

An Application of the Checkstand Antitheft System by Means of the Image Recognition Technique

Sung Min Hong, Dong Ryoung Seo, Dong Sin Kim, Do Won Hyun and Ju Wook Jang

Department of electronic engineering
Sogang University
Seoul, Korea

e-mail: feverbypassion@hotmail.com, drs@monet1.sogang.ac.kr, freeman@sogang.ac.kr, snatcher@monet1.sogang.ac.kr , jjang@sogang.ac.kr

Abstract - This study aims to develop an application to recognize any missing barcode scan despite normal scanning motion by means of the video image processing algorithm in order to prevent an omission of product barcode scanning, intentional or mistaken, by a checkstand. The algorithm that searches for a barcode from a moving object in a video is divided into two steps: the step of extracting the moving object from a video; the step of extracting a barcode of the moving object. The process of extracting moving objects consists of blocking the video frame images, calculating motion vectors of each block, and clustering vectors of similar direction and size in the order. The process of extracting a barcode from a moving object includes the step of changing the color of the moving object image to that of the HSL color model, selecting areas, part of whose block is colorless, for the portion to be scanned for the barcode, and recognizing the barcode when the portion consists of a certain number of blocks in the order. The application has been developed by means of this algorithm for Microsoft Windows. If no barcode scan output is found even though the scan motion was normally implemented, this application will check if the barcode exists and prevents barcode omission from taking place at checkstands of a discount outlet.

Keywords-Using motion vector; Clustering; Image recognition; Application implementation.

I. INTRODUCTION

According to the managers of major discount stores, the financial loss due to intentional or mistaken barcode scan omission by a cashier is as much as 2% of the total sales, but there is no system that can effectively supervise the process. In a major discount outlet in Korea, for example, the sales in 2009 reached some 10 trillion (HI Investment & securities co. ltd Research Center, October, 2009), which indicates the loss of more than 200 billion won. In the end, this type of loss results in increase of retail prices and damage to a wide range of consumers. Thus, the need for a system to completely prevent barcode scan omission from happening has been consistently emphasized.

This study, therefore, suggests and develops an algorithm to find a barcode on a moving object in a video with relatively small amount of operation. First of all, adopted was the moving object extracting method suggested previously to find a moving object [1]. This is a method to block each frame in an image and find the motion vectors

with the central point of the block as the center. In consideration of the fact that a moving object has motion vectors of the similar direction and size, the moving object is detected by clustering vectors of the similar direction and size. The use of this technique reduces the operation compared to that of finding vectors by means of characteristics or colors. Then the color of the extracted object is changed and the barcode is checked. As the barcode is searched for, not on the entire frame, but on the area of the moving object, this further contributes to the reduction of the operation. This study embodied and verified this application by means of the algorithm above in Microsoft Windows System.

Section 2 explains the way of extracting moving objects from a video, section 3 the way of extracting the barcode from the extracted moving object, section 4 the results of the actual representation, and section 5 concludes the study and presents issues for the future study.

II. SEPARATION OF A MOVING OBJECT FROM BACKGROUND

This section suggests a swift and effective algorithm to detect a moving object. The suggested algorithm block each frame of an image, and then extracts motion vectors with the central point of the block as the center. Extracting motion vectors for each block reduces the operation more than in extracting for all pixels or characteristics. Figure 1 shows the example of extracting motion vectors for each block.

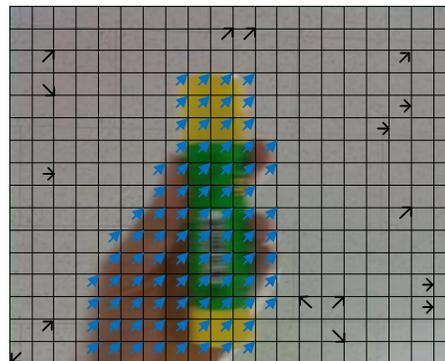


Figure 1. Image blocking and motion vector extracting for each block

For the motion vectors for the extracted block, a 3-dimensional histogram is prepared. The X, Y coordinates represents the directions of motion vectors while the Z coordinates represents the sizes of them. Figure 2 shows the 3D histogram mapping of the extracted motion vectors.

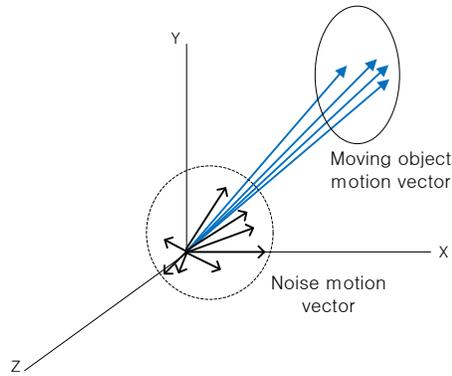


Figure 2. 3D histogram of motion vectors extracted for each block

As in Figure 2, motion vectors of the block of the moving object have similar directions and sizes. Noise vectors generated due to lighting or shadow may have different elements from those of the moving object vectors. After the vector clustering, remains the block of vectors of similar direction and size while removed are other blocks. Then only the moving object is extracted. However, there is a possibility that other blocks than the moving object may have similar directions and sizes with the moving object block. Thus, only when the block has more than a certain number of similar vectors, it is recognized as a moving object, and others are removed as noise. Figure 3 shows the result of extracting the moving object in the way above.

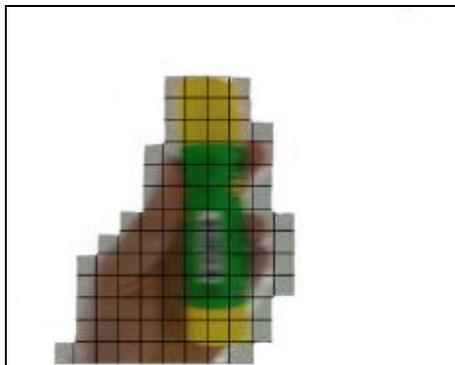


Figure 3. Extraction of blocks with the moving object's vectors and then of the moving object

Figure 4 is the flowchart of the algorithm to extract the moving object by means of motion vectors after the image blocking.

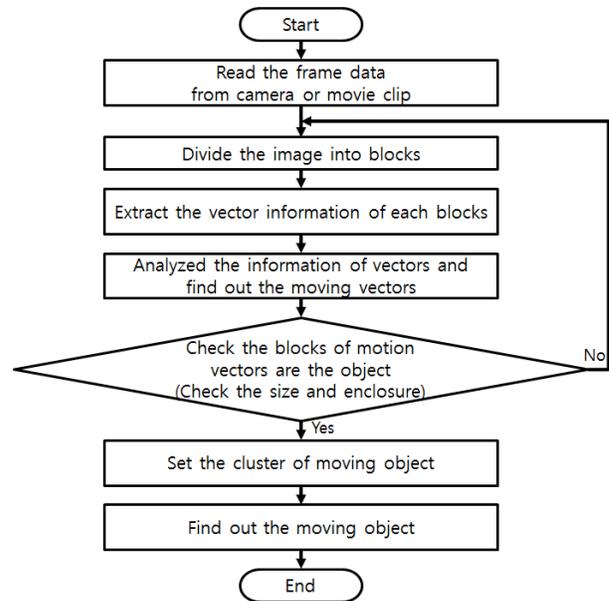


Figure 4. The moving object detecting algorithm by means of the motion vectors of the blocked image

This technique is appropriate for an embedded system as it adopts the blocking and clustering of motion vectors and thus reduces the amount of operation more than existing techniques.

III. EXTRACTING BARCODE FROM OBJECT IMAGE

This chapter explains the way of extracting the barcode area from the image of the moving object extracted in Chapter 2. Since the barcode scanning is implemented only in the area of the extracted moving object, the amount of operation may be reduced more than in scanning over the entire image.



Figure 5. The image of changing a frame of the video with HSL color elements

A barcode commonly consists of the white background and black bars. Thus, areas in the extracted moving object where a certain amount of white and black parts exist will be selected for barcode scanning. To this end, the image area of the moving object that consists of RGB color elements is reformed with HSL(Hue, Saturation, Lightness) color elements. Figure 5 shows that a frame of the video has been changed with HSL color elements.

When the saturation value of pixels in the image block of the moving object is 0.3 or less, it is judged as colorless, and when the colorless part is more than 70% of the block, this block is selected for barcode searching. This procedure is to remove color blocks among moving blocks. In addition, the black and white parts of the image blocks of each moving object are distinguished, the ratio of black and white is calculated, and any part whose percentage is 10% or less is excluded. This procedure is to exclude blocks that only consist of no other achromatic color than white and black parts. Although this method may effectively detect the block that contains the barcode, a part of the image that seems similar to the barcode can be detected. Thus, calculated is the complexity of pixels in the block. When the value of difference of luminance is lower than a certain degree, this is regarded as monochrome. This is possible since the blocks extracted earlier are colorless. Difference of luminance may not be the only factor to be considered if colors exist. Based on the process above, finally the areas for barcode searching are decided.

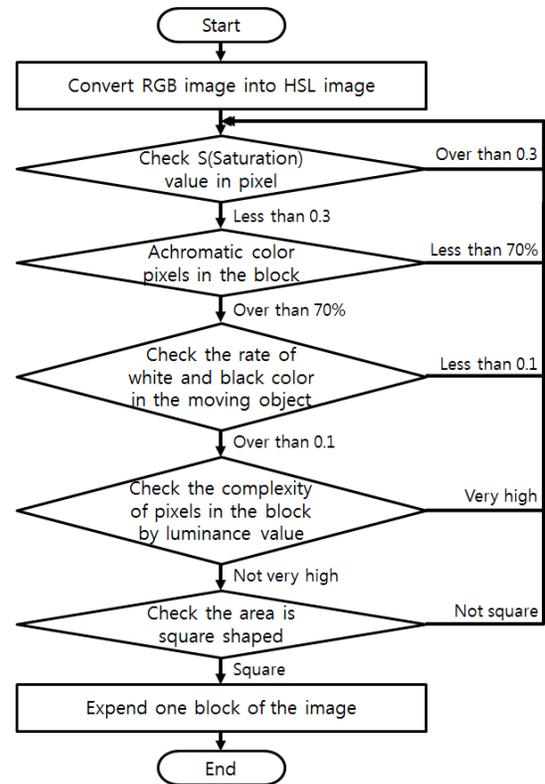


Figure 7. Flowchart of Extracting Barcode from Object Image



Figure 6. A barcode image to expand each block outward

Once the areas for barcode searching are decided, the shape of the block is checked to see if it is a quadrangle like a barcode, and then each block is expanded. The blocks are expanded to prevent a part of the barcode image from being removed or deleted in the process of blocking and to make the barcode image clearer. Figure 6 shows the expanded image of each block around the barcode.

Figure 7 shows the flow chart of the process of finding the barcode through color conversion.

IV. IMPLEMENTATION AND RESULTS

The following are the environments and software adopted to develop the application of this study:

- OS : Windows 7 Professional Edition 32bit
- Compiler : Visual Studio 2010
- Programming Language : C/C++

A. Application View

1) Main Window

Upon executing the application, the following screen in Figure 8 is displayed:

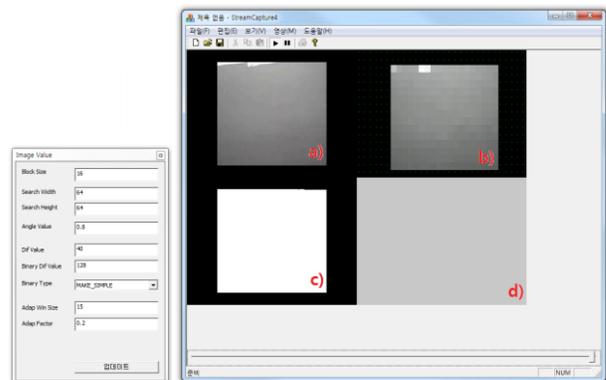


Figure 8. Main window of application

The small window to the left is an option window with functions to enhance the accuracy in barcode detection. The options of the program may be reset in this window. The big window to the right displays the following from the left: a)the original video; b)blocking and extracted motion vectors; c)black & white image of the area for barcode searching; and d)the detected barcode image.

2) Option Window

The application provides the following option window as in Figure 9:

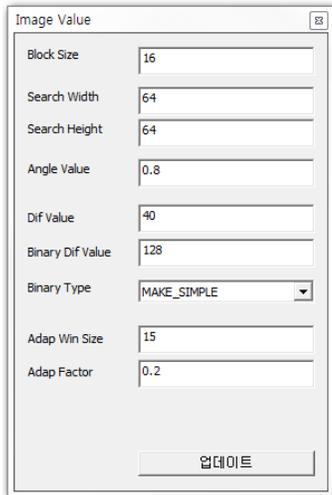


Figure 9. Option Window

The currently available options are as follows, and those that are not stated are currently not used.

- Block Size: the number of pixels for one block. If you input '16,' a 16x16 sized block is produced.
- Search Width: the width of the searching window to enhance the efficiency of block searching(unit: pixel)
- Search Height: the height of the searching window to enhance the efficiency of block searching(unit: pixel)
- Angle Value: the range of toleration to check if the motion vectors are in the same direction(0 ~ 1)
- Dif Value: The value of color difference to check if it is the same block(0 ~ 255)

B. Implementation Issues

1) Implementation issues of separation of a moving object from background

The solutions to the implementation issues of separation of a moving object are as follows:

- a) Detection of moving blocks is time consuming
 - Improve the efficiency by limiting the processing area
 - Parameterize the range of search in the program options
- b) Saturation occurs when the surface of the moving object is wide and uniform in color

- Determination of the blocks of similar colors by prioritizing the proximity search
- Utilize the process of filling in the interior of folia
- c) The boundary between the moving object and the background is irregular due to blocking

- Ignore the unclear boundary as the goal is not in extracting the perfect imagery of moving objects
- 2) Implementation issues of extracting barcode from object image

- a) Non-barcode area of wide achromatic color is erroneously extracted
 - Discard the extracted area if the ratio of black and white is less than 10%
- b) Retangular and linear images, visually close to barcode, are erroneously extracted
 - Identify barcode by calculating the complexity of pixels in the block
 - Treat the pixels as of uniform color if the difference of luminance is below certain level

C. Results

The application worked quite appropriately, and the barcode recognition of a moving object was outstanding.

Figure 10 shows the entire process of extracting the barcode from the camera image:

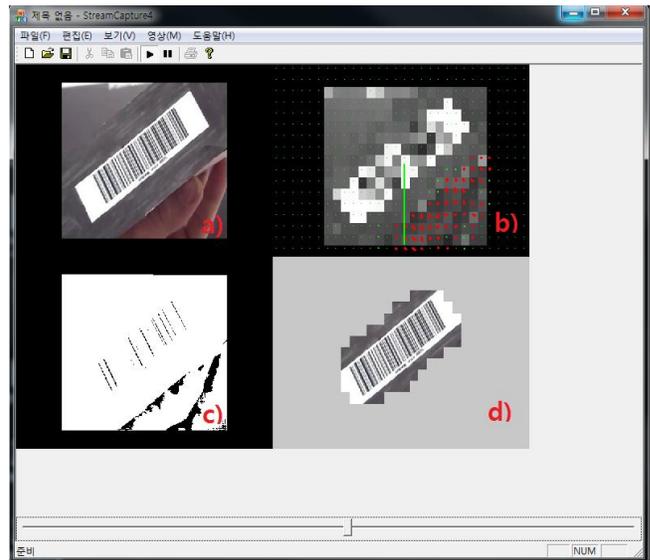


Figure 10. Extraction of the barcode from the object with the black background

Figure 10 shows the process of extracting the barcode from the object with the black background. a) the original video image taken by a camera; b) the detection based on the blocking and motion vectors; c) detection of the area for barcode searching based on the colors and shadowing; and d) the final result of locating the barcode.

This process was successfully implemented even when the object was white or in various colors.

Figure 11 shows the result of barcode extraction from an object with the white background. Figure 12 shows the result of extracting the barcode from an object in various colors.

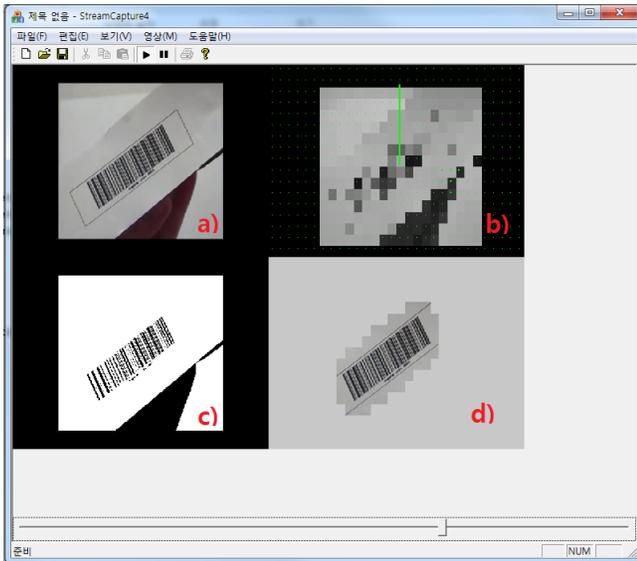


Figure 11. Barcode extraction from an object with the white background

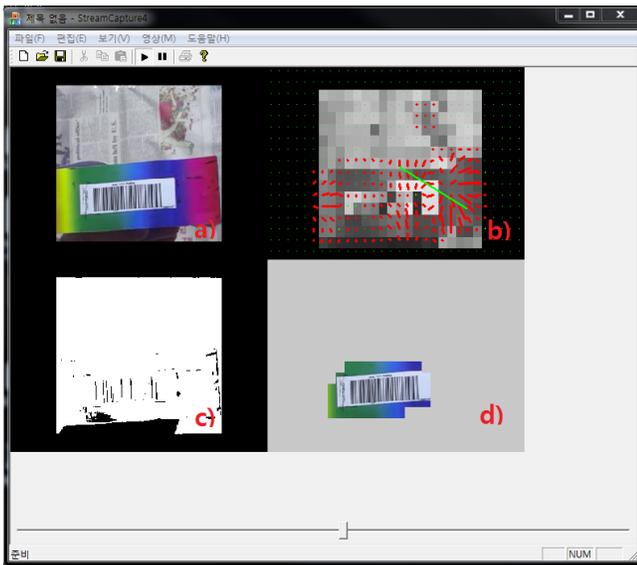


Figure 12. Barcode extraction from an object in various colors

V. CONCLUSION AND FUTURE WORK

This study aims to develop an application to detect any missing barcode scan despite the normal scanning process by

means of the video processing algorithm to prevent any intentional or mistaken barcode scan omission from happening at a checkstand.

To extract a moving object, each frame of the image is blocked, and the motion vectors are extracted with the central point of each block as the center. Among the extracted motions vectors, those of the similar direction and size are clustered and extracted. The image of this set of block is the moving object. The image of the extracted moving object is changed to a HSL color model, and the ratio of black and white and size are checked to extract the barcode. The actual application that adopts the algorithm above has been developed and tested with various video clips.

The primary factor affecting the success rate of barcode extraction was the optical focusing of the barcode area in the video images. There were cases where barcode was not properly extracted when the barcode area in the video images was out of focus, whereas the video images of which barcode area was in focus were properly processed by the the algorithm. The failures occurred because the shadows in the out of focus video images were not sufficiently clear enough for the barcode searching algorithm to properly extract the barcode based on the colors alone.

It is planned to comparatively analyze the rate and accuracy of extracting an object and a barcode in comparison with existing algorithm in the future.

ACKNOWLEDGMENT

This work (Grants No. 000428980111) was supported by Business for Cooperative R&D between Industry, Academy, and Research Institute funded Korea Small and Medium Business Administration in 2010

REFERENCES

- [1] Hyun-Sung Kim, Ju Wook Jang, "Using motion vector clustering for separation of a moving object from background," in proc. Conf. Institute of Embedded Engineering of Korea, vol 1. November, 2011
- [2] In-kwon Kim, Jae-bok Song, "Moving Object Detection based Motion Vector and Color Information," in proc. CASS, 5pages, 2007.
- [3] Ji-man Kim, Dae-jin Kim, "Detection of Moving Object from Moving Camera," Workshop on Image Processing and Image understanding 2010, 4pages, 2010.
- [4] G. D. Borshukov, G. Bozdagi, Y. Altunbasak, and A. M. Tekalp, "Motion segmentation by multistage affine classification", IEEE Transaction on Image Processing, 1997, Vol. 6, issue. 11, pp. 1591-1594
- [5] Y. Weiss, "Smmothness in layers: Motion segmentation using nonparametric on homographics", Proceeding IEEE Conference on Computer Vision Pattern recognition, 1997.