ISSN: 0975-766X
CODEN: IJPTFI
Research Article

Available Online through
www.ijptonline.com

SMART CODE COMPILER

J.Surender[1], G.Kapilya[2]
Student at Saveetha School of Engineering[1]
Assistant professor at Saveetha School of Engineering[2]
Email: samsunder008@gmail.com

Received on: 02.10.2016 *Accepted on: 28.10.2016*

Abstract

Compiling a programming code is always a tedious task as it involves typing the code in a suitable text editor, saving it, then opening the compiler, and then opening the particular source file and then running it. This involves a very long process. After the invention of smart phones the world has become much smarter each and every device we use became smart like smart tv, smart watch etc. so, this paper deals with compiling a programming language with just a snapshot of the source code. With the help of optical character recognizer (ocr) and mobile compiler this could be made possible.

Keywords: Optical character recognizer, Mobile compiler, Smart phones.

1. Introduction

This idea can be implemented using some techniques in image processing. The first one is converting an image which consists of text into an editable text format file. With the help of optical character recognizer we can convert such image to text. With the use google’s open-source ocr engine or tesseract or any other text recognition ocr engine it can be performed. Or we can also use some already available open-source apps. Until now there are so many softwares and applications that can scan an image and detect the text present in it. The text can be computerized or hand written, and in case of compiling there are also application available for that purpose. So by combining the features of these two a new single application that can convert and compile in a single stretch can be evolved.

2. Existing Model

As discussed above in the introduction there is no application to perform this function. But there are separate applications to for recognition and compiling. Even though with the use of these applications, it is not possible to render the output of the character recognizer and then feed it as input the compiler since they are developed in different environment and will not be supported.

The existing ocr models has a number of error factors like incorrect recognition, inappropriate rejection of correct text, less accuracy etc. And when comes to compiling there are lot of applications for smart phones which will only take the typed text as input. This drawback can be rectified by the proposed idea and a single smart application will work best for this purpose.

3. Proposed Model

In this model, as mentioned before only snapshot is the work that has to be done by the user, rest all operations are automated. So as soon as the application starts it will open up the camera or ocr reader then the user has to capture the image of the source code that he wants to compile. Once the image is captured, the codes are scanned, detected, and it will be converted into a editable text file format. This will be the output from the ocr reader which serves as the input to the compiler. As soon as the ocr completes its process, the compiler starts up to compile the ocr output and the result is generated. Here the ocr is made with several improvements in accuracy, fault tolerance and some error factors like recognition rate, rejection rate, error rate etc to get better results. And some common errors like variation in shape, size and spacing in case of detecting handwritten texts will be minimized. And if the captured image is pixilated or deformed or it contains any other defects which will eventually affect the ocr reader, then the application will prompt the user to once again capture the image for processing.

4. Architecture and Design: The design is simple like the other ocr reader and the compilers with minimal changes.

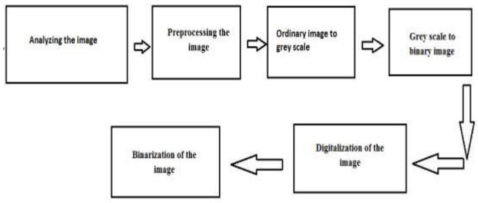
Analyzing the image
Preprocessing the image
Ordinary image to grey scale
Grey scale to binary image
Digitalization of the image
Binarization of the image

Fig1. Basic operation of the ocr.

The fig.1 briefly explains about the basic working of the common ocr reader. The process start when the image is captured, first the image is analyzed then the preprocessing is done. The preprocessing involves filtering the noise, fixing any deformation in the image and adjusting some of the properties of the image like brightness, sharpness, contrast etc. To obtain the text from the image. Then the process takes place as shown in the diagram.

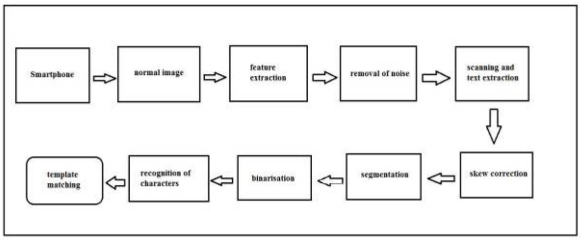

Fig.2 Architecture of the proposed model.

Fig.2 Describes about the internal workings starting from the image capturing to the final stage of text recognition. At last the recognized text is fed into an editable text file for further processing.

5. Implementation

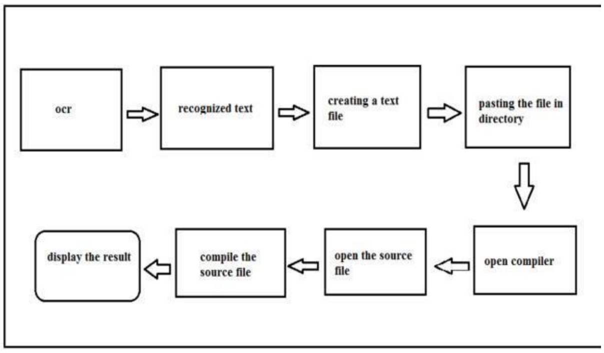

Fig.3 Implementing the recognized text in compiler.

The implementation can be done by the following methods or techniques. Some of the common approaches in ocr reader are

□ 5.1 Matrix matching

□ 5.2 Fuzzy logic

□ 5.3 Feature extraction

□ 5.4 Structural analysis

□ 5.5 Neural networks

5.1 Matrix matching

The most commonly used technique in ocr reader. It works by word by word matching of the given text in a matrix with the index. It is the most strong recognition for a mono-font or a single word or page. The problem occurs when it comes to using it for recognition of large number of text which consists of texts with different fonts, images, graphics and other details.

5.2 Fuzzy logic

It includes a human-way of logical thinking, mostly used in cases where the answer is to be true or false, yes or no or any other two or more options. It is similar to that of the Boolean logic.

5.3 Feature extraction

This method is very much suitable for almost every ocr, it scans and extracts all the text word by word. Here the accuracy is higher since this also removes noise and other factors in the image which affects the recognition. Hence is it perfect for any ocr which can be used in scanners, recognizers, and high quality images.

5.4 Structural analysis

Structural analysis basically works by recognition with the help of structures of the text like size of the text, shape, skew, vertical and horizontal histogram etc. Capability and accuracy is more in terms of low quality text and scripts.

5.5 Neural networks

Working is similar to that of the human neural system, it uses artificial intelligence to detect and recognize. It follows a specific pattern for identifying the text from image, takes the samples of the pixels and analyses. This technique is still developing as it is much advanced.

6. Conclusion

Thus by using this model even handwritten or printed or any form of text can be directly converted into your editable source code file and it makes programming and studying of programming language much easy and smart.

References

1. Ayatullah Faruk Mollah[1], Nabamita Majumder[2], Subhadip Basu[3] and Mita Nasipuri[4] ,Design of an Optical Character Recognition System for Camera-based Handheld Devices.
2. G.Vamvakas, B.Gatos, N. Stamatopoulos, and S.J.Perantonis Computational Intelligence Laboratory, Institute of Informatics and Telecommunications, National Center for Scientific Research “Demokritos”, GR-153 10 Agia Paraskevi, Athens, Greece.
3. Sarika Pansare1, Dhanshree Joshi2, A Survey on Optical Character Recognition Techniques, International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064 Impact Factor (2012): 3.358.
4. Ravina Mithe, Supriya Indalkar, Nilam Divekar, “Optical Character Recognition”, International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-2, Issue-1, March 2013.
5. Shalin A. Chopra1, Amit A. Ghadge2, Onkar A. Padwal3, Karan S. Punjabi4, Prof. Gandhali S. Gurjar5,” Optical Character Recognition”, International Journal of Advanced Research in Computer and Communication Engineering, Vol. 3, Issue 1, January 2014.
6. Er. Neetu Bhatia Kurukshetra institute of Technology & Management Kurukshetra, India.” Optical Character Recognition Techniques: A Review” Bhatia, International Journal of Advanced Research in Computer Science and Software Engineering 4(5), May - 2014, pp. 1219-1223.
7. Sukhpreet Singh M.tech Student, Dept. of Computer Engineering, YCOE Talwandi Sabo BP. India,” Optical

Character Recognition Techniques: A Survey", Vol. 4, No. 6 June 2013 ISSN 2079-8407 Journal of Emerging Trends in Computing and Information Sciences © 2009-2013 CIS Journal.

8. Namrata Dave G H Patel College of Engg. & Tech Gujarat Technological University Gujarat," Segmentation Methods for Hand Written Character Recognition" INDIA, International Journal of Signal Processing, Image Processing and Pattern Recognition Vol. 8, No. 4 (2015), pp. 155-164. http://dx.doi.org/10.14257/ijsip.2015.8.4.14namrata.dave@gmail.com
9. Rohit Dayama, Anil Chatla, Heena Shaikh, Medha Kulkarni PVPP College of Engineering, Sion, Mumbai, India" Android Based Meter Reading Using OCR" Rohit Dayama et al, International Journal of Computer Science and Mobile Computing, Vol.3 Issue.3, March- 2014, pg. 536-539 Email:nda12374@gmail.com, anilchatla7777@gmail.com,skheena.azmi@gmail.com,medha.kulkarni@yahoo.com
10. Amarjot Singh, Ketan Bacchuwar, and Akshay Bhasin," A Survey of OCR Applications" International Journal of Machine Learning and Computing, Vol. 2, No. 3, June 2012