2017 International Conference on Intelligent Computing and Control (I2C2)

A Font style classification system for English OCR

Bharath V
Department of Computer science
Amrita University
Mysore, Karnataka, India.
Bharath.03.yadav@gmail.com

N. Shobha Rani
Department of Computer science
Amrita University
Mysore, Karnataka, India.
n_shobharani@asas.mysore.amrita.edu

***Abstract*— The inclination of optical technologies like OCRlies in achieving higher recognition rates with optimal or reduced computational complexities. At present there exist optical technologies for automation of reading the text from document images with almost nearing to 100% accuracy. Especially, the Roman language OCR's are reliable and robust enough in producing higher accuracies by being able to recognize varying font styles of varying sizes. However for the font style/ size independent OCR's one of the main aspect is its computational complexity. It is significant concern to reduce the computational complexities involved in the process of character recognition through a font style / size independent OCR. In this paper, a technique for classification of the font style based on character image is proposed by employing the distance profile features with respect to left, right and diagonal directions of a character image. The major objective of this work is to reduce the complexity of the generic OCRsystems by font style recognition. The distance profile features of character images are fed to a support vector machine classifier. For experimentation, the training data sets are comprised of around 10 widely used font styles of upper case letters as well as lower case letters and numbers. The testing is conducted with the character images that are extracted from char74K dataset. The performance of algorithm is found to be satisfactory with an accuracy of80%.**

***Index Terms*— English OCR, font style classification, distance profiles, printed character recognition, SVM classifier.**

I. INTRODUCTION.

OCR (Optical Character Recognition) is a software or a system used for recognizing characters by a computer which are either printed or written. Normally an OCR system documents can be divided into 3 groups: Mono-font, Multi-font, and Omni-font. Here Mono-font OCR system considers documents which are written in one specific font, their accuracy is very high but they need a specific module for each font. Omni-font OCR system can handle characters of any font, and for this reason their accuracy is typically lower, their accuracy is related to the number and the similarity of the fonts under consideration. Whereas Multi-font OCR systems can handle only the subset of the fonts which are already existing. The various stages of an OCR includes:

- Input: also can be told as collection of data i.e., different type of images and different typeof formats.
- Image processing: once the images are collected the images are processed like noise removal, skew correction, brightness,contrast etc.
- Document and layout analysis: once the quality of the image is been enhanced it is then analysed for the line spacing, word spacing etc.
- Recognition: after the analysis is done the image is subjected for recognition like font style, font sizeetc.
- Verification and user interaction: once the recognition is done the system provides basic information like character coordinates, word and character recognition hypothesis etc.
- Output: after all the process is done the output is obtained.

Optical Character Recognition is one of the widely experimented research area since 1970's. Quite a good number of enhancements are integrated as a result of research outcomes. Few of the existing works relevant to the area of proposed research are discussed subsequently.

Ashwin et al [1] proposes an OCR system for printed text in south Indian language (kannada). The system segments the words extracted from the image of the document given to sub-character level pieces. The system uses the Support Vector Machine(SVM) to achieve the final recognition.

Ibrahim et al [2] proposes a method for a priori Arabic optical Recognition(AFR).here the words are segmented form the training set into symbols which are rescaled. Then they form templates and the training symbols which do not match the templates are formed into new templates. Then the fonts of the documents are matched with the templates and the fonts with the nearest matching are recorded. And the most frequently repeated font is the type font.

Kunte et al [3] present an OCR system for the identification of basic characters mostly consonants and vowels printed in one of the south Indian language (kannada). In the paper hu's invariant and Zernike movements are used to extract the features and recognize the fonts can handle multiple font types and font sizes. Based on these movements the characters are classified using Neural classifiers and has obtained a result of 96.8%.

Mohammed Javed et al [4] presents a paper where he gives an idea of learning and identifying the font size using simple line height features directly from the compressed text documents at the line level. In the model, mixed case text

documents are taken and segmented into compressed text lines and the ascender height and the lint height features are extracted which are intern used to obtain the regression line of the pattern. The modal gains an overall accuracy of 99.67%.

Shanthi et al [5] proposes a system for recognition of characters in south Indian language (Tamil) using Support Vector Machine(SVM). The documents are collected and stored as grey scale images which are then pre-processed to enhance the quality. For these density of the pixels are calculated for 64 different zones of image which are noted as their features. Then using these features the SVM is trained. And the result is achieved with the accuracy of 82.04%.

Victor et al [6] proposes a system where text in the images are automatically detected and extracted in 4 steps. The 1st step is focused on the region where the text may occur using texture segmentation. 2nd from the extracted texts strokes are recognized using features like spacing, alignment and similarity. In the 3rd step pre-processing is done. And at last it is passed through an OCR to recognize the font.

Abdelwahab et al [7] describes a work based on the global typographical features. This is mainly aimed to identify the weight, slope, typeface and size of the text without knowing about the content of the text. Here a multivariate Bayesian classifier is used for recognition of the fonts. The efficiency of the proposed method is tested using a set of 280 fonts and a 97% of accuracy was obtained.

Chaudhuri et al [8] proposes a system where the image is captured which is ten subjected to text correction, line segmentation, word and character segmentation, zone detection, text graphics separation using some modified and conventional. The characters are separated using zonal information and shape characteristics. Then the compound characters are recognized using template matching and tree classifier. For identifying root word and suffixes a dictionary-based error-correction process is used. And an accuracy of 95.50% is obtained.

II. PROPOSED METHODOLOGY

The proposed system for font style recognition and classification system mainly involves three main stages. In the stage 1, an input image is acquired for processing and proceeded for pre-processing and further directed for feature computation in stage two. Finally, the features computed are classified through a SVM classifier. Fig 1 depicts the block diagram of proposed system.

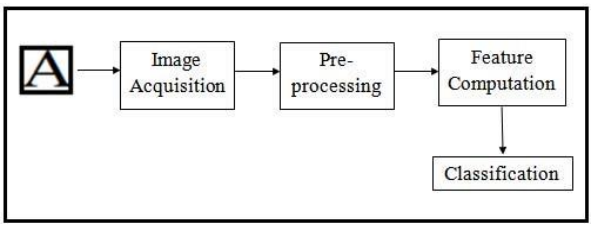

Fig 1: Block diagram of Font classification

III. IMAGE ACQUISITION

Image acquisition is the process of assuming an input to the systemfor subsequent processing. In the proposed method, a repository of character images C_1, C_2, C_3…C_n are organized within which a character image C_1 is assumed as input to font classification system. Figure 2 depicts few instances of character images organized in a repository. Each C_1 is subject to pre-processing in the subsequent stage.

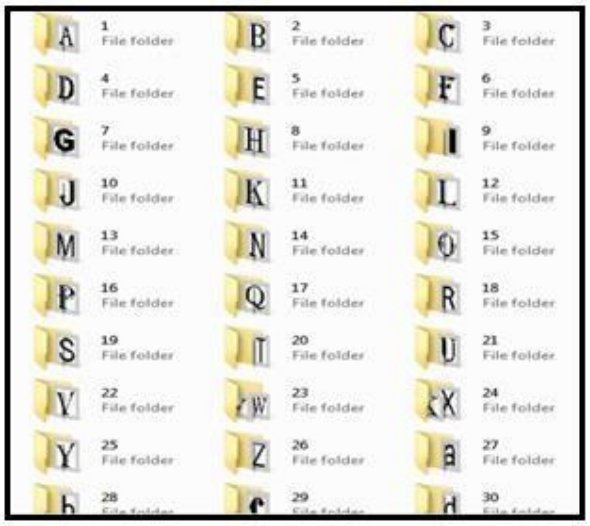

Figure 2: Instances of input characterimages

IV. PRE-PROCESSING

Each C_i is initially resized to a fixed dimension of around 35x35 so as to obtain consistent number of features for all inputs and maintain the robustness of system with respect to various changes that occurs in inputs. The rescaled images are transformed to binary images through Otsu's thresholding approach. The pre-processed images are directed towards the feature computation stage subsequently.

V. FEATURE COMPUTATION

Feature computation involves the process of extracting unique features from pre-processed images so that it can be employed for classification of fonts. In the proposed method a hybrid combination of features by extracting a set of distance profile features from pre-processed images. The combination of features computed from images include left distance profile, right distance profile and diagonal distance profile features.

A. Left distance profile features

Left distance profile feature is the distance between first column of the character image to first outer border pixel. The distance is computed as the sum of the pixels that are lying in between the first column and first outer border pixel of character image.

Let col_j where j=1,2,3…n, represents the columns of a character image C_i then a particular instance of left profile

distance L_dist(C_i) is the sum of the pixels $p_1, p_2, p_3 \ldots p_k$ such that the pixel p_k indicate the initial outer border pixel of the character image which is given by equation (1).

$$L - dist(C_i) = \sum_{i=1}^{k} p_i \quad (1)$$

Fig 3 emphasizes the left profile distance regions for a particular instance of a character image.

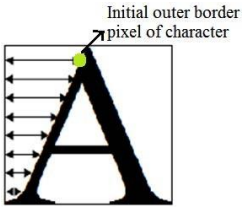

Fig 3: Left distance profile - Character image

The pixels $p_2, p_3 .. p_{k-1}$ are the intermediate background pixels (non-text pixels) in the image C_i.

B. Right distance profile features

Right distance profile features is the distance between the last outer border pixel of a character image to the last column of the character image. The distance is computed as the sum of the pixels between the last outer border pixel and the last column of the character image.

Let col_j where j=1,2,3…n, represents the columns of a character image C_i then a particular instance of right profile distance R_dist(C_i) is the sum of the pixels $p_{k+m}, p_{k+(m+1)}, p_{k+(m+2)} \ldots p_n$ such that the pixel P_n indicate the last column pixel of the character image and p_{k+m} is the last outer border pixel of a character which is given by equation (2).

$$R\_dist(C_i) = \sum_{i=k+m}^{n} p_i \quad (2)$$

The pixels $p_{k+(m+1)}, p_{k+(m+2)} \ldots p_{k+(m+n-1)}$ are the intermediate background pixels(non-text pixels) in the image C_i. Fig 4 emphasizes the left profile distance regions for a particular instance of a character image.

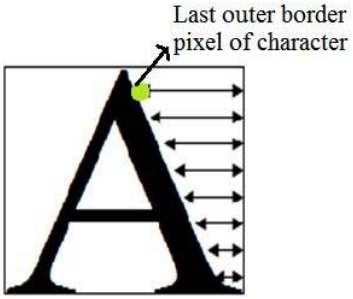

Fig 4: Right distance profile-Character image

C. Diagonal distance profile features

Diagonal distance profile features represents the diagonal distance from first pixel at position $C_i(1,1)$ (top left corner of C_i) to the outer border pixel of character in an image C_i by computing the sum of pixels in diagonal directions.

If m and n designate the number of rows and columns of a character image instance C_i and D_i_dist(C_i) represents the diagonal distance profile comprising of pixels $p(i, j), p(i+1, j+1), p(i+2, j+2) \ldots p(i+k, j+k)$ corresponding to distance from first pixel. Similarly $p(i, n), p(i+1, n-1), p(i+2, n-2) \ldots p(i+k, n-k)$ ind icates the distance from last pixel (top right corner of C_i) at position $p(1, last\_col)$ to the outer border pixel of character C_i. Similarly the diagonal distances from bottom left corner and right corner to the outer border pixel of character C_i comprises of pixels $p(m, j), p(m-1, j+1) \ldots p(m-k, j+k)$ and $p(m, n), p(m-1, n-1) \ldots p(m-k, n-k)$ where m is last row, j represents first column and n is last column. Fig 5 emphasizes the diagonal distance profiles computed with respect to four directions.

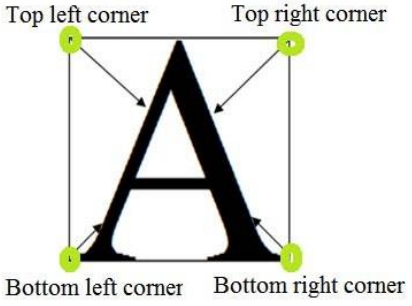

Fig 5: Diagonal distance profile-Character image

VI. FEATURE VECTOR GENERATION AND CLASSIFICATION

The feature vectors are generated for ten different font styles of upper case as well as lower case English alphabets. The entire dataset comprises of about 26x10 i.e., 260 upper case character instances and 260 lower case character instances leading to a total of around 520 training instances of character images. The test set is comprised of 260 character

image samples that are generated from various PDF English documents and other document images sources from web leading to five different font styles of about 26x5 upper case and 26x5 lower case character instances. Fig 6 and 7 depicts the few instances of training and test set samples and the table 1 represents the different font styles considered for experimentation.

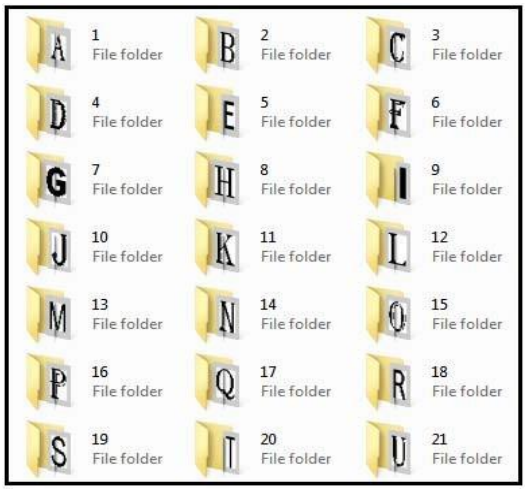

Fig 6: Few instances-Training samples composed forexperimentation

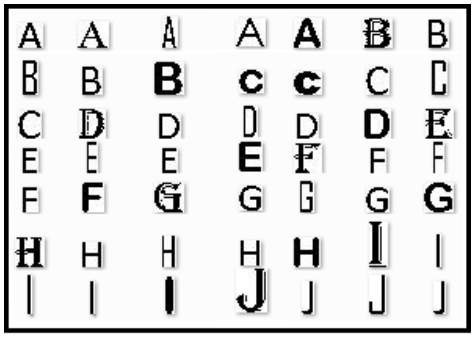

Figure 7: Few instances-Test samples extracted for experimentation

Table 1: Font styles employed forexperimentation

Character	Font style
A, a	Times new roman
A, a	Calibri
A, a	Cambria
A, a	Bodoni MT
A, a	Arial Narrow
A, a	Consolas
A, a	Arial
A, a	Arial black
A, a	Arial rounded MT bold
A, a	Arial Unicode MS

As each character image instance is rescaled to 35x35, the left distance profile of about 35 features, right distance profile of about 35 features and diagonal distance profile of 4 features leading to totally 74 features per character image are employed for generation of training feature vector.

In the proposed methodology, SVM classifier is employed for classification. The results of SVM classifier depicted for few of instances are depicted in the fig 8 in the form of a confusion matrix.

Test Character	No. of test samp les	Times new roman	Calibri	Cambria	Bodoni MT	Arial
A	5	2	1	1	0	0
B	5	2	2	0	0	1
C	5	4	0	1	0	0
D	5	3	1	0	1	0
E	5	2	0	0	0	3

Figure 8: Confusion matrix-Output of SVM classifier

Fig 9 depicts the accuracy of the font classification achieved through distance profile features and classified using SVM and KNN classifier. The number of characters considered in both cases are 130 characters comprising of both upper caseand lower case letters.

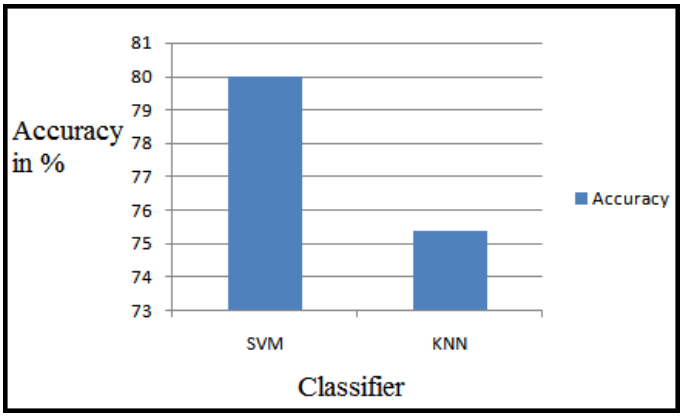

Fig 9: Performanceof classifiers - recognition offont styles

VII. CONCLUSION

Classification of font style optimizes the complexity of OCR by providing a scope for maintaining separate knowledge base for each type of font. The proposed method employs distance profile features with respect to left, right and diagonal orientations for font classification. The method employed provides satisfactory accuracy by leading an average accuracy of 80%. The work can be further improved by normalization of features so as to achieve more precision in the results.

ACKNOWLEDGMENT

We would like to express our heart-felt gratitude to our guide and our project coordinator Dr. N Shobha Rani for her valuable suggestions and guidance rendered throughout.

REFERENCES

[1] Ashwin, T. V., & Sastry, P. S. (2002). A font and size-independent OCR system for printed Kannada documents using support vector machines. *Sadhana*, *27*(1), 35-58.

[2] Abuhaiba, I. (2003). Arabic font recognition based on temp lates. *The International Arab Journal of Information Technology*, *1*, 33-39.

[3] Kunte, R. S., & Samuel, R. S. (2007). A simple and efficient optical character recognition system for basic symbols in printed Kannada text. *Sadhana*, *32*(5), 521.

[4] Javed, M ., Nagabhushan, P., & Chaudhuri, B. B. (2014). Automatic detection of font size straight from run length compressed text documents. *arXiv preprint arXiv:1402.4388*.

[5] Shanthi, N., & Duraiswamy, K. (2010). A novel SVM -based handwritten Tamil character recognition system. *Pattern Analysis and Applications*, *13*(2), 173-180.

[6] Wu, V., Manmatha, R., & Riseman, E. M. (1999). Textfinder: An automatic system to detect and recognize text in images. *IEEE Transactions on pattern analysis and machine intelligence*, *21*(11), 1224-1229.

[7] Zramdini, A., & Ingold, R. (1998). Optical font recognition using typographical features. *IEEE Transactions on pattern analysis and machine intelligence*, *20*(8), 877-882.

[8] Chaudhuri, B. B., & Pal, U. (1998). A complete printed Bangla OCR system. *Pattern recognition*, *31*(5), 531-549.

[9] Parthasarathi, V., Surya, M ., Akshay, B., Siva, K. M ., & Vasudevan, S. K. (2015). Smart control of traffic signal system using image processing. Indian Journal of Science and Technology, 8(16), 1.

[10] Ramanathan, R., Ponmathavan, S., Thaneshwaran, L., Nair, A. S., Valliappan, N., & Soman, K. P. (2009, December). Tamil font recognition using gabor filters and support vector machines. In Advances in Computing, Control, & Telecommunication Technologies, 2009. ACT'09. International Conference on (pp. 613-615). IEEE.

[11] Pushpa, B. R., Ashwin, M ., & Vivek, K. R. Robust Text Extraction for Automated Processing of M ulti-Lingual Personal Identity Documents.