SPL-1 Project Report,2019

Handwritten Digit Recognition(HDR)

Course Code : SE 305(Software Project Lab-1)

Submitted by:

Md Mehedi Hasan Sun
BSSE 1025

Supervised by:

Dr. B.M. Mainul Hossain
Associate Professor,Institute Of Information Technology
University of Dhaka

Institute Of Information Technology,
University of Dhaka
29-05-2019

Table of Contents:

1.Introduction

One of the very popular applications in computer vision is Handwritten Digits Recognition (HDR) in the field of character recognition. Digits like other universal symbols are widely used in technology, bank, OCR, analyzing of digits in engineering, postal service, numbers in plate recognition, etc. They are some of the famous applications on HDR . There are 10 classes corresponding to the handwritten digits from ‘0’ to ‘9’ which are very depend on the handwritten. The main difficulty in the handwritten digits recognition is different handwritten style which is a very personal behavior where there are a lot of models for numbers based on the angles, length of the segments, stress on some parts of numbers, size of the image etc.**(Figure 1.1)**

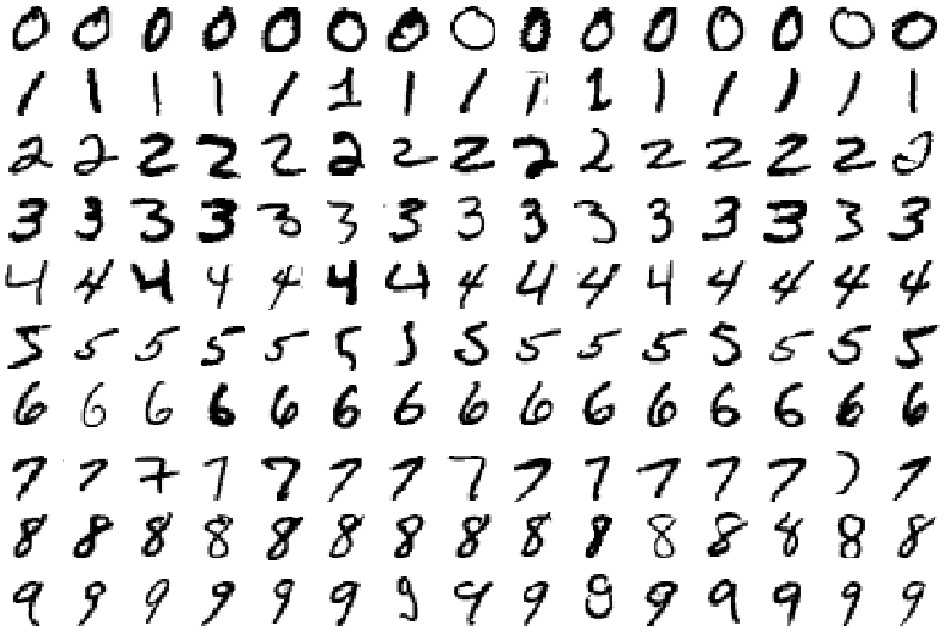


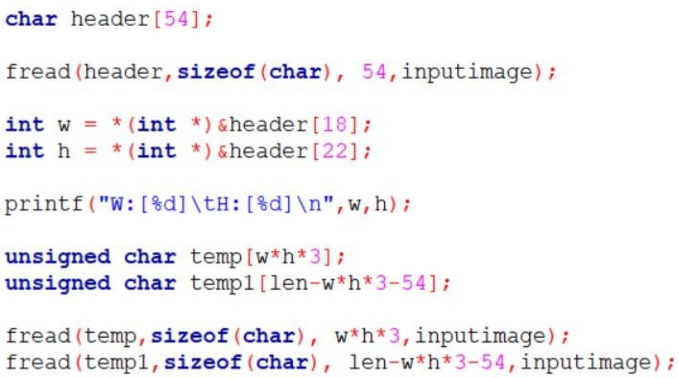
Figure 1.1: Different samples of handwritten digits in MNIST

However recognizing numbers is clear for human but it is not very easy for machines especially when there are some ambiguities on different classes . Recognizing digits is very important because it is related to the numbers thereby the recognition methods have to be very accurate. The HDR approach followed here is analysing pixels from training images then calculating the score between input image and template made by training dataset and finally highest scored digit is our chosen digit .

1.1. Background Study

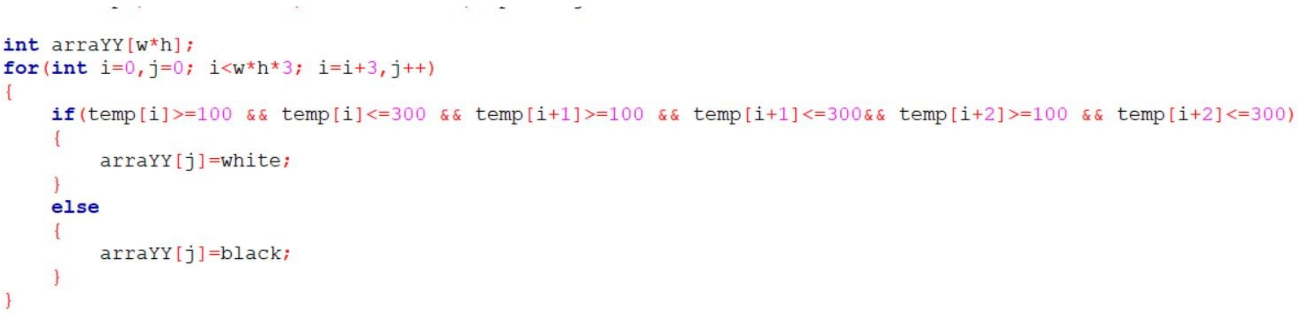
1.1.1.Image Processing

I had to learn how to process images in C . By processing ,I meant Image I/O(Figure 1.2),make corresponding Array from that image (Figure 1.3). As digits will be written in Binary Images ,i had to gather knowledge about binary image . I also had to make template from the training datasets.



```
char header[54];

fread(header,sizeof(char), 54,inputimage);

int w = *(int *)&header[18];
int h = *(int *)&header[22];

printf("W:[%d]\tH:[%d]\n",w,h);

unsigned char temp[w*h*3];
unsigned char temp1[len-w*h*3-54];

fread(temp,sizeof(char), w*h*3,inputimage);
fread(temp1,sizeof(char), len-w*h*3-54,inputimage);
```

Figure 1.2 :Image File I/O.



```
int arraYY[w*h];
for(int i=0,j=0; i<w*h*3; i=i+3,j++)
{
    if(temp[i]>=100 && temp[i]<=300 && temp[i+1]>=100 && temp[i+1]<=300&& temp[i+2]>=100 && temp[i+2]<=300)
    {
        arraYY[j]=white;
    }
    else
    {
        arraYY[j]=black;
    }
}
```

Figure 1.3 : Binary image Array.

1.1.2.BMP File Format

As our training dataset and input images are bmp image file so I had to learn about about bmp file format and needed to know how data are stored in a BMP images . BMP images can be divided into 4 segments.

- The File Header (14 bytes)

  Confirms that the file is (at least probably) a BMP file.
  Tells exactly how large the file is.
  Tells where the actual image data is located within the file.

- The Image Header (40 bytes in the versions of interest)
  Tells how large the image is (rows and columns).
  Tells what format option is used (bits per pixel).
  Tells which type of compression, if any, is used.
  Provides other details, all of which are seldom used.

- The Color Table (length varies and is not always present)
  Provides the color palette for bit depths of 8 or less.
  Provides the (optional) bit masks for bit depths of 16 and 32.
  Not used for 24-bit images.

- The Pixel Data
  Pixel by pixel color information
  Row-by-row, bottom to top.
  Rows start on double word (4-byte) boundaries and are null padded if necessary.
  Each row is column-by-column, left to right.
  In 24-bit images, color order is Red, Green and Blue.
  In images less than 8-bits, the higher order bits are the left-most pixels.

1.2.Challenges

- Encounter Bulk images
- Creating Binary Image Array
- Different angle of images
- Displacement of training images
- Template Storing
- Multiple digit in single image
- Images Can be larger or smaller.

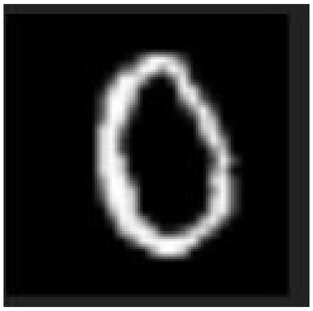
2.Project Overview

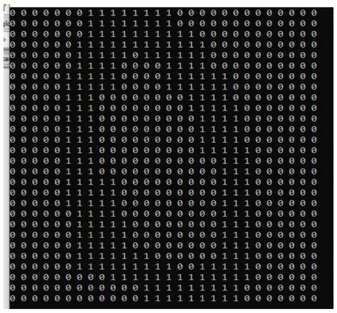
The whole project was devided into three different part.

1. Making template from MNIST dataset.
2. Operations on input.
3. Score calculate.

2.1.Making template from MNIST dataset

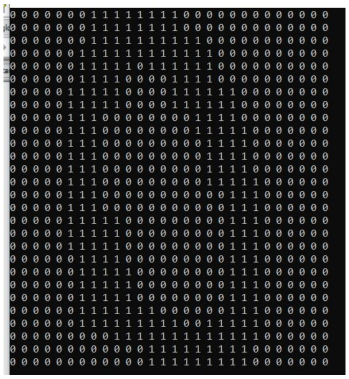
At first read an image and make a on/off array from it.





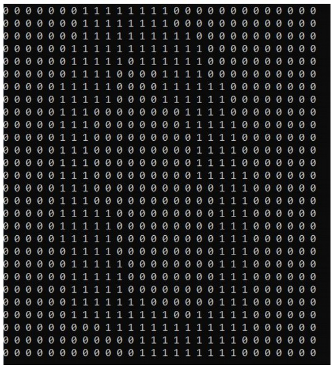
```
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
```

Then read another image and similarly create array and this array will be added with the previous one. And this process will go on until all images are read and stored. The final sum Array will be our template for corresponding digit.

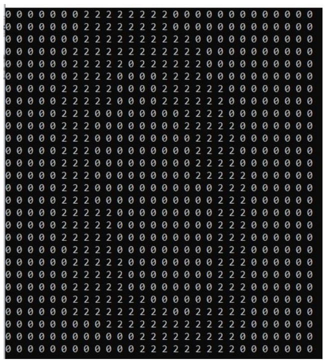


```
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
```

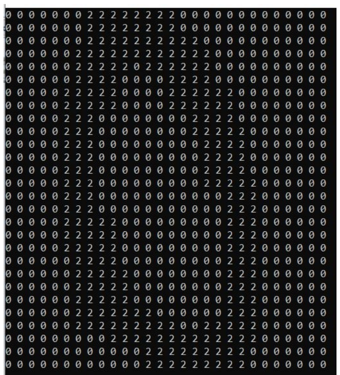
+



```
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
```

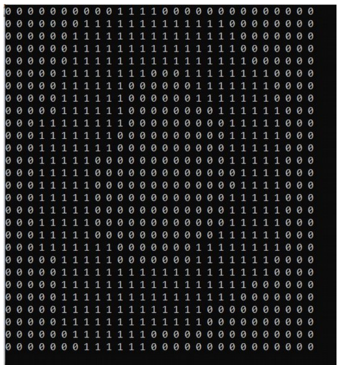


```
0 0 0 0 0 0 0 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 2 2 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 0 2 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 0 0 0 0 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 2 2 0 0 0 0 2 2 2 2 2 2 0 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 2 2 0 0 0 0 2 2 2 2 2 2 0 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 2 2 2 2 0 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 2 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 2 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 2 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 2 2 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 2 2 2 2 0 0 2 2 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 2 2 2 2 2 2 2 2 2 2 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 2 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 2 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0
```

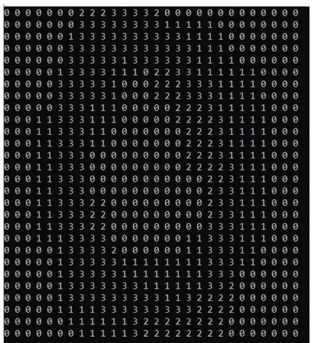


```
0 0 0 0 0 0 0 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 2 2 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 0 2 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 0 0 0 0 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 2 2 0 0 0 0 2 2 2 2 2 2 0 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 2 2 0 0 0 0 2 2 2 2 2 2 0 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 2 2 2 2 0 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 2 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 2 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 2 0 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 2 2 2 0 0 0 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 0 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 2 2 0 0 0 0 0 0 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 2 2 2 2 0 0 2 2 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 2 2 2 2 2 2 2 2 2 2 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 2 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 2 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0
```

+

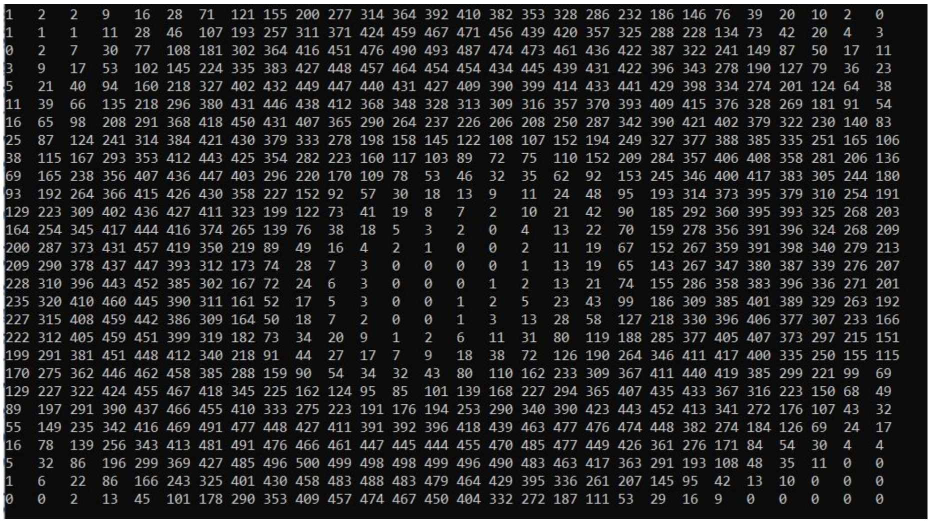


```
0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 1 1 1 0 0 0 0
0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0
0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0
0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0
0 0 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0
0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0
0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0
0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0
0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0
0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0
0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0
0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0
0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0
0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0
0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0
0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0
0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```



```
0 0 0 0 0 0 0 2 2 2 3 3 3 3 2 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 3 3 3 3 3 3 3 3 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 3 3 3 3 3 3 3 3 3 3 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 3 3 3 3 3 3 3 3 3 3 3 3 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 3 3 3 3 3 1 3 3 3 3 3 3 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 1 3 3 3 3 1 1 1 0 2 2 3 3 1 1 1 1 1 1 0 0 0 0
0 0 0 0 0 3 3 3 3 3 1 0 0 0 2 2 2 3 3 3 1 1 1 1 0 0 0 0
0 0 0 0 0 3 3 3 3 3 1 0 0 0 2 2 2 3 3 3 1 1 1 1 0 0 0 0
0 0 0 0 0 3 3 3 1 1 1 0 0 0 0 0 2 2 2 3 1 1 1 1 1 0 0 0
0 0 0 1 1 3 3 3 1 1 1 0 0 0 0 0 2 2 2 2 3 1 1 1 1 0 0 0
0 0 0 1 1 3 3 3 1 1 0 0 0 0 0 0 0 2 2 2 3 1 1 1 1 0 0 0
0 0 0 1 1 3 3 3 1 1 0 0 0 0 0 0 0 2 2 2 3 1 1 1 1 0 0 0
0 0 0 1 1 3 3 3 0 0 0 0 0 0 0 0 0 2 2 2 3 1 1 1 1 0 0 0
0 0 0 1 1 3 3 3 0 0 0 0 0 0 0 0 0 2 2 2 2 3 1 1 1 0 0 0
0 0 0 1 1 3 3 3 0 0 0 0 0 0 0 0 0 0 0 2 2 3 1 1 1 0 0 0
0 0 0 1 1 3 3 3 0 0 0 0 0 0 0 0 0 0 0 2 3 3 1 1 1 0 0 0
0 0 0 1 1 3 3 3 2 2 0 0 0 0 0 0 0 0 0 2 3 3 1 1 1 0 0 0
0 0 0 1 1 3 3 3 2 2 0 0 0 0 0 0 0 0 0 2 3 3 1 1 1 0 0 0
0 0 0 1 1 3 3 3 2 2 0 0 0 0 0 0 0 0 0 3 3 3 1 1 1 0 0 0
0 0 0 1 1 1 3 3 3 3 0 0 0 0 0 0 0 1 1 3 3 3 1 1 1 0 0 0
0 0 0 0 0 1 3 3 3 3 2 0 0 0 0 0 0 1 1 3 3 3 1 1 0 0 0 0
0 0 0 0 0 1 3 3 3 3 3 1 1 1 1 1 1 1 1 3 3 3 1 1 0 0 0 0
0 0 0 0 0 1 3 3 3 3 3 1 1 1 1 1 1 1 1 3 3 3 0 0 0 0 0 0
0 0 0 0 0 1 3 3 3 3 3 3 3 1 1 1 1 1 1 3 3 2 0 0 0 0 0 0
0 0 0 0 0 1 3 3 3 3 3 3 3 3 3 1 1 3 2 2 2 2 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 3 3 3 3 3 3 3 3 3 2 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 1 1 3 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 3 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0
```

Final Template Array for “0”:

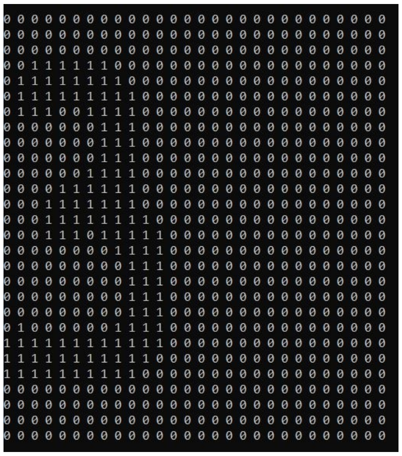


```
1   2   2   9   16  28  71  121 155 200 277 314 364 392 410 382 353 328 286 232 186 146 76  39  20  10  2   0
1   1   1   11  28  46  107 193 257 311 371 424 459 467 471 456 439 420 357 325 288 228 134 73  42  20  4   3
0   2   7   30  77  108 181 302 364 416 451 476 490 493 487 474 473 461 436 422 387 322 241 149 87  50  17  11
3   9   17  53  102 145 224 335 383 427 448 457 464 454 454 434 445 439 431 422 396 343 278 190 127 79  36  23
5   21  40  94  160 218 327 402 432 449 447 440 431 427 409 390 399 414 433 441 429 398 334 274 201 124 64  38
11  39  66  135 218 296 380 431 446 438 412 368 348 328 313 309 316 357 370 393 409 415 376 328 269 181 91  54
16  65  98  208 291 368 418 450 431 407 365 290 264 237 226 206 208 250 287 342 390 421 402 379 322 230 140 83
25  87  124 241 314 384 421 430 379 333 278 198 158 145 122 108 107 152 194 249 327 377 388 385 335 251 165 106
38  115 167 293 353 412 443 425 354 282 223 160 117 103 89  72  75  110 152 209 284 357 406 408 358 281 206 136
69  165 238 356 407 436 447 403 296 220 170 109 78  53  46  32  35  62  92  153 245 346 400 417 383 305 244 180
93  192 264 366 415 426 430 358 227 152 92  57  30  18  13  9   11  24  48  95  193 314 373 395 379 310 254 191
129 223 309 402 436 427 411 323 199 122 73  41  19  8   7   2   10  21  42  90  185 292 360 395 393 325 268 203
164 254 345 417 444 416 374 265 139 76  38  18  5   3   2   0   4   13  22  70  159 278 356 391 396 324 268 209
200 287 373 431 457 419 350 219 89  49  16  4   2   1   0   0   2   11  19  67  152 267 359 391 398 340 279 213
209 290 378 437 447 393 312 173 74  28  7   3   0   0   0   0   1   13  19  65  143 267 347 380 387 339 276 207
228 310 396 443 452 385 302 167 72  24  6   3   0   0   0   1   2   13  21  74  155 286 358 383 396 336 271 201
235 320 410 460 445 390 311 161 52  17  5   3   0   0   1   2   5   23  43  99  186 309 385 401 389 329 263 192
227 315 408 459 442 386 309 164 50  18  7   2   0   0   1   3   13  28  58  127 218 330 396 406 377 307 233 166
222 312 405 459 451 399 319 182 73  34  20  9   1   2   6   11  31  80  119 188 285 377 405 407 373 297 215 151
199 291 381 451 448 412 340 218 91  44  27  17  7   9   18  38  72  126 190 264 346 411 417 400 335 250 155 115
170 275 362 446 462 458 385 288 159 90  54  34  32  43  80  110 162 233 309 367 411 440 419 385 299 221 99  69
129 227 322 424 455 467 418 345 225 162 124 95  85  101 139 168 227 294 365 407 435 433 367 316 223 150 68  49
89  197 291 390 437 466 455 410 333 275 223 191 176 194 253 290 340 390 423 443 452 413 341 272 176 107 43  32
55  149 235 342 416 469 491 477 448 427 411 391 392 396 418 439 463 477 476 474 448 382 274 184 126 69  24  17
16  78  139 256 343 413 481 491 476 466 461 447 445 444 455 470 485 477 449 426 361 276 171 84  54  30  4   4
5   32  86  196 299 369 427 485 496 500 499 498 498 499 496 490 483 463 417 363 291 193 108 48  35  11  0   0
1   6   22  86  166 243 325 401 430 458 483 488 483 479 464 429 395 336 261 207 145 95  42  13  10  0   0   0
0   0   2   13  45  101 178 290 353 409 457 474 467 450 404 332 272 187 111 53  29  16  9   0   0   0   0   0
```

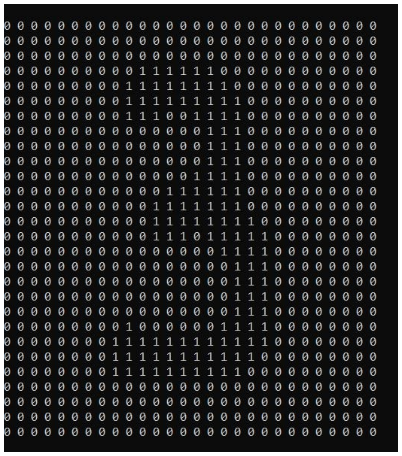
2.2. Operations on Input

Some operations are needed to get the perfect result from input image

- Image shifting

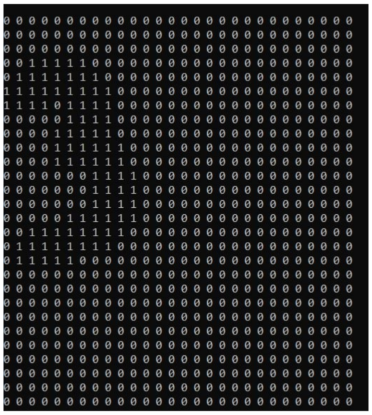


```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 1 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

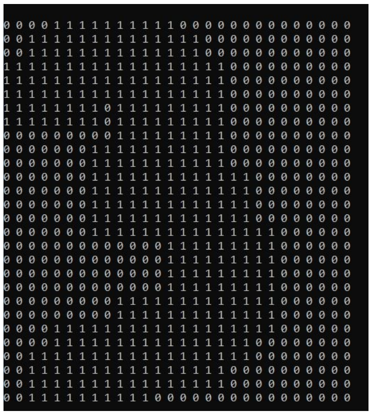


```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

- image Scaling



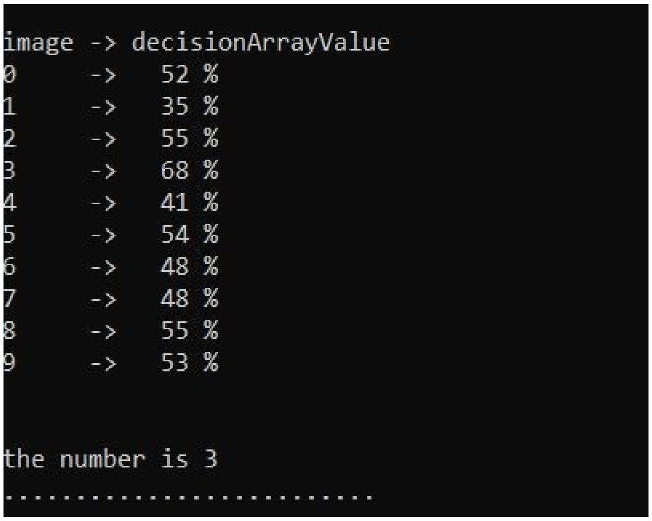
```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```



```
0 0 0 0 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

- Multiple digit cutting from single image

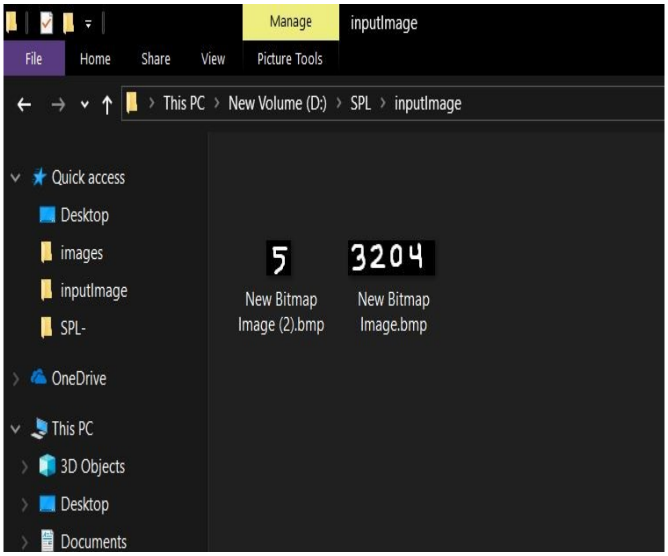
2.3.Score Calculate and detect digit

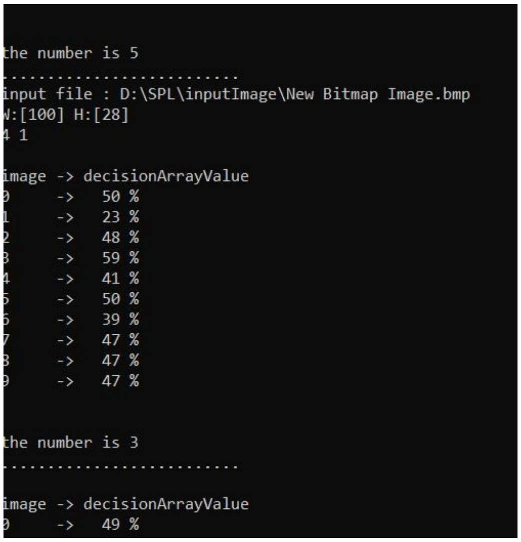


```
image -> decisionArrayValue
0     ->   52 %
1     ->   35 %
2     ->   55 %
3     ->   68 %
4     ->   41 %
5     ->   54 %
6     ->   48 %
7     ->   48 %
8     ->   55 %
9     ->   53 %


the number is 3
..........................
```

3.User Manual

User just have to put some binary image in the input folder. Handwritten digit will be there in those binary image. And the project will detect the digit .





```
the number is 5
.........................
input file : D:\SPL\inputImage\New Bitmap Image.bmp
W:[100] H:[28]
4 1

image -> decisionArrayValue
0     ->   50 %
1     ->   23 %
2     ->   48 %
3     ->   59 %
4     ->   41 %
5     ->   50 %
6     ->   39 %
7     ->   47 %
8     ->   47 %
9     ->   47 %


the number is 3
.........................

image -> decisionArrayValue
0     ->   49 %
```

4.Conclusion

The whole time this project gives me pleasure though I faced some challenges because I didn’t knew how machine understand handwritten digits,letters even there are thousands of patterns in handwrittining . Surely it will be a piller of my interest in Mechine Learning.

5.Reference

https://en.wikipedia.org/wiki/BMP_file_format(BMP file format)

https://www.researchgate.net/figure/9-Sample-digits-of-MNIST-handwritten-digit-database_fig1_264273647