Volume 5, Issue 4, 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

# Recognition and Conversion of Handwritten Modi Characters

**Sonali R. Mayne, Monika P. Dhobale, Rohini H Jadhav, Rushikesh A Ambildhok**
VIIT College, University of Pune,
Maharashtra, India

***Abstract***— ***The study on Handwritten automatic character recognition has attracted many researchers all over the world to contribute automatic character recognition domain. The research had been performed on many foreign languages like Japanese, Chinese, French, Italian, Korean , Arabic etc. The work on Indian ancient script is still not done that much. The survey conducted by "Bhasha Research and Bhasha Publication Centre", India speaks 780 languages out of which 220 language have lost in 50 years and another 150 could vanish in next half century. Modi is an ancient script had cursive and complex representation of character as compare to other Indian ancient languages. In this paper we describe a system for recognition of handwritten Modi script characters; the proposed method uses Image processing algorithms that were mentioned below.***

***Keywords— MODI Script, Handwritten character recognition (HCR), Image processing (IP), Optical character recognition(OCR),***

## I. INTRODUCTION

India is popularly known for its rich cultural heritage. India is a country where we find large diversity in culture, religions and languages. Modi language is very old language which can trace as long as 6th century. Very important ant vital documents were written in the Modi language .We can not afford to lose such precious knowledge about the various economic strategies and the ancient writing which can even help us in today's life. There are very few experts who have knowledge of the Modi script and hence it would take a long time to convert such documents into human-readable languages. Hence we have to convert these documents into English to preserve them forever and to extract important information after decoding them. Section II describes Modi script. In section III Preprocessing Algorithms discussed. Future work discussed in section IV ,conclusion explained in section V .

## II. MODI

Modi language used in the yester years as an official language for storing records, maintaining accounts, daily happening etc. Many such documents have been stored in museums, temples and at some few sacred places. Bulks of the documents of these kinds have stored in Maharashtra. With time, such important writings will soon decay and we will lose them. Modi script used in education, journalism and other routine activities before 1950's.All of these Modi documents provided authentic historical information and published original material to study political, social and economic history of Maharashtra." Modi is an ancient script of India and our aim is that it should be known to everyone, so we are going to do recognition and conversion of Modi character".

## III. PREPROCESSING ALGORITHMS

### *A. Gray-scale Algorithm for Image Processing*

Three algorithms for converting color image to grayscale image .How do you convert a color image to grayscale image? If each color pixel is described by a triple (R, G, B) of intensities for red, green, and blue.How do you map that to a single number giving a grayscale value? The imaging software has three algorithms.

The **lightness** method calculates the most prominent and least prominent colors: (max(R, G, B) + min(R, G, B))/ 2.
The **average** method simply calculates the average values: (R + G + B) / 3.
The **luminosity** method is a more sophisticated version of the average method. It also calculates the average values, but it forms a weighted average to account for human perception. We are more sensitive about green than other colors, so green is weighted most heavily. The luminosity formula is 0.21 R + 0.72 G + 0.07 B.

Fig.1 Original image

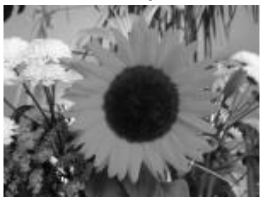


Fig.2 Lightness

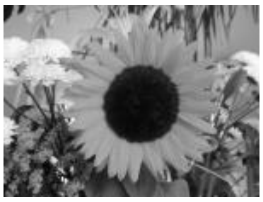


Fig.3 Average

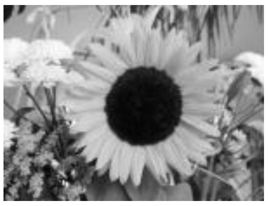


Fig.4 Luminosity

Out of this three methods, we used average method in our proposed system.

*1) How Grayscale algorithms fundamentally work:*

All grayscale algorithms uses the same basic three-step process:

1. Get the red, green, and blue values of a pixel
2. Use fancy math to turn those numbers into a single gray value
3. Replace the original red, green, and blue values with the new gray value

When describing grayscale algorithms, we are going to focus on step 2 – using math to turn color values into a grayscale value. The formula like this:

Gray = (Red + Green + Blue) / 3

*2) RGB To Grayscale Conversion*

• Algorithm

– Traverse through entire input image array.
– Read individual pixel color value (24-bit).
– Split the color value into individual R, G and B 8-bit values.
– Calculate the grayscale component (8-bit) R, G and B pixels using a conversion formula.
– Compose a 24-bit pixel value from 8-bit grayscale value.
– Store the new value at same location in output image.

*3) Traverse Through Entire Image*

24-bit Color Value

```
b = pix & 0xff;
g = (pix >> 8) & 0xff;
r = (pix >> 16) & 0xff;
```

Extract 8-bit R, G and B values from

```
for(y=0;y<height;y++) {
for(y=0;y<height;y++) {
for(x=0;x<width;x++) {
pix = input[y][x];
```

***B. Otsu's method for image thresholding***

Otsu's thresholding method includes iterating through all the possible threshold values and calculating a measure pixel levels each side of the threshold, i.e. the pixels that either fall in foreground or background. The aim is to find the threshold value where the sum of foreground and background spreads is at its minimum.The algorithm will be demonstrated using the simple 6x6 image shown below

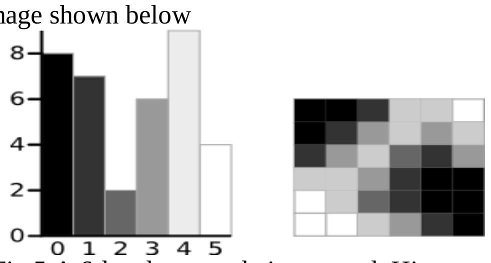


Fig.5 A 6-level grayscale image and Histogram

During the thresholding process, individual pixels in an image are marked as "object" pixels if their value is greater than some threshold value (assuming an object to be brighter than the background) and as "background" pixels otherwise. This convention is known as threshold above. Variants include threshold below, which is opposite of threshold above; threshold inside, where a pixel is labeled "object" if its value is between two thresholds; and threshold outside, which is the opposite of threshold inside .Typically, an object pixel is given a value of "1" while a background pixel is given a value of "0."A binary image is created by coloring each pixel white or black, depending on a pixel's labels.

1) Thresholding

• Steps / Algorithm

– Traverse through entire input image array.
– Read individual pixel color value (24-bit) and convert it into grayscale.
– Calculate the binary output pixel value (black or white) based on current threshold.
– Store the new value at same location in output image.

2) Thresholding Logic

```
gs = (r+g+b) / 3; // grayscale
if(gs < th) {
pix = 0; // pure black
}else {
pix = 0xFFFFFF; // pure white
}
```

### *C. Boundary Detection*

A lot of the attention is focused to edge detection, being a crucial part in most of the algorithms. Classically, the first stage of edge detection (e.g. the gradient operator, Robert operator, the Sobel operator, the Prewitt operator) is the evaluation of derivatives of the image intensity. Smoothing filter and surface fitting are used as regularization techniques to make differentiation more immune to noise.

*1) Sobel edge detection for image processing:*

What is edge detection ?

• A Spatial Edge Detection filter that detects edges by finding the gradient of an image. This means we will be finding a dramatic difference between the value of pixels.

*2) Why do we need edge detection?*

Edge detection is required for no of application such as:-

• Shape detection.

• Face recognition.

• Huge application in bio-medical.

Fig.6 The standard Leena image

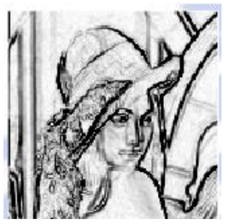
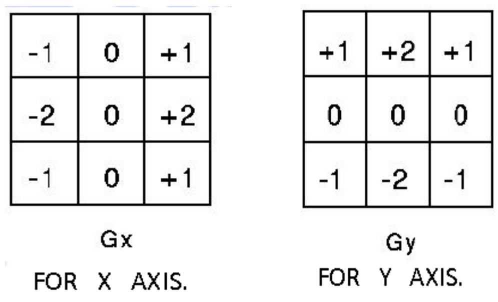


Fig.7 output of edge detection

we have two template matrices for applying Sobel algorithm



| | | |
|---|---|---|
| -1 | 0 | +1 |
| -2 | 0 | +2 |
| -1 | 0 | +1 |

Gx

FOR X AXIS.

| | | |
|---|---|---|
| +1 | +2 | +1 |
| 0 | 0 | 0 |
| -1 | -2 | -1 |

Gy

FOR Y AXIS.

*3) Algorithm steps:*

• Steps / Algorithm

• The input images are converted to gray
scaled image.

– Traverse through entire image.

– For each pixel in the image, we will take a window
of 3*3 pixel and multiply it the given template for matrix.

– Then we will calculate the G using formula

$$\mathbf{G} = \sqrt{\mathbf{G}_x{}^2 + \mathbf{G}_y{}^2}$$

Template:-

| | |
|---|---|
| -1 0 1 | 1 2 1 |
| -2 0 2 | 0 0 0 |
| -1 0 1 | -1 -2 -1 |
| X | Y |

Apply the templates to a 3x3 filter window.

a1 a2 a3
a4 a5 a6 3x3 filter window
a7 a8 a9

where a1 …. a9 are grey levels of each pixel in the filter window.

X = -1*a1 + 1*a3 - 2*a4 + 2*a6 - 1*a7 + 1*a9

Y = 1*a1 + 2*a2 + 1*a3 - 1*a7 - 2*a8 - 1*a9

Sobel Gradient = sqrt(X*X + Y*Y)

### *D. Cropping:*

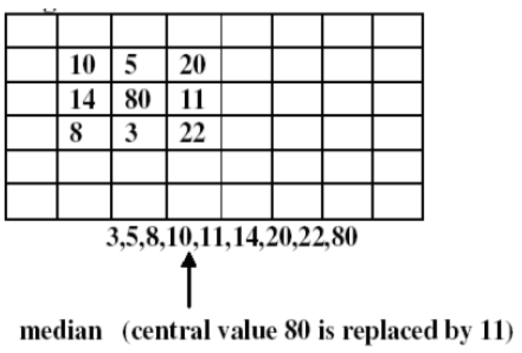
In printing, graphic design and photography industries, cropping refers to removing unwanted areas from a photographic or illustrated image. In the most basic photo manipulation processes, it is performed in order to remove an unwanted subject or irrelevant detail from a photo, change its aspect ratio, or to improve the overall composition. In telephoto photography, most commonly in bird photography, an image is cropped to magnify the primary subject and further reduce the angle of view when a lens of sufficient focal length to achieve the desired magnification directly is not available. It is considered one of the few editing actions permissible in modern photojournalism along with tonal balance, colour correction and sharpening. A crop made from the top and bottom of a photograph may produce an aspect which mimics the panoramic format (in photography) and the widescreen format in cinematography and broadcasting.

### *E. Noise Removal:*

Noise is a random variation of image Intensity and visible as grains in the image. It may arise in the image as effects of basic Physics-like photon nature of light or thermal energy of heat inside the image sensors. It may produce at the time of capturing or image transmission. Noise means, the pixels in the image show different intensity values instead of true pixel values. Noise removal algorithm is the process of removing or reducing the noise from the image. The noise removal algorithms reduce or remove the visibility of noise by smoothing the entire image leaving areas near contrast boundaries. These methods can obscure fine, low contrast details.

#### *1) Median Filter for Noise Removal:*

The median filter is a nonlinear digital filtering technique, used to remove noise. Median filtering is very often used in digital image processing because, under certain conditions, it preserves edges while removing noise. Mainly the median filter is to run through the signal entry by entry, replacing each entry with the median of neighboring entries. The pattern of neighbors is known as "window," which slides, entry by entry, over the entire signal. When the window has an odd number of entries, then the median is simple to define: it is just the middle value; after all the entries in the window are sorted numerically. For an even number of entries, there is more than one possible Median. The median filter gives the best result when the impulse noise percentage is less than 0.1 %. When the quantity of impulse noise is growing then median filter not gives the best result.



#### *2) Algorithm of Median Filter:*

The Median Filter Algorithm is as follows:

Step1. Select a two-dimensional window W of size 3*3. Assume that the pixels being processed is Cx,y.

Step2. Compute Wmed the median of the pixel values in window W.

Step3. Replace Cx,y by Wmed.

Step4. Repeat the steps 1 to 3 until all the pixels in the entire image processed.

#### *3) Advantage:*

a. It is easy to implement.

b. De-noising different types of noises.

### *F. Stentiford thinning algorithm.*

In 1983, Stentiford introduced a new approach for skeletonization algorithms using a mask concept. Four masks are used to scroll the image in an orderly manner in the form: M1, M2, M3, M4. The four different masks that are used in the algorithm Stentiford are shown in Fig8. The white circle represents a white pixel with a value of 255 in Fig1. The black circle represents a black pixel with a value of zero, and ”X” represents a pixel is either black or white. These masks cross the image in the following order:

M1 - from left to right and top to bottom;

M2 – from a bottom to top and from left to right;

M3 - from right to left and from a bottom to top;

M4 - from top to bottom and from right to left.

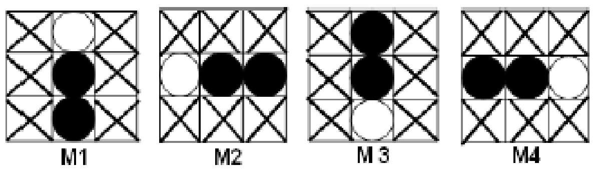


Fig.8 The four different mask used in Stentiford Thinning Algorithm

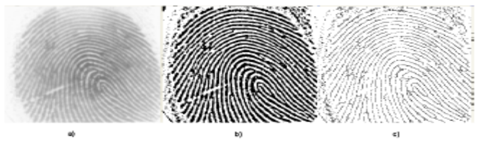


Fig.9 Example of the result of stentiford thinning algorithm applied to a fingerprint image a)fingerprint image b)binarized image c)thin image

### *G. Scaling*

In computer graphics, **image scaling** is the process of resizing a digital image. The Scaling is a non-trivial process that involves a trade-off between efficiency, smoothness and sharpness. With the help of bitmap graphics size of an image is reduced or enlarged. The pixels form the image become increasingly visible, making the image appear "soft" if pixels are averaged, or jagged if not.

*1) Bilinear Interpolation Method for Scaling:*

A Texture Mapping technique produces a reasonably realistic image, also known as "bilinear filtering" and "Bilinear Texture Mapping." This algorithm is used to map a screen pixel location to a corresponding point on the texture map. A weighted average attributes (color, alpha, etc.) of the four surrounding texels is computed and applied to the screen pixel. This process is continuously repeated for each pixel forming the object being textured.The term bilinear refers to the performing of interpolations in two dimensions (horizontal and vertical). The top and bottom pairs of each texel quadrant are averaged in horizontal and then their results are averaged in vertical. This method is widely used in conjunction with MIP mapping. Bilinear interpolation can be used where perfect image transformation with pixel matching is impossible so that any person can calculate and assign appropriate intensity values to pixels. Bilinear interpolation uses both fractional part and integer together to calculate the final pixel value according to four pixels.The fractional part is used as weighted value. It removes sharp and mosaic.

### *H. Template Matching:*

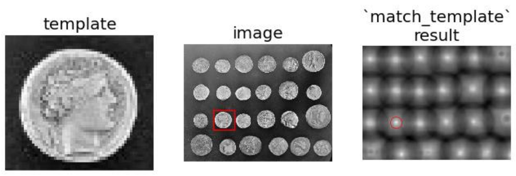
*1) What is Template Matching?*

Template Matching is a technique for finding areas of an image that match (are similar) to a template image (patch).We require two primary components:

1. **Source image (I):** The image in which we expect to find a match in Template image
2. **Template image (T):** The patch image which will be compared to the template image.

Template matching finds all the points inside an image which match a template. A template is simply a smaller image. Typically Template Matching algorithm is only used in highly controlled environments and doesn't work to well in natural scenes. A template matching algorithm works in computing a fit score for each pixel in the image and then looking for local maximums. Template Matching can be subdivided into two approaches: feature-based and template-based matching. The feature-based approach uses the features of the search and template image, such as edges or corners. As like primary match-measuring metrics to find the best matching location of the template in the source image. The template-based approach uses the entire template with a sum-comparing metric (using SAD, SSD,cross-correlation, etc.). That determines the best image location by testing all or a sample of the viable test locations that the template image may match up. For templates without strong features, or for when the bulk of the template image constitutes the matching image, a template-based approach is more effective.

Example-In this example template matching is used to identify the occurrence of an image patch (in this case, a sub-image centered on a single coin). Here, it returns a single match (the same coin), so the maximum value in the match_template result corresponds to the coin location. The other coins look similar, thus have local maxima; if you expect multiple matches, you should use a proper peak-finding function. The match-template function uses fast, normalized cross-correlation to find instances of the template in the image. The peaks in the output of match-template correspond to the origin (i.e. top-left corner) of the template.



A simple method of Template Matching uses a convolution mask (template), tailored to a specific feature of the search images, which we want to detect. This Template Matching technique can be easily performed on gray images or edge images. The convolution output will be at highest places where the image structure matches the mask structure, where large image values get multiplied by large mask values.

## IV. FUTURE WORK

A large collection of handwritten samples is a prerequisite to the proper performance of any Optical Character Recognition. As a future plan, we would like to enhance the approach using sufficiently large number of samples and extend the work by using all words in MODI language.

## V. CONCLUSION

As we know that very important ant vital documents were written in the Modi language .So we cannot afford to lose such precious knowledge about the various economic strategies and the ancient writing which can even help us in today's life. There are very few experts who have knowledge about the Modi script and hence it would take a long time to convert such documents into human readable languages.

Hence we have to convert these documents into English to preserve them forever and to extract important information after decoding them. So we have decided to work on this topic. With the help of our software one can easily convert any Modi character into English character and can extract important information from any Modi document.

## REFERENCES

[1] Sakal News Paper(9th July 2014)

[2] D. N. Besekar, R. J. Ramteke, *International Journal of Computer Applications*, vol. 64, no. 3, February 2013. "Study for Theoretical Analysis of Handwritten MODI Script – A Recognition Perspective".

[3] Lawrence Lo, 'ancientscripts.com A compendium of world-wide writing systems from prehistory to today', "MODI","www.ancientscripts.com", Accessed 28 March 2014 Accessed 28 March 2014

[4] Stephen Johnson (2006). Stephen Johnson on Digital Photography(http://books.google.com/books?id=0UVRXzF91 gcC& pg=PA17&dq=grayscale+ black-and-white-continuous-tone & ei=XlwqSdGVOILmkwTalPiIDw). O'Reilly. ISBN 0-596-52370-X.

[5] R. Brunelli, Template Matching Techniques in Computer Vision: Theory and Practice, Wiley, ISBN 978-0-470-517062,2009((http://eu.wiley.com/WileyCDA/WileyTitle/productCd-0470517069. HTML) TM book).

[6] A plugin for ImageJ (HTTP:// rsb.info.nih. gov/ij/plugins/otsu-thresholding. Html) using Otsu's method to do the threshold.

[7] David Lalmalsawma, India Insights, Reuters, Edition US, 7 Sept 2013. "India speaks 780 languages, 220 lost in last 50 years–survey", "apresearch.org/india-speaks-780-languages-220-lost-in-last-50-years-survey-india-insight", Accessed 28 March 2014

[8] Rajesh Khillari, "History of MODI Script", 30 May 2008, "http://modi-script.blogspot.in/2008/05/history-of-modiscript.html" Accessed 28 May 2014

[9] M.Y. Savant, Kriti Rakshana, a bi monthly publication of the national mission for menu script, vol 1 no 1, Auguest 2005, page 16. "Marathi MODI Script: Origin, Evolution Significance"http://www.namami.org/kriti%20rakshana7vol/1/final pdf/final newsletter.pdf. Accessed 28 March 2014

[10] Naren Ranadive, Randive's Indian Antiquities and Inscriptions, "The Origin and Development of Indian Writing System – MODI Script of Maharashtra", "http://narendranath.webs.com" Accessed 20 March 2014.