International Journal of Research in Engineering, Science and Management
Volume 4, Issue 7, July 2021
https://www.ijresm.com | ISSN (Online): 2581-5792

Automation of College Work using Artificial Intelligence

Khan Shabaz Umarhayat[1*], B. Sudhakara[2], Sudeeksha[3], Ashwitha Acharya[4]
[1,3,4]*Student, Department of Computer Science and Engineering, Srinivas Institute of Technology, Mangalore, India*
[2]*Assistant Professor, Department of Computer Science and Engineering, Srinivas Institute of Technology, Mangalore, India*

***Abstract*: College has many exams every month and the staff has to make multiple data points of the same. They also have to calculate the total, average and CGP/Percent of each and every student writing that exam. Exam Seat allotment is one task that every student has to make sure they are updated over the same for each exam or sometime for each subject too in the same exam. The problem here is students typically have to spend some amount of time to go to the notice area and find out where his seat is allotted. Automation of college work project is implemented in MATLAB image processing toolbox. The project is implemented for both Real time and Non-Real time. The proposed method has four stages. First is Pre-Processing and second is Feature Extraction and third is Segmentation and fourth Recognition. In case of Non-Real time, the first stage is used to browse the image, second stage is extraction of the features from images using OpenCV as they contain Pre-trained Models that are directly used to provide results accurately compared to other methods over such task.**

***Keywords*: Feature extraction, Segmentation, Recognition, OpenCV.**

1. Introduction

There are several problems or works in the college/University that can be automated. College has many exams every month and the staff has to make multiple data points of the same. They also have to calculate the total, average and CGP/Percent of each and every student writing that exam. The idea here is; the lecturer has to use our application in which he/she has to click the picture of the front page of the answer sheet/book. Our system will be able to find the details of the student and the subject using machine learning and automatically update the marks of the particular subject. As all subjects get updated into the data sheet. Our system will also be able to calculate the total, average and CGP/Percentage of each and every student The proposed method is based on the use of Python OpenCV library in complete detail. Computer Vision refers to the field of study which deals with how computers perceive images. It involves feeding images into a computer and then trying to gain high- level intelligence from it using different algorithms. It works in close coordination with fields like Machine Learning and Artificial Intelligence. Computer Vision is a broad field and is progressing rapidly. The recognition system first accepts a scanned image as an input. The next major upgrade in producing high OCR accuracies was the use of a Hidden Markov Model for the task of OCR.

This approach uses letters as a state, which then allows for the context of the character to be accounted for when determining the next hidden variable. This lead to higher accuracy compared to both feature extraction techniques and the Naive Bayes approach. Requirements like Flask, Numpy, OpenCV 3, Spell Checker Autocorrect. Henceforth Image Processing Technique using Artificial intelligence will be titled as “Automation of college work using AI”.

2. Literature Survey

In this paper they have recognize text from paper documents, photo graphs etc. The Handwritten text system is commonly used system in various applications, and it is a technology that is a mandatory need in this world as of now. Before the correct implementation of this technology we have dependent on writing texts with our own hands that result in errors. It’s difficult to store, access physical data and process the data in efficient manner. Manually it is needed to update, and labour is required in order to maintain proper organization of the data. Since for long time we have encountered a severe loss of data because of the traditional method of storing data. Adoption of the Handwritten Text Recognition software is a practical idea and, it is easier to store and access data that was traditionally stored. Furthermore, it provides more security to the data. One such example of handwritten text Recognition software is the Google Lens and example for hardware is OCR scanners. Handwritten character recognition is a complex problem, which is not easily solvable. The necessity is around dataset and database.

The proposed method is based on the use of speeded up robust feature. The next major upgrade in producing high OCR accuracies was the use of a Hidden Markov Model for the task of OCR. This approach uses letters as a state, which then allows for the context of the character to be accounted for when determining the next hidden variable. This lead to higher accuracy compared to both feature extraction techniques and the Naive Bayes approach. The main drawback was still the manual extraction features, which requires prior knowledge of

*Corresponding author: shabaz4646khan@gmail.com

the language and was not particularly robust to the diversity and complexity of handwriting. Systems of deep learning algorithms are known as artificial neural networks. The downside of this model is that it doesn't incorporate a language model to generate the sequence of characters and words. It is completely dependent on the visual classification of each character without considering the context of the constructed word.

In this paper we are Tensorflow is an open-source platform for machine learning. It is a deep learning framework, we use TensorFlow to build OCR systems for handwritten text, object detection, and number plate recognition. This solves accuracy issues. As a well- positioned AI development company, Oodles AI explores how to build and deploy handwritten text recognition using TensorFlow and CNN from scratch. Handwritten Text Recognition (HTR) systems power computers to receive and interpret handwritten input from sources such as scanned images. The systems are able to convert handwritten texts into digital text or simply can digitize, store, and extract valuable information for accurate analysis. At Oodles, we use tools like OpenCV and provide TensorFlow development services to build a Neural Network (NN) which is trained on line-images from the off-line HTR dataset. This Neural Network (NN) model split the text written in the scanned image into segmented line images. These line-images are smaller than images of the complete page image.

The algorithms used in character recognition can be divided into three categories: Image Pre-processing, Feature Extraction, and Classification. Neural Network based Handwritten Character Recognition system with feature extraction. They are normally used in sequence – image pre-processing helps makes feature extraction a smoother process, while feature extraction is necessary for correct classification. Here's how they work: Image pre- processing is crucial in the recognition pipeline for correct character prediction. These methods typically include noise removal, image segmentation, cropping, scaling, and more. The recognition system first accepts a scanned image as an input. The images can be in JPG or BMT format. Digital capture and conversion of an image often introduces noise, which makes it hard to identify what is actually a part of the object of interest. Considering the problem of character recognition, we want to reduce as much noise as possible, while preserving the strokes of the characters, since they are important for correct classification. In the segmentation stage, a sequence of characters is segmented into a sub-image of an individual character.

3. Implementation

A. OpenCV

OpenCV stands for Open Source Computer Vision Library. It is the most popular library used for Computer vision using Python. OpenCV is free for everyone to use, hence OpenCV is used for a wide variety of projects and applications. Machine learning and images have a great relationship; the image classification has been one of the main roles of machine learning over the years.

Using OpenCV, we can perform a number of tasks, like,

1. Read Image: There are a number of such color spaces in which images exist — Grayscale, RGB, HSV, CMYK, etc. You can imagine how computationally intensive things would get once the images reach dimensions, say 8K (7680×4320). The role of the ConvNet is to reduce the images into a form which is easier to process, without losing features which are critical for getting a good prediction.
2. Image processing and feature detection: The convolutional layer maps the image to a matrix format, the pooling layer reduces the dimensions for faster processing time and the flattening converts the image into a linear array format for feeding it into the neural network. Then, the features are extracted from the image. Then, the system is supplied with a training set to learn from it. Text extraction is implemented using Tesseract OCR package which contains an optical character recognition (OCR) engine - libtesseract and a command line program – Tesseract. Tesseract includes a new neural net called Long Short-Term Memory (LSTM) based OCR engine, which focuses on line recognition and also recognizes character pattern. This tool is a wrapper for Google's Tesseract-OCR Engine, where it reads all image types including jpeg, png, gif, bmp, tiff, and others. These images are supported by the Python Imaging Library. Once the text data from the image files is extracted using Python-Tesseract, the required credentials are then stored in the database for easy access.
3. Capture images and save: OpenCV is a library which processes the digital images, therefore we need to store these images for processing. The Mat class of OpenCV library is used to store the values of an image. It represents an n-dimensional array and is used to store image data of grayscale or color images, voxel volumes, vector fields, point clouds, tensors, histograms, etc.

B. CNN Algorithm

A Convolutional Neural Network is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the ability to learn these filters/characteristics. In CNN, pre-processing comprises the convolutional layer, pooling layer and flattening. The convolutional layer maps the image to a matrix format, the pooling layer reduces the dimensions for faster processing time and the flattening converts the image into a linear array format for feeding it into the neural network. Then, the features are extracted from the image. Then, the

system is supplied with a training set to learn from it. Finally, the extracted features of the image are compared to that of the training set images and is classified into various layers that goes in sequence such as the input, fully connected, hidden and output based on the match percentage.

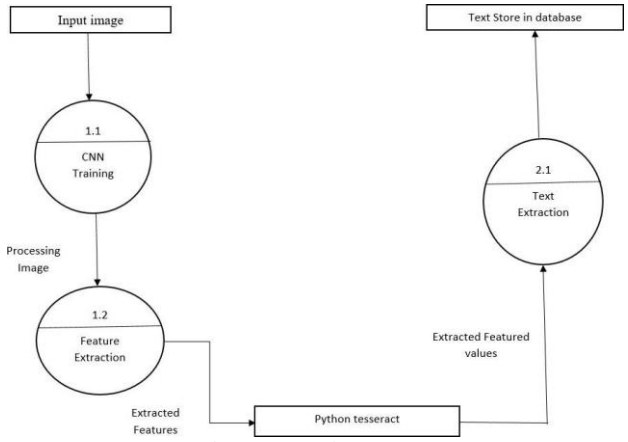

Fig. 1. Data flow diagram for the proposed system

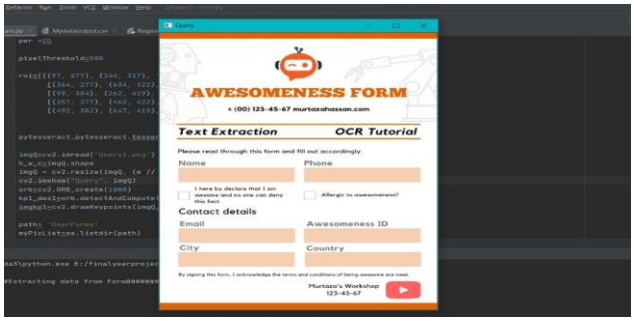

Fig. 2. Home page used in the system

4. Results, Discussion and Conclusion

The major contribution of this system is that it the proposed system classifies the image and then extracts the text data from them. Image Classification is done using the Convolutional Neural Network, which classifies images based on feature extraction. Text extraction is done using Python-Tesseract OCR package. The extracted text is then stored in the database. The proposed system gives better performance compared to the existing systems, based on accuracy, since CNN overcomes the problem of overfitting. The proposed system can be further improved by developing it into a full stack application, where a user interface is designed for the user to upload the image files into the system. The proposed system can be further improved by developing it into a full stack application, where a user interface is designed for the user to upload the image files into the system.

5. Future Work

In future Recognition accuracy can be increased by expanding the knowledge data base by including more images having different expressions. Android based implementation can be implemented to get the real time results. The different algorithm can also be used to improve the accuracy.

References

[1] Ned Horning, “Random Forests: An algorithm for image classification and generation of continuous fields data sets”.

[2] R. Smith, "An Overview of the Tesseract OCR Engine," in Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), Parana.

[3] Venkata Naresh Mandhalai, V. Sujatha, B. Renuka Devi, “Scene Classification Using Support Vector Machines,” in IEEE International Conference on Advanced Communication Control and Computing Technologies (lCACCCT), 2014.

[4] Kouzani, A. Z, Nahavandi, S, Khoshmanesh, “Face classification by a random forest,” in K. TENCON 2007 - 2007 IEEE Region 10 Conference. Deakin Univ., Geelong (October 30-November 2, 2007).

[5] N. Debachandra Singh, “Binary Face Image Recognition using Logistic Regression and Neural Network” - International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), 2017.

[6] Gayathri Devi. G, C. P. Sumathi, “Text Extraction from Images using Gamma Correction Method and different Text Extraction Methods – A Comparative Analysis”- International Conference on Information Communication and Embedded Systems (ICICES), 2014.

[7] Anubhav Kumar, Neeta Awasthi, “An efficient algorithm for text localization and extraction in complex video text images,” 2nd International Conference on Information Management in the Knowledge Economy, 2013.