

MAJOR PROJECT REPORT ON Handwritten Character Recognition

SUBMITTED BY :

Hariprasad N (20218105)

Rahul Baburaj (20218531)

Yedhu Krishna V (12180093)

Faaiz Mohammed M P (12180036)

Vishnu E (12170083)

In partial fulfillment of the requirements for the award of Degree of Bachelor of Technology in Computer Science and Engineering.



DIVISION OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF ENGINEERING
COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY
NOVEMBER 2021

DIVISION OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF ENGINEERING
COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY

CERTIFICATE

Certified that this is a bonafide record of the Major Project titled

Handwritten Character Recognition

done by

Hariprasad N (20218105)

Rahul Baburaj (20218531)

Yedhu Krishna V (12180093)

Faaiz Mohammed M P (12180036)

Vishnu E (12170083)

of VII Semester, Computer Science and Engineering in the year 2022 in partial fulfillment requirements for the award of Degree of Bachelor of Technology in Computer Science and Engineering of Cochin University of Science and Technology.

Dr. David Peter
Head of Division

Mr. Pramod Pavithran
Project Coordinator

Mrs. Lima Johnson K
Project Guide

Acknowledgement

We take this opportunity to thank the supreme being, the source of all knowledge whose blessings are our guiding light in any venture we take up. We are in short of words to express our gratitude to Mrs. Lima Johnson K, our project guide who guided us and helped us constantly with his inputs and suggestions, without which we could not have implemented this project the way it works today. We are highly indebted to Dr. David Peter, Head of Division , Computer Science and Engineering , project coordinator and staff advisor for his constant supervision and support in completing this project. A bouquet of gratitude to all the staff members who have helped us during this course of work.

Hariprasad N (20218105)

Rahul Baburaj (20218531)

Yedhu Krishna V (12180093)

Faaiz Mohammed M P (12180036)

Vishnu E (12170083)

Declaration

We, Hariprasad N , Rahul Baburaj ,Yedhu Krishna V , Faaiz mohammed M P and Vishnu E hereby declare that this major project is the record of authentic work carried out by us during the academic year 2020 - 2021 and has not been submitted to any other University or Institute towards the award of any degree.

ABSTRACT

Handwriting recognition is the ability of a machine to receive and interpret handwritten input from multiple sources like paper documents, photographs, touch screen devices etc. Recognition of handwritten and machine characters is an emerging area of research and finds extensive applications in banks, offices and industries. The main aim of this project is to design expert system for , “HCR(English) using Neural Network”.that can effectively recognize a particular character of type format using the Artificial Neural Network approach. Neural computing is comparatively new field, and design components are therefore less well specified than those of other architectures. Neural computers implement data parallelism. Neural computers are operated in way which is completely different from the operation of normal computers. Neural computers are trained (not Programmed) so that given a certain starting state (data input); they either classify the input data into one of the number of classes or cause the original data to evolve in such a way that a certain desirable property is optimized

Contents

1. Introduction

1.1 Project

1.2 Relevant Theory

1.2.1 Benefits of Character Recognition

1.2.2 Implementation of HCR

1.2.3 What is Neural Network

1.2.4 Why use Neural Network

1.3 Literature Survey

1.3.1 Offline Handwritten English Numerals Recognition using Correlation Method

1.3.2 Intelligent Systems for Off-Line Handwritten Character Recognition: A Review

1.3.3 Fuzzy Based Handwritten Character Recognition System

1.3.4 An Overview of Character Recognition Focused on Off-Line Handwriting

1.3.5 Image preprocessing for optical character recognition using neural networks

1.3.6 Recognition for Handwritten English Letters: A Review

1.3.7 Handwritten Devanagari Character Recognition using Neural Network:

1.4 Scop

1.5 Objective

2. Requirement Analysis

2.1 Requirement Specification

2.1.1 Functional Requirements\

2.1.2 Normal Requirements

2.1.3 Expected

2.1.4 Excited Requirements

2.1.5 Non-Functional requirements

2.2 Validation of Requirements

2.3 System Requirements

2.3.1 Hardware Requirements

2.3.2 Software Requirements

3. System Design

3.1 Process Model

3.1.1 Incremental Model

- 3.1.2 Advantages of Incremental Model
- 3.1.3 Why we use Incremental Model?
- 3.1.4 Characteristics of Incremental Model
- 3.2 Breakdown Structure (Modules)
- 3.2.1 Image Pre-processing
- 3.2.2 Segmentation
- 3.2.3 Feature Extraction
- 3.2.4 Classification

4. Technical Specification

- 4.1 Technology Details Used in Project
- 4.1.1 JAVA Development Kit
- 4.1.2 Eclipse
- 4.2 References to Technology

5. Software Implementation

- 5.1 Introduction
- 5.2 Important Modules And Algorithm Used
- 5.2.1 Module 1: Image Processing
- 5.2.2 Module 2: Segmentation
- 5.2.3 Module 3: Feature Extraction
- 5.2.4 Module 4: Training And Recognition

6. Conclusion and Future Scope

7. References

CHAPTER 1

Introduction

Handwriting recognition is the ability of a machine to receive and interpret handwritten input from multiple sources like paper documents, photographs, touch screen devices etc. Recognition of handwritten and machine characters is an emerging area of research and finds extensive applications in banks, offices and industries. The main aim of this project is to design expert system for, "HCR using Neural Network" that can effectively recognize a particular character of type format using the Artificial Neural Network approach.

Neural computing is comparatively new field, and design components are therefore less well specified than those of other architectures. Neural computers implement data parallelism. Neural computers are operated in way which is completely different from the operation of normal computers. Neural computers are trained (not programmed) so that given a certain starting state (data input); they either classify the input data into one of the number of classes or cause the original data to evolve in such a way that a certain desirable property is optimized.

1.1 Project Definition

This application is useful for recognizing all character (English) given as in input image. Once input image of character is given to proposed system, then it will recognize input character which is given in image. Recognition and classification of characters are done by Neural Network. The main aim of this project is to effectively recognize a particular character of type format using the Artificial Neural Network approach.

1.2 Relevant Theory

1.2.1 Benefits of Character Recognition :

- 1.The idea of Neural Network in HCR will brings us the reading of various combined style of writing a character.
- 2.In forensic application HCR will be an effective method for evidence collection.
- 3.It will also help to reduce noise from the original character.
- 4.Our method develop accuracy in recognizing character in divert font and size.
- 5.More set of sample invites more accuracy rate because of heavy training and testing session.

1.2.2 Implementation of HCR :

HCR works in stages as pre-processing , segmentation, feature extraction and recognition using neural network. Pre-processing includes series of operations to be carried out on document image to make it ready for segmentation. During segmentation the document image is segmented into individual character or numeric image then feature extraction technique is applied on character image. Finally feature vector is presented to the selected algorithm for recognition .Here this extracted features are provided to NN for recognition of character.

1.2.2 What is Neural Network

An Artificial Neural Network (ANN) is an information-processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information .The key element of this paradigm is the novel structure of the information processing system. It is composed of large no. of highly interconnected processing element (neurons) working in union to solve specific problems. ANN's like peopling, learning by example. An ANN is configured for a specific application. such as pattern recognition or data classification, through a learning process .Learning in a Biological system involves adjustments to the synaptic connections that exist between the neuron.

1.2.3 Why use Neural Network

Neural network with their remarkable ability to derive meaning from complicated or imprecise data can be use to extract pattern and detect trend that are too complex to be noticed by either human or other computer techniques. A trained neural network can be thought of as an “expert” in the category of information it has been given to analyse. This expert can then be used to provide projections

given new situations of interest and answer “what if” questions. Other Advantages Include:

- Adaptive Learning: An ability to learn how to do tasks based on the data given for training or initial experience.
- Self-Organization: An ANN can create its own organization or representation of the information it receives during learning time.
- Real Time Operation: ANN computations may be carried out in parallel, and special hardware devices are being designed and manufactured which take advantage of this capability.
- Fault Tolerance Via Redundant Information coding: partial destruction of network leads to the corresponding degradation of performance. However, some network capabilities may be retained even with major network damage.

1.3 Literature Survey

1.3.1 Offline Handwritten English Numerals Recognition using Correlation Method:

In this paper author has proposed system is to efficiently recognize the offline handwritten digits with a higher accuracy than previous works done. Also previous handwritten number recognition systems are based on only recognizing single digits and they are not capable of recognizing multiple numbers at one time .So the author has focused on efficiently performing segmentation for isolating the digits.

1.3.2 Intelligent Systems for Off-Line Handwritten Character Recognition:

A Review Handwritten character recognition is always a frontier area of research in the field of pattern recognition and image processing and there is a large demand for Optical Character 4 Recognition on hand written documents. This paper provides a comprehensive review of existing works in handwritten character recognition based on soft computing technique during the past decade.

1.3.3 Fuzzy Based Handwritten Character Recognition System:

This paper presents a fuzzy approach to recognize characters. Fuzzy sets and fuzzy logic are used as bases for representation of fuzzy character and for recognition. This paper describes a fuzzy based algorithm which first segments the character and then using fuzzy system gives the possible characters that match the given input and then using defuzzication system finally recognizes the character.

1.3.4 An Overview of Character Recognition Focused on Off-Line Handwriting:

Character recognition (CR) has been extensively studied in the last half century and progressed to a level sufficient to produce technology driven applications. Now, the rapidly growing computational power enables the implementation of the present CR methodologies and creates an increasing demand on many emerging application domains, which require more advanced methodologies.

1.3.5 Image preprocessing for optical character recognition using neural networks:

Primary task of this master's thesis is to create a theoretical and practical basis of preprocessing of printed text for optical character recognition using forward-feed neural networks . Demonstration application was created and its parameters were set according to results of realized experiments.

1.3.6 Recognition for Handwritten English Letters: A Review:

Character recognition is one of the most interesting and challenging research areas in the field of Image processing. English character recognition has been extensively studied in the last half century. Nowadays different methodologies are in widespread use for character recognition. Document verification, digital library, reading bank deposit slips, reading postal addresses, extracting information from cheques, data entry, applications for credit cards, health insurance, loans, tax forms etc. are application areas of digital document processing. This paper gives an overview of research work carried out for recognition of hand written English letters. In Hand written text there is no constraint on the writing style. Hand written letters are difficult to recognize due to diverse human handwriting style, variation in angle, size and shape of letters. Various approaches of hand written character recognition are discussed here along with their performance.

1.3.7 Handwritten Devanagari Character Recognition using Neural Network:

In this digital era, most important thing is to deal with digital documents, organizations using handwritten documents for storing their information can use handwritten character recognition to convert this information into digital. Handwritten Devanagari characters are more difficult for recognition due to presence of header line, conjunct characters and similarity in shapes of multiple characters. This paper deals with development of grid based method which is combination of image centroid zone and zone centroid zone of individual character or numerical image. In feature extraction using grid or zone based approach

individual character or numerical image is divided into n equal sized grids or zones then average distance of all pixels with respect to image centroid or grid centroid is computed. In combination of image centroid and zone centroid approach it computes average distance of all pixels present in each grid with respect to image centroid as well as zone centroid which gives feature vector of size $2 \times n$ features. This feature vector is presented to feed forward neural network for recognition. Complete process of Devanagari character recognition works in stages as document preprocessing, segmentation, feature extraction using grid based approach followed by recognition using feed forward neural network.

1.4 Scope

1. System will be designed in way to ensure that offline Handwritten Recognition of English characters.
2. Our old and epic HCR literature can be restore in digital form.
3. Use of Neural Network for classification.
4. Large number of training data set will improve the efficiency of the suggested approach.

1.5 Objective

1. Use Neural signs in literature domain.
2. Reduced man-power to convert old literature into digitized form manually.
3. proposed system served as guide and working in character recognition areas.
4. Making rich the digitized library with English language

CHAPTER 2

Requirement Analysis

Requirement analysis results in the specification of software's operational characteristics indicates software's interface with other system elements and establish constraints that software must meets. Requirement analysis allows the software engineer (sometime called Analyst or Modeler in this role) to elaborate on basis requirements during earlier requirement engineering task and build models that depict user scenarios ,functional activities , problem classes and their relationship system and class behaviour and the flow of data as it is transformed.

The requirements analysis task is a process of discovery , refinement , modelling and specification. The scope , initially established by us and refined during project planning , is refined in details .Model of the required data , information and control flow and operations behaviour are created.

2.1 Requirement Specification

2.1.1 Functional Requirements

The functional requirements for a system describe what system do.

- 1.The developed system should recognize handwritten English character present in the image.
- 2.System shall show the error message to the user when given input is not in the required format.
- 3.System must provide the quality of service to user.
- 4.System must provide accuracy for character recognition.

2.1.2 Normal Requirements

These are the requirements clearly stated by the customer hence requirement must be present for customer satisfaction.

N1 : Application should have graphical user interface.

N2 : Input of characters with various font size and styles should recognize. 8

N3 : Database should identify computer based English character by comparison .

N4 : Application should be able for matching the stored patterns on input handwritten character.

N5 : Minimum 10*50 (characters * patterns) should be available for each character.

2.1.3 Expected Requirements

These requirements are implicit type of requirements .These requirements are not clearly stated by customer but even though customer expects them.

Exp1 : Instead of only one character application should take set of characters or text.

Exp2 Application should be user friendly and also easy to install.

Exp3 : By using Neural Network bringing more accuracy in character recognition process.

Exp4 : Application also recognize English numerals.

Exp5 : Minimum 26*50 (character * patterns) should be available for each character.

2.1.4 Excited Requirements

These requirements are neither stated by customer nor expected. But to make the customer satisfied , developer may include some unexpected requirements.

Exc1: Application interpret all the English alphabets through NN training process.

Exc2: Using this application Continuous handwritten characters need to be recognize.

Exc3: Alphanumerical characters with special symbol should be recognized with proposed system.

Exc4: Development of HCR system for noisy images.

2.1.5 Non-functional requirements

As the name suggest these are the requirements that are not directly interacted with specific functions delivered by the system.

Performance: Handwritten characters in the input image will be recognised with an accuracy of about 90

Functionality: This software will deliver on the functional requirements.

Availability: This system will retrieve the handwritten text regions only if the image contains written text in it.

Flexibility: It provides the users to load the image easily.

Learn ability: The software is very easy to use and reduces the learning work.

2.2 Validation of Requirements

The project “HCR using Neural Network” will be recognized as successful implementation if it provide all the required images on the basis of suitable input with minimum time. The requirement specification define should be validated such a that the successful implementation of product can be recognized. Hence validation specifies classes of tests to be performed to validate function, performance and the constraints.

With respect to the system under consideration the following issues are to be validated to ensure consistency of system.

V1: The Neural Network are known to be capable of providing good recognition rate at the present as compare to other methods.

V2: Handwritten Character Recognition system give much better result in terms of performance and accuracy in comparison with existing usual approach due to the application of artificial way character recognition and Neural Network in detection of characters.

V3: Handwritten Character Recognition technology provides image definition, image pre-processing and image segmentation and recognition capabilities and still maintains high level of accuracy in the field of image processing.

2.3 System Requirements .

2.3.1 Hardware Requirements

- Intel i3 Processer
- 128 MB RAM
- 10 GB Hard Disk.

2.3.2 Software Requirements

- Windows 7/8/8.1
- Language: Java(J2SE) JDK 1.7.
- Eclipse

CHAPTER 3

System Design

3.1 Process Model

Process Model are processes of the same nature that are classified together into a model. Thus, a process model is a description of a process at the type level. Since the process model is at the type level, a process is an instantiation of it. The same process model is used repeatedly for the development of many applications and thus, has many instantiations. One possible use of a process model is to prescribe how things must/should/could be done in contrast to the process itself which is really what happens. A process model is roughly anticipation of what the process will look like. What the process shall be determined during actual system development.

The goal of a process model is to be:

- **Descriptive**

- 1.Track what actually happens during a process.
- 2.Take the point of view of an external observer who looks at the way a process has been performed and determines the improvements that must be made to make it perform more effectively or efficiently.

- **Prescriptive**

- 1.Define the desired processes and how they should/could/might be performed.
- 2.Establish rules, guidelines, and behaviour patterns which, if followed, would lead to the desired process performance. They can range from strict enforcement to flexible guidance.

- **Explanatory**

- 1.Provide explanations about the rationale of processes.
- 2.Explore and evaluate the several possible courses of action based on rational arguments.
- 3.Establish an explicit link between processes and the requirements that the model needs to fulfil.
- 4.Pre-defines points at which data can be extracted for reporting purposes.

3.1.1 Incremental Model

Incremental model is used as the process model in our system. Figure 3.1 shows the process model of the system. To save actual problems in an industry setting, Software Engineering must incorporate a development strategy that encompasses the process, method and the tool layers; this strategy is often referred as process model. A process model for Software Engineering is chosen base on the nature of the Project and its application. For our project, we have selected Incremental Model.

- 1.Using these models, a limited set of customer requirements are implemented quickly and are delivered to customer.
- 2.Modified and expanded requirements are implemented step by step.
- 3.It combines elements of linear Sequential Model with the iterative Philosophy of prototyping.
- 4.Each linear sequence produces a deliverable Increment of the Software.
- 5.Each linear Sequence is divide into 4 parts:-
 - Analysis
 - Design
 - Code
 - Testing

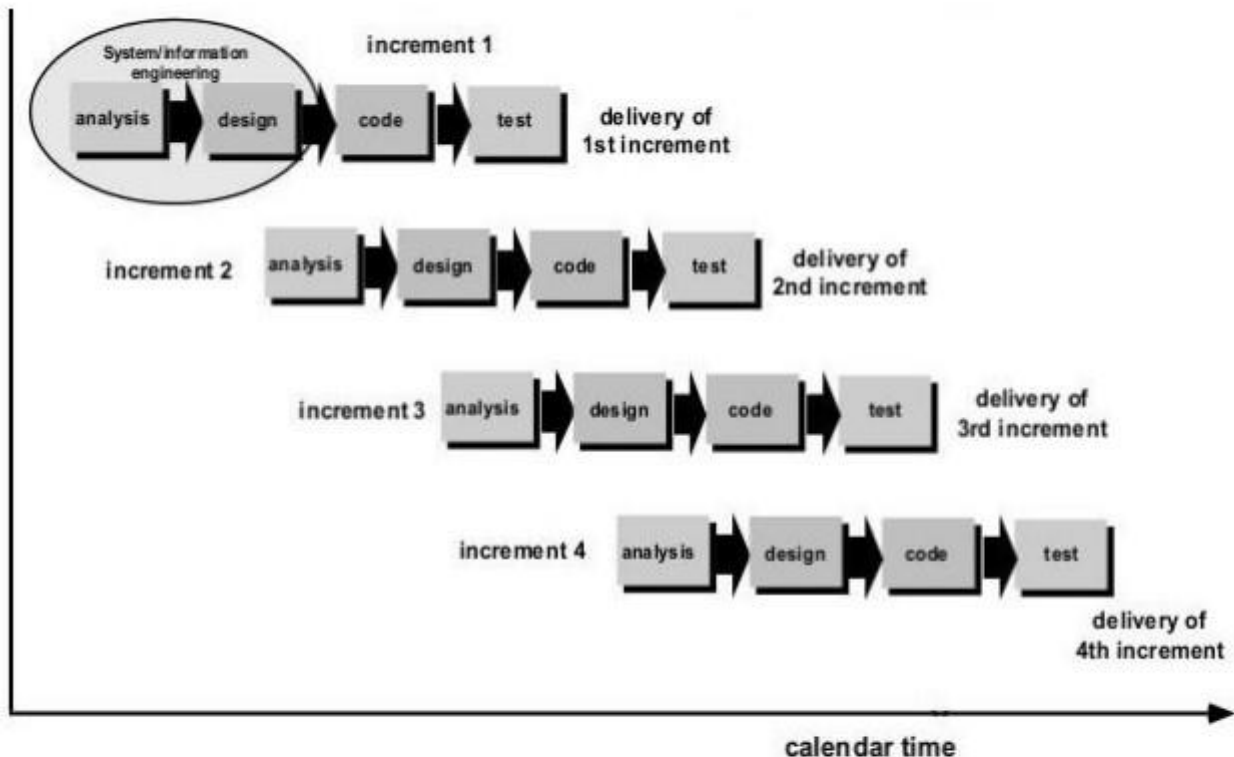


Figure 3.1: Incremental Model

1. Analysis

It includes understanding of information domain, required functions, behaviour, performance and interface. Requirements for the system and software are documented and reviewed with customer.

2. Design

It is multiple processes that include four attributes of program data structure, software architecture, interface representation and procedural detail.

3. Coding

Translation of design to machine code is done by this step.

4. Testing

It focuses on Logical internals of Software and ensures that all statement is correct to uncover all hidden errors. For an incremental model, the first Increment is developed as a core model, which is used by the customer. Then as things are added after the first delivery, product gets and better.

3.1.2 Advantages of Incremental Model

1. Generates working Software quickly and early during the software life cycle.
2. More Flexible-less costly to change Scope and Requirements.
3. Easier to test and debug during a smaller iteration.
4. Customer can respond to each built.

3.1.3 Why we use Incremental Model?

The main aim of using the model is the reason that we have to add more features in the existing modules to increase project reliability and usability. Using this model we can adapt to the changing requirements of the customer which helps in developing the project in relatively small amount of time. The next increment implements customer suggestions plus some additional requirements in the previous increment. The process is repeated until the project is completed.

3.1.4 Characteristics of Incremental Model

1. Using these models a limited set of customers' requirements are implemented quickly and are delivered to customer then modified and expanded requirements are implemented step by step.
2. Each increments produces the product which is submitted to customer and suggests some change and increment implements that changes with some extra requirements to previous.
3. Incremental model does not facilitate the development of project in one go. This is useful for developing modules and then testing them which helps us to modularize the entire project for better handling.

So finally, it is easier to develop project in increments. We can develop a working prototype 1st with just basic functions and then build upon this prototype in later increments. This will help to reduce the complexity of system by dividing entire system in different levels of priority. We have discarded the other process models based on following points:

1. When staff is less then we can go for step by step evolution of increments.
2. In case of increment process model, scope for intermediate requirements change and testing.

3.2 Breakdown Structure (Modules)

As shown in Fig 3.2 implemented of proposed work is divided in the following modules:

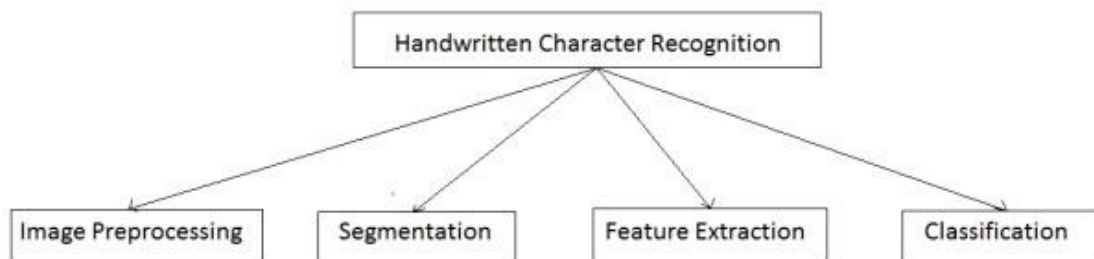


Figure 3.2: Breakdown Model

1. Image Pre-processing
2. Segmentation
3. Feature Extraction
4. Classification

3.2.1 Image Pre-processing

The image is pre-processed using different image processing algorithms like Inverting image , Gray Scale Conversion and image thinning.

3.2.2 Segmentation

After pre-processing of the image segmentation is done. This is done with the help of following steps:

1. Remove the borders
2. Divide the text into rows
3. Divide the rows (lines) into words
4. Divide the word into letters

3.2.3 Feature Extraction

Once the character is segmented we generate the binary glyphs and calculate the summation of each rows and columns values as an features.

3.2.4 Classification

In this phase, we are going to train and test the Neural Network.

CHAPTER 4

Technical Specification

4.1 Technology Details Used in Project

4.1.1 JAVA Development Kit

STEP 1: Download JDK.

1. Goto Java SE download site that is [www.oracle.com](http://www.oracle.com/technetwork/java/javase-downloads) visit tech network than java click on java-se then downloads to visit index.html page.
2. Click the "Download" button under "JDK" of "Java SE7".
3. Choose your operating platform, e.g., Windows x86 (for 32-bit Windows OS - "jdk-7u2-windows-i586.exe" 84MB); or Windows x64 (for 64-bit Windows OS).

STEP 2: Install JDK/JRE

1. Run the downloaded installer, which installs both the JDK (Java Development Kit) and JRE (Java Runtime). By default, the JDK and JRE will be installed into C drive:Program Files ,java and into jdk1.7.0" and Cdrive:Program Files ,java and into jre7", respectively .Refer JDK installed directory as JAVA HOME, by accepting terms and conditions and by clicking to next we can use jdk for our java programs. Setting PATH and CLASS PATH variables in order to compile or execute java (Environmental variable):

1. Click My computer ,properties and then advanced settings.
2. Now click Environment Variables, here variables are divided into two sections.
3. User variables : whenever it is modified the corresponding language only effected.
4. System variables : whenever it is modified not only a single language, corresponding all languages are only effected.
5. Set variable name = classpath and variable value as directory where java is installed. One more environment variable ,we have to set for Java home directory similar to path and class path. For verification ,whether java is installed successfully or not ,just goto command prompt ,type javac if you see java compiling commands then you have installed it successfully.

4.1.2 Eclipse

STEP 3:install eclipse.

1.Download latest version of eclipse .

2.install eclipse on our system.

3.Eclipse stores your projects in a folder called a workspace. Choose a workspace folder to use for this session.

6.2 References to Technology

www.eclipse.org/downloads.

www.install/windows/jdk installation-windows.html

CHAPTER 5

Software Implementation

5.1 Introduction

GUI Specification For the creation of GUI we have used the java swing toolkit.

Swing

Swing library is an official Java GUI toolkit released by Sun Microsystems. The main characteristics of the Swing toolkit

- 1.platform independent
- 2.customizable
- 3.extensible
- 4.configurable
- 5.lightweight

5.2 Important Modules And Algorithm Used

Following is a description of the various implementation steps , which were applied in order to achieve the final target our project.

5.2.1 Module 1: Image Processing

Pre-processing includes steps that are required to shape the input image into a form suitable for segmentation. Color image is converted into gray scale. Image transform into binary image that means in the form of black in white image.

Gray Scale Conversion

As each color pixel is described by a triplet (R,G,B) of intensities for red, green and blue color. we can map that to a single number giving a gray scale value. There are many approaches to convert color image into gray scale. Here average method is used for color to gray scale conversion.

Algorithm :Gray Scale Conversion

Input :Scanned Handwritten Document Image

Output :Gray Scaled Document Image

Step 1: Start

Step 2: Select Input Document Image.

Step 3: Repeat for x=0 to Width of Image.

Step 4: Repeat for y=0 to Height of Image.

Step 5: Extract RGB value for each of pixel as RGB(i,j)

```
int col = inPixels[x][y];
```

```
int r = col & 0xff;
```

```
int g = (col >> 8) & 0xff;
```

```
int b = (col >> 16) & 0xff;
```

```
int gs = (r + g + b)/3;
```

Step 6: Set Pixel with computed gray level as :

```
inPixels[x][y] = (gs / (gs << 8) / (gs << 16));
```

```
gimage.setRGB(x, y, inPixels[x][y]);
```

Step 7: Display Gray Scale image.

Step 8: Stop

Binarization

Image Binarization converts an image of upto 256 gray level to a black and white image. The simplest way to use image binarization is to choose a threshold value and classify all pixels with values above the threshold as black and all other pixels are white.

Algorithm :Image Binarization(Thresholding)

Input :Gray Scaled Image

Output :Black and White Image

Step 1: Start

Step 2: Select Gray Scaled Document Image.

Step 3: Repeat for x=0 to Width of Image.

Step 4: Repeat for y=0 to Height of Image.

Step 5: Set the Threshold.

Step 6: Extract RGB value for each of pixel as RGB(i,j)

```
int col = inPixels[x][y];  
int r = col & 0xff;  
int g = (col >> 8) & 0xff;  
int b = (col >> 16) & 0xff;  
int gs = (r + g + b) / 3;
```

Step 7: If pixel(gs) is Above Threshold Then

```
{  
    r=g=b=0;  
}  
else  
{  
    r=g=b=255;  
}
```

Step 8: Set Pixel with computed Threshold level as :

```
inPixels[x][y] = (b / (g << 8) / (r << 16));  
bimage.setRGB(x, y, inPixels[x][y]);
```

Step 9: Display Binarized Image.

Step 10: Stop

7.2.2 Module 2: Segmentation

Once image pre-processing is done it is necessary to segment document into lines, lines into words and words into characters. When characters has been extracted from document we can extract features from it for recognition. Segmentation of image is performed to separate the characters from the image. Characters separation from the input image involves three steps as:

- Line Segmentation
- Word Segmentation
- Character Segmentation

Line Segmentation

To perform line segmentation, we need to scan each horizontal pixel row starting from the top of document. The lines are separated where we find a row with no black pixels. This row acts as a separation between two lines.

Algorithm : Line Segmentation

Input : Binarized Document Image

Output : Segmented lines from Document Image

Step 1: Start

Step 2: Select Document Image.

Step 3: Repeat for $x = 0$ to Height of Image.

Step 4: Repeat for $y = 0$ to Width of Image.

Step 5: Scans Each pixels from Horizontal pixel row.

Step 6: Extract RGB value for each pixels in `Pixels[x][y]`

Step 7: If pixel with no Black pixel is found then

```
{  
Segment line from document image  
}
```

Step 8: Stop

Word Segmentation

To perform word segmentation, we need to scan each vertical pixel column starting from the left of line. The words are separated where we find a column with no black pixels for more than predefined columns. This column acts as a separation between two words.

Algorithm : Word Segmentation

Input : Segmented lines from Image and `avgpxl` = average pixel width for word separation

Output : Segmented words from line

Step 1: Start

Step 2: Select Document Image.

Step 3: Repeat for $x = 0$ to Height of Segmented line image.

Step 4: Repeat for $y = 0$ to Width of Segmented line image..

Step 5: Scans Each pixels from Vertical pixel column.

Step 6: Extract RGB value for each pixels in Pixels[x][y]

Step 7: If pixel with no Black pixel is found for more than avgpxl then

{

Segment Word from lines

}

Step 8: Stop

Character Segmentation

To perform character segmentation, we need to scan each vertical pixel column starting from the left of word. The characters are separated where we finds a column with no black pixels columns. This column acts as a separation between two character.

Algorithm : Character Segmentation

Input : Segmented words from lines

Output : Segmented characters from words

Step 1: Start

Step 2: Select Document Image.

Step 3: Repeat for $x = 0$ to Height of Segmented word.

Step 4: Repeat for $y = 0$ to Width of Segmented word.

Step 5: Scans Each pixels from Vertical pixel column.

Step 6: Extract RGB value for each pixels in Pixels[x][y]

Step 7: If pixel with no Black pixel is found then

{

Segment characters from words

}

Step 8: Stop

7.2.3 Module 3: Feature Extraction

As individual characters has been separated, character image can be re sized to 15 x 20 pixels. If the features are extracted accurately then the accuracy of recognition is more. Here we have use the 15 x 20 means 300 pixels as it is for feature vector. This extracted feature are stored in .dat file

7.2.4 Module 4: Training And Recognition

The Features extracted from previous modules are given as an input for Neural Network. The Kohonen algorithm is an automatic classification method which is the origin of Self- Organizing Maps .This SOM is used for training and recognition.

Training and recognition

This method is called to train the network. It can run for a very long time and will report progress back to the owner object.

The back end for recognition is neural network. The number of input to each network is associated with the size of the feature vector for each text. The no of output considered for this data sets are 27 (Characters a-z and 1 reject neuron for non character pattern).

CHAPTER 6

Conclusion and Future Scope

6.1 Conclusion

Many regional languages throughout world have different writing styles which can be recognized with HCR systems using proper algorithm and strategies. We have learning for recognition of English characters. It has been found that recognition of handwritten character becomes difficult due to presence of odd characters or similarity in shapes for multiple characters. Scanned image is pre-processed to get a cleaned image and the characters are isolated into individual characters. Pre-processing work is done in which normalization, filtration is performed using processing steps which produce noise free and clean output. Managing our evolution algorithm with proper training, evaluation other step wise process will lead to successful output of system with better efficiency. Use of some statistical features and geometric features through neural network will provided better recognition result of English characters. This work will be helpful to the researchers for the work towards other script.

6.2 Future Scope

This work further extended to the character recognition for other languages. It can be used to convert the fax and news papers into text format .In order to recognize words ,sentences or paragraphs we can use multiple ANN for classification .It can be used in post office for reading postal address

CHAPTER 7

REFERENCES

- <https://projectworlds.in/artificial-intelligence-project-handwritten-digits-recognition/>
- <https://data-flair.training/blogs/python-deep-learning-project-handwritten-digit-recognition/>
- <https://towardsdatascience.com/build-a-handwritten-text-recognition-system-using-tensorflow-2326a3487cd5>
- <https://www.pyimagesearch.com/2020/08/24/ocr-handwriting-recognition-with-opencv-keras-and-tensorflow/>