*International Journal of Scientific Engineering and Applied Science (IJSEAS) – Volume-3, Issue-1,Jan 2017*
*ISSN: 2395-3470*
*www.ijseas.com*

# Recognizing ancient Sinhala Inscription Characters using Neural Network Technologies

**K.G.N.D. Karunarathne[1], K.V. Liyanage[2], D.A.S. Ruwanmini[3], G.K.A. Dias[4], S.T. Nandasara[5]**
[1,2,3,4,5] University of Colombo School of Computing, Colombo, Sri Lanka

## Abstract

Recognizing ancient Sinhala inscription characters enable archeologists to reveal historical events in ancient Sri Lanka. Currently, this is done by the archaeology experts with a huge effort. The inefficiency of this manual procedure will negatively impact on the future research in field of archaeology. This research involves in developing an application with Optical Character Recognition (OCR) functionality to recognize ancient Sinhala inscription. This paper focus on the OCR module of the application. OCR module comprises of the technologies of Artificial Neural Network (ANN) and Convolutional Neural Network (CNN). Experiments were carried out to evaluate the recognition rate of the two OCR technologies which performs on train data, test data (preprocessed) and test data (real images). After evaluating each OCR solution, CNN was selected as the best resulted OCR solution. Lack of data is the main limitation of this research and it will be highly impacted on the OCR accuracy. As a result, 9 characters were identified by the CNN OCR engine.

***Keywords:*** *Epigraphy, Sinhala Inscriptions, Optical character recognition (OCR)*

## 1. Introduction

Many Inscriptions found in ancient cities of Anuradhapura and *Polonnaruwa* in Sri Lanka. *Thonigala* inscription, Mirror wall , *Galpotha* (Stone Book) inscription (see Fig. 1.1) are some of them.

Figure 1.1: *Galpotha* (Stone Book) Inscription in *Polonnaruwa*

These inscriptions are very important as they are the major sources of getting information about ancient Sri Lanka. These inscriptions also provide valuable information about the time, place and situation connected with the inscription and the evolution of the languages over centuries. [1][2]

Revealing the content of these inscriptions will be highly valuable to investigate the history of ancient Sri Lanka. Currently the content of each of these inscriptions are translated to modern Sinhala language manually by an archaeology expert who has specialized knowledge to understand the ancient scripts. The Inscriptions letters are read through human eye with great difficulty and this manual procedure would be time consuming.

Although the Inscriptions were used as a one of the information source to recognize expansion of Sinhala language, recognizing content of these inscription becomes a huge challenge due to various reasons. They may be damaged and/or partially erased. Lack of specialized knowledge and lack of available resources for inscription reading are also another major problem. Currently, the existing archaeology experts have to make considerable amount of effort to read inscriptions. Further they also have to recognize the characters of these inscriptions manually.

The main aim of the research is to recognize each isolated character set of these inscriptions using a proper character recognition process and map them into modern Unicode characters.

### 1.1 Research Objectives

The following are the objectives of our research.

- Recognize the ancient characters and digitize them.
- Map ancient character identified with modern Sinhala character
- Obtain Sinhala interpretation of Sri Lankan inscriptions via the character recognition feature.

### 1.2 Research Methodology

As mentioned the above section, evaluating feasible OCR solutions will be taken as the initial step of this research project. Before that, it is essential to prepare a proper dataset. In that sense research methodology is broken down into three steps as mentioned below.

Step 1: The first step refer to creating a database for training using a set of ancient characters.

Step 2: Try out different feasible OCR solutions and evaluate results of each. Each OCR solution will follow the character recognition procedure of preprocessing, feature extraction and character recognition. Here we will mainly focus finding a better OCR technique which has a high recognition rate.

Step ***3:*** Finally, in the third step, best resulted OCR solution is incorporated into the system.

## 2. Literature Review

### 2.1. Introduction to the Review

This review undertakes the responsibility of sharing the knowledge about some character recognition, Optical Character Recognition research done.

### 2.2. Review of Literature

#### 2.2.1. Inscription Reading Manual Methods

According to the article done at RootWeb.Ancestry.com about Alternative Gravestone Reading Methods (2015) [3]; Hand Rubbing is possible on a uniformly colored stone surface. One has to brush lightly the surface of the inscription with the palm of hand, which raises a light dust and leaves the recessed inscription as a dark color. Another method is by using a mirror to direct bright sunlight diagonally across the face of a gravestone so that it can easily cast shadow in indentations. This will make inscriptions much more visible and easy to read. Another approach is by using a viewing tube which is held against the stone to prevent light entering and then tilt the end of the tube touching the stone slightly thereby a little light enters and then views the inscription through the tube.

However currently researchers pay much attention on computerizing the character recognition task to achieve more efficiency and convenience.

#### 2.2.2. Use of Optical Character Recognition (OCR)

Publications focus on OCR are summarized below.

P.Niranjan (2015) [4], P.Nikhi, V.Jayakumar, S.kolkure (2015) [5] and Yasser Alginahi (2004) [6] have discussed in their research papers about Optical character Recognition process and technologies.

Hubert Mara, Jan Hering and Susanne Kromker (2014) [7] use Optical Character Recognition (OCR) in their research to read ancient Chinese texts which are carved into stones. An automated system was developed by them for processing an ancient Chinese inscriptions (sutras).

S.Rajakumar, V.Subbiah Bharathi (2011) [8] presented a methodology for Century Identification and Recognition of ancient Tamil Characters. In this paper they have addressed the issues of Tamil character recognition, such as difficulties in understanding Tamil characters. These characters are different from current century's Tamil characters. Therefore, this research paper presents a way of ancient Tamil character recognition in inscription using MATLAB with contourlet transform method. A neural network has been used to train the image and compare the data with the characters of the current centary.

Abhishek Tomar , Minu Choudhary , Amit Yerpude (2015) [9] conducted a survey research focused on Indian inscription character recognition areas by understanding the script and the languages in the images and carried out prehistoric inscription analysis.

H. K. Anasuya Devi (2006) [10] has conducted a research regarding preprocessing algorithms of an OCR system to read the Brahmi script. The researcher has presented various algorithms which were used in low level processing stage of image analysis for Optical Character Recognition.

Sachin S Bhat, H.V. Balachandra Achar (2015) [11] have conducted a research with the purpose of period identification of various ancient Kannada scripts using advanced recognition algorithms. They have proposed an algorithm including basic components of the images processing namely image acquisition, noise removal, and segmentation of character sets for feature extraction, classification and recognition of segmented characters. The main specialty of this research is that they have developed a system to predict the era by examining a few characters in Kannada inscription. Their experiments have been done using MATLAB and they have mentioned that they can achieve 80% of accuracy and efficiency of the final results.

Ayatullah Faruk Mollah , Nabamita Majumder , Subhadip Basu and Mita Nasipuri (2011) [12] have conducted a research to design a complete Optical Character

Recognition system for camera captured image/graphics embedded textual documents for handheld devices such as high-end cell-phones, Personal Digital Assistants (PDA), smart phones, iPhones, iPods, etc.

Indu Sreedevi, Rishi Pandey, N.Jayanthi, Geetanjali Bhola and Santanu Chaudhury (2013) [13] have conducted a research on NGFICA (Natural gradient-based flexible Independent Component Analysis) based on Digitization of Historic Inscription Images. This paper addresses the problems encountered during digitization and preservation of inscriptions such as perspective distortion and minimal distinction between foreground and background. This paper proposed a method to enhance the minimal difference between text and the background of the inscription image. For minimizing the dependency among the foreground and background of historical inscription images, they have used NGFICA to obtain independent components of the images. The proposed method improves word and character recognition accuracies of the OCR system by 65.3%.

G.Vamvakas, B.Gatos, N. Stamatopoulos, and S.J.Perantonis [14] in their work presented an OCR methodology for recognizing historical documents, either printed or handwritten without any knowledge of the font. This methodology consists of three steps; the first two steps refer to creating a database for training using a set of documents, while the third one refers to recognition of new document images. They presented segmentation-free approach for recognizing old Greek handwritten documents especially from the early ages of the Byzantine Empire.

Luke V Rasmussen,Peggy L Peissig, Catherine A McCarty, Justin Starren (2012) [15] have done research to develop optical character recognition pipeline for handwritten form fields from an electronic health record(EHR). This research work was done under the two main phases. First one is focused on designing forms to capture hand-printed data specifically for OCR processing. Second one has utilized custom-developed OCR engines to perform the handwriting recognition.

Google's Optical Character Recognition (OCR) [16] software now works in more than 248 world languages (including all the major South Asian languages and Sinhala). The technology extracts text from images, scans of printed text and even from handwriting, which means text can be extracted from old books, manuscripts or images. When processing a document, it attempts to preserve basic text formatting such as bold and italic text, font size and type and line breaks. However, detecting these elements is difficult and the user may not always succeed. Other text formatting and structuring elements such as bulleted and numbered lists, tables, text columns and footnotes or endnotes are likely to get lost.

## 3. OCR Design approach

The Optical Character Recognition process can be divided into three major steps: [17]

1. Preprocessing
2. Character Recognition
3. Post processing

Figure 3.1 shows the whole architecture of Ancient Sinhala Character Recognition. In this section, we describe each step of character recognition process separately.

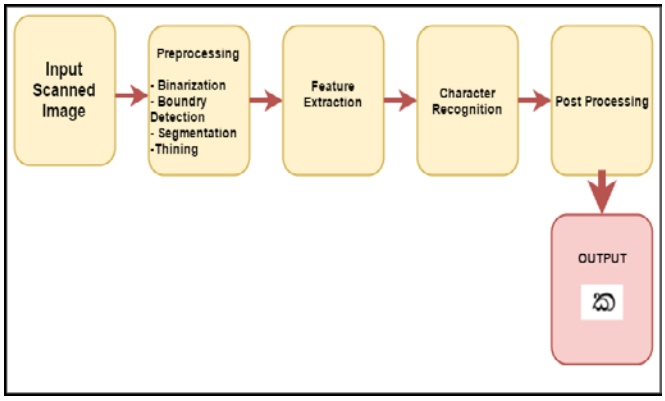


Figure 3.1: Character recognition process - Acquisition of image

Input scanned image is the pre-condition of the preprocessing process. Firstly original inscription image or digitized estampage image of input data is optically scanned. The scanned image can be any document of different dimensions. This scanned input image is fed to the pre-processing section for further analysis.

### 3.1 Preprocessing

Preprocessing the input images facilitate the increase of the recognition accuracy. OCR loads an image from a given source and performs different algorithms to clean the image. These algorithms apply techniques such as blurring, threshold to the image to reduce the noise. In this process it will enhance the visual appearance and the quality of ancient Sinhala characters in inscription images. Some different actions performed during pre-processing are listed below.

1. Binarization

2. Boundary Detection

3. Segmentation

4. Thinning

Following figure 3.2 shows the procedure used in OCR preprocessing stage.

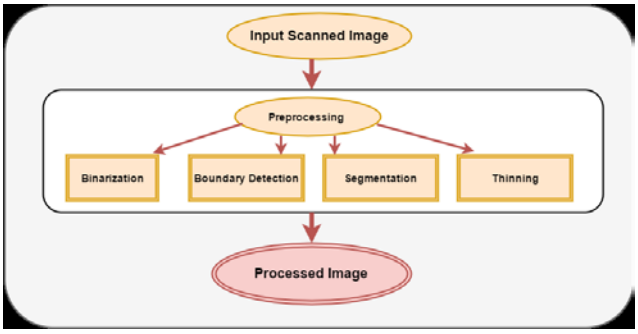


Figure 3.2: OCR Pre-processing procedure

### 3.1.1 Binarization

Binarization plays an important role in pre-processing. Binarization is a technique by which the gray scale images are converted into binary images. When converting a grayscale image into a binary image using thresholding, Otsu method (Otsu's method, named after Nobuyuki Otsu) was used to perform thresholding. Otsu's thresholding method involves in iterating through all the possible threshold values and calculating the measure of spread for the pixel levels from each side of the threshold, the pixels that either falls in foreground or background. Otsu's method automatically performs histogram shape-based image thresholding for the reduction of a gray-level image to a binary image. The algorithm assumes that the image for thresholding contains of two classes of pixels either foreground or background and then calculates the optimum threshold by separating those two classes so that their intra-class variance is minimal. [20,21]

### 3.1.2 Boundary Detection

The binarized image is now applicable for boundary detection with operation of dilation and erosion using OpenCV - Python. In this operation the boundaries of scanned image are detected. It is necessary to detect the boundaries so as to select an individual character. Therefore Dilation and Erosion are operations which increase or decrease objects in size and can be very useful during the preprocessing stage. Erosion makes an object smaller by removing or eroding away the pixels on its edges, dilation makes an object larger by adding pixels around its edges. [18]

### 3.1.3 Segmentation

Segmentation is done in order to improve the analysis of an image. This will be done by separating the pixels of an image according to semantic content and facilitating the manipulation and visualization of the data with a computer. Character detection is one of the main parts of the segmentation. Here all the characters' with middle zone consider as the character segmentation portion. The reason is when considering the inscriptions it may contain for eg. "කි" (ki), "කා" (kaa) , of that individual character "ක" (ka) with upper zone and lower zone. The main purpose of character segmentation is detecting individual character with middle zone. [18]

### 3.1.4 Thinning

Thinning is used to clean and visualize the skeleton of the scanned input image. This process deletes the dark points of the image which means remove selected foreground pixels from the binary image that approximate to the center skeletons of the regions. Thinning is used to infer shape in the original image. Consider the "ඒ" (a) Character. It consists basically of two strokes or line-link pieces (two horizontal) and two curves connected in a certain manner. The thickness of the strokes and curves are irrelevant to the recognition problem. What is important is the topology of how the strokes and curves are connected together. In such situations it is convenient to simplify the input as much as possible in order to make the topological analysis as simple as possible. In that sense, it is very important to obtain skeleton by thinning the character inside the image.

## 3.2 Character Recognition

After preprocessing the image document, OCR technology can begin to read and translate characters. Mainly Character Recognition consist of the basic stages of comparing the characters in the scanned image to the characters in the learned set involving:

- the procedure of extraction and isolation of each character from an image
- determined the properties of extracted character.
- comparison of the properties of extracted character and learned character.

As mentioned above these basic stages are named as feature extraction and classification.

### 3.2.1 Feature Extraction

In this stage, a set of features are extracted from the character. More features extracted for the character increases the probability of its recognition. Extracting the text data from images is important for reading, editing and analyzing the text context contained in the images. Computers cannot recognize the text data directly in images. Thus the design of computer program called "Optical Character Recognition" which are capable of recognizing text in images, play a vital role.

### 3.2.2 Classification

The extracted feature data must go through the process of classification. This step classifies the extracted individual character. Several approaches can be used to classify the characters such as clustering, using K-mean algorithm, etc.

### 3.3 Post Processing

After recognition process, post processing acts as the final stage. This is needed because sometimes the recognized character does not match with the original character. In some other cases, some characters could not be recognized from the input image. In post processing stage it mainly involves in error checking.

## 4.0 Methodology

This chapter describes the methodology of designing and implementation and also describes each and every algorithm which is used to develop the system. Before we implement the system for Sinhala inscription OCR, various OCR modules which are based on different techniques were tested. Every logical step is implemented and tested as separate modules and examined the outputs. Each module's results are explicitly analyzed and compared to find the best suitable OCR solution. Each technology based on OCR solution is discussed below. Those experimental technologies are template matching, ANN based OCR solution, and CNN based on OCR solution.

### 4.1 Preparing Training Dataset

It was a time-consuming task to find a proper dataset for this research. In this research, we are presenting the ancient Sinhala inscription character recognition and our OCR research is based on the source data of estampages. Therefore a new dataset is needed to be prepared. We had to request from the expertise in this specific industry to obtain these type of data. This is due to the fact that the estampages are obtained from the inscriptions by some authorized people in the government sector or the archaeology experts. As the general public, we couldn't obtain estampages directly from the inscriptions due to certain government restriction and protection rules on ancient properties.

### 4.2 Image Processing

Figure 4.1: Preprocessed image

Figure 4.1 shows a sample image and its corresponding preprocessed image. There are number of factors that will affect on obtaining better recognition rate through OCR engine. These factors are quality of the original source, scanner/camera quality, scan resolution, preprocessing techniques and so on [18].

In Image processing it is really important to obtain a cleaned /preprocessed image to get satisfactory recognition results. Hence various image processing techniques were applied to decrease the noise which is in the image. Here we digitize image data into two formats namely "bitmap" and "jpeg" because different tested OCR solutions require different formats. These Digitized images have been preprocessed using the image processing module which was developed on OpenCV python for all tested OCR solutions.

Figure 4.2 has shown the sample image at different preprocessing stages ,which was obtained using OpenCV functions.

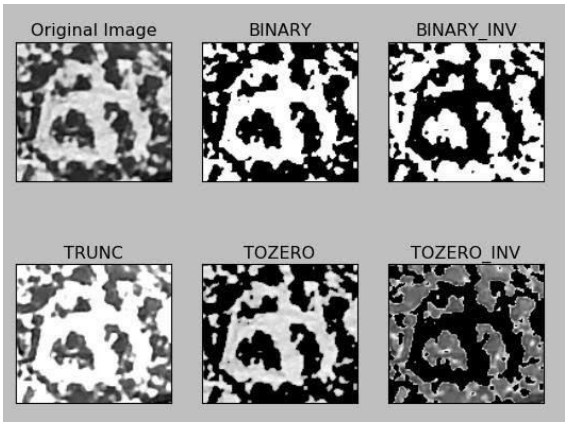

Figure 4.2 : Character preprocessing steps

## 4.3 OCR Techniques

OCR Techniques used for testing are briefly described below.

### 4.3.1 Template Matching

Template matching compares the pixel values in the template with the pixel values in the sub-region of an image. If the match is good then it would mention the particular objects which are present in the image. So template matching is the basic OCR technique to determine the similarities between template and region with the same size in an image that shows the highest similarity[18].

Basic template matching based on OCR is illustrated in Figure 4.3

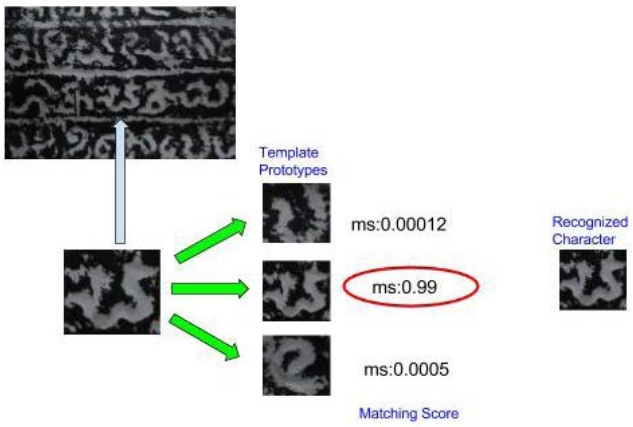


Figure 4.3 : Basic Template Matching based on OCR

To develop OCR solutions, feature extraction considered as one of the major steps. There are some commonly used feature extraction methods such as Histogram of Oriented Gradients (HOG), Speeded Up Robust Features (SURF), Local Binary Patterns (LBP), Haar wavelets, Color Histograms and etc. We used Speeded Up Robust Features methods [20], for the template-matching, since it is computationally much faster when comparing to the others.

After the feature extraction, classification is undertaken by comparing an input character image with a set of templates from each character class. After all the templates have been compared with the observed character image, the character's identity is assigned as the identity of the most similar template.

### 4.3.2 Artificial Neural Network based OCR Solution

4.3.2.1 Feature Extraction

The zoning technique is one of the feature extraction methods used to extract features in a character image. Every character image of size 100*100 pixels is divided into 10 equal zones, each of size 10x10 pixels. The features are extracted from every zone pixels by moving along with the diagonals of its respective 10x10 pixels. Each zone has a static number of diagonal lines and the foreground pixels present in each diagonal line is summed to get a single sub-feature. Then it is obtained the average value of these sub-feature values in order to form a single feature value for the corresponding zone. This procedure is sequentially repeated for each and every zone in the image. There could be certain zones whose diagonals are empty of foreground pixels. The feature values corresponding to these zones are zero. Finally, 100 number of features are extracted for each character. [21]

Figure 4.4 shows the extracted features from the isolated Character 'ක' (Ka).

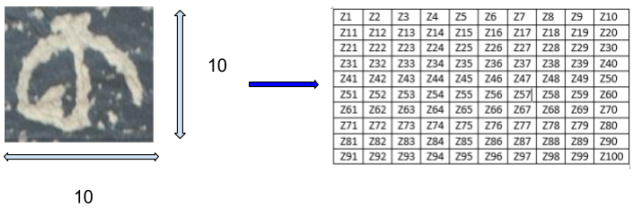


Figure 4.4 - The extracted features from the isolated Character 'ක' (Ka)

4.3.2.2 Character Recognition

We have followed most popular and simple approach for developing OCR engine based on feed forward neural network with back propagation learning. First we prepared and gave a training set and then train a neural network to recognize characters from the training set.

In the training process, we teach the network to respond with target output for a particular input. In that sense, it follows supervised learning. Each training sample is represented as possible input and target output for the

corresponding input. After the training process is done, we can give a new input to the network and it produces the output data.

*Back propagation learning:*

Input data is propagated through the network from the input layer to output layer to get the output. The output is compared with the target output and error is calculated. The error is used to calculate a new weight for a neuron which makes the difference between actual and desired output. The Back propagation algorithm undertakes when an error is propagated backward in the network. According to our observations, the Back propagation algorithm may cause problems when we train the same dataset over and over again when it loses its generalization capability.

### 4.3.2.3 Experiment done on ANN based OCR.

We train a network to recognize 30 ancient Sinhala inscription characters which convert into an input part of a training sample to create a vector of size 100, containing "1" in all positions corresponding to the letter pixel and "0" in all positions corresponding to the background pixels. Such sort of pattern coding will lead towards improvement of a greater learning performance. For each possible input, we create a desired network's output to complete the training samples. After having this training samples for all letters, we start to train our network. For the above task, we use one layer of neural network, because we try CNN with multilayers as a second module. Single layer network has 100 inputs corresponding to the size of input vector and 30 neurons in the layer corresponding to the size of the output vector. After completing neural network training procedure, pattern recognition task is provided. On each learning epoch, all samples from the training set are trained through the OCR engine and error is calculated. When the error becomes less than the specified error limit, then the training is done and the network can be used for recognition. Training data set runs on 12 iterations for recognizing. The output will be shown as computation code which means "ක" (ka) represented as "a40", "a41" likewise.

Figure 4.5 shows the predicted results of the classifier for the given 9 characters. When classifier finishes prediction, we can check its accuracy by printing the actual labels.

| Class label | Actual value | Recognition rate |
|---|---|---|
| A | "ක" | 82% |
| B | "ස" | 60% |
| C | "ප" | 72% |
| D | "ල" | 65% |
| E | "ම" | 71% |
| F | "ග" | 62% |
| G | "හ" | 55% |
| 7 | "ත" | 42% |
| 8 | "ය" | 40% |

Figure 4.5 : ANN based OCR prediction results

### 4.3.3. Convolutional Neural Network (library: keras) based OCR Solution

Convolutional Neural Networks (ConvNets or CNNs) are a type of Neural Networks that have proven very effective in areas such as image recognition and classification. A Convolutional Neural Network (CNN) contains of one or more convolutional layers and then followed by one or more fully connected layers as in a standard multilayer neural network. The main benefit of CNNs is that they are easier to train and have much less parameters than fully connected networks with the same number of hidden units. [19]

4.3.3.1 Feature extraction

The first conversion task is to extract features from the input image. Conversion assures the spatial relationship between pixels by learning image features using small squares of input data. When converting, every image is considered as a matrix of pixel values, the matrix is called a 'filter 'or 'kernel' or 'feature detector'. In CNN matrix is formed by sliding the filter over the original image and computing the dot product. This output matrix is called the 'Convolved Feature' or 'Activation Map' or the 'Feature Map'. This means filters act as feature detectors from the original input image. In that sense, different values of the filter matrix will produce different Feature Maps for the same input image. Therefore we prepared different filters from the input image and perform operations such as Edge Detection, Sharpen, and Blur just by changing the numeric values of our filter matrix before the convolution operation. Different filters can detect different features from an image, for example, edges, curves, lines etc.

During the training process, a CNN learns the values of these filters on its own. Here we are used to specify the number of filters, filter size for the training purpose. We practically learned that if we had a number of filters it tend to extract more features in the image and obtain higher recognizing rate. [19]

Non Linearity/ReLU is the second operation in feature extraction in CNN. It replaces all negative pixel values in the feature map by zero. Then in the pooling step in CNN reduces the dimensionality of each feature map keeping the most important information. Pooling is to progressively reduce the spatial size of the input representation and it facilitates more enhancement in the input representations. Such as making it smaller , more manageable, reduces the number of parameters and computations in the network, controlling over fitting, makes the network invariant to small transformations, distortions, and translations in the input image. Together all these layers extract the useful features from the original images. [19]

## Character Recognition.

CNN always has more layers. Those layers are input layer, convolutional layer, sample layer, and the output layer. And this is the main difference between ANN and CNN. The output from the convolutional and pooling layers in CNN represent high-level features of the input image. In a deep network architecture, the convolution layer and sample layer can have multiple layers. The purpose of the Fully Connected layer is to use these features for classifying the input image into classes based on the training dataset. Therefore a fully connected layer acts as a classifier. For the training purpose firstly we initialized all filters and weights with random values. [19]

In the training of the neural network, there is an additional layer called the loss layer. This layer provides feedback to the neural network indicating whether it identified inputs correctly or not, and if not, how far off its guesses were. This helps to guide the neural network to reinforce the right concepts as it trains. [19]

CNN Learning algorithm requires feedback. This is done by using a validation set where the CNN would make predictions and compare them with the true class labels. The predictions errors are fed backward to the CNN to refine the weights learned. This is called backward pass or back propagation of errors, and here we use different functions in CNN. [19]

## Experimented Results

Figure 4.6 shows the predicted results of the classifier for the given 9 characters. When classifier finishes the prediction, we can check its accuracy by printing the actual labels.

| Class Label | Actual Value | Recognition rate |
|---|---|---|
| 0 | "ග" | 82% |
| 1 | "ඟ" | 80% |
| 2 | "ඣ" | 93% |
| 3 | "ල" | 80% |
| 4 | "ම" | 81% |
| 5 | "ඵ" | 88% |
| 6 | "ඹ" | 87% |
| 7 | "ධ" | 82% |
| 8 | "ය" | 82% |

Figure 4.6: Classifier prediction results.

## 4.4 Evaluation

### 4.4.1 Introduction

Usually in OCR systems, the evaluation criterion is based on the overall recognition percentage and the effect of computational load on the system performance. According to that, the comparison between ANN (Artificial Neural Network) and CNN (Convolutional Neural Network) approach are presented in this section. Although template matching technique was tried, we recognized it as a weak solution due to the lack of recognition rate and it is not suitable to recognize this type of complex characters. In template matching, objects can be represented by storing as a sample images or "templates". Subsequently compare the pixel values in the template with the pixel values in the underlying region in the image. If a "good" match is found, OCR would recognize that the object is present in the image. According to our experiments, most of test data is not recognized due to the small variations in character that differ from the template. Therefore this is not suitable to recognize ancient Sinhala inscription.

For both solutions CNN and ANN, a different number of ancient inscription character sets are used for training and testing and are processed manually. The recognition percentage is calculated for multiple characters.

### 4.4.2 OCR Evaluation workflow

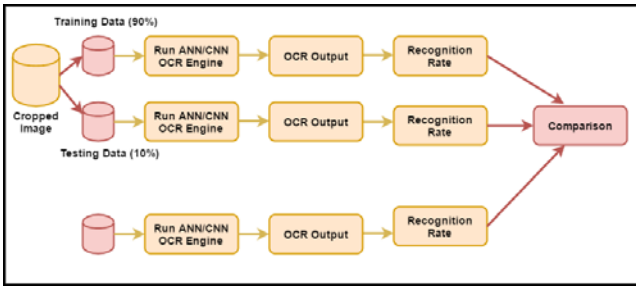


Figure 4.7 : Overall OCR Evaluation workflow

Figure 4.7 shows the evaluation workflow based on the dataset. Cropped dataset was divided into two set including training set and test data. All these cropped set was preprocessed and the recognition rate of these data set is separately estimated. Subsequently the recognition rate for real images with noise is obtained. Finally the recognition rates of all these three datasets are compared and determined which OCR engine was acceptable. An individual character wise evaluation was done by us rather than using the whole script for evaluation. As mentioned in the diagram the score was obtained using the equation which is based on individual character recognition rate calculation. The post evaluation was done manually.

Here we prepare separate data set for each character and split it into two sets as 90% for the training set, 10% for the test set. Then each data set was preprocessed through particular OCR engine. The recognition rate was calculated using the formula given in Figure 4.8.

$$\text{Score} = \frac{TNC - TNE}{TNC}$$

Figure 4.8 : Post evaluation recognition rate calculation

TNC - Total Number of Characters in dataset

TNE - Total Number of Errors (How many character inaccurately predict)

Using this equation, we were able to identify how well our ancient character recognition OCR currently works.

### 4.4.3 Experimental Results

#### 4.4.3.1 ANN based OCR

In the case of ANN based OCR solution used training dataset which has 850 training images. For each iteration, 10 percent of data (85 images) is used for validation and remaining 90 percent (765 images) for training.

| Char | Samples for | Samples for testing | Sample for | Acc (%) |
|---|---|---|---|---|
|  | 120 | 12 | 12 | 82% |
|  | 100 | 10 | 10 | 72% |
|  | 70 | 7 | 7 | 55% |
|  | 80 | 8 | 8 | 60% |
|  | 110 | 11 | 11 | 62% |
|  | 75 | 7 | 7 | 65% |
|  | 70 | 9 | 9 | 40% |
|  | 70 | 7 | 7 | 71% |
|  | 70 | 7 | 7 | 42% |

Figure 4.9 : ANN based OCR prediction results (Individual Character Wise)

### 4.4.3.2 CNN based OCR

| Char | Samples for Training | Samples for testing (Cleaned Images) | Sample for testing (Noisy images) | Acc (%) |
|---|---|---|---|---|
|  | 120 | 12 | 12 | 93% |
|  | 100 | 10 | 10 | 88% |
|  | 70 | 7 | 7 | 80% |
|  | 80 | 8 | 8 | 88% |
|  | 110 | 11 | 11 | 82% |
|  | 75 | 7 | 7 | 80% |
|  | 70 | 9 | 9 | 82% |
|  | 70 | 7 | 7 | 81% |
|  | 70 | 7 | 7 | 82% |

Figure 4.10 : CNN based OCR prediction results (Individual Character wise)

According to the above statistics, the recognition of individual character contains the range from 72% to 95%. The result of the character recognition highly depends on the input image quality. Both CNN and ANN OCR engine have performed satisfactorily for the noisy test data. High noise rate of the input images has decreased the recognition rate.

Figure 4.11 has shown the overall accuracy of both OCR engines on Training data, test data (preprocessed) and test data (Real images). Both OCR have shown less recognition rate for test data set of real images with noise.

| | Training Data set | | Test data (Preprocessed) | | Test Data (Real images) | |
|---|---|---|---|---|---|---|
| | Number of Samples | Acc (%) | No of Samples | Acc (%) | No of Samples | Acc %) |
| ANN | 765 | 85% | 85 | 80% | 85 | 45 |
| CNN | 765 | 95.2 | 85 | 92% | 85 | 65 |

Figure 4.11: Summary of Comparison

## 5.0 Conclusion

According to the above clarification and results CNN OCR engine has shown the better accuracy than ANN OCR engine. CNN OCR engine require a high time period to complete the whole training process even though it used the same data set. Our version of CNN (that got 5% error rate) can be implemented in about 2-4 hours, depending on how much one is familiar with it. Therefore the training time of OCR engine will be quite flexible. CNN based OCR engine could be further enhanced to the ancient Sinhala inscription character recognition system developed with more data sets.

## *Reference*

[1] Pieris.K [1918],"Inscriptions of ancient and medieval Sri Lanka", Sri Lanka's National Newspaper.

[2] Wikipedia,"Stone Inscriptions in Sri Lanka", March,[2016].

[3] Root Web [2015] Alternative Gravestone Reading Methods. http://www.rootsweb.ancestry.com

[4] P.Niranjan [2015] Literature Review of Segmentation Problems in Nepali Optical Character Recognition , Project Report, Masters of Technology in Information Technology, Department of Computer Science and Engineering, Kathmandu University.

[5] P.Nikhi,V.Jayakumar,S.kolkure [2015] OPTICAL CHARACTER RECOGNITION: AN ENCOMPASSING REVIEW [ONLINE] Available at:

http://esatjournals.net/ijret/2015v04/i01/IJRET20150401062.pdf

[6] Yasser Alginahi [2015] Preprocessing Techniques in Character Recognition [ONLINE] Available at: http://cdn.intechopen.com/pdfs/11405.pdf

[7] Hubert Mara, Jan Hering and Susanne Kromker [2014] GPU Based Optical Character Transcription for Ancient Inscription Recognition [ONLINE] Available at: https://www.researchgate.net/publication/224610728_GPU_Based_Optical_Character_Transcription_for_Ancient_Inscription_Recognition

[8] S.Rajakumar, V.Subbiah Bharathi [2011] Century Identification and Recognition of Ancient Tamil Character Recognition [ONLINE] Available at: http://www.ijcaonline.org/volume26/number4/pxc3874237.pdf

[9] Abhishek Tomar , Minu Choudhary , Amit Yerpude [2015] Ancient Indian Scripts Image Pre-Processing and Dimensionality Reduction for Feature Extraction and Classification: A Survey [ONLINE] Available at: http://www.ijcttjournal.org/2015/Volume21/number-2/IJCTT-V21P116.pdf

[10] H. K. Anasuya Devi [2006] Thinning: A Preprocessing Technique for an OCR System for the Brahmi Script [ONLINE] Available at: http://www.ancient-asia-journal.com/articles/10.5334/aa.06114/

[11] Sachin S Bhat, H.V. Balachandra Achar [2015] Character recognition and Period prediction of ancient Kannada Epigraphical scripts [ONLINE] Available at: http://www.ijarcce.com/upload/2016/si/nCORETech-16/nCORETech%2024.pdf

[12] Ayatullah Faruk Mollah , Nabamita Majumder , Subhadip Basu and Mita Nasipuri [2011] Design of an Optical Character Recognition System for Camerabased Handheld Devices [ONLINE] Available at: https://arxiv.org/ftp/arxiv/papers/1109/1109.3317.pdf

[13] Indu Sreedevi, Rishi Pandey, N.Jayanthi, Geetanjali Bhola [2013] NGFICA Based Digitization of Historic Inscription Images [ONLINE] Available at: https://www.researchgate.net/publication/258397837_NGFICA_Based_Digitization_of_Historic_Inscription_Images

[14] G.Vamvakas, B.Gatos, N. Stamatopoulos, and S.J.Perantonis [2008] A Complete Optical Character Recognition Methodology for Historical Documents ONLINE] Available at: http://users.iit.demokritos.gr/~bgat/3337a525.pdf

[15] Luke V Rasmussen, Peggy L Peissig, Catherine A McCarty, Justin Starren [2012] Development of an optical character recognition pipeline for handwritten form fields from an electronic health record [ONLINE] Available at :http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3392858/

[16] Google's Optical Character Recognition (OCR) software [2015] https://opensource.com/life/15/9/open-source-extract-text-images

[17] Optical Character Recognition [2016] https://en.wikipedia.org/wiki/Optical_character_recognition

[18] OpenCv , OpenCv python tutorials , [2014] [online].Available at : http://docs.opencv.org/3.0-beta/doc/py_tutorials/py_tutorials.html

[19] Grzegorzgwardys , Convolutional Neural Networks backpropagation: from intuition to derivation, April 22, [2016],[online].Available at : https://grzegorzgwardys.wordpress.com/2016/04/22/8/

[20] Amruta B. Patil, J.A.shaikh, OTSU Thresholding Method for Flower Image Segmentation , [May 2016] [online].Available at : http://www.ijceronline.com/papers/Vol6_issue5/A0650106.pdf

[21] Hetal, J. Vala, Prof. Astha Baxi , A Review on Otsu Image Segmentation Algorithm, Vol 2 , February [2013] [online].Available at : http://ijarcet.org/wp-content/uploads/IJARCET-VOL-2-ISSUE-2-387-389.pdf

**K.G.N.D. Karunarathne** is a final year undergraduate student of the four year special degree programme in Bachelor of Science in Information Systems. She has submitted her final year group project titled "Recognition of Ancient Sinhala Inscription Characters from 10 A.D to 12 A.D".

**K.V. Liyanage** is a final year undergraduate student of the four year special degree programme in Bachelor of Science in Information Systems. She has submitted her final year group project titled "Recognition of Ancient Sinhala Inscription Characters from 10 A.D to 12 A.D".

**D.A.S. Ruwanmini** is a final year undergraduate student of the four year special degree programme in Bachelor of Science in Information Systems. She has submitted her final year group project titled "Recognition of Ancient Sinhala Inscription Characters from 10 A.D to 12 A.D".

**G.K.A. Dias** received his BSc in Physical Science (1982) from the University of Colombo, Sri Lanka, Postgraduate Diploma in Computer Studies from University of Essex,UK (1986), and MPhil in Computer Science from University of Wales,UK (1995). He is currently a Senior Lecturer at the University of Colombo School of Computing (UCSC). He is a member of the Computer Society of Sri Lanka, Sri Lanka Association for Advancement of Science (SLASS) and a member of the ACM the world's largest educational and scientific computing society. He received the winner award for his 'Travel Meter' project in eSwarbhimani 2016, Sri Lanka under Culture & Tourism Category. He also won the Bronze award under Education & Training category in NBQSA Sri Lanka-National Best Qualaity Software Award -Oct 2015 for K8-Flight Simulation project. He is also a member of the "Vidusayura" Project group, which won several International and National Awards listed below.

- e-Swabhimani - National Best e-Content Award - Sri Lanka 2009
- Gold Award (Education & Training Category) – National Best Quality Software Award (NBQSA)- 2010
- Overall Bronze Award–National Best Quality Software Award (NBQSA)- 2010

He has more than 50 peer reviewed publications to his credit. His research interests are computer aided software engineering, multimedia for education, modeling and simulation, artificial neural networks and model driven engineering.

**S.T. Nandasara** is a long serving prominent academic since 1981, in the Science Faculty f University of Colombo, Sri Lanka. He developed the first ever national web site for Sri Lanka in 1996 while being a researcher on the WIDE Project at University of Keio, Japan. The site was officially launched in July, 1996 as the national web site of Sri Lanka (www.lk). Later he joined hands with the Internet Research and Development Unit of NUS, Singapore to transform www.lk to be a multilingual national portal available in Sinhala, Tamil and English. Later, he served as the project coordinator for the Kothmale Community Internet Radio Project, from 1998 to1999. The project was commissioned to assess the potential benefits of new communication technologies to remote areas by providing Internet facilities through the www.kirana.lk in all three languages. Nandasara was a Member of the Forum for Multilingual Information Processing (MLIT) and Asian Forum for Standardization of Information processing (AFSIT). Both were established by Center of the International Cooperation for Computerization (CICC) form 1997 to 2001. He also served as a committee member for the project "Cultural Diversity and Information Network - CDIN", National Graduate Institute for Policy Studies (GRIPS), Tokyo, form 2002 to 2004 to construct a network of science, culture and information through international comparative studies. He was a co-leader of the Asian Language Resource Network Project from 2005 to 2008 of Nagaoka University of Technology, Japan and is a member of IFIP's Working Group of the History of Computer Education. Nandasara has been a member of many Working Committees, recommending standards on use of Sinhala and Tamil in information technology development and implementation. He was a member of the Sectoral Committee on Information Technology of the Sri Lanka Standard Institute (1993-2008). Nandasara was the National Coordinator and a Member of the ISO WG2 (1997-1998), by closely working with ISO and the Unicode Consortium, he was instrumental in standardization of Sinhala scripts.

He has more than 80 peer reviewed publications in journals, books, conferences and technical reports in the field of R&D on national language development, localization and standardization. Nandasara is also an alumni of University of Colombo and holds a degree in development studies and statistics.