International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 04 | Apr 2021 www.irjet.net p-ISSN: 2395-0072

Machine Learning based Optical Character Recognition (OCR) Scanner

Deep Bhupesh Patel[1], Muskan Raj[2], Jainam Himanshu Shah[3], Sejal Thakkar[4]

[1-3]*Student, Department of Computer Engineering, Indus University, Gujarat, India*
[4]*Assistant Professor, Department of Computer Engineering, Indus University, Gujarat, India*

Abstract - *Optical character recognition (OCR) as a typical machine learning challenge has been a persistent topic in distinct applications in the field of education, healthcare, legal industries, insurance and to convert different types of electronic documents, such as insurance and scanned documents and, PDF files into fully editable and searchable text data. The speedy generation of digital images daily prioritizes OCR as a vital and foundational tool for data analysis. We have been able to save a fair amount of effort in processing, creating, and saving electronic documents, adapting them to different purposes with the help of OCR applications. This OCR system will be helpful to the students who are facing problem in converting hand-written image to digital text PDF file.*

Keywords: OCR, Text to PDF converter, Digital image, Electronic document.

1. INTRODUCTION

Machine Learning-based Optical character recognition (OCR) Scanner will convert images of a typed, handwritten or printed text into machine-encoded text. It has been man's ancient dream to develop machines which replicate human functions. One such replication of human functions is reading of documents encompassing different forms of text. Over the last few decades machine reading has grown from dream to reality through the development of sophisticated and robust Optical character recognition (OCR) systems [1].

The focus of this application is to help various educators, lecturers, and students to make a text document of their handwritten notes. The process of character recognition can be divided into two parts, namely, printed and handwritten character recognition. The printed documents a further be divided into two parts: good quality printed documents and degraded printed documents. Handwritten character recognition has been divided into offline and online character recognition [2].

1.1 Need & Motivation

There are many OCR based applications available that can extract text from an image but they lack the accuracy to provide the same function for han+dwritten text. In addition, the process of typing every word would be a time consuming and daunting task.

Another benefit of converting handwritten notes into a computerized text document is that we do not need to worry about the maintenance of the book. In addition, new information can be added and modified something that cannot be done on a simple scanned text document.

2. SCOPE

2.1 Functionalities

Using machine learning, the application will be able to convert any handwritten text to machine-encoded text. A free open-source OCR engine namely Tesseract OCR will be utilized. It will also convert hand-drawn diagrams to machine-encoded diagrams provided that the user mentions the area containing the diagram.

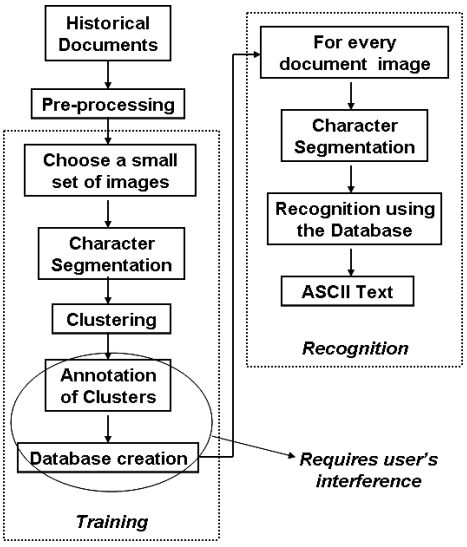

Figure 1. Flowchart of the proposed OCR methodology[4]

The app will also detect grammar, spelling, punctuation, error and correct them; it will also notify the user of those automatic corrections so that they can be undone.

2.2 Limitations

Due to varying handwritings, some text might be misinterpreted and it might not be detected by the user which may lead to misinformation and errors. While converting text the structure of the page might get altered which would further require manual editing.

3. LITERATURE SURVEY

The early OCR devices all required expensive scanners and special-purpose electronic or optical hardware: the IBM 1975 Optical Page Reader for reading typed earnings reports at the Social Security Administration cost over three million dollars (it displaced several dozen keypunch operators) [5]. A faculty member has to teach more than one subject and students have to learn numerous subjects as well. Both of them need to maintain proper notes of those subjects. One of the earliest OCR systems was developed in the 1940s, with the advancement in the technology over the time, the system became more robust to deal with both printed, and handwritten characters and this led to the commercial availability of the OCR machines [3]. Currently, students and faculties are maintaining handwritten notes which need to be taken care of all the time, this can be a troublesome task, especially for students who have to preserve these notes for as long as 4 yrs.

The typical hardware to collect data is a digitizing tablet which is electromagnetic or pressure sensitive. When the user writes on the tablet, the successive movements of the pen are transformed to a series of electronic signal which is memorized and analyzed by the computer [4]. This problem is solved by various mobile scanner applications available that takes pictures of all the notes and preserves them in pdf format. This solves the storage and preservation problem faced by everyone. However, the problem with these scanned notes is that once they are prepared one cannot edit them. The task of performing any large modification to these notes would be problematic. Also in a scanned format, the notes are in handwritten text, which would be difficult for others to understand.

Again, these complications can be overcome using various OCR applications that convert handwritten text to machine encoded text. But handwritten notes have more than text, it contains various diagrams, labelling, and a structure, which are not followed by these OCR applications.

This project would use machine learning to detect the location of the text and find a pattern so that the structure of notes is maintained. It will have the normal OCR functions along with functionalities to convert diagrams into a machine encoded diagram.

3.1 Objectives

- To help students, lecturers, staff, instructors maintain notes, which can be easily misplaced or lost.
- Save the trouble of storing numerous books.
- To relieve the task of maintaining the book's condition.

4. INCREMENTAL MODEL

Incremental Model is a software development model where requirements are divided into multiple standalone modules of the software development cycle. There are several iterations of this model. Each iteration expands the system's functionality. The first increment is frequently a core service that addresses the most basic requirements.

As the size of this project is small, hence no major planning or design is required. The whole system can be easily broken down into modules.

Based on the type and size of the project, the incremental model is the best suited model as its basic requirements can be covered in its first increment and additional functionalities can be added during further iterations.

5. FEASIBILITY STUDY

5.1 Economic Feasibility

Data generated through this system can be stored in a computer which will save the storage space required for keeping the physical copy of the file.

5.2 Technical Feasibility

There are numerous companies that have used OCR technology and also many OCR engines are available for free hence in terms of Technical feasibility this project is very much stable and established.

5.3 Operational Feasibility

As seen earlier, there are many problems faced by users for storing physical files, hence they user will automatically adapt to the new system as this alternative satisfies and solves the user's requirements and problems.

6. SYSTEM ARCHITECTURE

6.1 Grid Computing

Grid computing is a distributed architecture of large numbers of computers connected to solve a complex problem which can be used to accomplish complex tasks related to OCR. Servers or personal computers run autonomous tasks and are closely connected by the Internet or low-speed networks in grid-computing design. Computers may connect directly or via scheduling systems.

6.2 How Does Grid Computing Work?

In general, a grid computing system requires:

- At least one computer, which is usually a server that handles all the administrative duties for the System.
- A network of computers that is running a special grid computing network software.
- A middleware which is a collection of computer software.

The Architecture Of the optical character recognition system on a grid infrastructure consists of three main components.

1. Scanner
2. OCR Hardware or Software
3. Output Interface/device

7. DESIGN AND IMPLEMENTATION

7.1 Product Features

- The major features of Machine Learning based Optical character recognition (OCR) Scanner are as listed below.
- The product will help the user to convert their handwritten text document into machine-encoded text.
- This will help the user to maintain their document in an editable format, without worrying about its maintenance.
- As the users can convert their data in machine-encoded text, searching for any text or keyword will become easier.
- Data can be lost or accessed without authorization in paper format documents, which can be prevented using this system.
- Data recovery is not possible in paper format documents once they are destroyed. But if they are converted into computer text document recovery can be done if a proper backup is maintained.

7.2 Modules

1. Main Screen

The main screen module will display the document i.e., imported and will provide tools for editing purposes through the editor module and screen clipping tools.

2. File Manager

This module is responsible to import documents from the device into the main screen. It can also create a new document and save them after editing.

3. **Document**

The document module will contain the document, which is currently being operated on. This module will have an association with every other module.

4. **Editor**

All the editing to be done on the document on the main screen will be done using this module. This module will provide all the tools to edit a document according to the user's needs. It will contain all the functionalities of a general text editor.

5. OCR Processor

This module is the heart of the entire system. All the documents that are imported from the file manager into the main screen will be transported to this module for OCR operation.

8. TESTING AND RESULT

Testing is last and vital phase in the software development life cycle. We perform software testing to see whether the expected result of the system is equal to the result that we obtain from the system. Two important keywords in testing are verification and validation.

Verification is about are we building the project or a system right. It should work perfectly without any bug. E.g., if we want to add two numbers 2+2 should be equal to 4, it should not give you 5.

On the other hand, we have validation. It is more about what we are building should match the client's requirements. If the client wants 2+2=5 then the system should give 2+2=5.

When we are talking about various phases or levels of testing it can be stated as follows.

8.1 Unit Testing

It is the micro level of testing to make sure that each unit is working properly. A unit can be a specific piece of functionality; it can be a program or a particular procedure within the system. It plays a crucial role in verifying the internal design and logic.

8.2 Integration Testing

It is usually done immediately after unit testing. In this individual units are combined and are to test how they work as a group. It also identifies interface issues between modules.

8.3 System Testing

The next level of testing is system testing. As the name implies, all the components of the software are tested as a whole in order to ensure that the overall product is working properly as expected.

8.4 Acceptance Testing

The final level of testing is Acceptance Testing or UAT (User Acceptance testing). It determines whether or not the software is ready to be released. As the requirements keep on changing, in this level of testing the user ensures that all the requirements are met before the product is released.

8.5 Result

Test Case ID	1
Test Case Name	Recognize Character
Priority	1
Pre requisite	Import Document, OCR
Steps	Scan the input character and display it in machine encoded format
Expected Result	Output character should be equal to the input character

Test Case 1

Test Case ID	2
Test Case Name	Maintain Structure of the document
Priority	2
Pre requisite	Import Document, OCR
Steps	Scan the page size and locate the position of the character
Expected Result	The system should recognize the position of the character

Test Case 2

Test Case ID	3
Test Case Name	Editing text document
Priority	3
Pre requisite	Import Document, OCR
Steps	Enter space between two character, select all character and change their font
Expected Result	The format of the text should change accordingly.

Test Case 3

9. CONCLUSIONS

Machine Learning based Optical character recognition (OCR) Scanner is a combination of a word processor and OCR processor that is used to convert any type of paper-based document into a computerized document without altering the structure of the document.

The user is supposed to provide an image or scanned document of the data that is needed to be converted into computerized text, to the system. The system will take the input and extract the text from that input, then it will maintain the font or styling of the text if it is available else if the input is handwritten then the system will maintain the structure of the document.

This system will result in the efficient management of documents and any organization can step towards a paperless approach using this software.

REFERENCES

[1] Chaudhuri A., Mandaviya K., Badelia P., Ghosh S.K. (2017) Optical Character Recognition Systems. In: Optical Character Recognition Systems for Different Languages with Soft Computing. Studies in Fuzziness and Soft Computing, vol 352. Springer, Cham. https://doi.org/10.1007/978-3-319-50252-6_2

[2] Kumar M., Jindal M.K., Sharma R.K. (2011) Review on OCR for Handwritten Indian Scripts Character Recognition. In: Nagamalai D., Renault E., Dhanuskodi M. (eds) Advances in Digital Image Processing and Information Technology. DPPR 2011. Communications in Computer and Information Science, vol 205. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-24055-3_28

[3] S. Mori, C. Y. Suen and K. Yamamoto, "Historical review of OCR research and development," in Proceedings of the IEEE, vol. 80, no. 7, pp. 1029-1058, July 1992, doi: 10.1109/5.156468.

[4] Vamvakas, G. & Gatos, B. & Stamatopoulos, Nikolaos & Perantonis, Stavros. (2008). A Complete Optical Character Recognition Methodology for Historical Documents. Document Analysis Systems, IAPR International Workshop on. 525-532. 10.1109/DAS.2008.73.

[5] Nagy, G., Nartker, T. A., & Rice, S. V. (1999). . *Document Recognition and Retrieval VII.* doi:10.1117/12.373511