



(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

# **OPTICAL CHARACTER RECOGNITION**

A Report for the Evaluation 3 of Project 2

*Submitted by*

**ANIRUDH KUMAR**

(1613101132/ 16SCSE101533)

*in partial fulfillment for the award of the degree of*

**Bachelor of Technology**

**IN**

Computer Science and Engineering

**SCHOOL OF COMPUTING SCIENCE AND ENGINEERING**

**Under the Supervision of**

**Mr. B. Mallikarjuna, Professor**

**APRIL / MAY - 2020**



**SCHOOL OF COMPUTING AND SCIENCE AND  
ENGINEERING**

**BONAFIDE CERTIFICATE**

Certified that this project report “**OPTICAL CHARACTER RECOGNITION**”  
is the bonafide work of “**ANIRUDH KUMAR(1613101132)**” who carried out  
the project work under my supervision.

**Signature of the Head**

Dr. MUNISH SHABARWAL  
Professor & Dean,  
**School of Computing Science  
& Engineering**

**Signature of the Supervisor**

Mr.B.Mallikarjuna, Professor,  
**School of Computing Science  
& Engineering**

# TABLE OF CONTENTS

1. Abstract
2. Introduction
  - (i) Overall description
  - (ii) Purpose
  - (iii) Motivations and scope
3. Literature survey
4. Problem statement
5. Proposed model
6. References

## **Abstract:**

Optical character i.e. the character in the form of images, are really hard to recognized by the system. The system must need to recognized the text in the form of string or character for the processing of it. The recognizing of text from an image is made easier by the OCR(Optical character reconition).OCR do the recognition of text, it extract the text from an image and mapped it with the text having in its dataset . and then produced the mapped result in the form of string or we can say in the form of characters.

## Introduction:

- **Overall Description:**

The project is all about reading a Mathematical expression from the the image as a text and provide solution of it. Reading of text from an image is done by optical character recognition (ocr) and then the text is converted into expression and then the result is generated.

The large amount of documents, either modern or historical, that we have in our possession nowadays, due to the expansion of digital libraries, has pointed out the need for reliable and accurate systems for processing them. Historical documents are of more importance because they are a significant part of our cultural heritage. During the last decades a lot of research has been done in the field of Optical Character Recognition (OCR). Numerous commercial products have been released that convert digitized documents into text files, usually in ASCII format. Although these products process machine printed documents successfully, when it comes to handwritten documents the results are not satisfactory enough. Moreover, such products are unable to process historical documents due to their low quality, lack of standard alphabets and presence of unknown fonts.To this end, recognition of historical documents is one of the most challenging tasks in OCR.

In the literature, historical document processing is mainly focused on document retrieval. Word-spotting techniques for searching and indexing historical documents have been introduced. In [1], word images are grouped into clusters of similar words by using image matching to find similarity. Then, by annotating “interesting” clusters, an index that links words to the locations where they occur can be built automatically. In [2] and [3] holistic word recognition approaches for historical documents are presented based on scalar and profile-based features and on matching word contours respectively. Their goal is to produce reasonable recognition accuracies which enable performing retrieval of handwritten pages from a user-supplied ASCII query. In [4], a word spotting technique based on combing synthetic data and user feed-back for keyword searching in historical printed documents is described.

we have used the python libraries like: openCV ,numpy , pytesseract.

1. openCV is use to read the image and convert it into grayscale image .
2. Numpy covert the properties of image into an array.
3. Pytesseract recognize the text from the image using the OCR.
4. Tkinter use to give GUI effect for selecting of image.

Coding is done in following pahses:-

1. Reading of image from the file where it stored.
2. Converting of image into grayscale image.
3. Dilation and Erosion are applied to remove noise from image.
4. Threshold is applied to get image into black and white.
5. Recognition of text with tesseract.

6. Adding spaces to the after each token of expression.
7. Evaluating of expression.
8. Displaying of result.

- **Purpose:**

The purpose of the project is to use OCR(Optical Character Recognition) in recognizing of expression. So that we can easily able to convert expression from handwritten or typewritten images into machine encoded text and find solution of the difficult mathematical problem.

- **Motivation And Scope:**

- \* Can be used in solving Mathematical expression by just providing image of it.
- \* Use to solve large expression.
- \* Use to solve expression containing variable by just providing image of it.

**Literature Survey:**

This project is taken from the idea of OCR(optical Character Reconization).OCR is used to read text from an image. OCR is optical or mechanical conversion of typed,handwritten or printed text into machine encoded text.

In this project we used to extend the feature of OCR to read Mathematical expression from an image.so that we can easily able to solve written equation on a system.

Intially this project is just reading the simple mathematical expression and giving of it.Then after we extend its feature in to read and solve large mathmaticalexpression.Now, the code is also able to recognize variable from an expression and after giving the value of the variable it is able to solve it.

**Proposed Model:**

The idea behind this project is to extending the feature of OCR(optical character recognition) to solve mathematical expression.

In this project we are solving handwritten, typewritten, printed mathematical expression using OCR.

We have used python programming and there libraries for Implementing this project.

Various python libraries used:

1. OpenCv :-used in processing of image.
2. NumPy :-used in converting of image into NumPy array.
3. Pytesseract:-used to extract expression from an image.
4. Tkinter:-used to implement GUI effect for selection of image.

### **Implementation:-**

```
#read image with opencv
img=cv2.imread(img_path)

#convert to gray
gray=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

#apply dilation and erosion to remove some noise
kernal=np.ones((3,3),np.uint8)
gray=cv2.dilate(gray,kernal,iterations=1)
gray=cv2.erode(gray,kernal,iterations=2)

#write image after removed noise
cv2.imwrite(src_path+ "removed_noise.png",gray)

#apply threshold to get image with only black and white
_, gray=cv2.threshold(gray,127,255,cv2.THRESH_BINARY)

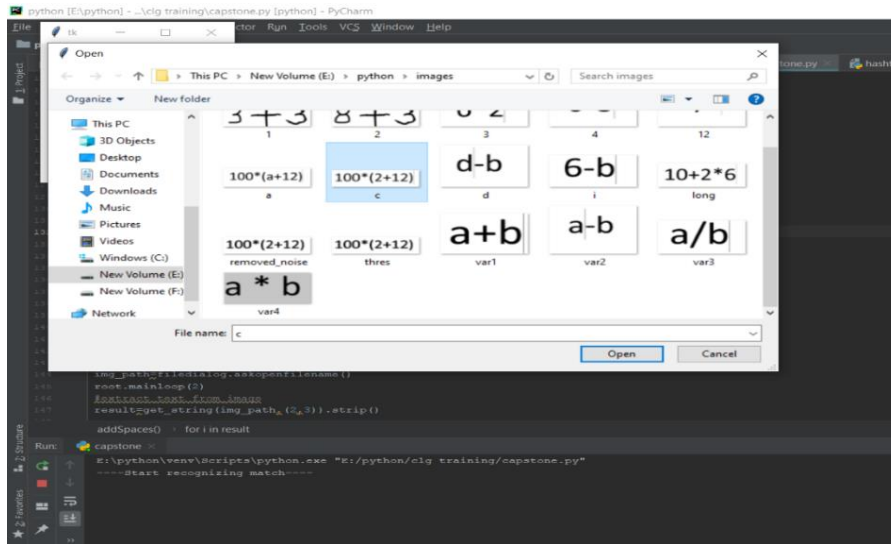
#write the image after apply opencv
cv2.imwrite(src_path+ "thres.png",gray)

#recognize text with tesseract for python
result=pytesseract.image_to_string(Image.open(src_path+"thres.png"))
```

The complete program is written in python. We have used it's some of its libraries also. In coding we have followed the implementation steps. From taking images from the location, conversion of it into grayscale, reading of expression to displaying of the result.

### **Result:-**





```
E:\python\venv\Scripts\python.exe "E:/python/clg training/capstone.py"  
---Start recognizing match---  
text:100*(2+12)  
text result:1400  
  
Process finished with exit code 0
```

**Conclusion:-**

1. Graphical user interface is used in the selection of different type of expressions.
- 2.OCR used to read text from image.
- 3.Able to Small, large and variable type mathematical expression.

**References:**

1. <https://youtu.be/83vFL6d57OI>
2. <https://www.geeksforgeeks.org/stack-set-4-evaluation-postfix-expression/>
3. <https://www.youtube.com/watch?v=H71ts4XxWYU&t=244s>