Preprocessing and Segregating Offline Gujarati Handwritten Datasheet for Character Recognition

Hetal R. Thaker
Atmiya Institute of Technology & science,
Kalawad Road,
Rajkot – Gujarat, India

C. K. Kumbharana, Ph.D
Head - Dept. of Computer Science,
Saurashtra University,
Rajkot – Gujarat, India

ABSTRACT

Ability of a computer to recognize handwritten character is a fascinating area of research due to the peculiarities involved in handwritten characters. Algorithm for Offline handwritten Character recognition differs as a result of diversities involved in writing with various language script. In a task of handwritten character recognition preprocessing and segmentation are two main phases and preliminary steps to be performed on acquired handwritten images. Achieving higher performance in handwritten character recognition depends on feature extraction process, which is highly influenced by preprocessing phase. Proposed work is a first step into an area of offline handwritten Gujarati character recognition. This paper presents algorithm for preprocessing image making it noise free and extracting region of interest for character recognition, segregating datasheet containing 30 characters written in Gujarati script to thirty different images having isolated characters. Further results obtained by employing proposed algorithm is discussed in this paper.

General Terms

Pattern Recognition, Off-line Handwriting recognition

Keywords

Character Recognition, Off-line handwriting recognition, Preprocessing, Gujarati handwritten character recognition

1. INTRODUCTION

Character recognition is divided into two types i.e. Online and Offline. In case of online handwritten character recognition input can be obtained when user writes using electronic device such as digitizer which can capture input and computer recognizes as user writes.

In case of offline handwritten character recognition document is digitized using scanner so hardcopy paper can be converted to softcopy. Digitized copy is an image stored in any graphics file format. Recognizing character from this image by computer is known as offline handwritten character recognition. There are many application areas such as searching from image, add, update or delete operation to characters in image etc.

Character recognition algorithm varies as diversities exist for language script and its characteristics such as direction of writing (i.e. left to right – English, Hindi, Gujarati), set of alphabets (i.e. English: A-Z, a-z), Nature of writing that defines how sentence are written (cursive script: English, Devanagari script: line at top of character and matras around). Further handwritten character algorithm varies due to the fact that every writer will have their own style of writing, even in different situation and temperament affects writing style by same writer.

For any pattern recognition task one or more steps such as preprocessing, segmentation, feature extraction, classification and post processing are involved. This paper presents algorithm for preprocessing and segregating of datasheet designed for collecting handwritten Gujarati character with results achieved by employing proposed algorithm. Gujarati is an official language of Gujarat – western part of India. Proposed work is divided into various sections as previous work, algorithm to preprocess and segregating datasheet, results obtained on employing algorithm, conclusion and future work.

2. PREVIOUS WORK

Humans are still outperforming than machines in an area of handwritten character recognition. Many researchers have contributed their work in area of both online and offline handwriting recognition in different language script by proposing various techniques and model. This paper discusses previous work of preprocessing and approach of collecting dataset in an area of offline handwritten character recognition.

Baheti M. J. et. al. [1] have reported that no standardize dataset of handwritten images for Gujarati script is available and hence proposed sample datasheet and collected handwritten Gujarati numbers from 80 writers belonging to various diversities and applied some preprocessing algorithms and employed k-nearest neighbor and principal component analysis classifier for Gujarati numeral recognition.

For handwritten Gujarati numeral recognition Apurva a. Desai [2] has collected Gujarati numerals from 300 writers and have applied preprocessing techniques to bring images into standard form, further how quality of paper influences writing and preprocessing required is discussed. Preprocessing task involved is adjustment of contrast, smoothing, resizing image to standard form i.e. 16x16 pixel. Using nearest neighborhood classification is performed.

For recognizing Kannada, telugu and devnagari handwritten numeral B.V. Dhandra et. al. [3] have proposed novel approach where noise is removed by median filter to remove scanning artifacts morphological operations are performed.

Kamal Moro et.al. [4] has reported that there is no standard database available for Gujarati and hence developed a database collecting handwritten characters from large number of writers and scanned at 300 dpi and have binarized and skeletonized images. For feature extraction horizontal and vertical and two diagonal profiles used and classified using neural network in a task of recognizing Gujarati handwritten numerical optical character.

Otsu's global thresholding method to extract the foreground from background and hilditch algorithm is applied for skeletonization is presented by N.Shanthi et. al. [5].

Prasad J. R. et. Al. [6] [7] have proposed a preprocessing approach in which they have used median filter to remove salt and pepper noise from the scanned images stored in png file format and have applied thinning to reduce character to minimum one pixel thickness, template matching for Gujarati character recognition. Various steps for template matching involves classification of templates, correlation analysis and calculating cross correlation coefficient which is repeated for every position and values were saved. Average overall recognition rate of 71.66 % is reported in an attempt. [8]

3. FORMAT OF HANDWRITTEN DATASHEET

For performing experimental work for recognizing handwritten character, dataset is required. Authors have collected handwritten data samples from five different writers in A4 size datasheet having grid of six rows and five columns producing 30 cells as shown in Fig 1(a). Each writer were given 10 datasheet producing 50 datasheets and 1500 handwritten characters. Sample handwritten character obtained from writer is shown as in Fig 1 (b). Further these datasheet are digitized using Brother DCP-7030 scanner at 300 dpi in png format.

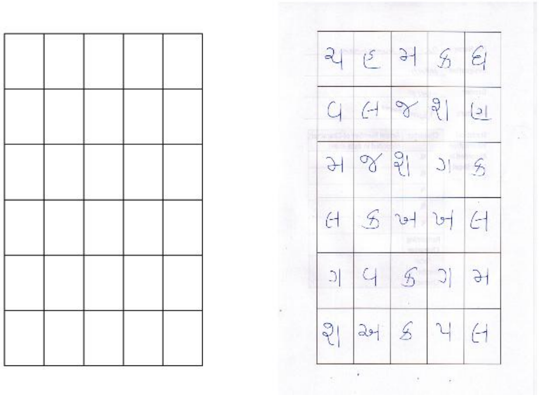

Fig: 1(a) Sample Blank Datasheet
(b) Sample Handwritten Data

4. ALGORITHM FOR PREPROCESSING AND FRAGMENTING DATASHEET

Rather than manually cropping this datasheet it is preprocessed and segregated by implementing proposed algorithm (Fig 2) to generate dataset for offline handwritten Gujarati character recognition.

4.1 RGB to Grayscale conversion

This step will transform true color image into grayscale intensity image. Digitized datasheet will be given as input and if image is not already grayscale it will be converted to grayscale. In RGB image each individual pixel have three components Red, Green and Blue. In Grayscale image will have matrix of values in the range of 0 and 1 where 0 represents black and 1 represents white pixel. While converting RGB image to grayscale image hue and saturation is eliminated and luminance is retained. [9] Fig 4(b) shows Grayscale conversion of RGB image shown in Fig 4(a).

Algorithm : Preprocessing and segregating handwritten Datasheet ***Input :*** *Raw RGB Image*	
Step:1	**RGB to Grayscale** <u>Input:</u> Raw RGB Image of handwritten datasheet <u>Output:</u> Grayscale Image
Step:2	**Reduce Noise and adjust contrast** <u>Input:</u> Grayscale Image <u>Output:</u> Grayscale Image with reduced noise * Determine threshold value for image * Adjust contrast * Apply Median filter to reduce noise
Step:3	**Convert image to binary image** <u>Input:</u> Image with reduced noise <u>Output:</u> Binary image containing 1 and 0
Step:4	**Segregate datasheet to extract cell data** <u>Input:</u> Binary Image of handwritten datasheet <u>Output:</u> 30 different images
Step:5	**Remove spurious pixel** <u>Input:</u> Image containing single character <u>Output:</u> image with concerned pixels Step is repeated for all 30 sub- images obtained in step4
Step:6	**Thinning** <u>Input:</u> Binary Image <u>Output:</u> Binary Image with Reduced lines to single pixel thickness
Step:7	**Detect edges and discard unwanted region** <u>Input:</u> Thin Image <u>Output:</u> Region with white pixel will be calculated and bounded, all black pixels around that boundary will be discarded
Output: *Preprocessed 30 sub-Image having isolated character*	

Fig: 2 Algorithm for Preprocessing and segregating handwritten Datasheet

4.2 Reduce Noise and adjust contrast

4.2.1 Thresholding

Thresholding is used to create binary image. [10] Sezgin et. al. [11] have categorized thresholding methods into six categories i.e. Histogram shape based methods, clustering based methods, Entropy based methods, Object attribute based methods, Spatial methods, local methods. For proposed algorithm Otsu's method is used for determining threshold value of grayscale image, which selects value by assuming bimodal distribution of gray level values and it minimizes within-class variance of two groups separated by thresholding operator [12].

4.2.2 Contrast Adjustment

Contrast can be adjusted using histogram equalization, for experimental work contrast is adjusted using contrast limited adaptive histogram method in which entire image is divide into smaller parts and histogram equalization is applied to all small parts and then result is interpolated [13] Contrast is adjusted in grayscale image and is shown in Fig 4(c).

4.2.3 Reducing Noise

Noise is an unwanted thing for image processing. Noise can be of many type i.e. salt and pepper noise, Gaussian noise, Speckle noise, Periodic noise. Dark pixels in bright regions and bright pixels in dark region can be found in Image having Salt and pepper noise which can be reduced using many methods such as Minimum filtering, Mean filtering, Maximum filtering, Rank Order filtering and Median filtering, For proposed algorithm Median Filter is applied for removing noise, result is shown in fig. 4(d).

4.3 Binarization of image

Threshold value is important factor while converting grayscale image to binary image in which luminance value above threshold value will be converted to 1 and remaining pixels will be converted to 0. [14] Fig 4(e) shows binary image.

4.4 Segregate datasheet to extract cell region

Fragmentation is required to obtain image file having single character for offline isolated handwritten Gujarati character recognition. For proposed algorithm fragmentation entails splitting of handwritten binary image into various sub-images in a way to obtain single handwritten Gujarati character in an image. To do fragmentation cell size is determined and cell data are extracted and stored as a separate images. As a result of fragmentation 30 different image files will be created which can be used further for handwritten character recognition task. Result of proposed algorithm to binary image is shown in fig 4(f).

4.5 Remove spurious pixel and Thinning

Spurious pixels are removed using morphological ‘spur’ operation and further thinning operation is performed on image for skeletonizing image. Reducing all lines to single pixel thickness which is achieved using morphological ‘thin’ operations applied infinite time until edges with one width thickness can be obtained. Fig 4(g) shows Thinning operation performed on image.

4.6 Detect Edges and discard unwanted region

Algorithm : Discarding undesirable region from image
Input : *Binary image*

```
m,n = size of input image
flag=0
for i=1 to m
    for j=1 to n
        if binary_image(i,j) is zero then
            first_row_zero = i
            flag=1
            break
        end
    end
    if flag=1 then break
    end
end
flag=0
for i=1 to n
    for j=1 to m
        if binary_image(j,i) is zero then
            first_col_zero = i
            flag=1
            break
        end
    end
    if flag=1 then break
   end
end
flag=0
for i=m to 1 by -1
    for j=n to 1 by -1
        if binary_image(i,j) is zero then
            last_row_zero = i
            flag=1
            break
        end
    end
    if flag=1 then break; end
end
flag=0
for i=n to 1 by -1
    for j=m to 1 by -1
        if binary_image(j,i) is zero then
            last_col_zero = i
            flag=1
            break
        end
    end
    if flag=1 then break; end
end
t1 = last_row_zero – first_row_zero + 1
t2 = last_col_zero – first_col_zero + 1
cropImg = corpped binary image with
              [first_column_zero first_row_zero t2 t1]
```

Output: Thirty separate images containing single isolated character per file.

Fig: 3 Algorithm to discard unwanted region from image

To remove unwanted part from image algorithm presented in Fig 3 is used where four boundary points are detected namely first row – col zero, last row – col zero using this pixels boundary is framed and cropped and all remaining pixels are discarded from image. Fig 4(h) represents output of applying above algorithm to images.

5. RESULT ANALYSIS

In Proposed approach one constraint imposed while collecting handwritten data that character should not touch boundary of cell in order to extract data correctly. Following table shows performance evaluated of proposed algorithm to preprocess and extract region from 50 datasheets. One datasheet contains 30 characters so 1500 images of isolated handwritten character is evaluated and result obtained is as per Table 1 and Table 2.

Table 1. Success Ratio of segregating datasheet

No. of Datasheet	Fragmented correctly	Accuracy
50	47	94%

Table 2. Success Ratio of preprocessing and discarding unwanted region from image

No. of isolated images	Preprocessed and extracted correctly	Accuracy
1500	1359	90.6%

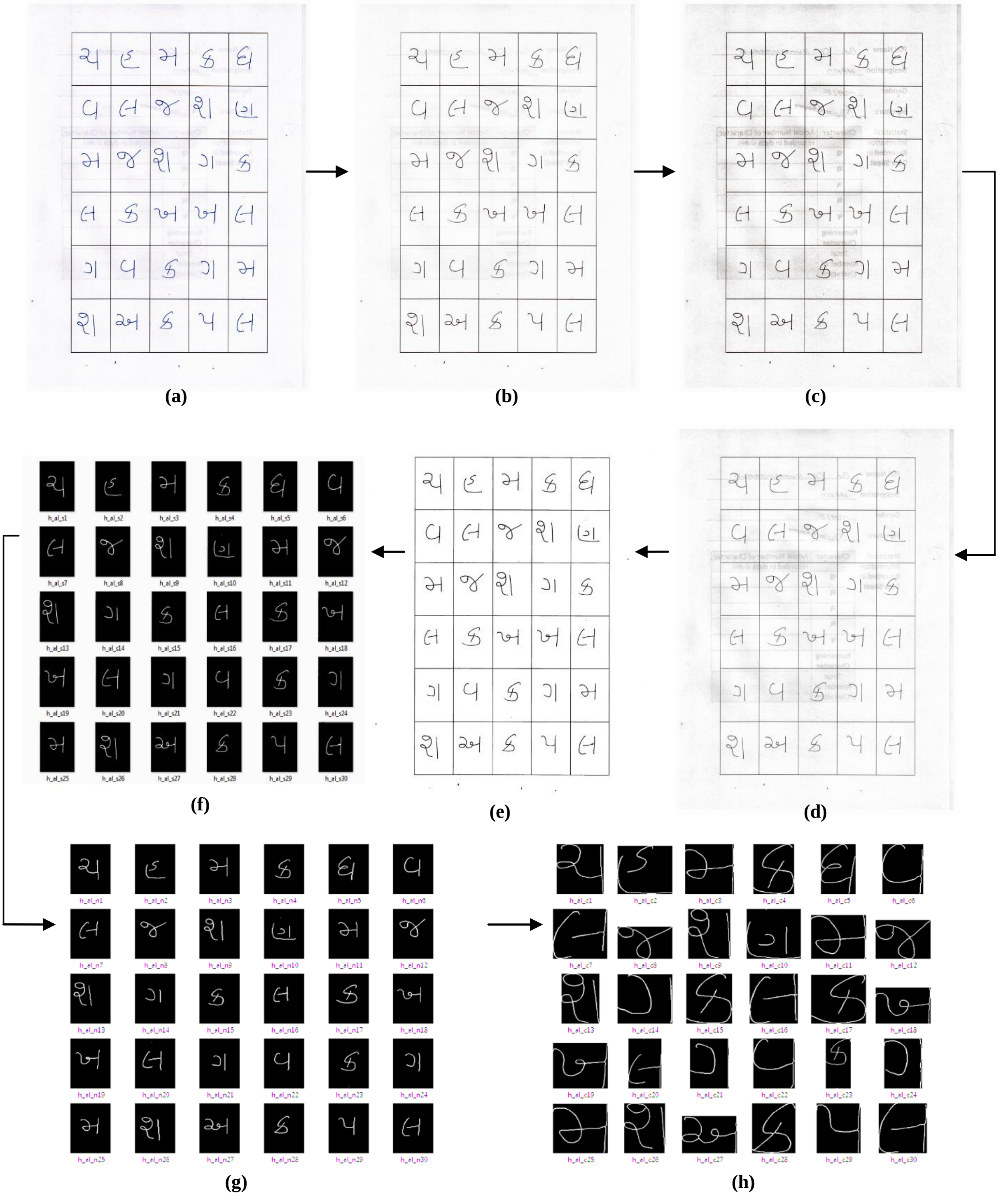

Fig: 4 (a) Scanned RGB Image (b) Grayscale Image (c) Adjustment of contrast in image (d) Reducing noise (e) Binary Image (f) Segregating datasheet into 30 different images (g) Thinning operation on image (h) Cropped image by removing undesirable region

Employing proposed algorithm on 50 handwritten datasheet yields 94% success for correctly segregating datasheet. As a result of segregating 1500 images containing isolated character obtained further preprocessing and algorithm to crop unwanted region is applied and authors are able to achieve 90.6% of accuracy. 6% of failure observed by employing proposed algorithm as indicated in Fig 5.

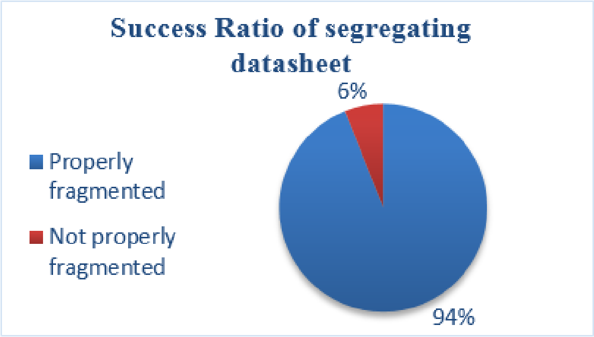

Fig: 5 Success Ratio of proposed algorithm for preprocessing and segregating

Fig 6 shows sample images where proposed algorithm doesn't yield correct result for cropping desirable region.

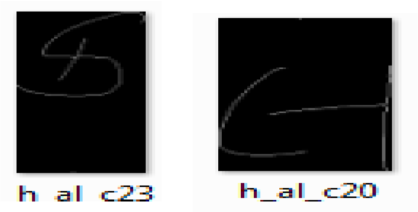

Fig: 6 Sample images where boundary copping algorithm is unsuccessful

6. CONCLUSION

For collecting handwritten data for Gujarati script a datasheet was designed. Instead of manually cropping it, character in cell of grid were segregated into various different images using proposed algorithm. With certain constraint imposed for filling up datasheet authors are able to achieve 94% accuracy. Further this images were preprocessed to provide images for next level of character recognition task where 90.6 % accuracy can be achieve. This work can further be extended by employing slant correction, size normalization. This approach can be utilized in developing dataset for offline handwritten character recognition for Gujarati script.

7. ACKNOWLEDGMENTS

Authors are very much thankful to all the writers who have contributed for providing handwritten input for proposed experimental work.

8. REFERENCES

[1] K. K. M. BAHETI M. J., "Comparison Of Classifiers For Gujarati Numeral Recognition," International Journal of Machine Intelligence, vol. 3, no. 3, pp. 160-163, 2011.

[2] A. A. Desai, "Gujarati handwritten numeral optical character reorganization through neural network," Pattern Recognition, vol. 43, 2010.

[3] R. M. H. K. B.Dhandra, "Telugu and Devanagari Handwritten Numeral Recognition with Probabilistic Neural Network : A Novel Approach, Architecture," pp. 83-88, 2010.

[4] M. f. Kamal moro, "Gujarati Handwritten Numeral Optical Character through neural network and skeletonization," jurnal of sistem komputer, vol. 3, no. 1, pp. 40-43, 2013.

[5] K. D. N. Shanthi, "A novel SVM-based handwritten Tamil character recognition system," pp. 173-180.

[6] J. Prasad, U. Kulkarni and R. Prasad, "Template Matching Algorithm for Gujarati Character," in In Proc. Of 2nd International Conference on Emerging Trends in Engineering and Technology (ICETET), 2009.

[7] J. Prasad, U. Kulkarni and R. Prasad, "Offline Handwritten Character Recognition of Gujarati script using Pattern Matching," in In Proc. Of 3rd International Conference on Anti-counterfeiting, Security, and Identification in Communication, 2009.

[8] K. K. V. Baheti M. J, "Recognition of Gujarati Numerals using Hybrid Approach and Neural Networks," in International Journal of Computer Applications, 2013.

[9] "http://nf.nci.org.au/facilities/software/Matlab/toolbox /images/rgb2gray.html," [Online].

[10] L. G. &. S. G. C. Shapiro, "Computer Vision," Prentice , 2002.

[11] M. S. a. B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," Journal of Electronic Imaging, vol. 13, no. 1, p. 146–165, 2004.

[12] "http://www.cse.unr.edu/~bebis/CS791E/Notes/Thresholding.pdf," [Online].

[13] "http://imageprocessingblog.com/histogram-adjustments-in-matlab-part-ii-equalization/," [Online].

[14] http://www.mathworks.in/help/images/ref/im2bw.html"
[Online].