





Few-Shot Learning for Tamil Handwritten Character Recognition Using Deep Siamese Convolutional Neural Network

Noushath Shaffi^(✉)  and Faizal Hajamohideen 

University of Technology and Applied Sciences - Sohar, Sohar, Sultanate of Oman
{noushath,faizalh}.soh@cas.edu.om

Abstract. Optical Character Recognition (OCR) is at the forefront of numerous applications such as digitalization of legal and legacy documents, automatic form processing, writer identification in forensic intelligence. Most of these applications seldom have sufficient training samples in order to achieve an accuracy worthy of real-time deployments. Inspired by the demonstrated performance of Siamese Neural Networks (SNN) in various fields such as Computer vision, Natural Language Processing, Signal processing etc., in this paper, we explore the application of SNN for Tamil Handwritten character recognition. The Siamese-CNN learning is implemented using cross-entropy loss and subsequently used to validate the few-shot learning. It achieved an optimal accuracy of 83.39% for n-way-40-shot learning. Rigorous experiments were conducted all through and the results are indicative of a promising new direction for the development of efficient Indic OCR models using Siamese networks.

Keywords: Siamese network · Cross-entropy loss · Few-shot learning · One-shot learning · Indic OCR

1 Introduction

Humans exhibit supernatural power when it comes to cognitive abilities and the automated systems are striving hard only to come near to this intelligence let alone surpassing it. Conversely, there are several AI based systems that exceed human intelligence but the pitfall is that the former relies on hundreds or thousands of training images than the latter [7]. These systems despite their state-of-the-art advancements, their reliance on the enormous data for model building may lead to failure when presented with less samples [12].

One such application that demands huge data for adequate training of the model is Optical Character Recognition (OCR) – a classic application in the field of Pattern Recognition for the longest time and it has immense value for multilingual multiscript countries like India [10]. Any official document (passport application form, competitive examination forms, judicial documents) may contain the same text in at least 2 or 3 languages. Hence, OCR carries a substantial application potential for a country like India which has 22 official languages and several hundreds of regional languages [10].

Development of a robust Indic OCR system using advanced Deep learning algorithms need huge amounts of training samples in order to generalize well on a previously unseen set of data [12]. Such models when presented with insufficient supervised data may overfit the training samples and/or fail to build a model with good generalizability [12]. Research communities have attempted to address the low-data regime applications through the application of Generative Adversarial Network (GAN), transfer learning or through various data augmentation techniques.

However, there are several shortcomings surrounding the application of these techniques to augment the dataset. For instance, the GANs face the problem of emulating samples that are true representative of character class which may lead to biased model learning. Transfer learning too has limitations such as fine-tuning of the model for the underlying dataset [6]. These are few reasons as to why OCR for many Indic languages have not reported state-of-the-art accuracy.

There exist many Indian languages that fall into the low-data regime – either lacking data or inadequate samples to leverage well established image recognition models (such as VGG16, ResNet, etc.) for the development of robust OCR. As reported in [10], there exists only 19 systematized and comprehensive databases pertaining to 8 Indian script such as Tamil, Telugu, Bangla, Oriya, etc. India is a country with diverse culture and there have been enormous contributions in the field of technology by people belonging to different ethnicities. Hence, it becomes paramount importance to contribute in the advancement of technology concerning all ethnicities. Currently, the research literature indicates that there have been numerous works concerning only those Indian languages that have comprehensive data. Some official Indian languages such as Konkani, Manipuri, Bodo have not even reported the baseline accuracy on the performance of OCR for the respective script. This incapacitated benchmarking can mainly be attributed to the non-availability of sufficient training samples that conventional machine learning and deep learning algorithm demand.

Nevertheless, this can be circumvented by the application of a deep learning technique known as the Siamese Convolutional Neural Networks (CNN) for its ability to learn the model from a limited sample size. The Siamese CNN can effectively assist in the classification task with the constraint that the model can learn only from a single sample per class. This is known as one-shot learning. A natural extension of this concept is zero-shot or few-shot learning, in which the model can either have no sample or only few samples for learning from the target classes. Our primary focus of this paper is to see few-shot learning scenario in the development of an efficient OCR for an Indian language.

- Overall the application of Siamese CNN is still in its infancy especially in the field of development of Indic OCR. Our paper set the benchmark as a new entrant for this field and can serve as a future reference material.
- for very few Indian languages, there are many Indic scripts that lack comprehensive samples for training and testing the model. We propose deep Siamese network models that could maximally leverage from limited data for developing an efficient Indic OCR.

- Models based on Siamese are robust to class imbalance as they rely only on a few samples per class. It is of no significance even if some classes are underrepresented.

In this paper, considering Tamil language as a case study, we leverage a Siamese-CNN for N-way-K-shot learning strategy for the classification of Tamil Handwritten characters, where K can be 1 (one-shot), 0 (zero-shot) or any positive constant (few-shot). This study will pave the way for studying the efficacy of a Siamese model to maximally leverage from the limited data concerning many Indian scripts. We build a twin CNN architecture for feature extraction for subsequent similarity learning between a pair of sample characters. This model is then used to measure the similarity-score between a pair of samples to measure the relative closeness in the classification of 156 Tamil characters. The models built are evaluated using binary cross entropy loss and experiments are substantiated with appropriate analysis. The remainder of this paper is structured as follows: Sect. 2 presents the comprehensive review Siamese CNN in various domains. Sect. 3 presents the model architecture and associated implementation details. Experimental results and analysis are presented in Sect. 4. Finally, conclusion and avenues for sequel are presented in Sect. 5.

2 Literature Review

The Siamese CNN has been applied in various research domains successfully ever since it was first proposed by Tiagman et al. in 2014 [13]. In this section, we report some notable research work done in the last 5 years that employs the Siamese based machine learning model for solving various problems of computer vision and pattern recognition. In [2], a method was presented for word spotting using Siamese-CNN based on similarity between two input word images. The trained model was used to spot words of varying writing styles with vocabulary words that are not in the training set. In [1], the Siamese model was adopted in the process of offline writer identification. The probability distribution functions along with auto-derived CNN features were fed into the Siamese neural network that resulted in encouraging performance. In another work, the Siamese model was used for vehicle reidentification purposes [8]. The model was fed with vehicle shape and features extracted from the license plate where these elements were merged using distance descriptors with a sequence of dense layers. The experiments conducted resulted in an accuracy of 98.7% on a 2 h of video containing 2982 vehicles. Another work was proposed [5] based on the Siamese model for text recognition by measuring the visual similarity and thereby predicting the content of the unlabeled texts. The results demonstrated that the predicted labels sometimes outperformed human labels. Very recently in [11], Siamese Denoising Autoencoder network was proposed which can automatically remove position noise, recover the missing skeleton points and correct outliers in joint trajectories in the process of gait recognition. The Siamese mechanism to reduce between class and increase within-class variations resulted in a robust

model against inaccurate skeleton estimation. Another work reported in [4] successfully employed Siamese graph convolution network (SGCN) using contrastive loss for the task of content-based image retrieval of very high resolution images. Using SGCN, a similarity function is learnt that uses region adjacency graph to better represent the semantically closer samples from dissimilar points thereby a robust CBIR performance was seen. In [3], authors have applied the Siamese framework for an object tracking process that has multiple stages:

Firstly, dynamic weighting module is introduced in the Siamese framework to predict the response maps discriminatively. Secondly a residual structure in order to form the residual dynamic weight module is introduced. Finally, a pyramid-re-detection module was included to avoid unnecessary local search. The resultant model outperformed state-of-the-art object trackers. Another object tracking method was proposed in [14] that introduced the attention module in the traditional Siamese network. A attention shake layer replaced the max pooling layer in the Siamese network which helped to enhance the expression power of Siamese without increasing the depth of the network. Empirical results exhibited good performance on multiple benchmarks. Based on this brief study of existing literature, we can conclude that:

- Siamese models have been applied successfully in various domains of pattern recognition and computer vision and hence it becomes imperative to see its performance in the field of OCR.
- Not much work has been reported in the field of document image processing, especially in the domain of OCR that makes use of a Siamese model in the recognition or clustering of the handwritten characters.
- Overall, the research adapting Siamese models for the development of efficient Indic OCR is fairly immature till-date. We believe that this work will pave the way for more such research in this field to fully exploit the benefit of much acclaimed Siamese models for the few-shot/one-shot/zero-shot learning process.

3 Methodology

We present here the overall architecture of our Siamese CNN for few-shot learning model to classify samples of Handwritten Tamil characters.

3.1 Siamese-CNN Model

The input to the Siamese CNN model are image pairs: Image-1 and Image-2 as shown in the left of Fig. 1. These two images are passed through a ConvNet encoder to transform the input images into an embedding space represented as h_1 and h_2 . The dual ConvNet encoders represent the Siamese network. Although the Siamese architecture is depicted as having dual ConvNet encoders, it basically has a single ConvNet encoder that sequentially extracts features for Image-1 and Image-2. The output of ConvNet encoder is a fully connected (FC) layer

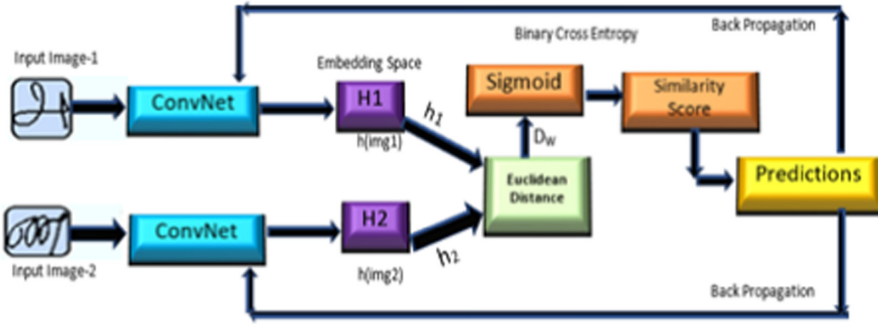


Fig. 1. The Siamese architecture

representing the embeddings. The Euclidean distance (D_w) is then computed between extracted features h_1 and h_2 and fed to a sigmoid activation function to determine the similarity between pairs of images that were input to the model.

The CNN that we used for ConvNet encoder has 2 convolutional layers, 2 pooling layers and 1 FC layer. The input layer contains 64×64 sized images. The architecture can be represented as: $64C2-MP2-64C2-MP2-48N$, where nCj indicates a convolutional layer with n filters and $j \times j$ kernel, MPk refers to a max pool layers with $k \times k$ kernel and fN refers to a fully connected dense layer with 48 neurons. This way a 64×64 sized image is transformed into a 48-dimensional feature vector. Objective is to learn a similarity model that represents image pairs to have high or low similarity for pairs belonging respectively to the same or different classes. Another component of the Siamese model is the loss function which has consequential effects on the overall output produced by the model. The loss function takes Euclidean distance between h_1 and h_2 features and determines if the Siamese model made the correct decision or not. This way the weights of the model are adjusted to output optimal prediction. The objective of the model is to minimize the loss while predicted values remain as closer to true labels as possible.

3.2 Cross-entropy Loss

The cross-entropy loss is also called logarithmic loss. The predicted probability y^p through SoftMax activation will be compared with the actual values y^a and penalized accordingly.

Based on how far the difference between y^p and y^a , the cross-entropy loss will inflict a large penalty closer to 1 for large difference and small penalty closer to 0 for small difference.

As can be seen from Fig. 1 that the cross-entropy loss function will be used for adjusting the weights of the model. A perfect model built will have a cross entropy loss of 0. For binary classification as in our case (similar or dissimilar), the binary cross entropy is defined as follows:

$$L = \frac{1}{N} \left[\sum_{j=1}^N y^a \log(P_j) - (1 - y^a) \log(1 - P_j) \right] \quad (1)$$

Where y^a is the actual value (1 or 0), P_j is the SoftMax probability of j^{th} sample, N is the total training sample.

The Siamese CNN model has a ConvNet for feature extraction followed by a neural network for learning the similarity model. Firstly, the ConvNet encoder will be instantiated as a feature extractor. Then, Image-1 and Image-2 will be transformed to h_1 and h_2 respectively (known as embedding space) using this feature extractor. The (h_1, h_2) pair will be considered as input of the neural network model. The Euclidean distance (L2 norm) between h_1 and h_2 are computed as follows:

$$D_w = ||h_1 - h_2||_2 \quad (2)$$

The Euclidean distance computed will be fed to sigmoid activation function - which will be considered as output from the neural network. Finally, the neural network model will be created with these input and output parameters. Pairwise training of the model will optimize the loss function based on these predicted output and actual ground truth values.

Finally, to check the class to which the input test image x_t belongs - we can pair images as (x_c, x_t) (where c is in range of 1, 2, 3, \dots 156 and t represent the class of test image x_t) and predict the class corresponding to the maximum similarity as follows:

$$c^* = \operatorname{argmax}_c(P(x_c, x_t)) \quad (3)$$

4 Experimental Results

In this section, we report on a series of experiments conducted to test the efficacy of Siamese architecture for the application of OCR systems. The Siamese model was used to test the efficacy of few-shot learning using the uTHCD database. The uTHCD database is a unified collection of offline and online handwritten samples collected from native Tamil speakers and it has a collection of 55000, 7870 and 28080 samples in train, validation and test sets respectively [9]. This database has approximately 600 samples in each of the 156 distinct classes of Tamil script.

The Siamese network needs positive and negative image pairs for training the similarity model. Positive pair is a pair that has samples from the same class and a negative pair has a pair of images from different classes. For every image in the training subset, we randomly pick:

- a sample from the same class for the second image (positive pair)
- a sample from a different class for the second image (negative pair)

All experiments were conducted using this database with different training subset configurations suiting the nature of the underlying test. However, for the



Fig. 2. Sample positive (P) and negative (N) pairs generated for training.

Table 1. Hyperparameter values used in the architecture

Hyperparameter	Value
Embedding Size	48
Loss function	Cross entropy
Batch size	32
Epochs	150
Activation function	ReLU and Sigmoid (last layer)
Learning rate	0.001
Batch size	64

validation set, the random draws are fixed once for all experiments to avoid validating on different datasets at each epoch.

The entire implementation was done using Tensorflow deep learning library with Keras API using Python 3.7.10 environment. The model training and evaluation was done on a GPU machine (NVIDIA GeForce MX330) running alongside an Intel i7 1.6 GHz CPU with 16 GB RAM. All models were learned with *EarlyStopping* callback in Keras to reduce overfitting.

The CNN architecture (as shown in Fig. 1) used in the Siamese model is not very deep but can extract powerful similarity features. There are several hyperparameters such as learning rate, epochs, optimizer etc. that control overall dynamics of the architecture. For the rest of the experiments, unless explicitly mentioned, the hyperparameter values are empirically fixed as shown in Table 1.

To test the effectiveness of similarity learning using the SCNN model, we conducted an experiment by considering the entire 55000 training set. For every image in the training set, we randomly created a positive pair and a negative pair resulting in a total of 110000 pairs. A corresponding label vector was generated that has either 1 or 0 to denote a positive or negative pair respectively. The pair images and corresponding label vectors were used for training the similarity using the SCNN model. Similarly, pair images were generated out of validation set (15740 pairs) and test set (56160 pairs). The Fig. 2 shows some sample positive and negative pairs.

The Siamese model used in this section resulted in a training accuracy of 90.19% with a validation accuracy of 88.04%. The plot of training and validation accuracy and loss are as demonstrated in Fig. 3. It can be seen that the model converged around the 80th epoch. The model resulted in a testing accuracy of 89.23%. It is to be noted that this experiment will result in different values for accuracy and losses depending on the image pairs randomly generated for train, test, validation sets. Hence it is important to fix the randomness of NumPy, Tensorflow and Python built-in pseudo-random generators in order to get reproducible results. Figure 4 shows similarity between random image pairs.

It can be noted that the model outputs a high and low similarity respectively for samples belonging to intra-class and inter-class image pairs. This experiment ensures that the Siamese model was successful in learning the similarity between random samples to a great extent.

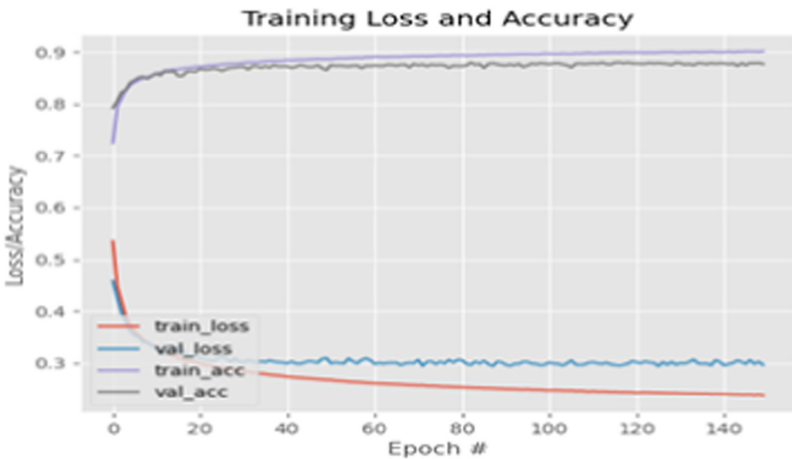


Fig. 3. Learning accuracy of Siamese architecture.



Fig. 4. Result of similarity learning for positive (top) and negative (bottom) image pairs.

The advantage of Siamese learning is in its applicability to learn a model using less, one or zero samples. This is respectively known as K-shot (few-shot), one-shot and zero-shot learning. As mentioned earlier, not all Indic scripts have adequate training samples to build a reliable OCR system. It is important to build a system that leverages maximally from the resources available to eventually reach a stage where it can be practically deployable. This is essentially investigating how well a typical Siamese based OCR performance changes with varying numbers of shots. In order to check the effectiveness of the Siamese network for this purpose, in this section, we conducted a series of experiments for few-shot learning by fixing only K number of samples from each of the 156 classes. The experiment was conducted by choosing randomly K samples from each class and for each sample, ten image pairs were generated with 5 negative and positive pair combinations. For testing purposes, we used the same subset as described previously. This test set is fixed for all experiments involving few shot learning models.

The results of validation accuracy and loss for different values of K are as shown in the Fig. 5. It can be ascertained from the plot that the Siamese models with $K = 10$ and $K = 20$ needed more data as it took more epochs to converge. In addition, after a certain number of epochs, the validation loss started to increase indicating that the model is suffering from a high-variance problem leading to divergence of the model on the validation set. This suggests that the model with too few samples ($K = 10$ or $K = 20$) using Siamese may not be adequate to develop a reliable OCR for Indic script. Rest of the models ($K = 40$, $K = 60$ and $K = 80$) exhibit performance that are on par with each other. Among these models, the model with $K = 40$ seems to be a good choice as it was able to learn sufficiently when presented with less data (40 samples per class) and there is no significant difference between results of this model with models using a higher number of samples.

Table 2 presents the optimal value of training and validation accuracy for different values of K . The testing accuracy saw an incremental improvement only when trained with rapidly increased pairs of images. The model with $K = 40$

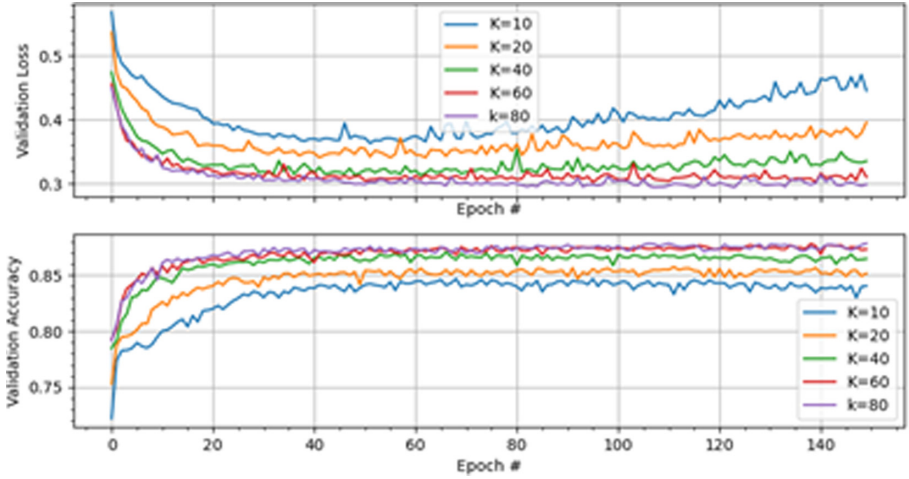


Fig. 5. The results of few-shot learning.

seems to strike a very good balance in the computational time and accuracy trade-off. Testing accuracy indicates the total number of image pairs that were correctly classified as similar or dissimilar with a threshold of 0.5. It is evident from these results that the performance metrics gradually increase with increased number of shots (varying K) as the model has access to an increased number of pairs for training.

Table 2. Result of few shot learning

Model	Number of pairs	Precision	Recall	F1-score	Validation accuracy	Test accuracy
10-shot	15600	0.8057	0.8052	0.8051	0.8396	0.8052
20-shot	31200	0.8147	0.8146	0.8146	0.8532	0.8146
40-shot	62400	0.8325	0.8323	0.8322	0.8745	0.8339
60-shot	93600	0.8404	0.8400	0.8399	0.8775	0.84001
80-shot	124800	0.8477	0.8471	0.8471	0.8881	0.8471
100-shot	156000	0.8495	0.8493	0.8493	0.8934	0.8493

5 Conclusion and Future Avenues

In this paper we have proposed the Siamese CNN for implementing few-shot learning - a mechanism that leverages a minimal number of samples to build a robust model - for the problem of Tamil OCR. We used the binary cross-entropy loss to calibrate the Siamese-CNN model. The model resulted with a

test accuracy of 83.39% with 40-shot learning. Among the models we tested, the model with 40-shots (40 samples per class) achieved an optimal accuracy. The work presented in this paper can be extended in a number of ways as below which deserves further study:

- The Siamese model can be implemented using the contrastive loss function instead of just relying on the cross-entropy loss function. As this loss function is based on distance measure, it ensures semantically closer examples are embedded closer as against binary cross-entropy loss function that adjusts the weights of the model based on probability output by the model.
- Tamil script is a language where there may exist only minor inter-class variation through the presence/absence of tiny-dot, a loop, a stroke etc. This way samples from different classes look near-identical and can drastically impact the performance metrics of the Siamese model. This can be mitigated by considering 50% each of training pairs from hard and easy categories. In the hard category, for every image (base) the negative pair was formed by considering any compound characters pertaining to the same base character. An easy category is where the negative pair was composed randomly as done in all experiments in this paper. This will ensure a robust similarity model learning unlike the model that we developed only based on random samples (easy category).
- The ConvNet encoder that we utilized in the implemented Siamese model is not deep. The Siamese architecture is known to perform even better when the CNN used for feature extraction is a deep architecture. Hence, the performance can be further increased if we can fine tune the pretrained models such as VGG16, VGG19, AlexNet, ResNet etc.

The Siamese Neural network has not seen a wide-spread applicability so far in the field of Indic OCR development. We believe that the work presented in this paper would serve as a prelude for many such works based on Siamese models.

References

1. Adak, C., Marinai, S., Chaudhuri, B.B., Blumenstein, M.: Offline Bengali writer verification by PDF-CNN and Siamese net. In: 2018 13th IAPR International Workshop on Document Analysis Systems (DAS), pp. 381–386. IEEE (2018)
2. Barakat, B.K., Alasam, R., El-Sana, J.: Word spotting using convolutional Siamese network. In: 2018 13th IAPR International Workshop on Document Analysis Systems (DAS), pp. 229–234. IEEE (2018)
3. Cao, Y., Ji, H., Zhang, W., Xue, F.: Visual tracking via dynamic weighting with pyramid-redetection based Siamese networks. *J. Vis. Commun. Image Represent.* **65**, 102635 (2019)
4. Chaudhuri, U., Banerjee, B., Bhattacharya, A.: Siamese graph convolutional network for content based remote sensing image retrieval. *Comput. Vis. Image Underst.* **184**, 22–30 (2019)
5. Hosseini-Asl, E., Guha, A.: Similarity-based text recognition by deeply supervised Siamese network. arXiv preprint [arXiv:1511.04397](https://arxiv.org/abs/1511.04397) (2015)

6. Kornblith, S., Shlens, J., Le, Q.V.: Do better imagenet models transfer better? In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2661–2671 (2019)
7. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. *Adv. Neural. Inf. Process. Syst.* **25**, 1097–1105 (2012)
8. de Oliveira, I.O., Fonseca, K.V., Minetto, R.: A two-stream Siamese neural network for vehicle re-identification by using non-overlapping cameras. In: 2019 IEEE International Conference on Image Processing (ICIP), pp. 669–673. IEEE (2019)
9. Shaffi, N., Hajamohideen, F.: uTHCD: a new benchmarking for Tamil handwritten OCR. arXiv preprint [arXiv:2103.07676](https://arxiv.org/abs/2103.07676) (2021)
10. Sharma, R., Kaushik, B.: Offline recognition of handwritten Indic scripts: a state-of-the-art survey and future perspectives. *Comput. Sci. Rev.* **38**, 100302 (2020)
11. Sheng, W., Li, X.: Siamese denoising autoencoders for joints trajectories reconstruction and robust gait recognition. *Neurocomputing* **395**, 86–94 (2020)
12. Strang, G.: *Linear Algebra and Learning from Data*. Wellesley-Cambridge Press, Cambridge (2019)
13. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: DeepFace: closing the gap to human-level performance in face verification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1701–1708 (2014)
14. Wang, J., Liu, W., Xing, W., Wang, L., Zhang, S.: Attention shake Siamese network with auxiliary relocation branch for visual object tracking. *Neurocomputing* **400**, 53–72 (2020)