4

by 4 4

Submission date: 16-Jul-2020 01:15PM (UTC+0800)
Submission ID: 1358111157
File name: 4.4.pdf (297.55K)
Word count: 2055
Character count: 10632

Detection of ATM Users' Helmet with Mini PC Raspberry Pi- Based Template Matching Method

Derisma[#1], Fikral Farhan[*2], Firdaus[#3]
[#1,2]Computer System, Universitas Andalas, Padang, West Sumatera, Indonesia, 25163
[#3]Politeknik Negeri Padang, West Sumatera, Indonesia, 25164

Abstract *One of the rules into the room ATM (Automatic Teller Machine) is the prohibition of using a helmet. To avoid violation on the helmet rule, a system to detect and limit the movement of helmet users who are going to enter ATM room is needed. The design of this system consists of Raspberry Pi, Raspberry Pi Camera and Servo Motor. For decision-making, the system uses Template Matching Method in performing image processing. The system is able to distinguish helmet users and non-helmet users form the value of template matching helmet. The average success rate of indoor helmet detection reached 100%, while the outdoors test reached 62.5%, 37.5%, and 75% respectively with different helmet type and colors.*

Keywords — *ATM, template matching, image processing, helmet detection*

I. INTRODUCTION

Today, door of ATM (Automatic Teller Machine) room can be opened freely by users who will perform a transaction. However, there is a rule that prohibit ATM users from wearing a helmet when entering ATM room. This rule is designed to prevent crime. However, ATM users still frequently violate the rule despite their awareness of being supervised by CCTV (Closed Circuit Television) and security guards. Today's ATM security t still needs improvement, one of which is by using image processing for helmet detection use by customers. This system compares helmet pattern and ATM users' head. Door controlling performed by camera will be more effective to decrease rule violation of entering ATM room.

II. THEORICAL FRAMEWORK

The definition of imaging is a representation of image, likeness or imitation of an object [1]. Open CV is abbreviation for Open Computer Vision which is an open source libraries specialized to perform image processing [2,4, 7]. Template matching [6] is a technique in digital image processing to locate small part of picture that matches image template.

Below are methods in the Template Matching in OpenCV [7]:

a. Squared Differences Matching Method (CV_TV_SQDIFF)
This method matches the squared difference, therefore, the perfect pair is 0 and incorrect pair will be big.

$$R(x,y) = \sum_{x',y'} (T(x',y') - I(x+x',y+y'))^2 \qquad (1)$$

b. Correlation Matching Method (CV_TV_CCORR)
This method matches template multiplication with picture, therefore, the perfect pair to will be big and non-good pair will be small or 0.

$$R(x,y) = \sum_{x',y'} (T(x',y') \cdot I(x+x',y+y')) \qquad (2)$$

c. Correlation Coefficient Matching Method (CV_TV_CCOEFF)
This method matches relative template with the average toward relative image to the average, therefore, the perfect match is 1 and the imperfect pair is -1; the value of 0 means no correlation (random sequence).

$$R(x,y) = \sum_{x',y'} (T'(x',y') \cdot I(x+x',y+y')) \qquad (3)$$

d. Normalization methods
Of the three methods described, there is also a version of the normalization initially developed by Galton and described by Rodgers [7]. Normalization method is useful because, as mentioned earlier, it can help to reduce the impact of differences in lighting between templates and images. In each case, the correlation coefficient is the same:

$$R(x,y) = \sqrt{\sum_{x',y'} T(x',y')^2 \cdot \sum_{x',y'} I(x+x',y+y')^2} \qquad (4)$$

The working system outline of Template Matching Method is as follows:

1. Loading Process / loading input image and the image path (template).
2. Performing template matching procedure by using OpenCV function template match with one of the 6 methods.
3. Users can choose methods by entering the ditrackbar selection.
4. Normalizing output from matching procedure.
5. Localizing in a place that enables higher matching.
6. Loading a rectangle around the region of the highest suitability.

Raspberry pi camera module can be used to take high definition videos and photos. It is very easy to use for beginners, and offers many advanced features to expand the user creations. There are many examples of online people who use it to time-lapse, slow-motion and other video skills [3,5].

III. RESEARCH METHODOLOGY

Program for Raspberry Pi is designed so that Raspberry Pi can take a picture / video frame from the camera module and processing the frames to take information about the questioned object. After receiving information, the program will make decisions and give orders to the GPIO to take that next step. The program in the Raspberry was created by using Raspbian, Python, OpenCV.

1. Start: Active Camera.
2. Load Template Image: At this stage, it will load the image on a template that contains images that we will explore.
3. Input image: The image will be fed to raspberries after captured by the camera.
4. Object Search: The object will be searched with the template image that resembles the input image. The object is searched by using the squared difference matching method (CV_TM_SQDIFF).
5. If ≠ Object Template: This program will see if the detected object is different from the pattern of the existing helmet on the template. If the object is different, then it will be forwarded to the next process (doors open).
6. If Object = Template: If the program sees the object detected is the same as helmet patterns that exist in the template, the door does not open.
7. Output: Open Door.
8. Finished

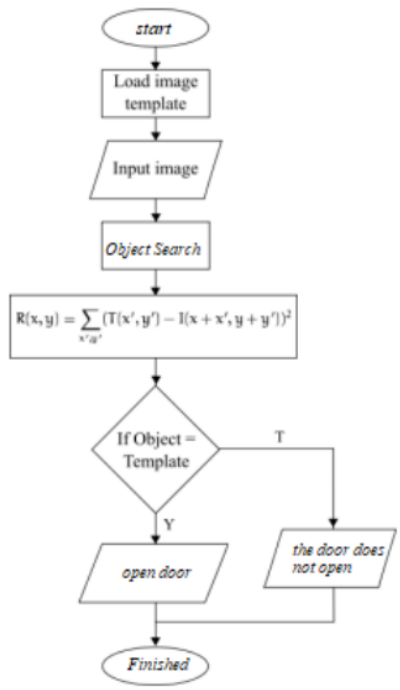

Fig. 1 Flowchart Program on Raspberry Pi

Tools needed in this research are in the form of hardware and software. Hardware and software used in the study are as follows:

1. Hardware
   - a.Laptop / PC
   - b.Mini PC raspeberry Pi
   - c.Camera module
   - d.Servo Motor
2. Software
   - a.Operating System: Raspbian
   - b.Programming Languages: Python
   - c.Library: Open CV

From the results of hardware and software design in this research, a system that can detect ATM users helmet by using CameraRaspberry then built. Raspberry Camera is placed on the side of the ATM prototype that is used to capture the ATM user.

System design of this tool consists of three main parts: raspberry pi, raspberry camera and servo motors. The figure captured through raspberry camera is then stored in the raspberry pi SD card to do template matching process, using cv2.TM_SQDIFF. When they are completed, raspberry pi template matching process will proceed output in the form of a servo motor that rotates 90 degrees to open the door on ATM prototype space.

IV.IMPLEMENTATION

4.1 Implementation of Hardware

The implementation of hardware of this tool uses some tools, they are: Raspberry to capture images from which pictorial patterns is made, Raspberry Pi is used as a place to process all the workings of the tool, Servo Motor is the output on automatic doors

Fig. 2. Implementation of Hardware

4.2 Implementation of Template Matching

Implementation of template matching is made by using OpenCV and Python. In this implementation, there are several image processing techniques used. Some of them are cutting (cropping) image, template matching.

4.2.1 Image Cutting (cropping)

OpenCV is used to connect the Raspberry camera with raspberry pi. In OpenCV there are several programs to enable Raspberry Camera. In the process of template matching, process of cutting an image to be used as a template is necessary in order to use the maximum results.

4.2.2 Resize

In this process, once the cameras capture the images of users who are going to enter ATM room, the camera automatically resizes the image size. The purpose of this resizing is to accelerate the work of raspberries for the picture matching process resulted from image capture and template for execution to servo motor.

4.2.3 Template matching cv2.TM_SQDIFF

In template matching process, the template used is in the form of a helmet to be matched to the image. Template matching in this method is matched by comparing the lowest values of each pixel. The matching process is carried out by getting the image size (WxH) and size of the template (WxH) as well as process templates cv2.TM_CCOEFF_NORMED ma tching to determine the value of R (result), R = (W + w - 1) x (H + h -1), w = x ', h = y', W = x, H = y, with formula (1).

The test was carried out in an open space by using three different types and color of helmets and different. At the time of image capturing, the position of the helmet changed with distance less than 150 cm - 400cm. The output of the servo motor showed that CV2.TM_SQDIFF value was limited from - 7.5 million for the closed state and more than 7.5 million to an open state.

Tests by using a black helmet was performed outdoors with lighting (130 lux), with eight times tests, the success rate reached 62.5 percent. The rate of success was obtained from successful trials conducted divided by the numbers of the trials and multiplied by 100%. The result was obtained because of the sufficient light intensity when image capture took place that the matching template process can detect the template (helmet) in accordance with the expected results and distance when ideal image capturing took place. With 62.5 percent of success rate, it is concluded that the tool is capable of detecting a dark colored helmet template well.

Tests by using a black helmet was conducted in a room with lighting of 60 lux, from eight trials, the success rate of object detection reached 100 percent. Although the light intensity decreased when image capture took place, the system still can detect the difference between helmet template and background still looked clear. Test by using white half face helmet was performed outdoors with lighting of 130 lux, from eight trials, the success rate of object detection reached 37.5 percent. The low level of success was due to the bright (white) helmet color, making detection difficult. Test by using black half face helmet was conducted in a room with 130 lux lighting, from ten trials, the success rate of object detection reached 100 percent. The success was due to sufficient light intensity when image capture took place that template matching process can detect the template (helmet). Test with black red and white retro typed helmet was conducted outdoors with lighting of 130 lux. From eight times trials, the success rate of object detection reached 75 percent. This is due to sufficient light when image capturing took place so that the matching template process could detect template (helmet) and distance when capturing ideal image.

V. CONCLUSIONS

Based on the research carried out by applying the design, it can be concluded that:

1) The system will detect helmet by comparing the captured image by using the template matching method.
2) Different coordinates of the helmet users does not affect the template matching process, the influence is light and background when capturing the image.
3) The ideal shooting distance to get the maximum matching is 160-240 cm.
4) Detection will be obtained from the value of the template matching. The value will set the limits for the decision, whether the door will open or remain locked.

For the development of mini pc-based ATM automatic door in the future, the following suggestions that can be used as a reference:

1) Fix lighting is needed to detect helmet.
2) Background with a bright color (white) is needed to ease the detection of helmet.

REFERENCES

[1] Munir, Rinaldi. 2004. Pengolahan Citra Digital dengan Pendekatan Algoritma. Bandung : Penerbit Informatika

[2] Viola, P., Jones, M. J., "Rapid Object Detection Using A Boosted Cascade of Simple Features", IEEE Conference on Computer Vision and Pattern Recognition, Jauai, Hawaii, 2001.

[3] Bauer, Peter. 2013. Raspberry Pi Computer. USA, bitkistl, Inc.

[4] Arihutomo, Mukhlas. 2010. Rancang Bangun Sistem penjejakan Objek Menggunakan Metode Viola Jones Untuk Aplikasi Eyebot. Proyek Akhir Jurusan Teknik Telekomunikasi Politeknik Elektronika Negeri Surabaya-ITS, Surabaya.

[5] http://www.raspberrypi.org/products/camera- module/ diakses tanggal 20 Oktober 2014

[6] OpenCV. 2011. Template Matching, http://docs.opencv.org.doc.tutorials/imgproc.hi stograms/template_matc hing/template_matching.html. Diakses tanggal 20 Oktober 2014.

[7] Gary B, Adrian K. 2008. Learning OpenCV: Computer Vision with the OpenCV Library. United Stated America : O'Reilly Media, Inc.

ORIGINALITY REPORT

8%	4%	2%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.ijcotjournal.org Internet Source	2%
2	Submitted to University of Wolverhampton Student Paper	1%
3	Submitted to Higher Education Commission Pakistan Student Paper	1%
4	Submitted to Swinburne University of Technology Student Paper	1%
5	Su, Kuo Lan, Jr Hung Guo, and Kuo Hsien Hsia. "Development of a Tracking System on the Mobile Robot", Applied Mechanics and Materials, 2015. Publication	1%
6	Submitted to Universitas Sultan Ageng Tirtayasa Student Paper	1%
7	Submitted to Universiti Tenaga Nasional Student Paper	<1%

8	Submitted to Nottingham Trent University Student Paper	<1%
9	Submitted to University of Nottingham Student Paper	<1%
10	Priya Singh, Milind Sukram Suryawanshi, Darshana Tak. "Smart Fleet Management System Using IoT, Computer Vision, Cloud Computing and Machine Learning Technologies", 2019 IEEE 5th International Conference for Convergence in Technology (I2CT), 2019 Publication	<1%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		