4th IEEE Conference on Automation Science and Engineering
Key Bridge Marriott, Washington DC, USA
August 23-26, 2008

Matching Book-Spine Images for Library Shelf-Reading Process Automation

D. J. Lee, *Senior Member IEEE*, Yuchou Chang, James K. Archibald, *Senior Member IEEE*, and Clint Pitzak

***Abstract*—Machine vision has become an important visual inspection technology for many automation applications. Using machine vision for automation can reduce operating costs and increase efficiency and accuracy. This paper presents an image matching technique designed specifically for improving the library inventory (shelf-reading) process. In contrast with more complex color image matching techniques, the proposed method quantizes color images of book spines into a limited number of color indices and performs image matching on the quantized color index images. This approach simplifies and speeds up the processing and improves the overall inventory process. The potential performance of this robust color quantization and image matching technique is demonstrated by the results of preliminary experiments.**

I. Introduction

TAKING inventory is a daunting task in any industry, but when the number of items reaches into the multi-millions, as is the case with most major libraries, and each item has to be accounted for without the benefit of automation, it turns into an almost impossible task. A comparison done by the Online Computer Library Center shows that libraries in the United States alone circulate more books every day than the shipping giant FedEx delivers packages. Approximately 5.4 million books are checked out daily from libraries across the U.S. Furthermore, libraries worldwide hold an estimated 16 billion volumes [1]. Even allocating just one second per book, a full inventory would require over 507 man-years. Many public libraries report spending hundreds of dollars annually to replace books thought to be missing but that are merely misshelved. In contrast, when inventory is not taken the costs are even higher; it "costs the patrons access and costs the library time and money in conducting missing book searches, processing extra interlibrary loans, and purchasing duplicate copies, not to mention the additional time reference and circulation personnel spend soothing disgruntled patrons." [2].

Manuscript received March 20, 2008.

D. J. Lee is with the Electrical and Computer Engineering Department, Brigham Young University, Provo, UT 84602 USA (phone: 801-422-5923, e-mail: djlee@ee.byu.edu).

Y. C. Chang is with the Electrical and Computer Engineering Department, Brigham Young University, Provo, UT 84602 USA (email: yuchou.chang@gmail.com)

J. K. Archibald is with the Electrical and Computer Engineering Department, Brigham Young University, Provo, UT 84602 USA (e-mail: jka@ee.byu.edu).

C. J. Pitzak is with the Electrical and Computer Engineering Department, Brigham Young University, Provo, UT 84602 USA (e-mail: crg36@byu.edu).

Shelf reading is the process of visually scanning each shelf of books to ensure all items are in the correct location. If a book is not near its correct location, the librarian will send it back to the circulation department to be reshelved; slightly out of order books are reshelved during the process. The major challenges that arise in maintaining the work force required to accomplish the large task of library inventory are due to its monotonous nature. The monotony can lead to higher than normal errors in checking and recording. It is also considered a high burn-out position, causing significant resources to be consumed for employee training because of the high turnover rate. Attempts to reduce the resources required for regular and accurate inventories have included studies to improve manual shelf-reading techniques. Better training methods and new strategies to reduce burn-out were adopted to improve shelf-reading accuracy and efficiency for one project [2].

When equipment such as barcode scanners is used, each book must be taken from the shelf, its cover opened, and the book scanned. According to one report, this increased the total time required to complete the shelf reading, but it resulted in the added benefit of having a recorded inventory at the end [3]. Even with such improvements, the amount of time and labor required—as well as the number of errors that occur—is still substantial. Unique labeling systems such as book spine color codes or 2-D barcodes have also been developed to help identify misshelved books more quickly and with less error. The misplaced books are easily identified with this system, but it does not address the issue of registering the books to take inventory. Another alternative is to use the modern technology of radio-frequency identification (RFID) chips. This approach requires replacing existing call numbers, special labels, or barcodes, constituting a substantial initial cost for a large library.

A portable wireless book-spine image scanning and processing device is proposed to improve the library inventory process. This shelf-reading device (SRD) is based on technologies developed for military micro unmanned air vehicle (UAV) applications. It has an embedded computer vision system that is capable of capturing high resolution digital images of the spines of books on the shelf, communicating with the library database server, and performing book recognition and/or verification. A list can then be generated of missing and misplaced items. This solution eliminates the need for comprehensive manual shelf-reading, reducing the time and cost of a full inventory, as well as increasing its accuracy. No changes to existing books are required, so the solution is more cost effective than the alternative approaches and technologies mentioned above.

Most color imaging devices capture and store images as red, green, and blue (RGB) components. The majority of book spines are colored, and color information is essential for accurate book spine recognition and verification. Matching color images in RGB is more complicated than matching black and white images. The proposed algorithm applies an image-independent color quantization method to quantize color book-spine images into color indices for a small set of representative colors. When used to represent the color of every pixel, these color indices can form a grayscale image whose "intensity" is the color index. Features such as mean, variance, and histogram that represent the quantized black and white image (consisting of color indices) can then be calculated and used in image matching to identify each individual book.

In Section II, we discuss the overall process of this improved shelf-reading system. The details of the selected color quantization method and the simple image matching method using the mean, minimum, and maximum as features for book spine recognition and verification is presented in Section III. We demonstrate the feasibility of the SRD system using examples of book spine images and present preliminary experimental results in Section IV. Finally, we summarize our work and conclude the paper in Section V.

II. Process Flow

As mentioned in the introduction, a shelf-reading device is used to scan and process book spine images and to determine if each book is properly shelved, misplaced, or missing. Fig. 1 shows the process flow of this automated library inventory system. Spine images of all volumes in the collection are captured and stored in the database of the server. This database contains the inventory information including the call number, shelf number, row number, and status (on the shelf or checked out) of each item in the collection. The proposed system will add the spine image and calculated image features that represent each book to the database.

A library worker uses the proposed shelf-reading device to capture images of the books on the shelf. Features of the captured image are calculated and compared against the features stored in the database. Since the server contains detailed inventory information, feature matching needs to be carried out for just a few books rather than the entire database. If there is no match in the close neighborhood of the location where a given book should be located in the captured image, then a flag will be set to indicate that it is missing. If a book in

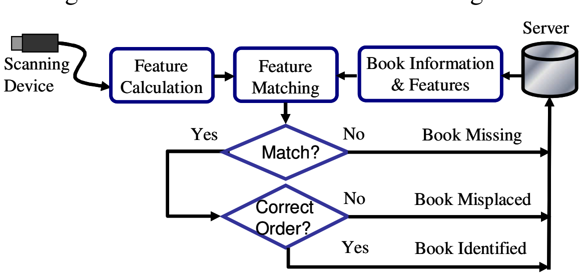


Fig.1. Process flow of the proposed system.

the captured image does not match database entries of books that should be found in the vicinity, the book is classified as misplaced. If the image is matched and the recognized book is in the correct order, then it is classified as identified. A report including inventory information and status will be generated for a library worker to correct all detected problems. Note that the resolution of specific problems is significantly less monotonous than conducting an entire inventory manually.

Fig. 2 (a) shows a book spine image that is captured and stored in the database and Fig. 2(b) shows an image captured by the shelf-reading device that contains the book. The goal of this research is to use the stored image to determine the presence and location of the book in captured images.

Fig.2. (a) stored book spine image and (b) image captured from a book

III. Algorithm

Many content-based image retrieval applications use the visual content in the image itself to measure image similarity. Image content is typically represented by summarizing visual features such as color, texture, and shape. The matching of book-spine images for library shelf-reading automation also requires measuring image similarity. It is difficult, however, to perform similarity measurements directly on captured RGB images since they consist of pixels with values ranging over a three-dimensional color space. An algorithm is proposed that first quantizes the color image into color indices and then determines—in terms of those color indices—the features of book-spine images to be used for similarity measurements.

A. *Color Quantization*

24-bit color images have three color components, red, green, and blue, which are combined to generate over 16 million unique colors. Compared to a 256 grayscale image, a color image can convey much more information and detail about the scene to the human perceptual system. However, not all 16 million colors are distinguishable by humans, particularly if shades and brightness are very similar.

Color quantization [4] is a sampling process of 3-D color spaces (RGB, CIE Lab, HSV, etc.) to form a subset of colors known as the palette which are then used to represent the original color image. Color quantization is particularly convenient for compressing, transmitting and displaying color images. Unlike most color quantization methods that generate a color palette with three separate color components for each color in the selected subset, quantization using Fibonacci lattices denotes colors using single scalar values. These scalar values can be used to denote a visual “distance” between distinct colors. In contrast, traditional color quantization algorithms such as uniform [5], median cut [5], and Octree [6] use palette indices only to point to the stored, quantized 3-D color values. Attributes of Fibonacci lattice quantization are very useful for our application: we use the technique to convert colors in captured images into a small number of scalar color indices for image matching.

The Fibonacci lattice sampling scheme proposed in [7] provides a uniform quantization of CIE Lab color space and a way to establish a partial order relation on the set of points. For each different L value in CIE Lab color space, a complex plane in polar coordinates is used to define a spiral lattice as a convenient means for sampling. The following set of points in the (a, b) plane constitutes a spiral lattice:

$$z_n = n^{\delta} e^{j2\pi \cdot n\tau}, \tau, \delta \in R, n \in Z \tag{1}$$

Figure 3 shows an example of the spiral Fibonacci lattice for $\tau = (\sqrt{5}-1)/2$ and $\delta = ½$. Each point $\boldsymbol{z_n}$ is identified by its index n. Parameters τ and δ determine the axial distribution and the radial distribution of the points respectively. If there exist N_L luminance (L) values and N_p colors in the corresponding (a, b) plane, for each color in the palette, the corresponding symbol is determined by adding its chrominance index n to a multiple of its luminance index i:

$$q = n + N_p \cdot i \tag{2}$$

Consequently, the L, a, and b values for any color from the palette can be reconstructed from its symbol q. For a pixel p, with color components L_p, a_p and b_p, the process of determining the closest palette point starts by finding the closest luminance level L_S from the N_L levels available in the palette. The luminance level L_S determines an (a, b) plane, and one of the points $\boldsymbol{z_n}$ $(0 \le n \le N_p)$ in that plane is the minimum mean square error (MSE) solution. The exact solution, q, is the point whose squared distance to the origin is the closest to $r_p^2 = a_p^2 + b_p^2$.

These L values can approximately denote the luminance

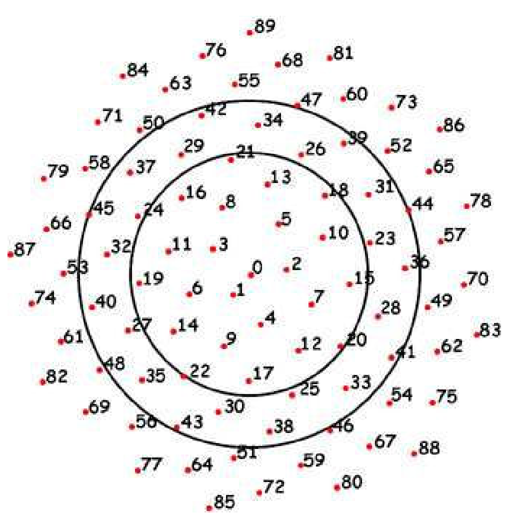
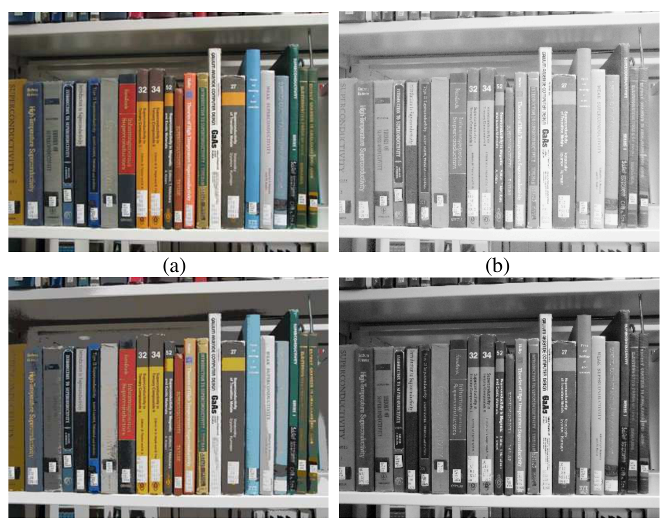


Fig. 3. Points of the Fibonacci Lattice in a complex plane

levels of the image. Since the (a, b) plane is not circular, there will be points in the Fibonacci Lattice whose colors are not valid in the RGB color space; all such points are labeled "range invalid". The quantized points are given by $z_n = S\sqrt{n}e^{j(2\pi n\tau + \alpha_0)}$, where $\tau = (\sqrt{5}-1)/2$, α_0=0.05, and S=1.5. For the 3264×2448 image shown in Fig. 4(a) having 525941 colors, the L component is quantized into 12 user-selected values {0,10,20,30,40,50,60,70,80,85,90,100}. These L values and N_p =800 points in each plane are used to construct the palette, so the size of the palette is 12×800 =9600. Fig. 4(b) shows (as a 245-level grayscale image) the resulting 67 indices of the original image. Each of these index values has been assigned an 8-bit value (0, 9, 19, 28, 38, …, 247) for display. Fig. 4(c) shows the quantized color image with 67 valid colors in the palette. Each pixel is labeled by the one dimensional symbol q, which not only is the index of an entry in the palette, but also represents the color information to some extent.

Compared with Fig. 4(d), a 256 grayscale image derived directly from the original, most color book spines are much easier to distinguish in the quantized image (Fig. 4(c)) despite the grayscale frame having more levels (256) than the frame quantized by Fibonacci lattices (with just 67). Easily distinguished colors can appear very similar in a grayscale image. Because human perception contrast in quantized images can be measured by the distance between the q symbols of two colors, it is more accurate to calculate book spine image features based on color indices to a palette constructed by Fibonacci Lattice-quantization than to use 256 levels of grayscale.



(a) (b) (c) (d)

Fig. 4. (a) Original color image, (b) 67 quantized indices, (c) quantized color image, and (d) grayscale image of 256 gray scales.

B. Features

Various color representation methods have been proposed for efficient retrieval [8-11]. The most fundamental and well-known color representation method is the color histogram, i.e. the frequency distribution of colors in an image. Color histograms can be constructed for each color component and the number of histogram bins can be chosen based on precision requirements and other preferences. In our algorithm, each multi-dimensional color has been converted to a one-dimensional color index, making it easy to calculate a histogram for comparison.

As discussed in Section II, each book spine image is captured and stored in the database. Fibonacci lattice quantization is applied to convert the color information into color indices. Its minimum and maximum color indices and color index histogram are pre-calculated and stored in the database for comparison. By counting the index values of pixels in the book spine image, the color index histogram is calculated as follows:

$$Hist(i - I_{Min} + 1) = \sum_{m=1}^{M} \sum_{n=1}^{N} 1, \ if \ Idx(m,n) = i, \qquad (1)$$
$$i = I_{Min}, I_{Min} + 1, I_{Min} + 2, \cdots, I_{Max}$$

where $Idx(m, n)$ is the Fibonacci lattice quantization index of the pixel (m, n) in the image, i is the color index of the histogram ranging from I_{Min} to I_{Max}, and M and N are the width and height of the image, respectively.

Of course, image texture and other features [12-14] could be used for this application. A common definition of texture is the repetition of basic structure elements. Texture information is represented by variations in pixel intensity rather than color, and thus is used most commonly on grayscale images, but it can also be applied to the converted color indices. Frequency, direction, phase, etc. are quantities used to describe the properties of image texture. Methods derived from Gabor wavelets, the conventional discrete wavelet transform, and discrete wavelet frames [15-17] and co-occurrence matrices (e.g., Gray Level Co-occurrence Matrices) are often used to describe texture information. In this paper, a color index histogram is used to demonstrate the feasibility of this approach. More robust features will be explored in our future work to improve performance.

C. Matching

Book spine images stored in the database are compared against the images captured from the bookshelves as shown in Figures 2(b) and 4(a). The histogram of a subset of the books in the captured image of the same size as the current (expected) book is calculated and compared against the stored histogram of the current book. The similarity measure is calculated as

$$D_c = \frac{1}{I_{Max} - I_{Min} + 1} \sum_{i=1}^{I_{Max} - I_{Min} + 1} \left| Hist_{Cur}(i) - Hist_{Patch}(i) \right| \qquad (2)$$

where D_c is the average distance between the histogram of the current book and the selected subset of the captured image. The search process starts from the upper left corner of the captured image. The matching process compares the histogram of the current book to the histogram of the image subset of the same size in the captured image. The distance between two histograms is calculated and recorded. The window defining the image subset is moved by a user-selected amount and a new histogram distance is calculated. Searching can be limited to a small region because the book is considered missing or misplaced if it is not shelved in the correct order. The region that has the smallest histogram distance is chosen as a match as long as the histogram distance is lower than a user-selected threshold. In our experiments, the threshold was set to 30.

Figure 5 show the searching criteria. Once the current book, specified by the data stored in the database, is found, the search for the next book starts from the right edge of the previous match. If the next match is found immediately adjacent to the previous match, then it is considered to be in the right order. If the next match is found at a distance from the previous match, then the book between the two matches is considered to be misplaced. Also, every search has a user-selected maximum searching distance. If the next match is not found within the maximum distance, then the book is considered missing.

Fig. 5. Searching parameters.

IV. Experimental Results

A. Experiment Design

Five captured images were used for preliminary testing. The first is shown in Figure 2(b), the second is shown in Figure 5, and the remaining three are shown in Figure 6. Ten books from each captured image were selected for testing. Three experiments were designed to evaluate the performance of the proposed algorithms.

The first experiment was to test the accuracy of successful matching. Individual spine images were captured for each of the ten books selected from each image, and their histograms were pre-calculated. Searching was performed according to the criteria discussed in Section III.C. The accuracy was evaluated as the percentage of successful matches.

The second experiment was to test the accuracy of detecting missing books. Fifty images of book spines that were not present in the five captured images were used for testing. The rate of false matching was calculated to evaluate the performance. The last experiment was to test the accuracy of detecting misplaced books. Fifty books were chosen for

Fig. 6. Three captured images for experiments..

this test. They were chosen randomly with misplaced books between them. The performance was evaluated as the rate of successful detection of misplaced books.

B. Performance Evaluation

Of the 50 books selected for the first experiment, only the fourth book from the left of image in Figure 5 and the fourth book from the right of image in Figure 6(a) were mismatched. All other 48 books were matched correctly in the right order. The matching accuracy for book spine verification was 96% (48 out of 50).

The second experiment was for detecting missing books. Ten book spine images were used for each of the five captured images for this experiment. Eight of these ten book spines were present in the captured image and two were not. With the histogram distance threshold set at 30, all books present in the captured images were successfully detected (100%) and all books not present were successfully identified as missing (100%).

The third experiment was very similar to the first one. Misplaced books were detected correctly in 48 of 50 cases. Thus, in 96% of cases, books inserted between consecutive matches were successfully determined as misplaced.

C. Anticipated System

Figure 7 shows an image of the anticipated scanning device that is currently being designed. This device utilizes field-programmable gate arrays (FPGA) to implement the hardware and software, reducing its overall size. In particular, our prototype system uses the Helios board (shown in Figure 8(a)) developed at Brigham Young University [18-19]. The board is based on a powerful Xilinx Virtex-4 FPGA chip that includes two 400 MHz PowerPC processors, allowing for both custom hardware and software. By carefully utilizing the strengths of both hardware and software, the scanning device can run in real time. The scanning device contains a CMOS camera (as shown in Figure 8(b)) that feeds image data to the camera interface logic on the FPGA. The actual connection to the camera is made through a daughter board (shown in Figure 8(c)) that connects to Helios via a 120-pin header. The FPGA captures images, accesses the library database of spine images wirelessly, computes histogram distances, determines the best match of the book-spine image, and transmits wirelessly the resulting status (match, miss, or misplaced) of each book back to the database.

Fig. 7. Anticipated scanning device.

V. Conclusion

Shelving is a tedious process that creates physical and mental strain on the library staff. As a result librarians sometimes place books at incorrect locations. Maintaining a high accuracy in reshelving is important. The yearly budget of $177,357 for library staff at Brigham Young University (BYU) could be reduced by implementing an automated book shelf-reading system. Rather than employing 32 shelvers to keep up with demand, the number could be reduced to over half, and the accuracy of book placement would increase as well.

The major cause of misshelved books is incorrect placement by library patrons. The BYU library continues to experience an increase of patrons using books in the library rather than checking out the books. Patrons that use books internally are requested to place their finished books on a reshelving cart or shelf, but some attempt to place their books back where they were originally found. The accuracy with which patrons correctly replace their books is considered to be very low. Thus, internal book use has significantly increased the workload of librarians.

Our initial experimental results show that automated shelf-reading with high accuracy is feasible. In this paper, we have presented an innovative approach that converts color information into a user selected number of color indices (9,600 in this paper). This color conversion significantly simplifies the calculation of color image features for image similarity measurement. The use of index histogram distance as a similarity measure showed very promising results. Our experiments were performed with vertical books, consistent lighting, and consistent image resolution. These conditions

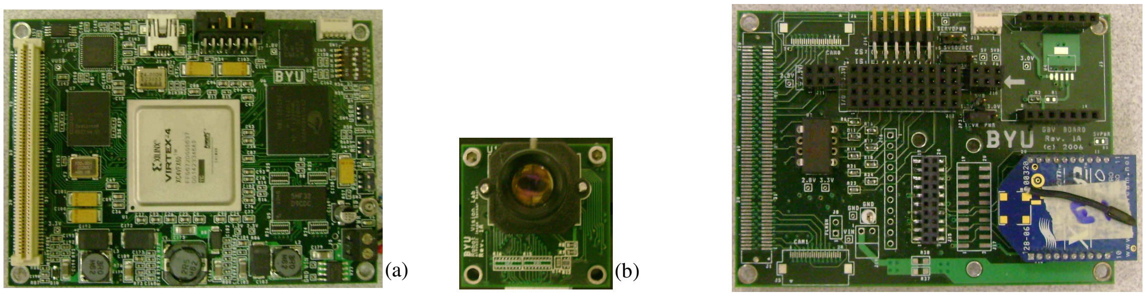


Fig. 8. Hardware components (a) Helios FPGA board, (b) CMOS imager, and (c) AVT daughter board

will not always be true. Books with similar color or subtle difference such as periodicals will also present significant challenges. Our future work will include more sophisticated texture and feature representations of the image as discussed in Section III.B to successfully identify books in these more challenging cases, as well as hardware implementations of histogram matching to boost performance.

REFERENCES

[1] OCLC Online Computer Library Center, Inc. "Libraries: How they stack up" http://www.oclc.org/reports/2003libsstackup.htm.

[2] Anderson, D. R., "Method Without Madness: Shelf-Reading Methods and Project Management." College and Undergraduate Libraries. vol. 5. no. 1, pp. 1069-1316, 1998.

[3] Miller, C., "An Investigation of an Automated Shelf Reading and Inventory System for the Clemson University Cooper Library." http://www.ces.clemson.edu/ie/academics/isaris.pdf

[4] Jain, A.K., Fundamentals of Digital Image Processing, Prentice Hall Information and System Sciences Series, 1989.

[5] Heckbert, P., Color Image Quantization for Frame Buffer Display, ACM Conference on Computer Graphics and Interactive Techniques, pp. 297–307, 1982.

[6] Gervautz, M., and Purgathofer, W., A Simple Method for Color Quantization: Octree Quantization. In New Trends in Computer Graphics, pp. 219–231, 1988.

[7] Mojsilovic, A., and Soljanin, E., Color Quantization and Processing by Fibonacci Lattices, IEEE Transactions on Image Processing, vol.10, pp. 1712-1725, 2001.

[8] Swain, M. and Ballard, D., "Color indexing," International Journal of Computer Vision, vol. 7, no. 1, pp. 11-32, 1991. 3

[9] Han, J. and Ma, K.K., "Fuzzy color histogram and its use in color image retrieval," IEEE Transactions on Image Processing, vol. 11, pp. 944{952, Aug. 2002.

[10] Wan, X. and Kuo, C.C., "A new approach to image retrieval with hierarchical color clustering," IEEE Transactions on Circuits and Systems for Video Technology, vol. 8, pp. 628-643, Sept. 1998.

[11] Lee, S. H., Cho, N. I., and Park, J.I., "Disparity estimation using color coherence and stochastic diffusion," IEEE International Conference on Image Processing, vol. 2, pp. 1373-1376, Oct. 24-27, 2004.

[12] Lowe, D.G., Distinctive Image Feature from Scale-Invariant Keypoints, International Journal of Computer Vision, vol.60, pp.91-110, 2004.

[13] Ke, Y., and Sukthankar, R., PCA-SIFT: A More Distinctive Representation for Local Image Descriptor, International Conference on Computer Vision and Pattern Recognition, vol.2, pp.506-513, 2004.

[14] Lowe, D.G., Object Recognition from Local Scale-Invariant Features, International Conference on Computer Vision, vol.2, pp.1150-1157, 1999.

[15] Zhao, C., Cheng, H., Huo, Y., and Zhuang, T., "Liver CT-image retrieval based on Gabor texture," The 26th IEEE Annual International Conference of the Engineering in Medicine and Biology Society, vol. 1, pp. 1491-1494, 2004.

[16] Kokare, M., Biswas, P., and Chatterji, B., "Texture image retrieval using new rotated complex wavelet filters," IEEE Transactions on Systems, Man and Cybernetics, vol. 35, pp. 1168-1178, Dec. 2005.

[17] Wolf, C., Jolion, J.M., Kropatsch, W., and Bischof, H., "Content Based Image Retrieval using interest points and texture features," The 15th IEEE International Conference on Pattern Recognition, vol. 4, pp. 234-237, Sep. 3-7, 2000.

[18] Robotic Vision Lab, Brigham Young University, http://www.ee.byu.edu/roboticvision/helios/index.phtml

[19] W. S. Fife and J. K. Archibald, "Reconfigurable On-Board Vision Processing for Small Autonomous Vehicles," EURASIP Journal on Embedded Systems, vol. 2007, Article ID 80141, 14 pages, 2007. doi:10.1155/2007/80141.