INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 5, Issue 2)
Available online at: www.ijariit.com

OCR to read embossed text from Credit/Debit card

Pranav Raka
rakapranav3@gmail.com
MIT College of Engineering, Pune, Maharashtra

Sakshi Pravin Agrawal
sakshigrawal64@gmail.com
MIT College of Engineering, Pune, Maharashtra

ABSTRACT

This paper is based on a real application in which the task was to recognize the digits and characters embossed on the credit/debit card i.e the card number and the expiry date. The problem was that the digit embossed are uneven. Using tesseract for OCR was not enough to recognize the digits accurately. The first problem was that to separate the digits and the second fact was that the separated characters or digits were distorted heavily. Challenge is to recognize the digits correctly. In this paper, we present an algorithm for credit/debit card number identification based on Optical Character Recognition using KNN. The algorithm is tested on five hundred credit and debit card images of different illumination of lights. When the images have captured the angle of view and distance were taken into consideration. The images were then preprocessed using an image processing algorithm. The resultant images were then fed to the OCR system. The achievement of this work was the correct identification of digits with 90% and zero false identification.

Keywords— *K-Nearest neighbours, Optical character recognition, Machine learning, Human-computer interaction*

1. INTRODUCTION

Recognition of characters and digits has been a unique ability of the human brain that is trained years and years from childhood and continuously improved in changing the environment. Though the functioning of the human brain has been unknown, it checks for the given pattern and recognizes the text and their meaning. Optical Character Recognition is the process of classification of optical pattern contained in a digital image corresponding to the alphanumeric or other characters [1]. OCR has become much popular in the last few decades, machine reading has grown rapidly due to much powerful OCR system. OCR technology allows converting scanned documents, pdf files and images from digital camera to editable and readable form [1]. OCR has become the most prominent and successful technological applications in the field of pattern recognition and artificial intelligence [2]. In this paper, we describe our solution for digit recognition of credit/debit card number for the case when digits are printed or embossed on the card in a high degree and moreover, it is difficult to distinguish them from the background. We present the result in human behaviour inspired approach.

2. PROBLEM DEFINITION

Our task was to recognize code embossed on the credit/ debit card. The code consists of account number and expiry date in a standard numeric form. We intend to create a prototype system which runs the OCR on the server and user access it from their browser at the client side. We faced two problems in this task the first was to separate the code from its background and second to recognize the characters correctly because they are embossed. Figure 1 and figure 2 show the difference between the normal card and embossed card.

Fig. 1: Normal Image

Fig. 2: Embossed Image

3. EXPECTED RESULT

The expected result of our system is that it will capture the image of the card and successfully recognize the 16 digit card number with at least 90% accuracy and expiry date. The image captured will follow the minimum quality so that the characters will be recognised correctly. After recognition of card number and expiry date, it will proceed to the checkout page for facial recognition.

4. OCR FEATURES

The OCR system provides the following features:

- No more retyping
- Saves Time
- No Human Mistakes
- Ease of Use

5. LITERATURE SURVEY

Computers understand alphanumeric characters as ASCII code which are typed on a keyboard. However, computers cannot distinguish characters and words from scanned images of paper documents. To solve this problem characters must first be converted to their ASCII equivalents before they can be recognized as readable text. Optical Character Recognition System (OCR) allows us to convert a document into electronic text. Character recognition is not a new problem but its roots can be traced back to systems before the inventions of computers. The earliest OCR systems were not computers but mechanical devices that were able to recognize characters, however, it had certain disadvantages of very slow speed and low accuracy rate. The early OCR systems lacked the capability of providing high speed and greater accuracy due to which not much research work was done during the ’60s and ’70s.OCR-A and OCR-B were developed by ANSI and EMCA in 1970, that provided comparatively acceptable recognition rates. In the last few years, great research work has been done in the field of OCR which has eventually lead to emergence of document image analysis, multi-lingual, handwritten and Omni-font OCRs. However, the ability of a machine to read text is still far below the human. Hence, more and more research is being done to provide greater accuracy and speed. Various methods have been proposed to increase the accuracy of optical character recognizers. The current challenge is to develop robust methods that remove as much as possible the typographical and noise restrictions while maintaining rates similar to those provided by limited-font commercial machines. Thus, currently, active research areas in OCR include handwriting recognition, and also the printed typewritten version of non-Roman scripts. There has not been the availability of any open source or commercial software available for complex languages like Urdu or Sindhi etc. Tesseract is in the top three OCR engines in terms of character accuracy in 1995. It is most widely used recent OCR technology. Available for all platforms including Linux, IOS, Windows. Tesseract up to version 2 could only accept TIFF images of simple one-column text as inputs. Since version 3.00 Tesseract has supported output text formatting, OCR positional information and page-layout analysis.

6. PROPOSED TECHNIQUES

Image Acquisition is the first process of Optical Character Recognition (OCR). The numeric code embossed on the card is captured in an image file Webcam. Initially, a histogram of about 100 images from the set of about five hundred credit and debit card were plotted. The histograms showed that data concentration is much more in the upper few grey levels of the histogram. As the images differ from each other in terms of size of the card and the degree of embossed codes, it is not possible to decide on a static global threshold because there could be a certain amount of loss of information in the same. Thus, preprocessing is required as a prerequisite to the OCR algorithm.

6.1 Preprocessing

6.1.1 Edge Detection: The first module of the preprocessing algorithm consists of horizontal and vertical edge detection. The edge detection is basically required to identify the characters on the card. This is a very important step. A Sobel mask is used because Sobel kernels are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. The mask is applied separately to the input image, to produce separate measurements of the gradient component in each orientation (Gx and Gy). These are then combined together to find the absolute magnitude of the gradient.

The gradient magnitude is given by mod G =qG2x + G2y

The process of edge detection is followed by segmentation, normalization and thinning.

6.1.2 Segmentation: In this module, we first horizontally segment out the credit card number from the image background. To do that we scan the image row-wise once from top to bottom and then from bottom to top, and discard all the rows till we have a row with some white pixels. The pixels which represent the credit card number characters have an intensity value of 255 (white) while the background has a value of 0 (black). A similar process is undertaken for vertical segmentation for which scanning is done column-wise. The column containing all black pixels is part of gaps between the rectangular boxes of the code. Identifying these regions, we separate each character of the credit card number in such a way that every individual character is segmented in rectangular boxes.

6.1.3 Normalization: The segmented characters of the credit card numbers are in rectangular boxes which have different sizes, so we normalize it to a fixed size of pixels by scaling. The rectangular boxes are then removed by discarding a few numbers of rows and columns along the boundary of the boxes on the normalized images. The decision of how many rows and columns should be discarded to get rid of the box without distorting the character was based on a study of hundreds of such images. Thus, after the box removal process, we have only the individual characters who have to be thinned in the next stage.

6.1.4 Thinning: Morphological thinning is performed on the individual image for reducing characters to single pixel thickness while preserving the full length of those lines (that is pixels at the extreme ends of the lines should not be affected). These characters are now given to KNN training/identification.

6.2 OCR using KNN

6.2.1 Image pre-processing: TextCleaner script by Fred Weinhaus has been used to remove the image background noise followed by an image sharpening step. Both these steps require ImageMagick library. Figure 3 is a denoised image.

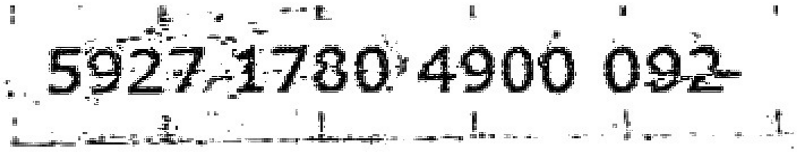

Fig. 3: Denoised image

6.2.2 Digits extraction and data preparation: Singling out each digit from an image using OpenCV find Contour operation did not produce reliable results due to noise. For this specific problem, it was more robust to detect the "bounding box" around the digits (image cropping) and then "single out" each digit out of the cropped image. The latter step is easy after finding the bounding box since each digit will have fixed coordinates relative to the upper-left corner of the cropped image.

Fig. 4: Top image cropping

The same technique is used for left and right crop.

Fig. 5: Cropped image

6.2.3 Digit extraction: Now that the bounding box is detected, it should be easy to single out each digit since each digit will have pre-fixed coordinates relative to the top-left corner of the cropped image. We have applied the above code on a set of images and manually sorted the images of each digit into separate folders labelled from 0 to 9 to create my training/testing dataset.

6.2.4 Feature extraction: Feature extraction or feature engineering is the process of identifying the unique characteristics of an input to enables a Machine Learning algorithm work. Of particular interest is the Histogram of Oriented Gradients (HOG) which has been successfully used in many OCR applications to extract handwritten text. Figure 6 below illustrates applying HOG on an image producing a vector of 200 values (that is features).

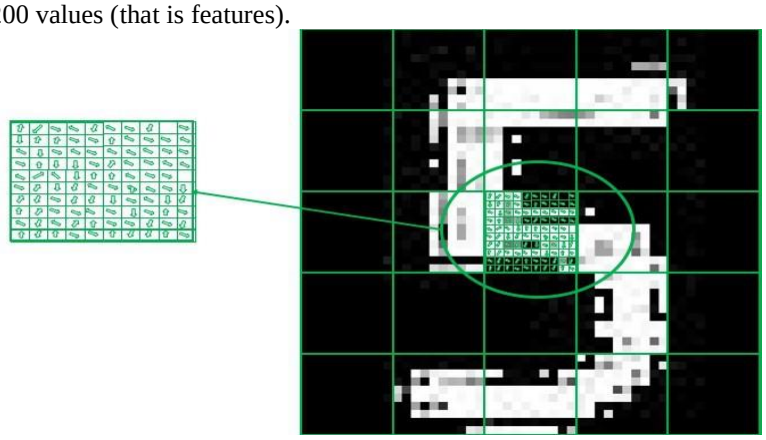

Fig. 6: Illustration of Histogram of Oriented Gradients (HOG)

6.2.5 Training: Now that we created the training dataset and stored it into features and features_label arrays, we then divided our training sets into training and test sets using sklearn function train_test_split and used the result to train a k-NN classifier and finally saved the model.

7. APPLICATIONS

OCR has been found into numerous sorts of area, for example, receipt OCR, receipt OCR, and check OCR, legitimate charging report OCR. They can be utilized for data section for business reports, for example, check, visa, receipt, bank explanation and receipt. Programmed number plate recognition.

8. CONCLUSION

The proposed credit card number-recognition algorithm based on OCR using KNN is well suited for intelligent systems. After performing all the steps, we are getting an accuracy of 90%. The achievements of the proposed system are that it gives considerably high value for Correct Identification Rate (CIR) along with zero Wrong Identification Rate (WIR). This is important because even a single instance of wrong identification of the card number may lead to incorrect interpretations, which can be catastrophic in certain applications. This system is built to ease the work of the users and a more smooth process.

9. REFERENCES

[1] Maninder Kaur," A Brief Review on Optical Character Recognition Techniques", International Journal of Computer Science and Mobile Computing, Vol.6 Issue.2, February-2017, pg. 95-100.
[2] A. Chaudhuri," Optical Character Recognition Systems for Different.
[3] R. Lienhart and A. Wernicke Localizing and segmenting text in images and videos, IEEE Transactions on Circuits and Systems for Video Technology, vol. 12 (4), 2002, pp. 256-268.
[4] M. Sarfraz, S.N. Nawaz, and A. Al-Khuraidly, Offline Arabic text recognition system, International Conference on Geometric Modeling and Graphics, 2003, pp. 30-35.
[5] D. Doermann, J. Liang, and H. Li, Progress in Camera-Based Document Image Analysis, Proceedings of the 7th International Conference on Document Analysis and Recognition (ICDAR 2003), 2003.
[6] X. Chen, J. Yang, J. Zhang and A. Waibel, Automatic detection and recognition of signs from natural scenes, IEEE Transactions on Image Processing, vol. 13 (1), 2004, pp. 87-99.
[7] S. Mariani, M. Gori, and G. Soda, Artificial neural networks for document analysis and recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27 (1), 2005, pp. 23-35.
[8] Rafael C. Gonzalez, Richard E.Woods and Steven L. Eddins, Digital Image Processing using MATLAB, Prentice Hall, 2004.
[9] Vilem Novak, Petr Hurtik, Hashim Habiballa" Recognition of distorted characters printed on metal using the fuzzy logic method.

