Real Time Industrial Application of Single Board Computer Based Color Detection System

Serdar SOLAK[1] and Emine DOĞRU BOLAT[2]

[1] Department of Informatics, Kocaeli University, 41380, Umuttepe Campus, Kocaeli,Turkey
serdars@kocaeli.edu.tr, serdarsolak@hotmail.com
[2]Department of Electronics & Computer Education, Kocaeli University, 41380, Umuttepe Campus, Kocaeli, Turkey
ebolat@gmail.com

Abstract

In this study, real time industrial application of single board computer based color detection system is realized. In this system, BeagleBoard-xM as a single board computer, a USB camera, a conveyor belt and an LCD7 touch screen are used. OpenCV is used as an image processing library in this color detection system. The main goal of this study is to define the number of different colored packages passing on the conveyor belt according to their color. Then, real time results of the number of the packages and the total package number are displayed on the LCD7 touch screen. At the same time, the USB camera image of the related package on the conveyor belt is monitorized on the same touch screen. If no image of any packages is taken by the USB camera during 60 seconds, the system is turned off.

1. Introduction

The importance and the number of academic and industrial studies on computer vision are increasing rapidly day by day. Especially in industrial area, many computer vision based applications on recognizing, distinguishing and counting the objects are realized. Furthermore, it is commonly used in video games, military, medical, industrial, automotive security, surveillance and security systems. Many academic and industrial studies on defining, tracking and detecting the moving and colored objects are performed.

In literature, many academic studies on color detection are found. Especially, color detection is used in studies on skin color detection [1,2 , 3], face recognition and detection [4,5,6], vehicle licence plate detection [7] and traffic sign detection [8].

Definition of colored packages passing on the conveyor belt and determining the numbers of packages according to their color are aimed in this study. BeagleBoard-xM Single Board Computer is utilized for definition and determining the number of the colored packages passing on the conveyor belt. Open Source Computer Vision (OpenCV) Library is used to detect the color of the packages. Furthermore, an LCD7 touch screen displays the table of the related colored package counts, total package count and the image taken by the USB camera.

The rest of the paper is organized as follows: OpenCV Library is presented in Section 2. Single Board Computer is given in Section 3. RGB and HSV color space are explained in Section 4. Realized Application for Color Detection and Display is clarified in Section 5 and Conclusion is presented in Section 6.

2. OpenCV Library

OpenCV is an open-source Berkeley Software Distribution (BSD) licensed library developed by Intel organization. It includes a great number of real-time computer vision algorithms. The OpenCV Library is developed using C and C++ programming languages by Intel organization. It runs in MAC OS, Windows, Linux, etc operating systems. A great number of real-time image processing applications can be implemented by using C, C++, Java and etc. programming languages [9, 10].

OpenCV is widely used on single board computer systems. Analyzing objects, security and intrusion systems, camera calibration, military applications, medical image processing, processing of satellite images and map applications are generally use OpenCV and single board computer systems [9,11].

OpenCV is designed to be modular structure and includes a lot of modules. Some modules are listed as follows [12].

- **Core:** This module includes basic data structures. Multi-dimensional arrays and functions used by other modules are included in this module.
- **Imgproc:** This module is an image processing module. Linear and non-linear image filters, geometric image transformations, color space transformations and histograms are in this module.
- **Video:** This is a video analysis module that includes motion estimation, background subtraction, and object tracking algorithms.
- **Calib3d:** This module includes basic multiple-view geometry algorithms, single and stereo camera calibration, object pose estimation, stereo correspondence algorithms, and elements of 3D reconstruction.
- **features2d:** This module includes salient feature detectors, descriptors, and descriptor matchers.
- **Objdetect:** This is the detection object module that uses to recognize face, eyes, nose, car, etc.
- **Highgui:** This module contains user interface GUI, images and video codecs.
- **Gpu:** This module contains GPU-accelerated algorithms.

3. Single Board Computer

A single board computer is designed as a single card to provide all the features of a normal computer. Processor, memory, input/output (I/O) and other units are integrated on the card. Single board computers generally have no card slot (video card, sound card, etc.). The system units (CPU, RAM, etc) are designed to be small and simple for low cost and low power consumption. Single-board computers are generally preferred to develop systems and prototypes for educational studies. System storage unit use flash drive technologies such as disk on module, disk on chip and compact flash [13, 14].

Single board computers have two different architectures defined as slotted and non-slotted. Since Non-slotted single board computers have all the components integrated on a single card, they can be used as a single system itself. Slotted single board computers include basic functions and installing larger systems can work in an integrated manner [13].

Single board computers in industrial applications are usually used to provide interface between the control and communication units. Compared with multi-card computers, single board computers have less space, less power consumption, easier usage and more lightweight.

BeagleBoard, BeagleBoard-xM, PandaBoard, PandaBoard ES, Raspberry Pi, CubieBoard and BeagleBone are widely used single board computers.

3.1. BeagleBoard-xM

BeagleBoard-xM [15, 16, 17] is produced as a single board computer development card having open-source hardware and low power by Texas Instruments (TI) and Digi-Key. The BeagleBoard-xM measures in at 82.55 by 82.55 mm and has a faster CPU core (clocked at 1 GHz compared to the 720 MHz of the BeagleBoard), more RAM (512 MB compared to 256 MB), onboard ethernet jack, and 4 port USB hubs [18]. The distribution of the Linux Operating System specially designed for BeagleBoard is named as Angström Operating System. This operating system is installed in an externally micro SD card. Ubuntu, Android, Fedora, Windows CE and Gentto distributions can also be installed and used in micro SD cards. BeagleBoard-xM can communicate with many kinds of devices using USB port without any problems. It also has a private port for the camera. This port is used to import images fast and simple by Leopard imaging camera. BeagleBoard-xM is shown in Fig. 1.

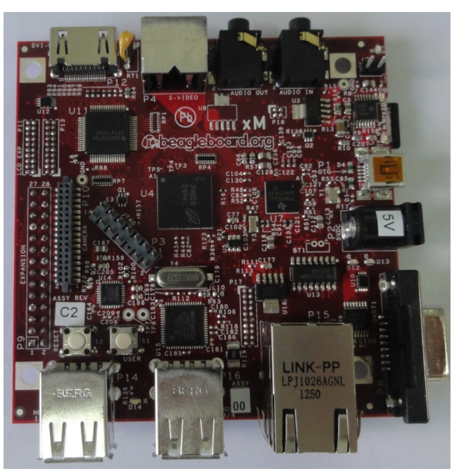

Fig.1. BeagleBoard-xM

In addition, it provides hardware specifications as follows [15].

- TI DM3730 CPU – 1 GHz ARM Cortex-A8 core
- 4 USB ports
- USB OTG (mini AB)
- 512MB @ 166MHz Memory
- uSD Card
- S-Video
- DVI-D
- Ethernet port
- RS-232
- Camera port for leopard imaging
- Stereo in and out jacks

4. RGB and HSV Color Space

Pixels are known as the essential units for images. A Picture is composed of lots of pixels depending on the resolution of the image. In a gray level image, one pixel is defined as 8 bits unsigned values. However, in a color image, the image is composed of three main colors as RGB (Red, Green, Blue). Therefore, three such 8 bits unsigned values are used per pixel for RGB. In other words, while a gray image requires one channel, a color image necessitates three channels for RGB. Since one channel is defined as 8 bits unsigned values, one channel gets values between 0 and 255. So, a color image includes three different such channels getting values from 0 to 255. In a color image, if one pixel gets RGB values as RGB (255, 0, 0), this means the color of this pixel is pure “RED”. If one pixel has RGB values as RGB (0, 255, 0), this pixel color is pure “GREEN”. If 100% of these three main colors are obtained in one pixel, this pixel gets RGB values as RGB (255, 255, 255). This means that the color of this pixel is “WHITE”. Contrarily, “BLACK” gets a pixel including RGB (0, 0, 0) values. If a pixel has the R, G and B values between 0 and 255 individually, different color tones are obtained [11, 19].

HSV is an abbreviation of **H**ue, **S**aturation and **V**alue. It is also named as HSB. In HSB, H and S have the same meaning with HSV and B is for **B**rightness. HSV color space was found out by Alvy Ray Smith in 1978. It is closer to human vision than RGB color space is. It is obtained by non-linear transformation of RGB color space**.** While transformation from RGB to HSV is shown in Equation (1), transformation from HSV to RGB is given in Equation (2) [20]. HSV color space is used to detect the color of the packages passing on the conveyor belt in this study.

$$
\begin{aligned}
&M = \max(R, G, B) \\
&m = \min(R, G, B) \\
&C = M - m \\
&H' = \begin{cases} \text{undefined}, & \text{if } C = 0 \\ \frac{G-B}{C} \bmod 6, & \text{if } M = R \\ \frac{B-R}{C} + 2, & \text{if } M = G \\ \frac{R-G}{C} + 4, & \text{if } M = B \end{cases} \\
&H = 60^\circ \times H' \\
&S_{HSV} = \begin{cases} 0, & \text{if } C = 0 \\ \frac{C}{V}, & \text{otherwise} \end{cases} \\
&V = M
\end{aligned} \tag{1}
$$

$$
\begin{aligned}
&C = V \times S_{HSV} \\
&H' = \frac{H}{60^\circ} \\
&X = C(1 - |H' \bmod 2 - 1|) \\
&(R_1, G_1, B_1) = \begin{cases} (0,0,0) & \text{if } H \text{ is undefined} \\ (C,X,0) & \text{if } 0 \le H' < 1 \\ (X,C,0) & \text{if } 1 \le H' < 2 \\ (0,C,X) & \text{if } 2 \le H' < 3 \\ (0,X,C) & \text{if } 3 \le H' < 4 \\ (X,0,C) & \text{if } 4 \le H' < 5 \\ (C,0,X) & \text{if } 5 \le H' < 6 \end{cases} \\
&m = V - C \\
&(R,G,B) = (R_1 + m, G_1 + m, B_1 + m)
\end{aligned}
\tag{2}
$$

5. Realized Application for Color Detection and Display

In this study, real time color detection system of the packages passing on the conveyor belt is realized. The main purpose of this study is to define and display the number of colored and passing packages on the conveyor belt. The photo of the experimental setup is given in Fig. 2. As shown in Fig.2, the system includes a DC power supply, a conveyor belt, a BegleBoard-xM with LCD7 touch screen and a USB camera.

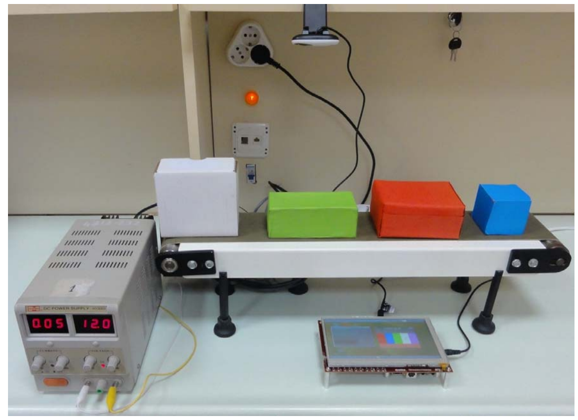

Fig.2. The system used in laboratory environment

The block diagram of the realized system is shown in Fig. 3. In this Figure, the image of the passing package on the conveyor belt is taken by the USB camera. This taken image is sent to Beagleboard-xM. The color of the package is detected by processing this image using OpenCV. The obtained results are displayed on the LCD7 touch screen.

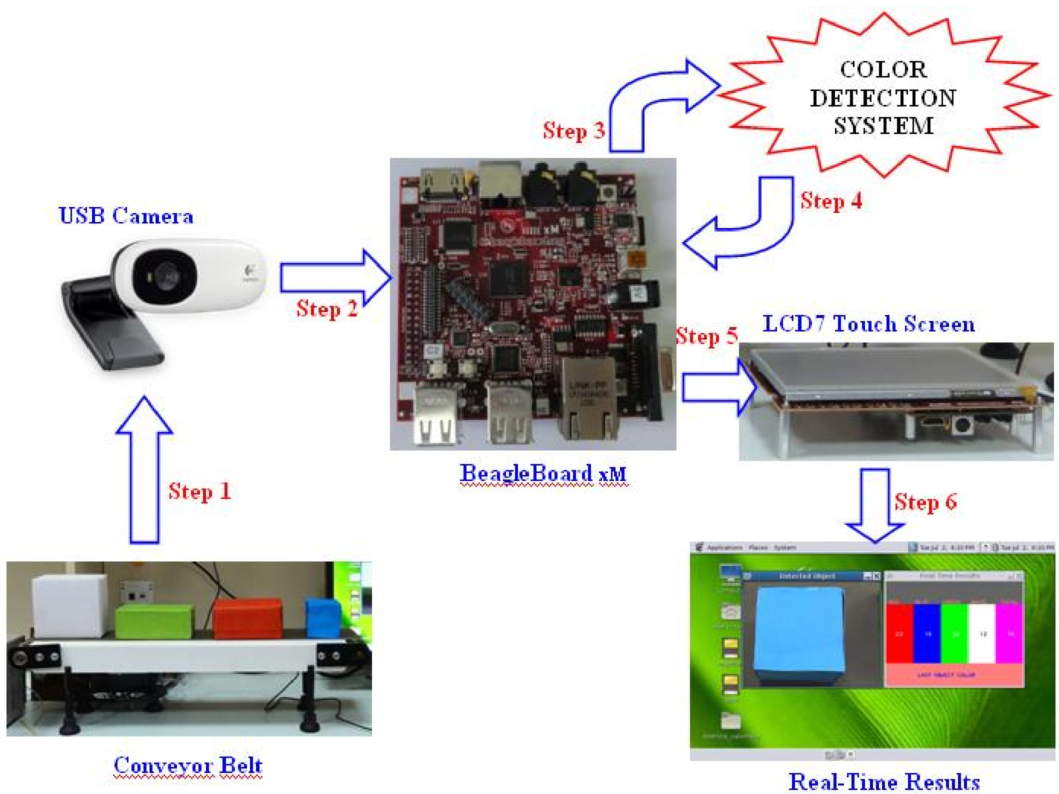

Fig. 3. The block diagram of the system

As shown in Fig. 3, four different colored packages as red, green, blue and white are used as products passing on the conveyor belt. The sizes of the packages are prepared different from each other to be able to detect the color of the packages in different sizes. The time interval between these packages on the conveyor belt is taken as 5000 ms. This time interval depends on the velocity of the conveyor belt. It is defined during the experimental studies. The used conveyor belt is supplied by 12 V DC voltage supply. The conveyor belt is on while getting the images of the colored packages by the USB camera. Therefore, the time interval of the image of the package is also taken as 5000 ms. The USB camera is placed in an appropriate distant from the conveyor belt. Since protecting the BeagleBoard-xM from using unnecessary system source as CPU and RAM, the required area of the colored package image is taken by the USB camera.

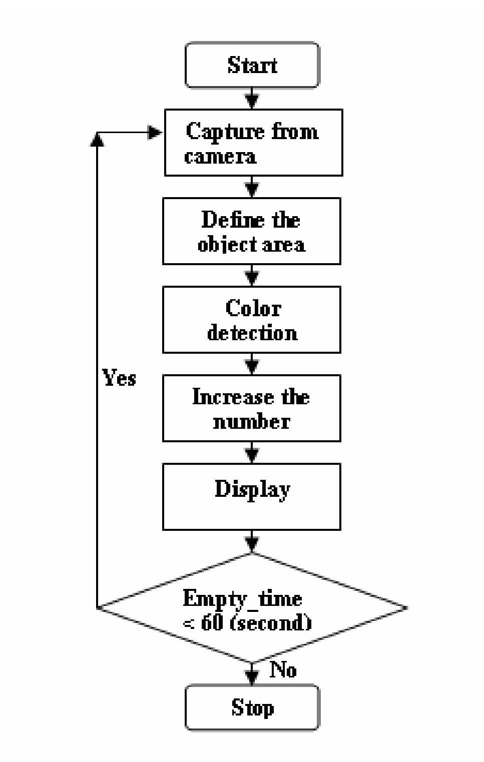

Fig. 4. Flowchart of the system

Color detection process is realized in BeagleBoard-xM using the flowchart shown in Fig. 4. Related area of the image taken as RGB by the USB camera is converted to HSV color space using Equation (1). The color detection of the related package is realized using H, S and V values. After color detection of the package is realized using this image, both the number of the related package and the total package number are increased. The increased number of the related package and total package number are updated in the table shown on the LCD7 touch screen. In Fig. 5 and Fig. 6, the blue colored package and the real time result table shown on the LCD7 touch screen are given respectively. This table is designed as 3 rows. The middle of these 3 rows is divided into five columns. The first four columns show both the color and the number of four different related colored packages. The fifth column shows the total package number passed on the conveyor belt up to that time. The last package color is also shown in the third row as "LAST OBJECT COLOR". If no package passes on the conveyor belt during 60 seconds, the system is turned off.

Fig. 5. Blue colored package

Fig. 6. Real time result table

6. Conclusions

An industrial work on real time color detection and counting of the colored packages passing on the conveyor belt is realized in this study. During this study red, green, blue and white colored packages are used as sample products. Every tone of color can be used in this system. Experimental studies are performed under stable light conditions. A quite successful performance is obtained from the experimental studies. This study is a good example of a single board computer based industrial application. This work detects, counts and displays different size and colored packages passing on conveyor belt sequentially by using only one conveyor belt. The most important gain in this study is real time distinguishing different kinds of products passing on the same conveyor belt sequentially.

7. Acknowledgements

Kocaeli University Scientific Research Projects Unit supports this study with the project number as BAP 2013/68HDP.

8. References

[1] Liu L., Sang N., Yang S., and Huang R., “Real-Time Skin Color Detection under Rapidly Changing Illumination Conditions”, Consumer Electronics, IEEE Transactions, 2011, Volume: 57, Issue: 3, PP: 1295 – 1302.

[2] Yang G., Li H., Zhang L., and Cao Y., “Research on a Skin Color Detection Algorithm Based on Self-adaptive Skin Color Model”, Communications and Intelligence Information Security (ICCIIS), 2010 International Conference, 2010, pp:266-270.

[3] Hosseini, S.M. ; Farsi, H. ; Amjadi, S. , “A Novel Color Space Creating Method Applied to Skin Color Detection”, Digital Image Processing, 2009 International Conference, 2009, pp:17-21.

[4] Yang D., Chen C.W., C.H., Chang Y.C., Ming-Der Shieh M.D., Wang J., and Lo C.C., “Face detection architecture design using hybrid skin color detection and cascade of classifiers” , Circuits and Systems (APCCAS), 2012 IEEE Asia Pacific Conference , 2012, PP:543 – 546.

[5] Sagheer, A. and Aly, S., “An Effective Face Detection Algorithm Based on Skin Color Information ”, Signal Image Technology and Internet Based Systems (SITIS), 2012 Eighth International Conference , 2012, pp: 90 -96.

[6] Li Z., Xue L., and Tan F., “Face detection in complex background based on skin color features and improved AdaBoost algorithms”, Progress in Informatics and Computing (PIC), 2010 IEEE International Conference, 2010 , pp:723 - 727 .

[7] G. Li, R. Yuan, Z. Yang and X. Huang , “A Yellow License Plate Location Method Based on RGB Model of Color Image and Texture of Plate” , Digital Media and its Application in Museum & Heritages, Second Workshop on , pp.42-46, 10-12 Dec. 2007.

[8] Soendoro D., and Supriana I., “Traffic sign recognition with Color-based Method, shape-arc estimation and SVM ”, Electrical Engineering and Informatics (ICEEI), 2011 International Conference, 2011, pp:1-6.

[9] Bradski, G. and Kaehler, A., “Learning OpenCV: Computer Vision with the OpenCV Library”, O’Reilly Media, USA, 2008

[10] Wikipedia, 2013 July 8, “OpenCV” Available: http://en.wikipedia.org/wiki/OpenCV

[11] Culjak I., Abram D., Pribanic T., Dzapo H., and Cifrek M., “A brief introduction to OpenCV”, MIPRO, 2012 Proceedings of the 35th International Convention , Opatija, Croatia,2012, pp: 1725 – 1730

[12] OpenCV API Refences, 2013 July 8, “OpenCV: Introduction” Available: http://docs.opencv.org/modules/core/doc/intro.html

[13] Wikipedia, 2013 July 9, “Single-board computer”, Available: http://en.wikipedia.org/wiki/Single-board_computer.

[14] Alee N, Rahman M., and Ahmad R.B., “Performance Comparison of Single Board Computer: A Case Study of Kernel on ARM Architecture”, The 6th International Conference on Computer Science & Education (ICCSE 2011), SuperStar Virgo, Singapore, 2011, pp: 521-524

[15] Beagleboard.org, 2010, “BeagleBoard-xM Rev C System Reference Manuel”, Available:http://www.beagleboard.org, USA

[16] Shen Khang Teoh, Vooi Voon Yap, Chit Siang Soh, Sebastian, P. , "Implementation and optimization of human tracking system using ARM embedded platform," Intelligent and Advanced Systems (ICIAS), 2012 4th International Conference on , vol.1, no., pp.353-356, 12-14 June 2012

[17] Patil H.Y., Bharambe S.V., Kotari A.G., and Bhurchandi K.M., “Face Localization and its Implementation on Embedded Platform”, Advance Computing Conference (IACC), 2013 IEEE 3rd International , 2013 , Pp: 741 – 745

[18] Wikipedia, 2013 July 12, “BeagleBoard”, Available: http://en.wikipedia.org/wiki/BeagleBoard

[19] Wikipedia, 2013 July 10, “RGB color model”, Available:http://en.wikipedia.org/wiki/RGB

[20] Wikipedia, 2013 July 11, “HSL and HSV”, Available:http://en.wikipedia.org/wiki/HSL_and_HSV.