A Technique to Detect Books from Library Bookshelf Image

Mohammad Imrul Jubair*, Prianka Banik**
* Department of CSE, Ahsanullah University of Science and Technology, Bangladesh
** Department of CSE, BGC Trust University Bangladesh
mister_jubair@yahoo.com, tithy_barta@yahoo.com

***Abstract*—In this paper, we present a technique to recognize books from library bookshelf images. The proposed technique includes several modules such as- noise removal, bookshelf row detection and book detection and extraction from bookshelf. Firstly, the image is filtered for noise removal and edges are detected. The proposed technique counts the number of pixels of every candidate for horizontal lines on the edge detected image and compares the value with a threshold. Elected lines are extended which creates some horizontal lines segmenting the image into different portions. These individual portions are considered as bookshelf row regions of the bookshelf and each of them is passed into the book detection module for further processing. Applying vertical line detection in the manner of horizontal line detection method on the segmented regions is not adequate to detect a book from shelf row because of the angular orientation of book arrangements. The proposed technique detects books arranged in different angular orientation on a shelf and differentiates between the book and non-book components based on some properties of the book shape. Experimental results show that the proposed technique is capable of extracting bookshelf rows and detecting individual books from library bookshelf image successfully.**

I. Introduction

A well-stocked library is an asset to the school, college, university or any institution. With the enormous growth of the number of books in libraries, some book related activities where human beings are directly involved such as arranging, finding, storing books, etc., are becoming more and more exhausting and time consuming day by day. In this case, intelligent robot can be a proper solution where robots can help human beings in library related works. For an example, robots can capture an image of a bookshelf and individual books can be extracted from that image to obtain the positions of individual books on several shelves. If we consider this scenario, long term research in the field of computer vision is needed to implement the book recognition method into the intelligent robot system. In many recent works, methods for book title extraction and recognition is developed where detecting books and extracting them from bookshelf are vital modules. In this paper, we focus on detecting books which is a great challenge in such research area.

Analyzing bookshelves in library provides several features and properties of shelves and books. Bookshelves are divided into several shelf rows and these rows are horizontally rectangular in shape. Books arranged on bookshelf rows are also rectangular in shape and they are kept in vertical orientation generally. Book region detection procedure becomes more complicated when multiple shelves exist within the image boundary and books are arranged in different orientation. In most of the cases, books are not kept vertically, or the orientation of books changes after removing one or more books from the same row. Fig. 1 shows examples of multiple shelves and arrangement of books arranged in different orientation.

Several researches on book detection technique have been presented yet where authors have proposed different methods to extract books from bookshelf image. In [1], [2] and [3] boundaries of books were taken into account in order to extract books. In these works, images were captured in a fixed manner intentionally so that every image contains single row within its boundary. These approaches can be unsuccessful in case of the images containing multiple rows. Book recognition approach based on book tags was presented in [4], where book tags were detected first to select among the candidates for book. This approach is inefficient when tags do not appear on the desired position on the books or unwanted objects printed on books create confusion. Besides, differentiating between tag and non-tag object can increase the computational time as well. An approach to detect books was presented in [5], where high frequency filtering was used to detect individual rows in an entire shelf. In that method, high pass Butterworth filter is applied on the

Figure 1. Here, (a) shows image of bookshelf containing single row. (b) shows image of bookshelf containing multiple rows. (c) shows vertically arranged books and (d) shows books arranged in different orientations (other than vertical).

978-1-4799-0063-3/13/$31.00 ©2013 IEEE

image and after the filtering operation, low frequency areas are rejected and the remaining high frequency regions were assumed to cover regions of rows. Books segmentation module in [5] is done through several sub-modules such as- Line segment, MSAC (m-estimator sample consensus [6]) based calculation of vertical dominant vanishing point (DVP) and Segment book candidates. The entire book region extraction procedure in that approach is quite a lengthy and complicated process. Implementation results show that, using high pass Butterworth filter is not sufficient in all the cases and sometimes it fails to detect every row in the shelf properly. Besides, [7] has presented a hybrid approach that combines a text-based spine recognition pipeline with an image feature-based spine recognition pipeline. The text within the book spine image is recognized and used as keywords to search a book spine text database. The image features of the book spine image are searched through a book spine image database. The search results of the two approaches are then carefully combined to form the final result [7]. In this approach, hybridization of several modules becomes error-prone and complicated and difficult to implement. Moreover, in all the above proposed approaches, the angular orientation in book arrangements was not considered while detecting books.

In this paper, we present a method to detect books efficiently from library bookshelf image. The proposed technique can perform satisfactorily in cases of multiple rows in the bookshelf and angular orientation of book arrangements.

II. PROPOSED TECHNIQUE

The proposed technique includes several modules. Firstly captured image is filterer using noise removing technique. After that, the filtered image is passed through two modules- bookshelf row extraction and book region detection.

A. *Noise Removal*

Existing noise removal techniques can be used to filter the captured image. Occurrence of impulse noise, such as salt-and-pepper noise, can create problem in the detection levels where binary conversion and edge detection is an important issue. In this circumstance, salt-and-pepper noise is removed using the proposed technique presented in [8]. Fig. 2 shows the result of applying noise removal technique on an input image.

B. *Bookshelf Row Detection*

The bookshelf row detection module includes some simple steps. As the horizontal line is one of the properties of a bookshelf, the proposed technique attempts to detect the existence of horizontal lines on the image. For this purpose, canny edge detection is performed firstly on the filtered image and small unwanted pixels are removed.

(a) (b)

Figure 2. (b) shows bookshelf image corrupted by noise and (b) shows filtered image.

After that, number of pixels of every candidate for horizontal lines are counted and compared with a threshold value. Candidates that reach the threshold are elected as horizontal line and they are extended up to image boundary. This method creates some horizontal lines on the image which segment the image into several regions. These regions are considered as bookshelf rows.

To illustrate the line detection operation, let's consider an image having m rows and n columns and imagine a horizontal line L_H on its edge detected image I_E. The coordinates of two end-points of that imaginary line are $A_1(a,1)$ and $B_1(a,n)$ where $1 \leq a \leq m$. I_E is a binary image where edge pixels are on (intensity value is 1) and other pixels are off (intensity value is 0). Our algorithm now counts the number of on pixels P_H lying exactly under the line L_H and compares the result with a threshold T_L, where T_L is determined as 70 percent of the total pixels underlying L_H. This comparison decides whether the current position of L_H can be elected as a shelf row line. If $P_H \geq T_L$ then all of the pixels lying under L_H are set to on. Otherwise, all of the pixels lying under L_H are set to off. In this manner L_H moves to the next position, where the new coordinates of two end-points are $A_2(a+1,1)$ and $B_2(a+1,n)$. At this stage, candidate for horizontal line at the current position is processed using the above method. If $P_H \geq T_L$ then all of the pixels of I_E lying under L_H are set to on. Otherwise all of the pixels lying under L_H are set to off. Fig. 3 shows an example of horizontal line detection method.

In the manner described above, to find out horizontal lines of bookshelf rows the whole image is needed to be scanned horizontally (e.g. top to bottom) where range of a is 1 to m. After scanning I_E is modified to a new image I_X containing some set of horizontal lines that indicate the presence of bookshelf rows. As the lines split I_X into several regions, cluster of some nearby horizontal lines can be combined together to get accurate regions. For this purpose morphological dilation operation [9] [10] is applied on I_X. After that, Connected Component Analysis (CCA) is applied to extract the individual regions from the input image separated by detected lines. In this step, shelf rows that partially exist in the image are rejected as they are out of our region of interest. For this purpose, height of bookshelf row is measured and shelf rows having comparatively low height are rejected. The remaining regions are selected as final shelf row regions. After scanning I_X is modified to a new image I_R containing some sets of horizontal lines that indicate the presence of bookshelf rows. Fig. 4 shows the results obtained from several steps in bookshelf row detection module.

	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
A1 (a,1)	0	1	1	1	1	0	1	1	1	0	0	1	1	1	1	B1 (a, n)
A2 (a+1,1)	1	0	0	0	0	0	1	1	0	0	1	0	0	0	0	B2 (a+1, n)
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

(a)

	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
A1 (a,1)	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	B1 (a, n)
A2 (a+1,1)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	B2 (a+1, n)
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

(b)

Figure 3. Horizontal line detection method for bookshelf row detection. Here, $T_L = 0.70 \times 15 = 10.5$. In (a), number of pixels underlying the line A_1B_1, $P_H = 11$. As $P_H \geq T_L$, every pixel underlying A_1B_1 is set to 1 which is shown in (b). Similarly, every pixel underlying A_2B_2 is set to 0 because for $A_2B_2, P_H < T_L$.

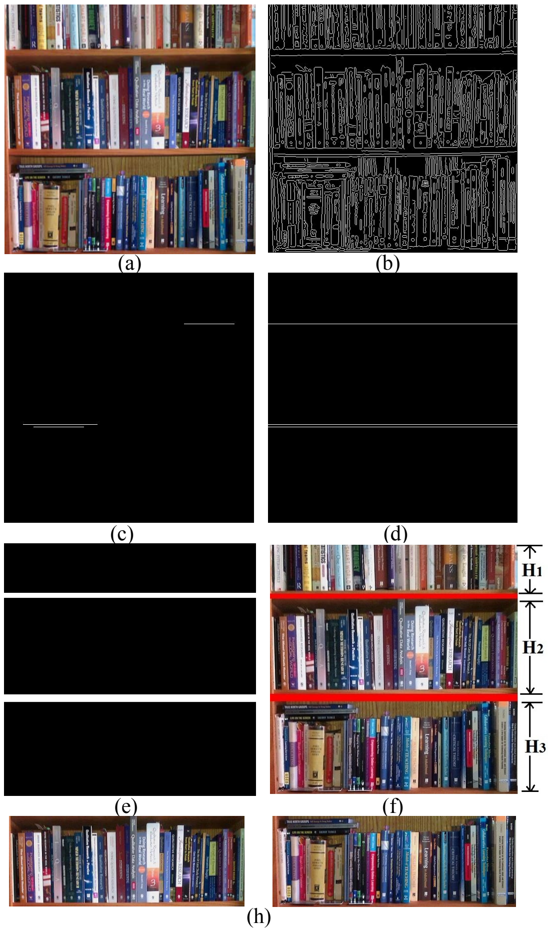

Figure 4. (a) is the input image, (b) is the image after canny edge detection, (c) shows the result after removing small and unwanted pixels, (d) is horizontal line detected image, (e) is the result after dilation operation, (f) shows shelf row detected regions(separated by red lines). Here three rows of height H_1, H_2 and H_3 are detected where the shelf row with H_1 is rejected. Shelf rows with height $H_2$and H_3 is selected and extracted which are shown in (h).

C. Book Detection

After extracting bookshelf rows from the bookshelf image, the extracted rows are passed to the book detection module. Extracted bookshelf rows contain book spines. The task of this module is to detect region of individual books from the book spines. Vertical line detection can be applied as like as the horizontal line detection by setting an imaginary vertical line on the image, but in this way, only books that are arranged vertically will be detected. Vertical arrangement means if we consider a bookshelf row as X axis then a book will make 90° with X axis. Problem arises when a book is not vertically arranged, that means the angle with X axis is not 90°. In case of the books which are not perpendicular to X axis, it is not possible to detect them using vertical line detection technique. Fig. 5 presents the scenario of vertical arrangement of books and non-vertical arrangement of books.

The proposed technique is capable of detecting books arranged in any angular orientation (making any angel with X axis). Firstly, the extracted bookshelf row image I_R from the previous module is filtered with canny edge detection and unwanted small pixels are removed. The concept of horizontal line detection is adopted here by setting an imaginary line L_θ on the image which makes an angel θ with X axis (horizontal shelf row). Now, L_θ slides vertically (e.g. left to right) over the image and counts the number of pixels P_θ underlying the line L_θ during each step. Every time whenever P_θ reaches a threshold $T_{L\theta}$ ($P_\theta \geq T_{L\theta}$), all of the pixels underlying L_θ are set to on (Where threshold $T_{L\theta}$ is determined as 70 percent of the total pixels underlying L_θ). Otherwise, all of the pixels underlying L_θ are set to off. In this process, books that make θ angel with a shelf row are detected as the candidates for book border. Once the bookshelf row image is scanned with L_θ, θ is incremented by 1 and a new imaginary line $L_{\theta+1}$ of angel $(\theta+1)$ rescans the I_R in the above way. Similarly, books that make $(\theta+1)$ angel with shelf row are detected. In the similar manner, books making $(\theta-1)$ angel can also be detected using the imaginary line $L_{\theta-1}$. The concept of the above procedure is illustrated in Fig. 6. In this way, shelf row image is processed to detect different angular lines using L_θ where $(90-x) \leq \theta \leq (90+x)$. To maintain optimal running time, $x = 30$ is considered and the increment of θ is set to 2 for each iteration, rather than 1. After scanning I_R is modified to a new image I_B containing some set of angular lines which indicate the regions of books in a shelf row. The lines in splits I_R into several regions, but these lines do not belong to individual books. Generally two consecutive books have a gap between them which is also detected. To avoid this, morphological closing [9] operation is applied on I_B for clustering the nearest neighboring lines and filling up the gap between the books. Fig. 7 presents example showing several steps in books detection module.

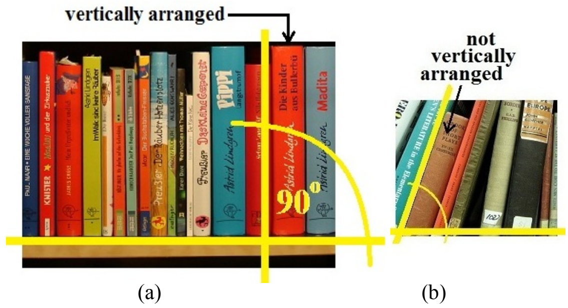

Figure 5. (a) shows a book arranged vertically and (b) shows a book not arranged vertically.

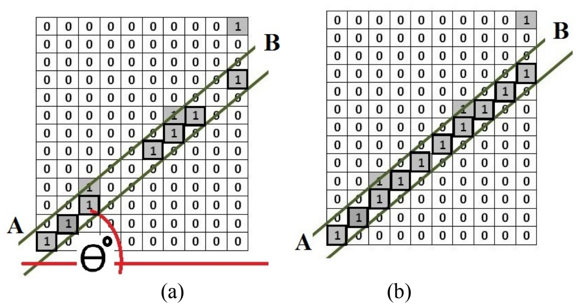

Figure 6. Angular line detection method for book detection. Here $T_{L\theta} = 0.70 \times 10 = 7$. In (a) number of pixels underlying the line $AB, P_\theta = 7$. As $P_\theta \geq T_{L\theta}$, every pixel underlying AB is set to 1 which is shown in (b).

Separated regions of individual books can be extracted using Connected Component Analysis (CCA). It can be observed from the extracted components that, most of them are books, but still some unwanted objects or components are also extracted. In this case, we can distinguish between book and non-book components by rejecting small shaped and low height components because of the height of a book can't be too short or its size can't be too small. The concept of differentiating between book and non-book components is presented in Fig. 8. Comparison between the heights of a book and a non-book component is shown in Fig. 8 as an example.

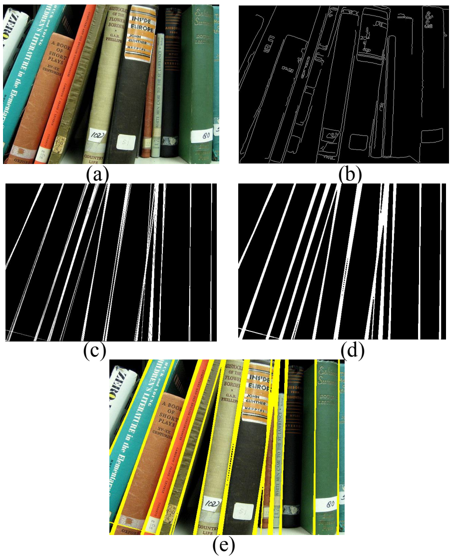

Figure 7. (a) is the bookshelf row image, (b) is the image after canny edge detection, (c) shows the result after angular line detection method and (d) is the result after performing morphological closing operation on (c). In (e) detected books are separated (indicated by yellow lines).

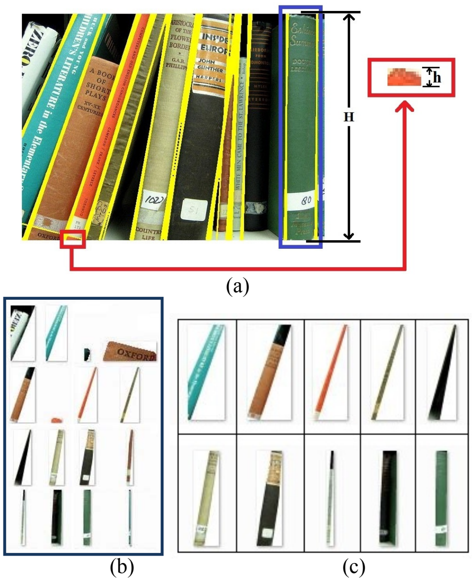

Figure 8. (a) shows example of typical height of book (H) and height of a small non-book component (h). It can be observed that $h << H$, so small component of height h will not be selected as a book. (b) shows all the components obtained by CCA. And (c) shows detected books after rejecting non-book items.

III. Simulation

The proposed technique has been implemented using MATLAB programming language. Several images have been captured and collected to use them as inputs of the proposed technique. Fig. 9-11 shows simulation results of applying proposed technique on different input images. From the figures, it can be observed that the proposed method shows adequate performance to detect books and extract them from library bookshelf images.

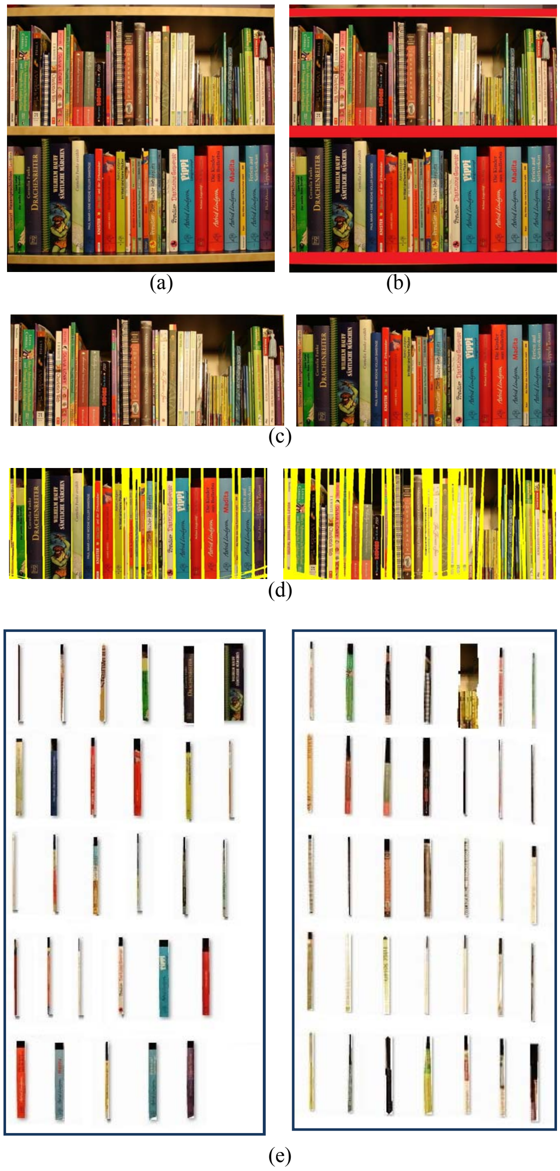

Figure 9. Results of applying proposed technique on a bookshelf image in (a). Here (b) shows detected regions of bookshelf rows (separated by red lines) and the extracted rows are shown in (c). (d) shows the detected books (separated by yellow lines) and (e) shows the extracted individual books after differentiating between book and non-book components.

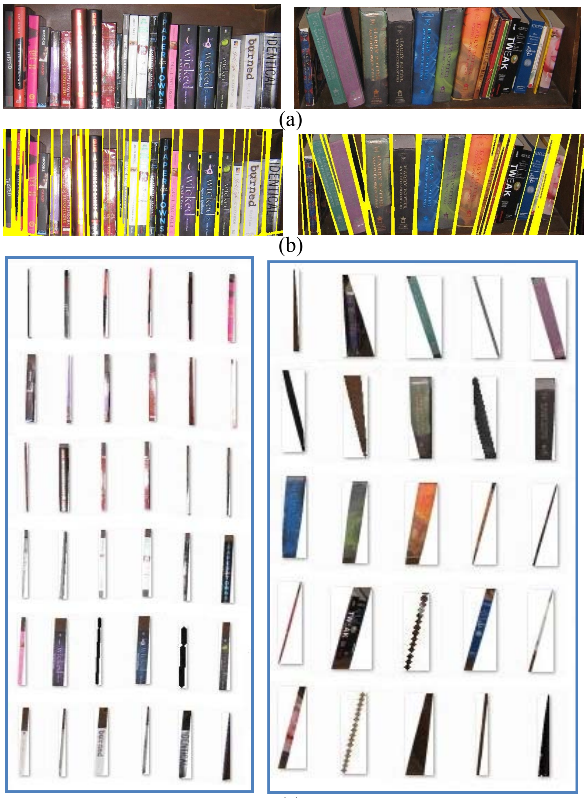

Figure 10. Results of applying angular line detection method on two individual shelf row image in (a). Here (b) shows the detected books (separated by yellow lines) and (c) shows the extracted books by applying CCA after distinguishing book and non-book components.

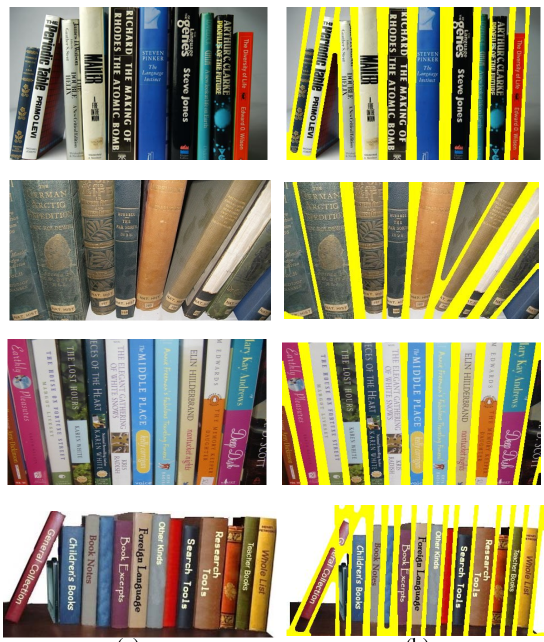

Figure 11. Detection of books arranged in angular orientation. Column (a) shows the input image and (b) shows the detected books (separated by yellow lines).

IV. FUTURE PLAN

Though the title extraction module is beyond the scope of our paper, the proposed method can be integrated successfully into any title extraction method and this challenge will remain as our future plan. There are some limitations of the proposed technique. Differentiating between book and non-book items depends only on the height and size of the components extracted by CCA. Here, many other features are needed to be considered e.g. color and width of a book. A better solution can be adopting a classifier which can be trained to classify book and non-book components. The proposed technique can be improved to detect books more accurately and this is a future plan of our research.

V. CONCLUSION

In this paper, a new technique is proposed that detects books from library bookshelf image. Firstly, the proposed technique detects bookshelf rows from the input image using a horizontal line detection method. After that, the bookshelf row image is processed for further detection. Angular line detection method is adopted rather than vertical line detection approach because of the angular orientation of book arrangement. The proposed technique detects individual books and separates them after distinguishing book and non-book components. Simulation result exhibits that the proposed technique successfully detects books from library bookshelf image.

REFERENCES

[1] Yukari. A., Minoru. I, “Book Recognition from Color Images of Book Shelves”, MVA 1998 IAPR Workshop on Machine Vision Applications, Makuhari, Chiba, Japan, 1998, pp. 106-107.

[2] Eiji. T., Seiichi. U., Hiroaki. S, “Block Boundary Detection and Title Extraction for Automatic Bookshelf Inspection”, 10th Korea-Japan Joint Workshop on Frontiers of Computer Vision, FCV 2005, Fukuoka, Japan (2004).

[3] Taira. P.E., Uchida. S., Sakoe. H, “Book Boundary Detection from Bookshelf Image Based on Model Fitting”, International Symposium on Information Science and Electrical Engineering, 2003

[4] Prats. M, Sanz. P. J, del Pobil. A. P, “Model-based Tracking and Hybrid Force/Vision Control for the UJI Librarian Robot”, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Canada, pp.1090–1095, 2005.

[5] Quoc. N. H, Choi . W, “A Framework for Recognition Books fn Bookshelves”, ICIC'09, Proceedings of the 5th international conference on Emerging intelligent computing technology and applications, 2009, pp. 386-395.

[6] Torr. P.H.S, “MLESAC: A New Robust Estimator with Application to Estimating Image Geometry”, Computer Vision and Image Understanding 78, 2000, vol. 78, pp. 138–156.

[7] Tsai. S. S. et al., “Combining Image and Text Features: A Hybrid Approach to Mobile Book Spine Recognition”, MM '11 Proceedings of the 19th ACM international conference on Multimedia, 2011, pp. 1029-1032.

[8] Jubair. M. I, Rahman. M.M, Ashfaqueuddin. S, and Ziko. I.M, “An Enhanced Decision Based Adaptive Median Filtering Technique to Remove Salt And Pepper Noise in Digital Images”, 14th International Conference on Computer and Information Technology (ICCIT), 2011, pp. 428 – 433.

[9] Gonzalez, R.C., and Woods, R.E., Digital Image Processing (2nd edition), Pearson Education, 2004.

[10] Haralick, R. M., and L. G. Shapiro, Computer and Robot Vision, Vol. I, Addison-Wesley, 1992, pp. 158-205.