

Available online @ <https://jjem.jnnce.ac.in>  
<https://www.doi.org/10.37314/JJEM.2020.040205>  
Indexed in International Scientific Indexing (ISI)  
Impact factor: 1.025 for 2018-19  
Published on: 30 March 2020

# Blue-Book Marks Recognition System

**Sandeep B**

J N N College of Engineering, Shivamogga

sandeepb@jnnce.ac.in

## **Abstract**

*The Blue-Book Marks Recognition System recognizes the marks obtained by the respective students in their Internal Assessment. The average marks are calculated and displayed. This system solves the real time problem where the faculty should enter the obtained marks and upload it into VTU database, which will be tedious one. The proposed system is implemented by taking some of the sample images, convert it into binary image and resize it. The region of interest is obtained from the resized image and the resulting image is then processed to box identification and box extraction. The digits are extracted from those boxes and are subsequently recognized. The recognized digits are displayed, and the average is calculated and displayed.*

**Keywords:** Neural Network, Segmentation, Image Processing

## **1. Introduction**

Visvesvaraya Technological University is a collegiate university in Karnataka. It is a mandate for the engineering colleges in the Karnataka to be affiliated with the university. Having VTU test patterns into consideration, three internal assessments are conducted. In those three, the best two internal assessment marks are considered. The final average is calculated accordingly. The internal assessment is done for twenty five mark (according to 2010 Scheme) and twenty marks (according to CBCS scheme) The marks obtained from the best of three assessments, they are averaged and added to the external marks.

The blue-book front sheet pattern changes from one college to another. For example, in JNNCE college, the blue-book pattern is as follows: There are number of boxes which includes the test number, date of the test, marks obtained (should be written both in numbers and words) followed by signature of the staff. The average marks is calculated from the best of three and entered in the separate box allotted for it.

A separate record is maintained by the concerned faculty to enter the average marks obtained by the students. After that, marks are entered manually into the VTU database by the respective faculty. This process is actually a time consuming one and hectic too.

To overcome this problem, “Blue-Book Marks Recognition System” is proposed. Instead of manually entering the marks, the faculty just needs to take an image of the marks region and give that image as an input to this proposed system. The system will automatically obtain the marks and display it along with the calculated average.

## **2. Literature Survey**

Number of articles were referred from IEEE journal and gathered some information from websites to gain sufficient knowledge about line detection and digit recognition.

### **2.1 Automatic Line Detection**

The Hough transform (HT), is a powerful global method for detecting edges. It transforms between the Cartesian space and a parameter space in which a straight line (or

other boundary formulation) can be defined.

The case where straight lines are present in image. First note is done for every point  $(x, y)$  in that image, all the straight lines passing through that point satisfy for varying values of line slope and intercept  $(m, c)$ .

$$\mathbf{y} = \mathbf{mx} + \mathbf{c} \quad (1)$$

Now if variables are reversed and looked, instead at the values of  $(m, c)$  as a function of the image point coordinates  $(x, y)$ , then

$$\mathbf{c} = \mathbf{y} - \mathbf{mx} \quad (2)$$

Equation 2 describes a straight line on a graph of  $c$  against  $m$

At this point, it is easy to see that each different line through the point  $(x_i, y_i)$  corresponds to one of the points on the line in the  $(m, c)$  space. Now, consider two pixels P1 and P2, which lie on the same line in the  $(x, y)$  space. For each pixel, representation can be made to all the possible lines through it by a single line in the  $(m, c)$  space. Thus a line in the  $(x, y)$  space that passes through both pixels must lie on the intersection of the two lines in the  $(m, c)$  space, which represent the two pixels. This means that all pixels which lie on the same line in the  $(x, y)$  space are represented by lines which all pass through a single point in the  $(m, c)$  space.

The advantage of the Hough transform is that the pixels lying on one line need not all be contiguous. This can be very useful when trying to detect lines with short breaks in them due to noise, or when objects are partially occluded.

## 2.2 Line Detection in Images through Regularized Hough Transform

Straight lines are present in almost all natural and manmade objects, that's major problem in image processing and computer vision, where determining the location and orientation of straight lines in images arises. In the authors convert a long sequence of images into an image with bands whose orientations are

indicative of the velocity of motion of the objects in the scene. Various application indicates the importance of the problem of line detection. Several approaches have been proposed for solving the line detection problem [1].

## Choice of Regularizer

Through the choice of the regularizer, suppress of regions is done in the transform domain which arise due to noise and nonlinear features in the image while preserving those features which correspond to lines in the image.

### A. Maximum Entropy Regularization

The problem of reconstructing dominantly black (low valued) fields with a few pixels having high values, arises frequently in NMR spectroscopy and astronomy. For those applications a popular regularization scheme which has been widely studied [2], is that of maximum entropy.

### B. L1 Regularization

It was argued that similar energy concentration to that achieved by maximum entropy methods could be obtained by using an norm penalty on the values of the field. In particular, It was shown that such-based regularization was well matched to "nearly black" objects. Motivated by this observation, as well as by ties to adaptive signal representation.

### C. $l_p$ Regularization With Small $p$

The regularizers do indeed accentuate the point structures in the parameter space relative to background clutter. But in their penalization of large values they also have the effect of reducing the amplitude of true peaks in the reconstructed image, leading to biased estimates. The amplitude of such peaks contains information about the length or "amount" of linear structure in the image at a given orientation and offset [2], and thus should be preserved if possible. In this regard, non-convex regularizers often perform better at

attenuating the smaller values while preserving the peaks.

### 2.3 Digit Recognition System Using Back Propagation Neural Network

Object character recognition (OCR) systems can contribute tremendously to the advancement of the automation process and can improve the interaction between man and machine in many applications, including office automation, check verification and a large variety of banking, business and data entry applications. The recognition rate of OCR (Optical Character Recognition) systems often decreases for degraded input digit images, such as stroke-connected, stroke-broken and noisy images. This can be caused by a lot of reasons, such as the poor quality of the source paper material, the bad quality of printing, the wrong threshold selected for scanning [3]. The recognition process of an OCR system can be divided into several steps like, preprocessing, segmentation, feature extraction and classification.

#### Pre Processing

When digit image is scanned, quality of image is degraded and some noise is added into the image. These parameters can reduce the accuracy of OCR system. So it is necessary to reduce the noise and improve the quality of the digit image for OCR system [3]. So main objective of preprocessing phase is to denoise and enhance the quality of scanned digit image.

#### Segmentation

Image segmentation is one of the most important steps leading to the analysis of processed image data. Its main goal is to divide an image into parts that have a strong correlation with objects or areas of the real world contained in the image. Segmentation is crucial for recognition system. Objective of segmentation is to partition an image into regions[4]. The main objective of segmentation of digit image is to separate the clear digit print

area from the non-digit area. For this purpose, suitable threshold is needed.

### 3. System Design

The figure 1 below shows the system architecture of the proposed system. These are few stages of development of the product. Control flow of the process shown in figure 2.

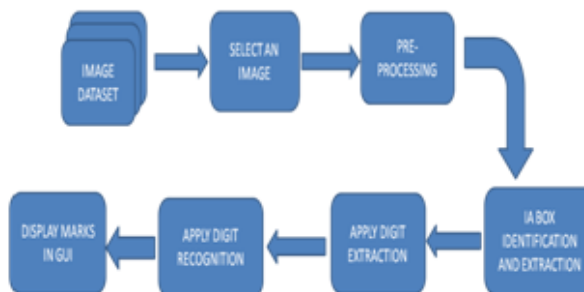


Figure 1: System architecture

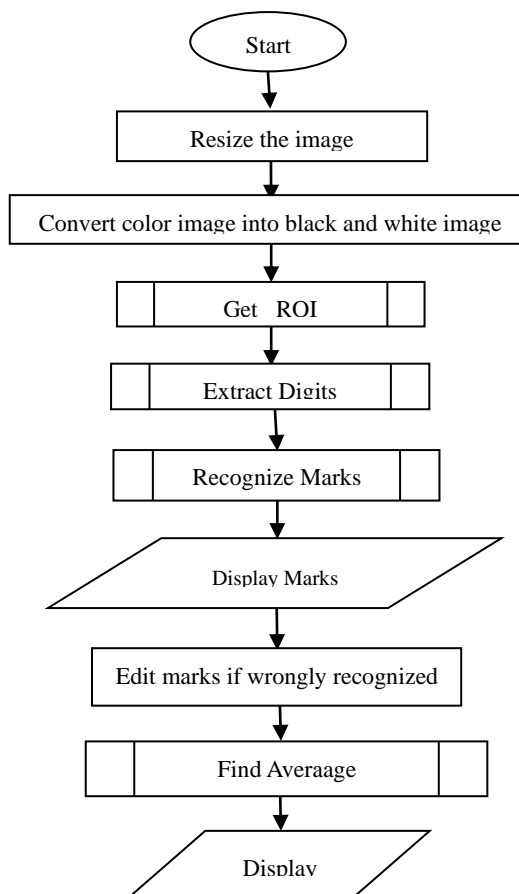


Figure 2: Program Control Flow

### 3.1 Image acquisition and Binarization

There are a few restrictions while selecting bluebooks for creating dataset since the main aim is marks recognition so the region containing marks is the point of concern. Following points should be taken into consideration while selecting an image to be added to dataset.

- 1) Digits should be written well within the box.
- 2) There should not be any cuts in the digits.
- 3) Two digits should not be connected i.e. digits should be written separately
- 4) Marks should be entered as two digits.
- 5) Marks box should not contain letters like A for absent etc.

After selecting the image, there may be some noise in it, like the image might be distorted or the marks might not be completely visible. To get the image in better quality, need to pre-process the selected image. In pre-processing, firstly resize of the image is done and then convert the image to black and white.

### 3.2 ROI Identification

Initially an image is taken, that image is converted to black and white image so that it would be easy for processing and the image will have only 2 pixel values i.e. 0 and 1. The conversion of black and white is done.

Reshape is used to convert Multi-dimensional arrays into 2-D arrays. The graythresh ignores any non-zero part of I. Now the converted image is resized to 500 X 900. The objects within the image which is less than 4500 pixels, those objects are filled with black. Now the objects which are in black needs to be complemented, as the region that is required needs to be in white. The object of interest will be in black, so in order to make it white, complement is done. After the image is complemented, the objects that are connected needs to be labeled.

After labelling the matrix, count of the number of pixels per object is done, then the labelled matrix and the number of pixels are compared element by element. After comparison the sum is made, and the maximum sum is found and that is considered as the biggest blob. The new image created, with the dimension of 500 X 900 and the image is filled with zero pixels [5]. The image is filled with zeroes by using the built in function *zeros*( ).

Then after having the new image with pixels filled with zeros, the condition is checked where in which, if the pixel value is 1 in the previous image, then that pixel is filled white in the new image. Now the objects which are less than 100 pixels in the new image are made 0 using, the reason behind giving the 100 pixels is because, if there is any noise in the image, those will also be made 0. To obtain the region of interest, Column pixels which are less than 300 and greater than 600 are made 0 and row pixels which are greater than 400 are also made 0 by making those regions as zero.

To obtain the region of interest as a separate thing, creation of new image is done of dimension 500X900, the region of interest is copied from the previous new image. The region of interest is copied pixel by pixel. Initially the left end and right end will be set to 0, the loop will be checking where the column has the pixel value 1, then from there the left end will have that value, for the right end value we will iterate from right to left. In order to get the right end value, the moment we get the pixel value as 1, the right end value will get that value. After getting the left end and right end values, the entire row is copied to the new image. Again the row value will be incremented and left end value will be initialized, row by row the pixels are copied to the new image. By this we can obtain the region of interest [6].

Now while copying the row by row pixels to the new image, there might be chances where the pixels are not aligned column wise. So there will be one range where in which that column value will differ. Now initialize is

made of that range and see that where that pixel value will fall, prior to range, if the pixels fall in black pixel, we will fill that region with white pixel. So from doing this, will get the region of interest in an aligned manner.

To crop the rectangular portion of original image bounded by (lines) should be initialized to 0. To get the respective values, conditions needs to be checked. For the top row value, once the pixel value as 1 is obtained, while iterating row wise, that value will be assigned to the top row. In the same way, the left column value is obtained. While iterating column wise, if the pixel value as 1 is obtained, that value is considered to the left column. The right column value is obtained by iterating from the right end of line to the left end line. Once the pixel value as 1 is obtained, that value is considered as the right column.

To obtain the bottom row value, subtraction of right column minus 10 is done and a point in the middle of the region of interest is obtained. The row wise iteration is done by checking for the white pixels. If while iterating, find the white pixels, the row value will be updated and goes to next row for iteration. When the black pixel is obtained, the row value will not be updated and finally bottom row value is obtained. The required region of interest is cropped [7].

### 3.3 Box Extraction and Digit Extraction

After getting the region of interest, extraction of the IA marks boxes is done separately. The boxes are separated by dividing at each horizontal line within the region of interest. The loop is written which checks for the pixel with the value 0, which indicates the black horizontal line in the region of interest. While iterating, consideration of each row and check column wise is done. If more than 5 white pixels in the column is obtained, the black line is not considered. If the black pixels is obtained throughout the row, then we will consider it as the horizontal line. Then the region above the line is extracted.

The same process is done to extract the remaining IA boxes. The first horizontal line is considered as the first IA marks box. The second horizontal line as the second IA marks box and so on. Now the three IA marks boxes is obtained. After getting the individual boxes, the digits within the box need to be extracted. The digits are extracted using the biggest blob concept. The obtained image is complemented in order find the biggest blob.

Where the objects in the image are filled with white and the object having the biggest blob is considered a single digit. That biggest blob digit is copied to another image. Now the previous is compared with the new image. Now the digit having same biggest blob is removed in order to get the remaining digit. This process is done to all the IA boxes. Now the required digit extraction is performed and obtained.

### 3.4 Digit Recognition

The digit recognition mainly depends on the neural network concept. After extracting the individual marks, it needs to be recognized. The marks are recognized one by one for each digit. The obtained digit images should go through some processing where in which dilation and erosion is applied. These functions are used in case in the image, if digits have some cuts and digits might not be visible completely.

Now we will store the leftmost column, rightmost column, top most row, bottommost row containing pixel value 1. Now having these values, rectangular portion of the image is cropped bounded by row and column value and store it as bounded image. Then the image is resized to 28 X 28.

In the proposed system, neural network along with back propagation algorithm is used and it has an input layer, along with a hidden layer and one output layer. The Input neurons are decided by training input vectors which is nothing but 10 in this system denoted by  $X_1$  .....  $X_{10}$ . Hidden layer will be having 15 neurons which are denoted by  $Z_1$  ...  $Z_{15}$ . And

one unit Output layer denoted by  $Y1$ . The bias on output  $Y1$  are denoted by  $W01$  and bias for hidden layer are denoted by  $Z_j$  ( $j = 1,2,3... 15$ ) is denoted  $V0j$  (where  $j = 1...15$ ). Weights for network are denoted by  $V_{ij}$  (where  $i = 1,2,3...10$  and  $j = 1,2,3...15$ ) for hidden and input units and weights are denoted by  $W_{j1}$  (where  $j = 1...15$ ) for output and hidden unit. In feed forward network, input neuron forwards the signal to the hidden units one by one. Hidden unit computes its activation and then forwards its signal to output unit [8]. Finally the output unit form the response of the network by computing the activation for the given input. Here we apply activation function given in equation (3).

$$f(x) = \frac{1}{1+e^{-x}} \quad (3)$$

Then the output at output layer need to be obtained, and again applies activation function as shown above. Now the output digit is obtained. Now the recognition is done to the left digit, and then the value is stored as the left digit. Then the same process is done to right digit and the value is stored as the right digit. Now the obtained left digit should be multiplied by 10 and the right digit is added to that. Now the recognized marks are displayed. If the marks are wrongly detected, it can be corrected by editing it. Then calculation of the average marks is done and displayed.

### 3.5 Average Computation

After obtaining the marks, the average of the best two internal marks should be done. In order to do so we check the individual marks with the remaining IA marks. Here all the three marks obtained are compared with each other for the computation of the average. If the first IA1 marks is less than IA2 and IA3 marks, then IA2 and IA3 marks are averaged. If the first IA2 marks is less than IA1 and IA3 marks, then IA1 and IA3 marks are averaged. If the first IA3 marks is less than IA1 and IA2 marks, then IA1 and IA2 marks are averaged. Then the average marks is displayed.

## 4. Results and Analysis

### 4.1. Snapshots

The snapshots have been included for following stages:

In figure 3, the snapshot shows the initial GUI window of our project where in which various options are provided to perform the respective operations. The first operation is the selection of the image. Second, the resetting of the image could also be done. Third, the obtained image can be converted to black and white image. Fourth, the region of interest within the image is obtained. Fifth, the obtained region of interest is subjected for digit extraction which will be helpful for digit recognition.



Figure 3: Initial GUI Window

The *select image* option is pressed, and then we need to select the image from the dataset and the respective image will be displayed in the available space as shown in the figure 4.

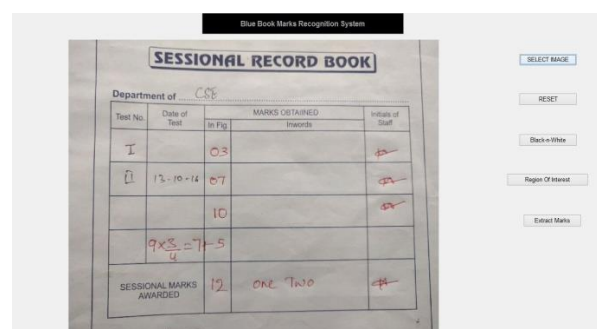


Figure 4: After selecting the image

In figure 5, the obtained image is converted to black and white image by pressing the Black and White option in the GUI window. The main advantage of doing this is that the image pixel values will be reduced to only 2 i.e. 0 and 1. This helps to distinguish between the two.

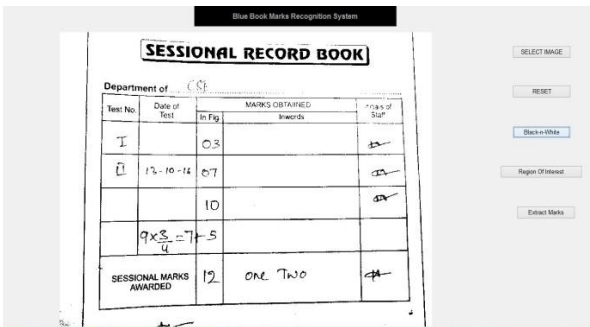


Figure 5: Converting the RGB image into black and white image



Figure 8: The average marks is calculated for old scheme.

In figure 6, after converting the image to black and white we select the region of interest of marks region for digit extraction.

In figure 9, after obtaining all the three IA marks, the average is calculated from the best of two IA marks and the displayed in the space provided according to CBCS scheme.

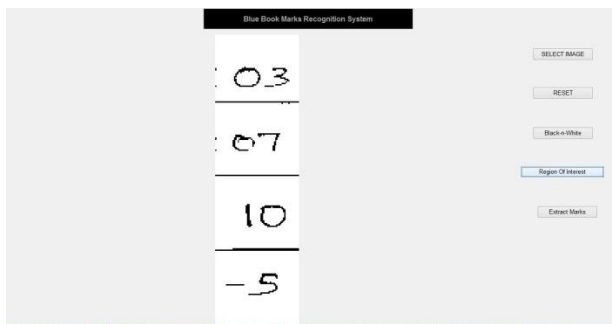


Figure 6: Obtaining the Region of Interest from the black and white image



Figure 9: Average calculated for CBCS scheme.

In figure 7, after extracting the region of interest we separate each and every IA marks. Then we separate each digit, and then we apply digit recognition to recognize the digits.

#### 4.2. Analysis

Table 1 shows the accuracy and processing time for the marks extraction and digit identification process.



Figure 7: Extracted Digits

Table 1: Analysis on accuracy and processing time

Particulars	Accuracy	Processing time
Extraction of marks region	100%	2 seconds
Digit Extraction	100%	3 seconds
Digit Recognition ( out of three IA marks )		3-4 seconds
At least 1 recognized	100%	
At least 2 recognized	90%	
At least 3 recognized	75%	

In figure 8, after obtaining all the three IA marks, the average is calculated from the best of two IA marks and the displayed in the space provided according to old scheme.

#### Remarks –

- Accuracy –The system has been tested for 30 inputs and has **good accuracy**.

- Processing time – The system has **low** processing time.

## 5. Conclusion

“Bluebook marks recognition system” is designed to help the faculty in saving time by automatically extracting marks from bluebook front sheet. It is the first step towards digitization. If 100% accuracy is achieved in recognizing digits then this system can be deployed in our college. The main intention is to extract and recognize the marks from the bluebook. By using MATLAB the system can be designed and implemented. Naturally care must be taken while entering the marks in the box. The processing of the system is fast and also the system is modifiable.

## References

1. R. Sharma, A. Jain, R. Sharma and J. Wadhwa, Character and Digit Recognition Aided by Mathematical Morphology, *International Journal of Computer Technology & Applications*, Vol. 4, Issue 5, 2013, 823-832.
2. Y. Chen, Q. Yang, Y. Gu, J. Yang, Detection of roads in SAR images using particle filter, *Proceedings of the International Conference on Image Processing, ICIP 2006*, Atlanta, Georgia, USA, October 8-11, 2006.
3. Shumjji Mori, Ching Y. Suen, Kazuhiko Yamamoto, Historical Review of OCR Research and Development, in *Proceedings of IEEE*, ISSN: 1558-2256, Volume: 80, Issue: 7, July 1992. 1029 – 1058.
4. M. Shridhar and A. Badreldin, Recognition of Isolated and Simply Connected Handwritten Numerals”, Department of Electrical Engineering, University of Windsor, Windsor, Ontario, Canada, Volume 19, Issue 1, May 1985.1-12. doi:10.1016/0031-3203(86)90025-7
5. Anil K. Jain, “Fundamentals of Digital Image Processing”, PHI, New Delhi, 10th Edition 2006.
6. R. Narayan and R. Nityanand, “Maximum entropy image restoration in astronomy,” *Annu. Rev. Astron. Astrophys.*, vol. 24,2003.
7. Pradeep Nijalingappa and Sandeep B. Machine learning approach for the identification of diabetes retinopathy and its stage, *International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*, IEEE International Conference, 2015.
8. Evelina Maria De Almeida Neves, Adilson Gonzaga, Annie France Frere Slaets, A Multi-Font Character Recognition Based on its Fundamental Features by Artificial Neural Networks, *IEEE*, 1997.