Automatic Classification of Handwritten and Printed Text in ICR Boxes

Abhishek Jindal
Newgen Software Technologies Ltd.,
A-6 Satsang Vihar Marg, Qutab Institutional Area,
New Delhi – 110067, India
abhishek.jindal@newgen.co.in

Mohd Amir
Newgen Software Technologies Ltd.,
A-6 Satsang Vihar Marg, Qutab Institutional Area,
New Delhi – 110067, India
mohd.amir@newgen.co.in

Abstract—**Machine printed and handwritten texts intermixed appear in the ICR cells of variety of documents. Recognition techniques for machine printed and handwritten text in these document images are significantly different. It is necessary to separate these two types of texts and feed them to the respective engine - OCR (Optical Character Recognition) and ICR (Intelligent Character Recognition) engine to achieve optimal performance. This paper addresses the problem of classification of machine printed and handwritten text from acquired document images. Document processors can increase their productivity and classify handwritten and printed characters inside the ICR cells and feed their images to the appropriate OCR or ICR engine for better accuracy. The algorithm is tested on variety of forms and the recognition rate is calculated to be over 91%.**

Keywords—ICR, OCR, Form Processing, ICR cells, Machine Printed Text, Handwritten Text

I. INTRODUCTION

Paper-based forms like applications, checks and other documents are a popular medium for capturing data for many enterprises. The data captured is submitted for electronic processing and the content is fed into a business system. Information on these documents can be filled with machine print, handwriting or a combination of both inside the ICR cells. This can be done either manually or through automatic form processing. Manual data entry involves human involvement and thus it is time consuming and error prone. Automatic form processing on the other hand is fast and cost effective but the results need not be 100 percent accurate. Automatic form processing solutions work best on structured forms. Structured forms are static forms that have precisely defined page layouts such that templates can be built. Automatic data recognition technology aims to feed the data retrieved from the ICR cells to the appropriate engine: OCR engine or ICR engine which is the challenge. ICR is known to have the ability to read handwritten characters whereas OCR is having the ability to read the printed characters. Document processors can increase their accuracy and improve the timings by employing comprehensive automated recognition software that incorporates the best of OCR and ICR in a single solution.

An obstacle to ICR systems is the mixture of printed and handwritten text in the same image. Each text type should be processed using different methods in order to enhance the recognition accuracy. Previous works addressed the problem of identifying each type by various classification techniques. These works utilize neural networks [1-7]; employ linear polynomial for discrimination function [8], Fisher [9-12] and tree classifiers [13-14], Hidden Markov Model (HMM) [15] or minimal distance classifiers [16-17]. In this paper we propose the use of various characteristics of the text to classify them into handwritten or printed. The main advantage of this, compared with other classifiers, is its accuracy, efficiency, simplicity and the low computation complexity.

The various steps involved in classifying text in ICR cells of form image as handwritten or printed are pre-processing, segmentation at character level, feature extraction and the classification.

II. THE PROPOSED APPROACH

This system for classifying machine printed and handwritten text can be decomposed into three stages, as shown in Figure 1. It involves capturing the document, preprocessing the captured document, the segmentation of zones containing ICR cells to individual components, the extracted features of the components and the classification process executed by the system.

A. Form Image

It considers application forms for various objectives, such as account opening forms, loan processing forms, customer application forms etc. ICR cells containing printed and handwritten characters can be found all over these documents. In the proposed approach, document to be processed is captured by using a capturing device. Figure 2 shows an example of possible images to be processed.

B. Pre-processing

Pre-processing aims to get a clean and de-skewed image. It is a three-step process. In the first step, input form image is binarised using Otsu's threshold selection method [18]. This method reduces a Gray level image to a binary image. It assumes that the image to be thresholded contains two classes of pixels or bi-modal histogram (e.g. foreground and background) then calculates the optimum threshold separating those two classes so that their combined spread (intra-class variance) is minimal. The binarised image may contain some

978-1-4799-2572-8/14/$31.00 ©2014 IEEE

noise. In the second step, the noise is removed on the basis of connected component analysis [19]. Finally, document image is de-skewed [20-21]. Figure 3 shows the final pre-processed image.

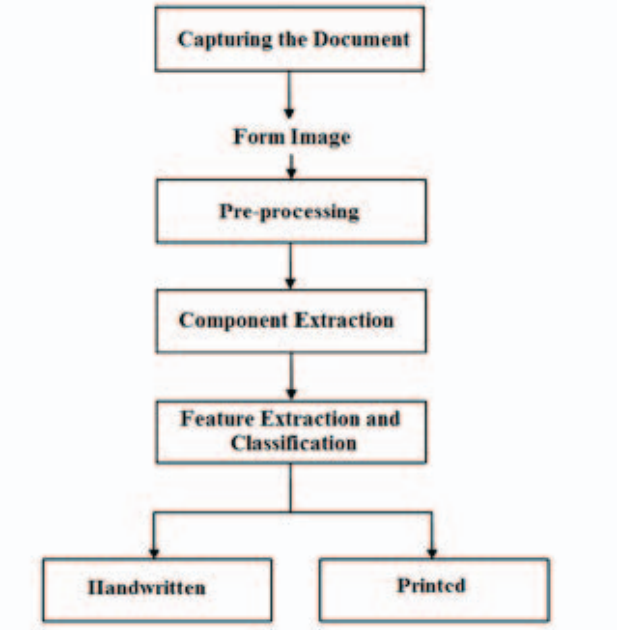

Figure 1. Overview of the system

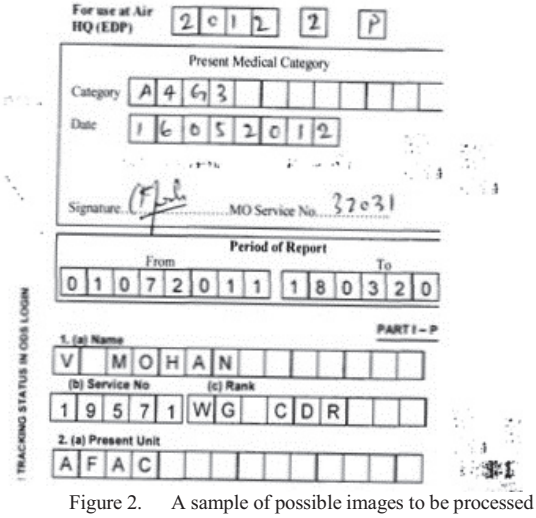

Figure 2. A sample of possible images to be processed

C. Component Extraction

In the Component Extraction, the zones are to be selected by the user on the pre-processed form image for extracting the features for a specific zone as shown in Figure 4. Selection of zones is to be carried out by the user because the proposed approach will declare the text inside the selected zone to be handwritten or printed. ICR cell detection is to be carried out for the selected zones on filled forms. ICR cell detection [22] involves identifying the total number of ICR cells with their positions. After detection of ICR cells, characters are to be extracted from the zones of the form. It is carried out by removing ICR cells boundary lines properly so that final image contains only ICR characters for better classification.

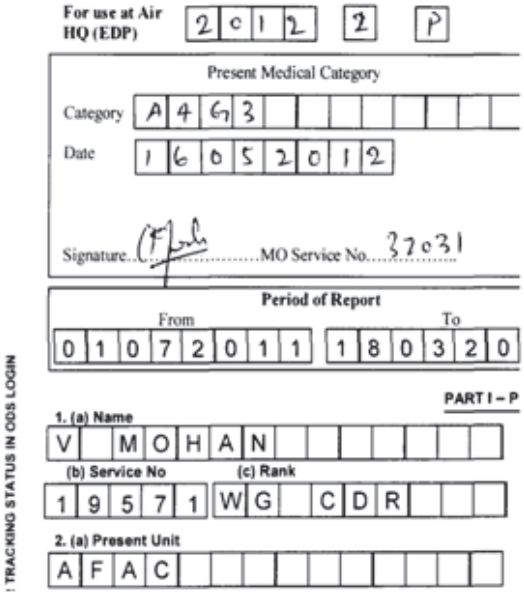

Figure 3. Final pre-processed image

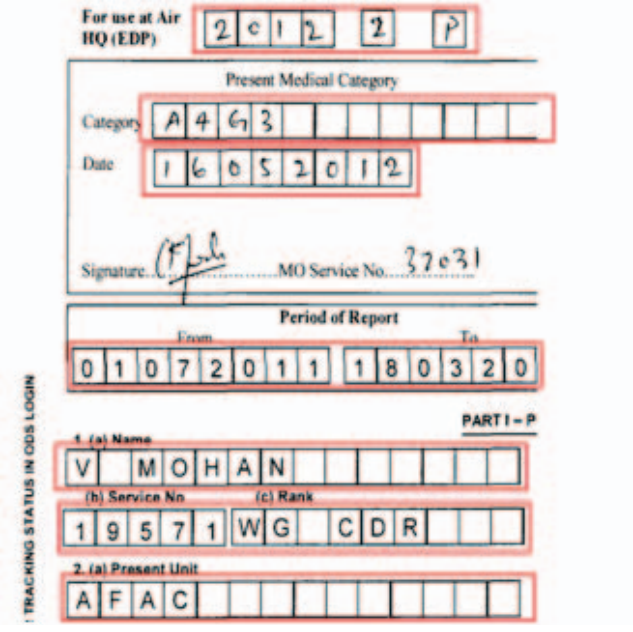

Figure 4. Selecting the zones for extracting the features zone-wise

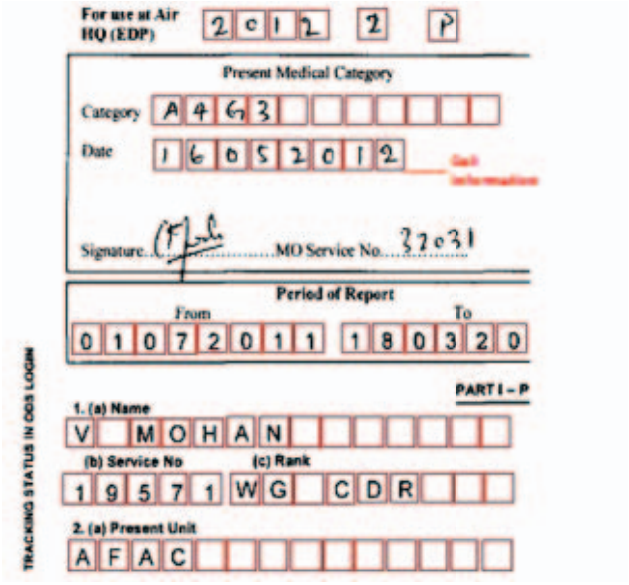

Figure 5. Output of Component Extraction step

To generate the classification approach, 1663 form zones scanned at 200 DPI and 300 DPI, having 947 handwritten zones and 716 printed zones are observed. Tolerance and threshold at various levels are computed on the experimental results of these images.

On the basis of experimental results, for each character of an ICR zone, horizontal and vertical smearing is carried out with a threshold of 0.0033 inches and 0.0067 inches respectively so as to join the broken components.

An eight-neighbor connected component-labeling algorithm [19] is then used to get the connected components. Figure 5 indicates the components extracted from the form image where cells drawn depict the cell information received from ICR cell detection.

D. Feature Extraction and Classification

Three features are defined and extracted for each zone. A zone consists of the continuous ICR cells in a single row. These features are extracted only for valid characters of a zone. Valid characters can be determined on the basis of the ICR cell positions, obtained in the previous step. Using cell positions, the number of components in each cell is identified. Ideally, each ICR cell should consists of maximum of one component. To identify valid characters, the heights of all the components of a cell are recorded. For a cell containing more than one component, the component with maximum height is considered only and rests of the components are removed. Then, arithmetic mean of the heights of all the components is computed as a feature. For each character, if the module of the difference of its height with the arithmetic mean of heights is greater than the tolerance, 0.06 inches, those characters are considered to be invalid characters. Tolerance is computed on the basis of experimental results. Feature Extraction and Classification phase is explained in subsequent section.

III. FEATURE EXTRACTION AND CLASSIFICATION

For the valid characters of each zone, three features that are extracted are: Inter-character gap (ICG), Character Height and Baseline.

A. Inter-Character Gap (ICG)

For each horizontal line of a zone, ICGs are computed. ICG is defined as the distance between characters of the adjacent cells.

- *First phase*

In the first phase, the inter-character gap between two characters i.e. the difference of the horizontal position of the mid-point of a valid character with the next valid character is recorded. Let a zone contains 'n' valid elements, which are sorted on the basis of their horizontal positions, having the horizontal position of mid-point of characters as $hMid_1$, $hMid_2$, $hMid_3$... $hMid_n$. The difference ICG_i, i.e. the inter-character gap between i^{th} character and $(i+1)^{th}$ character will be represented by (1). Figure 6 shows the inter-character gaps computed for a sample zone. In order to eliminate the blank spaces and invalid characters present in the ICR cells of the zone, Min (ICG_i), i.e. the minimum value of inter-character gaps is computed. If ICG_i is greater than (1.5 * Min (ICG_i)), it represents a blank space or an invalid character. α in (2), is rounded to the nearest whole number which is used to represent the total number of consecutive blank spaces or invalid characters between the two characters having ICG_i as the inter-character gap. ICG'_i is computed as in (3) which is the actual inter-character gap after removing blank spaces.

- *Second phase*

After removing the blank spaces and invalid characters from the first phase, ICG'_i i.e. the actual inter-character gap between i^{th} character and $(i+1)^{th}$ character is recorded. The arithmetic mean of the inter-character gap of all the valid characters of a zone is calculated. Let's consider a zone which contains n elements having (n-1) inter-character gaps obtained from first phase as ICG'_1, ICG'_2... $ICG'_{(n-1)}$. The arithmetic mean of all the valid inter-character gap of a zone will be computed as $\mu_{ICG(n-1)}$ which will be represented by (4). For each ICG'_i, the module of the difference of its inter-character gap with the arithmetic mean is stored as a feature. If the deviation of inter-character gap with arithmetic mean is greater than tolerance, i.e. 0.01 inches, then those gaps are eliminated from the bucket of acceptance. If total number of elements lying within the bucket of acceptance is (n-1), then the zone will be passed for further evaluation on the basis of other characteristics. Otherwise, the zone is considered to have handwritten characters.

$$ICG_i = hMid_{i+1} - hMid_i \quad \forall\ 1 \le i < n \ \&\ i \in N \tag{1}$$

$$if\left(ICG_i > \left(1.5 \times Min(ICG_i)\right)\right)$$
$$then\ \alpha = {ICG_i}/{Min(ICG_i)} \tag{2}$$
$$else\ \alpha = 0$$

Where ICG_i represents the inter-character gap between two characters and Min (ICG_i) represents the minimum of the inter-character gap between all the consecutive characters.

$$\text{if } (\alpha = 0) \text{ then } ICG_i' = ICG_i \quad (3)$$

$$else\ ICG_i' = ICG_i - \left((Round(\alpha) - 1) \times Min\ (ICG_i)\right)$$

$$\mu_{ICG(n-1)} = \left(\sum_{k=1}^{(n-1)} ICG_k'\right)/(n-1) \quad (4)$$

Figure 6. Identifying the midpoint of the characters and inter-character gap, ICG_i

B. Height

The heights of all the valid characters of a zone is recorded and passed to a process, which is using the arithmetic mean as the filter, as described below.

The arithmetic mean of the heights of valid characters of a zone is computed in this phase. If a zone contains n valid characters having heights h_1, h_2, h_3... h_n. Figure 7 shows the height of the characters computed for a sample zone. The arithmetic mean $\mu_{Height(n)}$ is represented by (5). For each valid character, the module of the difference of its height with the arithmetic mean of heights is stored as a feature. If the deviation of height of a valid character is greater than threshold, i.e. 0.01 inches, then those characters are eliminated from the bucket of acceptance. Tolerance is computed on the basis of experimental results which gives the threshold. If total number of elements lying within the bucket of acceptance is n, then the zone will be passed further evaluation on the basis of baseline. Otherwise, the zone is considered to have handwritten characters

$$\mu_{Height(n)} = \left(\sum_{k=1}^{n} h_k\right)/n \quad (5)$$

Figure 7. Identifying the height of the characters

C. Baseline

This feature is used when all the characters of a zone are topologically segmented. Here, to identify a baseline the distribution of the lowermost pixel of isolated components is noted down. It is observed that the distribution of valid characters in lowermost point is regular in machine-printed texts, and random in handwritten texts. The baseline of all the valid characters of a zone i.e. vertical position of the lowermost pixel of a character is recorded.

The arithmetic mean of the baselines of all the valid characters of a zone is calculated. Let a zone contains n valid elements having baseline b_1, b_2, b_3... bn. The mean of the baselines of all the valid characters of a zone will be computed as $\mu_{Baseline(n)}$ which will be represented by (6):

$$\mu_{Baseline(n)} = \left(\sum_{k=1}^{n} b_k\right)/n \quad (6)$$

Figure 8. Baseline of the characters and the tolerance

For each valid character, the module of the difference of its baseline with the $\mu_{Baseline(n)}$ is stored as a feature. If the deviation of baseline of a valid character is greater than threshold, i.e. 0.01 inches, then those characters are eliminated from the bucket of acceptance. Tolerance is computed on the basis of experimental results which gives the threshold. Figure 8 shows the baseline and Tolerance computed for a sample zone. If total number of elements lying within the bucket of acceptance is n, then the zone will be considered to have printed text. Otherwise, the zone is considered to have handwritten text. This process is carried out to eliminate those characters from the bucket which are having huge deviation in baseline from the other characters.

IV. EXPERIMENTS

Three features are extracted for each zone. To demonstrate the feasibility and the validity of the proposed approach, 967 zone images scanned at 200 DPI and 300 DPI containing 449 printed zones and 518 handwritten zones were tested.

Testing has been carried out on another 3392 zone images scanned at 300 DPI. Classification scheme is applied on 1665 printed zones and 1727 handwritten zones. TABLE I represents the recognition accuracy achieved from our solution.

TABLE. I RECOGNITION ACCURACY OF HANDWRITTEN AND PRINTED TEXT USING VALIDATING DATA

Type of Zone	*Total Number of Zones*	*Number of Zones Detected Correctly*	*Recognition Accuracy*
Handwritten	1727	1662	96.24
Printed	1665	1518	91.17

V. CONCLUSION

In this paper a set of new features to be extracted from images and an approach to find classification rules for

identifying printed and handwritten text in form documents is proposed. The system was implemented and tested on 3392 form images. Experiments show that the methodology being used is robust and applicable to a majority of document types. Moreover, the extracted features to represent the printed and handwritten words proposed make the system independent of the document layout in the identification task. Finally, as the approach presents good results by using only classification rules; it is also less time consuming then all others methodologies used till now. In future, to increase the recognition accuracy, other features such as stroke width and pixel distribution can be integrated with the proposed classification technique.

ACKNOWLEDGMENT

We are very grateful to the reviewers for their valuable comments. We also extend our gratitude to Mr. Raju Gupta, Mr. Prasad Nemmikanti, Mr. Mayank Kumar, Ms. Puja Lal and Mr. Dinesh Ganotra of Newgen Software Technologies Ltd. for giving their valuable suggestions during this work.

REFERENCES

[1] S. Imade, S. Tatsuta, and T. Wada, Segmentation and Classification for Mixed Text/Image Documents Using Neural Network , Proceedings of the Second International Conference on Document Analysis and Recognition, 20-22 Oct., pp. 930 - 934, 1993.

[2] S. Violante, R. Smith, and M. Reiss, A Computationally Efficient Technique for Discriminating Between Hand-Written and Printed Text, IEEE Colloquium on Document Image Processing and Multimedia Environments, 2 Nov.,pp. 17/1 - 17/7, 1995.

[3] I K. Kuhnke, L. Simoncini, and Z. M. Kovacs-V, A System for Machine- Written and Hand-Written Character Distinction, Proceedings of the Third International Conference on Document Analysis and Recognition,v. 2,14 - 16 Aug.,pp 811 - 814, 1995.

[4] J. E. B. Santos, B. Dubuisson, and F. Bortolozzi, A Non Contextual Approach for Textual Element Identification on Bank Cheque Images, IEEE International Conference on Systems, Man and Cybernetics, v. 4, pp 6 - 9, 2002.

[5] J. E. B. Santos, B. Dubuisson, and F. Bortolozzi, Characterizing and Distinguishing Text in Bank Cheque Images, Proceedings XV SIBGRAPI, pp. 203 - 209, 2002.

[6] F. Farooq, K. Sridharan, and V. Govindaraju, Identifying Handwritten Text in Mixed Documents, ICPR 2006, 18th International Conference on Pattern Recognition, v. 2, pp. 1142 - 1145, 2006.

[7] J. Koyama, M. Kato, and A. Hirose, Local-spectrum-based distinction between handwritten and machine-printed characters, 15th IEEE International Conference on Image Processing, 12-15 Oct., pp. 1021 - 1024, 2008.

[8] J. Franke, and M. Oberlander, Writing Style Detection by Statistical Combination of Classifiers in Form Reader Applications, Proceedings of the 2nd Intern. Conference on Document Analysis and Recognition, pp. 581 - 584, 1993.

[9] S. N. Srihari, Y. C. Shin, V. Ramanaprasad, and D. S. Lee, A System to Read Names and Addresses on Tax Forms, Proceedings of the IEEE, v. 84, n 7, pp. 1038 - 1049. DOI: 10.1109/5.503302, 1996.

[10] Y. Zheng, H. Li, and D. Doermann, The Segmentation and Identification of Handwriting in Noisy Document Images, , Document Analysis Systems V, Lecture Notes in Computer Science, v. 2423, pp. 95-105, 2002.

[11] Y. Zheng, H. Li, and D. Doermann, Text Identification in Noisy Document Images Using Markov Random Field, Proceedings of the Seventh International Conference on Document Analysis and Recognition, v. 1, pp. 599 - 603, 2003.

[12] Y. Zheng, H. Li, and D. Doermann, Machine Printed Text and Handwriting Identification in Noisy Document Images, , IEEE Transactions on Pattern Analysis and Machine Intelligence, v. 26, n 3, pp. 337 - 353, 2004.

[13] U. Pal, and B. B. Chaudhuri, Automatic separation of machine-printed and hand-written text lines, ICDAR '99. Proceedings of the Fifth International Conference on Document Analysis and Recognition, pp. 645-648, 1999.

[14] U. Pal, and B. B. Chaudhuri, Machine-printed and Hand-written Text Line Identification, Pattern Recognition Letters, v. 22, n 3 - 4, pp. 431 - 441, 2001.

[15] J. K. Guo, and M. Y. Ma, Separating Handwritten Material from Machine Printed Text Using Hidden Markov Models, , Proceedings. Sixth International Conference on Document Analysis and Recognition, pp. 439 - 443, 2001.

[16] E. Kavallieratou, and S. Stamatatos, Discrimination of Machine-Printed from Handwritten Text Using Simple Structural Characteristics, Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004, v. 1, 23 - 26 Aug., pp.437 - 440, 2004.

[17] E. Kavallieratou, S. Stamatatos, and H. Antonopoulou, Machine-Printed from Handwritten Text Discrimination, IWFHR-9 2004, 9th Intern. Workshop on Frontiers in Handwriting Recognition, 26-29 Oct., pp. 312 - 316, 2004.

[18] N. Otsu, A threshold selection method from gray-level histograms, IEEE Transactions on Systems, Man, and Cybernetics 9 (1) (1979) 62 –66.

[19] Dillencourt, M.B., Samet,H., and Tamminen,M., 1992. General approach to Connected-Component Labeling for Arbitrary Image Representations, In J.ACM Vol 39, No.2, 1992, pp. 253-280.

[20] Chih-Hong, K., Hon-Son, D., 2005. Skew Detection of Document Images Using Line Structural Information, In Third International Conference on Information Technology and Applications (ICITA'05) Volume 1.

[21] Shi, Z., Govindaraju, V., 2003. Skew Detection for Complex Document Images Using Fuzzy Runlength. In Seventh International Conference on Document Analysis and Recognition (ICDAR'03) - Volume 2.

[22] Agarwal, A., Kumar, P., & Kumar, S. (2006). ICR detection in filled form and form removal. In International Conference on Computer Application Theory and Application (VISAPP) (1) (pp. 271-276).