**CHAPTER 1: INTRODUCTION**

**Problem Definition**

This project focuses on recognising handwritten characters of Devnagari script written in different styles and fonts.

**Overview**

Character recognition is the mechanical conversion of images of typed, handwritten or printed text into machine-encoded, with the from a scanned document or a photo of a document. Devnagari is an ancient and widely used script. The Devnagari script is used for over 120 languages In the last half century, the English character recognition was studied and the results were of such type that’s it can produce technology driven applications. But the same approach cannot be used in case of Indian languages due to the nature of complication in terms of structure and computation. Now days there are different methodologies which are growing fastly in the area of Indian languages and character recognition. The offices, banks, schools and other organisations are working in the field of digital document processing. Devnagari is the national language of India and generally spoken by 600 million people in India. Devnagari should be given more special consideration for analysis and document retrieval due to its popularity. This paper is mainly concern for the people who are working in the Devnagari Optical character and it provides an over view about Devnagari character recognition system.

This method takes the input image from the user and applies various transformation. Then density based clustering is performed on this new image to identify the border points. Later matching of the input is done with the training data set.

The Devnagari Character Recognition System will be for the end user. This system will be designed to maximize the system’s reliability by providing a means to assist humans in the character recognition process which would otherwise have to be performed manually. By maximizing the efficiency and production, the system will meet the user’s needs while remaining easy to understand and use.

More specifically, this system is designed to recognise the Devnagari characters written in different styles, fonts and sizes. The system also contains a database that comprises of a training set containing the variations in each character of the Devnagari script.

Different Phases of handwritten character recognition :

* Creation of Database
* Image Processing
* Pixel processing
* Thinning
* Scaling
* Noise reduction
* Density Based Clustering
* Identify relevant border objects in four frames of a character image
* Combine multiple clusters for further mapping with dataset
* Chain coding Technique
* Identification of characters – by comparing and Mapping to standard character database

**Background**

**Image Processing** is a method to convert an image into digital form and perform some operations on it in order to get an enhanced image or to extract some useful information from it. Image processing system includes treating images as two dimensional signals while applying already set signal processing methods to them.

The most powerful technology which has a great potential to help companies focus on some important data in their data warehouse is data mining. **Data mining** is the extraction of hidden predictive information from large databases. In order to help businesses make more proactive and knowledge driven decisions, data mining tools predict future trends and behaviour.

**Clustering** is a process of partitioning a set of data or objects into a set of meaningful subclasses called clusters. These clusters are formed such that the intra cluster similarities are maximum and the inter cluster similarities are minimum.

**Types of Clustering:**

1. Density-based Clustering:

In density based clustering, clusters are dense regions in the data space separated by regions of lower object density. Given a set of points in some space, it groups together points that are closely packed(points with many nearby neighbours), marking as outlier points that lie alone in low density regions(whose nearest neighbours are too far away) [B1]. Density based clustering does not require a prior specification of the number of clusters. It is able to identify noise and outliers.

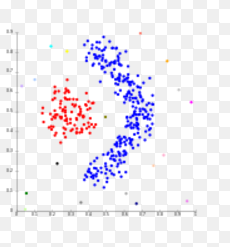


Fig 1.1- Density based Clustering [B2]

1. Partition-based Clustering:

Partition-based clustering is a simple division of the set of data objects into non-overlapping clusters such that each data object is in exactly one cluster [A3].

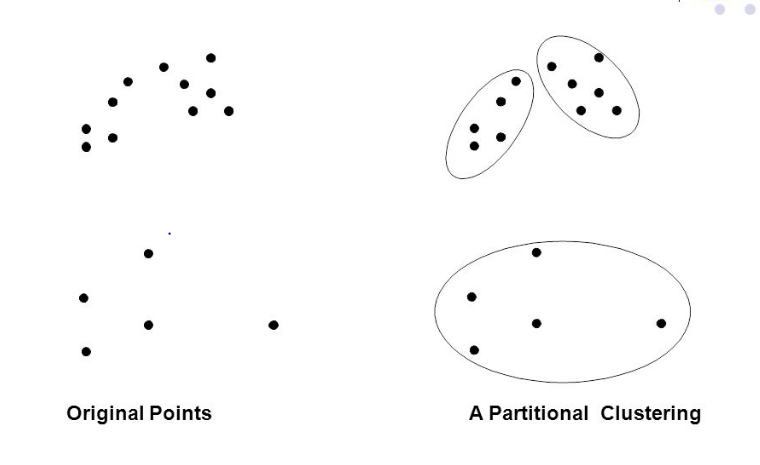


Fig 1.2- Partition based Clustering [B2]

1. Hierarchical Clustering:

Hierarchical clustering is a set of nested clusters that are organised as a tree. The generated tree may correspond to a meaningful taxonomy [A3]. The two types of hierarchical clustering are:

* Agglomerative: It is a bottom-up approach. It starts with the points as individual clusters and at each step merge the closest pair of clusters.
* Divisive: It is a top-down approach. It starts with one; all-inclusive cluster and each step split a cluster until only singleton clusters of individual points remain.

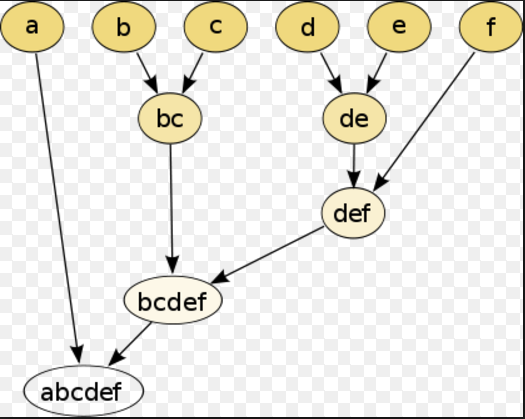
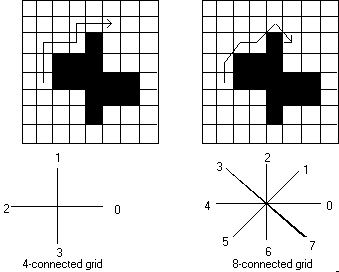


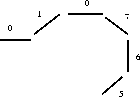
Fig 1.3- Hierarchical Clustering [B2]

A **chain code** is a way of representing a list of points. It may be defined either with respect to the pixels or the boundaries between pixels. A pixel in a rectangular grid has either 4 or 8 neighbours depending on the definition of connectivity, as represented in Figure below. The chain code is defined in this case by tracking around the white pixels in the outer boundary ( as opposed for example to tracking round the black pixels in the inner boundary ). For the examples shown, the codes are 1100100 etc. and 21017 etc. Alternatively, it is possible to track between the black and white regions along the intersections of the pixels using a 4-connected code. In Figure 8, the edge data is tracked directly, again using an 8-connected form; the short bars represent the thinned edges from a standard edge detection algorithm. A chain code describes the boundary as a series of one pixel vector transitions at different orientations, only the starting point is defined explicitly.

Chain codes are position independent if the start point is ignored, which may simplify matching of orientation-fixed linear figures. The derivative of the chain code (i.e. the change in the code either clockwise or anti-clockwise) may be invariant under boundary rotation.



**Figure :** Boundary chain codes



**Figure :** A chain code for edge pixels

**Methodology**

The process of recognition begins with acceptance of the input image of the handwritten character from the user. Further, the input image is pre-processed by carrying out scaling, cropping, conversion of image from RGB to Grey and binarization. This processed image is then converted into a matrix. now this matrix is fed to the novel density based clustering. Here instead of keeping the clusters we just check whether the point is dense , that is if it contains certain number of pixels in its neighbor. Applying this technique on the matrix, boundary of the input character is obtained using density as the similarity measure.

Therefore, a system needs to be developed which performs pixel by pixel processing on scanned copy of handwritten Devnagari characters. A database containing a training set is maintained. This training set contains distances of the pixels from the centroid for each image. There are diverse images of handwritten characters written in different styles, fonts and sizes. The input characters are thus compared with the characters in the training set and the character is recognized. Thus, the system tries to mock the human ability of recognizing the Devnagari characters even if they are written in different styles.

1. Database creation : A database is created which contains the training set which will be used to compare the handwritten scanned character and map it further to recognize the character.
2. Handwritten character scanning: The very first step is to scan the handwritten devnagri character which may be in any font and size.
3. Preprocessing : Now this image is fed to the preprocessing module, the preprocessing includes the conversion of RGB image to grayscale image. Next this image is cropped to extract only the written part in the scanned image and remove the space which is not a part of the character.
4. Conversion to matrix : The preprocessed character image is then converted into a matrix containing binary values for the pixels in the image.
5. Border extraction : Now novel density based clustering is applied to the matrix to extract the border of the character. Basically in this approach each and every pixel is tested for a candidature of being a border point. This candidature is based on the criteria of minimum number of pixels present in its vicinity.
6. Chain coding : After the border is extracted the chain coding approach is applied to the border to map the character to the training set.
7. Mapping and recognition : Now the character is mapped to the training set to compare the similarity of the character to any of the devnagri script characters and further declare it recognized.

**Domain**

Image Processing: Image Processing is processing of images using mathematical operations by using any form of [signal processing](https://en.wikipedia.org/wiki/Signal_processing) for which the input is an image, a series of images, or a video, such as a [photograph](https://en.wikipedia.org/wiki/Photograph) or [video frame](https://en.wikipedia.org/wiki/Video_frame); the output of image processing may be either an image or a set of characteristics or [parameters](https://en.wikipedia.org/wiki/Parameter) related to the image.

Data Mining: It is the computational process of discovering patterns in large [data sets](https://en.wikipedia.org/wiki/Data_set) involving methods at the intersection of [artificial intelligence](https://en.wikipedia.org/wiki/Artificial_intelligence), [machine learning](https://en.wikipedia.org/wiki/Machine_learning), [statistics](https://en.wikipedia.org/wiki/Statistics), and [database systems](https://en.wikipedia.org/wiki/Database_system). The overall goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use.

**Motivation of the Project**

India is a multilingual, multi script country and there are many different languages. Eleven scripts are used to write these languages and Devnagari script is the oldest one that is used to write languages such as Hindi, Marathi, Sanskrit and so on [A4]. Devnagari character recognition can mainly be used in automated systems for recognition of characters.

In Devnagari there are vowels, consonants, vowel modifiers, component characters and numerals. All these variations make the handwritten character recognition, a challenging problem, but there is a need to take some efforts for Devnagari character recognition. The main reason for such an effort is not only the challenges in simulating human reading but also the possibility of efficient applications in which the data present on paper documents has to be transferred into machine-readable format. Automatic recognition of printed and handwritten information present on documents like cheques, envelopes, forms, and other manuscripts is an important application among other applications like text-to-speech systems, identification of historical documents and handwriting recognition in forensics.

**Why density based clustering?**

It is better at handling outliers compared to other algorithms. The number of clusters need not be specified. It is able to find arbitrary sized and shaped clusters. For our system we need to extract border points, here density based clustering eases the task.

**Strengths**

* If a training set is provided for other scripts derived from Devnagari then these scripts can also be identified as the algorithm devised in our project uses distances for clustering.
* It can recognise characters written in different fonts and sizes.
* It can be used in text-to-speech systems for the blind.

**Weaknesses**

* This project only identifies the characters of the Devnagari script; however, by modifying database as per the requirement, it can be very easily made suitable for any language in the world.
* Identifying numbers, compound characters and modifiers of Devnagari script, is not in the scope of current work.

**CHAPTER 2: LITERATURE SURVEY**

Devnagari is the script for Hindi, Sanskrit, Marathi and many other languages. Devnagari script is a logical composition of its constituent symbols in two dimensions. Devnagari has 11 vowels and 33 simple consonants also called as non compound characters. Besides the consonants and the vowels, other constituent symbols in Devnagari are set of vowel modifiers called maatraa (placed to the left, right, above or at the bottom of a character), pure-consonant (also called half letters) which when combined with other consonants yield conjuncts. A horizontal line called shirolekha (a header line running through the entire strip of a word) [A1]. Devnagari script has two dimensional compositions of symbols: core characters in the middle strip, optional modifiers above and/or below core characters. While line segments(strokes) is the predominant feature for English, most of the characters in Devnagari script are formed by curves and strokes [A2]. Vowels occur either in isolation or in combination with consonants.

In our literature survey we found many techniques which were published and used for handwritten Devnagari character recognition, Dr. Sandhya Arora developed a method for handwritten Devnagari recognition system, she had used four feature extraction techniques namely, intersection, shadow feature, chain code histogram and straight line fitting features. Shadow features are computed globally for character image while intersection features, chain code histogram features and line fitting features are computed by dividing the character image into different segments. Weighted majority voting technique is used for combining the classification decision obtained from four Multi Layer Perceptron (MLP) based classifier.[1]

Off-line Devnagari handwritten character recognition is selected for the survey by Dongre and Mankar. They capture the image then preprocess the image by assigning pixels values: 0 or 1 for binary images, 0– 255 for gray-scale images, and 3 channels of 0–255 color values for color images. Then reduce noise, Skew Detection done by making skewed lines horizontal by calculating skew angle and making proper correction in the raw Image. Size normalization by making matrix 32x32 or 64x64 so that all characters have same data size. Thinning done by making boundary detection. Devnagari words can further be splitted to individual character for classification and recognition by removing Shirorekhain. Different feature extraction and classification methods are discussed in detail. [2]

Offline Handwritten Devnagari Character Recognition is done by using different feature as Chain code histogram, four side views, shadow are extracted and fed to Multilayer Perceptrons as a preliminary recognition step. Finally the results of all MLP’s are combined using weighted majority scheme. [3]

Gradient and curvature based feature extraction method is used and comparison of Nearest Neighbour, K-Nearest Neighbour, Euclidian Distance-based K-NN, Cosine Similarity -based KNN, Condensed Nearest Neighbour, Reduced Nearest neighbour, Farthest like neighbour and Nearest unlike Neighbor is done on gray level images which is handwritten images. [4]

Kailash S. Sharma, A. R. Karwankar and Dr. A.S. Bhalchandra develop a system which can recognize an online handwritten Devnagri character. Two layer self organizing map is used. Network is trained by unsupervised learning. [5]

The recognition is carried out using multistage feature extraction and classification scheme. The initial stages of feature extraction are depends on the structural features and then the classification of the characters is done as per their parameters. The final stage of feature extraction employs Radon transform and Euclidean distance transform and applied to two separate feed forward back propagation neural networks. [6]

All features of each input character is stored in a feature vector. Thus feature vectors of all characters in the database are constructed. A feature vector for the test sample is also constructed. Minimum edit distance algorithm is used for final recognition. [7]

Satish Kumar suggest three tier strategies to recognize the hand-printed characters of Devanagri script. In primary and secondary stage classification, the structural properties of the script are exploited to avoid classification error. The results of all the three stages are reported on two classifiers i.e. MLP and SVM [8]

Invariant moments is used for the feature extraction .Recognition rate increases if a character is divided in a systematic manner and features of each divided part are used in recognition system. The three methods of division are suggested in the paper for Recognition of Handwritten Devanagri Numerals. The Gaussian Distribution Function has been adopted for classification. [9], [10]

**CHAPTER 3: SOFTWARE REQUIREMENTS SPECIFICATION (SRS)**

**Purpose of the document:**

The purpose of this document is to present a detailed description of the handwritten Devnagari character recognition system. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to different sized inputs. This document is intended for both the stakeholders and the developers of the system.

**Scope of the project:**

The Devnagari Character Recognition System will be for a local editor of a regional historical society. This system will be designed to maximize the system’s reliability by providing a means to assist humans in the character recognition process which would otherwise have to be performed manually. By maximizing the efficiency and production, the system will meet the user’s needs while remaining easy to understand and use.

More specifically, this system is designed to recognise the Devnagari characters written in different styles, fonts and sizes. The system also contains a database that comprises of a training set containing the variations in each character of the Devnagari script.

A scanned copy of handwritten Devnagari character is used as an input. Therefore, before applying data mining techniques, it is important to process the image pixel by pixel. Novel Density based algorithm is used for detecting border objects of the input image. This algorithm is used as it will remove noise and outliers easily and will detect only the border points in the input image on which a specific threshold is applied so that outliers and noise can be handled efficiently. Once these border objects are detected then chain coding is done on that image so that features of that image can be extracted and these features are then used for comparing the input image from the database containing the training set.

The current scope of the project is to initially input an image which contains the Devnagari script characters and then identify the same by using data mining and image processing techniques. This project also focuses on handling special cases of Devnagari characters such as “maatraa”. This project has been restricted only to the Devnagari character set. However, it can be implemented for any other script. Currently identification of “compound characters, modifiers and numbers” are not in our scope. The future scope of this project will include compound characters and numerals.

* Glossary:

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Image Processing | Image Processing is a method to convert an image into digital form and perform some operations on it in order to get an enhanced image or to extract some useful information from it. Image processing system includes treating images as two dimensional signals while applying already set signal processing methods to them. |
| Data Mining | Data Mining is the process of discovering patterns in large data sets involving methods at the intersection of Artificial Intelligence, Machine Learning, Statistics and database systems. |
| Database | Training set containing the variations in each character. |
| Software Requirements Specification | A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document. |
| Stakeholder | Any person with an interest in the project who is not a developer. |
| User | End user. |

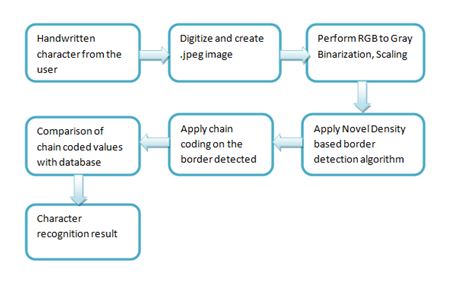
* References:

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

## Overview of Document:

The Overall Description section of this document gives an overview of the functionality of the system. It describes the informal requirements of the handwriting recognition system.

Block Diagram



* User personas and Characteristics

The end user is expected to be comfortable using the automated recognition system.

* Overview of Functional Requirements

The functionalities of the system aim at performing the following:

1. Scanning: Scanning of the input handwritten image.
2. Pre-processing: Pixel recognition, RGB to Grey scaling, auto crop, binarization of the grey scale image.

1. Border detection: Apply modified Novel Density based algorithm and collect border objects.
2. Chain-Code: Apply Chain code on the borders that are detected from previous module
3. Mapping and recognition: Compare the result with the Devnagari char set for recognition.

* Overview of Data Requirements

Training Set containing images of all handwritten characters in different fonts, sizes and styles.

* Operating Environment:

Operating System: Linux / Windows / Unix

IDE: Eclipse

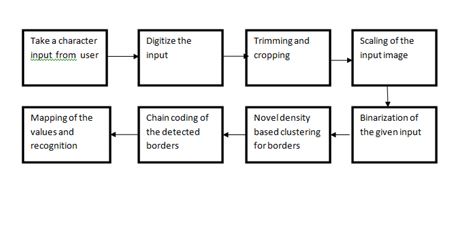
Programming Language : Core Java, MongoDB

* Specific Requirements:
* External Interface Requirements:

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Parameter | Measurement Requirement | Justification |
| 1 | CPU speed | 4 GHz | For accurate working |
| 2 | RAM | 8 GB | For faster processing |

The only link to an external system is the link to the database containing the training set, to verify the membership of a character to a particular character in the training set.

* Detail Description of Functional Requirements:



1. Matrix Formation:

A matrix of the input character is formed from the input image.

1. Scanning:

The matrix derived from the input image of handwritten character is scanned.

1. Scaling:

The process of resizing a digital image is known as Image Scaling. Scaling is a process that involves trade-of-between efficiency, smoothness and sharpness. Scaling is a non-trivial process. Enlarging or reducing the size of the image is done by commands. It changes the number of pixels it contains. This operation works on entire image.

By scaling the input image, it is brought to a normalised size which helps in comparing the input image with the image in the training set.

3) RGB to grey:

A grey scale image is simply the one in which the only colours are shades of grey. The reason for differentiating such images from any other sort of coloured image is that less information needs to be provided for each pixel. Using a grey scale representation, we represent the image in a binary format. This binary format of the image makes it easy to compare the input image with the training data set which contains the images of handwritten characters in different fonts.

4) Binarization:

Thresholding enables to achieve image segmentation in the easiest way. Image segmentation means dividing the complete image into a set of pixels in such a way that the pixels in each set have some common characteristics. Image segmentation is highly useful in defining objects and their boundaries.

1. Chain-Coding:

Chain-Coding is done to detect the features of the input image.In this process every direction of pixel relative to other pixel is calculated and these directions are represented by numerals. These numbers are then use to identify the flow of the character and each numbers frequency is calculated.

* Performance Requirements:
* The CPU speed should be 4 GHz for the system to work accurately.
* For faster processing, the minimum requirement of RAM is 8 GB.
* The performance of the system is dependent on the training set which contains the images of the characters in different styles and fonts.

* Quality Attributes:

* A Diverse training set containing the images of handwritten characters in different styles and fonts is used to compare with the input image. This gives accurate results as the images in the training set include images of all fonts, styles and sizes.
* The quality of the input image can be derived by the diversity of the training database in terms of different styles, fonts and sizes of characters.

**CHAPTER 4: ARCHITECTURE AND DESIGN**

* **Algorithm**:

Clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters).

Novel Density Based Algorithm for border detection : In density-based border detection , high density pixels are defined as areas of higher density than the remainder of the data set. Objects in these sparse areas, that are not crossing the limit of threshold for border detection, are usually considered to be noise and border points. It only connects points that satisfy a density criterion.

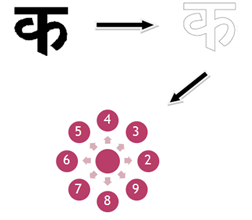
03/17/14 Devnagari Character Recognition 20of 62
Preprocessed Images (a) Original, (b) segmented (c)
Shirorekha removed (d... 03/17/14 Devnagari Character Recognition 20of 62
Preprocessed Images (a) Original, (b) segmented (c)
Shirorekha removed (d...

Border Detection

Chain-Coding:

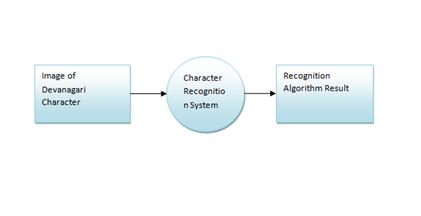
A chain code is a lossless compression algorithm for monochrome images. The basic principle of chain codes is to separately encode each connected component, or "blob", in the image. For each such region, a point on the boundary is selected and its coordinates are transmitted. The encoder then moves along the boundary of the region and, at each step, transmits a symbol representing the direction of this movement. This continues until the encoder returns to the starting position, at which point the blob has been completely described, and encoding continues with the next blob in the image.

This encoding method is particularly effective for images consisting of a reasonably small number of large connected components.

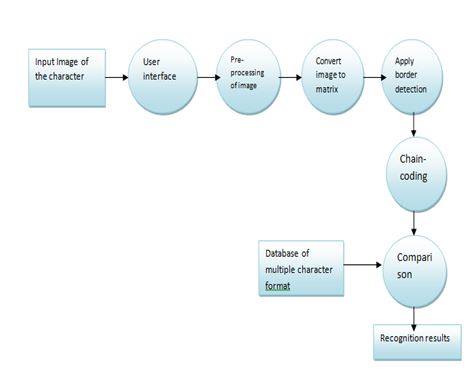


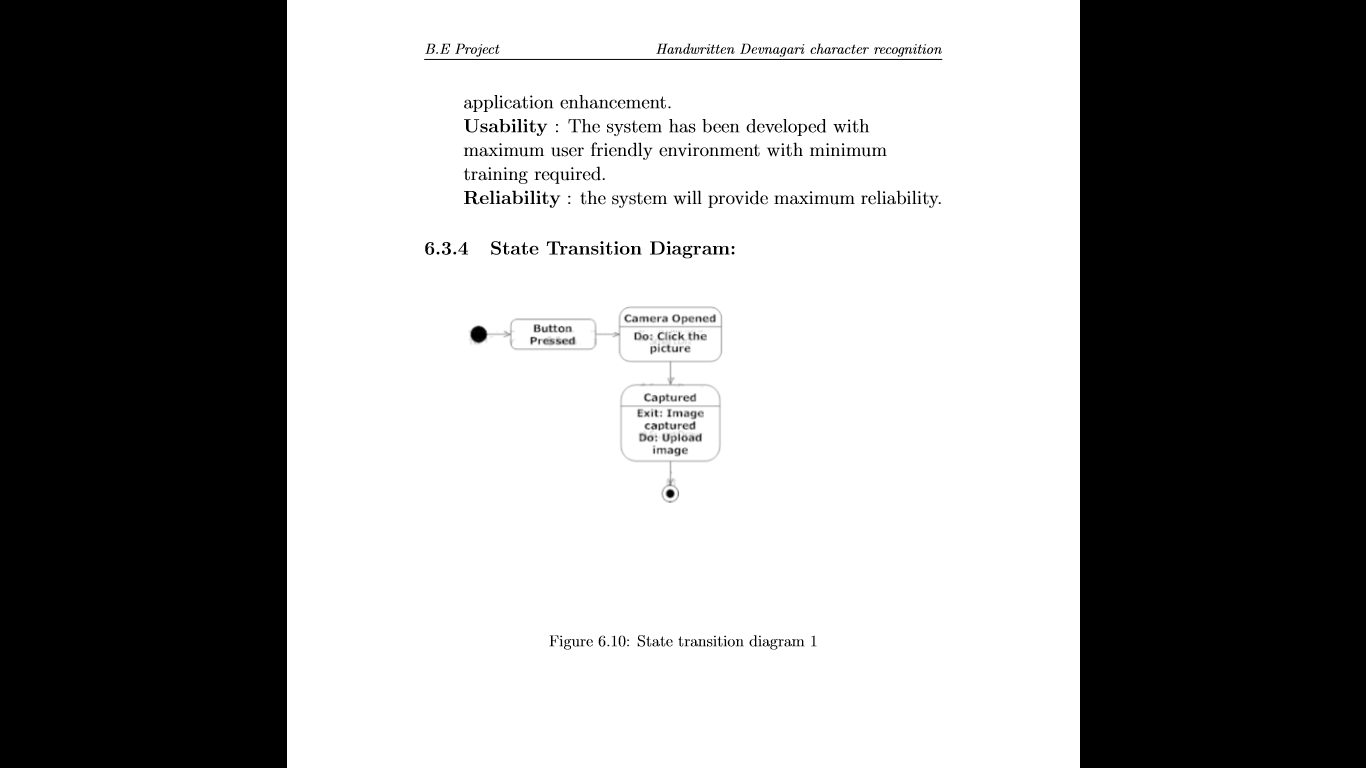
* **Data Flow Diagram** :

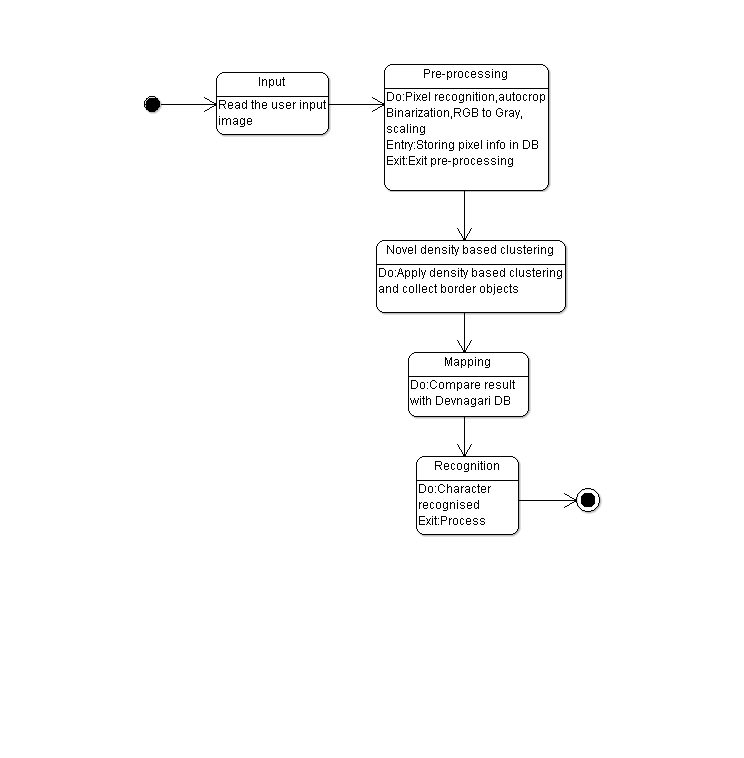
Level 0



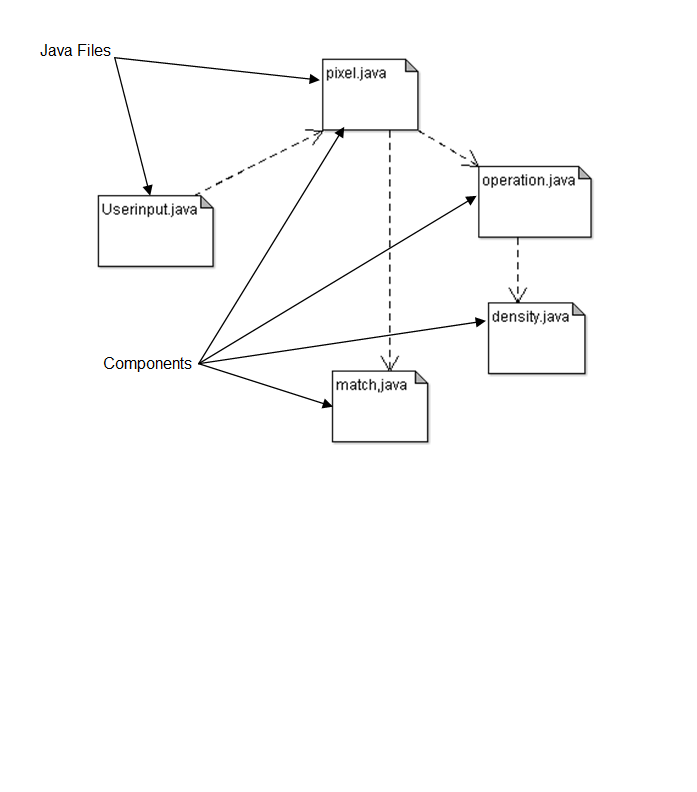
Level 1

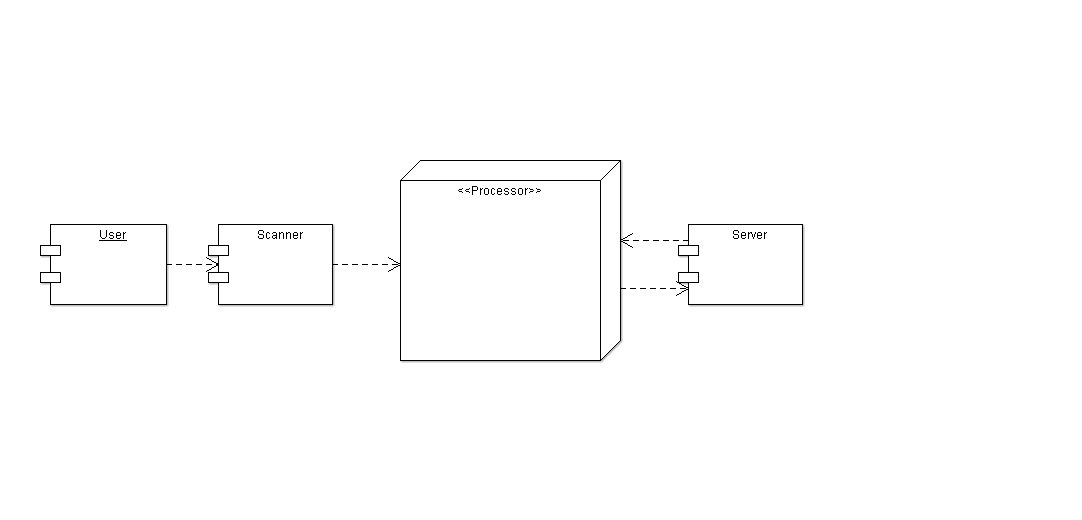


* **State Transition Diagram:**



* **Component Diagram**:



* **Deployment Diagram:**

**CHAPTER 4: TECHNOLOGY**

Devnagari character recognition is difficult because it contains consonants, vowels, vowel modifiers, numerals and complex characters. The handwritten character recognition becomes a challenging task when similar shaped characters and their variations are encountered.

This project will make use of density based clustering and chain coding for recognising handwritten Devnagari characters. **Clustering** is a process of partitioning a set of data or objects into a set of meaningful subclasses called clusters. These clusters are formed such that the intra cluster similarities are maximum and the inter cluster similarities are minimum.

**Types of Clustering**

1. Partition-based Clustering:

Partition-based clustering is a simple division of the set of data objects into non-overlapping clusters such that each data object is in exactly one cluster.

1. Hierarchical Clustering:

Hierarchical clustering is a set of nested clusters that are organised as a tree. The generated tree may correspond to a meaningful taxonomy. The two types of hierarchical clustering are:

* Agglomerative: It is a bottom-up approach. It starts with the points as individual clusters and at each step merge the closest pair of clusters.
* Divisive: It is a top-down approach. It starts with one; all-inclusive cluster and each step split a cluster until only singleton clusters of individual points remain.

1. Density Based Clustering:

In **density based clustering**, clusters are dense regions in the data space separated by regions of lower object density [B1]. Given a set of points in some space, it groups together points that are closely packed(points with many nearby neighbours), marking as outlier points that lie alone in low density regions(whose nearest neighbours are too far away). Density based clustering does not require a prior specification of the number of clusters. It is able to identify noise and outliers.

**Chain Coding**

A chain code is a [lossless](https://en.wikipedia.org/wiki/Lossless) [compression algorithm](https://en.wikipedia.org/wiki/Compression_algorithm) for [monochrome](https://en.wikipedia.org/wiki/Monochrome) [images](https://en.wikipedia.org/wiki/Image). The basic principle of chain codes is to separately encode each [connected component](https://en.wikipedia.org/wiki/Connected_component_%28topology%29), or "blob", in the image.

For each such region, a point on the boundary is selected and its coordinates are transmitted. The encoder then moves along the boundary of the region and, at each step, transmits a symbol representing the direction of this movement.

This continues until the encoder returns to the starting position, at which point the blob has been completely described, and encoding continues with the next blob in the image.

This encoding method is particularly effective for images consisting of a reasonably small number of large connected components.

**CHAPTER 7 : SOFTWARE TESTING**

|  |  |  |
| --- | --- | --- |
| Sr. No. | Test Case Description | Expected Outputs |
| 1 | Check Image Type | Image should be in .jpg format |
| 2 | Check number of images being loaded | Single image should be uploaded |
| 3 | Check loading of result into database | Result getting loaded into database in CSV format |
| 4 | Characters from other scripts supplied | Should not accept and print output accordingly |
| 5 | Check if standard character is being shown at output | Standard symbol displayed |
| 6 | Give valid input | Valid actual output should be displayed |

**CHAPTER 6: IMPLEMENTTION AND CODING**

**FRAMEWORK**

1. MODULE 1 : Scanning: Scanning of the input handwritten image.
2. MODULE 2 : Pre-processing: Pixel recognition, RGB to Grey scaling, auto crop, binarization of the grey scale image.
3. MODULE 3 :Border detection : Apply modified Density based clustering algorithm and collect border objects.
4. MODULE 4 : Mapping and recognition : Compare the result with the Devnagari char set for Recognition.

**MODULE WISE DESCRIPTION**

1. Scanning

The Devnagri character are handwritten for training set. All 43 characters of Devnagri script are scanned having 10 samples each. Example is shown below:

****

1. Pre-processing

The scanned image are then pre processed in order to map them efficiently with the input.

We have applied 2 pre processing techniques:

1. Auto crop : to trim the whit portion
2. Auto scale : to get the image in 100\*100 dimensions.

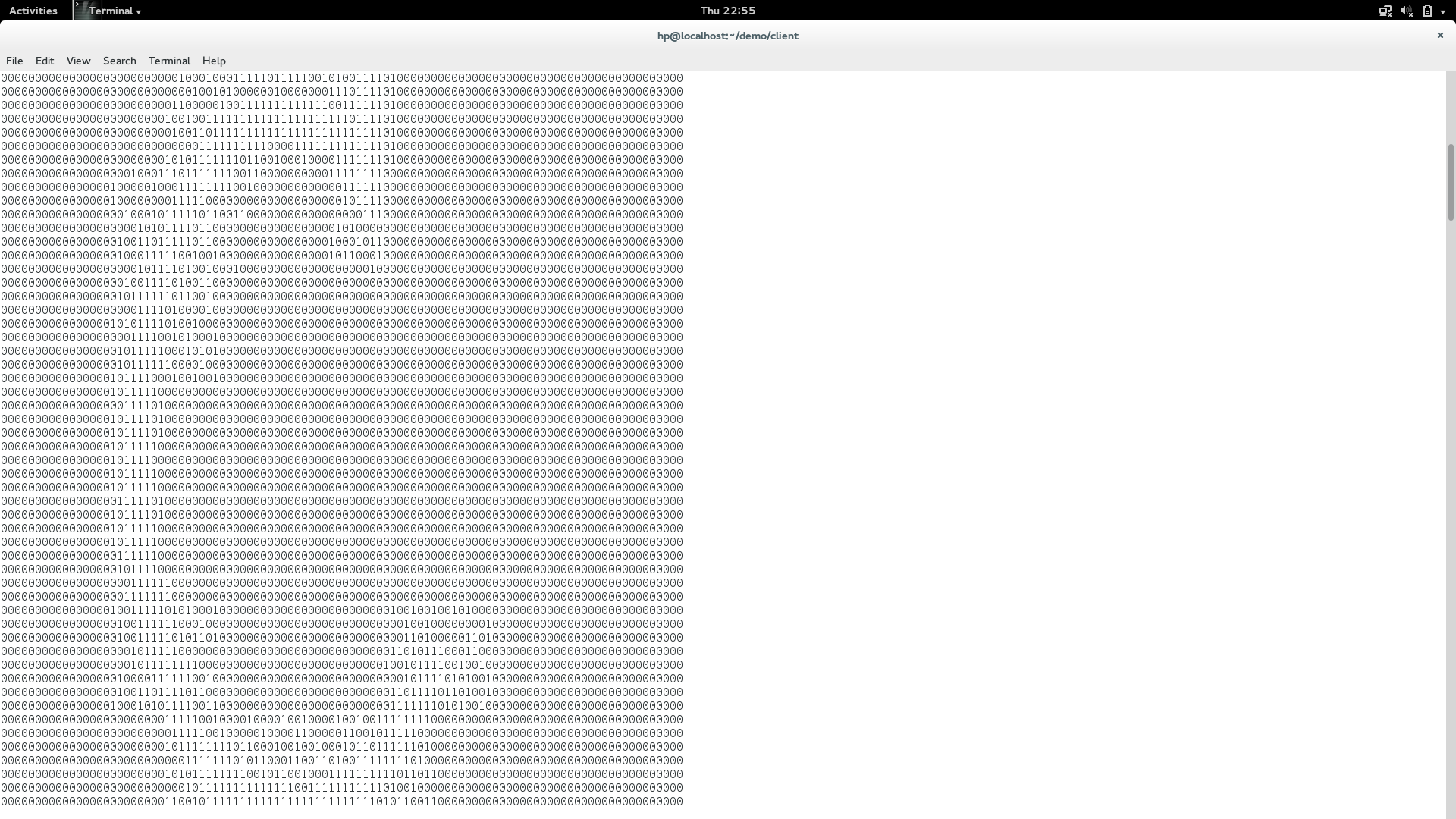
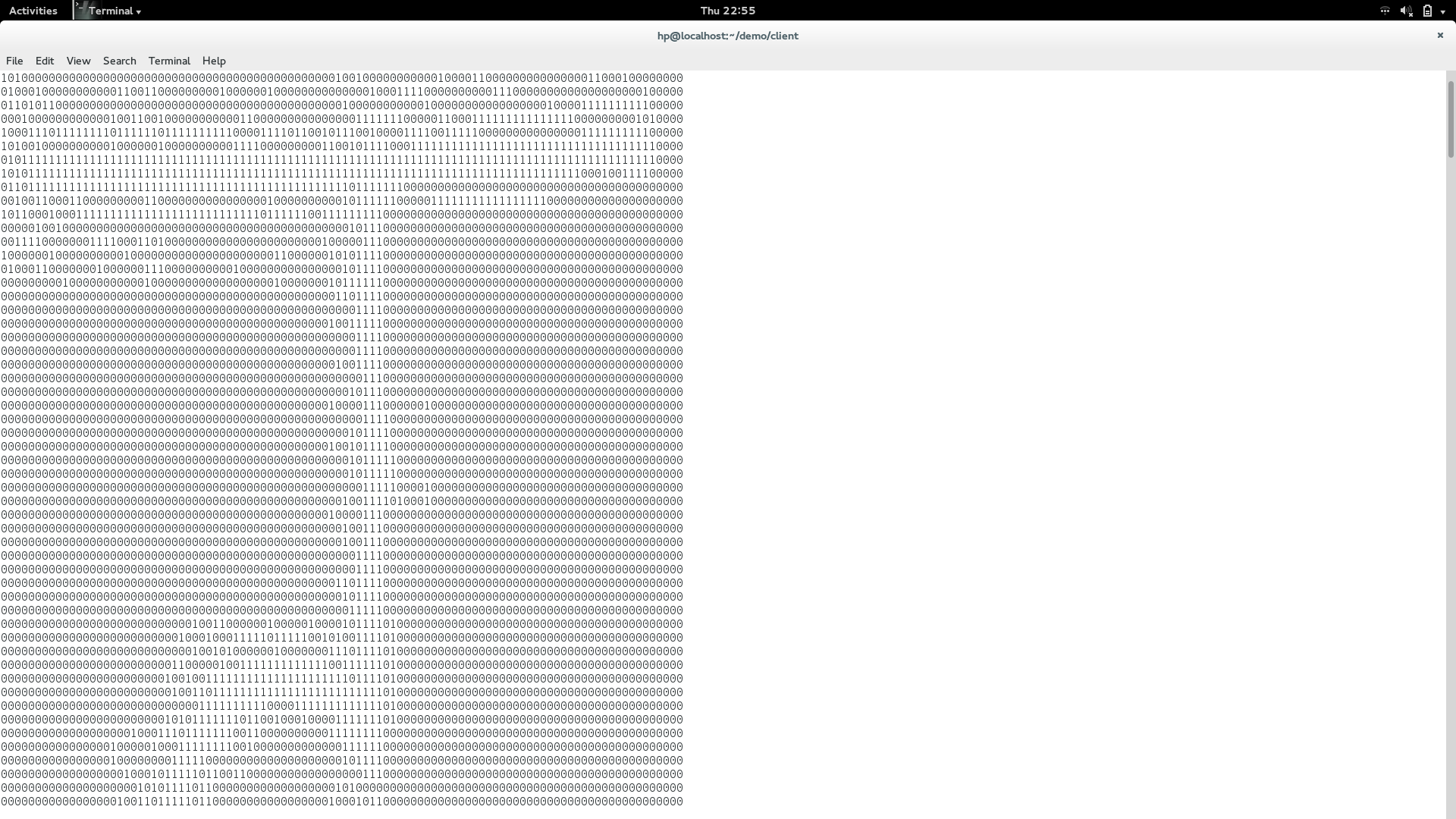


**Auto-crop**

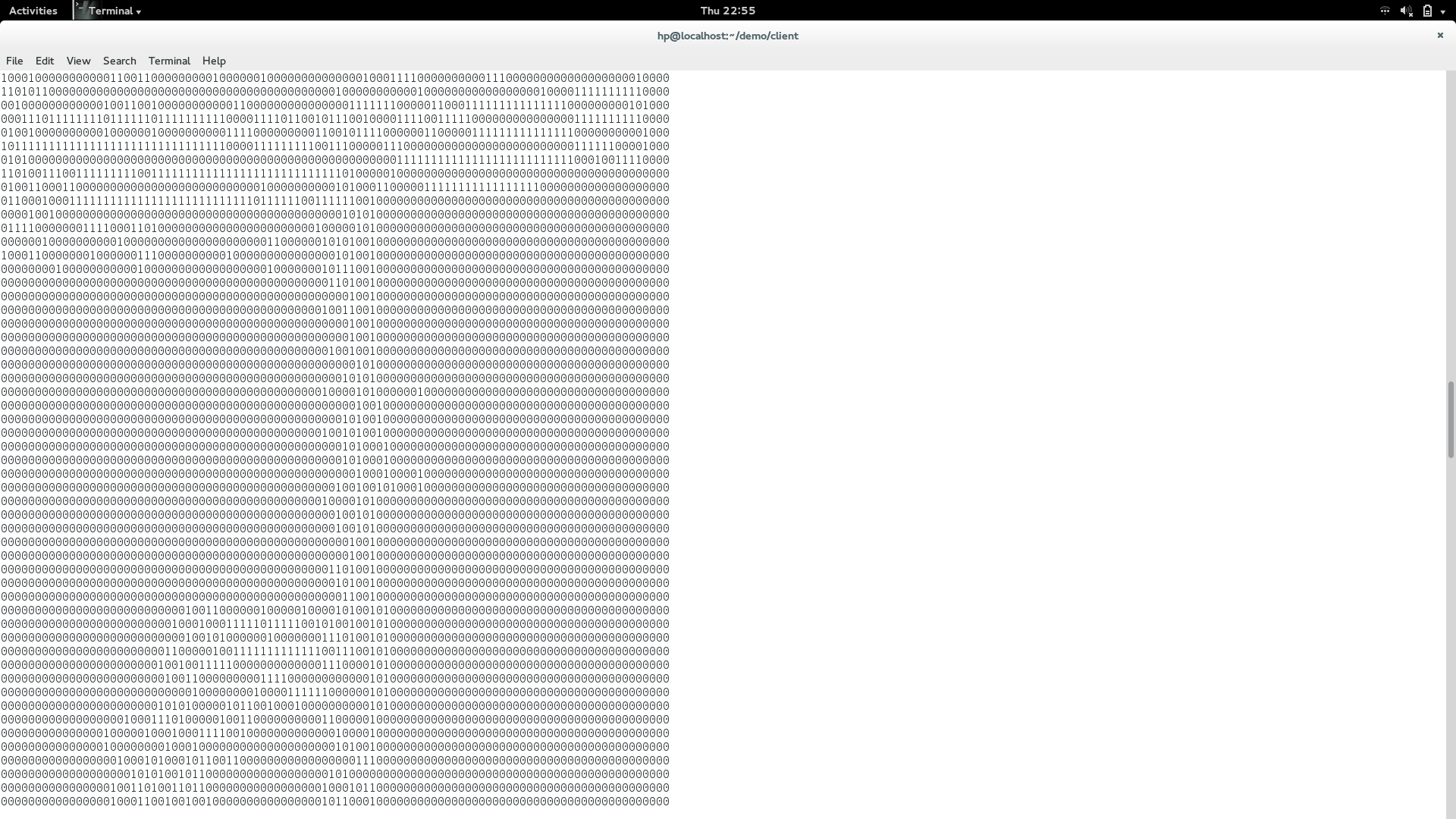
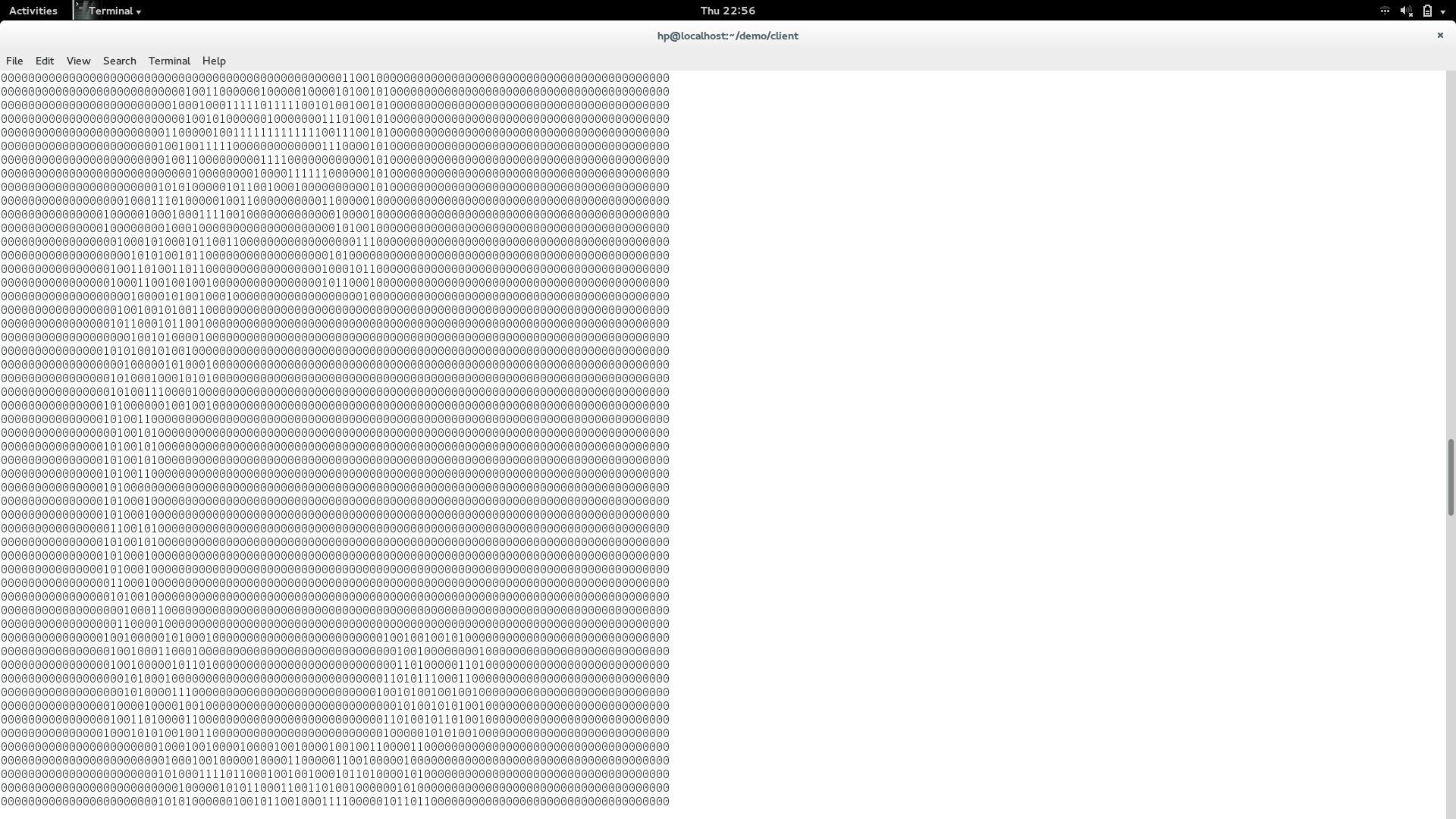
****

**Auto-scale**

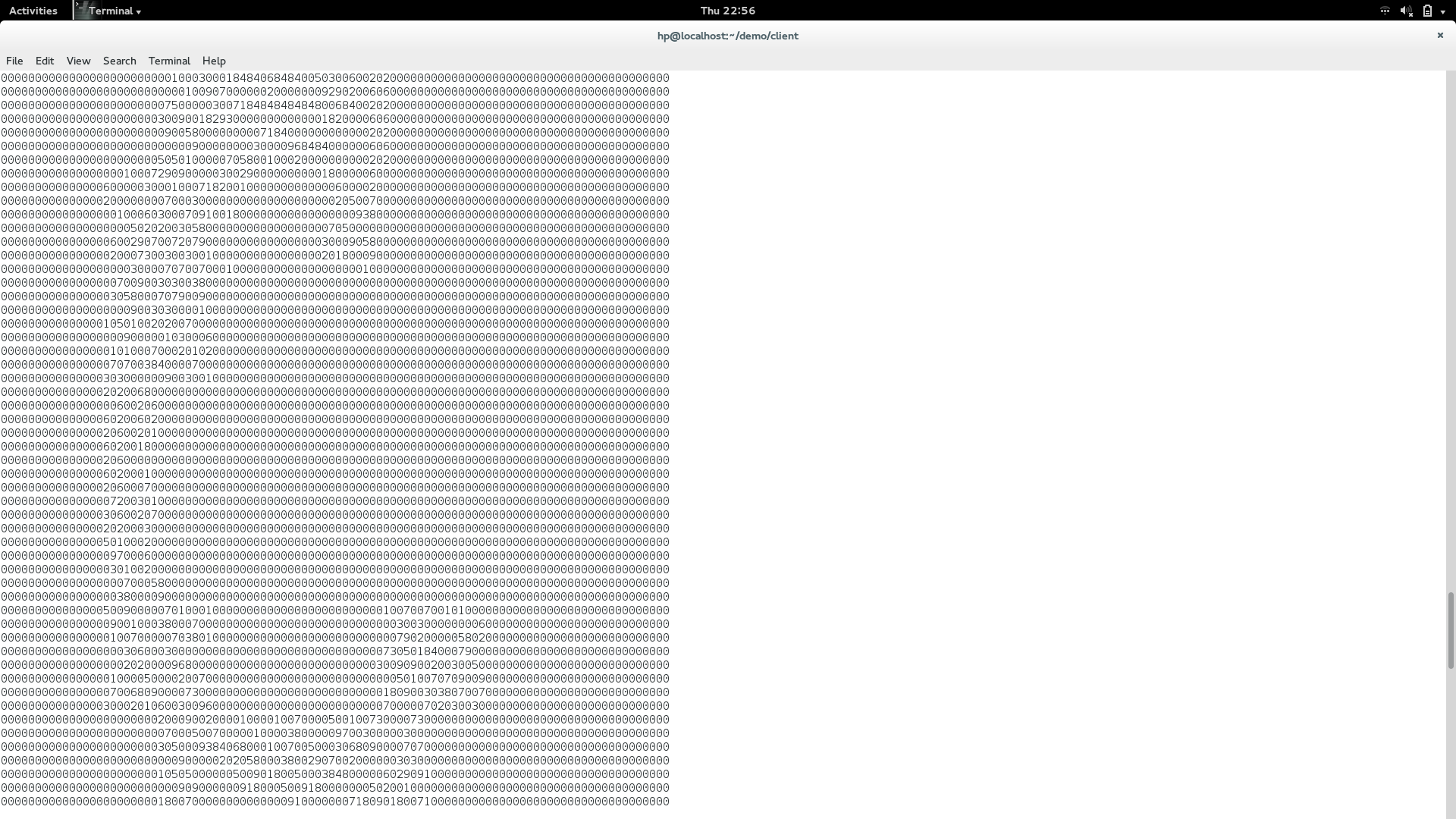
1. Conversion of the image into a matrix of pixels.



1. Border Detection : Next novel density based approach is used to detect the border of the pre processed character.

****

1. Chain coding : The chain coding approach is applied to the border of the character to get the direction and store them in a csv file to create a datastore.



1. Mapping : Finally the input character is mapped with the training set. In this a function is applied to calculate which is the closest character to the input image and the output is displayed as the recognised character.



**ALGORITHM**

**Input: Image of handwritten Character**

**Output: Recognised Character**

Densitybasedclustering(){

do

if neighbour>7

border object

else

consider it as core object

until last pixel is reached

}

Chaincoding(){

do

detect next connected point

assign direction code

until first pixel is reached

}

Mapping(){

Do

calculate differences with every sample

multiply with direction code

sum up all points

Until end of file

Identify min of sum

Display output character

}

**CHAPTER 8: SUMMARY AND CONCLUSION**

* **Summary**

Everyone has a different style of writing Devnagari alphabet. Recognizing these various styles to determine the characters using a large database is the main goal of this research project. We present a hand written character recognition system. It is implemented using novel density based clustering algorithm. Image processing, Data Mining, Database processing is done so as to achieve the goal and accuracy. The system is divided in stages. First one is taking the photo of the hand written character and uploading it. Second one is image processing. In that we do scaling or thinking if necessary. We perform noise removal. Third stage is applying border detection algorithm based on density based clustering approach. Then once these borders are identified chain coding is performed on these border objects. After this the values of the input are compared with the database and then they are matched. If matched the input character is recognised. Through this we have tried to gain accuracy as much as possible. In this project single character recognition is done it can be extended forward to words and sentences.

* **Conclusion**

There are hundreds of methods available in Image processing literature to recognize handwritten characters. However challenges to recognize Devnagari handwritten characters are still not completely addressed. Thus in this project we tried to use all together different techniques than mere image processing techniques to tackle this challenge. We used Novel density based clustering approach to separate pixels in the image of character. The standard density based clustering approach categorizes and store data objects as core objects and boundary objects. But in this algorithm we only require the boundary objects for further computation. We modified this approach to recognize the character in an image and also made it very efficient by simply extracting the boundary objects of the input character image. Using these boundary objects we apply chain coding on these objects.

Further, we tried to create a sample database of devanagri characters by writing them in different styles which we will make it available for further innovators on web for free download. In this project we have taken efforts to recognize the characters and obtain maximum accuracy and efficiency. This technique could also be extended to other Indian or foreign languages. It can also be pronounced as Automatic Character Generation.

**APPENDIX A:**

**LABORATORY ASSIGNMENTS ON PROJECT**

* **Concept of Idea Matrix**

|  |  |  |  |
| --- | --- | --- | --- |
| **I** | **D** | **E** | **A** |
| **Increase:**  -Speed and Efficiency  **-**Accuracy of individual character recognition  -Scope of compound letters | **Drive:**  -Recognition of ancient historical documents for preserving it  -Computerize the handwritten image. | **Educate:**  -End users  -Common people | **Accelerate:**  -Recognition process |
| **Improve:**  -Accuracy of recognition  -Scope of recognition of characters by increasing the number of training sets. | **Deliver:**  -Recognition of hand written character  -Reliability | **Evaluate:**  -Knowledge Opportunities  -Innovation potential | **Associate:**  -Common man  -Developer  -End users |
| **Ignore:**  -Font and font size of the letter  -Different handwriting styles for that character. | **Decrease:**  -Errors in detecting characters  -Time consumed for recognition of characters | **Eliminate:**  -Paper dependency  -External interface dependencies  -Compound characters | **Avoid:**  -Using Standard library functions |

* **Complexity Class of Problem Statement**

Project problem statement feasibility assessment using NP-hard and NP-Complete.

**Classes of Problems**

* P- Class problems:

The class of polynomially solvable problems, P contains all sets in which membership may be decided by an algorithm whose running time is bound by a polynomial.

* NP- Class Problems:

The class of non-deterministic polynomially acceptable problems, NP, contains all sets in which membership can be verified in polynomial time. Problems that can be solved using super polynomial exponential time algorithm are called intractable.

* NP-Complete Problems:

NP problems for which we can design decision problems are called NP-Complete problems. Decision problems have output either 1 or 0.

* NP-Hard problems:

Problems for which it is not possible to design decision problem are called NP-Hard problems. If NP-Hard problem can be solved in polynomial time, then all NPC problems can be solved in polynomial time.

In our project, we are devising density based clustering algorithm. It has complexity O (n logn) which is a polynomial time complexity. Therefore density based clustering is P class problem. Also, all the algorithms used for implementing image processing techniques are solvable in polynomial time. Therefore, our system falls under P class problem.

* **UML Diagrams**

1)Input character as an image

****

2)Perform crop,scaling and Rgb to gray conversion on the input image.



3)Perform Density based clustering on the image.

1. The input of this module is a Devnagari character and output is to check whether the character has been recognised .



* **Mathematical Model**

S= {s, e, X, Y, DD, NDD, MEMShared, CPUcorecount, Success, Failure}

Where,

S= Start State

e= End State

X= Set of Inputs

Y= Set of Outputs

F= {Fme, Ffriend}

Fme= Kernel Functions

Ffriend= Associate Functions

DD= Deterministic Data

NDD= Non Deterministic Data

MEMShared= Shared Memory

CPUcorecount= No. Of CPU cores

Success= No. of inputs for which the desired outcome is generated

Failure= No. of inputs for which the desired outcome is not generated

For our project,

S= {S1, S2, S3, S4, S5}

Where,

S1 = {To scale the image}

S2 = {To crop the image and remove noise}

S3 = {To convert RGB to gray image}

S4 = {To convert gray scale image to binary matrix}

S5 = {To map the obtained binary matrix to the matrix in the training set}

S1, S2, S3, S4, S5 are subsets of S

For S1,

1. s= {Init\_val\_regs|Init\_val\_regs is the value of registers on Power on or Reset}
2. e= Fexit()

=exit(0)

1. X= {Image from the user}
2. Y= {Y1, Y2}

Where,

{Y1} belongs to Success

{Y2} belongs to Failure

1. F= {Fme}

Where,

Fme= {Fk1}

Fk1= resizeImage()

The image is scaled using this function.

6) Success= {Y1}

Image of size 250x250

7) Failure= { Y2}

Y2 = The size of the input image is not as required

For S2,

1. s= {Init\_val\_regs|Init\_val\_regs is the value of registers on Power on or Reset}
2. e= Fexit()

=exit(0)

1. X= {Scaled image}
2. Y= {Y3,Y4}

Where,

{Y3} belongs to Success

{Y4} belongs to Failure

1. F= {Fme}

Where,

Fme= {Fk2}

Fk2= Autocrop

The image is cropped automatically and the noise is removed.

6) Success= {Y3}

Noise removed successfully.

7) Failure= { Y4}

Y4=Disjoint character is cropped.

For S3,

1. s= {Init\_val\_regs|Init\_val\_regs is the value of registers on Power on or Reset}
2. e= Fexit()

=exit(0)

1. X= {Uniform sized image containing no noise}
2. Y= {Y5, Y6}

Where,

{Y5} belongs to Success

{Y6} belongs to Failure

1. F= {Fme}

Where,

Fme= {Fk3}

Fk3= toGray()

It converts the RGB image into Grayscale.

6) Success= {Y5}

Grayscale image

7) Failure= { Y6}

Y6 = Grayscale image not obtained.

For S4,

1. s= {Init\_val\_regs|Init\_val\_regs is the value of registers on Power on or Reset}
2. e= Fexit()

=exit(0)

1. X= {GrayScale Image}
2. Y= {Y7, Y8}

Where,

{Y7} belongs to Success

{Y8} belongs to Failure

1. F= {Fme}

Where,

Fme= {Fk4}

Fk4= img2Matrix()

Converts the image into matrix form.

6) Success= {Y7}

Matrix having values 0 or 1.

7) Failure= { Y8}

Y8 = Matrix formation unsuccessful.

For S5,

1. s= {Init\_val\_regs|Init\_val\_regs is the value of registers on Power on or Reset}
2. e= Fexit()

=exit(0)

1. X= {Matrix}
2. Y= {Y9, Y10}

Where,

{Y9} belongs to Success

{Y10} belongs to Failure

1. F= {Fme}

Where,

Fme= {Fk5}

Fk5= mapping()

Maps the image to the training set.

6) Success= {Y9}

Character recognition successful.

7) Failure= { Y10}

Y10 = Not able to map the character correctly.

Ffriend={ FAS1,FAS2}

FAS1= accept()

Accept a handwritten Devnagari character as the input

FAS2= display()

Display the standard image as the result

1. DD= {Standard size of the image, type of the image, Database}
2. NDD={ }
3. Success= {Y11}

Character recognized correctly.

1. Failure= { Y12, Y13, Y14, Y15}

Y12= The system failed to recognize the character.

Y13= The image is not of the required specification.

Y14= The system recognized the character but, displayed the wrong result.

Y15= The record is not in the database.

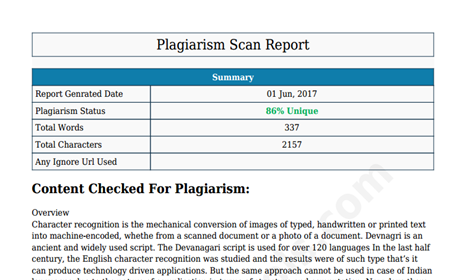
**APPENDIX C**

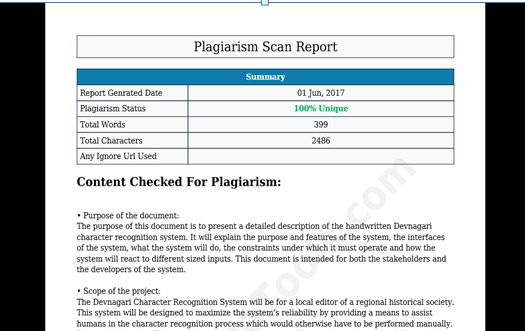
**PROJECT PLANNER**

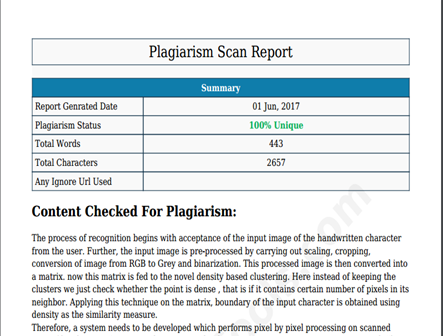
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr. No | Task/Activity | Month/Period | Expected Deliverables | Work done/Target achieved |
| 1 | Project related session | 3rd year | To understand the basic project expectations according to the curriculum | Yes |
| 2 | Project approval process | 2nd week of July | NA | NA |
| 3 | Internal mentor allocation | Last week of august | NA | NA |
| 4 | Project description and problem statement | 1st week of September | To define the project outline and designing the flow of the system | Yes |
| 5 | Learning image processing functions in java and also density based clustering | 2nd and 3rd week of September | All the java functions for image transformation process | Yes |
| 6 | Performing image transformation on the input | 4th week of September | Convert the image into matrix form and gray scale | Yes |
| 7 | Applying density based clustering on the input image | 1st week of October | To form clusters based on the pixel data received from the input image | Partially , as the algorithm was working only for one image hence new algorithm has to be applied |

**APPENDIX E**

**PLAGIARISM REPORT**

****

****

****

**APPENDIX H**

**Glossary**

1. Clustering : Clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters).
2. Density Based Clustering : In density-based clustering, clusters are defined as areas of higher density than the remainder of the data set. Objects in these sparse areas, that are required to separate clusters are usually considered to be noise and border points. It only connects points that satisfy a density criterion, a cluster consists of all density-connected objects.
3. Outliers : It is an observation point that is distant from other observations.An outlier may be due to variability in the measurement or it may indicate experimental error.
4. Data mining :  It is the computational process of discovering patterns in large [data sets](https://en.wikipedia.org/wiki/Data_set) involving methods at the intersection of [artificial intelligence](https://en.wikipedia.org/wiki/Artificial_intelligence), [machine learning](https://en.wikipedia.org/wiki/Machine_learning), [statistics](https://en.wikipedia.org/wiki/Statistics), and [database systems](https://en.wikipedia.org/wiki/Database_system). The overall goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use.
5. Database : A database is an organized collection of [data](https://en.wikipedia.org/wiki/Data_(computing)). It is the collection of [schemas](https://en.wikipedia.org/wiki/Database_schema), [tables](https://en.wikipedia.org/wiki/Table_(database)), [queries](https://en.wikipedia.org/wiki/Query_language), reports, [views](https://en.wikipedia.org/wiki/View_(SQL)), and other objects. The data are typically organized to model aspects of reality in a way that supports [processes](https://en.wikipedia.org/wiki/Process_(computing)) requiring information.
6. Image Processing :  Image Processing is processing of images using mathematical operations by using any form of [signal processing](https://en.wikipedia.org/wiki/Signal_processing) for which the input is an image, a series of images, or a video, such as a [photograph](https://en.wikipedia.org/wiki/Photograph) or [video frame](https://en.wikipedia.org/wiki/Video_frame); the output of image processing may be either an image or a set of characteristics or [parameters](https://en.wikipedia.org/wiki/Parameter) related to the image.
7. Scaling :  It is a [linear transformation](https://en.wikipedia.org/wiki/Linear_transformation) that enlarges (increases) or shrinks (diminishes) images by a [scale factor](https://en.wikipedia.org/wiki/Scale_factor) that is the same in all directions.
8. Binarization : It is the process of converting a pixel image to a binary image.

**REFERENCES:**