Recognizing Call Numbers for Library Books: The Problem and Issues

Hanh Pham
Computer Science Department
State University Of New York
New Paltz, USA
phamh@newpaltz.edu

André Dvorak
Computer Science Department
State University Of New York
New Paltz, USA
dvoraka1@newpaltz.edu

Miguel Perez
Computer Science Department
State University Of New York
New Paltz, USA
perezm28@newpaltz.edu

***Abstract*—This paper addresses a challenging and practical problem in image processing and pattern recognition where we need to recognize book call numbers, or book IDs using the Library of Congress format, in images of book spines. For this special case of Optical Character Recognition (OCR) we review and evaluate a number of standard and specialized OCR systems. Our experiment results and analysis show that the available OCR systems are not effective enough for recognizing book call numbers on the book spines as library labels are very small and may be worn out while the symbols printed on them are very thin. Since the recognition success rates of current OCR systems are not acceptable to be used in any meaningful application, a better solution is necessary.**

Keywords—image processing, pattern recognition, OCR, identifying books, library labels, book call number.

I. INTRODUCTION

One of the most popular problems in image processing and pattern recognition is Optical Character Recognition (OCR) where we need to recognize symbols in images. This problem has been there for such a long time that it seems that any OCR-related problem would not be an issue any more. However, we find that even well-known OCR systems do not work well when applied to a special case of OCR: recognizing book call numbers in the Library of Congress (LC) format. The goal of this paper is to define the challenges of this specific problem and evaluate current OCR systems which can be used to solve it.

The problem of recognizing LC book call numbers in labels on library book spines comes from the need of automating the book search or detection of misplaced books in big libraries. Each book in any US big library has an ID that is created using the rules defined by LC. This book ID is also called "book call number" and has a specific format that can be understood by human users. Libraries develop plans, based on this call number and the LC rules, to define where (which bookshelf and which room or floor) each book should be placed. This placement for the given book must be strictly followed so that anyone who knows the LC rules can allocate the book using the book call number. The book allocation takes place when a library user first looks up for a book in the catalog and then wants to read or borrow the book. Next, we would need to know the book current location. The book call number would displayed in the catalog search results. Using the LC rules this call number should lead us to its supposed-to-be location in the library. Thus, we would need to make sure that the given book must reside at that particular place. However, in big libraries most users have direct and unlimited access to the shelves and the books themselves. They can easily and often misplace the books when they try to put them back after taking out to have a look at. Unfortunately, most of these book misplacements may not be detected by anyone until someone else needs the book, goes to where it is supposed to be but cannot find it. Therefore, the library would need to have enough staffs to walk around the bookshelves and check each individual book to see if it is in the right place. Since human labor is very costly, manual book location checking can only be done as often as a library can afford.

There have been attempts in using new technologies such as RFID [5] or bar codes [11] to identify books. However, that would require re-labeling every single book, purchase of RFID or barcode readers, antennas, and software license which would lead to very high initial and maintaining costs. Besides, RFID tags and barcodes are impossible to be read by human staffs or users in case the reading equipment or system fails. Therefore, there are demands for low-cost automated systems which can detect book misplacements such as the one proposed in [9]. In these kinds of systems, one of the key problems is to recognize the call number in the label on each library book spine.

At first it appears as this problem is already solved as OCR has been studied for a long time and gained significant success. However, in this paper, throughout a number of experiments we show that current OCR systems do not produce satisfied results in this specific situation. We discover many underline issues which cause the poor results, such as many library labels fade away or worn out especially when the background of the labels are white and the symbols printed on them are very thin. Another example of issues we face is the uneven lighting in the libraries around the bookshelves which often creates un-uniform brightness, in some cases, it is not bright enough to read what is on the old labels. These issues point to the need of studying this problem more thoroughly and analyzing the performance of the available OCR systems more carefully before developing new and more effective algorithms for this particular book call number recognition problem. This paper is to address just that.

In this paper, we conduct experiments where the performance of a number of OCR algorithms will be measured and analyzed in different situations due to the issues and challenges in the given problem. Our paper is organized as

978-1-7281-1360-9/18/$31.00 ©2018 IEEE
DOI 10.1109/CSCI46756.2018.00079

Authorized licensed use limited to: National Chung Hsing Univ.. Downloaded on October 16,2023 at 07:37:46 UTC from IEEE Xplore. Restrictions apply.

follows. Section 1 briefly describes the necessity for analyzing this special case of OCR for library application. The given specialized OCR problem is defined in section 2. Related work is reviewed in section 3. We describe our system which is built for testing the current OCR software in section 4. The experiments and our analysis are presented in section 5. Conclusions and future work are provided in section 6.

II. The Problem of Recognizing Call Numbers in Library Labels

A. Problem Description

Given an image of book spines on library bookshelves. Each book should have a library label or sticker on its spine as shown in Fig.1. It contains the book call number, usually on a white background. We need to have an OCR software that can extract the book call number or book ID in each label for all books in the given image.

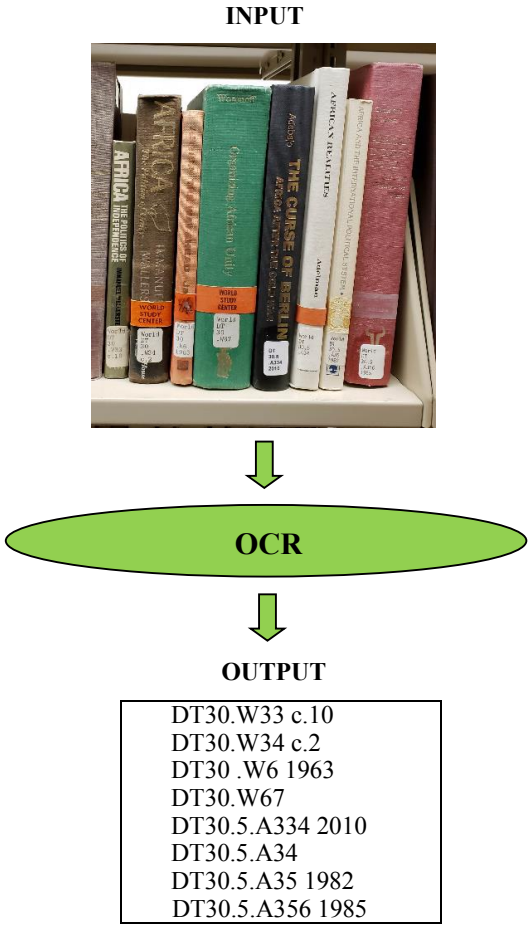


Fig. 1. Recognizing Call Numbers in Library Labels

This problem is a special case of the general OCR problem where we need to recognize texts and symbols in images. At first, it seems to be simple and probably is already solved as the general OCR problem has been there for a long time and we already have many satisfactory OCR applications. However, as our experiment results show in section 5 later, this is still a quite challenging problem with hidden issues.

B. Parameters

In an OCR problem the images or input quality plays a big role in the outcome success. For this particular OCR problem we need to examine various situations and the environment when and where the images of book spines are obtained. In other words, we need to identify which parameters may have effects on the recognition success rate in solving the given special OCR problem for book call numbers.

1) Spine Background: The color of book spines may make a big differnce in recognizing texts and symbols of the book call numbers on the book labels which usually have white background. Would it be easier to do OCR when the white labels are on top of a dark spine background? as shown in Fig.2. Would it be more difficult to extract the frame of the white labels out of the whole image when the spine background itself is also bright or white? as shown in Fig.3.

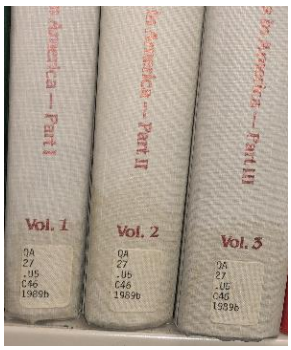


Fig. 2. Bright Spine Background

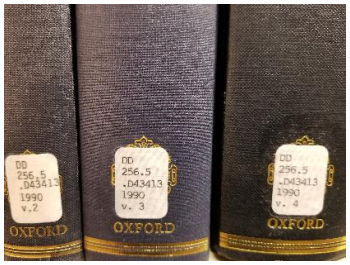


Fig. 3. Dark Spine Background

To answer these questions we will conduct experiments with 5 different levels of the darkness/brightness for the spine background, as shown in Fig.4. The experiment results and analysis are reported in section 5.

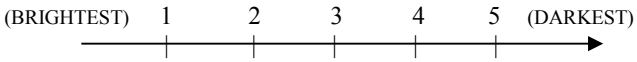


Fig. 4. Range of Darkness Levels for Spine Background

Authorized licensed use limited to: National Chung Hsing Univ.. Downloaded on October 16,2023 at 07:37:46 UTC from IEEE Xplore. Restrictions apply.

2) Label Quality: The book call number is printed on a label on its spine. The label background color is usually white and the call number, which is a mix of letters and numbers with dots in between, is printed so thinly that it looks grey rather than black. The labels are first made at the time when the books are added to the library inventory. Through the years many labels may wear off as they are partialy pilled off and the ink fades away. Unfortunately, they are not immediately replaced with new ones as it takes a long time to detect, print and replace labels. Therfore, the book labels we see in the library may vary from very clear ones with dark black ink to fuzzy ones where the symbols for the book call numbers are blurred and not clear any more, as shown in Fig. 5. and Fig.6.



Fig. 5. Old and Worn Out Labels



Fig. 6. New and Clear Labels

Although the library labels can be in various states it is difficult to quantify the clearness or fuzziness. Therefore, we categorize images into only three levels: 1 for fuzzy or worst, 2 for medium, and 3 for clear or best labels, as shown in Fig.7.

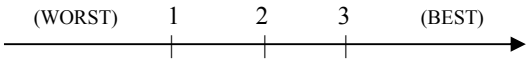


Fig. 7. Range of Levels for Label Quality

3) Height of Cameras (Lighting): As we all know, lighting can have big effects on OCR performance. The difference in the light intensity may be difficult to be detected by humen eyes but for the OCR software it may lead to a big difference in the recognition success rate. In the library, the main source of light usually comes from the ceiling. Therefore, the higher the camera position is the brighter and clearer the image would be. Vice verse, the lower the camera is the darker the image would be. The easiest way to get images with different light strengths is to position the camera to different shelves, from high to low ones, of the library bookshelf. We can get images with a lot more light for the top level of the bookshelf, as shown in Fig.8. or images with much less light for the bottom level of the bookshelf, as shown in Fig.9.

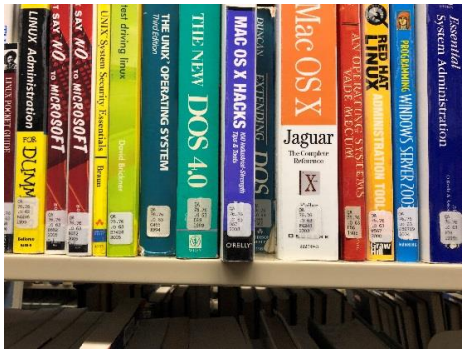


Fig. 8. Top Shelf/Level with More Light



Fig. 9. Bottom Shelf/Level with Less Light

Since the height of the camera position can be easily quantified by the shells of the bookcase we use the number of possible shell levels in a library book shell, usually 8, for the range of light strength. The lowest value in this range for light strength would be when the camera is at the level of the bottom shell. Similarly, the highest value in this range for light strength would be when the camera is at the level of the top shell, as shown in Fig.10.

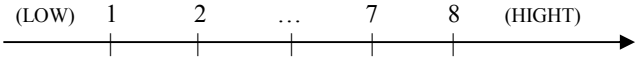


Fig. 10. Range of Levels for Light Strength or Camera Height

III. Related Work

There have been many OCR algorithms developed over the years for recognizing texts. They can be (i) general and universal for all kinds of texts and backgrounds [7,12,13,20] or (ii) specialized on specific problems and situations [2,14,18,21]. Because of the lack of OCR specifically designed for our problem: recognizing call numbers on library book labels, we will review all possibly-related OCR text-based systems. They include the general text-based OCR, the systems for book identifications based on book spines and covers, and specialized OCR for the kinds of problems which are similar to our problem.

Authorized licensed use limited to: National Chung Hsing Univ.. Downloaded on October 16,2023 at 07:37:46 UTC from IEEE Xplore. Restrictions apply.

A. Standard OCR systems

There are many OCR systems which can extract texts from images. However, according to TechRadar.com, the current most popular and best systems are:

1. Google
2. Nuance
3. Abbyy
4. Adobe
5. (Canon)Readiris

The Google system is available in different versions: Google Drive OCR, Google Lens and Google Cloud Vision, where you can download the program and run on a PC. Nuance also has a number of various systems with a focus on office documents and general pictures taken by phones. Abbyy also focuses on office use but has a version called Abbyy Mobile OCR Engine for recognizing texts in photos. Acrobat Reader by Adobe can also extract texts from a scanned PDF-based image. Readiris is a Windows-based OCR system by Canon which lets you extract texts from scanned images. All of these top OCR systems are developed by well-known companies. Therefore, it would be worthwhile to run some tests with them. Another general OCR software is the classical Tesseract[12] initially developed by HP and is now available in OpenCV.

B. OCR for Book Identification

Recently, there have been a number of OCR systems developed for identifying books based on pictures of book covers or book spines. The main factor leading to this boom is the rapid improvement and popularity of phone cameras. Most of these phone applications are based on the standard OCR listed above.

1) OCR for Book Covers: There are several OCR specialized on book covers although they are usually a part of a general OCR family which recognizes texts on book pages or scanned documents[2,6,19].

2) OCR for Book Spines: At first, it seems that this family of OCR could solve our problem. Unfortunately, they focus only on the book biographic information such as titles, authors and publishers[1,3,4,8,10,15,16,17]. CLARA[9] is a very rare system which focuses solely on recognizing the call numbers.

Both of these OCR families, except [9], are designed only to pick up the book main biographic information and then use it to look up in the book catalogs, usually are cloud-based such as Amazon's or Barnes & Noble's ones, to get the full information about the given book. That is different from our problem where we must identify a specific copy of a book that the library owns. Although we can still use these kinds of systems to look up in the library catalog and guess the call number of the given copy of the book, we would waste a lot of computing power and network traffic searching. It would be much more economical and precise to recognize the book call number instead.

C. Systems for Similar OCR Problems

There are specialized OCR systems for the kinds of problems which are similar to our problem, from the technical point of view. Such OCR systems include recognizing texts or symbols on a white or fixed color of the background. For examples, among the similar problems there are License Plate recognition, Receipt recognition and Check recognition problems. Especially the license plate recognition appears to be very similar to our problem: recognizing book call numbers on library book labels, as shown in Fig. 11.



Fig. 11. Similarity between the "license plate" and the "library book call number" recognition problems.

IV. Our Testing System

In order to measure the performance of OCR software which can be used for our book call number recognition problem we put together a testing system which integrates several OCR, selected from section 3, and run them with the same set of input data, as shown in Fig.12.

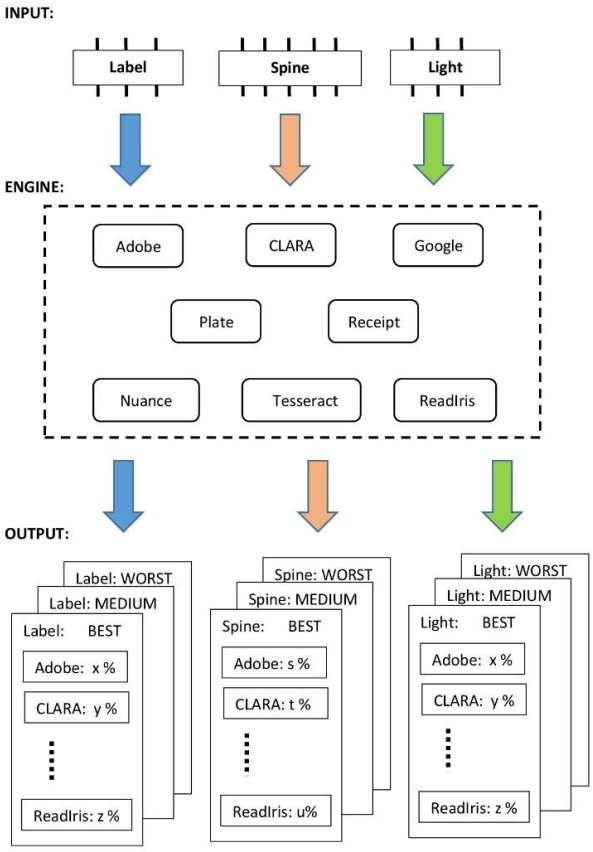


Fig. 12. Testing System for the Problem of Recognizing Book Call Numbers

Authorized licensed use limited to: National Chung Hsing Univ.. Downloaded on October 16,2023 at 07:37:46 UTC from IEEE Xplore. Restrictions apply.

A. The Engine

We include in the engine of our test system the following OCRs where we have representatives from all groups described in the previous section:

- Adobe Acrobat
- CLARA
- Google Cloud Vision
- License Plate ALPR
- Nuance OmniPage
- ReadIris
- Tagun Receipt
- Tesseract

B. The Input

The same input is used for all OCR in the engine. They are pictures taken with different values for all of the parameters described in section 2. They represent various situations and conditions of the environment where our problem of recognizing book call numbers may take place. The input data can be organized in three groups:

- Spine Background
- Label Quality
- Lighting (or Camera Height)

C. The Output

The output of processing each picture or image of book spines with library labels is a list of the book call numbers in a text file. We then score the performance of each OCR system for each parameter by comparing the results in the text files and the book call numbers in the corresponding pictures.

V. Experiments & Result Analysis

We use two different kinds of cameras with different resolutions: (i) a very basic web cam with 1.3 MP and (ii) a high-end smart phone camera with 12 MP. As the given problem needs to be solved in big libraries with a number of robots the high-end cameras may too expensive and therefore may not be available. Thus, it would be useful to see the effects of camera quality on the performance of each OCR software. Using the testing system described above we have conducted the following three experiments.

A. General Test

First, we carry out a simple test for all OCR systems using the same images which has the book labels of the best quality taken by the phone camera (i.e. the best input data you can have). The obtained results show us that only Google, Tesseract, Nuance, and CLARA delivered meaningful results while ReadIris, Tagun Receipt, and especially and surprisingly License Plate ALPR recognized almost nothing (that is despise the fact that it is designed to solve a problem that is very similar to our problem). Adobe Acrobat did recognize the labels, however it could not isolate and export the call numbers out of the images and therefore would be useless for our problem. Therefore, in further experiments we put License Plate ALPR, ReadIris, Tagun Receipt, and Adobe Acrobat into one group under the name "Plate and Others".

B. Compare Performances of All OCR

Next, we measure the success rates for all OCR systems using different sets of input images with the parameter ranges as described in section 2. The success rate is defined by a score from 1 to 10 with 10 is the best or recognizing 100% labels or all call numbers correctly and 1 is for the worst performance when an OCR cannot recognize more than 10% of the call numbers. The results are shown in Fig. 13, Fig. 14 and Fig.15. All images used in this experiment are taken by phone camera.

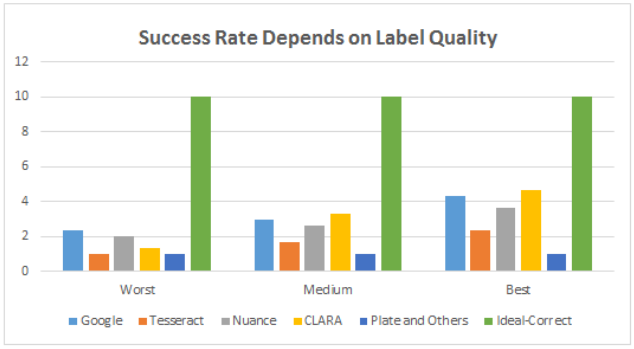


Fig. 13. Effects of Label Quality

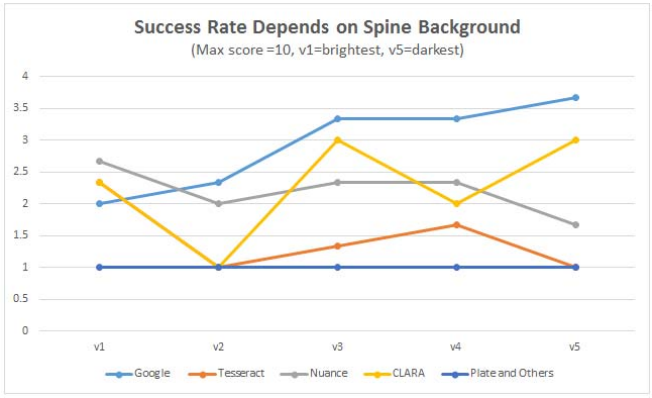


Fig. 14. Effects of Spine Background

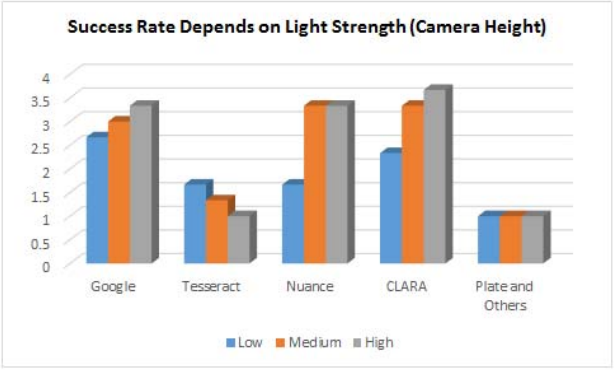


Fig. 15. Effects of Lighting or Camera Height

Authorized licensed use limited to: National Chung Hsing Univ.. Downloaded on October 16,2023 at 07:37:46 UTC from IEEE Xplore. Restrictions apply.

C. Compare the Effects of Image or Camera Quality

Finally, we use both webcam and phone camera to take a number of pictures of the same sets of book spines, with the parameters change within the ranges described in section 2, and then run all OCR systems to see how each of the OCR systems perform, depending on the kinds of cameras.

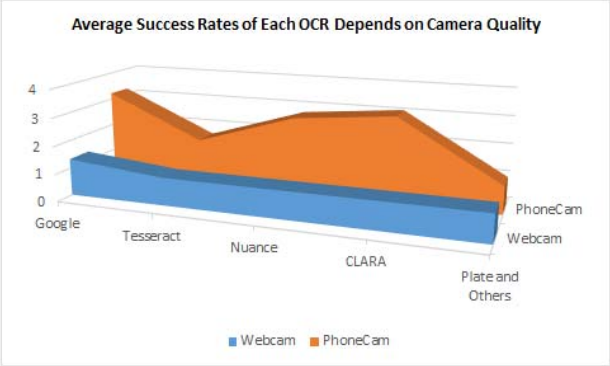


Fig. 16. Effects of Camera Quality

The experiment results show us that the quality of images or cameras have a significant effect on the performance of all OCR software. With a bad camera all OCR systems fail. Therefore, it is essential to invest and obtain the best cameras, despite the cost, for the OCR system specialized on recognizing library book call numbers.

VI. Conclusions

We have shown that the problem of recognizing book call numbers, or book ID in the LC format, in images of library book spines, is a very challenging OCR problem. Even with the current advanced OCR systems and technologies from top IT companies, the success rates in recognizing call numbers in images of book labels on their spines, taken by the currently best high-end phone cameras, are only around 40% or less which is not acceptable. Given that a good solution for this specialized OCR problem would lead to a very low-cost automated book location tracking system for big libraries, without the need of adding any new tags to the books or require any extra human labor, it would be very important that this problem is known to and receive attention and help from the research community. In the future, machine-learning mechanisms such as neural networks and fuzzy logics can be added to our current solution [9] to achieve better results. More study on any other factors which may affect the outcomes in this specific OCR problem can also help to develop more effective solutions for this very interesting and challenging problem.

References

[1] Chen, D. M.; Tsai, S. S.; Girod, B.; Hsu, C.-H.; Kim, K.-H.; and Singh, J. P, "Building book inventories using smartphones", In Proceedings of the 18th ACM international conference on Multimedia, 651–654. 2010, ACM.

[2] Hua Yang, N. Onda, M. Kashimura and S. Ozawa, "Extraction of bibliography information based on image of book cover," Proceedings 10th International Conference on Image Analysis and Processing, Venice, Italy, pp. 921-926, 1999.

[3] Lao Y., Lu J. and Wu M. "Mobile Phone Book Finder", Electrical Engineering Department, Standford UniversityTechnical Report. 2012.

[4] Lee, D.-J.; Chang, Y.; Archibald, J. K.; and Pitzak, C. 2008. Matching book-spine images for library shelf-reading process automation. In Automation Science and Engineering, 2008. CASE 2008. IEEE International Conference on, 738–743. IEEE.

[5] Li, R., Huang, Z., Kurniawan, E. & Ho, C. K. AuRoSS: an autonomous robotic shelf scanning system. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Sept 28 – Oct 2, 2015, Hamburg Germany. pp. 6100–6105.

[6] Meng G., N. Zheng, S. Du, Y. Song and Y. Zhang, "Shading extraction and correction for scanned book images," in IEEE Signal Processing Letters, vol. 15, pp. 849-852, 2008.

[7] Neumann, L., and Matas, J. 2012. "Real-time scene text localization and recognition". In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, 3538–3545. IEEE.

[8] Nevetha, M., and Baskar, A. "Automatic book spine extraction and recognition for library inventory management", In Proceedings of the Third International Symposium on Women in Computing and Informatics, 44–48, 2015, ACM.

[9] Pham, H., Giordano A., Miller L., Giannitti J., Mena M., DiNardi A., "A Ubiquitous Approach for Automated Library Book Location Management", International Conference on Computing and Big Data, September 08-10, 2018, ACM, 2018.

[10] Quoc, N.-H., and Choi, W.-H. 2009. "A framework for recognition books on bookshelves". In Emerging Intelligent Computing Technology and Applications. Springer. 386–395.

[11] Singh, M. "A snapshot of Traditional Library to Modern Library Using Barcode Technology: An Experience of Central LibraryGNDEC, Ludhiana", Journal of Advances in Library and Information Science 3(2): 146–149. 2014.

[12] Smith, R. 2007. An overview of the tesseract ocr engine. In icdar, 629–633. IEEE.

[13] T. Kasar, J. Kumar, and A. Ramakrishnan. "Font and background color independent text binarization". In Second international workshop on camera-based document analysis and recognition, pages 3--9, 2007.

[14] Tabassum N., S. Chowdhury, M. K. Hossen and S. U. Mondal, "An approach to recognize book title from multi-cell bookshelf images," 2017 IEEE International Conference on Imaging, Vision & Pattern Recognition (icIVPR), Dhaka, 2017, pp. 1-6.

[15] Taira P., S. Uchida, and H. Sakoe. "Book boundary detection from bookshelf image based on model fitting". In International Symposium on Information Science and Electrical Engineering, pages 601—604, 2003.

[16] Talker L. and Y. Moses. "Viewpoint-independent book spine segmentation". In Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on, pages 453--460. IEEE, 2014.

[17] Tsai S., Chen, D. Chen, H. Hsu, Kim, K. Singh, J. and Girod, B. "Combining Image and Text Features: A Hybrid Approach to Mobile Book Spine Recognition". Proceeding of MM '11 Proceedings of the 19th ACM international conference on Multimedia, Pages 1029-1032, Scottsdale, Arizona, USA, November 28 - December 01, 2011, ACM.

[18] Wang, K.; Babenko, B.; and Belongie, S. 2011. "End-to-end scene text recognition". In Computer Vision (ICCV), 2011 IEEE International Conference on, 1457–1464. IEEE.

[19] Windisch G., M. Mihályi and Z. Vamossy, "Building book catalog using image processing," IEEE 15th International Symposium on Computational Intelligence and Informatics (CINTI), pp. 231-235, 2014.

[20] Xianggrong Chen and Alan Yuille, "Detecting and reading text in natural scenes", In Computer Vision and Pattern Recognition, Proceedings of the 2004 IEEE Computer Society Conference on, volume 2, II–366. 2004, IEEE.

[21] Xunrui Duan, Qingjie Zhao and S. Anwar, "Identifying books in library using line segment detector and contour clustering," 2012 7th International Conference on Computing and Convergence Technology (ICCCT), Seoul, pp. 998-100, 2012, IEEE.

Authorized licensed use limited to: National Chung Hsing Univ.. Downloaded on October 16,2023 at 07:37:46 UTC from IEEE Xplore. Restrictions apply.