BHIIRAR UNIVERSITY INSTITUTE OF TECHNOLOGY



School of Computing and Electrical Engineering

Department of Computer Science and Engineering

Course: Artificial Neural Network

Proposal Document on: Ge’ez Number Optical Character Recognition

Group members:

Awel Abdo 056/2000

Bekele Haile 069/2000

Dawit Abrha 103/2000

Elzabeth Bahta 239/1999

Habtamu Worku 164/2000

Contents

[1. Introduction 3](#_Toc293289188)

[2. Problem description 4](#_Toc293289189)

[3. Objective 6](#_Toc293289190)

[4. Scope and limitation 7](#_Toc293289191)

[5. Methodology 9](#_Toc293289192)

[Detail solution approach for the problem 9](#_Toc293289193)

[6. Conclusion and future work 11](#_Toc293289194)

# Introduction

An Artificial Neural Network is a network of many very simple processors ("units"), each possibly having a (small amount of) local memory. The units are connected by unidirectional communication channels ("connections"), which carry numeric (as opposed to symbolic) data. The units operate only on their local data and on the inputs they receive via the connections.

The design motivation is what distinguishes neural networks from other mathematical techniques: A neural network is a processing device, either an algorithm, or actual hardware, whose design was motivated by the design and functioning of human brains and components thereof.

There are many different types of Neural Networks, each of which has different strengths particular to their applications. The abilities of different networks can be related to their structure, dynamics and learning methods.

Neural Networks offer improved performance over conventional technologies in areas which include: Machine Vision, Robust Pattern Detection, Signal Filtering, Virtual Reality, Data Segmentation, Data Compression, Data Mining, Text Mining, Artificial Life, Adaptive Control, Optimization and Scheduling, Complex Mapping, Speech Recognition, Face Recognition, ROC (optical character recognition) and more.

Character recognition, usually abbreviated to optical character recognition or shortened OCR, is the mechanical or electronic translation of images of handwritten, typewritten or printed text (usually captured by a scanner) into machine-editable text. We are dealing with the Ge’ez numbers optical character recognition meaning the characters used as input are of Ge’ez numbers.

It is a field of research in pattern recognition, artificial intelligence and machine vision. Though academic research in the field continues, the focus on character recognition has shifted to implementation of proven techniques. For many document-input tasks, character recognition is the most cost-effective and speedy method available. And each year, the technology frees acres of storage space once given over to file cabinets and boxes full of paper documents.

Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans 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 analyze.

# Problem description

Before optical character recognition can be used, the source material, that is the “Ge’ez numbers” must be scanned using an optical scanner to read in the page as a bitmap (a pattern of dots).

Software to recognize the images is also required. The character recognition software then processes these scans to differentiate the “Ge’ez numbers” and determine what are represented in the light and dark areas. The OCR engines add the multiple algorithms of neural network technology to analyze the stroke edge, the line of discontinuity between the characters, and the background. Allowing for irregularities of printed ink on paper, each algorithm averages the light and dark along the side of a stroke, matches it to known characters and makes a best guess as to which character it is.

Neural networks can be used, if we have a suitable dataset for training and learning purposes. Datasets are one of the most important things when constructing new neural network. Without proper dataset, training will be useless. So first we have to scan the numbers. After that, we will define processing algorithm, which will extract important attributes from the numbers and map them into a database. Extracted attributes will have numerical values and will be usually stored in arrays. With these values, neural network can be trained and we can get a good end results. The problem of well defined datasets lies also in carefully chosen algorithm attributes. Attributes are important and can have a crucial impact on end results. The most important attributes for handwriting algorithms are:

1. Negative image of the number, where the input is defined as 0 or 1. 0 is black, 1 is white, values in between shows the intensity of the relevant pixel.
2. The horizontal position, counting pixels from the left edge of the image, of the center of the smallest rectangular box that can be drawn with all "on" pixels inside the box.
3. The vertical position, counting pixels from the bottom, of the above box.
4. The width, in pixels, of the box.
5. The height, in pixels, of the box.
6. The total number of "on" pixels in the number image.
7. The mean horizontal position of all "on" pixels relative to the center of the box and divided by the width of the box. This feature has a negative value if the number is "left heavy".
8. The mean vertical position of all "on" pixels relative to the center of the box and divided by the height of the box.
9. The mean squared value of the horizontal pixel distances as measured in 6 above. This attribute will have a higher value for images whose pixels are more widely separated in the horizontal direction.
10. The mean squared value of the vertical pixel distances as measured in 7 above.
11. The mean product of the horizontal and vertical distances for each "on" pixel as measured in 6 and 7 above. This attribute has a positive value for diagonal lines that run from bottom left to top right and negative value for diagonal lines from top left to bottom right.
12. The mean value of the squared horizontal distance tunes the vertical distance for each "on" pixel. This measures the correlation of the horizontal variance with the vertical position.
13. The mean value of the squared vertical distance times the horizontal distance for each "on" pixel. This measures the correlation of the vertical variance with the horizontal position.
14. The mean number of edges (an "on" pixel immediately to the right of either an "off pixel or the image boundary) encountered when making systematic scans from left to right at all vertical positions within the box.
15. The sum of the vertical positions of edges encountered as measured in 13 above. This feature will give a higher value if there are more edges at the top of the box.
16. The mean number of edges (an "on" pixel immediately above either an "off pixel or the image boundary) encountered when making systematic scans of the image from bottom to top over all horizontal positions within the box.
17. The sum of horizontal positions of edges encountered as measured in 15 above.

# Objective

The objective of this project is to identify handwritten characters of Ge’ez numbers with the use of neural networks. We have to construct suitable neural network and train it properly by using suitable learning architecture, algorithm, and paradigm. The program should be able to extract the characters one by one and map the target output for training purpose. After automatic processing of the image, the training dataset has to be used to train classification engine for recognition purpose. The program code has to be written in MATLAB and supported with the usage of Graphical User Interface (GUI).

The system will able to solve the Ge’ez character (numbers) recognition, usually abbreviated to optical character recognition or shortened OCR, is the mechanical or electronic translation of images of handwritten, typewritten or printed text (usually captured by a scanner) into machine-editable text.

Recognizing some of the Ge’ez numbers from 1 to 100 numbers (፩ ,፪, ፫, ፬ ,፭, ፮ ,፯, ፰ ,፱,፲, ፳,፵, ፶,፷,፸,፹, ፺, ፻).

This will help for a wide usage of the Ge’ez numbers.

Providing a simple and a Ge’ez numbers recognition this recognition can come up with a better job such as the recognition of Amharic, Tigrigna, Ge’ez and other Semitic languages which use the same letters.

Showing that any language character recognition problems can be solved easily using the artificial neural network.

Ge’ez text to audio converter can be developed with the help of the Ge’ez number (character) recognition system.

This project will motivate every person to come up with a better job for the development of our own Ethiopian languages.

# Scope and limitation

The scope for this type of project can be classified like the following:

**Input image scope:**

To facilitate scanning of input hand written geez number character, the paper which holds the characters should have the following characteristics:

* The paper should be strong, white and clean to get clear scanned input characters.
* The paper size should preferably be A4, viz. 210 mm x 297 mm or 8 1/2 by 11 inches (which is the de-facto North-American standard).
* Sheets should be free of creases, holes and should not be rolled.
* The paper should not be absorbent in order to avoid smearing of the ink
* Double-sided writing of the paper should be avoided.
* The characters should be solid black on a white background.
* A minimum margin of 2 centimeters should be present at the top, bottom and sides of each sheet, and a minimum margin of 2.5 centimeters on the left side of each sheet.
* Each number character must write clearly and neatly with enough spacing between them.
* The written character must not be underlined for best clarity.
* The scanned image data must be converted to bmp format for the following reasons
* Bitmap graphics should be created or scanned either at, or very close to, the size required for the finished product.
* You can’t successfully change the size of the image in your word processor. You need to load the image into a suitable program and re-size it by re-sampling
* If you do need to re-size an image, avoid drastic changes in size.
* When re-sizing an image, always use a program that has some form of smart sizing feature and make sure that it is turned on.
* If you are creating a large image, you will need plenty of memory available
* The geez number which is written on the paper sheet must at list include the following numbers for training purpose:

፩ ,፪, ፫, ፬ ,፭, ፮ ,፯, ፰ ,፱,፲, ፳,፵, ፶,፷,፸,፹, ፺, ፻

The scanned image data must be pre processed before feed to the neural network system. This scanned image data must pre process by the following specification:

The pre process is done by dividing the image using its pixel; each number character is divided horizontally in 50 pixels and vertically 70 pixels.

Based on the divided pixel we must identify the shadow part form light shadow and empty parts.

It will not accept values which are different from the above specified characters, if there is it will discard and work on other.

The input layer neurons are determined by the number input parameter which is sample for our condition as listed above like(፩ ,፪, ፫, ፬ ,፭, ፮ ,፯, ፰ ,፱,፲, ፳,፵, ፶,፷,፸,፹, ፺, ፻)

**Output text scope:**

* The output text must be understandable by the abanet, power geez, visual geez Unicode or other font type.
* The size must be 12 or 14 point as a default value.
* Then we can modify, update and edit the file

# Methodology

The way to develop this system it may have a lot of schemes and ways. But the appropriate method that we choose is as follows:-

* Fist we preprocessed the input data to become appropriate format
* After that we choose the appropriate architecture and learning algorithm which is convent for this problem
* Then the preprocessed data should we divided in to two parts. That means two third (2/3) the input data for training and one third (1/3) of the data for testing

Detail solution approach for the problem

To solve the defined handwritten character recognition problem of classification we used MATLAB computation software with Neural Network Toolbox and Image Processing Toolbox add-on. And

The computation steps are divided into the next categories:

1. **Automatic Image Preprocessing**
* The image is first being converted to grayscale image follow by the threshing technique, which make the image become binary image. The binary image is then sent through connectivity test in order to check for the maximum connected component, which is, the box of the form. After locating the box, the individual characters are then cropped into different sub images that are the raw data for the following feature extraction routine.
* The size of the sub-images are may not fixed since they are expose to noises which will affect the cropping process to be vary from one to another. This will causing the input of the network become not standard and hence, prohibit the data from feeding through the network. To solve this problem, the sub-images have been resize by certain value and then by finding the average value in each by certain size blocks, the image can be down to N by M matrices, with fuzzy value, and become L inputs for the network. However, before resize the sub-images, another process must be gone through to eliminate the white space in the boxes.
1. **Feature Extraction**
* The sub-images have to be cropped sharp to the border of the character in order to standardize the sub-images. The image standardization is done by finding the maximum row and column with 1s and with the peak point, increase and decrease the counter until meeting the white space, or the line with all 0s.
* The we have to Read Image through matlab space and Convert to grayscale image
* Here ,Now the greater work is Convert the image in to binary image
1. **Neural Network Training**

Then after we perform the rest of procedure we start to train the network and testing. This procedure includes:

* Creating Vectors data for the Neural Network (objects)
* Creating and training of the Neural Network
* Encoding/Decoding

# Conclusion and future work

Handwritten character recognition is a complex problem, which is not easily solvable. There may be a lot of problem lies in pre-processing of data. Described application of character recognition can be divided into three main parts.

* Image preprocessing to get the training data, training the neural network and at the end testing with final recognition results.
* Image preprocessing to get the training data for the neural network is based on input training.
* Training and testing the neural network was only a matter of two MATLAB commands. We decided to use Multilayer Perceptron with two hidden layers.

Future work will extend the OCR technology towards the recognition of other applications such as Amharic text to audio format converter, Amharic optical character recognition, Ge’ez numbers to audio converter.