DOI: 10.21917/ijsc.2025.0513

# AI-POWERED MARK RECOGNITION IN ASSESSMENT AND ATTAINMENT CALCULATION

**J. Annrose, A. Shibin Sam, S. Samnath, Jennings Rousseau Raj and I.R. Shanjeya Bhaarath**
*Department of Information Technology, St. Xavier's Catholic College of Engineering, India*

***Abstract***

***The cornerstone of a quality education system is Outcome-Based Education (OBE), which revolves around predefined outcomes or goals. Based on the best practices of NEP 2020, all the education institutions and accreditation authorities rely on Outcome-based teaching, learning, and evaluation. To measure the defined outcome of each course, it is essential to enter the question-wise marks in the student exam portal for CO attainment calculation. Traditional methods adopt the manual entry of marks. This is time-consuming and error-prone, leading to additional work for each examiner after evaluation. This project uses computer vision and AI to automate the identification and calculation of marks from the corrected handwritten exam answer sheets. It extracts information like roll numbers, question numbers, and marks, calculates total marks, and exports data to Excel sheets, hence reduces manual effort and improves efficiency. Deep neural network-based pre-trained models are used for the effective identification of handwritten marks. We have trained and tested the proposed system through continuous assessment and internal examination answer sheets available in our institution.***

***Keywords:***

***Handwritten Digit Recognition, Outcome Based Education (OBE), Measuring Outcomes, Attainment Calculation, Computer Vision, Deep Learning***

## 1. INTRODUCTION

The management of examination scores is crucial in our current educational system as it ensures that the students' performances are individually assessed and recorded. The present education relies on OBE-based teaching, learning, and evaluation. To promote a learner-centric approach it is essential to accurately capture and quantify the students' knowledge, skills (Cognitive, Affective, and Psychomotor), and attitude on assessments. The Cognitive or intellectual skills are evaluated by formative and summative assessments. However, the process often involves tedious manual tasks such as evaluating and assigning scores on the front sheet against the respective questions, calculating scores for each question, totaling and entering marks for each student in the respective exam portal student wise and question-wise to determine course outcome (CO) attainment, which is time-consuming and prone to errors. Despite technological advancements, many educational institutions still rely on traditional methods to avoid potential errors, which highlights the need for more efficient and automated solutions.

Traditional methods usually involve manual data entry, paper-based record-keeping, and spreadsheet-based calculations. For accreditation purposes, most of the education institutions follow the above traditional methods and it is tedious for the faculty and staff to maintain this process. The proposed work offers a paradigm shift in exam score management, enabling educators to automate labour-intensive tasks, reduce errors, and improve overall efficiency.

This paper proposes an innovative system that utilizes modern techniques to streamline the management of exam scores. By utilizing various image processing algorithms, OpenCV-based table detection, and deep learning models for mark detection and total calculation. Our system aims to significantly reduce the time and effort required for these tasks while ensuring high levels of accuracy and reliability.

Image processing techniques like binarization, deskewing, line detection using the Hough transform, and cell detection by sliding window technique are used to automatically identify and extract tables or score sheets from scanned or digital images.

Deep learning algorithms, especially Convolutional Neural Networks (CNNs), have shown remarkable capabilities in tasks such as object detection, pattern recognition, and classification. Many works are in progress related to automatic evaluation and mark entry. Still, the answer script evaluation is not completely automated. There are many challenges in accurately evaluating the descriptive answers automatically using image processing and AI. This proposed work focuses on automatic mark recognition after manual evaluation.

The main contributions of the works are as follows:

- Automated Table Detection: Uses various image processing algorithms like Hough transform and sliding window technique to identify and extract the handwritten regions from the table that are detected earlier.
- Automated Mark Detection: Employs pre-trained Convolutional Neural Networks (CNNs) that are trained using MNIST and EMNIST datasets to accurately detect and recognize marks from the answer sheets.
- Total Calculation: Totalize the detected marks.
- Automated conversion to spreadsheet format: Converting the detected marks along with the total into a spreadsheet.
- Web Interface Creation: Finally, a web interface is created, which acts as the front end for the mark detection system.

We have done our evaluations through continuous assessment internal examination answer sheets available in our institution and the results of the table and mark recognition seem to be more effective and promising than the traditional methods.

The rest of the paper is organized as follows. A literature review of this proposed work is available in section 2. The proposed methodology is depicted in section 3. The next section includes the description of the dataset and its experimental results and discussion. The last section depicts the conclusion and future work.

## 2. RELATED WORK

AI-based Examination Assessment Mark Management System using deep learning techniques was proposed [1]. It automatically detects tables from the students' answer scripts detects individual marks from each table cell and calculates the total from the students' answer scripts. To detect the marks pre-trained Convolutional Neural Network (CNN) was used for training and testing on the handwritten digit data. The experimental results show high mark detection accuracy from the scanned answer sheet. [2] proposes an ensemble technique and a unique feature extraction method to recognize handwritten digits. The results were compared and analysed on state-of-art machine learning methods including SVM, AdaboostM2, and artificial neural network algorithms. It was proved that the unique feature selection and ensembling methods were essential for attaining high performance. The misclassification rate and the performance of the work were analyzed using a confusion matrix and F1 score. A machine learning-based application to evaluate answers was proposed [3]. The main contribution of the work is to reduce the time and manpower. In the manual evaluation, the manpower requirement was very high, and the scheme of evaluation was subjective and differed from person to person. Hence, there may be a chance of awarding different marks for the same answers. This method works on a key basis, here the important keyword and the scheme of evaluation was separately provided for each question. This application system would map the question with the corresponding keyword provided by the moderator [4]. A cutting-edge machine learning approach and techniques are utilized to create a new evaluation model based on supervised learning. Here, student answer scripts are scanned, and handwritten words are identified using Tesseract OCR with Python, handwritten words are matched to keywords in an adaptive database using Python-based code, and marks are then assigned per the evaluation scheme that the course coordinator provided. Once all students' grades have been combined, an authorized signature may also be supplied. A secure email account must be set up to transmit the evaluated grades to the assigned staff members. [5] This work also uses CNN techniques to classify the handwritten digits. To observe the variation of accuracies in pre-trained CNN classification models on handwritten digits, hyperparameters like hidden layers neurons, and epochs are used to do the classification task and the observations were recorded for varying values. From the experimental observations, it was proved that the highest accuracy was obtained in the case of following neural network structure with Conv1, pool1, Conv2, pool2, and 2 dropouts, which resulted in 99.21% accuracy in the 15th epoch. [6] A camera-based multiple-choice grading idea is proposed in this work. According to them, the answer sheets are first captured by a camera and allocated using the Hough transform and then skewed corrected to get the "bird's eye" view of the paper. Next, the tick mark corresponding to the answer for each question can be recognized by allocating the marks, which wraps the answer area. [7] have offered a technique for identifying tables in documents. The suggested technique locates the image's horizontal and vertical lines. An SVM classifier determines whether a given object is a table or not, by looking at the intersections of the horizontal and vertical line groups. Although table patterns that are entirely encircled by lines are easily recognized, other types of table layouts exist as well. Some tables may have no lines at all, or they may just have non-intersecting lines that isolate the table header. Then it is submitted to the Regional Proposal Network (RPN) for table detection. The suggested method employs a Faster Recurrent Convolutional Neural Network (RCNN) to recognize tables from pictures and offers the notion of categorizing text areas and non-text regions. This study shows that systems based on deep learning are suitable for students to first identify handwritten words using Tesseract OCR with Python.

From the related review, it was found that till now the answer script evaluation is not completely automated. There are many challenges in accurately evaluating the descriptive answers automatically using AI. This proposed work focuses on automatic mark recognition after manual evaluation.

## 3. PROPOSED SYSTEM

### 3.1 MARK TABLE DETECTION

Once the answer sheets are fed into the program, they are processed using image processing techniques such as binarization, desk Ewing, noise removal, Hough Transform and sliding window technique to detect various tables in the given image along with the coordinates of individual cells in the table.

The Hough transform works by mapping the points in the image to lines in the form of parameters and then identifying the line clusters that correspond to the lines present in the answer sheet image. The sliding window algorithm works by sliding a window across the image and checking for the presence of lines at the window edges. The window is resized until a rectangular region containing lines at all four edges is found. After the cells were detected, the Connected Component Labeling technique which involves labeling connected regions in the image that contain cells is used to detect the tables. The above algorithms were generally used to make the process of finding tables easier.

Tesseract OCR library is now used to identify the tables that are used for entering marks. This is done by selecting the tables which contain the keyword 'ques' and 'num'. The table that contains the mark is identified by selecting the images with red handwritten numbers, as the mark is written in the answer sheet using a red pen.

The mark is extracted from the detected mark table by detecting the keyword 'mark' in the cells of the table and extracting the cells from the same row as the above-detected cell as the handwritten mark cells. These cells are cropped out and are converted into grayscale for further analysis.

Gaussian blur is applied on the grayscale image to reduce noise. Then, canny edge detection is used to find the edges that are present in the image. The contours that are greater than 5 in width and 15 in height are assumed to be potential digits. Each contour is redrawn as a 28X28 pixels binary image. This image is given as an input to a pre-trained CNN model which is trained on MNIST and EMNIST datasets.

The prediction output of each contour is combined to give the mark that is written in each cell. The output is stored in a .csv spreadsheet file which can be viewed and edited.

This system encompasses various benefits. Firstly, it significantly improves efficiency by reducing the time and effort required for manual grading, allowing educators to focus on more value-added tasks.

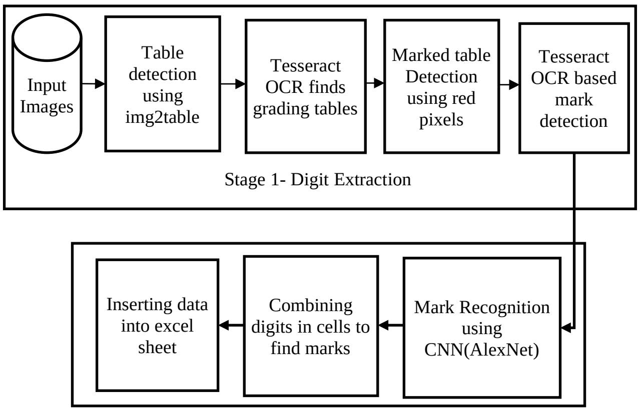



Fig.1. Block diagram of the proposed system

It is a two-stage object detection process, in the first stage tables are detected and hand written marks are identified using Tesseract and in the second stage marks are recognized using pre-trained CNN model called AlexNet trained on MNIST dataset.

Convolution Neural Network architecture uses stack of layers which consist of Convolution operation, pooling and fully connected layers. It uses RELU activation functions at the intermediate layers and sigmoid functions at the output layers. In order to reduce the cross-entropy loss in each iteration, Adam optimizers are used to finetuning the hyperparameters.

Convolution is a mathematical operation performed on two functions, ie, the original image and filters or kernel functions to generate the reduced sized feature maps. Further it gets reduced by pooling layers and vectorized by fully connected layers

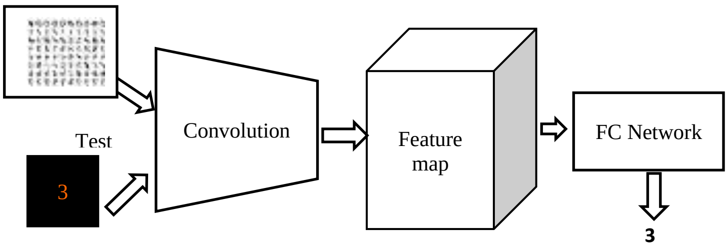



Fig.2. Digit Recognition using CNN

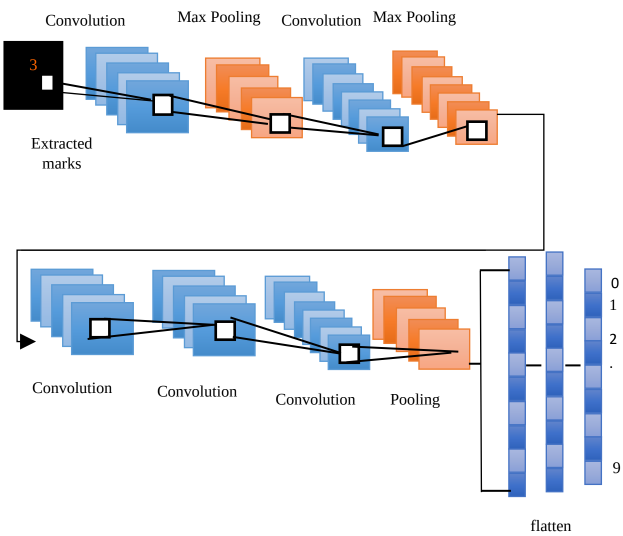



Fig.3. AlexNet Architecture for Digit Recognition

AlexNet uses 8 layers, with five convolution layers, 3 pooling layers and 2 fully connected layers. Moreover, it enhances the accuracy and consistency of the grading process, ensuring fair evaluation of student performance. Additionally, the system is highly scalable and capable of handling large volumes of answer sheets efficiently, thus catering to the needs of educational institutions of varying sizes. The implementation of the system involves several phases, including research, prototyping, integration, deployment, and maintenance. Throughout these phases, the system undergoes rigorous testing and optimization to ensure reliability and performance. By extracting information such as question numbers, and marks, the system significantly reduces the time and effort required for manual grading. Moreover, the accuracy of the system ensures reliable results, improving the overall efficiency of the exam scoring process.

## 4. EXPERIMENTAL RESULTS

The experiments to implement the project are conducted on a jupyter notebook using Python language. The openCV library is used for various image processing steps that are required to reduce noise from the input data and improve the output quality. The following are the steps of experimentation. I've tried two different approaches in order to find the table regions in the answer sheet's image.

### 4.1 TABLE AREA DETECTION USING IMAGE PROCESSING

In the first method we tried, and converted the image into grayscale and defined a rectangular structuring element for morphological operations. The wide and short nature of this structuring element is suitable for emphasizing horizontal lines commonly found in tables.

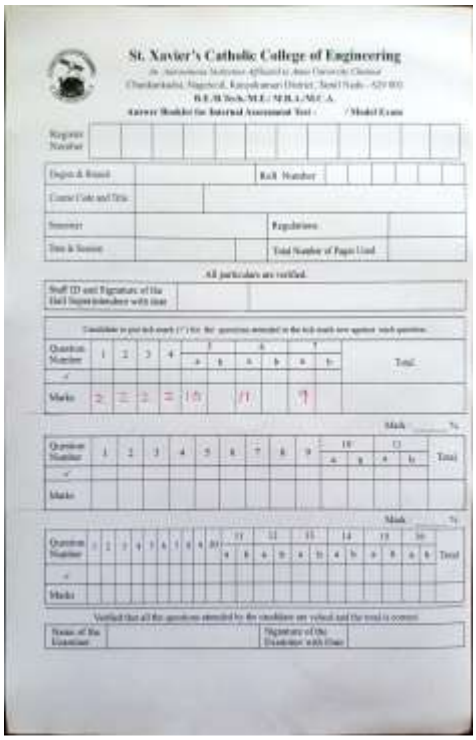



Fig.4. Input images of Answer Script (Sample1)

Now, gaussian blur is applied to reduce the noise and soften the image a bit. Also, a blackhat morphological operation is done to darken the lines present in the image for better detectability. Now, the gradient in the x-direction is calculated using the Sobel operator. This further highlights the horizontal lines. Now a closing operation is performed to fill up the broken horizontal lines that occurred because of the steps before. Otsu's thresholding is applied to obtain a binary image where potential table lines are strongly represented in white.

This method of table detection proved to be less reliable, as Otsu's thresholding can be unreliable at times, marking various non-table regions as tables.

Fig.5. Sample evaluated answer script of End Semester Examination (Sample2)

### 4.2 TABLE AREA AND CELL DETECTION USING IMG2TABLE LIBRARY

In the second method, a table detection library called img2table which uses openCV image processing and OCR to find out table regions. This library proved to be effective in not only detecting the table regions, but also to detect the table cells that are present in the tables. The input image is converted into grayscale and then fed into the img2table library for detecting the tables. The library returns an array of custom Table objects for each input image, which contains the position data of the cells present in the table. This can be used to extract the marks by using row and column IDs that are provided by the table detection operation.

### 4.3 MARK TABLE DETECTION

Now the detected tables contain various data including the roll number area along with the other mark tables, in which only one of them contains actual mark data. To find the contours that are mark tables, we use Tesseract OCR to detect the text contents of each contour. We already know that, if the table is intended for entering marks, it should contain the word "ques" and "num" in it. Thus, the OCR result is scanned for the above strings and the contour that contains the strings is added to the detected tables. Only one of the tables detected contains the handwritten marks. The table that contains the reddest elements is assumed to be the mark table, as the handwritten marks are usually marked using a red pen.

### 4.4 HANDWRITTEN MARK PROCESSING

The mark is extracted from the detected mark table by detecting the keyword 'marks' in the cells of the table and extracting the cells from the same row as the above-detected cell as the handwritten mark cells. These cells are cropped out and are converted into grayscale for further analysis.

Gaussian blur is applied on the grayscale image to reduce noise. Then, canny edge detection is used to find the edges that are present in the image. The contours that are greater than 5 in width and 15 in height are assumed to be potential digits. Each contour is redrawn as a 28X28 pixels binary image. This image is given as an input to a pre-trained CNN model which is trained on MNIST and EMNIST datasets.

The prediction output of each contour is combined to give the mark that is written in each cell. The output is stored in a .csv spreadsheet file which can be viewed and edited.

Fig.6. Table Detection

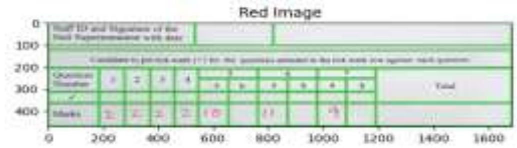



Fig.7. Identified Table Containing Marks with recognized cells

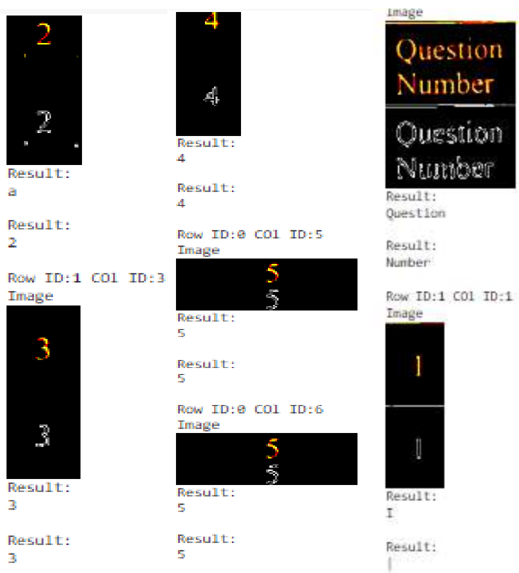
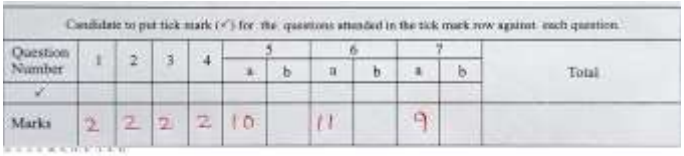



Fig.8. Extraction of question numbers



| Candidate to put tick mark (✓) for the questions attended in the tick mark row against each question. | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Question Number | 1 | 2 | 3 | 4 | 5 | | 6 | | 7 | | Total |
| | | | | | a | b | a | b | a | b | |
| ✓ | | | | | | | | | | | |
| Marks | 2 | 2 | 2 | 2 | 10 | | 11 | | 9 | | |

[2, 2, 2, 2, 10, 0, 11, 0, 9, 0, 0]

Fig.9. Detected marks

Fig.10. Mark stored in spreadsheet for sample1

| Questio | | | Question | | | i | | ii | | iii | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|
| No. | () | Marks | No. | | () | Marks | () | Marks | () | Marks | Marks |
| 1 | | 0 | 11 | a | | | | | | | |
| 2 | | 0 | | b | | 4 | | | | | 04 |
| 3 | | 0 | 12 | a | | 12 | | | | | 12 |
| 4 | | 0 | | b | | | | | | | |
| 5 | | 0 | | a | | | | | | | |
| 6 | | 0 | 13 | b | | 12 | | | | | 12 |
| 7 | | 0 | 14 | a | | | | | | | |
| 8 | | 0 | | b | | 4 | | | | | 04 |
| 9 | | 1 | 15 | a | | | | | | | |
| 10 | | 0 | | b | | 12 | | | | | 12 |
| | | | 16 | a | | | | | | | |
| | | | | b | | | | | | | |
| Total | | 01 | | | | | | | | | 44 |

Fig.11. Mark stored in spreadsheet for sample 2

## 5. CONCLUSION AND FUTURE WORK

The system's scalability and efficiency contribute significantly to its effectiveness in handling large volumes of answer sheets. The system can successfully process corrected answer sheets to extract image regions, which contain the handwritten mark as separate cropped images. It also detects the handwritten marks using a pre-trained CNN model and the resultant marks are added to a spreadsheet for further reference. A web interface can be developed in the future to encapsulate the system, thus giving the user a neat and clean interface to interact with the proposed system.

## REFERENCES


[1] L. Logeshvar, A.B. Premnath, R. Geethan and R. Suganya, "AI based Examination Assessment Mark Management System", *Proceedings of International Conference on Secure Cyber Computing and Communications*, pp. 144-149, 2021.

[2] L. Li, W. Yaonan and Z. Yexin, "A Handwritten Digit Recognizer using Ensemble Method", *Chinese Automation Congress*, pp. 469-473, 2015.

[3] Prince Sinha and Ayush Kaul, "Answer Evaluation using Machine Learning", *Conference: McGraw-Hill Publications*, 2018.

[4] S.T. Prakruthi, V. Hanuman Kumar, "Automated Students Answer Scripts Evaluation System using Advanced Machine Learning Techniques", *International Journal for Research in Applied Science and Engineering Technology*, pp. 1-7, 2018.

[5] N. Mu and J. Gilmer, "MNIST-C: A Robustness Benchmark for Computer Vision", *Proceedings of International Conference on Machine Learning*, pp. 1-6, 2019.

[6] John Kelvin Tetteh Domeh, Emmanuel Kofi Akowuah, "VisioMark: An AI-Powered Multiple-Choice Sheet Grading System", *Master Thesis, Department of Computer Science, Kwame Nkrumah University Of Science and Technology*, pp. 1-46, 2023.

[7] S. Ahlawat, A. Choudhary, A. Nayyar, S. Singh and B. Yoon, "Improved Handwritten Digit Recognition using Convolutional Neural Networks", *Sensors*, Vol. 20, No. 12, pp. 1-6, 2020.

[8] E. Enstrom, G. Kreitz, F. Niemelä, P. Söderman and V. Kann, "Five Years with Kattis-using an Automated Assessment System in Teaching", *Frontiers in Education Conference*, pp. 1-6, 2011.

[9] W. McIlhagga, "The Canny Edge Detector Revisited", *International Journal of Computer Vision*, Vol. 91, No. 3, pp. 251-261, 2011.

[10] M. Kalbasi and H. Nikmehr, "Noise-Robust, Reconfigurable Canny Edge Detection and its Hardware Realization", *IEEE Access*, Vol. 8, pp. 39934-39945, 2020.

[11] M. Mittal, A. Verma, I. Kaur, B. Kaur, M. Sharma, L.M. Goyal, S. Roy and T.H. Kim, "An Efficient Edge Detection Approach to Provide Better Edge Connectivity for Image Analysis", *IEEE Access*, Vol. 7, pp. 33240-33255, 2019.