**PROJECT REPORT**

**on**

**Symbol Recognition using MATLAB**

**(CSE V Semester Mini project)**

**2020-2021**

****

**Submitted to: Submitted by:**

Mr. Chandradeep Bhatt Mr. Samar Maithani

(CC-CSE-B-V-Sem) Roll. No.: 1918647

**Guided by:** CSE-B-V-Sem

Ms. Aastha Gour Session: 2020-2021

(Resource Person)

**DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY**

**GRAPHIC ERA HILL UNVERSITY, DEHRADUN**

**CERTIFICATE**

Certified that Mr. Samar Maithani (Roll No.- 1918647) has developed mini project on “Symbol Recognition using MATLAB” for the CS V Semester Mini Project in Graphic Era Hill University, Dehradun. The project carried out by Student is their own work as best of their knowledge.

Date: 16/12/2021

(Mr. Chandradeep Bhatt) (Ms. Aastha Gour)

**Class Co-ordinator Project Guide**

**CSE-B-V-Sem Resource Person**

(CSE Department) (CSE Department)

GEHU Dehradun GEHU Dehradun

**ACKNOWLEDGMENT**

I would like to express our gratitude to The Almighty God, the most Beneficent and the most Merciful, for helping in making of the project.

I wish to thank our parents for their continuing support and encouragement. We also wish to thank them for providing us with the opportunity to reach this far in our studies.

I would like to thank particularly our project Co-ordinator Mr. Chandradeep Bhatt and our Project Guide Ms. Aastha Gour for her patience, support and encouragement throughout the completion of this project.

At last, but not the least I greatly indebted to all other persons who directly or indirectly helped us during this work.

**Mr. Samar Maithani**

**Roll No.- 1918647**

**CSE-B-V-Sem**

**Session: 2020-2021**

**GEHU, Dehradun**

**CONTENTS**

1. **INTRODUCTION 1**

1.1 About the Project 1

1.2 What is MATLAB 1

1.3 Uses of MATLAB 1

1.4 Image and Image Processing 2

1.5 Methodology 3

1. **PROJECT 4**

2.1 Requirement of the Project 4

2.1.1 Hardware Requirements 4

2.1.2 Software Requirements 4

2.2 Working Plan 4

2.2.1 Pre-Processing 4

2.2.2 Segmentation 4

2.2.3 Feature Extraction 5

2.2.4 Classification Process 5

1. **SCREENSHOT OF THE PROJECT 7**

3.1 MATLAB Software Interface 7

3.2 Some Code 7

3.3 OUTPUT 8

1. **CONCLUSION 9**

4.1 SUMMERY 9

4.2 FUTURE WORKS 9

**APPENDIX 10 REFERENCES 13**

1. **INTRODUCTION**
   1. **About the Project**

As the name suggests, The Symbol Recognition using MATLAB application is an

application that is designed in order to recognize the symbols in the images. This

application can allow in automating the process of recognizing the symbols in the

images with great ease. It will reduce considerably the difficulties faced on

existing system, with minimum error and difficulties. The work of the users can be

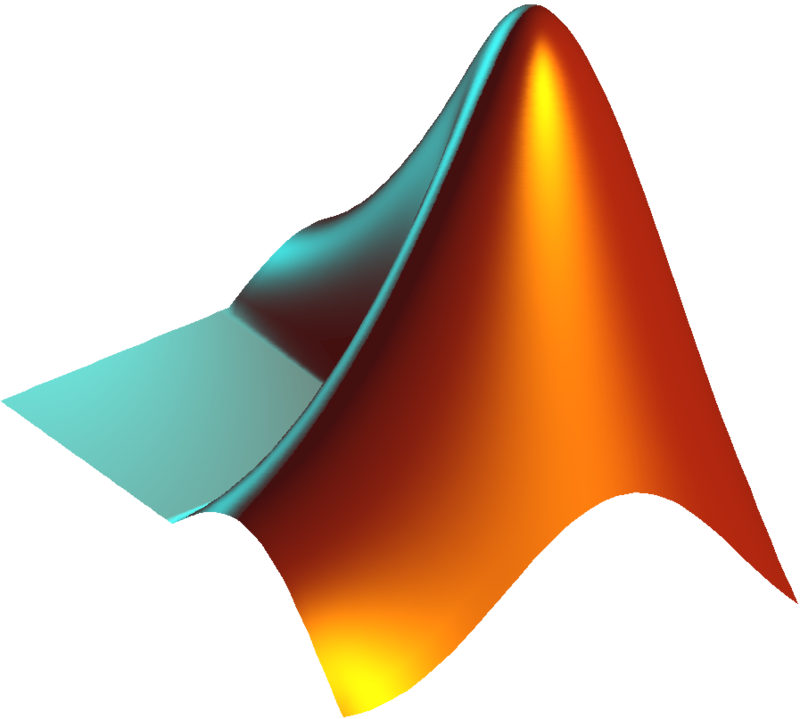
reduced with great ease through the use of this application. People can rely on

this application with great ease. This application is implemented using the

MATLAB. This application will help in recognizing the symbols in the images

without any difficulty with great ease.

* 1. **What is MATLAB?**

MATLAB is a programming platform designed specifically for engineers and scientists to analyse and design systems and products that transform our world. The heart of MATLAB is the MATLAB language, a matrix-based language allowing the most natural expression of computational mathematics.

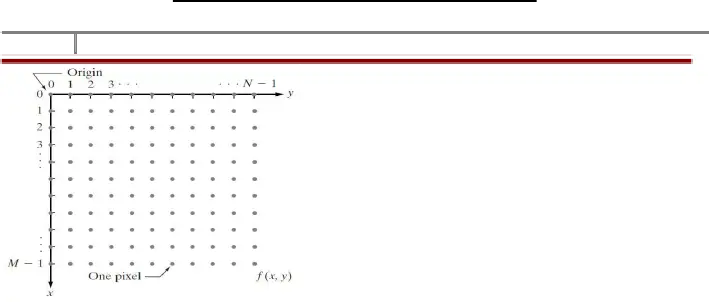
The name MATLAB stands for matrix laboratory. MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK projects, which together represent the state-of-the-art in software for matrix computation.

* 1. **Uses of MATLAB**
* Math and computation

MATLAB logo

* Algorithm development
* Modeling, simulation, and prototyping
* Data analysis, exploration, and visualization
* Scientific and engineering graphics
* Application development, including Graphical User Interface building
  1. **Image and Image Processing**

Image is a two-dimensional function *f(x, y)*, where x and y are spatial coordinates and the amplitude f at any pair of coordinates *(x, y)* is called the intensity or gray level. When x, y, and f are discrete quantities the image is digital. ‘f’ can be a vector and can represent a color image, e.g., using the RGB model, or in general a multispectral image. The digital image can be represented in coordinate convention with M rows and N columns as in Figure 1.1. In general, the gray-level of pixels in an image is represented by a matrix with 8-bit integer values.



Co-ordinate convention is used to represent an image

Digital image processing is the use of a digital computer to process digital images through an algorithm. As a subcategory or field of digital signal processing, digital image processing has many advantages over analog image processing. It allows a much wider range of algorithms to be applied to the input data and can avoid problems such as the build-up of noise and distortion during processing. Since images are defined over two dimensions (perhaps more) digital image processing may be modeled in the form of multidimensional systems. The generation and development of digital image processing are mainly affected by three factors: first, the development of computers; second, the development of mathematics (especially the creation and improvement of discrete mathematics theory); third, the demand for a wide range of applications in environment, agriculture, military, industry, and medical science has increased.

* 1. **Methodology**
* User will upload symbol. It will be in RGB format.
* RGB image is converted into grayscale image.
* Gray scale image is converted into black and white image.
* We had applied some image pre-processing steps to remove some unwanted objects and environmental interference.
* Symbol templates will be stored in directory.
* Dataset of these templates will be created.
* User will provide query image system will resize the query image.
* System will finally compare the query image values and template image values in dataset and will display the result in text format.

1. **PROJECT**
   1. **Requirement Analysis**
      1. Hardware Requirement:

* Any regular laptop/PC
* **RAM**: Minimum of 4GB, Recommended 8GB
* **Disk Space:** Minimum: 3.4 GB for MATLAB only, 5-8 GB for a typical installation. An SSD is recommended. A full installation of all MathWorks products may take up to 30 GB of disk space
* **CPU:** Minimum: Any Intel or AMD x86-64 processor

Recommended: Any Intel or AMD x86-64 processor with four logical cores and AVX2 instruction set support

* No specific Graphics Card is required. Hardware accelerated graphics card supporting OpenGL 3.3 with 1GB GPU memory is recommended.
  + 1. Software Requirement:
* Operating System: Windows 11, Windows 10, Windows 7
* IDE: MATLAB R2021b
  1. **Working Plan**
     1. Pre-processing:

The pre-processing is a series of operations or steps performed on the scanned input image. It basically enhances the image rendering it suitable for segmentation. The various tasks performed on the image in pre-processing stage are Binarization, noise removal, morphological operation. e.g., Binarization process converts a Gray scale image into a binary image using global thresholding technique

* + 1. Segmentation:

It is an operation that decomposes an image of sequence of characters into sub images of individual symbols. After accepting the document, the document image is subjected to pre-processing for background noise elimination and skew correction to generate the bit map image of the text. The pre-processed image is then segmented into lines, words and characters or symbols.

The purpose of the segmentation is to extract each character from the text present in the image. The process of segmentation for the proposed work mainly follows the following pattern:

1. First, it identifies the page layout, identifies the line from the page, identifies the word from that line, and finally, identifies the character from that word.
2. Starts from the first pixel still it finds the continuous black pixel from left to right direction.
3. After that if white pixel found then this indicates the one character or symbol from the image.
4. Apply bounding box on that symbol or character image.
   * 1. Feature Extraction:

After pre-processing on the image of text, features of character or symbols are extracted. This step is heart of the system. This step helps to classify the characters based on their features. Feature extraction is the name given to a group of procedures for measuring the relevant shape information contained in a pattern so that the task of classifying the pattern is made easy by a formal procedure. The feature extraction stage analyses a text segment and selects a set of features that can be used to exclusively identify the text segment. The issue of choosing the features to be extracted should be guided by the following concerns:

1. The features should carry sufficient information about the image and should not necessitate any domain-specific knowledge for their extraction.
2. They should be easy to calculate in order for the approach to be feasible for a large image collection and rapid retrieval.
3. They should be related well with the human perceptual characteristics because users will finally decide the correctness of the retrieved images
   * 1. Classification Process:

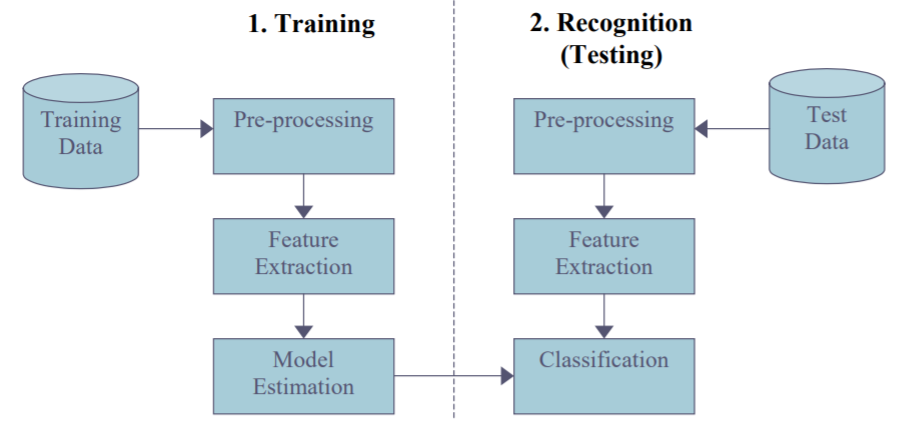
Classification is used as the decision-making stage of recognition system. It uses the features extracted in the previous stage to identify the text segment according to present rules. Many types of classifiers are applicable to OCR like K-Nearest Neighbour, Hidden Markov Model, SVM etc There are two steps in building a classifier: training and testing. These steps can be broken down further into sub-steps.

Training:

1. *Pre-processing* – Processes the data so it is in a suitable form for further processing.
2. *Feature extraction* – Reduce the amount of data by extracting relevant information—Usually results in a vector of scalar values. (We also need to NORMALIZE the features for distance measurements!)
3. *Model Estimation* – from the finite set of feature vectors, need to estimate a model (usually statistical) for each class of the training data

Testing:

1. *Pre-processing* –Process the data
2. *Feature extraction* – Extract the feature of the data
3. *Classification* – Compare feature vectors to the various models and find the closest match. One can use a distance measure.



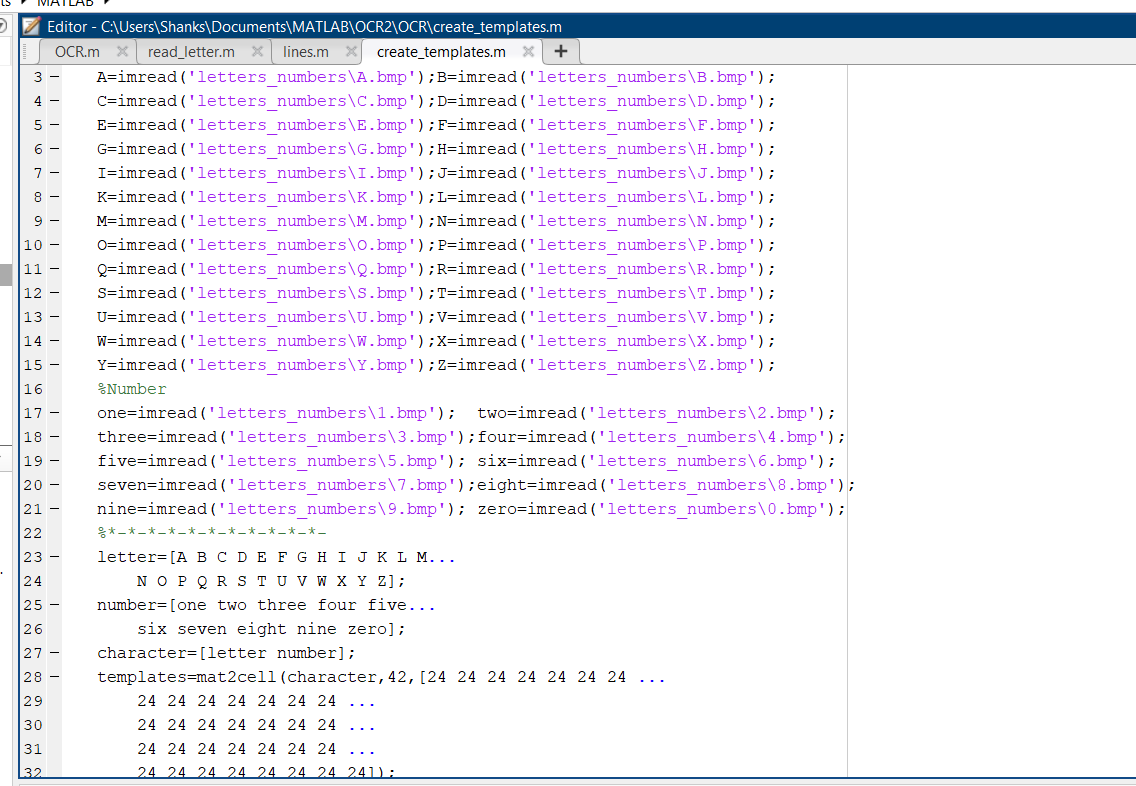
Classification Process

1. **SCREENSHOTS OF THE PROJECT**
   1. **MATLAB Software Interface**

**Graphical user interface, text, application

Description automatically generated**

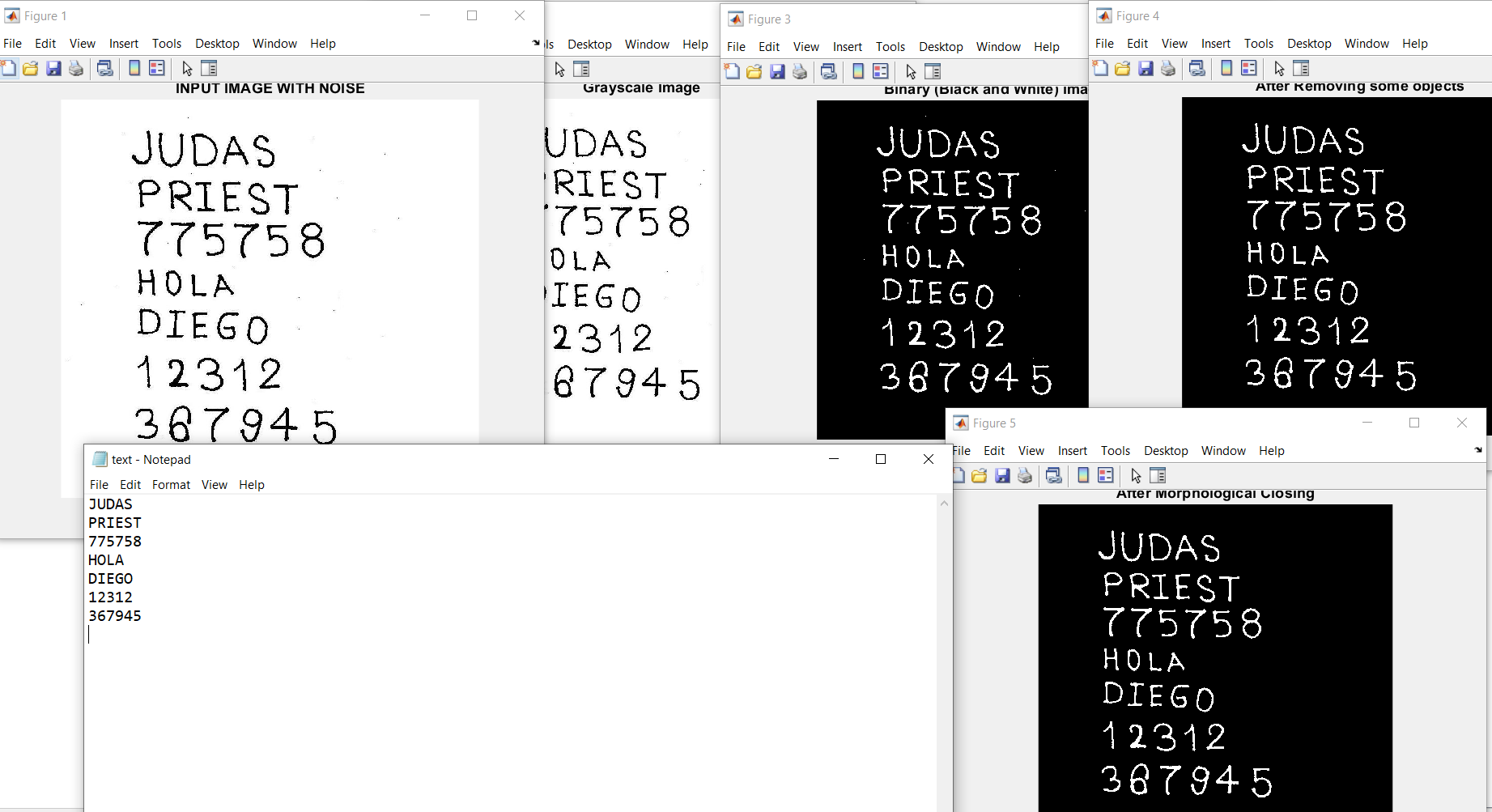
* 1. **Some Code**

****

* 1. **OUTPUT**

**Graphical user interface, application, Word

Description automatically generated**

****

1. **Conclusion**
   1. **SUMMERY**

We here proposed a system where symbol is recognized by the system, user will input symbol images system will apply algorithm to identify the symbol. Here in this system, we applied some image processing steps in order to work with images. We converted the RGB image into gray scale image. Image is converted into gray scale image to apply further image processing steps. Then gray scale image into black and white image, this is done due to accuracy. We had applied some image preprocessing steps in order to remove some unwanted objects and environmental interference. Here we store symbol templates in one directory each image will be of fixed size so that it would be easier to recognize correct symbol. These template images will be in black and white form. System will create dataset of these templates. User will provide query image system will resize the query image. System will finally compare the query image values and template image values in dataset and will display the result in text format. System will take image as an input and output the result in text format. In order to recognize the symbol, we had used the concept of Optical Character Recognition. System will provide result with 60%-80% accuracy.

* 1. **FUTURE WORK**

The system is not perfect it only able to identify English alphabets and numeric and that to not with 100% accuracy, there is still much scope available for improvement, like:

* Adding more character/ symbol dataset to train the system.
* Adding other languages characters rather than only English.
* Implementing some UI designs for better use.

**APPENDIX**

**OCR**

% Read image

imagen=imread('TEST\_1.jpg');

% Show image

figure; imshow(imagen);

title('INPUT IMAGE WITH NOISE')

% Convert to gray scale

if size(imagen,3)==3 %RGB image

imagen=rgb2gray(imagen);

end

figure; imshow(imagen); title('Grayscale Image');

% Convert to BW

threshold = graythresh(imagen);

imagen =~imbinarize(imagen,threshold);

figure; imshow(imagen); title('Binary (Black and White) Image');

% Remove all object containing fewer than 30 pixels

imagen = bwareaopen(imagen,30);

figure; imshow(imagen); title('After Removing some objects');

g=strel('disk',2);

imagen = imclose(imagen,g);

figure; imshow(imagen); title('After Morphological Closing');

%Storage matrix word from image

word=[ ];

re=imagen;

%Opens text.txt as file for write

fid = fopen('text.txt', 'wt');

% Load templates

load templates

global templates

% Compute the number of letters in template file

num\_letras=size(templates,2);

while 1

%Fcn 'lines' separate lines in text

[fl re]=lines(re);

imgn=fl;

% Label and count connected components

[L Ne] = bwlabel(imgn);

for n=1:Ne

[r,c] = find(L==n);

% Extract letter

n1=imgn(min(r):max(r),min(c):max(c));

% Resize letter (same size of template)

img\_r=imresize(n1,[42 24]);

% Call fcn to convert image to text

letter=read\_letter(img\_r,num\_letras);

% Letter concatenation

word=[word letter];

end

%fprintf(fid,'%s\n',lower(word));%Write 'word' in text file (lower)

fprintf(fid,'%s\n',word);%Write 'word' in text file (upper)

% Clear 'word' variable

word=[ ];

%\*When the sentences finish, breaks the loop

if isempty(re) %See variable 're' in Fcn 'lines'

break

end

end

fclose(fid);

%Open 'text.txt' file

winopen('text.txt')

clear all

**REFERENCE**

1. Download MATLAB from <https://in.mathworks.com/help/install/ug/download-without-installing.html>
2. Tutorial reference MATLAB programming language for Digital Image Processing from<https://www.youtube.com/playlist?list=PL6wr_B29b3UQnO73LSVCJI2Nkf7tNRFMw>
3. Reference about Digital Image Processing from <https://en.wikipedia.org/wiki/Digital_image_processing>
4. Reference regarding MATLAB Requirements: <https://in.mathworks.com/support/requirements/matlab-system-requirements.html>
5. Reference regarding the project from <https://nevonprojects.com/symbol-recognition-using-matlab/>