

# Conversion of Early Tamizh Brahmi Characters into Modern Tamil Characters Using Template Matching Algorithm

S. Mageshwaran

Department of Computer Science and Engineering,  
Anna University, India

C. Ravishankar

Department of Computer Science and Engineering,  
Anna University, India

G. Alagumalaikannan

Department of Computer Science and Engineering,  
Anna University, India

M. Kavin

Department of Computer Science and Engineering,  
Anna University, India

S. S. L. DuraiArumugam

Department of Computer Science and Engineering,  
Anna University, India

**Abstract:-** Optical Character Recognition by using Template Matching is a system prototype that useful to recognize the character or alphabet by comparing two images of the alphabet. The objectives of this system prototype are to develop a prototype for the Optical Character Recognition (OCR) system and to implement the Template Matching algorithm in developing the system prototype. This system prototype has its own scopes which are using Template Matching as the algorithm that applied to recognize the characters, characters to be tested are Early Tamil Brahmi letters, grey-scale images were used, using .PNG image format with 53 x 53 image size and recognizing the characters by comparing between two images. The purpose of this system prototype is to solve the problem in recognizing the character which is before that it is difficult to recognize the character without using any techniques and Template Matching is as one of the solution to overcome the problem. *Python3.5* is the software tool that was used in developing the system prototype. There are a few processes that were involved in this system prototype. The processes are starting from the acquisition process, filtering process, threshold the image, clustering the image of alphabet and lastly recognize the alphabet. All of these processes are very important to get the result of recognition after comparing the two character images.

**Keywords:** *Image Processing, Corner Detection, Template matching, Python, OCR*

## 1. INTRODUCTION

Optical Character Recognition is the process whereby typed or printed pages can be scanned into computer systems, and their contents recognized and converted into machine-readable code [1]. Template matching is one of the Optical Character Recognition techniques. Template matching is the process of finding the location of a sub image called a template inside an image. Once a number of corresponding templates is found their centres are used as corresponding points to determine the registration parameters. Template matching involves determining similarities between a given template and windows of the same size in an image and

identifying the window that produces the highest similarity measure. It works by comparing derived image features of the image and the template for each possible displacement of the template. The Optical Character Recognition (OCR) applications are very important in many fields. Some of the fields which the OCR applications were used are such as in business, banking, government, travel industry and hotel industry. In the business, the OCR applications are used for data entry automation for ordering entry and file folder tracking of names and numbers. Besides, in government, the OCR applications were used for utility billing such as tax, water, fee, voting cards and license bills. Airline tickets and passports are also the OCR applications in Travel Industry field [2].

## 2. PROBLEM STATEMENT

The earliest OCR that was only converts or reads only English, modern language characters from an image .There is no system to read ancient Early Tamil Brahmi characters from an image. So to overcome this problem, Template Matching is one of the solutions that were suitable to implement in recognizing the optical character because of the simple algorithm that was used.

## 3. PROPOSED SYSTEM

The main drawback of the existing system is that it translates only modern languages <sup>[1][2]</sup>. The main idea of our proposed system is that applying image processing techniques <sup>[5]</sup> for this translation process and create the basic methodology for translating the Early Tamizh Brahmi characters.

- To convert the inscription Tamizh letters into modern form.
- To reduce human power.
- To reduce conversion process of inscription fonts to Modern fonts.

#### 4. METHODOLOGY

The usage of image processing methods has following modules,

- Data Collection and Training dataset
- Input Image Pre-processing
- Image segmentation
- Implementation of Template Matching
- Implementation of CNN

##### 4.1. Data collection and Training dataset

Tamizh scripts have various transformation in its Character representation. In this data collection process involves only characters from early Tamizh Brahmi Characters

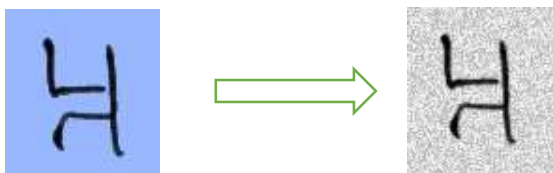
##### 4.2. Input Image Pre-processing

The importance of the pre-processing stage of a character recognition system lies in its ability to remedy some of the problems that may occur due to some of the factors. Thus, the use of pre-Processing techniques may enhance a document image preparing it for the next stage in a character recognition system.

- Binarization
- Noise Removal

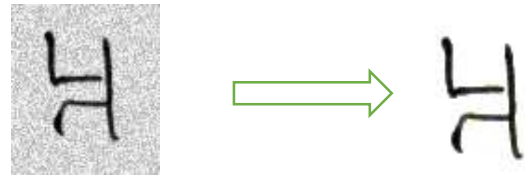
##### 4.2.1. Binarization

The Binarization Method converts the grey scale image (0 up to 256 gray levels) in to black and white image (0 or 1). The result of character recognition highly depends upon the binarization. The high quality binarized image can give more accuracy in character recognition as compared original image because noise is present in the original image. (Figure represent the first alphabet in Tamil language ஹ in the form of Early Tamil Brahmi character)



##### 4.2.2. Noise Removal

Noise is random variation of brightness or colour in an image that can make the text of the image more difficult to read. Certain types of noise cannot be removed in the binarization step, which can cause accuracy rates to drop.



##### 4.3. Image segmentation

Image segmentation is the process of partitioning a digital image into multiple segments (set of pixels, also known as super-pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyse. Here the segmentation phase is done by Thresholding method. This is very basic segmentation algorithm used for portioning the character from an image. (Figure shows the portioned Early Tamil Brahmi character in an image modern form of that character is ஹ )



##### 4.4 Implementation of Template Matching

- Firstly, the character image from the detected string is selected.
- After that, the image to the size of the first template is rescaled.
- After rescale the image to the size of the first template (original) image, the matching metric is computed.
- Then the highest match found is stored. If the image is not match repeat again the third step.
- The index of the best match is stored as the recognized character.

##### 4.4.1 Template Matching Approach

Template matching models were developed as an answer to the problem of object recognition, and they incorporate at least implicitly the idea of similarity comparison. The representations assumed by template models carry much more detailed information about stimulus structure than do the element representations just described. These models are usually applied to spatially extend visual objects, and their representation can be thought of as being spatially organized. The central concept behind the template matching algorithm is reference points. Reference points are points at the center of spatial regions in 3-D space. For this particular system, the regions were defined as an x, y, z center point, and three distance values, one for each axis. By alternately adding and subtracting the distance values along

the appropriate axis about the center point, a region in the shape of a cube is formed. A possible alternative would be to define a point and a radius, and subsequently describe a sphere as the region. The origin for the coordinate system is defined to be the center of the subject's right shoulder socket. Since the data from the wrist tracking sensors is normalized to this origin, it is relatively easy to determine which reference point region the sensor is in at a given moment. Reference point regions are defined to correspond to the locations of the sensors when gestures are executed. Also, by negating the right/left axis value for the center point, a symmetric set of regions is defined for the left hand sensor.

4.4.2 Filtering Image

Filtering is one of the processes in image processing before further steps are taken. Filtering is a process by which unwanted substances are removed from a mixture of elements, thus leaving useful material behind. Filtering also a technique for modifying or enhancing the image. There are many types of filtering the image. Some of the filtering types are such as the minimum filtering, maximum filtering, median filtering, average filtering and others. All of them have their own algorithm. Some of the algorithm that was applied in the technique is such as below:

Average filtering technique algorithm

$$S = \frac{1}{N^2} \sum_{(r,c) \in W} d(r,c) \tag{5}$$

Overall, the filtering technique that used by using average filtering algorithm is counting the average values of the image from each window to another window. The windows are called  $N \times N$ . This calculation is using the changes concept of the windows to process the image. So that, the filtering of the image is using the algorithm as above to filter the image.  $N_2$  means the total of pixels in each window which called as  $W$ . The maximum and minimum filters are two order filters that can be used in filtering the image. The maximum filter selects the largest value within an ordered window of pixel values, whereas the minimum filter selects the smallest value.

4.4.3 Threshold Image

The second process after filtering the image is the threshold technique. There a lot of techniques for threshold the image such as the minimum threshold, maximum threshold, median threshold and the average threshold.

Threshold technique is the technique that used to transform a grayscale image into a binary image which is using (1 or 0) values. A

threshold is set which each pixel is compared to other pixels. If the pixel is greater than or equal to this threshold, it

is outputted as a 1. Otherwise it is outputted as a 0. Threshold converts each pixel into black, white or unchanged depending on whether the original color value is within the threshold range.

$$\text{A pixel becomes } \begin{cases} \text{white if its gray level is } < T \end{cases} \tag{6}$$

$$\begin{cases} \text{black if its gray level is } \geq T \end{cases} \tag{7}$$

The algorithm that shown as below is the algorithm that was used to threshold the image. This algorithm is called average threshold technique. The average threshold technique is based on the average value of the image. Every point is the pixels value of the image. The value of each point is added and it divided by the number of points that is counted for each image. This is important to get the threshold value of the image which is 0 and 1.

$$S = \frac{1}{N^2} \sum_{(r,c) \in W} d(r,c) \tag{8}$$

Other techniques for threshold the image are Maximum threshold, Minimum threshold and Median threshold. All of them also have their own algorithm. Maximum threshold technique is based on the maximum value of the image, while the minimum threshold technique is based on the minimum value among all of the pixels in the image.

Maximum and Minimum threshold technique algorithm:

$$y_i = \begin{cases} x_i & \text{if } x \geq y_{i-1} \\ y_{i-1} & \text{if } x_i < y_{i-1} \text{ and} \\ x_{i-n} = y_{i-1} \end{cases} \tag{9}$$

Median technique is the threshold technique that counts the median value of the pixel in the image.

$$x_1 < x_2 < \dots < x_n \tag{12}$$

$x$  in the algorithm above is refers to the value of the image and it will be count until getting the median value among all the pixels in the image.

6. RESULT

This system works flawlessly without hassle. The Main advantages of this system is to create a basic OCR requirements and extract the ancient Tamizh characters from image by using image processing.

Hence it works only for early Tamizh Brahmi letters from AD2-AD4. Further research is continue on this system to scale it up further and to add more features that enables to the language analysts/translators to carry out further more information about ourselves

## REFERENCES

- [1] R. Smith "An Overview of the Tesseract OCR Engine" Google Inc. <http://ieeexplore.ieee.org/document/4376991>
- [2] Amitabh Wahi Dept of Information Technology, Bannari Amman Institute of Technology, Sathyamanagalam, Erode, Tamilnadu, India "Handwritten Tamil character recognition" <http://ieeexplore.ieee.org/abstract/document/6921982>
- [3] C.A. Perez, A. Palma, C.A. Holzmann and C. Pena Dept. of Electr. Eng., Chile Univ., Santiago, Chile "Face and eye tracking algorithm based on digital image processing" <http://ieeexplore.ieee.org/document/973079/>
- [4] J. Gllavata, R. Ewerth and B. Freisleben SFB/FK, Siegen Univ., Germany "A robust algorithm for text detection in images" <http://ieeexplore.ieee.org/document/1296349/>
- [5] Frazer K. Noble Centre for Additive Manufacturing School of Engineering and Advanced Technology, Massey University, New Zealand "Comparison of OpenCV's feature detectors and feature matchers" <http://ieeexplore.ieee.org/document/7827292>
- [6] T. F. Cootes, C. J. Taylor, D. Cooper, J. Graham, "Active shape models - their training and application", *Computer Vision and Image Understanding*, vol. 61, pp. 38-59, Jan. 1995.
- [7] T. Cootes, G. Edwards, C. Taylor, "Active appearance models", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, pp. 681-685, Jun. 2001.
- [8] S. Liao, A. K. Jain, "Partial face recognition: an alignment free approach", *Proc. IAPR/IEEE International Joint Conference on Biometrics (IJCB 11)*, pp. 11-13, Oct. 2011.
- [9] S. Liao, A. K. Jain, S. Z. Li, "Partial face recognition: an alignment free approach", *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 26, pp. 1193-1205, May. 2013.
- [10] L. Di Stefano, S. Mattoccia, F. Tombari, "Comparison of Various Template Matching Techniques for Face Recognition", *International Journal of Engineering Research and Development*, vol. 8, pp. 16-18, Oct. 2013.
- [11] L. Di Stefano, S. Mattoccia, F. Tombari, "ZNCC-based template matching using bounded partial correlation", *Pattern Recognition Letters*, vol. 26, pp. 2129-2134, May. 2005.
- [12] N. N. Dawoud, B. B. Samir, J. Janier, "Fast Template Matching Method Based on Optimized Metrics for Face Localization", *Proc. International MultiConference of Engineers and Computer Scientists (IMESC 12)*, pp. 700-704, Mar. 2012.
- [13] C.H. Chan, M.A. Tahir, J. Kittler, M. Pietikainen, "Multiscale local phase quantization for robust component-based face recognition using kernel fusion of multiple descriptors", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 1164-1177, May. 2013.
- [14] S. Nikan, M. Ahmadi, "Performance Evaluation of Different Feature Extractors and Classifiers for Recognition of Human Faces with Low Resolution Image", *Proc. Int. Conf. on Advanced Technology & Sciences (ICAT 14)*, pp. 13-18, Aug. 2014.
- [15] H. Hu, "Variable lighting face recognition using discrete wavelet transform", *Pattern Recognition Letters*, vol. 32, pp. 1526-1534, Oct. 2011.
- [16] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, Y. Ma, "Robust face recognition via sparse representation", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, pp. 210-227, Feb. 2009.
- [17] A. Martinez, R. Benavente, "The AR face database", *CVC Technical Report*, vol. 24, 1998.
- [18] G. B. Huang, M. Ramesh, T. Berg, E. Learned-Miller, *Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments*, Technical Report, University of Massachusetts, 2007.
- [19] P. J. Phillips, H. Moon, S. A. Rizvi, P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, pp. 1090-1104, 2000.
- [20] D. H. Ballard, C. M. Brown, Matching. In: —Computer vision, New Jersey: Prentice Hall, pp. 352-382, 1982.