Implementation of Smart Parking System in Telkom University.Subsystem:Digital Image Processing with Canny Edge Detection and Embedded System

[1]Restu Aulia Ulfah, [2]Agus Virgono, [3]Agung Nugroho Jati
Department of Computer Engineering
Telkom University
Bandung, Indonesia
[1]restuaulia@students.telkomuniversity.ac.id , [2]avirgono@telkomuniversity.ac.id, [3]agungnj@telkomuniversity.ac.id

***Abstract*— Four-wheel vehicle requires a wider space so the driver must surround the parking area to get a parking spot. In this final project will be implemented smart parking system with car parking availability detection based digital image processing and Raspberry pi. Enter the data from usb camera in the form of a model and then made the conversion RGB image into grayscale color then performed thresholding to produce binary image output to be sent to the server via a LAN. The image processing system using the Canny edge detection. Based on tests performed gained an average percentage of data consistency in detecting signs of the triangle by 83 333% and the system's success in ensuring that the information obtained in accordance with the database on the server data on Raspberry pi has a percentage of 92%. Time computing system in detecting three marks reaching 46.8 ms, the detection of two marks for 36.1 ms, and the detection of a sign that is 33.2 ms. Different light conditions affect the stability of the system in detecting signs. Stable system in detecting the mark on the light intensity value range of 6000 - 4000 luxmeter.**

keywords - Digital image processing, Raspberry pi, Local Area Network, Canny edge detetction, availability of parking.

I. INTRODUCTION

Based on data from the Traffic Police Corps of the Republic of Indonesia increase in the number of cars rose 11% course this will lead to problems in finding an empty parking space [1]. On campus parking lot is usually the driver must find their own parking area is empty, the driver must revolve parking in advance if the location of the car park is very spacious. Such problems can be avoided if the driver has to know the condition of the empty parking area. Seeing the problems that have been presented, the authors intend to make "Implementation of smart parking system in Telkom University; Digital image processing subsystem and Raspberry pi ". This parking system is expected to help the driver or the outdoor car park in order to get a proper parking area and should not be circling looking for parking space is empty.

This study aims untuk.mendeteksi mark on the slot parking area in order to provide information to the user about the availability of parking slots are empty. Usb camera is placed on the parking location with a height and a certain angle in order to reach the parking slots in accordance with the display on the monitor connected to the Raspberry Pi. At each slot parking lot made a sign that would be a sign of image acquisition. To find out a parking area there is a vacant parking slot which can be done by putting a camera that can detect signs of the parking slot. Sign at a parking slot can be detected by performing image processing on the captured digital camera is connected Raspberry pi as embedded systems. Delivery of data via Local Area Network (LAN) and then the server sends an empty parking slot information to the driver via the android based smartphone which has installed smart parking applications.

II. RELATED WORK

A. *Smart Parking System*

Smart system means the system which is smart, smart has a sense of being able to do something well, organized and tidy in accordance with applicable rules, and be able to absorb information better and faster as a result of a study carried out by the system. Smart parking system is or is part of a smart city that is set in terms of governance of the parking area so that a more orderly and efficient [8]. In this study the management of the parking area is intended to provide information to motorists about the availability of parking in a parking area in a campus.

B. *Citra RGB*

Each pixel in an image has a specific color which is a combination of three colors: red, green, blue is often called RGB image. If there is a color other than it is the result of the combination of colors of the three colors. Each primary color has a value range between 0 to 1505. In this way will be obtained as mixed color 1506x1506x1506 = 16777216 kinds of colors and each pixel requires 3 bytes. [2]

C. *Citra Grayscale*

Grayscale image is an image that has a range of shades between black (minimum) and white (maximum). The maximum number of colors in accordance with a bit of storage used. If the 4-bit gray scale, the number of possibilities is 24 = 16 colors, with the possibility of color 0 (as a minimum) and

DOI : 04.IJAI.2015.1.5

15 (a maximum value). Intensity grayscale image data is usually stored as 8 bit or 28 = 1506 gray color intensity of a value of 0 (as the value of black) and 1505 (as a white color value). While the value of gray in between. [2]

D. Citra Binary

Binary image is the image that each pixel only has a value of 0 or 1 so that the color produced only in the form of black and white. Binary image can be generated by the floating process (thresholding).

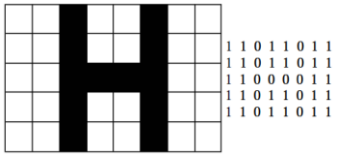

Fig 1. Representasi Black and White[2]

E. Digital Inage Processing

Image processing is image processing, in particular by using a computer, the image of which has a better picture quality.

F. Edge Detection

Edge is a change in the value of the degree of gray intensity increased in a short time. The difference is this intensity that reveals details on the image. Usually found on the edge of the boundary between two different areas in an image. Edges can be oriented in one direction, and this direction is different depending on the change in intensity. Edge characterizes the boundaries of the object to the process of segmentation and identification of objects in the image. Destination edge detection operations to improve the appearance of the boundary line of an object that is in the image. [9]

G. Computer Vision

Computer Vision can be said as a discipline that studies how to reconstruct, interpret, and understand the 3D display of 2D in terms of the nature of the structure of the display. Computer Vision can be said to mimic human vision using the software and hardware on a computer that combines knowledge in the field of computer science. [3]

H. Algoritma Canny Edge Detection

Fig 2. *Canny Algorithm* [10]

I. Raspberry pi

Raspberry pi image processing with a more economical and technically smarter than doing interfacing system images on a personal computer. Raspberry pi on using open source software such as Linux which makes it more efficient and the software can be downloaded for free. In the case of Raspberry pi image processing will be easy to do and will be more effective because it is lighter weight, lower power consumption and efficient use. [4]

Fig 3. Raspberry pi

J. USB Camera

Fig 4. USB Camera Havit HV-V612

DOI : 04.IJAI.2015.1.5

III. Design of System

A. Description of System

In this system, the camera used to take pictures of the sign on the parking slot then Raspberry pi perform digital image processing with Canny edge detection method to obtain the results of the detection of existing triangular mark on each parking slot. In general design of the system in this final project is as follows:

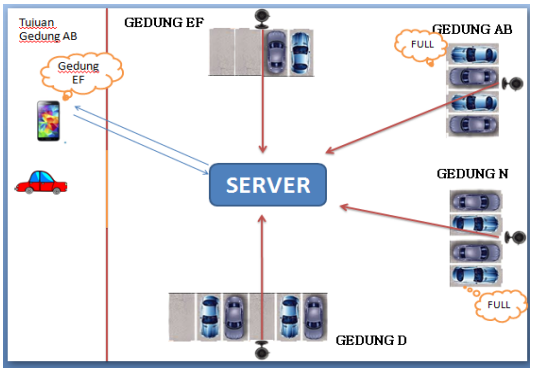

Fig 5. *Smart Parking System*

Usb camera is placed in the parking area with a height and a certain angle in order to reach the parking slots in accordance with the display on the monitor connected to the Raspberry pi. At each slot parking lot made his mark as a sign that will be detected by the system. Sign will be given to each slot parking lot that is a sign of a triangle. When the camera did capture the area of the parking slot, the digital image processing performed by the Raspberry pi and further image data that has been processed to be sent to the server every change of data. When the digital camera captures the image of the triangle mark in a slot, the slot in the parking lot is empty, but if it only captures some signs triangle or even not seen it on the slots have been filled.

The output of the image processing on the Raspberry pi in the form of a binary image is 1 and 0. The output results provide information filled or empty in a parking slot and then the results are sent to the server using a LAN cable and is stored in a local database residing on the server. Data transmission speeds of digital image processing results depend on the speed detecting triangular mark on the captured image of the parking lot

B. Definition of Ideal Parking

Ideal parking conditions referred to in this thesis which includes some of the following:

1. Cars parked neatly fit a parking slot in the parking area.
2. The parking area is used exclusively by car / four-wheel vehicle.
3. There are no objects or trees blocking the camera did capture the parking slot.
4. The system will send a value of 1 to the server when the camera detects the mark, and will send a value of 0 to a server when it does not detect the mark.
5. A sign on the closed parking slot all by car.

C. Design of System

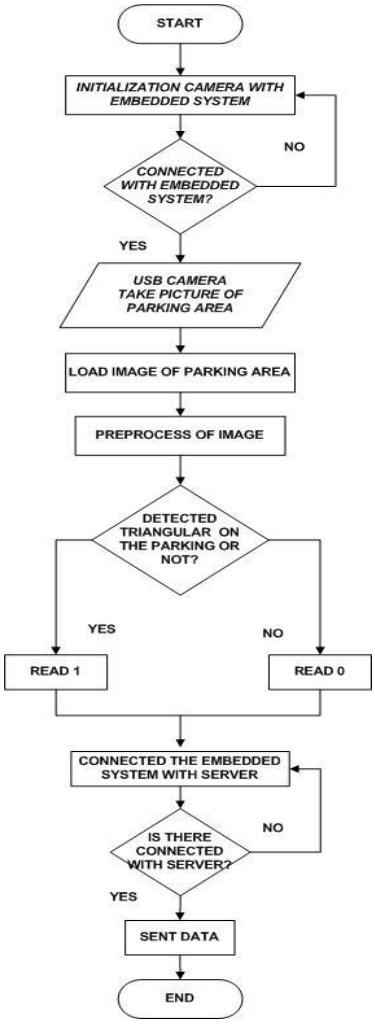

Fig 6. Flowchart of system

D. Spesification of System

The design specifications of the system used in this final project, namely:

1. The camera can detect signs of the parking slot.
2. The use of the usb camera can detect signs so far
3. The image of a sign which is detected as input can be processed by the system to generate output binary image.
4. If the system detects a sign at a parking slot then it will have a value of "1".
5. If the system does not detect a mark on a parking slot for signs covered by the car in a parking slot, the system will have a value of "0".

DOI : 04.IJAI.2015.1.5

6. The system can transmit the results of detection of the camera to the server.
7. The data communication between the camera detects a sign with the server using a LAN cable.

E. Design and Implementation of Hardware

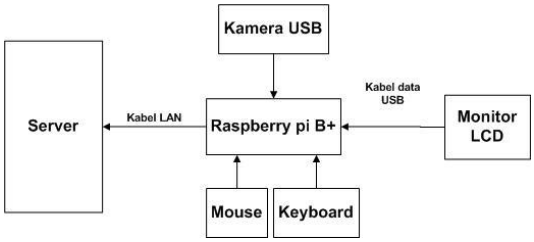

Fig 7. Design of Hardware

F. Data Input and Output

The results of image capture performed by the camera installed in the parking slot will be the input data on the smart parking system is. Data output from this system in the form of a binary image from the image processing. Data output will be worth "1" and "0" is then sent to the server.

TABEL 1 DATA INPUT DAN OUTPUT

No.	Data *input*	Data *output*	Keterangan
1.	Citra RGB	1	Terdeteksi tanda
2.	Citra RGB	0	Tidak terdeteksi tanda

G. Design The Format of Data

The data format is needed in data communication in these systems. Sending data to the server will be successful if the data format on the Raspberry pi and MySQL database on the same server. The following are examples fotmat form of data used:

TABEL 2 DATA INPUT DAN OUTPUT

Format Data	Keterangan
slot_id, time_log, status example : E001, 2015-03-14 03:06:55,1	- slot_id : id for each parking slot based on the parking area, for example: E001 = E (the parking area of the building E), 001 (a parking slot number 001) - Time_log: a sensor to take pictures, for example: 2015-03-14 03:06:55 (indicating the date and time) - Status: the status of the parking slot, for example: 1 for empty and 0 to contents

H. Design Process of System

On the software flow chart below shows that the system starts by pairing between the database server with Raspberry pi then check the connections to make sure that the server can perform data communication with the Raspberry pi. Communication between the server and Raspberry pi is extremely important for the delivery of data when the data is completed in the process. After doing a check connections then the camera will capture a sign on the parking slot to be used as an input system, input data in the form of an RGB image.

Then the image is converted to grayscale image and done threshold in order to obtain the results of the image or the image better. The next image processing is canny edge detection algorithm to determine whether there are signs in the parking slot. The output of the image processing in the form of a binary image, if there is a sign on the captured then it will send the value "1" to the server otherwise if there are no signs it will send the value "0". Data will be stored on a MySQL database on the server. As in the flowchart below:

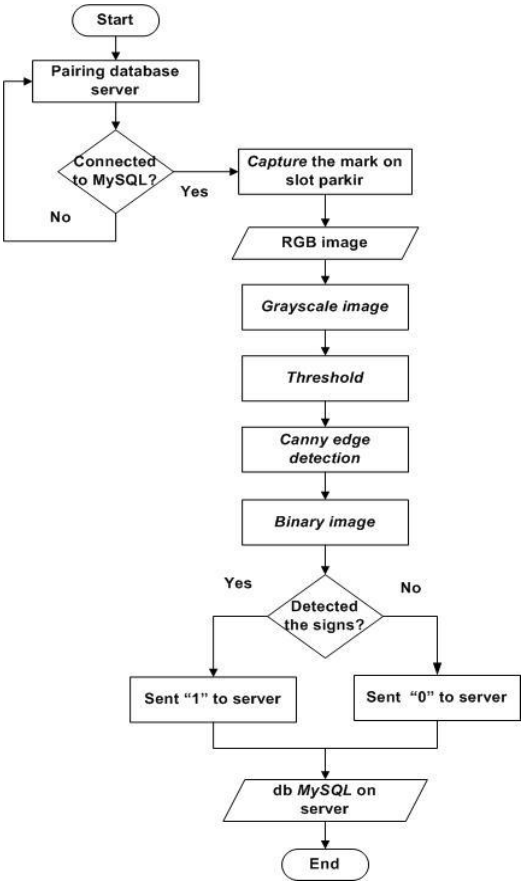

Fig 8. Flowchart Process of System

I. Grayscale and Thresholding

Usb camera capture the image of the first converted in the form of a grayscale image to facilitate image computing. Then performed thresholding to produce binary image to be sent to the server.

DOI : 04.IJAI.2015.1.5

J. Canny Algorithm

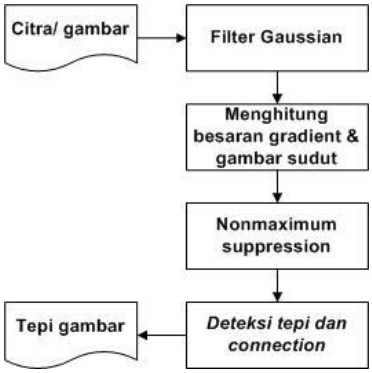

Fig 9. The step of Canny Algorithm

IV. ANALYSIS

To determine the performance of the system that has been designed and consistency of the data, it is necessary to test the system. The purpose of this test is as follows:

1. Knowing the system performance based on the consistency of data when the camera did capture the mark.
2. Determine the influence of illumination on the test in the morning, afternoon, evening and night.
3. Knowing the system performance by Accuracy Shipment data to the server whether the data in accordance with the Raspberry pi incoming data on the database server.
4. Knowing the computing time in processing the input image.

A. The Result Testing of Detection Signs

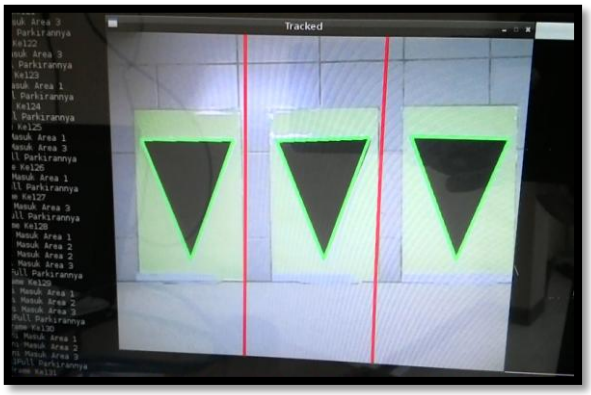

Fig 10. The result of mark detection

TABEL 3 THE DETECTION OF SIGNS

No.	*Input* (image)	Lot of area	Condition	*Output* of Monitor	Description
1	Detected of 3 signs	E003	Detected	Go to area 3	Stable
		E002	Detected	Go to area 2	
		E001	Detected	Go to area 1	
2	Detected of 3 signs	E003	Detected	Go to area 3	Unstable
		E002	Detected	Go to area 2	
		E001	Not detected	-	
3	Detected of 2 signs	E003	Detected	Go to area 3	Stable
		E002	Detected	Go to area 2	
		E001	Detected	Go to area 1	
4	Detected of 2 signs	E003	Detected	Go to area 2	Stable
		E002	Detected	Go to area 3	
		E001	Detected	-	
5	Detected of 1 signs	E003	Detected	Go to area 3	Unstable
		E002	Detected	Go to area 2	
		E001	Not detected	-	

Based on the table above it can be seen that the cameras capture the image in the form of a sign in the slot area with the detection results mark displayed on the monitor has no constant percentage value of 16 667% of the data retrieval 150 times. If the sign is detected, the output on the monitor will display the information entered area 1, 2 or 3 in accordance with the parking slot where the detected mark. From the results of data collection as much as 150 times there are 25 data is not constant or unstable, from observations of a constant amount of data that is not directly proportional to the number of marks detected. More and more signs are detected, the more likelihood that the system is not constant in detecting signs. So the value of consistency in data detection is done in the amount of 83 333%.

DOI : 04.IJAI.2015.1.5

B. The Result Testing of Intensity of the light

TABEL 4 THE INNTENSITY OF LIGHT

No	Time	Luxmeter	Description stability
1	Morning (7 – 10 AM)	4000 – 6000 *lux*	69.333%
2	Afternoon (10 AM – 2 PM)	7000 –9000 *lux*	60%
	Afternoon (2 – 3 PM)	8000 – 6000 *lux*	73.333%
3	Afternoon (3 – 5 PM)	6000 – 4000 *lux*	83.333%
4	Night (6 – 9 PM)	Minimum 173 *lux*	68%

Based upon the intensity of the light is very influential in the process of detecting the mark on the area of the parking slot. System with light intensity value range of 6000 - 4000 lux has a percentage value is not constant at 30 667%. Light intensity with a value range of 7000 - 9000 lux, the percentage value is not constant at 39 333%. Light intensity with a value range of 8000 - 6000 lux, the percentage value is not constant at 26 667%. At night the necessary light bulb that can light up a sign at a parking slot so that the light intensity reaching 173 lx with a presentation that is not a constant value of 32%.

C. The Reslut Testing of Accuracy Sent The Data

TABEL 5 SENT THE DATA

No	*Input* (image)	Lot of area	*Output* of Raspberry Pi	*Output* of *Server*	Description
1	Detected of 3 signs	E003	Go to area 3	1	True
		E002	Go to area 2	1	
		E001	Go to area 1	1	
2	Detected of 1 signs	E001	-	0	True
3	Detected of 1 signs	E001	Go to area 1	1	True
4	Detected of 2 signs	E002	-	0	True
5	Detected of 2 signs	E003	-	1	False
		E001	-	0	
6	Detected of 1 signs	E003	Go to area 3	1	True
7	Detected of 2 signs	E002	Go to area 2	1	True
		E001	Go to area 1	1	
8	Detected of 1 signs	E003	Go to area 3	0	True
9	Detected of 1 signs	E001	-	0	True
10	Detected of 1 signs	E003	Go to area 3	1	True

Image detection input in the form of three marks, two marks and a sign of the parking slot has an output corresponding to the output server. Data shown on the server only the data slot parking area changes. If there is data that does not change but the output is displayed in the server error occurs in the system. The system is said to successfully perform data transmission if the detection mark on the Raspberry pi and server have the same value or status. In this test an error occurs or a data value that is not constant at 8% and 92% in the value of the successful delivery of data as much as 150 times.

D. The Result of Time Execution

TABEL 6 TIME EXECUTION

No	Description	Average of time detection
1	Deteksi 3 tanda	46.8 *ms*
2	Deteksi 2 tanda	36.1 *ms*
3	Deteksi 1 tanda	33.2 *ms*

Based on the test results can be known when computing system in detecting the mark on the area of the parking slot. The average time required to process the image with the detection of three marks are for a maximum time of 46.8 ms to 52 ms and 42 ms minimum time. On detection of two average marks obtained in the computing time doing image processing for a maximum time of 36.1 ms to 40 ms and 32 ms minimum time. Then on the detection of the marks obtained by the average computation time required for the maximum time of 33.2 ms to 32 ms and 36 ms minimum time. Analysis obtained from these tests marks detected amount proportional to the length of time computing in detecting signs. The more the number sign is detected then the longer time required to perform the mark detection computation.

V. CONCLUSION

Smart parking system with canny edge detection algorithm can detect a mark on the outdoor parking area slot with value stability success in detecting the data for 83 333% and instability in detecting the mark by 16 667% based on 150 times for taking data. Effect of light intensity on the detection of a mark on the outdoor parking area slot has five percentage keberhasilam in detecting signs of the different conditions of light intensity that is 69 333% of the light intensity of the morning, 60% and 73 333% on the intensity of light during the day, at an intensity of 83 333% afternoon light, and 68% in light intensity night. Influence appears that the camera would be difficult to detect signs. Intensity value range of 6000 - 4000 lux make the system stable enough in detecting signs. While the intensity value range of 7000 - 9000 lux result in an unstable system. Lighting is very small such as the condition of the evening with a value of 3 lux may cause the system unable to detect signs. Therefore, it needs to be installed lamp bulb with a minimum value of 173 lux light intensity in order to detect signs. Information on the results of image processing Raspberry pi successfully sent to the server via a LAN cable. Sending data only when the data changes the value of the mark detection. The percentage of accuracy in data transmission is 92% based on 150 times the data retrieval. Signs at the slot parking area successfully detected with a variety of computing time when the system detects three signs, two signs and a sign. The average computation time lasts for 46.8 ms upon detection of three signs, the detection of two marks obtained in the computing time doing image processing for 36.1 ms. Then, upon detection of the sign computation time required for 33.2 ms.

DOI : 04.IJAI.2015.1.5

1) References

[1] Agung Kurniawan, "*Jumlah Kendaraan Indonesia Capai 104,211 Juta Unit,"* April 2014. [Online]. Available : http://www.tribunnews.com/otomotif/2014/04/15/jumlah-kendaraan-di-indonesia-capai-104211-juta-unit.[Accessed 28 Oktober 2014].

[2] Ahmad Saprudin, "*Jurnal Membangun Jaringan LAN pada Perusahaan Three Pringsewu Lampung dengan menggunakan Metode OSPF by. Asep,"* Lampung: Akademi Manajemen Informatika dan Komputer Dian Cipta Cendekiawan Pringsewu, 2013.

[3] Alvin Antonius, Dedi Triyanto and Ikhwan Ruslianto,"*Penerapan Pengolahan Citra dengan Metode Adaptive Motion Detection Algorithm pada Sistem Kamera Keamanan dengan Push Notification ke Smartphone Android,"* Pontianak: Universitas Tanjungpura, 2015.

[4] Anoop Mishra and Arshita Dixit, "*Embedded Image Capturing & Digital Converting Process using Raspberry pi System interfacing and Comparision of Generation 2 verses Generation 1 models in Raspberry pi"* International Journal of Computer Science and Information Technologies (IJCSIT), Vol. 6, no. 2, pp. 1798-1801, April 2015.

[5] C.Danil, "*Deteksi tepi dengan Algoritma Canny,"* Medan: STMIK IBBI, 2013.

[6] Decy Nataliana, Iqbal Syamsu and Galih Giantara, "*Sistem Monitoring Parkir Mobil menggunakan Sensor Infrared berbasis Raspberry pi,"* Bandung: Institut Teknologi Nasional, 2014.

[7] Helmiriawan, "*Rancang Bangun dan Analisis Sistem Pemantau Lalu Lintas menggunakan OpenCV dengan Algoritma Canny dan Blob Detetction, "* Jakarta: Universitas Indonesia, 2012.

[8] I Putu Agus Eka Pratama, "*Smart City beserta Cloud Computing*," Bandung : Informatika, 2014.

[9] Johanes W. Yodha and Achmad W. Kurniawan, "*Perbandingan Penggunaan* Deteksi tepi *dengan Metode Laplace , Sobel, dan Prewit dan Canny pada Pengenalan Pola*," Semarang: Universitas Dian Nuswantoro, 2014.

[10] Rashmi, Mukesh Kumar and Rohini Saxena, "*Algorithm and Technique on Various Edge Detection : A Survey*," Signal & Image Processing : An International Journal (SIPIJ), Vol.4, No.3, June 2013.

[11] Mazid Kamal and Ruri S. Basuki, "*Segmentasi Citra Daun Tembakau Berbasis* Deteksi tepi *menggunakan Algoritma Canny,"* Semarang: Universitas Dian Nuswantoro, 2011.

[12] Nanda Septiana, "*Implementasi dan Analisis Motion Identification Berupa Gaya Berjalan Manusia serta Aplikasinya dalam Sistem Keamanan Rumah menggunakan Support Vector Machine,"* Bandung: Telkom University, 2014.

[13] Eko Wibowo, R.Isnanto, and Ajub Ajulian Zahra, "*Perbandingan Kinerja Operator Sobel dan Laplacian of Gaussian (LoG) terhadap Acuan Canny untuk Men*deteksi tepi *Citra,"* Semarang: Universitas Dian Nuswantoro, 2013.

[14] Ping ZHOU, Wenjun YE, Yaojie XIA and Qi WANG, "*An Improved Canny Algorithm for Edge Detection*," Journal of Computational Information Systems (JOFCIS), Vol 7, no. 5, pp. 1516-1523, 2011.

DOI : 04.IJAI.2015.1.5