0 is taken as the maximum value in the equation and the loss will be zero. This signifies that the predicted label is equal to the actual label. In contrast, if the value of d is small then the difference between m and d will be the m value, so m is taken as the maximum value and the loss will be large. In this case, the network will be optimized to maximize the distance between dissimilar image pairs.

Minimizing the total loss in the proposed network is considered by applying the Adaptive Moment Estimation (ADAM) optimization algorithm. ADAM combined advantages of two algorithms: the gradient descent with momentum algorithm and the Root Mean Square Propagation (RMSProp) algorithm [32]. This combination gives it ability to update network weights repeatedly by maintaining a learning rate for each parameter instead of applying only a single learning rate for all weight updates. Besides that, the exponentially decaying averages of past squared gradients are stored. These abilities help ADAM to converge faster.

4 Experimental Results and Evaluation

In this section, two experiments are conducted. The first one is for evaluating performance of the proposed network in few-shot learning. While the second one is for evaluating it in zero-shot learning.

One of the official datasets is used in our experiments which is the Chars74K package [33,34]. All experiments are implemented in MATLAB 2021a on a personal computer Intel Core i7 with 2.60 GHz processor and 16 GB of RAM.

4.1 Datasets and Experiments Description

4.1.1 Chars74k Datasets

Chars74K package comprises two datasets of handwritten characters. The first dataset is the EnglishHnd which consists of 62 classes of English characters ($0-9, A-Z, a-z$). Each class has 55 different handwritten samples of the same character. Each sample is an RGB image of size 900×1200. While the second dataset is the KannadaHnd which covers 657 classes of Kannada characters. Each class has 25 handwritten samples that have the same type and size of English characters. Fig. 8 shows some samples from each dataset.

Figure 8: Samples of Chars74K datasets (the first row represented samples from EnglishHnd dataset and the second row represented samples from KannadaHnd dataset)

4.1.2 Experiments Description

The proposed network is trained on each dataset separately to conduct two experiments (A and B). The details, and the purpose of each experiment are summarized as follows:

- Experiment (A): This experiment is performed on EnglishHnd dataset. The high challenge of this dataset is due to some troubles in the writing style of English characters. One of these troubles is the shape of capital and small samples in some letters are very close to each other. Such as letters c, k, o, x, and z. Another trouble is the shape of some digits is near to shape of some letters. Such as letter o and digit 0. This makes difference between some classes hardly observable. We intend to exploit this challenge for testing the proposed network in distinguishing new unseen samples that belong to seen classes in the training phase (few-shot learning experiment). So, all the samples of each class in EnglishHnd dataset are divided to $\sim$ 30% for training (17 samples) and $\sim$ 70% for testing (38 samples).
- Experiment (B): This experiment is performed on KannadaHnd dataset. The high challenges of this dataset are due to the large-scale vocabularies and the complicated structural hierarchy of kannada characters. This makes number of the dataset's classes extremely large. We intend to exploit this challenge for testing the proposed network in distinguishing samples that belong to unseen classes in the training phase (zero-shot learning experiment). So, classes of KannadaHnd dataset are divided to 70% for training (467 class) and $\sim$ 30% for testing (200 class).

4.2 Preprocessing and Training Setting

4.2.1 Preprocessing Setting

Firstly, all images are resized to have the same input size of AlexNet which is 227×227 and converted to be gray scale images rather than RGB. The needed time for processing colored image is

threefold longer than the needed time for processing a grayscale one. Secondly smoothing images and reducing the level of noise are considered by applying an unsharp masking followed by Gaussian filter with kernel value equal to 2. Finally, all images are binarized with Luminance threshold equal to 0.9.

4.2.2 Training Setting

After generating the training sample pairs, the proposed network's parameters are adjusted. The batch size is set to 180, this means that the network is trained by 180 paired images at each iteration on the training loop. For ADAM optimization, the value of margin, learning rate, gradient decay factor, and squared gradient decay factor is set to 0.3, 0.0001,0.9, and 0.99, respectively. In addition, the trailing average gradient and trailing average gradient-square decay rates are initialized. Finally, after specifying training options, the proposed network is trained for 10000 iterations.

4.3 Experimental Results

4.3.1 Experiment (A) on EnglishHnd Dataset

The loss of the proposed network almost converges to zero after the specified training iterations. This denotes that the network reaches to an optimal weight learning and generates a correct mapping function. Such function enables it to map the extracted features from same class's images close to each other and the features from different classes' images far apart. Fig. 9 displays a test batch of image pairs with the proposed network's prediction and similarity score, and the ground truth label.

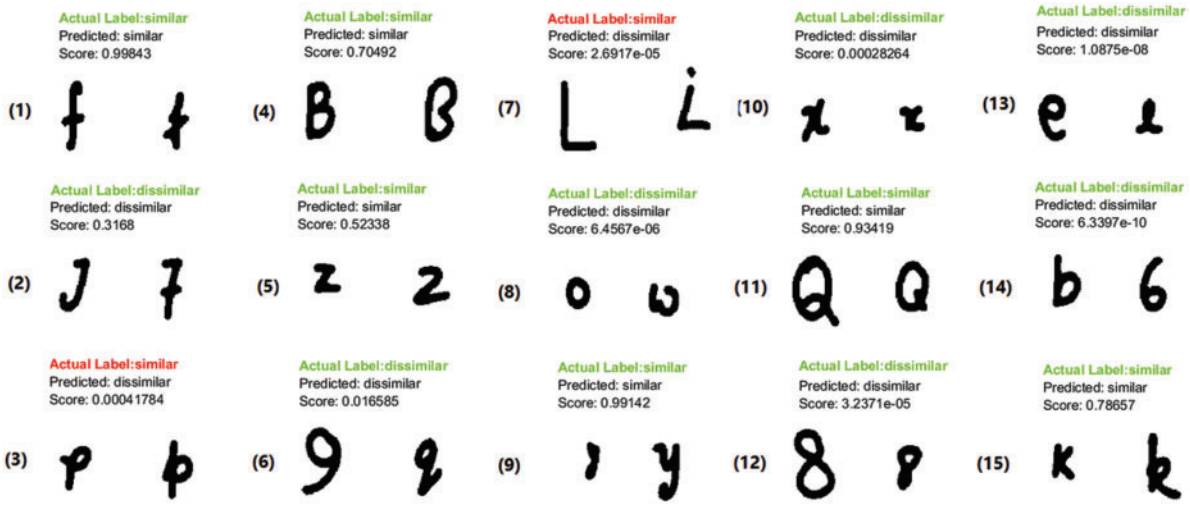

Figure 9: Test batch of size 15 to visually verify if the proposed network correctly recognizes similar and dissimilar new test samples of EnglishHnd dataset

From the test samples appeared in Fig. 9, it is observed that even though the proposed network trained with few samples for each class, but it achieves high success in distinguishing the characters' classes. The proposed network proves its ability in 1) recognizing writing variations of characters belong to the same class such as test pairs (1), (4), (5), (9), (11), and (15). 2) Distinguishing dissimilar characters that belong to different classes but have similar form of writing such as test pairs (2), (6), (8), (10), (12), (13), and (14). For errors that occurred by the proposed network in the test pairs (3) and (7), it may be due to incorrect writing that causes outliers.

Training the proposed network using contrastive loss instead of binary cross-entropy loss helps in increasing the performance. Tab. 1 displays the recognition results of the proposed network under two training cases. The first case when it is trained by contrastive loss while second case when it is trained by binary cross-entropy loss. Training with contrastive loss achieves the highest recognition results according to four measures: accuracy, precision, recall, and specificity.

Table 1: Performance of different training cases for the proposed network on EnglishHnd dataset

Recognition measure	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)
The proposed network trained by contrastive loss	85.6	98.5	79.3	97.7
The proposed network trained by binary cross-entropy	83.3	96.7	76.7	95.3

For evaluating the proposed network on few-shot learning, its performance is compared with the performance of traditional Siamese CNN on EnglishHnd dataset. The Siamese CNN that used in the comparison is built by following the same architecture that was introduced in [11]. It was considered the default structure for the Siamese architecture. Each subnetwork was a CNN with four CLs and one FC. In addition, binary cross-entropy loss was used in training. Tab. 2 presents the results of the comparison.

Table 2: Comparison between performance of the proposed network and traditional Siamese CNN on EnglishHnd dataset

Model	Training time (10000 iteration)	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)
Proposed Network	12 h	85.6	98.5	79.3	97.7
Siamese CNN	98 h	83.9	96.8	76.3	95.7

Tab. 2 indicates that training the proposed network takes time approximately $\frac{1}{8}$ the training time of traditional Siamese CNN. Applying transfer learning by using a pre-trained CNN with learned weights leads to faster training than building a CNN from scratch with initial weights. In addition, training the Siamese AlexNet with contrastive loss helps in increasing the network performance compared with binary cross-entropy loss used in the Siamese CNN.

For further evaluations on EnglishHnd dataset, recognition results of the proposed network are compared with recent state-of-the-art classification models used for it. Tab. 3 shows results of the comparison. It is remarked that the state of the art deep model represented in [35] was applied data augmentation technique with AlexNet to overcome the existence of few training samples. However, the proposed network achieves the highest recognition accuracy without any augmented data. The Siamese architecture can create enormous training sample pairs from a few samples without affecting the model size. So, the proposed network is considered a more robust and general recognition model for few-shot HCR tasks.

Table 3: Recognition results for the state-of-the art classification models and the proposed network on EnglishHnd dataset

Model	Recognition results (Accuracy (%))
Random forest algorithm + extra tree classifier [36]	68.23
Data augmentation + AlexNet [35]	78
Data augmentation + Adjustment Convolution Network (ACN) [37]	79.4
The proposed network (Siamese AlexNet trained by contrastive loss)	85.6

4.3.2 Experiment (B) on KannadaHnd Dataset

This experiment differs from the experiment (A), It concentrates on testing the proposed network in recognizing handwritten characters from zero-shot classes. Fig. 10 displays results of the proposed network on a test batch of image pairs that belong to new classes not seen before in the training phase. Results submitted that extracted features from the proposed network are strong enough for acquiring the generalization ability. Even though training is done with the existence of few samples.

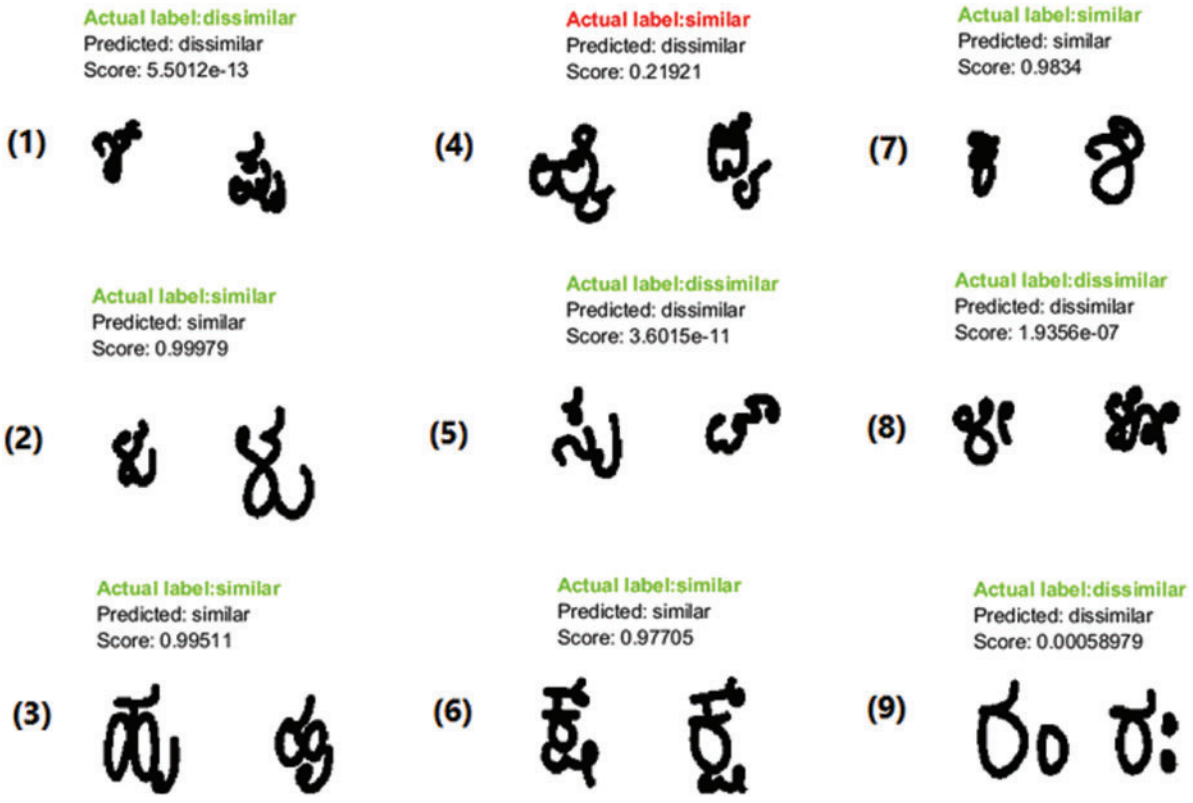

Figure 10: Test batch of size 9 to visually verify if the proposed network correctly recognizes similar and dissimilar new test classes of KannadaHnd dataset

Tab. 4 shows recognition results of the proposed network on zero-shot learning under the two training cases (mentioned before in experiment (A)). Results prove that training with contrastive loss helps in improving performance of the proposed network on zero-shot learning. In addition. For evaluating the proposed network on zero-shot learning, its performance is compared with the performance of traditional Siamese CNN on KannadaHnd dataset. Tab. 5 represents results of the comparison which assures ability of the proposed network in outperforming traditional Siamese CNN in zero-shot learning.

Table 4: Performance of different training cases for the proposed network on KannadaHnd dataset

Recognition measure	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)
The proposed network trained by contrastive loss	82	91.1	79.7	86.1
The proposed network trained by binary cross-entropy	80	90.9	76.9	85.7

Table 5: Comparison between performance of the proposed network and traditional Siamese CNN on KannadaHnd dataset

Model	Training time (10000 iteration)	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)
Proposed network	13 h	82	91.1	79.7	86.1
Siamese CNN	239 h ($\sim$ 10 *days*)	81.2	91	76.1	85.9

There are no recent state-of-the-art models that used KannadaHnd dataset for zero- shot learning to do further evaluations on the dataset. So, the models that test this dataset for solving the challenge of few-shot learning are only stated. Even though, few-shot learning is considered more easier than zero-shot learning, but the proposed network outperforms three models as shown in Tab. 6.

Table 6: Recognition results of the state-of-the-art classification models and the proposed network on KannadaHnd dataset

Model	Recognition results (Accuracy (%))
Artificial neural network (ANN) + histogram of oriented gradients (HOG) [38]	53.4%
CNN [39]	57%
Transfer learning from Devanagari handwritten recognition system + VGG19 NET [40]	73.51%
The proposed network (Siamese AlexNet trained by contrastive loss)	82%

4.4 Discussion

The two experiments are conducted to test behavior of the proposed network on few-and zero-shot learning in HCR. In experiment (A), the samples of EnglishHnd dataset are divided to 30% for training and 70% for testing. The proposed network is tested in recognizing new samples of classes appeared in training but have few training samples. But in experiment (B), the classes of KannadaHnd dataset are divided to 70% for training and 30% for testing. The proposed network is tested in recognizing the new classes that not seen before in training while the appeared classes also have only few training samples.

Building the proposed network by utilizing AlexNet in a Siamese architecture instead of training a Siamese CNN from scratch helps in reducing the training time by a significant margin. Training Siamese CNN from scratch takes approximately 4 days on EnglishHnd dataset and 10 days on KannadaHnd dataset. Both datasets contain few training samples in each class, but KannadaHnd dataset contains large number of classes compared with EnglishHnd dataset. This is the reason that makes training from scratch in KannadaHnd dataset takes longer time. On the other hand, Fine-tuning Siamese AlexNet for the two datasets reduces the training time from days to hours. Fine-tuning Siamese AlexNet for EnglishHnd and KannadaHnd takes approximately 12, *and*13 hours respectively.

In addition, training the Siamese AlexNet by contrastive loss function improves total performance of the proposed network. The comparative results show that training by contrastive loss achieves high recognition results compared with training by binary cross-entropy loss that was used in traditional Siamese architectures. Finally, the proposed network outperforms recent three states of the art classification models used for EnglishHnd dataset. The existed state-of-the-art models on KannadaHnd dataset were tested for few-shot learning not zero-shot learning. Even though, zero-shot learning is considered more difficult than few-shot learning, but the proposed network also outperforms three recent states of the art models applied for KannadaHnd dataset.

5 Conclusion and Future Work

In this paper, a novel Siamese network is proposed for supporting few/zero-shot HCR tasks. The proposed network is built by utilizing transfer learning. The pretrained AlexNet is used to be the feature extractor in the Siamese architecture instead of building twin CNNs from scratch with initialized weights. Applying transfer learning guarantees fast and not complex training. In addition, the proposed network is trained with contrastive loss instead of the traditional binary cross-entropy. Two experiments are conducted using Chars74K datasets to evaluate the proposed network in few- and zero-shot learning. For the few-shot learning experiment, even though the proposed network is trained by 30% of training data which holds only 17 samples. But its recognition accuracy reaches to 85%. For the zero-shot learning experiment, the recognition accuracy of proposed network reaches to 82% when it is trained by 70% of classes and tested on 30% of unseen classes. The proposed network achieves higher recognition results than the traditional Siamese CNN in terms of accuracy, precision, recall, specificity, and training time. Training the proposed network takes time approximately $\frac{1}{8}$ the training time of traditional Siamese CNN. In addition, the proposed network achieves the highest recognition results compared with recent state-of-art on Chars74K datasets.

For future work, we intend to build the proposed network by utilizing a deeper transfer learning model such as GoogleNet, and VGG16 and test it in different few/zero shot learning domains.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] G. S. Monisha and S. Malathi, "Effective survey on handwriting character recognition," in *Computational methods and data engineering*, Singapore: Springer, pp. 115–131, 2021.

[2] N. Babu and A. Soumya, "Character recognition in historical handwritten documents–a survey," in *2019 Int. Conf. on Communication and Signal Processing (ICCSP)*, Chennai, India: IEEE, pp. 299–304, 2019.

[3] M. B. Bora, D. Daimary, K. Amitab and D. Kandar, "Handwritten character recognition from images using CNN-ECOC," *Procedia Computer Science*, vol. 167, pp. 2403–2409, 2020.

[4] R. Vaidya, D. Trivedi, S. Satra and M. Pimpale, "Handwritten character recognition using deep-learning," in *2018 Second Int. Conf. on Inventive Communication and Computational Technologies (ICICCT)*, Coimbatore, India: IEEE, pp. 772–775, 2018.

[5] L. Zhang, J. Liu, M. Luo, X. Chang, Q. Zheng *et al.,* "Scheduled sampling for one-shot learning via matching network," *Pattern Recognition*, vol. 96, pp. 1–11, 2019.

[6] Y. Wang, Q. Yao, J. T. Kwok and L. M. Ni, "Generalizing from a few examples: A survey on few-shot learning," *ACM Computing Surveys (Csur)*, vol. 53, no. 3, pp. 1–34, 2020.

[7] L. Brigato and L. Iocchi, "A close look at deep learning with small data," in *2020 25th Int. Conf. on Pattern Recognition (ICPR)*, Milan, Italy: IEEE, pp. 2490–2497, 2021.

[8] A. J. Moshayedi, A. S. Roy, A. Kolahdooz and Y. Shuxin, "Deep learning application pros and cons over algorithm," *EAI Endorsed Trans. AI Robot*, vol. 1, pp. 1–13, 2022.

[9] H. Kim, J. Lee and H. Byun, "Discriminative deep attributes for generalized zero-shot learning," *Pattern Recognition*, vol. 124, pp. 1–11, 2022.

[10] Y. Zheng, R. Wang, J. Yang, L. Xue and M. Hu, "Principal characteristic networks for few-shot learning," *Journal of Visual Communication and Image Representation*, vol. 59, pp. 563–573, 2019.

[11] G. Koch, R. Zemel and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," *ICML Deep Learning Workshop*, vol. 2, pp. 1–27, 2015.

[12] B. M. Lake, R. Salakhutdinov and J. B. Tenenbaum, "Human-level concept learning through probabilistic program induction," *Science*, vol. 350, no. 6266, pp. 1332–1338, 2015.

[13] S. Jadon, "An overview of deep learning architectures in few-shot learning domain," arXiv preprint arXiv:2008.06365, 2020.

[14] Z. Li, Y. Xiao, Q. Wu, M. Jin and H. Lu, "Deep template matching for offline handwritten Chinese character recognition," *The Journal of Engineering*, vol. 2020, no. 4, pp. 120–124, 2020.

[15] N. Shaffi and F. Hajamohideen, "Few-shot learning for tamil handwritten character recognition using deep siamese convolutional neural network," in *Int. Conf. on Applied Intelligence and Informatics*, Cham: Springer, Cham, pp. 204–215, 2021.

[16] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian *et al.,* "Supervised contrastive learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 18661–18673, 2020.

[17] W. Han, X. Ren, H. Lin, Y. Fu and X. Xue, "Self-supervised learning of Orc-Bert augmentator for recognizing few-shot oracle characters," in *Proc. of the Asian Conf. on Computer Vision*, Japan, Springer, pp. 1–17, 2020.

[18] C. -H. Xue and X. -F. Jin, "Characters recognition of Korean historical document base on data augmentation," in *2020 5th Int. Conf. on Mechanical, Control and Computer Engineering (ICMCCE)*, Harbin, China: IEEE, pp. 2304–2308, 2020.

[19] T. Hayashi, K. Gyohten, H. Ohki and T. Takami, "A study of data augmentation for handwritten character recognition using deep learning," in *2018 16th Int. Conf. on Frontiers in Handwriting Recognition (ICFHR)*, Niagara Falls, NY, USA: IEEE, pp. 552–557, 2018.

[20] D. Varghese and A. Tamaddoni-Nezhad, "One-shot rule learning for challenging character recognition," in *Int. Joint Conf. on Rules and Reasoning (RuleML + RR)*, Leuven, Belgium, vol. 2644, pp. 10–27, 2020.

[21] Y. Zhang, S. Liang, S. Nie, W. Liu and S. Peng, "Robust offline handwritten character recognition through exploring writer-independent features under the guidance of printed data," *Pattern Recognition Letters*, vol. 106, pp. 20–26, 2018.
[22] J. C. A. Jaramillo, J. J. Murillo-Fuentes and P. M. Olmos, "Boosting handwriting text recognition in small databases with transfer learning," in *2018 16th Int. Conf. on Frontiers in Handwriting Recognition (ICFHR)*, Niagara Falls, NY, USA: IEEE, pp. 429–434, 2018.
[23] L. Sadouk, T. Gadi, E. H. Essoufi and A. Bassir, "Handwritten phoenician character recognition and its use to improve recognition of handwritten alphabets with lack of annotated data," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 1, pp. 171–181, 2020.
[24] N. Dlamini and T. L. van Zyl, "Author identification from handwritten characters using siamese CNN," in *2019 Int. Multidisciplinary Information Technology and Engineering Conf. (IMITEC)*, Vanderbijlpark, South Africa: IEEE, pp. 1–6, 2019.
[25] G. Sokar, E. E. Hemayed and M. Rehan, "A generic OCR using deep siamese convolution neural networks," in *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conf. (IEMCON)*, Vancouver, BC, pp. 1238–1244, 2018.
[26] J. Zhang, X. Jin, Y. Liu, A. K. Sangaiah and J. Wang, "Small sample face recognition algorithm based on novel siamese network.," *Journal of Information Processing Systems*, vol. 14, no. 6, pp. 1464–1479, 2018.
[27] A. Krizhevsky, I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.
[28] W. S. Hua, X. Shipeng, C. Xianqing, G. D. S. T. Chaosheng *et al.,* "Alcoholism identification based on an AlexNet transfer learning model," *Frontiers in Psychiatry*, vol. 10, pp. 1–13, 2019.
[29] A. James, J. Manjusha and C. Saravanan, "Malayalam handwritten character recognition using AlexNet based architecture," *Indonesian Journal of Electrical Engineering and Informatics (IJEEI),* vol. 6, no. 4, pp. 393–400, 2018.
[30] S. -G. Lee, Y. Sung, Y. -G. Kim and E. -Y. Cha, "Variations of AlexNet and GoogLeNet to improve Korean character recognition performance," *Journal of Information Processing Systems*, vol. 14, no. 1, pp. 205–217, 2018.
[31] M. Cogswell, F. Ahmed, R. Girshick, L. Zitnick and D. Batra, "Reducing overfitting in deep networks by decorrelating representations," arXiv preprint arXiv:1511.06068, 2015.
[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv Prepr. arXiv1412.6980, 2014.
[33] T. E. De Campos, B. R. Babu and M. Varma, "Character recognition in natural images.," *VISAPP*, vol. 7, no. 2, pp. 2009.
[34] R. Venkatesan, "Novel image representations and learning tasks." *ASU Electronic Theses and Dissertations, ARIZONA STATE UNIVERSITY*, 2017.
[35] A. Shalakhmetov and S. Aubakirov, "Optical character recognition with neural networks," *Journal of Mathematics, Mechanics, Computer Science*, vol. 100, no. 4, pp. 28–41, 2018.
[36] R. Dey and R. Chandra Balabantaray, "A reduced feature representation scheme for offline handwritten character recognition," in *Data Engineering and Intelligent Computing*, Singapore: Springer, pp. 629–637, 2021.
[37] W. Setiawan, "Character recognition using adjustment convolutional network with dropout layer," in *IoP Conf. Series: Materials Science and Engineering, Makassar*," Indonesia, vol. 1125, no. 1, pp. 1–7, 2021.
[38] D. P. Yadav and M. Kumar, "Kannada character recognition in images using histogram of oriented gradients and machine learning," in *Proc. of 2nd Int. Conf. on Computer Vision & Image Processing*, Singapore: Springer, pp. 265–277, 2018.

[39] K. G. Joe, M. Savit and K. Chandrasekaran, "Offline character recognition on segmented handwritten kannada characters," in *2019 Global Conf. for Advancement in Technology (GCAT)*, Bangalore, India: IEEE, pp. 1–5, 2019.

[40] N. S. Rani, A. C. Subramani, A. Kumar and B. R. Pushpa, "Deep learning network architecture based kannada handwritten character recognition," in *2020 Second Int. Conf. on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India: IEEE, pp. 213–220, 2020.