Using Machine Learning and Image Processing to Effectively Analyze Scanned Answer Scripts after an Examination

Apoov Sadana
Delhi Private School, DUBAI

ABSTRACT
The growth of artificial intelligence over the years has managed to solve some of the most difficult problems in the field of technology. The applications of machine learning are vast. This paper attempts to propose a way to use artificial intelligence to increase the efficiency of paper analysis. The aim of the paper is to use machine learning and image processing for the effective digital transformation of answer papers after an examination. In a large portion of schools in India, after every test correction, there are certain procedures that need to be followed. These procedures make the task cumbersome, time consuming and prone to errors. In order to tackle the problem, a program was developed to work on the scanned copies of answer scripts for generating an excel file consisting of the analyzed data.

First, the answer script was redesigned to increase the efficiency of the OpenCV algorithms. Secondly, the various functions of the OpenCV library were used to collect data from papers. Finally, the collected data was then passed through the MNIST dataset for recognition of handwritten digits and generation of excel file.

The program is able to identify the location of all the handwritten data within the answer script (100%) and is able to predict the handwritten digits up to an accuracy of 92%. In order to eliminate errors, an easy checking algorithm is also made. The application of artificial intelligence in paper analysis has many advantages ranging from reduced effort to time saving.

Keywords-- analysis of scanned answer scripts, artificial intelligence, digit prediction using MNIST dataset, image processing (OpenCV), using excel via python

I. INTRODUCTION

1.1 Problem
In many schools in India, after evaluating all the answers scripts, the teacher has to perform the following tasks
1. Totaling the paper
2. Writing down the marks secured by the student in each question on the first sheet of the answer script
3. Creating an excel file from the first sheet of all answer scripts
4. Storing the papers for security purposes
These procedures make the task cumbersome, time consuming and prone to errors.

II. ALGORITHM AND SOLUTION

2.1 Redesigning the answer script
In order to tackle the problem, the answer script was first redesigned. Fig 1 is a sample of the old answer script and Fig 2 is a sample of the redesigned answer script.

In the old answer script, the question numbers were written in the left margin and the marks were written along with the answers. This made the data unorganized and difficult for the computer to extract all information from the answer script.

In order to collect all the relevant data in one place, a table was added to the redesigned answer script. It was made dark black so that it could be easily thresholded. The right column was used by the students to write down the question number and the left column was used by the teachers to write down the marks secured by the student in each question. The marks for each question were written in the box just to the left of the box containing the question number. This modification gave better results and also retained the simplicity of the process.
The first page of the answer script was also modified to include the admission number of the student enclosed within a box.

Fig 2. Question numbers in left column of table and marks in tight column of table

Fig 3. The first page of answer script consists of the admission number of the student

2.2 Collecting Data from answer scripts

Two scanned copies were taken for each answer script – one colored and one black and white (not grayscale). (Using image processing to produce the black and white copy did not give the desired results.)

The following algorithm was followed:

Thresholding, contouring and dilating were performed on the black and white image for accurate results. Once desirable contours were traced in the black and white image, they were cropped from the color image as fine details were lost in the black and white scan. This could be done as the black and white & color images were superimposable.

1. Inverted thresholding to switch black and white colors within the image (this made the table and admission box white in color and thus, it could be contoured)

2. Dilating of the thresholded image with a kernel size of 10. This filled up the black holes within the image, if any

3. Contouring the image with tree retrieval mode to detect all contours within image.

4. Checking if the image is of the first page of the answer script (containing the admission number) or of a page containing the answers. This is done
by checking the number of contours in the image. It was found that the first page of the answer scripts had less than 30 contours whereas other pages had more than 30. (This value will fluctuate depending upon the modifications made by the school on the answer script. For example: adding the school logo). As the answer scripts are scanned in order, this step allows the program to know when a new answer script has started.

5. Contouring image with external retrieval mode to detect only the outermost contours.

6. Cropping the table or the admission box from the image. This is done by finding the external contour of maximum area. In case of the admission box, the rightmost x coordinate of the admission box must be greater than one third of the width of the image. In case of the table, the x coordinate is less than one third of the width.

7. Cropping digit boxes within the image cropped in step 6. The digit box is the box in which the marks, question number or a digit of the admission number is written. The digit boxes are cropped by contouring again.

8. Saving images of the digit boxes. In case of the table, digit boxes with approximately same y coordinate value are stored with similar filenames. The left one is saved as marks and the right one as question number.

2.3 Modifying images of the digit boxes

Three types of digit boxes are possible in an answer script:

1. Those containing only a single digit
2. Those containing more than one digit (may or may not be separated by a decimal point). Example: question number 12 or marks 3.25
3. Those which are empty

In order to distinguish between the above three types of digit boxes and process them accordingly, the following algorithm is followed:

1. Finding the contour of maximum area within the digit box.
2. If the height of the found contour is less than a specific value (will depend upon the resolution of scanned images), then the digit box is empty and not used in any further processing. Else, the contour consists of one digit at least.

The steps below are executed only if the digit box consists of one digit box at least.

3. Finding contours of height similar to the largest contour. If found, these contours will also be digits and the digit box would contain more than one digit.
4. Checking for the presence of a small contour between any two consecutive digit contours. If present, it would be a decimal point.

5. Cropping all the digits from the digit box and saving them with appropriate filenames to express the position of digits and decimal point in the actual number.

2.4 Predicting digits with MNIST dataset

All the digits which are cropped out are passed into a neural network trained on the MNIST dataset.

2.5 Creating the check sheet

There is a very high probability that at least one number in an answer script will be predicted incorrectly by the program. However, while analyzing papers, the accuracy must be 100%. In order to rectify these errors, a check sheet is created in excel. Fig 4. shows a part of this check sheet. It displays the numbers predicted by the neural network as well as an image of the actual numbers (which was cropped in step 8 of section 2.2). The teacher is expected to compare the actual and the computer generated numbers and in case of any errors, the teacher may rectify them there itself and save the file.

In order to speed up the process, the computer generated and actual number have been placed side by side. This allows the teacher to quickly skim through the excel sheet and locate the errors.

2.6 Generating master sheet

After the errors in the check sheet have been rectified, a master sheet is created using the data from the check sheet. This ensures that the master sheet is 100%
accurate. The master sheet is made based on a layout that is provided by the teacher which includes the number of questions in the paper and number of subparts per question, if any. The master sheet displays

1. The marks secured by the student in each question
2. Total marks secured by the student
3. Questions left unattempted by the student
4. Questions that are not marked by the teacher (but have been attempted by the student)
5. Answer scripts. Clicking on the marks secured by a student in any question, say X, displays that sheet of the answer script on which the student has answered question X.

![Master Sheet](image)

Fig 5. Master sheet displaying answer script analysis. ‘NA’ stands for not attempted

### III. RESULTS

Section 2.2 and 2.3 produce results that are nearly 100% accurate. Prediction of digits using MNIST dataset has an accuracy of 92% (digits and not complete numbers). This may seem ambiguous because neural networks trained on the MNIST dataset have achieved accuracies greater than 99%. But these neural networks were tested on images that were originally 28x28 in dimensions (the MNIST format). However, while cropping out digits from the answer scripts, the dimensions of the digits will vary as the handwritings of different students will vary and no student can accurately make a digit which is exactly 28x28 in dimensions. Hence, the images of cropped digits first need to be converted into the required format and then predicted with the neural network. It is due to this conversion that the accuracy is around 92%.

The program can also detect decimal numbers which means that it can detect question subparts and fractional marks.

### IV. FUTURE AND SCOPE

The advantages of an application capable of analyzing answer scripts include:

1. **Saves time.** All tasks mentioned in section 1 have been reduced to scanning of papers and rectifying of errors in the check sheet.
2. **Reduces Errors.** No more errors in totaling the paper. The program also notifies the teacher if any question has been left unmarked.
3. **Prevents paper tampering.** As the excel sheet has a record of the answers, students cannot modify their answers and ask for marks.
4. **Provides easy access.** Previously, all papers had to be stored and during paper reopening, teachers had to look for specific papers among thousands of others. Using the program, they can access the paper from the excel sheet as it also contains the answer scripts.
5. **Paper Recycling.** Papers no longer need to be stored as the excel file contains all information.

### REFERENCES

No paper was referenced to as the research actually attempts to apply machine learning and image processing algorithms to a real life problem.
Appendix
GUI Demonstration

1. Input

Similarly, a black and white copy of all the above images was also scanned. The images must be scanned in the order as shown above – one paper followed by another.
2. Starting the GUI – Entering location of scanned pictures

![Fig 7](image)

3. Entering question paper layout

![Fig 8](image)

4. Start Processing

![Fig 9](image)
5. After Processing

Fig 10

6. Saving master sheet

Fig 11
7. Master Sheet

Clicking on D4 gives

Clicking on H6 gives

Fig 12