IRSTI 50.10.01

M. Sovet[1]
[1]Suleyman Demirel University, Kaskelen, Kazakhstan

CHEATING RECOGNITION IN PAPER EXAMS USING CV

Abstract. With the shift from exams to electronic examinations, to pen and paper (paper exams), concerns were raised about whether this would make cheating easier. Cheating and academic dishonesty have always been disturbing practice in an academic setting, it kills the creativity of a student. Roughly speaking all teachers meet a high rate of academic dishonesty among their students. This article explores how teachers and students perceive differences in the ease of cheating during written exams, especially paper exams. Nowadays we have control systems that detect cheating and abnormal behaviors during exams. Despite early controls determining cheating during the checking of exam papers is also a great idea. Manually checking each work will take up most of the time and energy, which is also difficult to identify plagiarism. That's why the paper gives using Computer Vision to optimize checking paper exams and detect cheating levels among students.

Keywords: Paper examinations, PyTesseract, OCR, text recognition, cheating, document image analysis (DIA).

***

Аннотация. С переходом от экзаменов к электронным экзаменам, к ручке и бумаге (бумажные экзамены) были высказаны опасения по поводу того, облегчит ли это мошенничество. Обман и академическая нечестность всегда были тревожной практикой в академической среде, она убивает творческий потенциал студента. Грубо говоря, все учителя сталкиваются с высоким уровнем академической нечестности среди своих учеников. В этой статье исследуется, как учителя и студенты воспринимают различия в легкости списывания во время письменных экзаменов, особенно бумажных экзаменов. В настоящее время у нас есть системы контроля, которые обнаруживают обман и ненормальное поведение во время экзаменов. Несмотря на ранний контроль, определить обман во время проверки экзаменационных работ также отличная идея. Ручная проверка каждой работы займет большую часть времени и энергии, что также трудно выявить плагиат. Вот почему статья дает возможность с помощью компьютерного зрения оптимизировать проверку бумажных экзаменов и выявить уровень обмана среди студентов.

Ключевые слова: Бумажный вариант экзамена, Тессеракт, распознавание текста, обман, анализ изображений документов.

***

Аңдатпа. Емтихандарды электронды емтихандарға және қалам мен қағазға (қағаз емтихандары) ауысқанда, бұл алаяқтықты жеңілдетеді ме деген алаңдаушылық туындады. Алдау және академиялық адалдық академиялық ортада әрдайым алаңдатарлық тәжірибе болды, ол студенттің шығармашылық жағынан дамуын тежейді. Шамамен айтқанда, барлық мұғалімдер өз студенттері арасында академиялық адалдықтың жоғары деңгейіне тап болады. Бұл мақалада мұғалімдер мен студенттердің жазбаша емтихандар, әсіресе қағаз емтихандары кезінде алдаудың қарапайымдылығындағы айырмашылықтарды қалай қабылдайтындығы қарастырылады. Қазіргі уақытта бізде емтихан кезінде алдау мен қалыптан тыс мінез-құлықты анықтайтын бақылау жүйелері бар. Ерте бақылауға қарамастан, емтихан жұмыстарын тексеру кезінде алдауды анықтау да өте жақсы идея. Әр жұмысты қолмен тексеру көп уақыт пен энергияны қажет етеді, сонымен қатар плагиатты анықтау қиын. Сондықтан мақала компьютерлік көру арқылы қағаз емтихандарын тексеруді оңтайландыруға және студенттер арасындағы алдау деңгейін анықтауға мүмкіндік береді.

Түйін сөздер: Емтиханның қағаз нұсқасы, Тессеракт, мәтінді тану, алдау, құжат кескінін талдау.

Introduction

During recent years, information and communication technologies (ICTs) have developed rapidly and have a direct impact on human existence, especially in the field of education. As a result, e-learning has become increasingly popular and widespread in educational institutions over the past few years. It allows you to provide information anytime and anywhere on the Internet when students need it. Because of this, it is also called web learning or online learning. “Learning assessment is the process of finding and interpreting evidence for use by students and their teachers to decide where students are in the learning process, where they need to go, and how best to get there” [1]. Assessment is one of the main tasks of the educational process. This is essential when designing any e-learning course. In educational testing and academic examination, the term cheating is used to denote all forms of illegitimate activities that are aimed at increasing one’s test performance. These activities comprise using unauthorized materials (e.g., calculators), resorting to additional information during an exam (e.g., via cheat sheets), answer copying, collusion among examinees, the acquisition of test questions or having another person take the test instead of oneself. Cheating affects the validity of examination in higher education and impairs all decisions that are based on test results. Cheating is frequent among students at high-schools and universities.

Cheating on exams was a broad phenomenon in the world, regardless of the level of development of detection. Over the past decade, many studies have

been conducted on student fraud and the means by which the university could try to combat this problem [2].

Nowadays, there is a growing demand for the software systems to recognize characters in computer when information is scanned through paper documents as we know that there are number of papers which are in printed format. Day by day due to atmospheric changes or due to improper handling they get damaged. Therefore, nowadays there is a huge demand in "storing the information available in these paper documents in to a computer storage disk and then later reusing this information by searching process". One simple way to store information in these paper documents in to computer system is to first scan the documents. Whenever we scan the documents through the scanner, the documents are stored as images in the computer system [3]. These images contain text that cannot be edited by the user. But to reuse this information it is very difficult for the computer system to read the individual contents and search the contents form these documents line-by-line and word-by-word.For automotive detecting differences of student exams this is a optimal method. It will decrease time for checking exams and keep time of teachers.

For research used main 2 projects with despite of their programming language. For example, this literature is very useful for my research, I took from here a lot of new things.

1. Image2Text Project.

Image2Text is a python application to grab text from images and save as text files using Google Tesseract Engine. Tesseract is an optical character recognition engine for various operating systems. The aim of this Repository is to be able to recognise text from an image file using the Tesseract Library in the Python programming language.

The application is a simple document image analysis using Python-OpenCV. The input folder contains forms that were pre-processed with given center of the circles. The circles should be classified in three different categories: shaded, not shaded, and crossed-out [4].

Methods

The main technologies which used in this paper is grab text from scanned image and comparing with scanned image text. There is popular library which used to read characters from images is PyTesseract OCR Engine Library in the Python language. Tesseract is an Open Source library for Optical Character recognition (OCR) [5]. We will be using PyTesseract to print the recognized text given an input image of any of the following formats: jpeg, png, gif, bmp, tiff, and others. Tesseract is compatible with many programming languages and frameworks through wrappers that can be found here. It can be used with the existing layout analysis to recognize text within a large document, or it can be used in conjunction with an external text detector to recognize text from an image of a single text line. Recognition then proceeds as a two-pass process. In the first pass, an attempt is made to recognize each word in turn. Each word that

is satisfactory is passed to an adaptive classifier as training data. The adaptive classifier then gets a chance to more accurately recognize text lower down the page [6].

Word and Line Finding with Tesseract

1) Line Finding

The line finding algorithm is one of the few parts of Tesseract. The line finding algorithm is designed so that a skewed page can be recognized without having to de-skew, thus saving loss of images quality. The key parts of the process are blob filtering and line construction [7]. Assuming that page layout analysis has already provided text regions of a roughly uniform text size, a simple percentile height filter removes drop-caps and vertically touching characters. The median height approximates the text size in the region, so it is safe to filter out blobs that are smaller than some fraction of the median height, being most likely punctuation, diacritical marks and noise. The filtered blobs are more likely to fit a model of non-overlapping, parallel, but sloping lines. Sorting and processing the blobs by x-coordinate makes it possible to assign blobs to a unique text line, while tracking the slope across the page, with greatly reduced danger of assigning to an incorrect text line in the presence of skew. Once the filtered blobs have been assigned to lines, a least median of squares fit [8] is used to estimate the baselines, and the filtered-out blobs are fitted back into the appropriate lines. The final step of the line creation process merges blobs that overlap by at least half horizontally, putting diacritical marks together with the correct base and correctly associating parts of some broken characters.

2) Baseline Fitting

Once the text lines have been found, the baselines are fitted more precisely using a quadratic spline. This was another first for an OCR system, and enabled Tesseract to handle pages with curved baselines , which are a common artifact in scanning, and not just at book bindings. The baselines are fitted by partitioning the blobs into groups with a reasonably continuous displacement for the original straight baseline. A quadratic spline is fitted to the most populous partition, by a least squares fit. The quadratic spline has the advantage that this calculation is reasonably stable, but the disadvantage that discontinuities can arise when multiple spline segments are required. A more traditional cubic spline [9] might work better.

3) Proportional Word Finding. Non-fixed-pitch or proportional text spacing is a highly non-trivial task. Fig. 2 illustrates some typical problems. The gap between the tens and units of ‘11.9%’ is a similar size to the general space, and is certainly larger than the kerned space between ‘erated’ and ‘junk’. There is no horizontal gap at all between the bounding boxes of ‘of’ and ‘financial’. Tesseract solves most of these problems by measuring gaps in a limited vertical range between the baseline and mean line. Spaces that are close to the threshold at this stage are made fuzzy, so that a final decision can be made after word recognition.

Fixed Pitch Detection and Chopping

Tesseract tests the text lines to determine whether they are fixed pitch. Where it finds fixed pitch text, Tesseract chops the words into characters using the pitch, and disables the chopper and associator on these words for the word recognition step. Fig. 3 shows a typical example of a fixed-pitch word.

Results

By summarizing opportunities of result, we get result like this:

- OCR doesn't do well with images affected by artifacts including partial occlusion, distorted perspective, and complex background;
- It is not capable of recognizing handwriting;
- It may find gibberish and report this as OCR output.It is not always good at analyzing the natural reading order of documents. For example, it may fail to recognize that a document contains two columns, and may try to join text across columns;
- Poor quality scans may produce poor quality OCR;
- It does not expose information about what font family text belongs to;
- As a research result of project PyTesseract and their methods can perfectly perform own works.

A man signing in at Google's main office, Googleplex.

Google Inc. is an American multinational corporation that is best known for running one of the largest search engines on the World Wide Web (WWW). Every day, 200 million (200,000,000) people use it. Google's main office ("Googleplex") is in Mountain View, California, USA.

With Google Search, people can also search for pictures, Usenet newsgroups, news, and things to buy online. By June 2004, Google had 4.28 billion web pages on its database, 880 million (880,000,000) pictures and 845 million (845,000,000) Usenet messages — six billion things.

Figure-1. Input Image

```
A man signing in at Google's main office, Googleplex.

Google Inc. is an American multinational corporation
that is best known for running one of the largest search
engines on the World Wide Web (WWW). Every day,
200 million (200,000,000) people use it. Google's main
office ("Googleplex") is in Mountain View, California,
USA.

With Google Search, people can also search for pictures,
Usenet newsgroups, news, and things to buy online. By
June 2004, Google had 4.28 billion web pages on its
database, 880 million (880,000,000) pictures and 845
million (845,000,000) Usenet messages – six billion
things.
```

Figure-2. Ouput Image

Discussion

By researching this field I decide to make own project using Computer Vision algorithms and Tesseract library.But there is also second more difficult analogue to grab text from images is to recognize an image containing a single character using a Convolutional Neural Network (CNN). Text of arbitrary length

is a sequence of characters, and such problems are solved using RNNs and LSTM is a popular form of RNN. Read this post to learn more about LSTM.

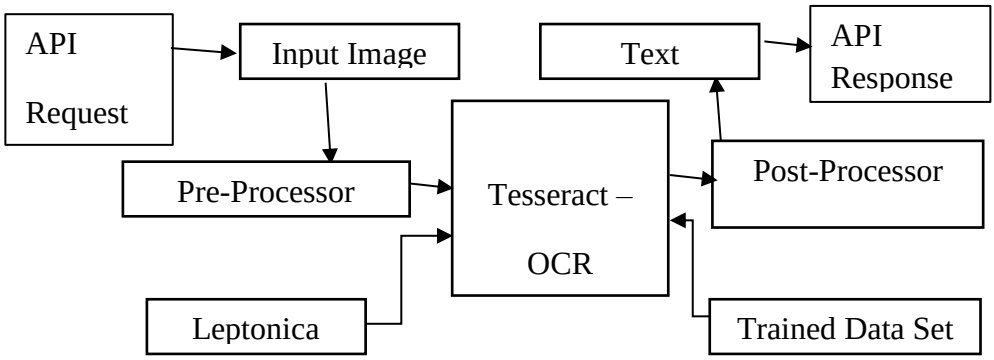

Figure-3. Working principle of Tesseract OCR Engine (OCR Process Flow)

Long Short Term Memory networks – usually just called "LSTMs" – are a special kind of RNN, capable of learning long-term dependencies. They were introduced by Hochreiter & Schmidhuber (1997), and were refined and popularized by many people in following work.They work tremendously well on a large variety of problems, and are now widely used.

LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior, not something they struggle to learn!

To this place I have gave main attention to algorithms and technologies that perform grabbing text from image.And the second main part of project is doing web application that shows result of chating level. For those purposes I prefer to use Python language, because whole semester I am learning a Python language, and his libraries. All projects which I researched they are useful for my projects. At this moment I am learning deep learning,convolutional and recurrent neural networks to realizing LSTM algorithm on my project.

As a significant prior research I will work on multi sheet scanners, but I researched it watching tutorials, recognizing their work principles. I hope that next semester I will find or take from you, and practice with them. Also I should create web view for showing the result. For it I told that I will use Laravel and VueJs.

References

1 Assessment Reform Group, "Assessment for Learning: 10 Principles", 2002.URL: https://assessmentreformgroup.files.wordpress.com/2012/01 /10principles _english.pdf> [Accessed: 23- Mar- 2017].

2 Curran, K., Middleton, G., Doherty, C. Cheating in exams with technology, 2011. URL:

https://pdfs.semanticscholar.org/1ba7/bc7b96f0bbc3ecbbcd958f9bd755852c1c02.pdf [Accessed: 23- Mar- 2017].
3 Gaurav, K., Cheating Detection in online examinations. San Jose State University, May 2015. – 59 p.
4 Zelic, F., Sable, A. A comprehensive guide to OCR with Tesseract, OpenCV and Python. URL: https://nanonets.com/blog/ocr-with-tesseract/.
5 Smith, R. An Overview of the Tesseract OCR Engine, Google Inc. 1-5 pp.
6 Valke, A., Lobov, D. Character recognition algorithms. Omsk State Technical University. *Journal of Physics Conference Series.* (2019): pp. 1-5.
7 Muchhadiya, P. The different image processing techniques to text from natural images. 1 (3), (2014): pp. 1-7.
8 Rousseeuw, P.J., Leroy, A.M. *Robust Regression and Outlier Detection*, Wiley-IEEE, 2003. – 329 p.
9 Schneider, P.J. An Algorithm for Automatically Fitting Digitized Curves", in A.S. Glassner, *Graphics Gems I,* Morgan Kaufmann, (1990): pp. 612-626.