

# **Automated Evaluator of Handwritten Malayalam Answer Scripts**

*A Project Report*

*Submitted to the APJ Abdul Kalam Technological University  
in partial fulfillment of requirements for the award of degree*

***Bachelor of Technology***

*in*

***Computer Science and Engineering***

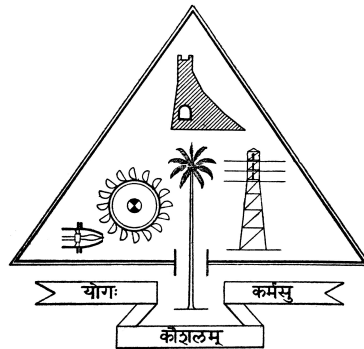
*by*

**Devi Krishna M K(TCR19CS026)**

**Maria Viji George(TCR19CS039)**

**Navneeth Variar(TCR19CS047)**

**Niranjan Neelakantan(TCR19CS049)**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING  
GOVERNMENT ENGINEERING COLLEGE THRISSUR  
KERALA  
December 2022**

**DEPT. OF COMPUTER SCIENCE & ENGINEERING GOVERNMENT  
ENGINEERING COLLEGE THRISSUR  
2022 - 23**

**CERTIFICATE**

This is to certify that the report entitled **Automated Evaluator of Handwritten Malayalam Answer Scripts** submitted by **Devi Krishna M K** (TCR19CS026), **Maria Viji George** (TCR19CS039), **Navneeth Variar** (TCR19CS047) & **Niranjan Nee-lakantan** (TCR19CS049) to the APJ Abdul Kalam Technological University in partial fulfillment of the B.Tech. degree in Computer Science and Engineering is a bonafide record of the project work carried out by him under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

**Prof. Princy Ann Thomas**  
(Project Guide and Project Coordinator)  
Assistant Professor  
Dept.of CSE  
Government Engineering College  
Thrissur

**Prof. Valsaraj K S**  
(Project Coordinator)  
Associate Professor  
Dept.of CSE  
Government Engineering College  
Thrissur

**Dr. Shibily Joseph**  
Associate Professor and Head  
Dept.of CSE  
Government Engineering College  
Thrissur

## DECLARATION

I, on behalf of authors of the report: Devi Krishna M K (TCR19CS026), Maria Viji George (TCR19CS039), Navneeth Variar (TCR19CS047), Niranjana Neelakantan (TCR19CS049), hereby declare that the project report **Automated Evaluator of Handwritten Malayalam Answer Scripts**, submitted for partial fulfillment of the requirements for the award of the degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by us under supervision of Prof. Princy Ann Thomas, Department of Computer Science and Engineering, Government Engineering College Thrissur .

This submission represents our ideas in our own words and where ideas or words of others have been included, we have adequately and accurately cited and referenced the original sources.

We also declare that we have adhered to the ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. We understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed as the basis for the award of any degree, diploma, or similar title of any other University.

Thrissur

12-12-2022

**Niranjana Neelakantan**

# Abstract

Teachers often spend a lot of time evaluating answer papers which is mostly a repetitive task that can be automated if a model is well-trained. Object character recognition is a main part of auto-evaluation when it comes to handwritten answers. There are many auto-evaluation systems in English, even for handwritten input. But regional languages lack such facilities. Even a decent Optical Character Recognition system can be found missing when it comes to regional languages like Malayalam. In light of these concerns, we propose ””, an automated evaluator for handwritten Malayalam answer scripts. The focus of the project is to create a system that can identify characters in Malayalam and evaluate basic spell-checking tests at the kindergarten level, which can be later scaled up. This is a system made for all teachers evaluating spell-checking tests in Malayalam.

# Acknowledgement

We take this opportunity to express our deepest sense of gratitude and sincere thanks to everyone who helped us to complete this work successfully. We express our sincere thanks to Dr. Shibily Joseph, Head of Department, Computer Science and Engineering, Government Engineering College Thrissur for providing us with all the necessary facilities and support.

We would like to express our sincere gratitude to the Prof. Valsaraj K S, Department of Computer Science and Engineering, Government Engineering College Thrissur for the support and co-operation.

We would like to place on record our sincere gratitude to our project guide Prof. Princy Ann Thomas, Assistant Professor, Department of Computer Science and Engineering, Government Engineering College Thrissur for the guidance and mentorship throughout this work.

Finally, we thank my family, and friends who contributed to the successful fulfillment of this project work.

**Devi Krishna M K**  
**Maria Viji George**  
**Navneeth Variar**  
**Niranjana Neelakantan**

# Contents

<b>Abstract</b>	<b>i</b>
<b>Acknowledgement</b>	<b>ii</b>
<b>List of Figures</b>	<b>v</b>
<b>List of Tables</b>	<b>vi</b>
<b>List of Symbols</b>	<b>vii</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Literature Review</b>	<b>3</b>
2.1 Optical Character Recognition of Hand-written Devanagiri Scripts using Machine Learning Techniques . . . . .	3
2.2 Study of different methods for Malayalam handwritten character recognition using Optical Character Recognition. . . . .	9
2.3 Recognition of handwritten digits using various machine learning approaches. . . . .	12
2.4 Summarization of Malayalam text using natural language processing techniques. . . . .	15
<b>3 System Development</b>	<b>17</b>
3.1 System Architecture . . . . .	17
3.1.1 Architectural Design . . . . .	17
3.1.2 Decomposition Description . . . . .	18
3.2 Data Design . . . . .	20

3.2.1	Data Description . . . . .	20
3.2.2	Data Dictionary . . . . .	20
3.3	Human Interface Design . . . . .	21
3.3.1	Overview of User Interface . . . . .	21
3.3.2	Screen Images . . . . .	21
3.3.3	Screen Objects and Actions . . . . .	23
<b>4</b>	<b>Results and Discussion</b>	<b>24</b>
4.1	Results of Literature Suvey . . . . .	24
4.1.1	Comparing Handwritten Devanagiri Script recognition . . . . .	24
4.1.2	Comparing Handwritten Malayalam Character Recognition . . . . .	25
4.1.3	Comparison of Handwritten Digit Recognition . . . . .	26
4.1.4	Inferences from different Text Summarization Approaches . . . . .	29
<b>5</b>	<b>Conclusion</b>	<b>31</b>
5.1	ADVANTAGES OF OUR WEBSITE . . . . .	31
5.2	LIMITATIONS AND FUTURE SCOPE . . . . .	32
	<b>References</b>	<b>33</b>

# List of Figures

3.1	Architectural Design . . . . .	17
3.2	Data Flow Diagram Lvl 0 . . . . .	19
3.3	Data Flow Diagram Lvl 1 . . . . .	20
3.4	Login Screen User Interface . . . . .	22
3.5	Output User Interface . . . . .	23
4.1	Improved accuracy with iterations . . . . .	27
4.2	SVC with different regularization parameter and different kernels . . .	28
4.3	Performance of KNN and RFC . . . . .	28



# List of Tables

2.1	Literature Survey on Optical Character Recognition of Hand-written Devanagiri Scripts using Machine Learning Techniques . . . . .	8
4.1	Comparative study of Deep Learning algorithms for Devanagari Character Recognition . . . . .	25
4.2	Comparison of different techniques for Malayalam HCR . . . . .	26
4.3	Table showing variation of Accuracy with number of iterations . . . . .	27
4.4	Comparison of KNN, SVM, RFC and CNN . . . . .	28
4.5	Comparison of KNN, SVM, BP and CNN . . . . .	28
4.6	Comparison of SVM and RFC . . . . .	29
4.7	Recognition rates of various machine learning algorithms . . . . .	29
4.8	Results of Abstractive Approach . . . . .	30
4.9	Results of Extractive Approach . . . . .	30

# List of Symbols

$\Omega$  Unit of Resistance

$\varepsilon'$  Real part of dielectric constant

$c$  Speed of light

$\lambda$  Wavelength

$\delta$  Delta

# Chapter 1

## Introduction

Handwriting recognition has been one of the most enchanting and demanding research areas in today's digitalized world, which has evolved through the combination of artificial intelligence and machine learning. The initial approaches of solving handwriting recognition involved Machine Learning methods like Hidden Markov Models(HMM), SVM, etc. Once the initial text is pre-processed, feature extraction is performed to identify key information such as loops, inflection points, aspect ratio, etc. of an individual character. These generated features are now fed to a classifier say HMM to get the results. The performance of machine learning models is pretty limited due to the manual feature extraction phase and their limited capacity for learning. Feature extraction step varies for every individual language and hence is not scalable. With the advent of deep learning came tremendous improvements in the accuracy of handwriting recognition. Offline recognition of handwritten text is one of the most challenging research areas due to the lack of temporal information as available in the online data and the large variations encountered in the writing style of different writers. Even though the OCR research is well advanced for foreign languages, the research on Indic scripts, especially South Indian languages is still in the infancy stage. Among the South Indian languages, the recognition of Malayalam scripts poses an even greater challenge due to the extremely large character set, highly similar writing style of the characters, complex curved features of characters, presence of compound characters. Malayalam is one of the 4 major Dravidian languages of South India. The basic character set of Malayalam consists of 15 vowels and 36 consonants. Apart from these

basic characters, the script consists of other vowel modifiers, conjunct consonants etc which together with the basic characters constitute the complete Malayalam character set consisting of 128 characters.

# Chapter 2

## Literature Review

The literature survey was done in four stages based on the individual seminar topics of the team members. The four different aspects based on which the literature survey is conducted are the following:

- Optical Character Recognition of Hand-written Devanagiri Scripts using Machine Learning Techniques.
- Study of different methods for Malayalam handwritten character recognition using Optical Character Recognition.
- Recognition of handwritten digits using various machine learning approaches.
- Summarization of Malayalam text using natural language processing techniques.

### **2.1 Optical Character Recognition of Hand-written Devanagiri Scripts using Machine Learning Techniques**

The purpose of the literature survey on Optical Character Recognition of Hand-written Devanagiri Scripts using Machine Learning Techniques is to provide detailed insight into different machine-learning techniques for optical character recognition of handwritten Devanagiri scripts. Devanagari handwritten character recognition system is based on deep learning technique, which manages the recognition of Devanagari

script. Depending upon the dataset and accuracy of each character the techniques differ.

<b>Title</b>	<b>Author</b>	<b>Year of Publication</b>	<b>Abstract</b>
Machine Learning Algorithms for Handwritten Devanagari Character Recognition: A Systematic Review	Mimansha Agrawal, Bhanu Chauhan, Tanisha Agrawal	2022	In this survey, a deep learning-based Handwritten Devanagari Character Recognition concept is chosen. There are approximately 42 papers selected interrelated to the survey. The effectiveness of this survey is analyzed and compared using various parameters and algorithms. In this section, the different kinds of algorithms, and methods, based on deep learning in Character Recognition concept papers are discussed effectively.

A Survey on Optical Character Recognition for Handwritten Devanagari Script Using Deep Learning	Pragati Hirugadea, Nidhi Suryavanshia, Radhika Bhagwata, Smita Rajputa, Rutwija Phadkea	2022	This paper gives a comprehensive review of the deep learning methods as well as some transfer learning techniques used for Handwritten Character Recognition for Devanagari script. The paper also reports the benefits and limitations of each method along with the challenges that need to be addressed to have an efficient and accurate Devanagari handwritten character recognition system.
---	---	------	---

Digitization of handwritten Devanagari text using CNN transfer learning – A better customer service support	Sandeep Dwarkanath Pande, Pramod Pandurang Jadhav, Rahul Joshi, Amol Dattatray Sawant, Vaibhav Muddebihalkar, Suresh Rathod, Madhuri Navnath Gurav, Soumitra Das	2022	This work employs the best-suited techniques that are useful to enhance the recognition rate and configures a Convolutional Neural Network (CNN) for effective Devanagari handwritten text recognition (DHTR). This approach uses the Devanagari handwritten character dataset (DHCD) which is a vigorous open dataset with 46 classes of Devanagari characters and each of these classes has two thousand different images. After recognition, conflict resolution is subtle for effective recognition therefore, this approach provides an arrangement for the user to handle the conflicts.
---	--	------	--



Comparison between Neural Network and Support Vector Machine in Optical Character Recognition	Michael Reynaldo Phangtriastua, Jeklin Harefaa, Dian Felita Tanoto	2017	<p>This paper uses several techniques as a comparison for some extracted features, such as zoning algorithm, projection profile, Histogram of Oriented Gradients (HOG) and combination of those feature extractions ( zoning + projection, projection + HOG, zoning + HOG, zoning + projection + HOG). For the evaluation of the proposed system, this paper compares the most common classifiers: Support Vector Machine (SVM) and Artificial Neural Networks (ANN). This experiment achieves the highest accuracy of 94.43% using a Support Vector Machine (SVM) classifier with the feature extraction algorithms are: projection profile and the combination of zoning + projection.</p>
---	--	------	--

A Machine Learning and Deep Learning Approach for Recognizing Handwritten Digits	Ayushi Sharma, Harshit Bhardwaj, Arpit Bhardwaj, Aditi Sakalle, Divya Acharya, and Wubshet Ibrahim	2022	Machine Learning and Deep Learning algorithms are used in this project to measure the accuracy of handwritten displays of letters and numbers. Also, the classification accuracy and comparison between them are shown. The results showed that the CNN classifier achieved the highest classification accuracy of 98.83%
Handwritten Digits Recognition using Novel Long Short Term Memory with Enhanced FMeasures Over K-Nearest Neighbour to Improve the Accuracy	Shivam Sangam, T. Rajesh Kumar	2022	The major goal of this research is to develop a model that can recognize digits utilizing Long Short-Term Memory and LSTM cells, as well as to compare F scores for optical character recognition using LSTM and KNN on the Modified National Institute of Standards and Technology dataset. The results showed that LSTM with LSTM cells performed substantially better than KNN in optical character recognition

Table 2.1: Literature Survey on Optical Character Recognition of Hand-written Devanagiri Scripts using Machine Learning Techniques

## **2.2 Study of different methods for Malayalam handwritten character recognition using Optical Character Recognition.**

The purpose of this section is to provide a detailed insight into different systems that use different techniques for feature extraction and classification in recognizing handwritten Malayalam characters. The concepts involved in the techniques used for feature extraction and classification are also studied in detail. The accuracy obtained using each of the techniques is also noted.

### **Recognition based on Chain Code Histogram**

The system uses chain code for feature extraction. Chain codes are used to represent the boundary of an image by a connected sequence of straight line segments of specified length and direction. In 4 connectivity, each of the 4 directions are given a number. In 8 connectivity, each of the 8 directions are given a number. For each image, we can generate a chain code that will uniquely identify the image. This property is utilised for feature extraction. For each character, we generate the chain code by starting from the left lowermost point and going in clockwise direction till the starting point is reached. The chain code is used to calculate the chain code histogram of the image. CCH is a scale invariant shape descriptor which is used to calculate the feature vector. After feature extraction, classification is done using a 2 layer feed forward neural network with sigmoid activation function. From the data samples, 70% is used for training, 30% used for testing. Mean squared error is used as the performance measure. The accuracy obtained using this approach was approximately 72.1%.

### **Multiple Classifier System**

In this system, 2 features are used. Gradient features and density features. Gradient features at each pixel position points in the direction of the greatest rate of change of intensity. Gradient features are used to represent the local characteristics of an image. Gradient feature in this system is found out using Sobel operator along both x and y direction.

- Gradient along vertical direction is given by the formula -  $f(i-1,j+1)+2f(i,j+1)+f(i+1,j+1)-f(i-1,j-1)-2f(i,j-1)-f(i+1,j-1)$ .
- Gradient along the horizontal direction is given by the formula -  $f(i-1,j-1)+2f(i-1,j)+f(i-1,j+1)-f(i+1,j-1)-2f(i+1,j)-f(i+1,j+1)$
- Gradient direction is found out by the formula -  $\tan^{-1}(G_v/G_h)$

. Density feature is another feature used for feature extraction.

- The pixel density is given by the formula -  $D(i) = \text{No of foreground pixels in zone } i / \text{Total no of pixels in zone } i$ .
- The image is divided into 4x4 zones and for each zone, the density feature is calculated.

.The two features are given to 2 feed forward neural networks. These neural networks are trained using resilient back propogation algorithm. The different accuracies are improved by combining those results using 4 schemes.

- Max rule - selects the class with the maximum confidence value among both classifiers as output.
- Sum rule - sums up the confidence values for each class and selects the one with the highest sum as the recognition result
- Product rule - multiplies the confidence values for each class and selects the one with the highest value.
- Borda count method - the classes are sorted in descending order of their Borda count values and the class with the highest Borda count value is selected as the output class. The Borda count for a class c is the sum of the number of classes ranked below the class by each classifier.

The datasets are split in the ratio 80:20. Maximum accuracy is obtained when the product rule is used.

## Convolutional Neural Network

CNN is one of the most popular Deep Neural Networks. CNN has given the best possible performance in many machine-learning problems. CNN has been used in different areas as pattern recognition, image processing, and voice recognition. In CNN, the number of parameters used is less and parameter sharing is also present [1]. In CNN feature extraction of an image is automatically performed when the input propagates through deeper layers. The different layers in CNN are:

- Convolutional layer is the layer where convolution operation occurs that is the same as image processing. A filter of the same row and column or square size matrix is taken and multiplied across the window that fits the filter. The element-wise product is done and then the summation is done. The concept of stride is generally used, as how much pixel shifts after doing one convolution. Here more the number of filters, the more accuracy can be achieved but computational complexity increases.
- Max-Pooling layer takes some pixels from previous layers. Pool size is defined and then that pool size is used on input pixels. The pool matrix is moved over the entire input and the max value within the overlapped input is taken.
- Dropout layer is mainly used to avoid overfitting. This layer randomly cuts the unnecessary connection between two neurons of different layers
- Flatten layer is one where multiple-sized input is converted into a 1D vector.
- Dense layer is used to do classification after doing the whole convolution process.

The model is trained using a dataset having over 90,000 images of 44 malayalam characters. After successful training, the model is deployed in such a way that any user can input real time images of Malayalam handwritten script and see the results. An accuracy of 97.26% is obtained using CNN. The model is implemented in real time using Keras with Tensorflow as the backend and OpenCV for image acquisition.

## **Recognition using HLH intensity patterns**

Here an algorithm is proposed which can accept the scanned image of handwritten characters as input to produce the editable Malayalam characters in a predefined format as output without applying any resizing or skeletonization methods. Characters are grouped into different classes based on their HLH intensity patterns. These patterns are separated from the image and fed for recognition. The algorithm is tested for 4 sets of samples ranging from 661 letters in the noiseless environment and produces an accuracy of 88%. This work separates the entire character set into three different classes. Ra-type characters, Pa-type characters, and Special symbols. This classification is based on the shape and appearance of the character. This shape feature is extracted to recognize the letter. This method employs recognition of isolated handwritten characters in a noiseless environment. The basic principle is to identify specific terminologies in each character and extend the same to a set of characters in order to achieve accurate results with very low-complexity algorithms. The separation letters are shown in figure 1 which uses intensity variations for segregating the line and character from the scanned image.

## **2.3 Recognition of handwritten digits using various machine learning approaches.**

The literature survey on the Study of different methods for Malayalam handwritten character recognition using Optical Character Recognition discusses the cumulative results obtained from the research of different papers that conducted the performance evaluation of different algorithms on HDR. Even though the papers are different, since every research is carried out on the same data set, it can be accumulated and presented as a single comparison table. Below is a brief description of each algorithm and their respective performance result from each reference.

Sl. No.	Title	Author	Results
1.	Handwritten Digits Recognition Using SVM, KNN, RF and Deep Learning Neural Networks	Yevhen Chychkarova, Anastasiia Serhiiienkob, Iryna Syrmamiikha, Anatolii Karginc	This article compares the performances of SVM, KNN, RFC and CNN. For handwritten digit recognition, the best recognition accuracy is provided by a convolutional neural network, as 97.6% accuracy. After building recognition models using all the algorithms mentioned above, the recognition accuracy of all handwritten digits on the test program turned out to be within 98-100%
2.	Handwritten Digit Recognition with Feed-Forward Multi-Layer Perceptron and Convolutional Neural Network Architectures	Harikrishnan A, Sourabh Sethi , Rashi Pandey	This article compares the performances of MLP and CNN on the MNIST dataset at different iterations. In the best case, MLP showed an accuracy of 97.73% and CNN showed an accuracy of 99.05%.
3.	Analysis of machine learning algorithms for character recognition: a case study on handwritten digit recognition	Owais Mujtaba Khanday, Dr. Samad Dadvandipour	This paper covers the work done in handwritten digit recognition and the various classifiers that have been developed. Methods like MLP, SVM, Bayesian networks, and RF were discussed with their accuracy and are empirically evaluated. A variation of CNN called Boosted Letnet 4 showed the maximum accuracy of 99.3% with other CNNs just behind, then followed by other algorithms.

4.	Handwritten Digit Recognition	Priyanshu Singh, Pranali Pawar, Nikhil Raj	The purpose of this project is to use the classification algorithm to identify handwritten digits. Background results are probably the most widely used Machine Learning Algorithms such as SVM, KNN and RFC and in-depth reading calculations like CNN multilayer using Keras and Theano and Tensorflow. Using these, 98.70% accuracy was used by CNN (Keras + Theano) compared to 97.91% using SVM, 96.67% using KNN, 96.89% using RFC was obtained
5.	Comparisons on KNN, SVM, BP and the CNN for Handwritten Digit Recognition	Wenfei Liu, Jingcheng Wei, Qingmin Meng	This paper takes the MNIST handwritten digit database as samples, discusses algorithms KNN, SVM, BP neural network, CNN and their application in handwritten digit recognition. In the training process, this work rewrites KNN with Python, SVM with scikit-learn library, and BP, CNN with tensorflow, and finetunes the algorithm parameters to get the best results for each algorithm. CNN showed the best performance with accuracy of 97.7%



6.	Improvised number identification using SVM and random forest classifiers	Anand Upadhyay, Mahipal Singh and Vivek Kumar Yadav	The paper deals with the methods for improvised number identification analyse the pre-processing methods on Support Vector Machine (SVM) and Random Forest (RF) on the handwritten digit dataset. Testing is done using pre-processed data as well as data without pre-processing.
----	--	---	--

## 2.4 Summarization of Malayalam text using natural language processing techniques.

"An efficient text summarization using term and inverse frequency with key phrase identification in malayalam language." [2]	Rosna P Haroon, Abdul Gafur M, Nasreen Ali, and Barakkath Nisha U.	2021	"TF-IDF is an extractive approach to text summarization. In this paper, the key phrases were identified with the above mentioned model and tested on a limited dataset. On the dataset, it gave very good accuracy but it is not guaranteed as a generic text summarization technique that can be used in malayalam language" [3]
Abstractive summarization of malayalam document using sequence to sequence model [3]	S. K. Nambiar, S. David Peter and S. M. Idicula	2021	"Sequence to Sequence model works well in a generic case, but with limited accuracy. This is one of the challenging methods in front of us and requires a lot of preprocessing steps which are still being developed."

An Extractive Malayalam Document Summarization Based on Graph Theoretic Approach [4]	Ajmal E B and Haroon iR P	2015	The sentences in the documents are represented as nodes in an undirected graph. Two sentences are connected with an edge if the two sentences share some common words, or in other words, their similarity is above some threshold. This representation yields two results: The partitions contained in the graph and form distinct topics covered in the documents.
"Text summarization for malayalam documents an experience" [5]	R.Kabeer and S. M Idicua	2014	"This paper presented two text summarization methods that create generic text summaries for Malayalam documents- standard statistical measures to rank the sentences in the document and the detailed semantic processing of the document to generate the summary. Despite the very different approaches taken by the two summarizers, they both produced quite compatible performance scores."

# Chapter 3

## System Development

### 3.1 System Architecture

#### 3.1.1 Architectural Design

First, the machine learning model is created and trained with as much as data possible(Image vectors of Malayalam characters). After, optimising the model, it is stored into a compressed format (like pickle file) for prediction purpose in future. Then we create a REST API using Flask microframework. It acts as a server and helps to create the backend composed of the saved trained machine learning model and the webpage's front end.

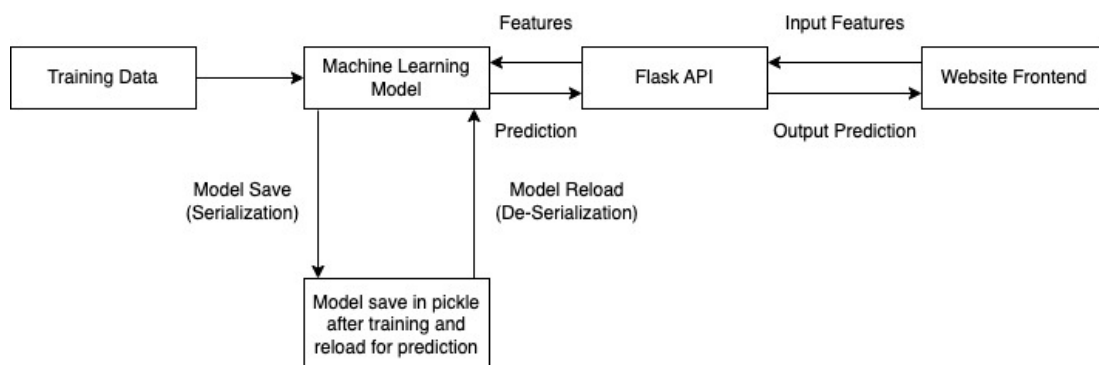


Figure 3.1: Architectural Design

## **Software Interfaces**

The software interface should follow the Model-View-Controller (MVC) model for rendering and modeling data objects. The interface must be able to connect to a database to store XML schema defined using XSD and data streams. Source and destination formats for data must include XML and may also include: Extensible Stylesheet Language Transformation (XSLT), JavaScript Object Notation (JSON), Portable Document Format (PDF), Comma Separated Value (CSV), and American Standard Code for Information Interchange (ASCII). The website requires the following software interfaces to run.

## **Hardware Interfaces**

Since we are dealing with cross-platform as well as the cross-device types, the hardware components mainly consist of gadgets used by evaluators with limited computational capability.

### **3.1.2 Decomposition Description**

The System is divided into 3 modules based on the functionality of the system, that is, it is divided into modules for each of the functionality the system would provide. The Modules are :-

#### **Authentication**

This Module mainly deals with user Authentication. The evaluator could create an account in the platform using their email id and a password. Then, the request from the user to gain access to the system would be received. This request would be processed and if verified, the user would be authorized into the system and can make use of its functionalities.

## Handwritten Malayalam character recognition

This Module tackles the task of converting the uploaded Malayalam handwritten answer scripts in jpg format to the machine encoded textual format which is needed to carry out downstream tasks. The module also handles the necessary pre-processing needed to improve the quality of the inputs.

## Similarity matching

This Module compares the answers in the answer scripts to that in the answer key and tries to recognize similarities based on the information known about them. The documents are represented as vectors of features, and compared by measuring the distance between these features.

## Data Flow Diagram

The Dependencies between the modules and the flow of information is expressed in the DFD diagrams shown in figure 2 and figure 3.

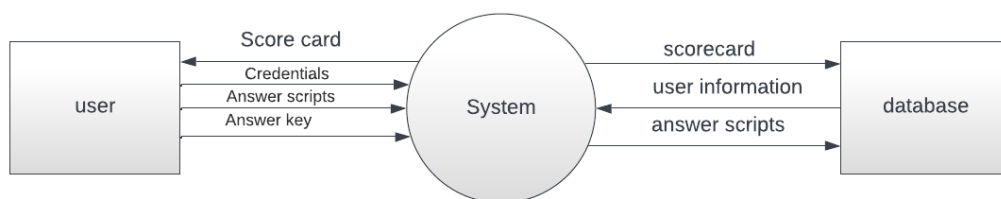


Figure 3.2: Data Flow Diagram Lvl 0

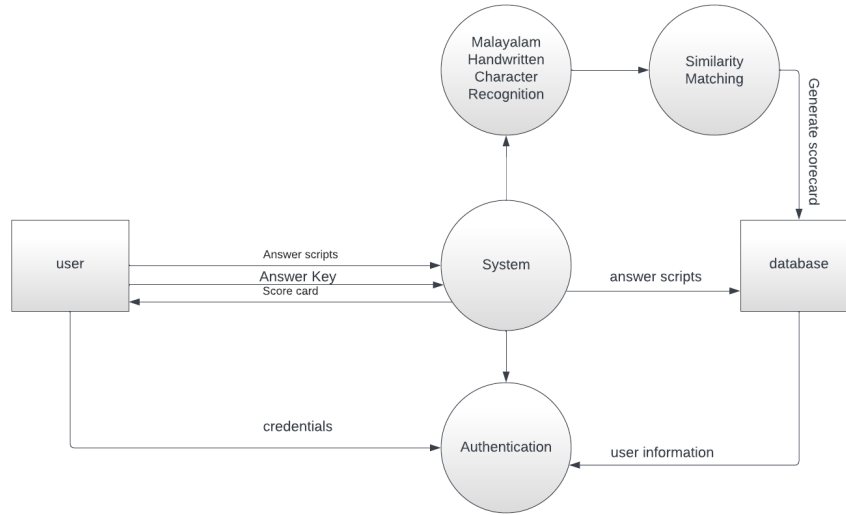


Figure 3.3: Data Flow Diagram Lvl 1

## 3.2 Data Design

### 3.2.1 Data Description

The authentication details of the evaluator are the first details stored in the database. Scanned copies of the answers are uploaded from the local storage of the user via the frontend web page. These images are passed on to the database as well as input to the pre-trained model. The correct answer provided by the teacher is compared with the output given by the model. The teacher's answer is already stored in the database. The output is the corresponding input file's name and the marks obtained in percentage. It is termed as the score card and is stored in the database itself.

### 3.2.2 Data Dictionary

- User : The account details are first data given by the user. Then she/he creates the answer template which consists of the correct answer to be compared with.

Along with this, the scanned copies of the papers to be evaluated are also given by the user, which is given to the system.

- **System:** The system accepts the scanned copies as the input and recognizes the handwritten characters and converts to digital characters, which is the output. Also, it accepts account details as well as answer template from user and gives it to database.
- **Database:** Accepts the details of user via the system. Stores the scanned images, correct answer provided by evaluator, and finally the score card.

### **3.3 Human Interface Design**

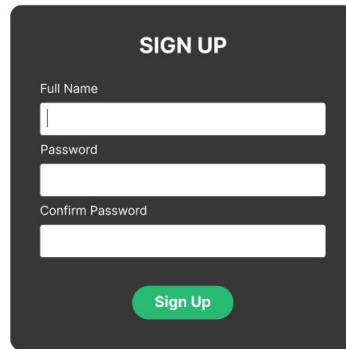
#### **3.3.1 Overview of User Interface**

The website will have a user-friendly interface. Buttons and dialog-boxes for insertion of scanned versions of malayalam handwritten scripts will be available. The user can select "Upload answer script" button from the menu to upload an answer script. To view the score generated, user can select "View score" button from the menu. Buttons will be available to download the score-card generated in pdf format. There is also an option to download the answer template which can be downloaded and printed by the teachers. The students are supposed to fill the answers in the this template.

#### **3.3.2 Screen Images**

##### **Login Screen User Interface**

There will be a login screen user interface where the user can create an account where he/she can upload the answer scripts and marks evaluated will be displayed.



A dark gray rectangular box with rounded corners. At the top, the text "SIGN UP" is centered in white. Below it, there are three white input fields stacked vertically. The first field is labeled "Full Name", the second "Password", and the third "Confirm Password". At the bottom of the box, there is a green rounded button with the text "Sign Up" in white.

Figure 3.4: Login Screen User Interface

### **Output User Interface**

There will be an output user interface where the user can set the answer key. There is also an option to upload the answer scripts. A button to download the answer template is also present which generates an answer template corresponding to the answer key. An option to download the scorecard after evaluation is also present.



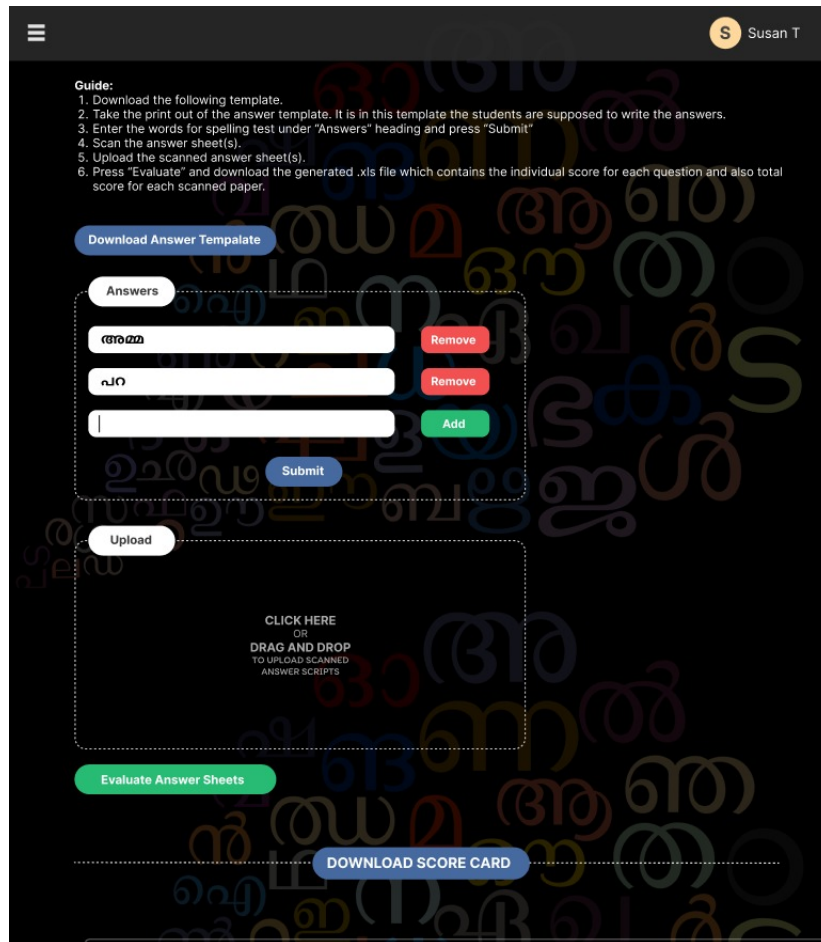


Figure 3.5: Output User Interface

### 3.3.3 Screen Objects and Actions

- Login Screen User Interface - The Evaluator can create an account by entering the full name, and creating a password.
- Output User Interface - The evaluator can create the answer template by adding answers to each question. Also, here the user can upload scanned copies of handwritten answer scripts as well as download the final score card after evaluation

# **Chapter 4**

## **Results and Discussion**

Each chapter is to begin with a brief introduction (in 4 or 5 sentences) about its contents. The contents can then be presented below organised into sections and subsections.

### **4.1 Results of Literature Suvey**

#### **4.1.1 Comparing Handwritten Devanagiri Script recognition**

This section focuses on analyzing the different kinds of algorithms, and methods, based on deep learning in the Character Recognition concept after comparing using various parameters and algorithms. The best possible machine-learning technique for the handwritten text recognition of Devanagiri script is also discussed.

From several experiments and surveys conducted, we see that the proposed technique uses Long Short Term Memory(LSTM) for better accuracy and perfomance. LSTM models can capture long-term dependencies between word sequences and hence LSTMs are better used for text classification. LSTM leads to many more successful runs, and learns much faster. From the survey, a model developed using LSTM

algorithm gives an accuracy of 99.10%, which is the best among others.

Dataset used	Architecture	Accuracy	Pre-Processing Technique
92,000 images	CNN	98.47%	Resizing, Gray scaling, padding
Total 60 documents where 60% used in training and 40% used during testing	SVM	98.35%	Noise Removal, Skew Detection, Normalization, Gray scaling, Binarization
300 samples of handwritten characters	LSTM	99.10%	Binarization
In-house dataset containing 200 images of the character set (each image contains all Hindi characters)	KNN	90%	Normalization, noise removal, and grayscale conversion.

Table 4.1: Comparative study of Deep Learning algorithms for Devanagari Character Recognition

#### 4.1.2 Comparing Handwritten Malayalam Character Recognition

The section focuses mainly on comparing different systems mainly the system that uses chain code histogram, the system that uses gradient and density based features, and the CNN model. Among all the systems, the best possible system is also inferred from the comparison. The results have shown that CNN yields the best possible results in terms of accuracy. The main advantage of CNN compared to its predecessors is that it automatically detects the features without any human intervention. Weight sharing is also present in CNN and it also minimizes computation in comparison with a regular neural network. The current limitation of the existing CNN model to recognize handwritten Malayalam characters is that it fails to recognize compound characters present in Malayalam. But this limitation can be resolved using further machine learning techniques.

The comparative analysis of the different systems is done and among all the systems compared, the best possible system is found. The results show that in the system

that uses chain code based feature extraction technique, two characters may have the same chain code histogram. The system that combines both gradient and density-based features for improving accuracy finds difficulty in finding and choosing the best combination method. In the case of the CNN model, the best possible accuracy is obtained but it fails to recognize compound characters present in the document. But at present, CNN is the best choice for implementing a handwritten Malayalam character recognition system.

System	Preprocessing techniques	Feature Extraction techniques	Classifier	Algorithms used	Datasets	Accuracy Obtained.
Multiple Classifier	Binarization,segmentation,normalisation	Gradient based features,density based features	2 feed forward neural networks.	Resilient back propagation algorithm	20,000 images approx.	81.82%
Based on Chain Code Histogram	Noise removal,Binarization, Segmentation,normalisation.	Chain code,image centroid	2 layer feed forward neural network	Scaled conjugate gradient back propagation algorithm	60 handwritten pages from different persons.	72.1%
CNN	Noise removal,Binarization, segmentation,	Automatic learning	CNN	Back propagation algorithm	90,000 images	97.26%

Table 4.2: Comparison of different techniques for Malayalam HCR

### 4.1.3 Comparison of Handwritten Digit Recognition

This chapter focuses on the results obtained from the literature survey analysis and its conclusions and the results of the testing conducted by the authors. From the literature survey conducted, we obtained the performance measures of various machine learning models on the same dataset which is the MNIST dataset. The consolidated results from all the articles are shown below in tabular as well as graphical form. The conclusions that we reach from the comparative study are discussed in the next chapter.

From the comparative study conducted on Handwritten Digit Recognition with Feed-Forward Multi-Layer Perceptron and Convolutional Neural Network Architec-

tures, the performance of both the models were measured at each iteration of the training. The accuracy increased with the increase in iterations for both models.

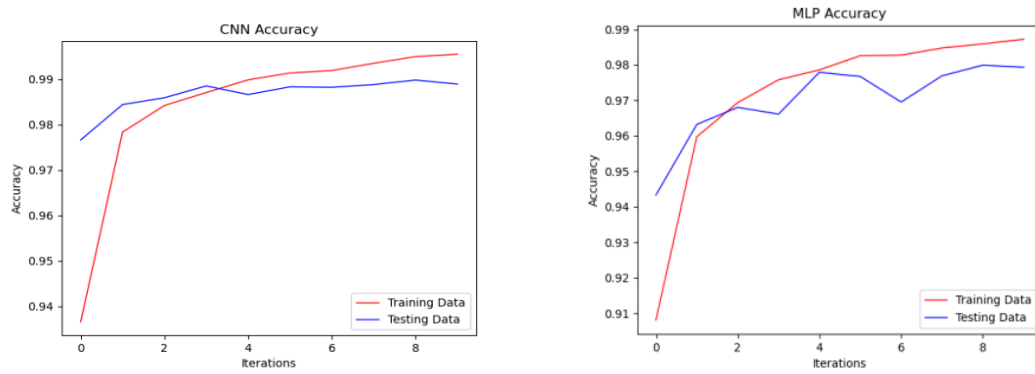


Figure 4.1: Improved accuracy with iterations

Model	Iterations	Accuracy
MLP	5	97.44%
CNN	5	98.76%
MLP	10	97.73%
CNN	10	99.05%

Table 4.3: Table showing variation of Accuracy with number of iterations

From the comparison between SVM, KNN, RF, and CNN. For each algorithm mentioned the testing was conducted by changing the hyperparameters of each algorithm and observing the difference in performance. The images were pre-processed for better results as well. Below is the performance of Support Vector Classifier with different regularization parameters and different kernels.

The different performance results for KNN with different K values and RFC with different numbers of trees is shown below:

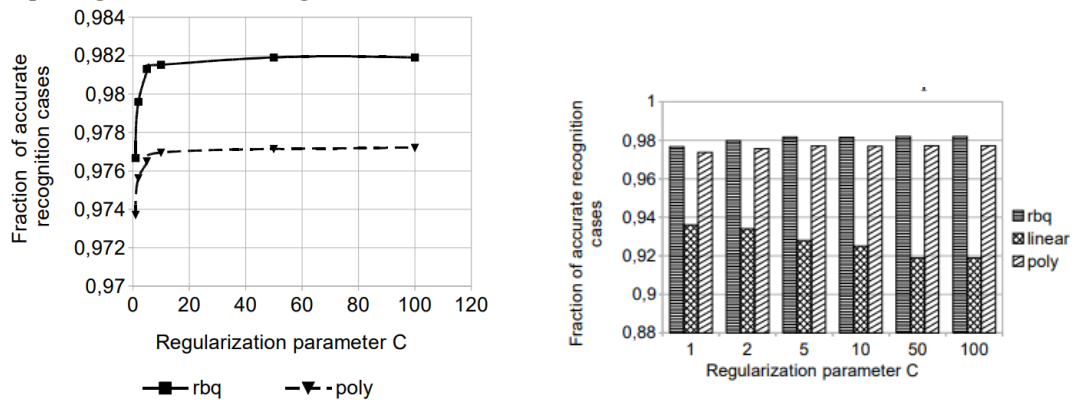


Figure 4.2: SVC with different regularization parameter and different kernels

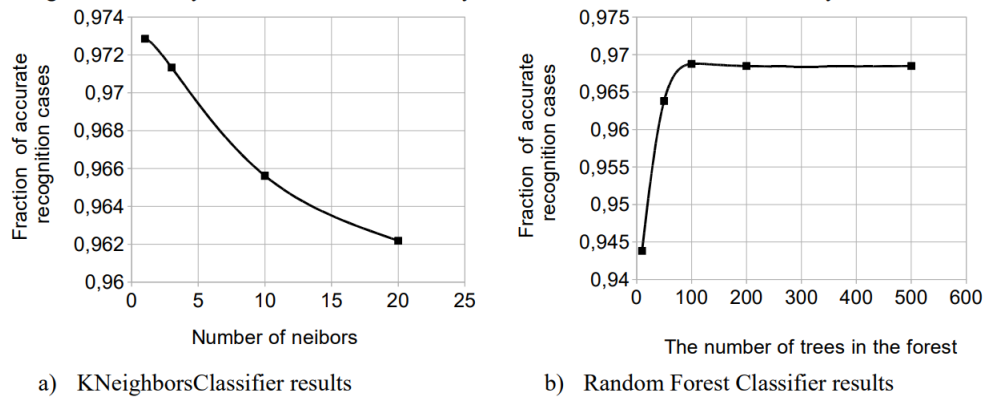


Figure 4.3: Performance of KNN and RFC

In the comparative study of KNN, SVM, RFC, and CNN, the performances obtained is tabulated in Table 5.2.

Algorithm	Accuracy
KNN	96.67%
SVM	97.91%
RFC	96.9%
CNN	98.7%

Table 4.4: Comparison of KNN, SVM, RFC and CNN

A similar comparison was performed between KNN, SVM, BP(MLP) and CNN which yielded the results:

Algorithms	KNN	SVM	BP	CNN
Accuracy(%)	94.6	94.1	96.6	97.7

Table 4.5: Comparison of KNN, SVM, BP and CNN

Another paper focused mainly on the comparison between RFC and SVM which focused on the difference in performance of algorithms on pre-processed images and normal images during training.

Accuracy	SVM	RFC
Without Preprocessing(%)	92.76	94.28
Normalisation(%)	94.10	94.35
Standardisation(%)	95.40	94.16

Table 4.6: Comparison of SVM and RFC

From all these results from various papers we took the best performances in each and made a consolidated table of results:

SINo.	Model	Accuracy
1	MLP	97.44%
2	MLP	97.73%
3	CNN	98.76%
4	CNN	99.05%
5	SVM	91.4%
6	SVM	92.76%
7	KNN	94.6%
8	KNN	96.2-97.4%
9	RFC	94.28%
10	RFC	94-97%

Table 4.7: Recognition rates of various machine learning algorithms

#### 4.1.4 Inferences from different Text Summarization Approaches

There are various abstractive and extractive text summarization methods being researched in the Malayalam language. Abstractive techniques are technically more sound and efficient but require immense pre-processing in the case of Indic languages like Malayalam because of the morphological features it possesses, and the all together different structure of the language, different ways to write the same words etc. Even though abstractive techniques like sequence-to-sequence models work better in a long run, in the short run, extractive techniques provide more accuracy given the text input is not complex and requires only an admissible level of pre-processing techniques. Table

signifies how the Sequence to Sequence abstraction performs. It performs with very minimal accuracy over a small dataset and improves its performance significantly with the increase in the number of data points. Table signifies how the TF-IDF extraction performs. It performs well in small datasets and Its performance in large datasets in very expensive and gives very less accuracy.

Sample Documents	Precision	Recall	F1-Score
200	0.058	0.055	0.056
500	0.064	0.65	0.064
1000	0.073	0.072	0.072
5000	0.72	0.70	0.71

Table 4.8: Results of Abstractive Approach

Table 6.1 signifies how the Sequence to Sequence abstraction performs. It performs with very minimal accuracy over a small dataset and improves it performance significantly with the increase in number of datapoints.

Sample Documents	Precision	Recall	F1-Score
10	0.91	0.87	0.889
20	0.89	0.81	0.848
30	0.9	0.86	0.878
40	0.94	0.89	0.914
50	0.98	0.88	0.927
60	0.95	0.75	0.862
70	0.96	0.81	0.878
80	0.93	0.85	0.888

Table 4.9: Results of Extractive Approach

Table 6.2 signifies how the TF-IDF extraction performs. It perfoms well in small datasets and Its performance in large datasets in very expensive and gives very less accuracy.



# **Chapter 5**

## **Conclusion**

The project mainly aims to automate the evaluation of handwritten malayalam answer scripts thereby, eliminating the repetitive mundane task for the evaluators and saving a lot of time and energy. Offline character recognition is still an area of a lot of research and there hasn't been a significant development of a system to recognize handwritten malayalam characters. Although there are systems that have been developed to recognize handwritten malayalam characters, there hasn't been any system till now that auto-evaluates handwritten malayalam answer scripts. Different machine learning techniques have been used to recognize malayalam characters. Among all the techniques, CNN can be used in this project to implement a model to recognize handwritten malayalam characters. For the answer script evaluation part, different NLP techniques can be used.

### **5.1 ADVANTAGES OF OUR WEBSITE**

1. The website provides a platform for the auto-evaluation of handwritten malayalam answer scripts.
2. Basic spelling checks and one word answers can be evaluated using this website.
3. Saves time and eliminates the repetitive tasks of evaluation.

## **5.2 LIMITATIONS AND FUTURE SCOPE**

1. The website currently is meant for the evaluation of one word answers and basic spelling checks. The project can be scaled up to incorporate the evaluation of descriptive answers in the future releases.
2. The project currently is implemented as a website. But, on future releases, it can be implemented as a mobile application to be supported by android devices.

# References

- [1] J. Ali and J. Joseph, “A convolutional neural network based approach for recognizing malayalam handwritten characters,” 12 2018.
- [2] R. P. Haroon, A. G. M, N. Ali, and B. N. U, “An efficient text summarization using term and inverse frequency with key phrase identification in malayalam language,” in *2021 IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE)*, January 2021, pp. 145–148.
- [3] S. K. Nambiar, S. David Peter, and S. M. Idicula, “Abstractive summarization of malayalam document using sequence to sequence model,” in *2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)*, vol. 1, June 2021, pp. 347–352.
- [4] E. Ajmal and R. P. Haroon, “An extractive malayalam document summarization based on graph theoretic approach,” in *2015 Fifth International Conference on e-Learning (econf)*, 2015, pp. 237–240.
- [5] R. Kabeer and S. M. Idicula, “Text summarization for malayalam documents — an experience,” in *2014 International Conference on Data Science and Engineering (ICDSE)*, 2014, pp. 145–150.
- [6] S. Sangam and T. R. Kumar, “Handwritten digits recognition using novel long short term memory with enhanced f measures over k-nearest neighbour to improve the accuracy,” *Journal of Pharmaceutical Negative Results*, pp. 728–735, 2022.

- [7] A. Sharma, H. Bhardwaj, A. Bhardwaj, A. Sakalle, D. Acharya, and W. Ibrahim, "A machine learning and deep learning approach for recognizing handwritten digits," *Computational Intelligence and Neuroscience*, vol. 2022, 2022.
- [8] S. D. Pande, P. P. Jadhav, R. Joshi, A. D. Sawant, V. Muddebihalkar, S. Rathod, M. N. Gurav, and S. Das, "Digitization of handwritten devanagari text using cnn transfer learning—a better customer service support," *Neuroscience Informatics*, vol. 2, no. 3, p. 100016, 2022.
- [9] M. Agrawal, B. Chauhan, and T. Agrawal, "Machine learning algorithms for handwritten devanagari character recognition: A systematic review," *vol*, vol. 7, pp. 1–16, 2022.
- [10] P. Hirugade, N. Suryavanshi, R. Bhagwat, S. Rajput, and R. Phadke, "A survey on optical character recognition for handwritten devanagari script using deep learning," *Available at SSRN 4031738*, 2022.
- [11] S. Singh, N. K. Garg, and M. Kumar, "Feature extraction and classification techniques for handwritten devanagari text recognition: a survey," *Multimedia Tools and Applications*, pp. 1–29, 2022.
- [12] M. R. Phangtriastu, J. Harefa, and D. F. Tanoto, "Comparison between neural network and support vector machine in optical character recognition," *Procedia computer science*, vol. 116, pp. 351–357, 2017.
- [13] R. Dey, P. Gawade, R. Sigtia, S. Naikare, A. Gadre, and D. Chikmurge, "A comparative study of handwritten devanagari script character recognition techniques," in *2022 IEEE World Conference on Applied Intelligence and Computing (AIC)*, 2022, pp. 431–436.
- [14] R. R. Akhil, K.K. and V. Anoop, "Parts-of-speech tagging for malayalam using deep learning techniques," in *2019 International Conference on Smart Systems and Inventive Technology (ICSSIT)*, September 2020, p. 741–748.
- [15] M. R. Raj and R. P. Haroon, "Malayalam text summarization: Minimum spanning tree based graph reduction approach," in *2016 2nd International Conference on*

- Advances in Computing, Communication, Automation (ICACCA) (Fall)*, 2016, pp. 1–5.
- [16] P. P. Nair, A. James, and C. Saravanan, “Malayalam handwritten character recognition using convolutional neural network,” in *2017 International Conference on Inventive Communication and Computational Technologies (ICICCT)*, 2017, pp. 278–281.
  - [17] J. John, K. V. Pramod, and K. Balakrishnan, “Offline handwritten malayalam character recognition based on chain code histogram,” in *2011 International Conference on Emerging Trends in Electrical and Computer Technology*, 2011, pp. 736–741.
  - [18] A. M. M. Chacko and P. Dhanya, “Multiple classifier system for offline malayalam character recognition,” *Procedia Computer Science*, vol. 46, pp. 86–92, 2015, proceedings of the International Conference on Information and Communication Technologies, ICICT 2014, 3-5 December 2014 at Bolgatty Palace & Island Resort, Kochi, India.
  - [19] A. Chacko and D. Pm, “Handwritten character recognition in malayalam scripts-a review,” *International Journal of Artificial Intelligence & Applications*, vol. 5, 02 2014.
  - [20] M. A. Rahiman, A. Shajan, A. Elizabeth, M. Divya, G. M. Kumar, and M. Rajasree, “Isolated handwritten malayalam character recognition using hlh intensity patterns,” in *2010 Second International Conference on Machine Learning and Computing*, 2010, pp. 147–151.
  - [21] D. Svozil, V. Kvasnicka, and J. Pospichal, “Introduction to multi-layer feed-forward neural networks,” *Chemometrics and Intelligent Laboratory Systems*, vol. 39, no. 1, pp. 43–62, 1997.
  - [22] M. Jogin, Mohana, M. S. Madhulika, G. D. Divya, R. K. Meghana, and S. Apoorva, “Feature extraction using convolution neural networks (cnn) and deep learning,” in *2018 3rd IEEE International Conference on Recent Trends*

- in *Electronics, Information & Communication Technology (RTEICT)*, 2018, pp. 2319–2323.
- [23] W. Liu, J. Wei, and Q. Meng, “Comparisons on knn, svm, bp and the cnn for handwritten digit recognition,” in *2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications( AEECA)*, 2020, pp. 587–590.
- [24] I. S. Yevhen Chychkarova, Anastasiia Serhiiienkob and A. Karginc, “Handwritten digits recognition using svm, knn, rf and deep learning neural networks,” 2021.
- [25] V. C. Sunita S. Patil, V. Mareeswasri and P. Singh, “Recognition of handwritten digits with the help of deep learning,” 2020.
- [26] A. A. Yahya, J. Tan, and M. Hu, “A novel handwritten digit classification system based on convolutional neural network approach,” *Sensors*, vol. 21, no. 18, 2021. [Online]. Available: <https://www.mdpi.com/1424-8220/21/18/6273>
- [27] S. Ahlawat, A. Choudhary, A. Nayyar, S. Singh, and B. Yoon, “Improved handwritten digit recognition using convolutional neural networks (cnn),” *Sensors*, vol. 20, no. 12, 2020. [Online]. Available: <https://www.mdpi.com/1424-8220/20/12/3344>
- [28] E. Tuba, R. Capor Hrosik, A. Alihodzic, R. Jovanovic, and M. Tuba, “Support vector machine optimized by fireworks algorithm for handwritten digit recognition,” in *Modelling and Development of Intelligent Systems*, D. Simian and L. F. Stoica, Eds. Cham: Springer International Publishing, 2020, pp. 187–199.
- [29] D. Beohar and R. Akhtar, “Handwritten digit recognition of mnist dataset using deep learning state-of-the-art artificial neural network (ann) and convolutional neural network (cnn),” in *2021 International Conference on Emerging Smart Computing and Informatics (ESCI)*, 2021, pp. 542–548.
- [30] R. Sethi and I. Kaushik, “Hand written digit recognition using machine learning,” in *2020 IEEE 9th International Conference on Communication Systems and Network Technologies (CSNT)*, 2020, pp. 49–54.

- [31] A. Harikrishnan, S. Sethi, and R. Pandey, “Handwritten digit recognition with feed-forward multi-layer perceptron and convolutional neural network architectures,” in *2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, 2020, pp. 398–402.
- [32] A. Upadhyay, M. Singh, and V. K. Yadav, “Improved number identification using svm and random forest classifiers,” *Journal of Information and Optimization Sciences*, vol. 41, no. 2, pp. 387–394, 2020. [Online]. Available: <https://doi.org/10.1080/02522667.2020.1723934>