Journal of Engineering and Applied Sciences 13 (3): 760-765, 2018
ISSN: 1816-949X

Implementation of Stand-Alone Embedded Image Processing Device

[1]Heon Jeong and [2]Ho Young Lee
[1]Department of Fire Service Administration, [2]Department of Drone System,
Chodang University, Muan-ro, Korea

Abstract: We develop a Stand-alone Embedded-based Image Processing Device (SEIPD) with vision inspection program for customized factory automation. SEIPD designed in this study composed of raspberry pi as main board, I/O board and webcam. Developed image processing devices will act as smart manufacturing applications for product quality management. It is applied to the system which judges the character position and shape printed on the green case what is tire the semi-finished product in the tire manufacturing process as the actual application target of the developed product and determines whether or not the character is normal. SEIPD was developed in a small size to solve the spatial problem. It also designed to be robust enough to withstand less external influences.

Key words: Embedded-based, image processing device, inspection program, character recognition, process, vision inspection

INTRODUCTION

There are many fields in which computer vision is involved including terrestrial and aerial mapping of natural resources, crop monitoring, precision agriculture, robotics, automatic guidance, non-destructive inspection of product properties, quality control and classification on processing lines and in general, process automation. Especially, vision system using image processing technology has been applied to automation of factory automation in various fields. Through the application of this image processing technology it has been possible to accurately and quickly measure the parts that were manually measured, judged and processed by people in the past. As a result, quality and productivity were improved (Cubero *et al*., 2011; Malamas *et al*., 2003).

Most factory automation vision systems are computer-based. The computer has been developed for general use. Therefore, such a computer-based vision system is somewhat lacking in durability against a harsh environment at a manufacturing site and there is an inconvenience that maintenance such as OS-based program update is required. In addition, the vision inspection system can perform high speed processing and various image analysis functions but it has disadvantages such as being restricted by complicated structure and field installation conditions and being expensive (Al-Jumma, 2017; Nagaraja *et al*., 2015).

In this study, we develop a Stand-alone Embedded-based Image Processing Device (SEIPD). The system consists of main board using Raspberry Pi, I/O board and webcam camera into standalone. The main board can be connected with webcam to capture real-time image and recognize the state of the object to be detected by applying various image processing algorithms. The obtained result interfaces with an external device such as PLC through the IO board. In addition, the proposed device was developed in a very small size and designed to be robust enough to withstand less external influences.

As a feature, it is possible to solve spatial problem and costly parts. When implementing automation, it is often encountered that the device must be applied in a very small space. In addition, spatial freedom may be required for accurate image acquisition. The proposed system is designed with embedded system considering the spatial constraints.

SEIPD was applied to the tire curing process. In this study, we developed a character recognition system for green case which can be installed in the field condition of narrow space. The green-case character detection system proposed in this paper represents a standalone embedded type and OpenCV image processing library is used. The proposed device is a system that inspects the green case (semi-finished product of tire) before putting it on the curing machine. The tire's specification number is printed on the tread portion of the green case. The verification test contents of the proposed device are to determine the position of the printed character and whether it is the shape of the character. The developed product was installed in 6 places and operated for about 1 month to evaluate the reliability of the product.

Corresponding Author: Ho Young Lee, Department of Drone System, Chodang University, 380 Muanro, Muaneup, 58530 Muangun Jeollanamdo, South Korea

MATERIALS AND METHODS

Design of the independent embedded image processing device

Embedded image processing device: The image processing device designed in this study is composed of raspberry pi (main board) I/O board and webcam and designed to perform the following functions.

Raspberry Pi (main board): Debian OS, image processing algorithm built-in I/O board and interface function, webcam interface function.

I/O board: Implemented to interface with external devices such as PLC. It is largely divided into five parts. It consists of a lighting control part, a power supply part an external input part an external output part and a display part. The 4 port photo-coupled INPUT, 2 ch relay output, FND, LED display (Fig. 1):

- Webcam: resolution 1024×720
- Program implementation: using OpenCV of image processing library

Software specifications: The operating system under which the proposed project is executed is Raspbian which is derived from the Debian operating system (Zidek and Hosovsky, 2014). The algorithm were developed based on Codeblock which is a free, open-source cross-platform IDE that supports multiple compilers including GCC, Clang and Visual C++. C ++ was used as main language. The functions in algorithm are called from the OpenCV library. OpenCV is an open source computer vision library which is written under C and C++ and runs under Linux, Windows and Mac OS X. OpenCV was designed for computational efficiency and with a strong focus on real-time applications.

Raspberry Pi (main board): Raspberry pi is a SoC (System on Chip) that integrates several functional components into a single chip or chipset. The SoC used in Raspberry Pi 2 is the Broadcom BCM2836 SoC multimedia processor (Zidek and Hosovsky, 2014). The CPU of the Raspberry Pi contains an ARM Cortex-A7 900 MHz processor which makes use of the RISC architecture and low power draw. It is not compatible with traditional PC Software. Hence, it has to be connected to a monitor separately. Raspberry pi has an on-chip DSP processor which is used to perform the floating point operations. It facilitates development of multi-processor designs with large numbers of controllers and peripherals. The GPIO pins of the Pi differ by the model. In model B there are 40 pins, out of which there are 4 power pins and 8 ground pins. Rest of the pins is used as GPIO's. The networking capabilities of the Pi can be used as a wired Ethernet (IEEE 802.3) or the wireless IEEE 802.11 Wi-Fi. Raspberry pi has an internal memory of 1GB RAM and external memory is extendable upto 64 GB.

IO board circuit: The block diagram for the IO board is shown in Fig 2. The IO board is based on the CPU of ATmega128 and transmits I/O signals from the PLC to the main board and is designed to perform communication functions such as FND display device and RS485. It also controls the LEDs of the lighting function.

Green-case character detection system

Necessity of green-case character detection system: The semi-finished product immediately before the curing

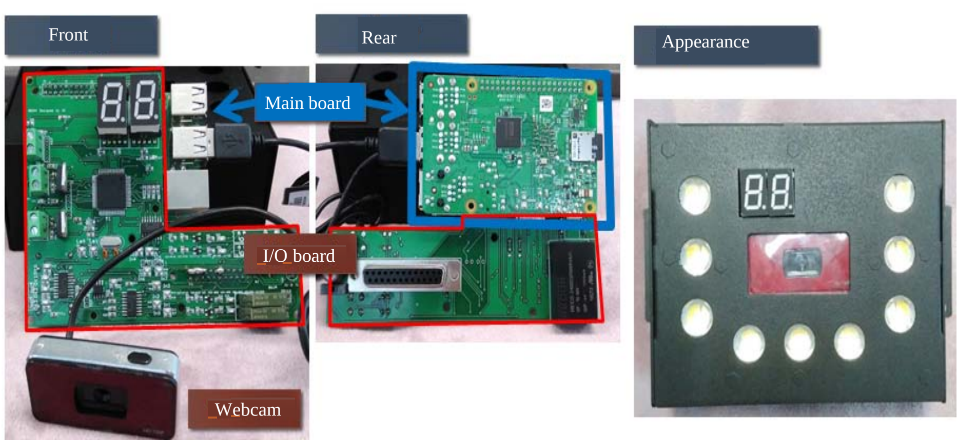

Fig. 1: Design of Stand-alone Embedded-based Image Processing Device (SEIPD)

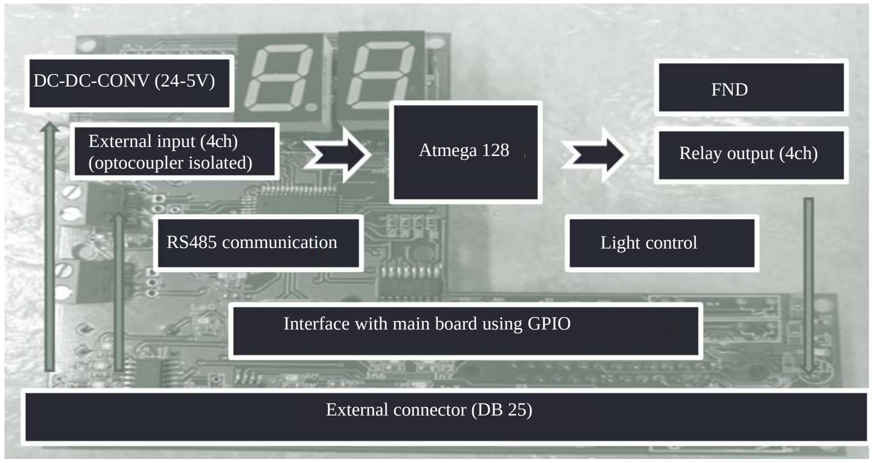

Fig. 2: Block diagram of IO board

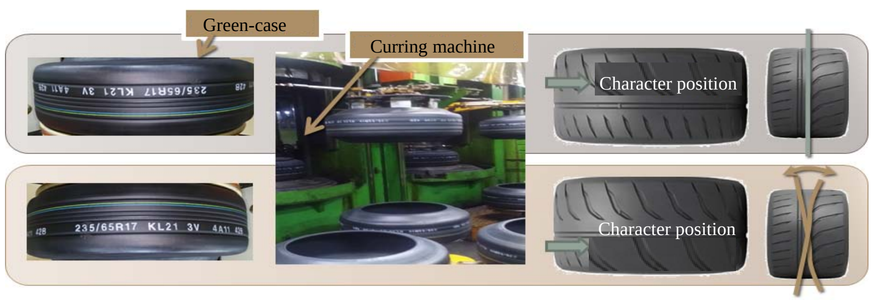

Fig. 3: Necessity of green-case character detection system

process which has the shape of a tire in the tire manufacturing process is called a green-case. Before putting these green-cases into the curing machine it is necessary to first check whether the letters of the green-case are in the forward direction or in the reverse direction. If the green-case is turned upside down, it will produce a defective tire. Therefore, the image processing product is required to confirm the state of the green-case character before putting it on the curing machine. In this study, we developed a character recognition system for green-case which can be installed in the field condition of narrow space (Fig. 3).

Operation sequence of green-case character detection system: The operation sequence of the proposed device is as follows: the capture start signal is input from the PLC installed in the curing machine. Take a picture of the green case through the webcam (image-off) Turn on the device's own illumination and retake the green case (image-on). Apply various image processing algorithms (character area detection, image template matching, find blob, etc.). Output to PLC the result of forward/reverse status of character (Fig. 4).

Image processing algorithm implementation: First, the forward and backward directions of the character are extracted and the result is determined. If it is difficult to determine the outcome due to difficulty in character extraction it is determined from the top and bottom of the position where the character is placed based on the center of the green case. The overall program process is as follows:

- Read a predetermined forward/reverse font from a file
- Load the selection switch value of the character forward/reverse and upper/lower set value
- Capturing the green case image using the light control and webcam
- The green case area is extracted through the preprocessing process. [Extraction result: G-area-Pos-U (Upper position), G-area-Pos-D (lower position), G-area-Pos-C (Center position)]
- Extract the character string area from the obtained green case area [Extraction result: Char_pos (position of character)]
- Extract the character region from the string

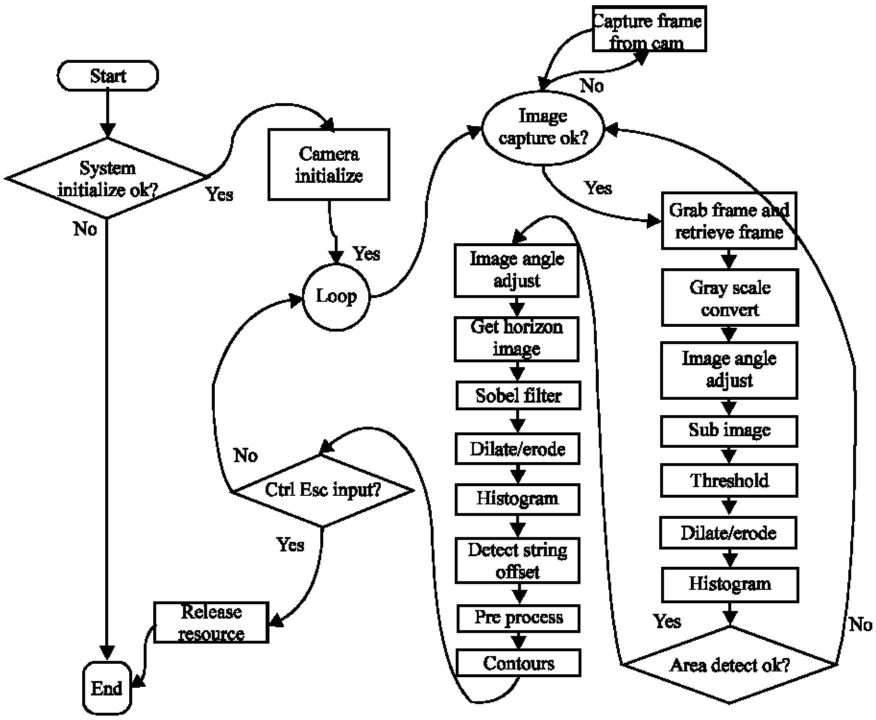

Fig. 4: Flowchart of green-case character detection system

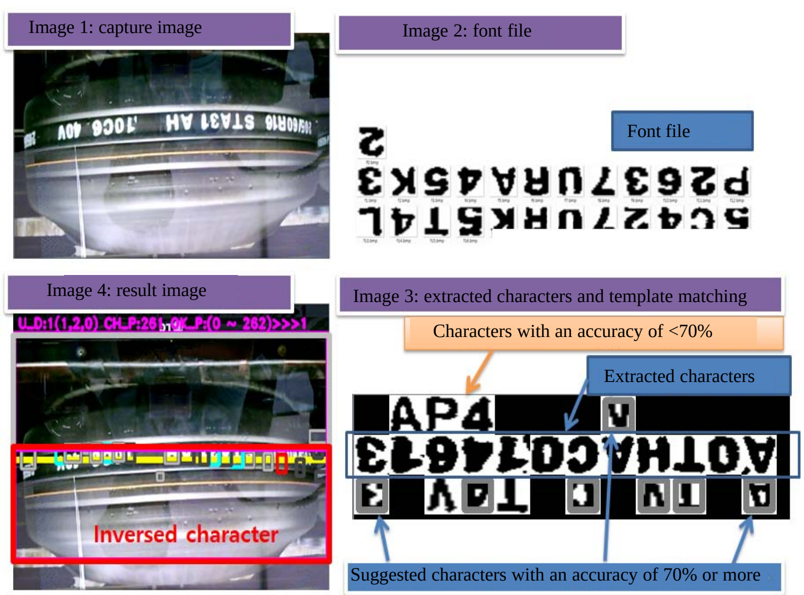

Fig. 5: Image processing algorithm implementation

- The extracted characters are subjected to font image and template matching process
- Extract fonts with matching similarity of 70% or more
- Compare the number of valid fonts in the forward direction (N-forward-valid) and the number of valid fonts in the reverse direction (N-reverse-valid)
- If there are two or more valid number differences, output positive/negative results

Otherwise, the result is output through extracted Char-pos and G-area-Pos-C. The preprocessing process to extract the character region is shown in Fig. 5 and 6:

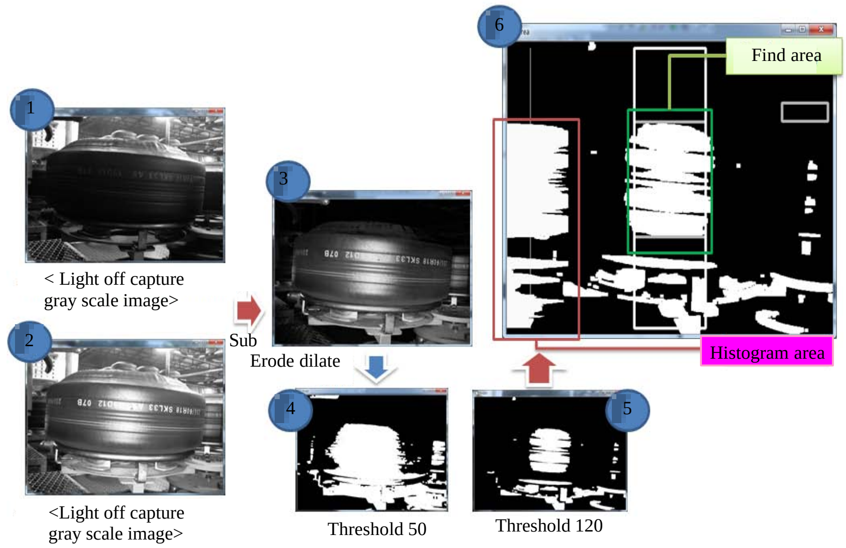

Fig. 6: Preprocessing algorithm for green-case character detection system

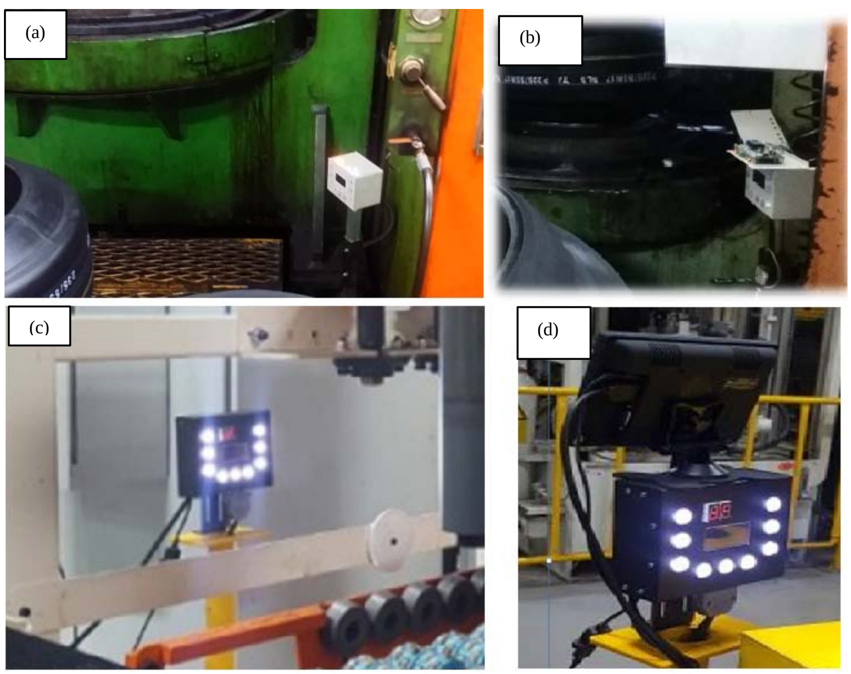

Fig. 7: a-d) Experiment for green-case character detection system

- Obtain light-off① and light on②
- Delete the part other than the area of interest③ through the sub function
- Eliminate noise through Erode/Dilate④
- Change the threshold value to extract the green case area of interest⑤
- Extract vertically drawn case area⑥ through vertical project

RESULTS AND DISCUSSION

Experiment: The developed product was applied to the tire curing process. The developed product was installed in 6 places and operated for about 1 month to evaluate the reliability of the product (Fig. 7). More than 10,000 capturing experiments were performed and about 99.9% of recognition results were obtained. The accuracy of area detection was 98% and the recognition rate of character recognition was 70%. In order to increase the recognition rate, various algorithms are applied in parallel (Fig. 7).

CONCLUSION

In this study, we developed an independent embedded image processing device. The developed product was made by combining the enclosure design, Raspberry board, I/O board and webcam camera and applied OpenCV for embedded image processing. For the performance evaluation, it was applied to the green-case detection system at the tire manufacturing site. As a result, the performance was about 99.9%. As a result, development products with spatial and cost features are expected to be used efficiently in various factory automation areas.

ACKNOWLEDGEMENT

This research was supported by the Technology Development Program (S2524097) funded by the Ministry of SMEs and Startups (MSS, Korea).

REFERENCES

Al-Jumma, L.M., 2017. Development of an embedded system with machine vision for quality inspection and process automation. Master Thesis, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh.

Cubero, S., N. Aleixos, E. Molto, J. Gomez-Sanchis and J. Blasco, 2011. Advances in machine vision applications for automatic inspection and quality evaluation of fruits and vegetables. Food Bioprocess Technol., 4: 487-504.

Malamas, E.N., E.G.M. Petrakisa, M. Zervakis, L. Petit and J.D. Legat, 2003. A survey on industrial vision systems, applications and tools. Image Vision Comput., 21: 171-188.

Nagaraja, L., R.S. Nagarjun, N.M. Anand, D. Nithin and V.S. Murthy, 2015. Vision based text recognition using raspberry Pi. Intl. J. Comput. Appl., 1: 1-3.

Zidek, K. and A. Hosovsky, 2014. Image thresholding and contour detection with dynamic background selection for inspection tasks in machine vision. Intl. J. Circuits Syst. Signal Process., 8: 545-554.