

PERANCANGAN SISTEM PARKIR OTOMATIS MENGGUNAKAN MOBIL MINI BAGIAN PENGOLAHAN CITRA

DESIGN OF AUTOMATIC PARKING SYSTEM BY USING MINI CAR PART IMAGE PROCESSING

Anggi Tiovary Siregar¹, Agus Virgono,Ir.,MT², Gelar Budiman, ST, MT³

^{1,2,3} Prodi S1 Sistem Komputer, Fakultas Teknik, Universitas Telkom Bandung

¹anggitiovary@gmail.com, ²avirgono@ittelkom.ac.id, ³gblb@ittelkom.ac.id

ABSTRAK

Pada saat ini, banyaknya jumlah kendaraan yang beredar di masyarakat sering mengakibatkan kerumitan di jalan raya. Kerumitan tidak hanya persoalan kemacetan, tapi juga kesulitan dalam proses parkir kendaraan. Kendaraan bermotor mungkin tidak begitu sulit diparkirkan namun tidak begitu dengan kendaraan roda empat yaitu mobil. Diperlukan teknologi yang bisa memberikan informasi jarak celah parkir yang tersedia. Pada tugas akhir ini dirancang sebuah mobil mini yang dilengkapi dengan *webcam* untuk mengambil gambar dan *raspberry pi* untuk mengolah gambar. Gambar yang berhasil di-capture akan diolah oleh *raspberry pi* untuk kemudian dilakukan perhitungan jarak antara dua pembatas parkir pada gambar. Pengolahan gambar menggunakan *computer vision* yaitu *OpenCV* dan *SimpleCV*. Tahap pengolahan citra untuk menemukan area parkir yang tepat meliputi konversi *RGB* ke *Grayscale*, *Grayscale* ke biner, penentuan wilayah (*blob*), perhitungan jarak dan komunikasi serial untuk mengirim informasi kepada mikrokontroler. Hasil pengolahan gambar menghasilkan piksel yang merepresentasikan jarak antar dua tiang. Bila jarak antar tiang sesuai dengan ukuran mobil maka *raspberry pi* mengirim instruksi untuk parkir di *space* parkir tersebut namun bila tidak sesuai dengan ukuran *space* parkir maka *raspberry pi* memberikan instruksi untuk jalan dan menemukan *space* parkir yang lain.

Kata kunci : *Raspberry pi*, Pengolahan Citra, *Blob*, Komputer Visi, Tempat Parkir

ABSTRACT

At this time, the large number of vehicles which are circulating in the community often lead to complexity in highway. Complexity is not just about problem of congestion everywhere, but also the difficulties in the process of vehicle parking is. Motor vehicles may not be so difficult parked but not so with the four-wheeled vehicles, namely car. Technology is needed to give information about distance of parking slot. In this final project, it is designed a car that is equipped with a webcam to take pictures and raspberry pi for image processing. The image will be processed by Raspberry pi and then it calculates the distance between two barriers on the picture. Image processing uses computer vision, theses are OpenCV and SimpleCV. Image processing stages of finding the right slot of parking are including conversion of RGB to Grayscale, conversion of grayscale to binary, determining area (blob), calculating the distance, and serial communication to send information to mikrokontroler. The result of image processing put out pixel that represent the distance between two barriers. If the distance fits with the car size, raspberry pi send instruction for parking in that area and if the distance doesn't fit with car size, raspberry pi send instruction for looking for the other space parking.

Keyword : *Raspberry pi*, Image Processing, *Blob*, Computer Vision, Parking Lot

1. Pendahuluan

Perkembangan teknologi transportasi pada saat ini sudah mengalami kemajuan yang sangat pesat. Transportasi tidak hanya berfokus pada sebuah kendaraan yang bisa mengantarkan orang dari suatu tempat ke tempat lain tapi lebih dari itu. Alat transportasi sudah dilengkapi dengan perangkat-perangkat pendukung yang mengoptimalkan kepuasan pengguna transportasi. Kemajuan teknologi multimedia juga ikut berkontribusi dalam perkembangan teknologi transportasi di zaman ini. Dengan multimedia memungkinkan pengguna dapat melakukan navigasi, berinteraksi dan berkomunikasi[5]. Seperti yang kita ketahui saat ini pengguna mobil pada saat melakukan parkir tidak hanya dibantu oleh dua kaca spion tetapi juga dimudahkan dengan adanya rear camera. Rear camera adalah kamera yang berada di belakang mobil sehingga pengemudi tidak perlu khawatir menabrak saat melakukan parkir mundur. Walaupun sedemikian banyak teknologi yang memberikan kepuasan bagi pengguna kendaraan tetap saja selalu ada masalah. Hal yang paling banyak terjadi saat ini adalah penggunaan lahan parkir tidak efektif, padahal pengelola tempat parkir sudah memberi batas garis / sekat-

sekat demi penggunaan tempat parkir yang optimal namun masih belum terapkan dengan baik. Kasus ini mungkin tidak terlalu menyulitkan bagi pengendara senior, lain halnya dengan para pengendara baru yang mungkin sulit menggunakan area tempat parkir secara efektif dan efisien dan para pengendara baru juga belum terlalu terbiasa memarkirkan mobil di lahan yang tidak terlalu lebar. Masalah lain yang membutuhkan teknologi transportasi tingkat tinggi adalah bagi para pekerja yang memiliki rutinitas padat sehingga tidak memiliki waktu untuk memarkirkan mobil. Untuk itu dalam tugas akhir ini dilakukan perancangan dan implementasi sistem parkir otomatis pada mobil mini dengan menggunakan pengolahan citra. Sistem parkir otomatis ini mempermudah pengendara dalam melakukan parkir secara aman dan efisien. Mobil akan bergerak otomatis dan menemukan tempat parkir sendiri. Dengan sistem parkir otomatis ini para pengendara baru ataupun pekerja dengan tingkat rutinitas tinggi akan lebih mudah dalam melakukan parkir secara aman, efektif dan efisien.

Penelitian ini bertujuan untuk dapat mengukur jarak antara dua tiang berdasarkan area terluas yang terdeteksi oleh webcam. Penentuan wilayah menggunakan BLOB (Binary Large Object) atau area putih terbesar. Tujuan dari deteksi blob adalah untuk menentukan titik pusat blob dalam frame dikodekan dengan XY koordinat. Koordinat pusat sangat dipengaruhi oleh piksel pinggir blob tersebut[1]. Hasil output merupakan pengurangan koordinat X maksimum dan X minimum dalam piksel kemudian dibandingkan dengan ukuran dalam sentimeter (ukuran jarak sebenarnya). Webcam dipasang diatas mobil yang sudah dipasang tatakan. Webcam akan mengambil gambar saat mendapat instruksi dari mikrokontroler. Gambar yang telah dihasilkan akan langsung diolah. Gambar RGB kemudian dikonversi menjadi grayscale. Untuk lebih memudahkan komputasi, citra grayscale dikonversi menjadi citra biner. Citra biner merupakan citra digital dengan dua kemungkinan nilai *pixel* yaitu *pixel-pixel* objek bernilai 1 dan *pixel-pixel* latar belakang bernilai 0[3]. Pengkonversian citra menjadi bentuk biner dibutuhkan nilai *threshold* yang ditentukan sesuai kebutuhan. *Thresholding* merupakan konversi citra hitam – putih ke citra biner dilakukan dengan cara mengelompokkan nilai derajat keabuan setiap pixel kedalam 2 kelas, hitam dan putih[2]. Setelah gambar menjadi bentuk biner, dilakukan pendeteksian blob. Proses pengolahan citra dilakukan menggunakan *computer vision* tipe *OpenCV*. *OpenCV* (*Open Computer Vision*) merupakan semacam *library* yang digunakan dalam membuat aplikasi yang berkaitan dengan *computer vision*. *OpenCV* juga menyediakan banyak fungsionalitas, yang tentunya dapat mempersingkat waktu dalam pembuatan aplikasi yang lumayan rumit[4]. Tujuan dari komputer vision adalah untuk mengembangkan algoritma mengambil gambar sebagai masukan dan menghasilkan interpretasi simbolik objek yang ada dan beberapa informasi tentang hubungan tiga dimensi spasial pada objek[6]. Penggunaan *computer vision* memudahkan proses citra ini. Blob akan lebih mudah ditemukan dalam citra biner, saat blob berhasil dideteksi maka dilakukan operasi selanjutnya yaitu menemukan koordinat X maksimum dan X minimum. Hasil pengukuran ini akan merepresentasikan jarak pembatas parkir. Bila hasil pengukuran sesuai dengan ukuran mobil, raspberry pi akan mengirimkan instruksi parkir namun bila tidak sesuai dengan ukuran mobil, maka raspberry pi akan mengirimkan instruksi bergerak seperti sebelumnya dan mencari space lain.

2. Metodologi (Perancangan Sistem)

2.1 Perancangan Perangkat



1. Raspberry Pi

Raspberry pi yang digunakan dalam penelitian ini adalah Raspberry pi Tipe B dengan spesifikasi sebagai berikut :

- Processor : 700 MHz ARM1176JZF-S core

- GPU : Broadcom VideoCore IV @ 250 MHz
- RAM : 512 MB
- USB : 2 (via the built in integrated 3-port USB hub)
- Berat : 45 g
- Ukuran : 85.60 mm × 56 mm

Dalam melakukan pendeteksian objek menggunakan raspberry pi dibutuhkan beberapa perangkat lunak pendukung, untuk itu dilakukan 3 hal berikut yaitu *Install OS Wheezy Raspbian*, *Update dan Upgrade Package* dan *Install Open CV*

2. Webcam

Pengambilan gambar menggunakan kamera logitech c270H dengan spesifikasi sebagai berikut resolusi maksimal 1280 x 720, 3.0 megapiksel, *Hi-Speed USB 2.0 certified*. Kamera Video Konferensi yang juga dikenal dengan Kamera *Web (Webcam)* adalah kamera yang dirancang penggunaannya pada *World Wide Web*. Webcam dapat mentransmisikan video berwarna ke TV atau PC[7].

3. SDCard

SDCard digunakan untuk media penyimpanan mini prosessor, dalam tugas akhir ini penulis menggunakan SDCard Toshiba 16 GB sebagai media penyimpanan, namun penggunaan SDCard.

4. Power Supply

Power supply yang dapat digunakan untuk menyalakan raspberry pi harus melalui mini *USB Port* dengan tegangan 5 volt *DC* dan arus minimal 700mA

2.2 Perancangan Program

Perancangan sistem ini terdiri dari beberapa bagian program, tahapannya sebagai berikut:

- Instruksi pengambilan gambar
- Konversi Gambar RGB ke Grayscale
- Konversi Gambar Grayscale ke Binary
- Tahap Menemukan Tiang Parkir
- Tahap Menghitung Jarak antara Dua Tiang
- Komunikasi Serial

Tahap pertama

- Program ini dimulai saat mobil mini berhenti di depan space parkir
- Mikrokontroler mengirimkan sebuah karakter kepada raspberry pi
- Raspberry pi mulai mengambil gambar menggunakan webcam saat menerima karakter dari mikrokontroler

Tahap kedua

- Gambar yang berhasil di-capture oleh webcam kemudian disimpan dalam direktory `/home/pi`
- Gambar kemudian dikonversi menjadi grayscale untuk memudahkan komputasi
- Gambar grayscale lalu dikonversi menjadi biner agar gambar yang dideteksi hanya berupa hitam putih tanpa banyak variasi warna.

Tahap ketiga

- Proses selanjutnya adalah findBlobs atau menemukan objek yang menjadi titik acuan pada gambar biner
- Objek akan disimpan dalam variable blobs yang kemudian akan dideteksi blobs[-1].maxX untuk koordinat X maksimum dan blobs[-1].minX untuk koordinat X minimum
- Hasil perhitungan akan dimasukkan ke dalam variabel ruang, seperti berikut ruang = $\text{abs}(\text{blobs}[-1].\text{maxX} - \text{blobs}[-1].\text{minX})$

Tahap keempat

- Hasil perhitungan jarak akan di bandingkan dengan jarak tiang dalam sentimeter dengan resolusi dan jarak webcam dan objek relatif konstan
- Bila hasil perhitungan sesuai dengan ukuran mobil maka raspberry akan mengirim (“/x01”) dan mobil akan melakukan parkir di space tersebut
- Bila hasil perhitungan tidak sesuai dengan ukuran mobil maka raspberry akan mengirim (“/x00”) dan mobil akan maju serta mengikuti linefollowing

3. Pengujian

3.1 Pengujian A - Pengambilan Gambar Berdasarkan Resolusi

a. Skenario Pengujian

Pengujian ini bertujuan untuk mendapatkan nilai resolusi yang ideal sehingga proses komputasi dapat dilakukan dengan waktu yang relatif singkat. Skenario yang dilakukan dengan meng-*capture* gambar dua tiang berwarna hitam dan *background* berwarna putih. Jarak pengambilan gambar antara *webcam* dan objek adalah 25 cm. Selama pengujian jarak pengambilan gambar dan objek yang di-*capture* konstan. Hal ini dilakukan untuk mendukung penganalisaan. Percobaan dilakukan masing-masing 10 kali pengambilan gambar, pengambilan gambar dengan resolusi 640x480 sebanyak 10 kali dan pengambilan gambar dengan resolusi 320x480 sebanyak 10 kali. Program pengambilan gambar dilakukan dengan menggunakan *SimpleCV*. Parameter yang menjadi titik acuan adalah luas area, hasil perhitungan, dan waktu.

b. Hasil Pengujian

Tabel 4.1 Pengolahan Gambar Menggunakan Resolusi 640x480

Resolusi Frame	Jarak Webcam & Objek	Jarak Sebenarnya (cm)	Luas Area	Hasil Perhitungan (piksel)	Waktu (detik)
640x480	25 cm	10 cm	146558,0	316	17,35
640x480	25 cm	10 cm	146583,5	316	17,58
640x480	25 cm	10 cm	146544,5	316	17,24
640x480	25 cm	10 cm	146597,0	316	17,19
640x480	25 cm	10 cm	146554,5	316	17,16
640x480	25 cm	10 cm	146555,0	316	17,12
640x480	25 cm	10 cm	146600,5	316	17,44
640x480	25 cm	10 cm	146589,0	316	17,13
640x480	25 cm	10 cm	146590,5	317	17,29
640x480	25 cm	10 cm	146695,0	316	16,76

Tabel 4.2 Pengolahan Gambar Menggunakan Resolusi 320x240

Resolusi Frame	Jarak Webcam & Objek	Jarak Tiang (cm)	Luas Area (piksel)	Hasil Perhitungan (piksel)	Waktu (detik)
320x240	25 cm	10 cm	36594,5	160	15,53
320x240	25 cm	10 cm	36595,5	160	15,76
320x240	25 cm	10 cm	36609,5	160	15,94
320x240	25 cm	10 cm	36623,5	160	15,66
320x240	25 cm	10 cm	36613,5	160	15,83
320x240	25 cm	10 cm	36615,5	160	15,72
320x240	25 cm	10 cm	36626,5	160	15,73
320x240	25 cm	10 cm	36635,5	160	15,85
320x240	25 cm	10 cm	36629,5	160	15,79
320x240	25 cm	10 cm	36706,5	160	15,94

Pengamatan yang dilakukan dengan melihat perbedaan hasil perhitungan piksel antara pengambilan gambar dengan resolusi 640x320 dan pengambilan gambar dengan resolusi 320x240. Hasil pengujian menunjukkan penggunaan resolusi yang lebih kecil akan mengeluarkan hasil perhitungan piksel yang lebih kecil pula, yaitu sekitar setengah hasil perhitungan dengan resolusi 640x480. Waktu yang dibutuhkan gambar dengan resolusi 320x240 juga lebih cepat sekitar 3 detik dibanding gambar dengan resolusi 640x480. Luas area yang diolah juga terjadi perubahan yang *significant*, penggunaan resolusi 640x480 menghasilkan luas area lebih kurang 146500,0 piksel sedangkan penggunaan resolusi 320x480 menghasilkan luas area 36600,0 piksel. Dari hasil pengamatan tersebut, dapat dijadikan patokan bahwa 10 cm adalah sekitar 316 piksel pada resolusi 640x480 dengan jarak ke objek 25 cm dan 10 cm adalah sekitar 160 piksel pada resolusi 320x480 dengan jarak ke objek 25 cm.

3.2 Pengujian B – Pengambilan Gambar Berdasarkan Jarak

a. Skenario Pengujian

Pengujian ini bertujuan untuk membandingkan hasil perhitungan ketika terjadi perubahan jarak antara webcam dan objek. Skenario yang dilakukan adalah menggunakan pengambilan gambar dengan resolusi yang sama namun jarak pengambilan berbeda. Percobaan ini dilakukan masing-masing 10 kali dengan dua jarak pengambilan berbeda. Pengambilan gambar dilakukan dengan menggunakan resolusi yang tetap yaitu 640x480. Hal ini bertujuan untuk melihat perubahan hasil perhitungan jika menggunakan jarak pengambilan gambar yang berbeda. Program pengambilan gambar menggunakan computer vision tipe *SimpleCV*. Parameter yang menjadi titik acuan adalah luas area, hasil perhitungan dan waktu.

b. Hasil Pengujian

Tabel 4.3 Pengolahan Gambar dengan Jarak 25 cm ke Objek

Resolusi	Jarak ke Objek	Jarak (cm)	Luas Area	Hasil	Waktu
640x480	25 cm	10 cm	146558,0	316	17,35
640x480	25 cm	10 cm	146583,5	316	17,58
640x480	25 cm	10 cm	146544,5	316	17,24
640x480	25 cm	10 cm	146597,0	316	17,19
640x480	25 cm	10 cm	146554,5	316	17,16
640x480	25 cm	10 cm	146555,0	316	17,12
640x480	25 cm	10 cm	146600,5	316	17,44
640x480	25 cm	10 cm	146589,0	316	17,13
640x480	25 cm	10 cm	146590,5	317	17,29
640x480	25 cm	10 cm	146695,0	316	16,76

Tabel 4.4 Pengolahan Gambar dengan Jarak 30 cm ke Objek

Resolusi	Jarak Webcam & Objek	Jarak Tiang (cm)	Luas Area (pixel)	Hasil (pixel)	Waktu (s)
640x480	30 cm	10 cm	124909,5	271	17,00
640x480	30 cm	10 cm	124848,5	272	16,82
640x480	30 cm	10 cm	124849,5	272	17,06
640x480	30 cm	10 cm	124864,5	272	17,14
640x480	30 cm	10 cm	124868,5	272	16,85
640x480	30 cm	10 cm	124838,0	272	17,03
640x480	30 cm	10 cm	124837,0	271	16,72
640x480	30 cm	10 cm	124862,0	272	16,74
640x480	30 cm	10 cm	124851,5	271	17,10
640x480	30 cm	10 cm	124866,0	271	17,69

Hasil pengujian menunjukkan perubahan jarak pengambilan gambar memberikan hasil perhitungan yang berbeda pula. Perhitungan menggunakan jarak yang lebih jauh menghasilkan pikse yang lebih kecil. Dari segi waktu, perubahan jarak pengambilan tidak menyebabkan perubahan terhadap lama proses pengolahan. Perubahan juga terjadi pada luas area, luas area yang dihasilkan menggunakan jarak yang lebih jauh menjadi lebih kecil, namun tidak terjadi perubahan yang begitu drastis.

3.3 Pengujian C – Pengambilan Gambar Berdasarkan Warna Objek

a. Skenario Pengujian

Pengujian ini bertujuan mendapatkan objek yang ideal dalam pengolahan gambar. Percobaan dilakukan dengan cara meng-*capture* gambar tiang berwarna ungu, kuning, oren, dan hijau. *Background* yang digunakan berwarna putih. Percobaan dilakukan dalam keadaan jarak pengambilan gambar yang sama serta penggunaan resolusi yang tetap. Hal ini akan memudahkan dalam penganalisaan. Pokok perhatian dalam percobaan ini adalah seberapa besar perubahan warna objek mempengaruhi hasil perhitungan, luas area dan waktu pengolahan. Program menggunakan *OpenCV*.

b. Hasil Pengujian

Tabel 4.5 Pengolahan Gambar Menggunakan Tiang Warna Ungu

Resolusi	Jarak Webcam & Objek	Jarak Tiang (cm)	Luas Area (pixel)	Hasil (pixel)	Waktu (s)
640x480	35 cm	25 cm	245279,5	569	15,88
640x480	35 cm	25 cm	265577,0	613	15,63
640x480	35 cm	25 cm	265816,0	613	15,46
640x480	35 cm	25 cm	265635,5	613	15,40
640x480	35 cm	25 cm	227929,0	569	16,33
640x480	35 cm	25 cm	215933,5	564	16,25
640x480	35 cm	25 cm	241964,0	565	16,19
640x480	35 cm	25 cm	265420,0	612	15,36
640x480	35 cm	25 cm	236765,0	557	16,20
640x480	35 cm	25 cm	265854,0	614	15,42

Tabel 4.6 Pengolahan Gambar Menggunakan Tiang Warna Kuning

Resolusi	Jarak Webcam & Objek	Jarak Tiang (cm)	Luas Area (pixel)	Hasil (pixel)	Waktu (s)
640x480	35 cm	25 cm	286211,5	638	15,25
640x480	35 cm	25 cm	246385,0	599	16,34
640x480	35 cm	25 cm	275037,0	621	15,76
640x480	35 cm	25 cm	272862,0	622	16,03
640x480	35 cm	25 cm	244032,5	589	16,97
640x480	35 cm	25 cm	264007,5	620	15,61
640x480	35 cm	25 cm	256111,0	620	16,03
640x480	35 cm	25 cm	258203,5	616	16,10
640x480	35 cm	25 cm	252691,5	620	16,05
640x480	35 cm	25 cm	252436,5	611	16,54

Tabel 4.7 Pengolahan Gambar Menggunakan Tiang Warna Orange

Resolusi	Jarak Webcam & Objek	Jarak Tiang (cm)	Luas Area (pixel)	Hasil (pixel)	Waktu (s)
640x480	35 cm	25 cm	265542,0	608	15,56
640x480	35 cm	25 cm	232821,5	579	16,70
640x480	35 cm	25 cm	247306,5	580	15,78
640x480	35 cm	25 cm	262320,0	597	15,55
640x480	35 cm	25 cm	225451,5	569	16,04
640x480	35 cm	25 cm	249839,5	586	16,84
640x480	35 cm	25 cm	239658,5	580	15,63
640x480	35 cm	25 cm	203263,5	563	16,72
640x480	35 cm	25 cm	265569,5	615	15,29
640x480	35 cm	25 cm	260032,5	589	14,38

Tabel 4.8 Pengolahan Gambar Menggunakan Tiang Warna Hijau

Resolusi	Jarak Webcam & Objek	Jarak Tiang (cm)	Luas Area (pixel)	Hasil (pixel)	Waktu (s)
640x480	35 cm	25 cm	242472,0	592	16,30
640x480	35 cm	25 cm	255606,0	586	16,20
640x480	35 cm	25 cm	269661,5	622	15,41
640x480	35 cm	25 cm	268818,0	618	15,72
640x480	35 cm	25 cm	269359,0	619	15,58
640x480	35 cm	25 cm	268989,0	619	15,52
640x480	35 cm	25 cm	262549,0	606	16,06
640x480	35 cm	25 cm	269558,5	621	15,63
640x480	35 cm	25 cm	269707,0	621	15,23
640x480	35 cm	25 cm	269485,0	622	15,42

4. Kesimpulan

Jarak dan resolusi sangat berpengaruh terhadap hasil perhitungan jarak tiang. Dengan menggunakan jarak dan resolusi yang sama, piksel yang dihasilkan dapat dibandingkan dengan ukuran jarak dalam satuan sentimeter sehingga informasi ini dapat digunakan untuk pengambilan keputusan. Waktu yang dibutuhkan *raspberry pi* dalam pengolahan gambar sekitar 15- 16 detik setiap gambarnya. Untuk mengurangi waktu pengolahan gambar dapat digunakan resolusi yang lebih rendah dalam proses namun perbedaan waktu tidak akan berubah sangat jauh. Kualitas gambar yang baik dihasilkan dari tiang yang berwarna hitam dan *background* berwarna hitam karena warna lain menghasilkan piksel yang tidak tetap walaupun perbedaan perhitungan tidak terlalu *significant*.

Daftar Pustaka

- [1] Alexandre Bochem; Rainer Herpers; Kenneth B. Kent. 2010 . *Hardware Acceleration of BLOB Detection for Image Processing*.
- [2] Hendy Mulyawan; M. Zen Hadi Samsono; Setiawardhana. 2012 . *Identifikasi dan Tracking Objek Berbasis Image Processing Secara Real Time*.
- [3] Max R. Kumaseh; Luther Latumakulita; Nelson Nainggolan. 2013. *Segmentasi Citra Digital Ikan Menggunakan Metode Thresholding*.
- [4] M. Irwan Bustami, Agus Siswanto, Irawan, Afrizal Nehemia Toscani, M. Fajri Ramdhani, Chindra Saputra. 2013. *Mendeteksi Bentuk dan Warna Bola pada Robot Humanoid Soccer dengan Menggunakan Raspberry Pi*.
- [5] Sayanti Banerjee; Pallavi Choudekar; M.K Muju . 2011. *Real Time Car Parking System Using Image Processing*. IEEE, pp. 978-1- 4244- 8679-3.
- [6] Setiawan, Roni. 2012. *Pengembangan Robot Pendeteksi Objek Berdasarkan Warna dengan Sensor Kamera sebagai Media Pembelajaran*. [unpublish]
- [7] Wingky Firnando; Muhamad Mujahidin; Irdam Adil; Mohd Iqbal. 2014. *Rancang Bangun Kamera Monitoring Untuk Menunjang Transportasi Pelabuhan Laut Berbasis Mini Komputer*.