Few Shot Learning Approach to Online Malayalam Handwritten Character Recognition

MSc Research Project
MSc Data Analytics

Vishnu Venugopal Palakkath
Student ID: 20177470

School of Computing
National College of Ireland

Supervisor: Dr.Bharathi Chakravarthi

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Vishnu Venugopal Palakkath
Student ID:	20177470
Programme:	Master's in Data Analytics
Year:	2021
Module:	MSc Research Project
Supervisor:	Dr.Bharathi Chakravarthi
Submission Due Date:	16/12/2021
Project Title:	Few Shot Learning Approach to Online Malayalam Handwritten Character Recognition
Word Count:	5882
Page Count:	21

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Vishnu Venugopal Palakkath
Date:	31st January 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Few Shot Learning Approach to Online Malayalam Handwritten Character Recognition

Vishnu Venugopal Palakkath
20177470

Abstract

In the field of image processing, handwritten character recognition has become one of the most intriguing studies. An image, document, or real-time device like a tablet, tabloid or digitizer can be used as input for the Handwritten character recognition method. The input is subsequently translated into digital text. Online Handwritten Recognition and Offline Handwritten Recognition are the two commonly utilized methods for recognizing handwriting. Writing styles, such as line spacing, word spacing, character sizes, and the shape of each character, can vary from person to person. Feature extraction and character identification are the most time-consuming and difficult parts of OCR since they depend on the language. According to the features of a language, feature extraction can be different for each one. It is the curvy and non-cursive nature of Malayalam characters that distinguish them from other scripts. To understand the recognition of online handwritten characters in Malayalam, this paper compares traditional classification approaches like SVM Kernels that use Histogram of Oriented Gradients as features ,CNN and RNN based deep learning models and CNN based transfer learning models with Siamese Networks.

1 Introduction

1.1 Overview

Improved digitization of character images is being used for human-computer interaction and other purposes. Documents in their original handwritten form should be converted to electronic format for greater preservation, editability, and protection against manipulation. Any documentation program must be used to manually type the entire document if it is to be digitized without Optical Character Recognition (OCR). To a large extent, this challenge was eliminated by the introduction of OCR. OCR aims to identify characters in a picture of text, in order to encode the text in a more convenient format for editing. Text images will be printed or handwritten, depending on your preference. When the document is printed, all of the characters will be the same size and style. Regardless of who wrote the characters, they will all look the same. However, when it comes to a handwritten work, the character differs greatly from writer to writer, making it difficult to process.

Much of the current research is focused on handwriting recognition. The demand for handwritten recognition expanded as devices like Smart phones and digital computers were developed that could successfully interpret and reproduce the data entered. In offline systems, data is entered by scanning an image, while in online systems, data is entered by utilizing a digitizer and an electronic pen to write on a screen. Scanning and subsequent processing take place regardless of whatever method is used. Online recognition has an advantage over offline recognition since the points of writing are kept as a function of time, and so the order of strokes helps to distinguish the letter from one another in the case of overwriting characters.

1.2 Motivation

Problems arise when attempting to decipher diverse people's handwriting styles, as well as character sizes and shapes. When composing a piece of content, some people may employ a variety of writing styles that the system may not be able to recognize. Overwriting of characters is a common occurrence, making it even more difficult to recognize them. It is also more difficult to train neural networks for recognition if the language has a large number of characters. Smartphones' limited processing power and memory capacity make it difficult to recognize many different characters.

In the discipline of pattern recognition, OHCR of Indic scripts is one of the most difficult problems. Because of the large character set, its complicated shapes, confusion among them, and varying writing styles, it is difficult to achieve a high identification accuracy. Despite these difficulties, handwriting is being used to communicate with computers. Tamil, Telugu , and Kannada are all examples of South Indian languages where OHCR systems have been developed.

There has not been a significant development of an OHCR system in Malayalam ,which is yet another South Indian Language, though isolated researches like Rahiman and Rajasree (2009),John et al. (2012),Nair et al. (2017),Kishna and Francis (2017),John et al. (2011),Manjusha et al. (2015) exists.

Figure 1: Malayalam Character Set

Malayalam is widely spoken in the state of Kerala, where it originated. It is spoken and written by more than 95 percent of the population of Kerala. Malayalam character recognition is more challenging than other Indian language scripts due to the sheer significant similarity of character shapes, the huge number of character classes, and the use of old and modern Malayalam script in writing.Because of its curved form, Malayalam script cannot be considered cursive. The script has a wide range of characters that include vowels, consonant letters, consonant signs, conjuncts, and Chillu characters. Syllables that begin with a vowel are written using independent vowels. Consonant clusters and single consonant symbols are the only bases on which dependent vowel signals can appear.Figure 7 shows the Malayalam Character set.In this research we have used data that consisted of 48 single letters of Malayalam script. On average, there were 127 handwritten script images in the dataset for each letter .

1.3 Research Question

This paper hopes to address the following research question:

- What is the impact of Few shot learning on Online Malayalam Handwritten Character Recognition compared to traditional Classification techniques?

1.4 Research Objective

The main objective of this research is to come up with a robust and minimalist approach to Online Malayalam Handwritten Character Recognition and convert recognized characters to corresponding Machine Readable format.

1.5 Research Outcome

The Few shot learning approach mentioned in this study was able to identify characters to a certain extent with minimum amount of time invested in training.The major contributions of this paper is as follows:

- Introduced a new Few shot learning architecture using Siamese Networks for recognizing Online Malayalam characters
- Introduced a robust CNN-RNN model for recognizing online Malayalam characters
- Introduced a robust CNN model for recognizing online Malayalam characters
- The performance of the proposed work is compared with traditional classification based models based on character recognition rates.

1.6 Report Structure

As for the remainder of the study, it is organized as follows: Section 2 highlights the most intriguing HCR studies, while Section 3 focuses on the Methodology used for Online Malayalam handwritten character recognition. Sections 4,5 and 6 will discuss the design, implementation, and evaluation of the project, respectively. Section 7 sums up the findings of the research reported in this paper in a concise manner.

2 Related Work

This section talks about the critical analysis of the works of different researchers who have worked in OHCR.It has been divided into mainly 3 different categories.

- Structural Features Based Approaches:This would include researches that takes into consideration the structural features of characters

- Neural Network Based Approaches:This section would talk about deep learning based classification approaches

- Hybrid Approaches:The research papers in this section would incorporate a mix of structural and statistical features along with deep learning .

2.1 Structural Features Based Approaches

2.1.1 Handwritten Malayalam Character Recognition using Regional Zoning and Structural Features

In James, Raveena and Saravanan (2018) ,a handwritten Malayalam character identification system based on extended regional zone based features and structural features is implemented in this study. A list of structural traits and the locations and zones that developed them like Length of character in horizontal,Length of character in vertical,Number of endpoints,Number of intersections in Horizontal ,Number of intersections in Vertical ,Number of loops,Direction of writing, Number of horizontal lines ,Number of vertical lines are taken into account. The language presented here combines a regional zone based method with structural feature extraction in conjunction with a list of structural features. a 126-by-126-pixel feature vector image.A decision tree classifier was used to predict on the unseen dataset.A 95.4 percent accuracy and 78.67 percent efficiency were achieved for a feature size of 126 and a feature vector of size 9. The Vowel recognition was 98.72 percent accurate, while consonant recognition was 95.68 percent accurate, for a feature vector of size 126, with some classification labels affecting the recognition.The author highlights that rate of recognition increases with larger feature vectors.

2.1.2 Character Recognition from Palm-leaf Manuscripts

In Athisayamani et al. (2020) the authors have implemented B-spline Curve recognition methodology in identifying vowels from historical documents.The author asserts that as handwritten text contains lot of curves B-spline representation of the letters can be very effective as it supports scaling,rotation and other feature transformations.The methodology used by the authors included binarization as the initial step for extracting the skeleton of the characters.A single pixel width representation of the character is obtained as the result . This was accomplished by utilizing the Hildich Algorithm which involved lowering the pixels at the character's edge,if the pixel's shortest distance was the same then it was positioned on the median axis. The next step was to represent each character as the B-spline curve.The results of the experiments revealed that characters with lesser number of curves were recognized more accurately.The method was compared with BFS graph based Tamil character recognition.The author concludes by highlighting the need for better predictive models for characters with more curves.

2.1.3 Structural representation-based off-line Tamil handwritten character recognition

The authors of Raj and Abirami (2020) used three different novel approaches for the structural representation of offline Tamil Handwritten characters.The first is the Strip Tree procedure which produced the required structure without changing the true form of the character, but if the structure contained extraneous loops or junction points, the number of features were bound to rise.The shape is discovered by Strip Tree using a rectangular configuration. It produced the same tree structure for varied forms.The position of the structure was revealed using two separate statistical methodologies: Z-ordering and PM-Quad tree approaches. The fundamental goal of using location-based feature extraction features was to recognize features based on their default properties.By using quad division and a hierarchical tree, the PM-Quad tree exposes the location.The value is located via Z-ordering during its N-order trip, N-order visits that had taken place from each midpoint of one grid to another. If the foreground pixel was visited throughout the voyage, the bit shuffling algorithm was used to determine its position.Finally, based on the structure and length of the tree, these formations were transformed into numerical values and fed into aa SVM model as input. SVM was utilized in a hierarchical fashion with a divide-and-conquer technique for classification of the characters. This combination, according to the author's final result analysis, was capable to handle all 100 distinct Tamil characters.The author states that the classification will be difficult when the level of complexity is great.The author asserts that this method is suitable for any language with curvy nature.

2.1.4 Efficient Handwritten Character Recognition of MODI Script Using Wavelet Transform and SVD

Wavelet Transform-Singular Value Decomposition(WT-SVD), a revolutionary approach for MODI script recognition, was proposed in Joseph and George (2021). The suggested model outperformed MODI Script character recognition's reported highest accuracy of 94.92 percent, achieving 99.56 percent. WT and SVD are used in a hybrid approach to implement the procedure. SVD is used to reduce the number of dimensions in the extracted features, resulting in a reduction in computational complexity. Classification relied on the Euclidean distance classifier.However the authors highlighted that the research in MODI script was required by focusing on segmentation of unconstrained MODI script documents in the future.

2.2 Neural Network Based Approaches

2.2.1 OCR System Framework for MODI Scripts using Data Augmentation and Convolutional NeuralNetwork

The research in Joseph and George (2021)used a CNN-based approach to recognize handwritten MODI script characters.On-the-fly data augmentation and CNN architecture were used to implement the proposed method called dubbed ACNN. The 45-degree flip of the horizontal and vertical axes is used to supplement the data.It was revealed that the data augmentation done on-the-fly allowed the network to have access to fresh data, which boosts the system's overall efficiency. The proposed method's CNN architecture was constructed with the right parameters in order to achieve the best possible recognition of MODI script characters. A 99.78 accuracy was reached after experimenting with various settings, such as the number of convolution layers, filter size, activation function type, pooling layer size and number of fully connected layers.

2.2.2 Bangla-Meitei Mayek scripts handwritten character recognition using Convolutional Neural Network

In Hazra et al. (2021) The authors followed a standard CNN architecture for identifying handwritten Bangla-Meitei Script. The architecture consisted of three convolutional layers followed by three fully connected layers connected to a Softmax layer.The author has taken a robust parameter estimation approach. The paper talks about the benchmarks of various configurations of the standard CNN model, like batch size, number of neurons, type of optimizer, etc. The author observed that the selection of hyper-parameters is a paramount task.It was found that the Adam optimizer worked better than all other optimizers for various batch sizes.The author concludes by stating that Deep NN, based on CNN, provides a more competent solution to a variety of real-world situations. The author has also laid emphasis on the recognition of compound characters in the future.

2.2.3 Optimally configured convolutional neural network for Tamil Handwritten Character Recognition by improved lion optimization model

The authors of Lincy and Gayathri (2021) have laid emphasis on research in pre-processing and recognition to construct a novel Tamil Handwritten Character Recognition method. With two key procedures, pre-processing and recognition, the authors of this research have devised a new TCR technique. Phases of the pre-processing included RGB to grayscale conversion, binarization with thresholding, morphological procedures, and linearization. Finally, the pre-processed image that was obtained at the end of the linearization process was used to recognize objects using an appropriately constructed CNN. A new SALA, an upgraded variant of the regular LA, was used to fine-tune the fully linked layer and weights of the CNN. On the basis of a variety of performance metrics, including positive and negative outcomes, the suggested work outperformed other current state-of-the-art models. In terms of TD = 50, the proposed model (CNN + SALA) is 12.7 percent, 3.4 percent, 7.2 percentage points, and 11.2 percentage points better than RNN, LA, EHO-NN, and DCNN.

2.2.4 HCR-Net: A deep learning based script independent handwritten character recognition network

In Chauhan et al. (2021) The authors presented a unique deep learning-based script-independent HCR network, which they termed HCR-Net. Transfer learning and image augmentation techniques were used in this HCR-Net to learn quicker, learn on smaller datasets, and produce better generalizations than typical deep learning systems, which require enormous amounts of data. In the proposed model, HCR-Net is initially initialized with a few layers of an already-trained VGG16 network, which employs a novel transfer learning approach. The authors showed that HCR-Net works well with datasets of Bangla, Punjabi, Hindi, and English. They also showed that the HCR-Net can be used to learn new languages. It has been found that the most common causes of miss-classification included poor handwriting, noisy datasets, and the similarity of distinct characters. The authors highlight that there is a major drawback to HCR-Net, which is that it does not perform well in languages that have a large number of classes. When it comes to languages with many classes, like Chinese, HCR-Net was going to be expanded in the future. The author also brings to notice that since the majority of the miss-classifications were based on the similarity of characters, this is another region for further investigation.

2.3 Hybrid Approaches

2.3.1 Recognition of online handwritten Gurmukhi characters using recurrent neural network classifier

In Singh et al. (2021) an Indian script known as Gurmukhi, which is used by more than 100 million people, was the focus of the research. In this study, the main difficulties in recognizing online hand-written Gurmukhi characters were explored.It was presented in this paper that a recurrent neural network may be used to recognize online handwritten fundamental Gurmukhi characters.For the purpose of performing the recognition task, the authors employed a stroke-level handwritten dataset .It consisted of 52,570 Gurmukhi words, written by 175 different people, to identify the 41 basic characters in Gurmukhi. To represent the 41 fundamental characters, a total of 81 stroke-classes were been discovered. The handwritten data was stored and annotated at the stroke-level. The authors took 150–170 average samples of each stroke-class to train the classifier . Training and testing experiments yielded a stroke classification accuracy of 98.67 and a test accuracy of 90.93, respectively.The results of this paper showed that Recurrent neural networks outperformed the other classifiers on the test dataset.

2.3.2 Multi-layer classification approach for online handwritten Gujarati character recognition

In Naik and Desai (2019)Online handwritten Gujarati character identification utilizing hybrid characteristics was proposed by the researchers. First layer convolutional feature maps were replaced by scattering transform-based feature maps in this research. Training data of roughly 2000 samples were used by the authors. The accuracy achieved was 94.13 percent and the average execution time per stroke was 0.103 s, according to the authors. SVM (polynomial) and SVM (linear) classifiers were used to compare the suggested multilayer classification technique.The authors then highlights that as long as there is a first layer classifier that doesn't work, the second layer classifier won't be able to give correct answers.

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

The study in Manjusha et al. (2018a) focuses on replacing the first layer's convolutional feature maps with scattering transform-based feature maps. The Malayalam handwritten characters were used to test a new hybrid CNN architecture (ScatCNN). A total of 85 Malayalam character classes from 77 people were used in the experiment. When compared to the same CNN architecture, ScatCNN was able to attain superior performance in terms of validation loss and recognition accuracy. ScatCNN's initial and converging loss levels were lower than CNN's. ScatCNN can outperform other handwriting databases in terms of recognition performance, according to experiments. For character images, the predefined scattering wavelet filters outperformed the self-taught filters in terms of generating invariant features.It was found that the proposed ScatCNN idea applied on Resnet achieved better recognition accuracy than an analogous Resnet architecture for the dataset without data augmentation.But the Resnet outperformed the ScatCNN in terms of performance when data augmentation was applied. The author concludes by stating that ScatCNN's hybrid architecture could benefit from the evaluation of data augmentation techniques which could be one of the possibilities for future research.

2.3.4 Reduced Scattering Representation for Malayalam Character Recognition

In Manjusha et al. (2018b) the same authors of Manjusha et al. (2018a) came up with an improvement of their research.In this research the scattering network employed predetermined wavelet filter coefficients instead of CNN's self-trained filters. Decompositions of wavelets were followed by modulus and averaging operations in the scattering network to yield feature descriptors. Character databases for the Malayalam language, MAL PrintedDB and MAL HandwrittenDB were used, contained 130 and 85 distinct character classes, respectively.In the second layer of the scattering network in Manjusha et al. (2018a) the scattering coefficients exploded, making higher-layer features unsuitable for use in a classifier. In order to reduce the dimensions of high-dimensional features, the SVD-based dimension reduction technique was used to transform them into lower-dimensional informative features. When paired with zeroth- and first-layer scattering coefficients and reduced representation of second-layer feature, the recognition accuracy was even more enhanced. Accuracy in MAL PrintedDB and MAL HandwrittenDB was recorded as 95.73 percent and 96.73 percent for the proposed scattering feature descriptor, respectively. The authors conclude by highlighting the scope for improvement by integrating scattering features with self-learned features in CNN.

2.3.5 A Novel Hybrid Approach for Feature Extraction in Malayalam Handwritten Character Recognition

In James, Sujala and Saravanan (2018) the authors highlight that due to the identical shapes of the characters, previous Malayalam handwritten recognition systems have consistently produced incorrect results. To reduce the likelihood of classification errors due to similar-looking characters, they proposed a hybrid approach (the SSF Method) that takes into consideration both statistical and structural properties. In Malayalam, structural characteristics include curves. Malayalam's use of curve elements gives the language a distinctive shape. Statistical factors like zoning and region-based zoning were utilized to improve accuracy and reduce miss classifications. The proposed model outperformed the available Malayalam character recognition systems in terms of accuracy (around 97 percent). The author clarifies that the method resulted in slight mis-classifications among selected characters. This was mostly due to the fact that people having different writing styles. Research conducted here demonstrates that datasets are critical to character recognition accuracy. The accuracy of character recognition was compared using roughly 2000 samples per character in this study. When compared to previous publications in Malayalam character recognition, the dataset used in this study was far more extensive. As a result, the more character samples used for training, the better the results were. The author concludes by stating that The lack of a consistent dataset is the fundamental issue in Malayalam character recognition . This challenge was solved by making the dataset used in this work available online so that researchers can use it to compare their own findings.

3 Methodology

- Dataset Augmentation:An augmentor tool freely available online was leveraged in the first phase of the implementation.Image distortion and rotation were used to create images with a wide range of variation.Each character had 500 images at the end of this step.

- Pre-Processing:The first step in Preprocessing was to convert the image to grayscale, then the images where binarized or converted into white and black pixels alone, and then skeletonization of the characters.As a result of skeletonization, the characters deteriorated and shrunk.The images were then cropped and resized to 32x32 pixels size.

- Training,Validation and Test Sets:Afterward, we had to construct the training, validation, and testing datasets with a ratio of 70 ,10 and 20 respectively.

- Model Evaluation:The model trained was evaluated on the test set.

- Online Prediction:The models after evaluation were utilized for making predictions and the quality of their predictions were assessed.

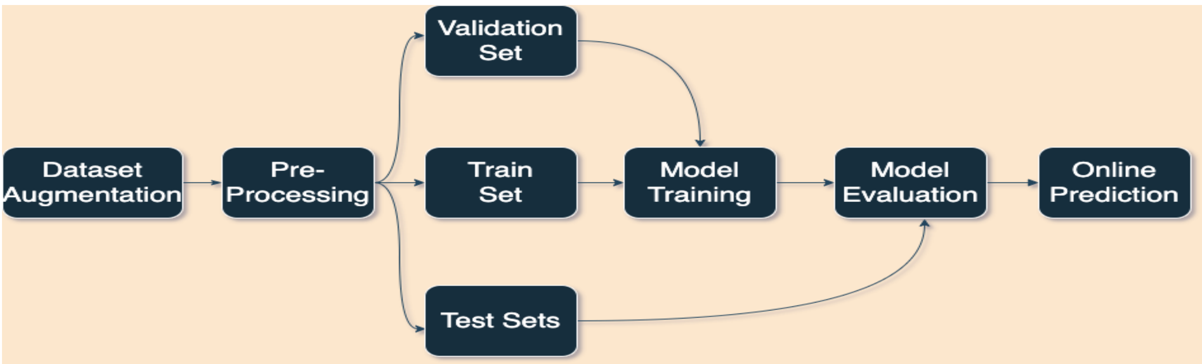


Figure 2: Methodology

4 Design Specification

This section details the method of Few shot Learning using Siamese networks which was utilized in addressing Online Malayalam Character Recognition.

4.1 Siamese Networks

The Siamese Network used here is inspired by the work of the authors of Koch et al. (2015).The conventional model used by the authors is a siamese convolutional neural network with L layers each having Nl units, where h1,l indicates the hidden vector in layer l for the first twin, and h2,l symbolizes the same for the second twin. In the first L 2 layers, we employ just rectified linear (ReLU) units and sigmoidal units in the rest of the layers. For each of the convolutional layers, we utilized a single channel and various-sized filters, with a stride of 1 as the fixed value. Optimized performance was achieved by specifying the number of convolutional filters as a multiple of 16. If desired, the network can optionally perform maxpooling on the output feature maps with a filter size and stride of 2 before applying the ReLU activation function. Thus, each layer's kth filter map is as follows:

$$a^{(k)}_{1,m} = max - pool(max(0, W^{(k)}_{l-1,l} * h_{1,l-1} + b_l), 2) \quad (1)$$

$$a^{(k)}_{2,m} = max - pool(max(0, W^{(k)}_{l-1,l} * h_{2,(l-1)} + b_l), 2) \quad (2)$$

Wl-1,l is a three-dimensional tensor that represents the feature maps for layer l. The * in the equation stands for valid convolutional operation that it returns only those output units that were the result of a total overlap across each convolutional filter and the input feature maps.

In this study, the last convolution layer of Koch et al. (2015) is replaced with a Bi-Directional LSTM layer. It is because of this that outputs from the last convolutional layer are reshaped so that they look like (batch size, H, W * channel).An LSTM layer generally outputs the cell state and the hidden state and also sequences of the hidden states at all timesteps.But in this research we wouldn't require the sequences.In this case, the Bi-Directional LSTM will output 2 cell states and 2 hidden states.Thus the feature maps from the Bi-LSTM is as follows:

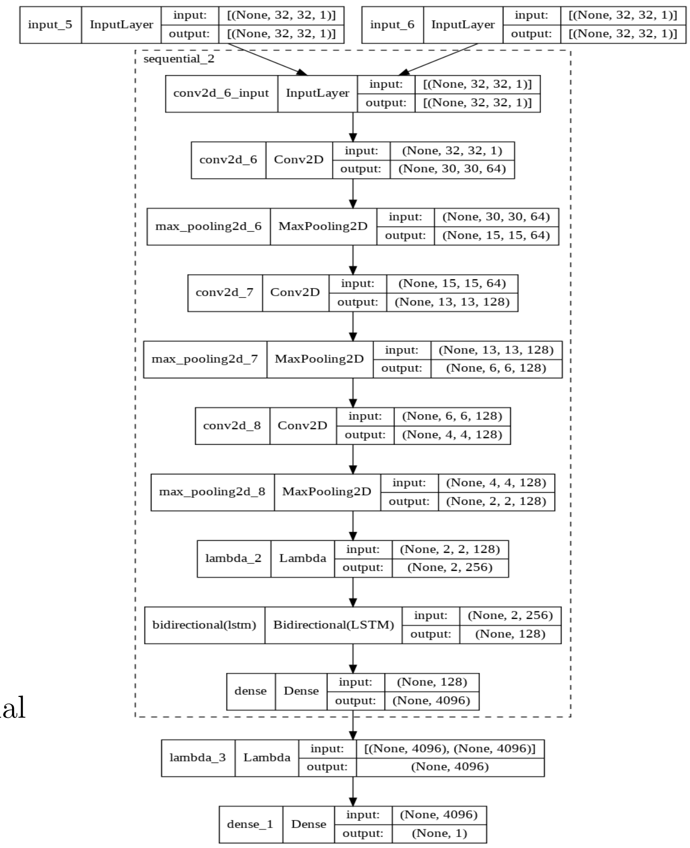


Figure 3: Siamese Network

$$f_{1t} = \sigma(W^T_{f1} * [x_{1t}, h_{1_{(t-1)}}] + b_{f1}) \quad (3)$$

$$i_{1t} = \sigma(W^T_{i1} * [x_{1t}, h_{1_{(t-1)}}] + b_{i1}) \quad (4)$$

$$o_{1t} = \sigma(W^T_{o1} * [x_{1t}, h_{1_{(t-1)}}] + b_{01}) \quad (5)$$

$$c_{1t} = f_{1t} * c_{1t-1} + i_{1t} * c_{1t} \quad (6)$$

$$h_{1t} = o_{1t} * tanh(c_{1t}) \quad (7)$$

where f1t corresponds to the forget gate, i1t to the input gate and o1t to the output gate,c1t to the current cell state ,h1t to the current hidden state of the first LSTM in the Bi-Directional LSTM layer.

Following this recurrent layer is a fully-connected layer. Then there is a third layer that calculates the mediated distance metric between each pair of siamese twins, which is passed on to a sigmoidal output unit.

$$P = \sigma(\sum_j \alpha_j |h_{1,L1}^{(j)} - h_{2,L1}^{(j)}|) \tag{8}$$

where sigma is the activation function with a sigmoidal shape.This last layer scores how similar two feature vectors are to each other based on how much they look like.The alpha j are extra parameters that the model learns while it is being trained.They help the model figure out how important the component-wise distance is.The final Lth layer for the network is fully connected .It connects the two siamese twins.The loss function used here is binary cross entropy.

In this study, all network weights that were initialized in the convolutional layers came from a normal distribution with zero-mean and standard deviation of 0.01 as mentioned in the paper Koch et al. (2015). The biases were also made from normal distribution, but they were made with mean 0.5 and standard deviation .01. Fully-connected layers had biases that were set up the same way as convolutional ones. The weights were drawn from a much broader normal distribution of zero-mean and standard deviation of 2 x 10-1.

4.2 Few Shot Learning

The idea behind Few-shot learning is inspired by the works of authors of Lake et al. (2011) in one-short learning.This section will demonstrate how the SiameseNet's learned features can be used at few-shot learning to show how well they can distinguish between different types of objects.

Assume we are provided a test image x, which is a column vector that we want to put into one of c different groups. Another set of images is also given to us: a set of column vectors that show examples from each of the c groups. We can now use x, xc as our input for a range of c = 1,..., C. Then, figure out which class has the most similarity from the below equation

$$C = argmax_c p^{(c)} \tag{9}$$

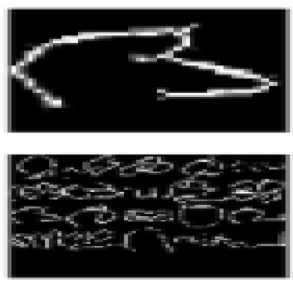


Figure 4: Sample Input

For the few shot classification task ,first, a letter from the evaluation set is chosen, along with twenty random characters. Then, a 20-way classification task is done. It also picks two of the twenty letters out of a pool of characters that are being used to test them. These two characters then make a sample of the twenty characters that they have. Each of the characters from the first drawer is called a "test image," and each of the twenty characters from the second drawer is individually compared to the test image. The goal is to figure out which class the test image belongs to from the evaluation subset's characters.

5 Implementation

This section will focus on the online prediction.The implementation was carried out in Python,using mainly tensorflow,keras and OpenCV libraries.

Since different kinds of models have been used,the prediction pipeline for each is different.Hence they have been divided into following categories:

- Histogram of Oriented Gradients and SVM Kernels
- Deep Neural Networks
- One Shot Prediction

5.1 Histogram of Oriented Gradients and SVM Kernels

The Histogram of Oriented Gradients (HOG), is a feature descriptor just like Canny Edge Detector and SIFT. Computer vision and image processing are two areas where it is used to identify and locate objects. Images are segmented into certain regions where gradient direction is counted. An object's structure or shape is a focus of the HOG description. It is Comparable to Edge Orientation Histograms and SIFT feature descriptors. Since it employs both the magnitude and angle of the gradient, it is superior to any other edge descriptor. It generates histograms based on the magnitude and orientation of the gradients in each section of the image.

The Specifications of the HOG feature descriptor for this project is as follows:

- orientations=4
- pixels per cell =2x2 kernel
- cells per block = 4x4 kernel

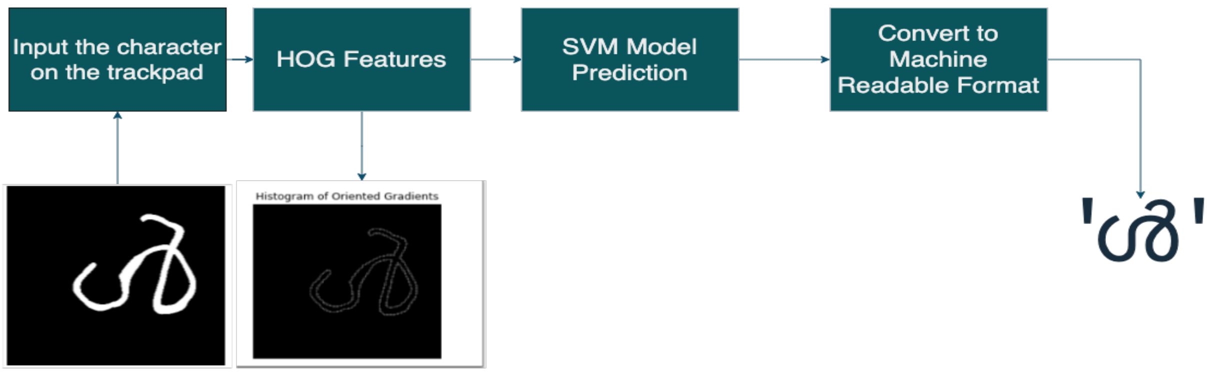


Figure 5: SVM prediction

The HOG features are then fed into two types of SVM models ,one that uses a Polynomial kernel and another that uses a Radial Bias Function (RBF).The predictions are then post processed to a Machine Readable format.

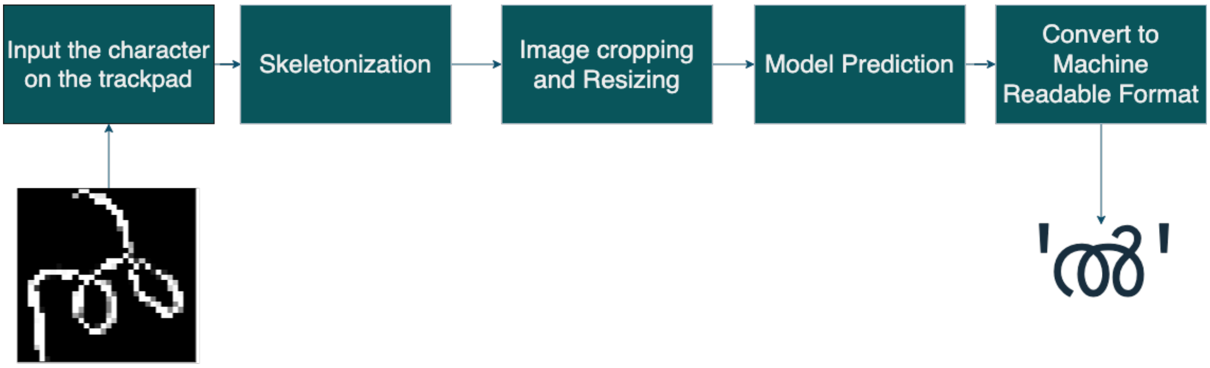


Figure 6: DNN prediction pipeline

5.2 Deep Neural Networks

This section will cover 6 different models.The pipeline for prediction is the same for all of them and all of them follow the pipeline shown in Figure 6.

5.2.1 Custom CNN model

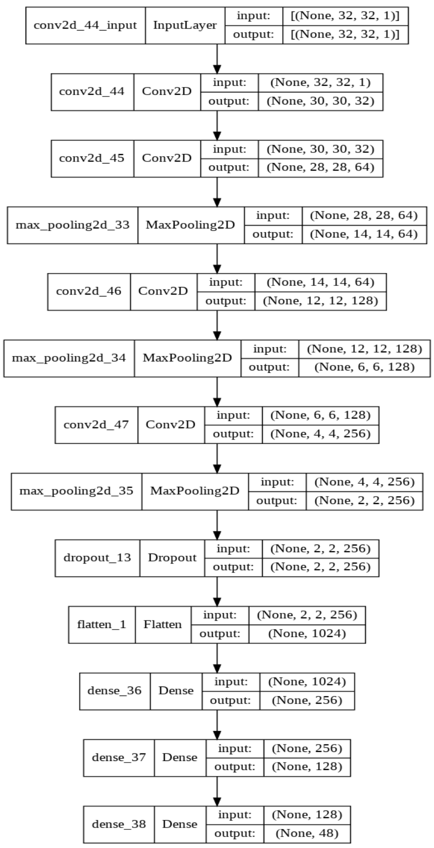


Figure 7: CNN model

The proposed CNN model's design specification consists of a Convolutional Layer with 32 hidden units, a kernel size of 3x3, and "Relu" activation making the first layer.Kernel size is 3x3, activation is "Relu," and there are 64 'hidden units' in Layer 2.In layer 3, Maxpool and a two-by-two stride are used.A Convolutional Layer with 128 hidden units, a kernel size of 3x3, and "Relu" activation make up the fourth layer.Maxpool and a 2x2 stride make up layer 5.In this layer, there are 256 hidden units, the kernel is 3x3 and the activation is "Relu".Max pool and a 2x2 stride make up layer 7.Dropout of 0.25 is the eighth layer.With 256 hidden units and a relu activation makes up layer 9 .Dense layer 10 has 128 hidden units with a relu activation and is likewise a dense layer.There are 48 hidden units and a soft-max activation in layer 12, which is a Dense layer and the output layer.

5.2.2 CNN-Bi RNN model

The CNN and Bi-Directional RNN model consists of 2 Bi-Directional layers,one is a Long Short Term Memory layer(LSTM) and the other a Gated Recurrent Unit layer(GRU).The idea behind using two Bi-Directional Layers is so that one Bi-Directional layer learns the features from one angle of an image and the other learns from a 90 degree rotated position of the same image.The Bi-Directional layers are followed by Maxpooling 1D layers,dropout layers and a Global Maxpooling Layer.The rotation of the images is performed using a custom lambda layer that alters the order of dimensions.The features from the RNN units are concatenated together and are further concatenated with the features that come from the CNN model.The architecture of the CNN part of this model is similar to that of the CNN model mentioned in the section above.

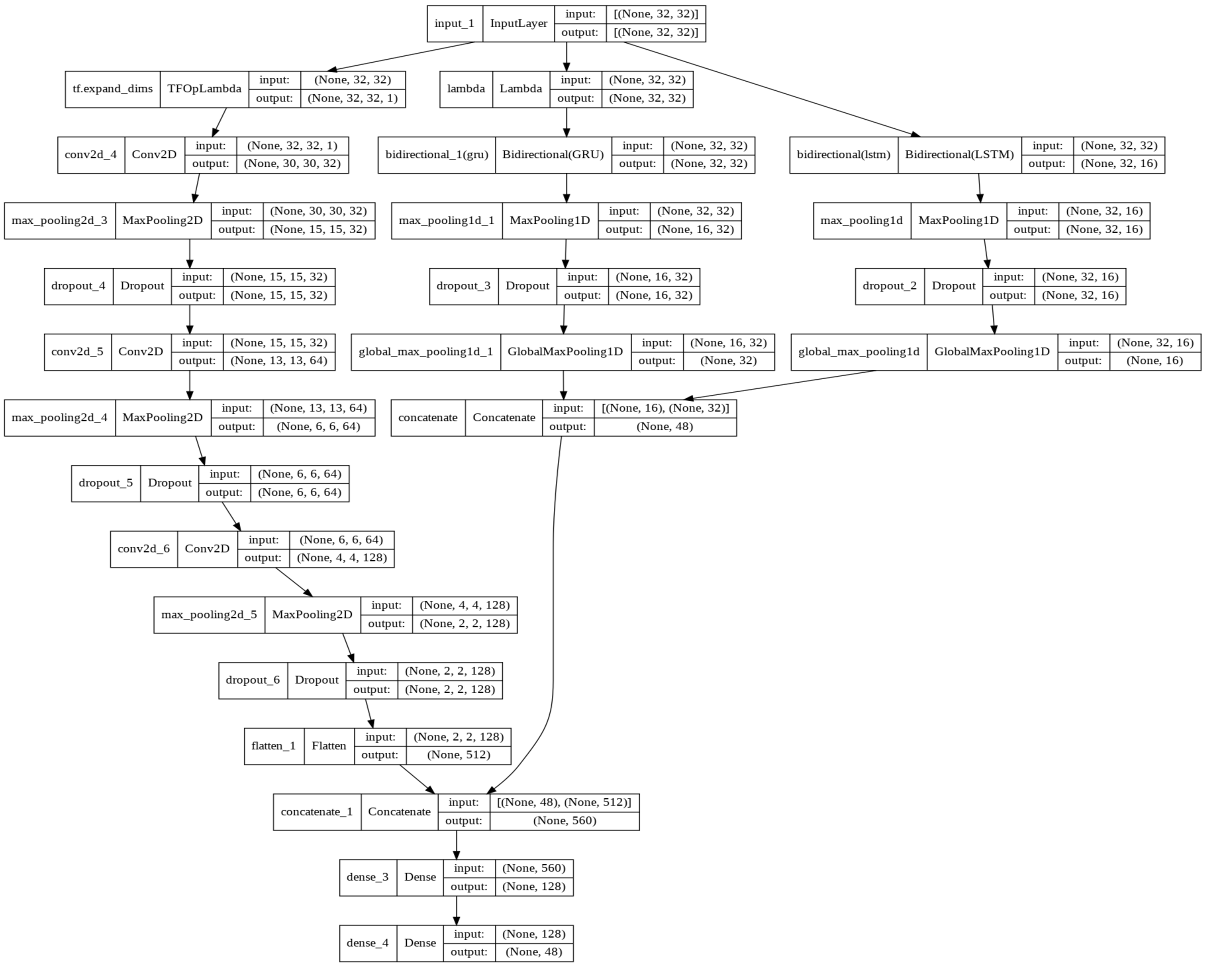


Figure 8: CNN-BiRNN model

5.2.3 Transfer Learning Models

When a machine learning model is pre-trained and can be applied to a new situation, it is known as transfer learning. The main advantage of using transfer learning is to avoid having to train numerous machine learning models for the same task from beginning.In areas of machine learning that need a lot of resources, such as image classification,this technique has been proven to be efficient.In this study CNN based transfer learning architectures such as VGG16,Resnet50,DenseNet and EfficientNet have been used.

5.3 One-Shot Prediction

The prediction pipeline using Siamese Networks is shown in the figure. The input character from the trackpad is used to create a pairs with 48 classes that are under consideration in this study.The superset as shown in the figure 8 will consist of all the 48 characters.The output of the siamese network will give the most similar pair.The letter with the highest similarity is passed on to front end as the predicted character.

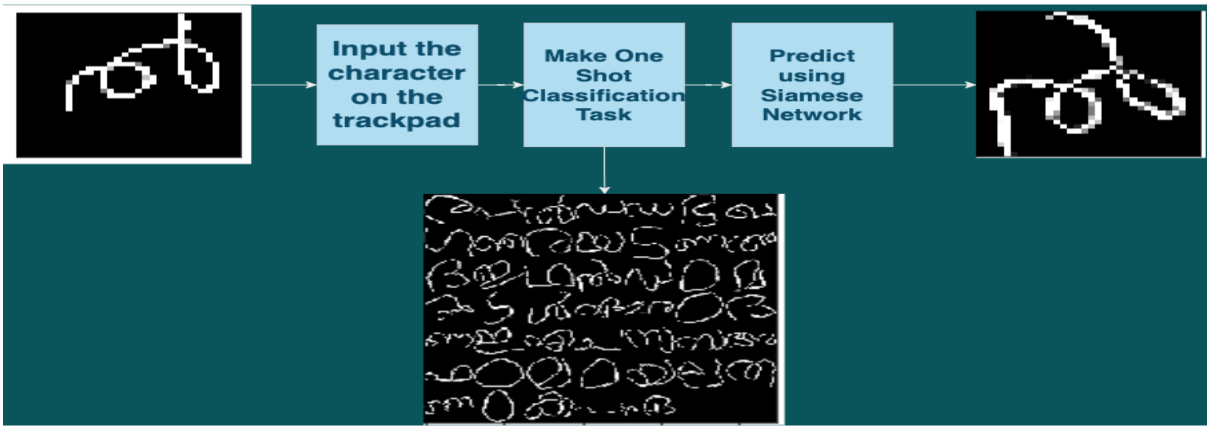


Figure 9: One Shot Prediction

6 Evaluation

The Training set consisted of 16800 images with an image size of 32x32.The validation set consisted of 2400 images and the testing set consisted of 4800 images.The evaluation metric under consideration is only accuracy which is also know as the recognition rate.This section is divided into four,section one compares SVM models,section 2 compares the deep learning based models ,section 3 compares the transfer learning models and section 4 compares the Few-Shot Learning approaches

6.1 HOG-SVM

After extracting Hog based Features, the SVM models were subject to a 5 fold cross validation.The average accuracy of the models in the training phase were 95 percent and 94 percent for Polynomial and RBF kernels respectively.It is evident from the table below that the HOG SVM models gave decent accuracy on the unseen test data and did not show any sign of overfitting.

Model	Training Accuracy	Testing Accuracy	Features
HOG-SVM-RBF	95	94.5	784
HOG-SVM-Poly	94	93.5	784

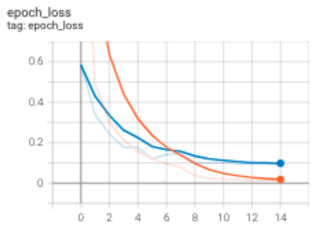
Table 1: SVM models evaluation results

6.2 Basic Deep Learning Models

From the results we can conclude that both the Neural Networks have beat the scores of the SVM based models.The CNN-BiRNN model turns out to be a more robust model

Model	Training Accuracy	Validation Accuracy	Testing Accuracy	Epochs	Parameters
CNN	98	97.29	97.52	15	689,328
CNN-BiRNN	97.74	97.3	97.29	20	178,096

Table 2: DNN models evaluation results

CNN 

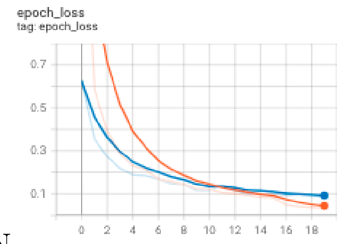
CNN-BiRNN 

Figure 10: Epoch vs Loss

with lesser number of parameters to train.It is also evident from Figure9 and Figure 10,which shows the epoch vs loss of training and test data, that the CNN-BiRNN model fits better whereas in CNN there are signs of slight overfitting.

6.3 Transfer Learning Models

Since the objective of this research was to find a minimalist solution and for the purpose of being comparable with the other models experiments concerning higher layers in transfer learning models where left out.

- For VGG 16, only the first 9 layers were retrained out of the 16 layers.
- Whereas in the Resnet50 ,the first 50 layers and all layers that were multiples of 33 out of 177 layers were trained.
- In EfficientNetB2,every 33rd layer until the layer 341 was enabled for training.
- In DenseNet ,every 9th layer until layer 300 was enabled for training.

Model	Training Accuracy	Validation Accuracy	Testing Accuracy	Epochs	Parameters
VGG16	99.23	98.04	98.08	20	874,864
Resnet50	97.74	97.3	97.89	20	607,744
EfficientNetB2	84.33	83.92	84.33	20	649,798
DenseNet	99.95	98.42	97.75	20	588,190

Table 3: Transfer Learning evaluation results

It can be inferred from this section that the DenseNet Architecture was the most robust with only 588K parameters.EfficientNet was the least performing model for 20 epochs.

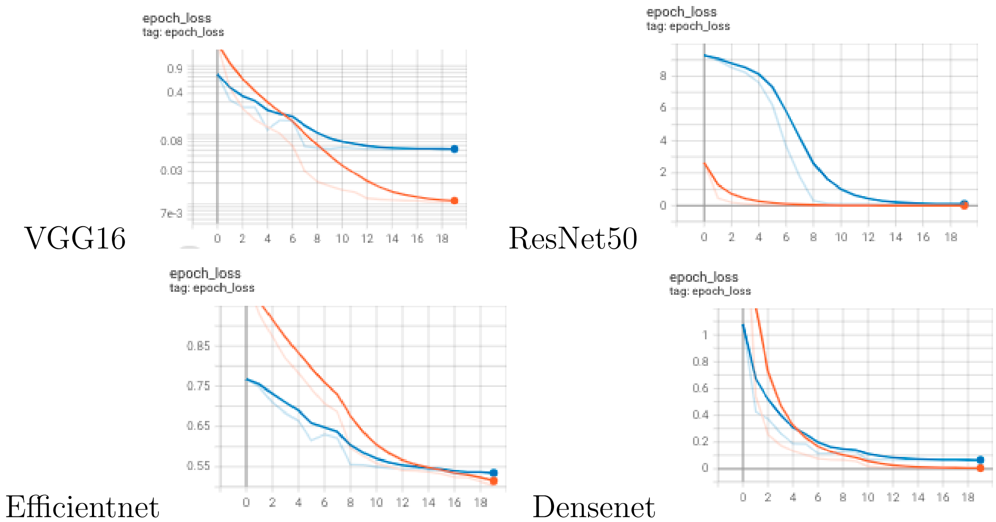


Figure 11: Epoch vs Loss

6.4 Few Shot Learning Experiment

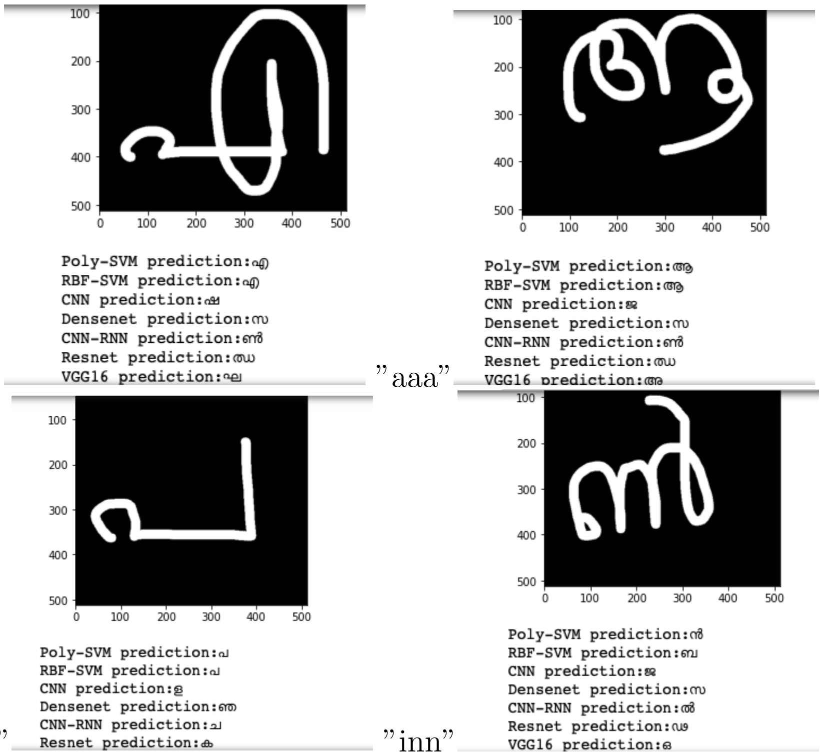
For the evaluations of Few shot learning experiment ,the proposed model was able to replicate the same results as that of Koch et al. (2015) which has a convolutional siamese network.The CNN-BiLSTM Siamese network proposed in this paper not only replicated the results but also achieved the same with lesser number of parameters to train.The experiment1 performed is such is the experiment that the authors of Koch et al. (2015) followed which was inspired by Lake et al. (2011).Experiment 2 is just different in the number of classes being considered for one-shot and the number of validations.

Model	Training Accuracy	Test Accuracy	Number of validations	No of classes	batch size	Iterations	Parameters
Koch et al. (2015)	100	100	550	20	1000	100	1,410,177
CNN-BiLSTM Siamese	100	100	550	20	1000	100	918,9136

Table 4: Siamese Network models experiment 1 evaluation results

Model	Training Accuracy	Test Accuracy	Number of validations	No of classes	batch size	Iterations	Parameters
Koch et al. (2015)	100	100	1600	100	1000	100	1,410,177
CNN-BiLSTM Siamese	100	100	1600	100	1000	100	918,9136

Table 5: Siamese Network models experiment 2 evaluation results



"Ae" "aaa" "paa" "inn"

Figure 12: Prediction for letters

6.5 Discussion

The classification models weren't able to identify most of the characters when it was input on the trackpad .This could be due to the absence of structural features like the number of loops,half-loops,small loops and the curvy nature of Malayalam Characters.The sample predictions are provided in Figure 12.

SVM models predicted with greater accuracy than any other models.The SVM models where able to predict the letters "paa" "Ae" and "aa" correctly.None of the models were able to predict the letter "inn" even though the polynomial SVM was close.

It was demonstrated in the previous part that the siamese networks were the best models in terms of quantitative performance based on their results. However, this is not the case, and it is a textbook example of overfitting. This could be due to a bias in the selection procedure for the superset, which could explain the results. As a result of the way the approach was designed, only one match will be found in the superset. As a result, there are more negative samples in a given superset as and the model sees images that are dissimilar most of the time. Even though this novel classification approach featured a greater number of parameters than old classification methods, it is important to note that it converged more quickly than traditional classification methods. Figure 13 illustrates an example of predictions for one-shot classification.

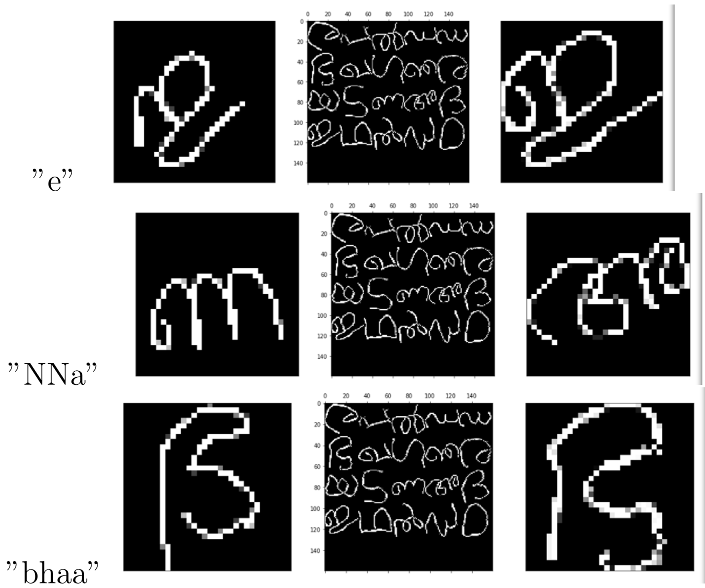


Figure 13: Prediction for letters by Oneshot classification

7 Conclusion and Future Work

The purpose of this study was to determine how well Few Shot Learning performed as compared to traditional classification models for Online Handwritten Malayalam Character Recognition (OHMCR). The fundamental purpose of this research is to provide a robust and minimalist solution for Online Malayalam Handwritten Character Recognition, as well as to convert the characters that are recognized into a machine-readable format.

In this study, it was demonstrated that Siamese Networks can be utilized to recognize online handwritten Malayalam characters, which was previously unproven.However, the impact it had was not significant.The selection bias introduced during the process of generating pairs of negative samples resulted in a significant overfitting of the model.However, Few shot learning in this field has been successful at recognizing characters with limited input and in a short time.Additionally, optimal models have been established in this study through the use of deep neural networks and transfer learning models.

When it came to measuring the quality of predictions, the SVM models in conjunction with the HOG features were found to be the most effective of the bunch.In Handwritten Character Recognition, structural features are a significant influence in the process.The incorporation of structural characteristics might aid in the training of better models.An improved method of picking characters for the superset in the Few shot classification system must be devised in order to be more efficient.To continue this research, it is necessary to integrate structural traits as a filtering step, such that the superset contains characters that are substantially similar to one another rather than letters that are widely off from one another.

References

Athisayamani, S., Singh, A. R. and Athithan, T. (2020). Recognition of ancient tamil palm leaf vowel characters in historical documents using b-spline curve recognition, *Procedia Computer Science* **171**: 2302–2309.

Chauhan, V. K., Singh, S. and Sharma, A. (2021). Hcr-net: A deep learning based script independent handwritten character recognition network, *arXiv preprint arXiv:2108.06663* .

Hazra, A., Choudhary, P., Inunganbi, S. and Adhikari, M. (2021). Bangla-meitei mayek scripts handwritten character recognition using convolutional neural network, *Applied Intelligence* **51**(4): 2291–2311.

James, A., Raveena, P. and Saravanan, C. (2018). Handwritten malayalam character recognition using regional zoning and structural features, *International Journal of Engineering & Technology* **7**(4): 4629–4636.

James, A., Sujala, K. and Saravanan, C. (2018). A novel hybrid approach for feature extraction in malayalam handwritten character recognition., *Journal of Theoretical & Applied Information Technology* **96**(13).

John, J., Pramod, K. and Balakrishnan, K. (2011). Offline handwritten malayalam character recognition based on chain code histogram, *2011 International Conference on Emerging Trends in Electrical and Computer Technology*, IEEE, pp. 736–741.

John, J., Pramod, K. and Balakrishnan, K. (2012). Unconstrained handwritten malayalam character recognition using wavelet transform and support vector machine classifier, *Procedia Engineering* **30**: 598–605.

Joseph, S. and George, J. (2021). Efficient handwritten character recognition of modi script using wavelet transform and svd, *Data Science and Security*, Springer, pp. 227–233.

Kishna, N. T. and Francis, S. (2017). Intelligent tool for malayalam cursive handwritten character recognition using artificial neural network and hidden markov model, *2017 International Conference on Inventive Computing and Informatics (ICICI)*, IEEE, pp. 595–598.

Koch, G., Zemel, R., Salakhutdinov, R. et al. (2015). Siamese neural networks for one-shot image recognition, *ICML deep learning workshop*, Vol. 2, Lille.

Lake, B., Salakhutdinov, R., Gross, J. and Tenenbaum, J. (2011). One shot learning of simple visual concepts, *Proceedings of the annual meeting of the cognitive science society*, Vol. 33.

Lincy, R. B. and Gayathri, R. (2021). Optimally configured convolutional neural network for tamil handwritten character recognition by improved lion optimization model, *Multimedia Tools and Applications* **80**(4): 5917–5943.

Manjusha, K., Anand Kumar, M. and Soman, K. (2015). Experimental analysis on character recognition using singular value decomposition and random projection, *International Journal of Engineering and Technology* **7**(4): 1246–1255.

Manjusha, K., Kumar, M. A. and Soman, K. (2018a). Integrating scattering feature maps with convolutional neural networks for malayalam handwritten character recognition, *International Journal on Document Analysis and Recognition (IJDAR)* **21**(3): 187–198.

Manjusha, K., Kumar, M. A. and Soman, K. (2018b). Reduced scattering representation for malayalam character recognition, *Arabian Journal for Science and Engineering* **43**(8): 4315–4326.

Naik, V. A. and Desai, A. A. (2019). Multi-layer classification approach for online handwritten gujarati character recognition, *Computational Intelligence: Theories, Applications and Future Directions-Volume II*, Springer, pp. 595–606.

Nair, P. P., James, A. and Saravanan, C. (2017). Malayalam handwritten character recognition using convolutional neural network, *2017 International conference on inventive communication and computational technologies (ICICCT)*, IEEE, pp. 278–281.

Rahiman, M. A. and Rajasree, M. (2009). Printed malayalam character recognition using back-propagation neural networks, *2009 IEEE International Advance Computing Conference*, IEEE, pp. 197–201.

Raj, M. A. R. and Abirami, S. (2020). Structural representation-based off-line tamil handwritten character recognition, *Soft Computing* **24**(2): 1447–1472.

Singh, H., Sharma, R. K., Singh, V. and Kumar, M. (2021). Recognition of online handwritten gurmukhi characters using recurrent neural network classifier, *Soft Computing* **25**(8): 6329–6338.