



iJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 Issue: IV Month of publication: April 2023

DOI: <https://doi.org/10.22214/ijraset.2023.50090>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

Utilizing OCR to Retrieve Text from Identity Documents

Dr. D. Arulanantham¹, S. Sneha², K. Logeshwaran³, R. Nishanth⁴, K. Lavanya⁵

¹Associate Professor, ^{2, 3, 4, 5}UG Students, Department of ECE, Nandha Engineering College, Erode 638 052, Tamil Nadu, India.

Abstract: Nowadays, automatic information extraction can offer a big boost in efficiency, accuracy, and speed in all those business processes where data capture from documents plays an important role. Feeding enterprise information systems with data coming from documents often requires manual data entry that is a long, tedious, and error-prone job. Therefore, automating data entry can save a lot of time and speedup the execution of business processes allowing employees to focus on core and more valuable aspects of their daily activities. Organizations in all industries may require to process myriads of documents with a variety of formats and contents. Such documents are often digitized as images and then converted to text format. This work addresses the text recognition task for identity documents. In particular, identity documents are paramount in business processes related to customer subscription and on boarding where users frequently submit photos taken by smartphones or poor-quality scanned images that are blurred, unclear and with very complex framing angles can be bring off by this process. The text recognition method present in this work is based on transfer learning and Scene Text Recognition (STR) networks. In this project, user can explore how to extract text from a image. In addition, a method to convert multiple images into editable string format with in single click.

Index Terms: Introduction, Literature Survey, Existing & Proposed Method, Software & Design Flow, Feasibility Study, Modules, Algorithm, Problem Definition & System Testing, Experimental Result.

I. INTRODUCTION

A. Introduction

Optical Character Recognition (OCR) is a process that allows a system to recognize the script or characters written in a customer's conversation without human intervention. Optical recognition of men and women has become one of the fastest growing intelligent applications of pattern detection and artificial intelligence. Explore the different OCR methods in our survey. In this article, we describe a hypothesis and numerical system for visual identification of males and females. Individual OCR (Optical Class Recognition) and MCR (Individual Magnetic Reputation) techniques are commonly used to identify patterns or scripts. In standard, the alphabet can be handwritten or stamped as an extension of the pixel pix, and can be of any series, shape, etc. Instead, in OCR, the alphabet is printed in magnetic ink and learners sort the alphabet. According to the unique magnetic field formed by each alphabet. OCR and Optimized OCR are available for banks and selected exchanges. Previous studies of optical sensing, or people's prestige, have shown that there are no obstacles to handwriting methods. Familiarity with handwriting is difficult due to the diversity of human handwriting and differences in angle, size, and shape of calligraphy. A set of strategies for visually identifying a male or female during performance is invoked here. The reputation of being an optical male or female is one of the most glamorous and shocking areas of modeling reputation, with many practical applications. The rejection case demonstrates that the understanding of OCR has been built by many researchers over a long period of time, forming an excellent global network of candid human studies. In these nuanced conversations, people redoubled their "confrontational and collaborative" research efforts. Therefore, global workshops and initiatives for local improvement are planned. For example, global induction on the limits of handwriting detection and global conversations about article psychoanalysis and popularity ratings serve as explanatory parts in the realm of knowledge and confidentiality.

B. Introduction to OCR

Optical Character Recognition (OCR) is also known as text recognition. OCR programs extract and reuse data from scanned documents, camera images, and ID cards. OCR software extracts characters from images, combines them into words, and then combines words into sentences so you can access and edit the original content. It also eliminates the need for manual data entry. OCR systems use a combination of hardware and software to convert physically printed documents into machine-readable text.

Hardware such as optical scanners or special printed circuit boards copy or read the text. The software then usually performs advanced processing. OCR software can use artificial intelligence (AI) to implement advanced ICR (Intelligent Character Recognition) techniques such as language or handwriting style recognition. The OCR process is most commonly used to convert paper legal or historical documents into PDF documents so that users can edit, format, and search the documents as if they were created in a word processor.

II. LITERATURE SURVEY

A. *An Automatic Reader of Identity Documents*

Automated document identity scanning and confirmation is an attractive creation of today's business sector, and considering that this activity still requires manual use and a lot of manual work, it is essential to lose money and time. In this mode, individual automated document identity analysis machines are distributed. The machine is believed to be able to extract the identity details of the first Italian document from an image of suitable quality, as is often required for virtual subscribers to many services.

Records are initially limited to snapshots, then categorized and finally recognized by text. Synthetic data sets are used for training neural networks and general evaluation of overall machine performance. Synthetic datasets avoid the privacy concerns associated with using real snapshots of real documents, instead allowing them to be used for future machine development. In common cases, Internet subscription contracts (mobile phone service, bank, etc.) require characters to be sent via digitized ID records. One of its contract validity scenarios is that the data in the record is the same as the data provided in the resource used by the subscriber in the subscription form, and the record is not always out of date.

Validation of digitized datasets against expected scenarios is mandatory and must be performed by Human Resources. It's a very time-consuming and stupid job, which makes it very expensive and particularly prone to human error. The problem is companies that have to manage most of their contracts, companies that have to manage a large number of subscribers (who are not always inclined to use new technologies these days) and governments. Important for 2 physical activities (ex. candidate type). In the future, emerging resources using new solutions for highly private identities will surely overcome this problem, or at least change it significantly. But human review of large numbers of scanned documents will pave the way, at least for a few years.

B. *Scene Text Recognition Model Comparisons Dataset and Model Analysis*

In recent years, many new proposals for scene text (STR) content recognition schemes have been added. While each statement pushes the boundaries of technology, the discipline largely lacks holistic, real-world comparisons due to conflicting choices of schooling and assessment data sets. This article addresses this problem through 3 important contributions. First, inconsistencies in school and assessment data sets, and the impact of inconsistencies on overall performance gaps. Second, a unified four-degree STR framework is introduced in which the maximum current STR mode is justified. Use of this framework allows substantial evaluation of previously proposed STR modules and enables the invention of previously unexplored module combinations. Third, on a regular set of school and assessment datasets, we examine the module's contribution to accuracy, speed, and overall performance in question sentence recall. This type of analysis eliminates the dilemma of contemporary comparisons to identify the overall performance advantages of current modules. Our code is public. Reading text in herbal scenes, known as scene text recognition (STR), has become an important task in various commercial applications.

The maturity of the Optical Character Recognition (OCR) framework has made it a popular tool for scanning documents cleanly, but most traditional OCR techniques are not as robust on STR bonds due to the different text appearances that occur in actual text appearances. Internationalization and the Imperfect Case of These Scenarios Despite incredible advances in new scene textual content recognition (STR) models, they have been compared to inconsistent benchmarks, primarily due to the difficulty of determining whether and how the base STR model of the proposed module. These plots analyze the contribution of previously stalled current STR patterns under inconsistent test settings. To achieve this, in addition to the regular datasets, a common localization framework is complemented by key STR techniques: seven benchmark assessment datasets and education datasets (MJ and ST).

C. *Character Region Awareness for Text Detection*

NAVER Corp Scene textual content detection strategies entirely based on neural networks have recently emerged with promising results. Previous strategies using inflexible sentence degree delimiters show boundaries by representing text positions in arbitrary forms. This paper advocates a novel scene text detection technique to efficiently find text locations by exploring the affinity between each individual and the characters.

To overcome the lack of character individuality annotations, our proposed framework exploits each given annotation at the character level of artificial pixels, and the expected ground truth based on individuality for real pixels obtained over the time of invention. During intermediate rendering. To estimate the affinity between characters, the community masters the proposed new affinity specification. Extensive experiments on six benchmarks, including the Total Text and CTW-1500 datasets, which include particularly curved text in plant-based pixels, show that our individual-level text detection significantly outperforms detectors of point. Based on the results, our proposed technique provides excessive flexibility in detecting text content pixels in complex scenes, including arbitrarily oriented, bent, or distorted text.

Most strategies encounter textual content in terms of sentences, but defining the scope of sentences to detect is not trivial because sentences can be distinguished by a variety of criteria, including their meaning, size, or color. Also, the boundary of sentence segmentation cannot be strictly defined, so the sentence itself does not have the grand semantics to which it refers. This ambiguity in the sentence annotations dilutes the significance of the floor fact for each regression and segmentation method. Our primary goal is to accurately locate each individual character within the herbal pixel. Learning deep neural communities to predict affinities between various regions and characters. Rendering is efficient with poor supervision as no public datasets of individual degrees are available.

III. EXISTING & PROPOSED SYSTEM

A. Existing System

This system uses optical character recognition technology to extract text from image files, as shown in Figure 3.1. No optimization techniques have been implemented to increase the level of accuracy of text predictions. Here it only applies to any font style provided in the training set. Pixel-based comparisons have not been developed. As a result, the accuracy of predicting text from image files decreases slightly.

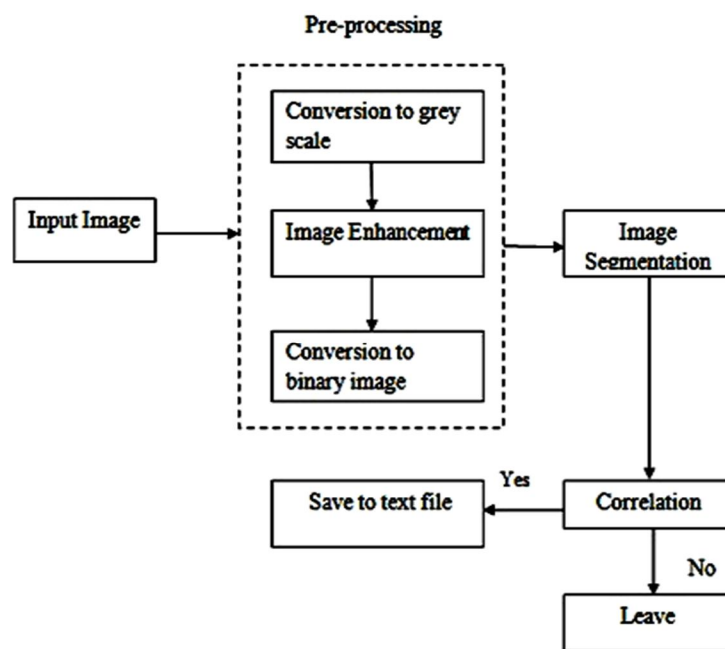


Figure 3.1 Block Diagram Of Existing System

Drawbacks Of Existing System

- 1) It will take longer if the image size is larger.
- 2) No optimization technique has been developed
- 3) Text recognition accuracy is poor.
- 4) Text is only recognized if it exactly matches the images in the training set.
- 5) Pixel-based procedures are not implemented.
- 6) Delay in processing HD images.

B. Proposed System

The proposed system implements an optimized optical character recognition technique to efficiently extract text from image files. This optimization technique increases the accuracy level of text recognition compared to existing systems. Implement pixel-based comparisons to process high-definition images without providing output. As a result, the accuracy of predicting text from image files is significantly improved over previous systems. The proposed system design flow is shown in Fig. 3.2.

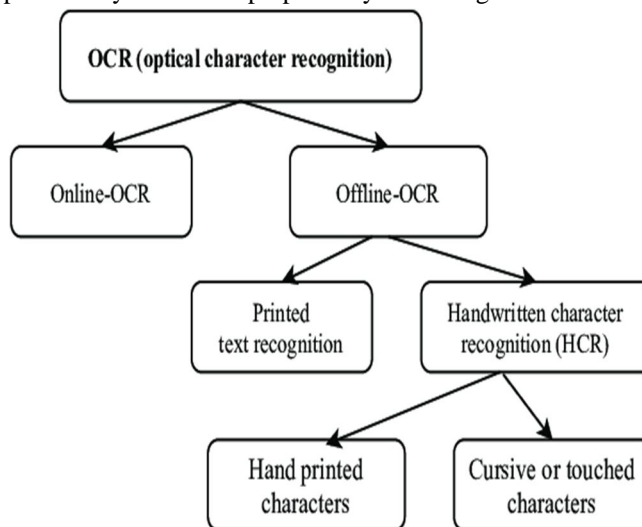


Figure 3.2 Design Flow Of Proposed System

Advantage Of Proposed System

- 1) Takes less time even if the image size is large.
- 2) Improved OCR technology is under development.
- 3) Text recognition is more accurate.
- 4) Recognize the text even if it is not present in the images of the learning game.
- 5) A pixel-based process has been implemented here.
- 6) The lag when processing HD images is ignored here.

IV. SOFTWARE & DESIGN FLOW

A. Python

Python is a dynamic, overly hierarchical and loose open source and interpreted programming language. In addition to procedural oriented programming, it also facilitates object-oriented programming. In Python, you don't need the variable form because it's a dynamically typed language.

1) Features of Python

Python has many features, some of which are described below,

- a) Ease of coding.
- b) Free and open source.
- c) Object-oriented language.
- d) Support for GUI programming.
- e) High Level Languages.
- f) Extensible Features.
- g) Python is a Portable Language.
- h) Python is an Embedded Language.
- i) Interpreted Languages.
- j) Large standard library.
- k) Dynamically typed language.

2) *Python Applications*

a) *Web And Internet Development*

Python makes web application development easy. It includes internet protocol libraries such as HTML and XML, JSON, email management, FTP, IMAP and an easy-to-use socket interface. But there are more libraries in the package index.

- Requests - HTTP client library.
- BeautifulSoup - HTML parser.
- Feed Parser - Parses RSS/Atom feeds.
- Paramiko - implements the SSH2 protocol.
- Twisted for asynchronous network programming.

There also has a variety of frameworks available. Some of them are Django, Pyramid. He also gets micro frames like cans and bottles. This was also covered in the Introduction to Python Programming article. You can also write CGI scripts and use advanced content management systems such as Plone and Django CMS.

b) *Application of Python Programming in Desktop GUI*

Most Python binary distributions come with Tk, a standard graphics library. It allows you to define the user interface of your application.

Additionally, some toolkits are available:

- WxWidgets.
- Kivy – for writing multitouch applications.
- Qt via pyqt or pyside.

Then platform-specific toolkits are available:

- GTK+.
- Microsoft Foundation Extensions for win32
- Delphi

c) *Science And Numeric Applications*

It is one of the most common applications of Python programming. With its powerful capabilities, it's no surprise that Python has a place in the scientific community.

They are:

- SciPy– A collection of math, science and engineering packages.
- Pandas - Library for data analysis and modeling.
- IPython - A powerful shell to facilitate editing and recording of work sessions.
- It also supports visualization and parallel computing. Software Carpentry Course – teaches basic scientific computing skills and hosts a boot camp. It also offers free access to educational materials.
- In addition, NumPy allows us to handle complex numerical calculations.

d) *Software Development Application*

Software developers use Python as a supporting language. They use it for build control and management, testing and more:

- SCons – for build control
- Buildbot, Apache Gump – for automated and continuous compilation and testing.
- Roundup, Trac – for project management and debugging.
- IDE grid.

e) *Python Application In Education*

Simplicity, simplicity and large community make Python a great beginner programming language. Python programming is widely used in education because it is a great language to teach in schools and can be learned on your own. Read about the pros and cons of Python if you haven't already. Also learn about Python functions.

f) Python Application In Business

Python is also suitable for developing ERP systems and e-commerce systems.

- Tryton is a three-tier, advanced general-purpose application platform.
- Odoo - management software with a suite of business applications. When used as a versatile tool, you get a complete business management application.

g) Database Access

With Python you can:

- User interface and ODBC interface for MySQL, Oracle, PostgreSQL, MS SQL Server, etc.; All of this is free to download.
- Object databases such as Durus and ZODB.
- Standard database APIs.

B. Design Flow of Text Extraction

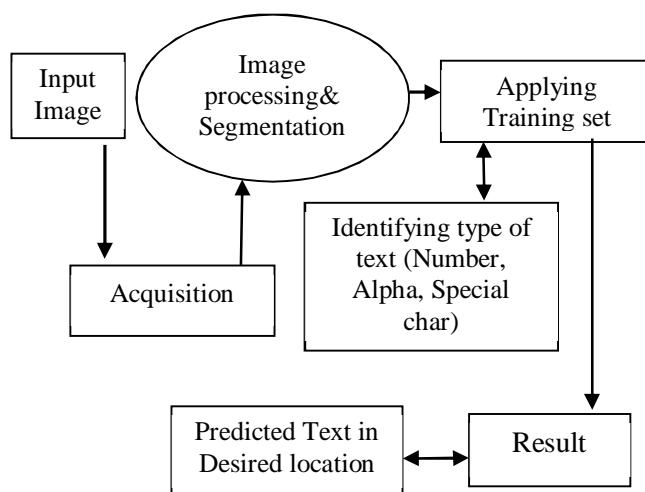


Fig Iv.2.1 Block Diagram Of Text Extraction

This process is defined as the act of extracting an image from any source (usually hardware such as a camera sensor). Acquire receives an input image.

Upon completion of this process, image processing and segmentation is performed (see Figure IV.2.1). In this process, image quality is improved, noise is removed, images are segmented into symbols corresponding to the training set, and symbols are assigned to the results. Output is saved to the specified location.

V. FEASIBILITY STUDY

A feasibility study considers all the analyzes necessary for project development. Each structure needs to be user-friendly to the end user, so think through it when developing your project. You need to know the type of information being collected, and systems analysis consists of collecting, organizing, and evaluating facts about a system and its environment.

Three considerations involved in feasibility analysis are

A. Economic Feasibility

The organization must purchase a personal computer with a keyboard and mouse, which is a direct expense. Superimposing a manual system on top of a computerized system has many immediate benefits. Users get answers when they ask questions. The rationale for any investment is to lower costs or improve the user experience. Users with basic knowledge of Microsoft technologies can use these services by visiting the services provided on the website.

B. Operational Feasibility

We propose a system approach process to solve problems in the existing system. Your organization's day-to-day operations can be applied to this system. Essentially, operational feasibility should include an analysis of how the proposed system will affect organizational structures and procedures. The system proposed in requires less human-computer interaction, and anyone with a basic computing device can access the service and handle part of the service required to get the service the user wants.

C. Technical Feasibility

A cost-benefit analysis leads to the conclusion that computerized systems are beneficial in today's fast-paced world. Technical feasibility assessment should be based on the overall design of inputs, outputs, documents, procedures and system requirements for procedures. This project aims to use web applications to share information faster and reduce the hassle of handling requests from website users. The current system is designed to overcome the challenges of the old system. The current system is designed to reduce technical requirements to make apps accessible to more users.

VI. MODULES

A. Image Acquisition

In image acquisition, the pattern pictures are accumulated, which might be required to educate the classifier set of rules and assemble the classifier model. Images have been taken at unique angles, showing the unique environmental and lights as shown in the figure 6.1. The preferred JPG and PNG layout are used to shop those pics. In this study, pictures had been accumulated from farms in unique font types.

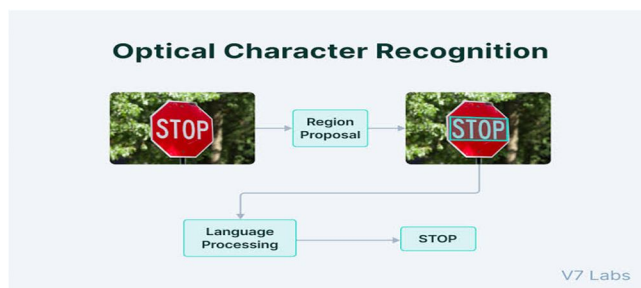


Figure 6.1 Image Acquisition.

B. Image Preprocessing

Following the Image acquisition as shown in the figure 6.1, photo processing was performed to improve the image quality. All the original pics have been saved in one folder. The photos were named as we and can accept any price of numbers. Only horizontal photos were grown to become round through a manner of ninety levels and resized through the use of 200x300 pixels. Vertical images want to 200x300 pixels, and at the same time as the width and pinnacle of the image graph are equal, the ones pics were resized to 250x250 pixels. When the picture graph duration is surely too long, the processing mission takes extra time. After that, one of the noise discount techniques changed into used to eliminate the noise from snapshots and boom the sharpness of photos. Later, all the preprocessed photos have been stored in a folder.

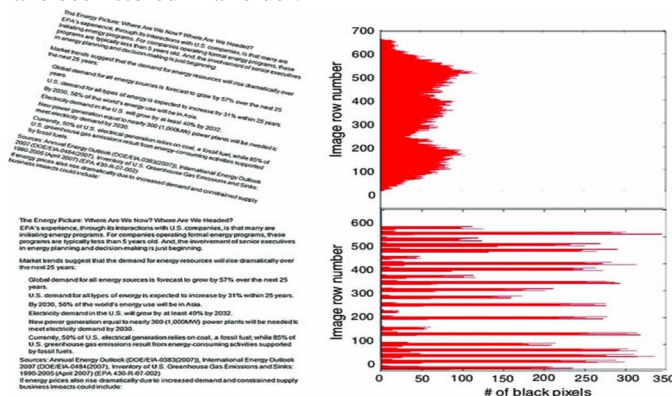


Figure 6.2 Image Preprocessing

C. Image Segmentation

The 1/3 segment of the technique is photo segmentation as shown in the figure 6.3. As a first step, all preprocessed images were converted to grayscale and saved in their original format. Due to the fact that identifying the appropriate shade version for preprocessing is one of the consequences of this research. After that, the image is transformed to a binary layout. These layout values had been clustered using the OCR method. In this step, an image segmentation and its set of rules has been performed.

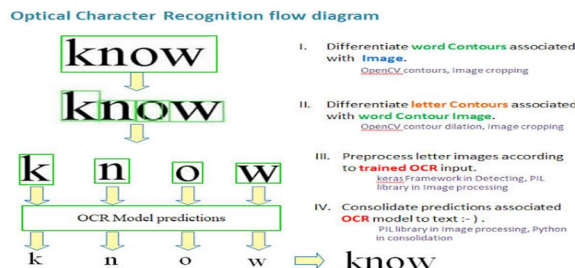


Figure 6.3 Image Segmentation

D. Applying Training Set

In this phase of the technique, it makes use of training set pics. The segmented output has been executed, which has been created using characteristic extraction. However, photo sets have been created to do experiments. The education of those picture units is mentioned right here. A field information guide was taken for the categorization of snap shots, and each picture had been decided from the categorized units of a picture randomly.

VII.ALGORITHM

A. Optical Character Recognition (OCR).

Optical Character Recognition (OCR) is a technology that recognizes text in images such as scanned documents and photos. Since taking pictures takes less time than taking notes, people are too lazy to take notes or type, so they have to take pictures of the text. Fortunately, modern smartphones allow users to apply OCR directly, allowing users to directly copy previously captured text without writing or retyping.

Python can do this with just a few lines of code. One of the frequently used OCR tools is Tesseract. Tesseract is an optical character recognition engine for various operating systems. Originally developed by HP as proprietary software. Later development was acquired by Google.

B. Installation

Tesseract currently works well on Windows, macOS and Linux platforms. Tesseract supports Unicode (UTF-8) and supports over 100 languages.

This article begins with the process of installing Tesseract OCR and testing text extraction from images. The first step is to install Tesseract. To use the Tesseract library, you must first install it on your system.If you are using Ubuntu you can simply install Tesseract OCR using apt-get.

```
python -mpip install tesseract-ocr (or)
```

```
pip install tesseract-ocr
```

For macOS users, use Homebrew to install Tesseract.

```
brew install tesseract
```

C. Implementation

After installation is completed, let's move forward by applying tesseract with python. First, import the dependencies.

```
from PIL import Image
```

```
import pytesseract
```

```
import numpy as np
```

D. Design of Input Window

Python's user input (see Figure 7.4.1) is written in the Python programming language. Developers often need to interact with users to get data or provide some kind of result. Most modern programs use dialog boxes to prompt the user for input. User input in Python is important when creating a system or project because Python's user input is how it learns about the progress of the system or project. I create a font called Python IDLE based programming to modify the PostScript font. PostScript is not just a "page description language", it is a fully fledged programming language.

In fact, the original PostScript fonts were programs that could be corrupted. You can open it in a text editor to edit the content and see what happens. So learn how to do it. As you grow, programming becomes more and more important. I want to progress and learn more, but I have no programming background and traditional programming languages like C or Pascal are relatively difficult to use. is to try something so you can focus on the problem and not the programming problem. is most closely related to my brother Guido, who often gives me programming advice.

Eventually he was introduced to the language he was working on called Python. He created Python as the scripting language for this experimental operating system and helped develop it in a research lab in the Netherlands. There was a good Mac version and he said "Why don't you give it a try? Let's see if what you want to experiment with is easier in Python because Python is a friendlier language." It's true that the syntax is much simpler than in the more common languages of the time, especially C. It's not a programming problem, it's just an attempt to do things in a way that focuses on the problem. Python has become the type design and dominant programming language in font development.

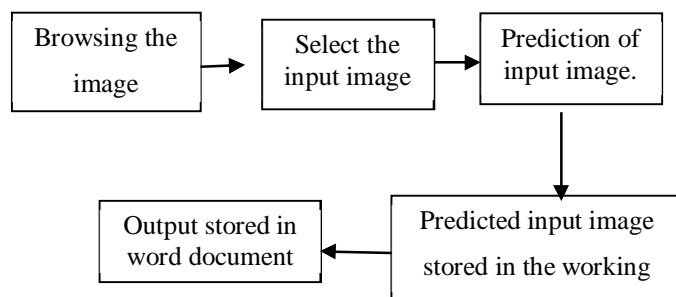


Figure 7.4.1 Browsing Input Image

E. Python in Type Design

Python really turns out to be the perfect tool for what you need and want to do. Erik van Blokland bothered him and he soon realized it. Eric's older brother Peter van Blokland also used him for bullying. In the late 90's, an application called RoboFog appeared. This application was a remix of Fontographer 3.5 with an added Python scripting layer.

It was originally for our own use, but over time we sold licenses to others. For example, Jonathan Hoefler is very passionate about this kind of work. He learned Python and built a very complex tool chain for his foundry. At some point, even the Adobe folks chose Python. They used Python more and more in their internal tools. As a result, the use of Python in the context of type design has expanded. Font Lab has also chosen Python as its scripting language of choice. Thus, almost by accident, Python became the dominant programming language for font design and development.



Flow Chart For Proposed Algorithm

VIII. PROBLEM DEFINITION & SYSTEM TESTING

A. Problem Definition

One of the hottest areas of industrial technology, optical letter tracking is now commonly used for projects of all sizes, from occasional document scanning to creating large digital document archives. However, it remains an active area of technical research and innovative engineering. The following major research advancements in the field of OCR are ranked recently. Flexible OCR seeks powerful use of images from various printed items by solving the following problems:

- Recognizing multiple letters.
- Recognizing multiple font styles.
- Recognizing multiple sizes.
- Recognizing digital documents.

B. System Testing

When the source code is ready, it is stored as a linked data structure. When finished, you should test and validate your project with subscriptions and definitions to find bugs. The project developer will underestimate, if you develop and run the test, the program will work and you won't find the error, unfortunately the error exists and if the project developer doesn't find the error, it will prove that the user can find the error.

Project developers are always responsible for testing individual modules. program module. In most cases, developers also do integration testing, which is the testing phase that builds the entire structure of the program.

This article has been tested to ensure accuracy.

- Unit tests.
- User acceptance tests.

- 1) *Unit Testing:* Unit tests first test the programs that make up the system. Therefore, unit testing is also called program testing. The software units of a system are modules and subprograms that are assembled and integrated to perform a specific function, and modules that are independent of each other are first checked for errors. This allows detection of code and logic errors contained only in the module. Testing is done at the stage of programming itself.
- 2) *User Acceptance Tests:* In this testing procedure, the project is handed over to the customer, tested against all requirements, and executed when the user is completely satisfied. Your project is completely ready. When users request changes and find errors, everything needs to be considered and corrected to make the project perfect.

C. Integration Testing

Integration testing is done to ensure that individual modules work together as a unit. Integration tests allow separate modules that require integration to be used for testing. As a result, the manual test data for testing the interface is replaced by automatically generated data from the various modules. This can be used to test how the module actually interacts with the proposed system. These modules are integrated and tested to identify problematic interfaces.

D. System Implementation

Upon completion of the initial system design, an acceptance design was negotiated with the customer to proceed with further system development.

After the system was developed, it was demonstrated in operation. The purpose of a system diagram is to identify malfunctions in a system.

The system implemented in the company is initially implemented in parallel with the existing manual system after approval by the system administrator. The system has been tested using real-time data and found to be error-free and user-friendly.

Implementation is the process of converting a new or modified system design into an active design after the initial system design has been completed. Information about the running system is provided to the end user. This procedure is used to test and identify logical clutter in a system by entering various combinations of test data. After the system is approved by end users and management, the system is implemented. System implementation involves many activities.

The six main activities are:

- 1) **Coding:** Coding is the process of translating physical design specifications created by a team of analysts into computer code that a team of programmers is working on. Coded here in python. Python is a computer programming language commonly used to create websites and software, automate tasks, and perform data analysis. Python is a general purpose language. That is, they can be used to create a variety of programs and are not designed to solve any specific problem.
- 2) **Testing:** Each software module can be tested as the encoding process starts and continues in parallel. This is where automated testing begins. Automated testing is the use of non-human scripts to execute a test plan (the parts of the application you want to test, the order you want to test them in, and the expected responses). Python already comes with a set of tools and libraries to help you create automated tests for your applications.
- 3) **Installation:** Installation is the process of replacing your current system with a new one. This includes transforming existing data, software, documents and operating procedures for the new system.
- 4) **Documentation:** This is the result of the installation process and the user manual provides information on how to use the system and its process. Extract text from a single image using Python. We'll start by extracting text from a single image using Python. This example uses the first image from the previous section, sampletext1-ocr.png. All images are placed in the images folder and code is placed in main.py. Image path: images/sampletext1-ocr.png.
- 5) **Training and Support:** A learning plan is a user training strategy that enables users to quickly learn a new system. Training provides knowledge and training in specific skills often required to complete a job, such as leadership training or meeting specific compliance standards. Think of training as the first step in ensuring your employees have the skills and knowledge they need to perform project functions. Training is therefore a relatively simple and straightforward solution. Training plan development can begin early in the project.

IX. SAMPLE SCREENS & EXPERIMENTAL RESULTS

A. Sample Screens

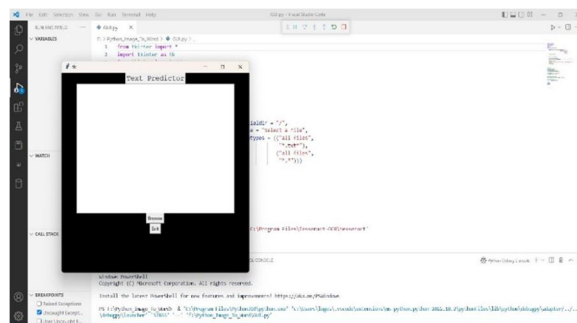


Figure 9.1.1 Browser Template Of Image

In this section, explore how to extract text from images using Python. All images are placed in the folder images and the code resides in main.py. One way of extracting text from every image would be to use the browse option to take the images from the folder as shown in the figure 9.1.1.

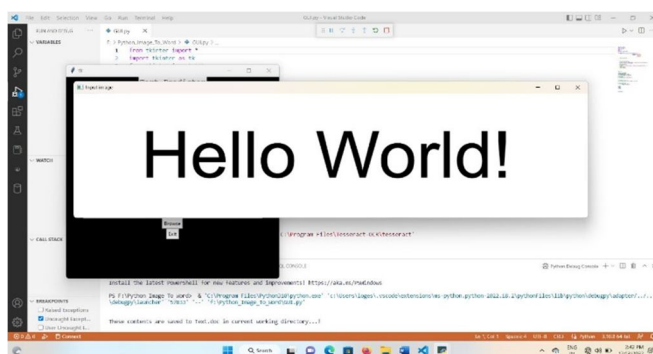


Figure 9.1.2 Select The Input Image.

After clicking the browse button the input image is given as shown in the figure 9.1.2.



Figure 9.1.3 Prediction Of Input Image.

You can capture text from a scanned image, upload your image file from your computer, or take a screenshot on your desktop. It scans the entire picture, extracts the readable text, and generates the accurate text report as shown in figure 9.1.3.

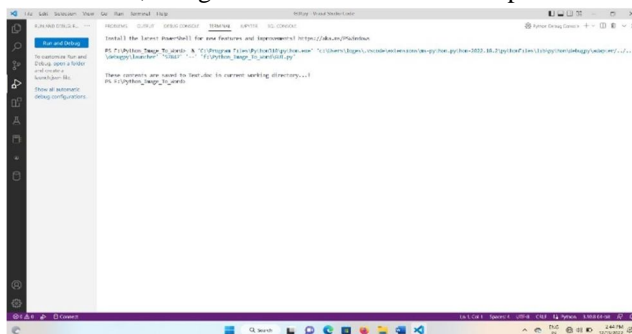


Figure 9.1.4 Predicted Input Image Stored In The Working Directory

Using the OS library, all of the file names in a given directory can access as shown in figure 9.1.4. Once get access to all of the file names in the images folder, iterate over them and extract text from each image using Python.

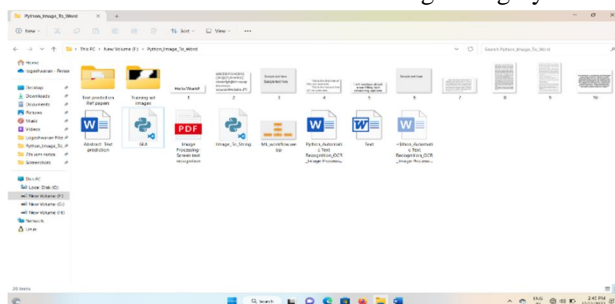


Figure 9.1.5 Output Stored In Word Document

The Extracted text from the image is stored in the word document as shown in the figure 9.1.5

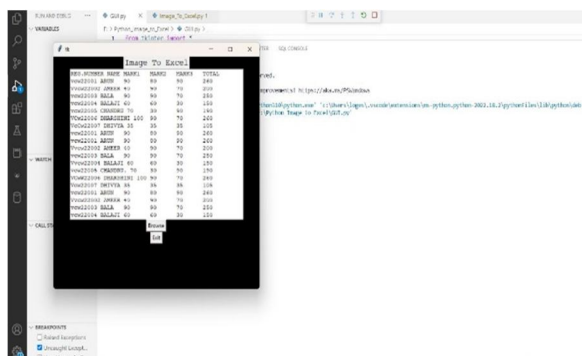


Figure 9.1.6 Selected Input Image For Excel Output

After clicking the browse button in figure 9.1.3 the input image for excel output is given as shown in the figure 9.1.6.

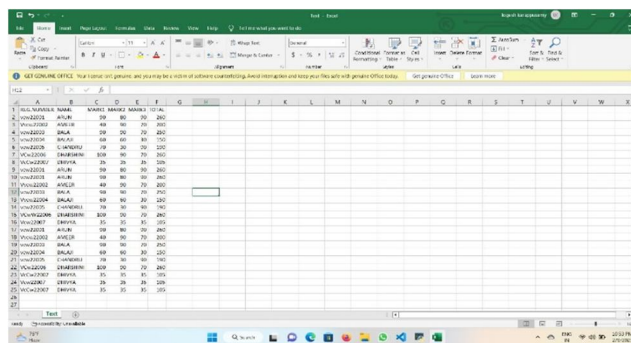


Figure 9.2.2 Output Of The Sample Screen For Excel

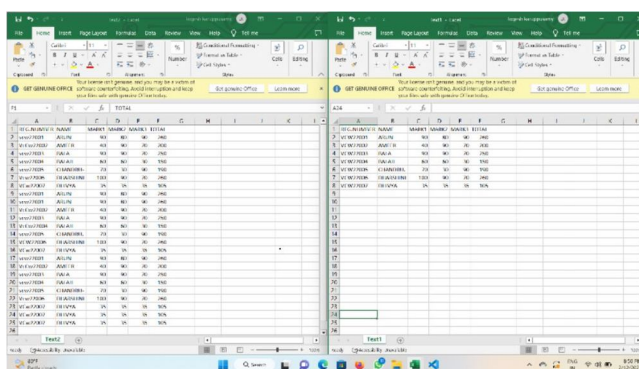


Figure 9.2.3 Excel Output Of The Multiple Image

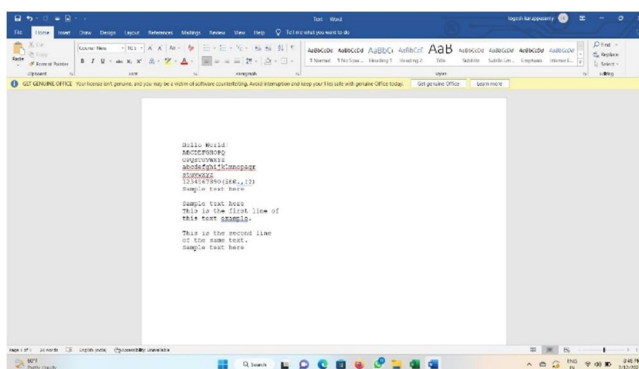


Figure 9.2.4 Word Output Of The Multiple Image

X. CONCLUSION

Our project give an assessment of the performance of the optical alphabet exam. In this study, the theoretical and mathematical versions of the maximum hard trouble within the scope of optical alphabet identity, which is changed through scale, translate, and rotation in optical alphabet detection are analyzed. The possible deployments of the OCR techniques are also studied. The precision and identity aren't good enough for practical deployment. It can also additionally require a good-sized improvement.

REFERENCES

- [1] Sushruth shastri, gunasheela g, thejus dutt, vinay d s and sudhir rao rupanagudi, "i" - a novel algorithm for optical alphabet recognition (ocr). 978-1-4673-5090-7/13/\$31.00 ©2013 ieee.
- [2] Lulu zhang, xingmin shi, yingjie xia, kuang mao, "a multi-filter based license plate localization and recognition framework". 978-1-4673-4714-3/13/\$31.00 ©2013 ieee.
- [3] Ibrahim el khatib, yousef samir-mohamad omar, and ali al ghouwayel, "an efficient algorithm for automatic recognition of the lebanese car license plate. isbn: 978-1-4799-5680-7/15/\$31.00 ©2015 ieee.
- [4] Jieun kim, and ho-sub yoon "graph matching method for alphabet recognition in natural scene images. 978-1-4244-8956-5/11/\$26.00 ©2011 ieee

- [5] Feng yanga, and fan yangb, “alphabet recognition using parallel bp neural network”. 978-1-4244-1724-7/08/\$25.00©2008ieee. the international journal of computational science, information technology and control engineering (ijcsitce) vol.4, no.1, january 2017 14
- [6] Rókus arnold, and póth miklós “alphabet recognition using neural networks. 11th ieee international symposium on computational intelligence and informatics • 18–20 november, 2010 • budapest, hungary.
- [7] Amarjot singh , ketan bacchuwar , akash choubey, and devinder kumar , “an omr based automatic music player”. 978-1-61284-840-2/11/\$26.00 ©2011 ieee.
- [8] Shan du, member, ieee, mahmoud ibrahim, mohamed shehata, senior member, ieee, and wael badawy, senior member, ieee “automatic license plate recognition (alpr) : a state-of the art review ,ieee transactions on circuits and systems for video technology, vol. 23, no. 2, february 2013.
- [9] Imran shafiq ahmad, boubakeur boufama, pejman habashi, william anderson and tarik elamsy, “automatic license plate recognition: A comparative study”. 2015 ieee international symposium on signal processing and information technology (isspit).
- [10] Rejean plamondon, fellow, ieee, and sargur n. srihari, fellow, ieee, “on-line and off-line handwriting recognition:a comprehensive survey1ieee transactions on pattern analysis and machine intelligence. vol. 22, no. 1. january 2000.
- [12] Hadar i. avi-itzhak, thanh a. diep, and harry garland, “high accuracy optical alphabet recognition using neural networks with centroid dithering ieee transactions on pattern analysis and machine intelligence, vol. 17, no. 2, february 1995.
- [13] Shalin a. chopra, amit a. ghadge, onkar a. padwa, karan s. punjabi, prof. gandhali s. gurjar, “optical alphabet recognition”, international journal of advanced research in computer and communication engineering vol. 3, issue 1, january 2014.
- [14] C. nelson kennedy babu, member ieee and krishnan nallaperumal, senior member ieee, “a license plate localization using morphology and recognition”, 978-1-4244-2746-8/08/\$25.00 ©2008 ieee.
- [15] S. n. nawaz, m. sarfraz, a. zidouri, and w. g. ai-khatib, “an approach to offline arabic alphabet recognition using neural networks”, 0-7803-8163-7/03/\$17.00 © 2003 ieee.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)