

# IDENTIFIKASI RUANGAN DENGAN SENSOR KAMERA PADA ROBOT KRPAI BERKAKI

Rendi Pambudi, Totok Winarno, Sidik Nurcahyo

**Abstrak** - Saat ini perkembangan teknologi di bidang robotika berkembang pesat. Terutama pada pemilihan sensor yang akan digunakan pada robot. Dengan kamera sebagai sensor maka robot mampu melakukan visualisasi objek dan keadaan sekitarnya. Dari data visual hasil tangkapan sensor kamera tersebut kemudian dilakukan proses pengolahan citra (*image processing*) sehingga didapatkan sebuah persamaan untuk pola tindakan robot. Melengkapi robot dengan sensor visi / kamera meningkatkan fleksibilitas dari robot tetapi juga menyebabkan kendalinya semakin kompleks. Meskipun tingkat kesulitannya meningkat, sensor visi menjadi semakin menarik untuk navigasi robot karena memberikan informasi yang kaya tentang lingkungan robot. Pada robot KRPAI berkaki, sensor kamera dapat digunakan sebagai sensor untuk melakukan identifikasi ruangan bersama dengan sensor *ultrasonic* SRF08. Dengan melakukan pengolahan citra menggunakan metode *thresholding*, *edge detection*, dan *corner detection* robot dapat ber-*alignment* untuk memposisikan diri dalam melakukan proses identifikasi ruangan. Proses *alignment* dengan mendeteksi sudut pada ruangan. Setelah sudut dari ruangan terdeteksi, sudut terluar akan dibandingkan untuk mengetahui posisi robot pada saat ini menghadap kemana. Selanjutnya dapat ditentukan robot harus bergerak kemana untuk memposisikan dengan bidang pengukuran. Dengan melakukan *alignment* untuk memposisikan terhadap bidang pengukuran, dapat meningkatkan ketelitian dalam identifikasi ruangan.

Rendi Pambudi adalah Mahasiswa D4 Teknik Elektronika Jurusan Teknik Elektro Politeknik Negeri Malang, email: [rendipambudi93@gmail.com](mailto:rendipambudi93@gmail.com)

Totok Winarno adalah dosen Jurusan Teknik Elektro Politeknik Negeri Malang

Sidik Nurcahyo adalah dosen Jurusan Teknik Elektro Politeknik Negeri Malang

**Kata Kunci** – *Image processing*, *Edge Detection*, *Corner Detection*, *alignment*, Identifikasi Ruang.

## I PENDAHULUAN

KRPAI (Kontes Robot Pemadam Api Indonesia) divisi robot berkaki adalah salah satu kontes robot yang bergengsi di Indonesia. Dalam KRPAI divisi berkaki, robot

memiliki misi untuk mencari dan memadamkan api di arena lapangan berbentuk simulasi interior suatu rumah. Pada divisi ini yang diutamakan adalah kemampuan robot bernavigasi dan bermanuver serta kecepatan dalam menyelesaikan misinya tersebut. Dalam bermanuver, robot diharuskan dapat menelusuri dinding yang merupakan ketentuan dari sistem dalam lomba robot ini. Selain itu, dengan banyaknya jumlah ruang pada arena dapat menjadi tantangan tersendiri dalam hal efisiensi waktu robot dalam menemukan titik api. Sehingga robot harus memiliki sensor yang dapat mendeteksi keberadaan dinding, dan mengidentifikasi ruangan-ruangan pada arena.[1]

Salah satu sensor yang dapat digunakan adalah menggunakan kamera. Dengan kamera sebagai sensor maka robot mampu melakukan visualisasi objek dan keadaan sekitarnya[2]. Dari hasil data gambar tersebut kemudian diproses (*image processing*)

sehingga didapatkan sebuah persamaan untuk pola tindakan robot. Melengkapi robot dengan sensor visi / kamera meningkatkan fleksibilitas dari robot tetapi juga menyebabkan kendalinya semakin kompleks. Meskipun tingkat kesulitannya meningkat, sensor visi menjadi semakin menarik untuk navigasi robot karena memberikan informasi yang kaya tentang lingkungan robot[9].

Berdasarkan latar belakang tersebut maka penulis membuat penelitian mengenai bagaimana melakukan identifikasi ruangan pada robot KRPAI berkaki dengan menggunakan sensor kamera. Diharapkan dengan adanya alat ini dapat membantu penelitian tentang *image processing* pada robot berkaki dan juga diharapkan memberikan motivasi kepada mahasiswa Program Studi Teknik Elektronika Politeknik Negeri Malang.

## II. TINJAUAN PUSTAKA

### 2.1 Pengolahan Citra

Pengolahan citra atau *Image Processing* adalah suatu sistem dimana proses dilakukan dengan masukan (*input*) berupa citra (*image*) dan hasilnya (*output*) juga berupa citra (*image*). Pada awalnya pengolahan citra ini dilakukan untuk memperbaiki kualitas citra, namun dengan berkembangnya dunia komputasi yang ditandai dengan semakin meningkatnya kapasitas dan kecepatan proses komputer, serta munculnya ilmu-ilmu komputer yang memungkinkan manusia dapat mengambil informasi dari suatu citra maka *image*

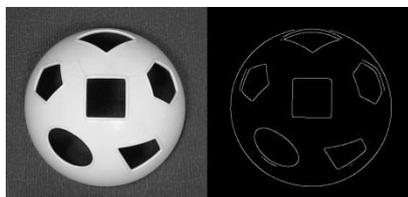
*processing* tidak dapat dilepaskan dengan bidang *computer vision*[8].

### 2.1.1 Tresholding

Proses *thresholding* sering disebut dengan proses binerisasi. Pada beberapa aplikasi pengolahan citra, terlebih dahulu dilakukan *threshold* terhadap citra *gray level* untuk dapat menjadi citra biner (citra yang memiliki nilai level keabuan 0 atau 255)[6]. Sebuah citra hasil proses *thresholding* dapat disajikan dalam histogram citra untuk mengetahui penyebaran nilai-nilai intensitas piksel pada suatu citra/bagian tertentu dalam citra sehingga untuk citra bimodal, histogram dapat dipartisi dengan baik (segmentasi objek dengan *background*) dan dapat ditentukan nilai *threshold*-nya[5].

### 2.1.2 Edge Detection

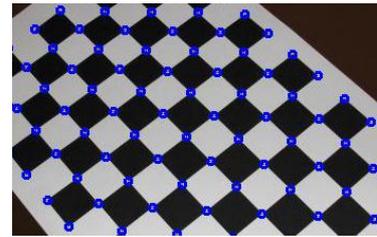
Deteksi tepi (*Edge Detection*) pada suatu citra adalah suatu proses yang menghasilkan tepi-tepi dari objek-objek citra yang tujuannya adalah untuk menandai bagian yang menjadi detail citra dan memperbaiki detail dari citra yang kabur, yang terjadi karena *error* atau adanya efek dari proses akuisisi citra. Tepi adalah perubahan nilai intensitas derajat keabuan yang mendadak (besar) dalam jarak yang singkat. Suatu titik (x,y) dikatakan sebagai tepi (*edge*) dari suatu citra bila titik tersebut mempunyai perbedaan yang tinggi dengan tetangganya. Keberadaan tepi unsur didandaikan dengan perubahan nilai piksel atau kontras. Tepi unsur pada citra dideteksi dengan operator deteksi (detektor) tepi diantaranya berupa matriks template berukuran tertentu[4].



Gambar 2.1 Hasil deteksi tepi

### 2.1.3 Deteksi Sudut

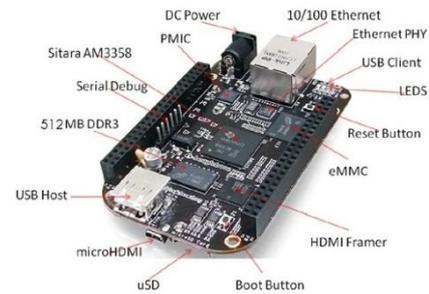
Corner Detection merupakan suatu pendekatan yang digunakan dalam sistem Computer Vision untuk mengekstraksi beberapa jenis fitur dan menyimpulkan isi dari suatu gambar. Corner Detection sering digunakan dalam deteksi gerakan, pencocokan gambar, pelacakan, mosaicing gambar, panorama stitching, pemodelan 3D dan pengenalan obyek.[7]



Gambar 2.2 Hasil deteksi sudut

## 2.2 Beaglebone Black

*Beaglebone Black* merupakan salah satu embedded system yang sering disebut *single-board computer* dan juga *open source hardware*. *Beaglebone Black* mempunyai bawaan OS berbasis Linux Angstro ARM. Dengan kemampuan memori 512 mb DDR3, *ethernet onboard* memungkinkan *Beaglebone Black* dapat terkoneksi dengan internet menggunakan kabel LAN tanpa harus menambahkan modul-modul *ethernet*. *Beaglebone Black* mempunyai banyak pin multiguna disebalah kanan dan kiri *board*-nya yang dapat digunakan sebagai GPIO (*General Purpose Input Output*). Berikut gambar dari *Beaglebone Black*[3].



Gambar 2.3 Beaglebone Black

## 2.3 Mikrokontroler ATmega128

Mikrokontroler ATmega128 adalah mikrokontroler yang diproduksi oleh Atmel dengan arsitektur RISC dan jenis mikrokontroler 8-bit dengan konsumsi daya yang rendah. Pada dasarnya ATmega128 sama dengan ATmega8, yang membedakan adalah memori dan port-portnya.



Gambar 2.4 ATmega128

## 2.4 Kamera Webcam

Kamera adalah alat untuk mengakuisisi citra digital. Citra digital yang ditangkap oleh

kamera tersebut dapat diproses sehingga didapatkan banyak parameter yang dibutuhkan oleh sistem pengendalian robot, diantaranya parameter posisi, parameter bentuk, bahkan parameter kecepatan [6].

Sensor kamera *logitech webcam C170* merupakan kamera dengan kualitas gambar 8MP (*Mega Pixel*). Kamera ini memiliki kualitas resolusi HD 720p (*1280x720 pixels*) *webcam*. Selain itu, kamera ini dapat menyimpan 30 *frames* per detiknya. Dilengkapi kabel USB, sehingga dapat dikoneksikan dengan Beaglebone Black ataupun ke *embedded system*.



Gambar 2.5 Kamera Webcam

### 2.5 Motor Servo

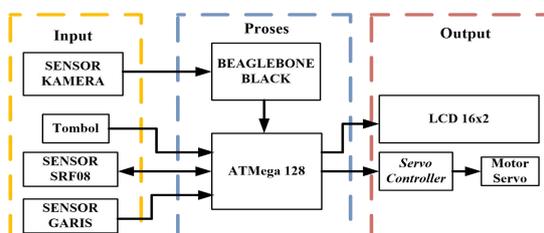
Motor servo adalah sebuah motor yang dirancang dengan sistem kontrol umpan balik loop tertutup, sehingga dapat diatur untuk menentukan dan memastikan posisi sudut dari poros output motor. motor servo merupakan perangkat yang terdiri dari motor DC, serangkaian gear, rangkaian kontrol dan potensiometer.



Gambar 2.6 Motor Servo

## III. METODOLOGI

### 3.1 Diagram Blok Sistem



Gambar 3.1 Diagram Blok sistem

### 3.2 Prinsip Kerja

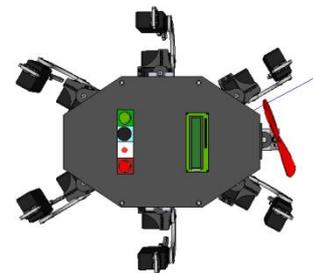
Ketika robot ini dijalankan, robot ini telah memiliki data ukuran tiap-tiap ruang. Kemudian robot berjalan mencari ruangan. Setelah robot menjumpai ruangan, robot akan memposisikan untuk berada tepat di depan pintu ruangan. Setelah robot berada tepat didepan ruangan, kamera akan

mendeteksi garis tepi dari perpotongan dinding dengan lantai yang berada di depan robot. Robot akan berusaha memastikan tepian perpotongan dinding dengan lantai agar tepat lurus sejajar bidang horisontal, hal ini mengindikasikan bahwa robot telah lurus terhadap dinding dan siap melakukan pengukuran. Selanjutnya robot akan melakukan pengidentifikasian ruangan dengan cara melakukan pengukuran panjang dari ruangan tersebut. Kemudian prosesor akan memproses data untuk mencocokkan hasil pengukuran dengan ukuran ruangan yang telah tersimpan sebelumnya untuk mengetahui robot saat ini berada di ruang berapa. Proses ini akan berulang ketika robot menjumpai ruangan selanjutnya.

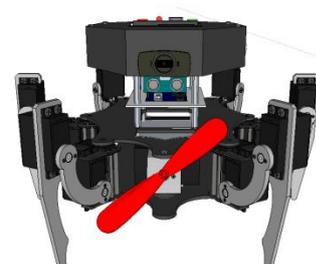
### 3.3 Spesifikasi Alat

1. Dimensi Alat  
Panjang = 31 cm  
Lebar = 25 cm  
Tinggi = 20 cm  
Berat ± 4 kg
2. Spesifikasi  
Sensor kamera Logitech c525  
Sensor SRF08  
Beaglebone Black  
ATMega 128  
LCD 16x2  
Servo kontrol Arduino Nano  
Motor Servo HS645MG dan HS225MG  
Baterai Li-po 7.4v 5400mAh  
Baterai Li-po 11.4v 1000mAh  
UBEC  
Regulator 5v

### 3.4 Desain Mekanik

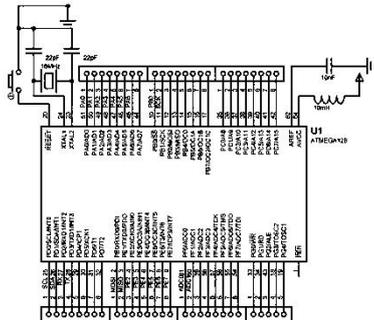


Gambar 3.2 Tampak Atas



Gambar 3.3 Tampak Depan

3.5 Perancangan Rangkaian Minimum Sistem Atmega 128



Gambar 3.4 Rangkaian minimum sistem Atmega 128

3.6 Perancangan Regulator UBEC

Regulator UBEC (*Universal Battery Eliminator Circuit*) digunakan untuk membatasi tegangan yang akan digunakan oleh servo. UBEC memiliki output tegangan 5-6V (melalui selector) dan arus hingga 15A.

Gambar 3.5 UBEC

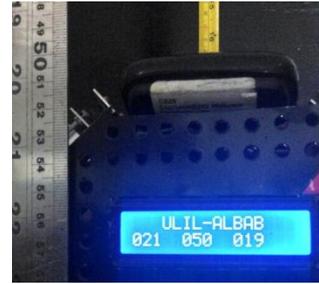
IV PENGUJIAN DAN ANALISA

4.1. Pengujian Sensor Ultrasonic

Pengujian sensor ultrasonic bertujuan untuk mengetahui apakah hasil pengukuran dari sensor sesuai dengan jarak sesungguhnya. Pengujian ini dilakukan dengan melakukan pengukuran mulai dari jarak 10 cm hingga 1 meter



Gambar 4.1 Pengujian SRF08 jarak 30cm



Gambar 4.2 Pengujian SRF08 jarak 50cm



Gambar 4.3 Pengujian SRF08 jarak 75cm



Gambar 4.4 Pengujian SRF08 jarak 100cm

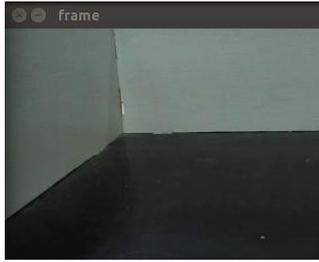
Tabel 1. Hasil pengujian sensor ultrasonic

No	Jarak Asli	Hasil Pengukuran	Error
1	10cm	10cm	0
2	20cm	20cm	0
3	30cm	30cm	0
4	40cm	40cm	0
5	50cm	50cm	0
6	60cm	60cm	0
7	70cm	70cm	0
8	80cm	80cm	0
9	90cm	90cm	0
10	100cm	100cm	0

Dari hasil pengujian sensor ultrasonic diatas, dapat diketahui bahwa sensor ultrasonic SRF08 tidak terdapat error pada hasil pengukuran jarak. Sehingga sensor SRF08 ini baik digunakan sebagai sensor jarak

4.2. Pengujian Sensor Kamera

Pengujian kamera dilakukan untuk mengetahui apakah sensor kamera telah sesuai untuk keperluan proses pengolahan citra. Pengujian dilakukan dengan menggunakan bahasa pemrograman C++ dan *library* OpenCV untuk pemrosesan citra.

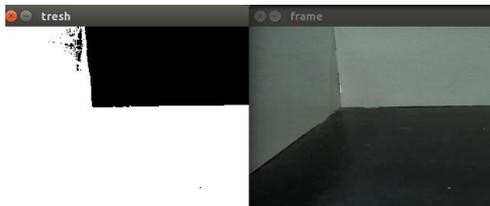


Gambar 4.5 citra asli kamera

Setelah melakukan pengujian citra asli dari sensor, proses selanjutnya adalah melakukan pengolahan citra *tresholding*, deteksi tepi, dan deteksi sudut

4.2.1. Pengujian Nilai Treshold

Pengujian nilai *treshold* dilakukan untuk mengetahui berapa nilai *treshold* yang dibutuhkan untuk memisahkan antara warna lantai dan dinding.



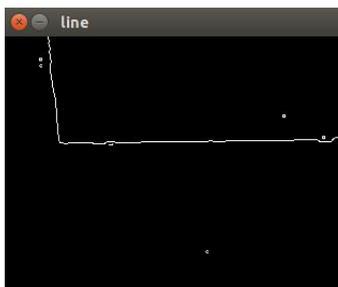
Gambar 4.6 perbandingan citra asli(kanan) dengan citra hasil *tresholding*

Dalam pengujian ini digunakan metode *TRESH\_BINARY\_INV* yang akan menjadikan lantai yang berwarna hitam menjadi citra biner 1 (putih) dan dinding yang berwarna putih menjadi citra biner 0 (hitam)

Setelah melalui proses *tresholding*, selanjutnya dapat dilakukan proses deteksi tepi

4.2.2. Pengujian Deteksi Tepi

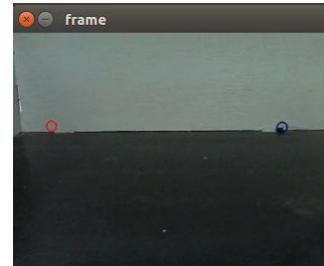
Pengujian deteksi tepi dilakukan untuk mendapatkan garis tepi dari citra yang ditangkap



Gambar 4.7 hasil proses deteksi tepi

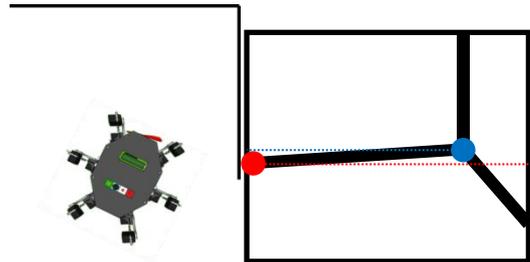
4.2.3. Pengujian Deteksi Sudut

Pengujian deteksi sudut dilakukan untuk mendapatkan sudut-sudut dari ruangan yang akan digunakan sebagai acuan *alignment* dari robot

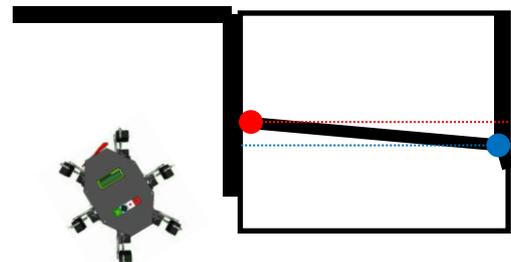


Gambar 4.8 hasil deteksi sudut

Dari hasil deteksi sudut diatas, terdapat dua lingkaran kecil berwarna merah dan biru. Lingkaran merah menandakan sudut sebelah kiri, dan lingkaran biru menandakan sudut sebelah kanan. Setelah didapatkan hasil seperti pada gambar 4.13, maka metode ini telah siap digunakan untuk melakukan *alignment* pada robot. Selanjutnya dilakukan pencarian nilai error untuk menentukan *alignment* pada robot



Gambar 4.9 Ilustrasi ketika robot serong kanan



Gambar 4.10 Ilustrasi ketika robot serong kiri

Dari ilustrasi pada gambar 4.9 diatas dapat dilihat ketika robot serong ke kanan maka keadaan ruang tersebut yang ditangkap oleh kamera. Dapat dilihat pada ilustrasi keadaan ruang diatas bahwa sudut sebelah kiri(merah) terlihat lebih rendah dari sudut kanan(biru). Sedangkan dari ilustrasi pada gambar 4.10 ketika robot

serong ke kiri maka keadaan ruang tersebut yang ditangkap oleh kamera. Dapat dilihat pada ilustrasi keadaan ruang diatas bahwa sudut sebelah kiri(merah) terlihat lebih tinggi dari sudut kanan(biru). Dari kedua ilustrasi diatas dapat disimpulkan bahwa nilai koordinat sumbu y dari kedua sudut yang dideteksi dapat digunakan untuk perhitungan nilai error sebagai berikut:

$$error = Y_{left} - Y_{right}$$

Dari rumus error diatas, ketika nilai koordinat sumbu y sudut sebelah kiri lebih besar dari nilai koordinat sumbu y sebelah kanan maka nilai error bernilai positif. Dalam keadaan nilai error positif ini, robot pada posisi serong kiri sehingga robot harus berputar ke kanan untuk meluruskan dengan dinding didepannya sebagai bidang pengukuran. Sebaliknya ketika nilai koordinat sumbu y sudut sebelah kiri lebih kecil dari nilai koordinat sumbu y sebelah kanan maka nilai error bernilai negatif. Dalam keadaan nilai error negatif ini, robot pada posisi serong kanan sehingga robot harus berputar ke kiri untuk meluruskan dengan dinding didepannya sebagai bidang pengukuran.

#### 4.3. Pengujian Sistem

Pengujian sistem ini bertujuan untuk mengetahui apakah robot dengan menggunakan program – program diatas dapat mengidentifikasi ruangan sesuai dengan tujuan atau tidak. Pengujian ini dilakukan pada seluruh ruangan dengan memberikan gangguan awal pada robot berupa peletakan awal robot serong ke kiri atau serong ke kanan, selanjutnya akan dilihat apakah robot dapat melakukan *alignment* untuk memperbaiki posisi dengan baik atau tidak dan apakah robot berhasil mengidentifikasi setiap ruangan atau tidak. Pengujian dilakukan dengan menggunakan metode deteksi tepi tanpa menggunakan *backup* dari kontur dan dengan *backup* dari kontur

Tabel 2. Persentase Keberhasilan Identifikasi Ruangan

No	Ruang	Tanpa Kontur	Dengan Kontur
1	1	80%	100%
2		90%	90%
3	2	100%	100%
4		90%	100%
5	3	90%	100%
6		100%	100%
7	4	70%	90%
8		60%	90%

#### 4.4. Analisa

Dari seluruh data tabel hasil pengujian robot diatas dapat dilihat bahwa proses *alignment* dan pendeteksian pada masing masing ruangan telah berhasil. Persentase keberhasilan tertinggi terdapat pada ruang 2 dan ruang 3, namun persentase error tertinggi terdapat pada ruang 4. Namun terdapat perbedaan hasil dari penggunaan proses kontur dan tanpa penggunaan proses kontur. Dimana persentase keberhasilan *alignment* dan identifikasi pada ruang 4 naik. Hal ini dikarenakan ruang 4 adalah ruang dengan ukuran panjang ruang terkecil. Karena ukurannya yang kecil, membuat dinding pada depan robot yang menjadi bidang pengukuran menjadi sangat dekat dengan robot. Hal ini membuat tampilan pada hasil penangkapan citra semakin besar. Selain itu dengan semakin dekat nya jarak antar dinding tersebut, membuat cahaya dari lampu yang masuk terhalangi. Hal ini dapat mengakibatkan nilai *threshold* akan berbeda dengan nilai *threshold* pada ruangan lain yang lebih luas. Dengan berbedanya nilai *threshold* pada ruangan ini akan mempengaruhi metode deteksi tepi dan deteksi sudut. Sedangkan dengan menggunakan proses kontur, titik terluar dari citra yang ditangkap tetap dapat terdeteksi sehingga sangat membantu untuk menggantikan sudut yang tidak terdeteksi sebelumnya.

## V. PENUTUP

### 5.1. Kesimpulan

Setelah melakukan penelitian, pengujian, dan menganalisa, maka dapat ditarik beberapa kesimpulan yaitu:

1. Penerapan kamera sebagai sensor pada robot berkaki dapat dilakukan dengan cara pengolahan citra pada *embedded system beaglebone black* menggunakan *library opencv*.
2. Dengan menggunakan sensor kamera untuk mendeteksi sudut-sudut pada ruangan, robot dapat melakukan *alignment* untuk memposisikan agar sejajar dengan bidang pengukuran. Dengan menggunakan metode deteksi sudut dan ditambah dengan kontur akan membuat proses *alignment* pada robot berjalan lebih baik
3. Dengan menggunakan sudut dari ruangan sebagai acuan *alignment* robot, robot dapat melakukan pengukuran ruangan dengan baik sebagai metode untuk mendeteksi ruangan.

### 5.2. Saran

1. Mencoba menggunakan *embeded system* yang lain dengan spesifikasi lebih tinggi untuk mempercepat proses pengolahan citra (*image processing*).

2. Mencoba metode deteksi tepi dan deteksi sudut yang lain untuk mengurangi *noise* pada hasil pengolahan citra
3. Mencoba menggunakan tipe kamera lain yang memiliki *noise* yang rendah, sehingga pemrosesan citra menjadi lebih mudah.

## VI. DAFTAR PUSTAKA

- [1]Aydan, Mohammad Habib. 2015. *Implementasi Kontrol Pid Pada Motor Dc Sebagai Penggerak Robot Beroda 4wd Omni Wheels Dengan Metode Wall Following*. Skripsi Teknik Elektro Politeknik Negeri Malang
- [2]Budiharto, Widodo dan Djoko Purwanto. 2015. *Robot Vision Teknik Membangun Robot Cerdas Masa Depan*. Yogyakarta: Andi Offset
- [3]Datasheet *Beaglebone Black*, [http://www.adafruit.com/datasheets/BBB\\_SRM.pdf](http://www.adafruit.com/datasheets/BBB_SRM.pdf), diakses tanggal 15 Oktober 2015
- [4]Haryono, Dani. 2015. *Deteksi tepi Objek dengan Kamera pada Robot Humanoid menggunakan Metode Deteksi Tepi (Edge Detection)*. Skripsi Teknik Elektro Politeknik Negeri Malang
- [5]Hermawan, Tito. 2014. *Pengolahan Warna Pada Robot Untuk Menyeleksi Warna R.G.B.* Skripsi Program Studi Teknik Elektronika, Jurusan Teknik Elektro. Malang : Politeknik Negeri Malang
- [6]Kusumanto, R.D. 2011. *Klasifikasi Warna Menggunakan Pengolahan Model Warna HSV*. Jurnal Ilmiah Elite Elektro, Vol 2, No. 2, September 2011: 83-87. Palembang : Politeknik Negeri Sriwijaya
- [7]Marsetio, Sany Aji. 2014. *Deteksi Sudut Pada Gambar 2d Berurutan Dengan Menggunakan Metode Harris/Plessey Corner Detector*. Jurnal Jurusan Teknik Informatika. Politeknik Elektronika Negeri Surabaya
- [8]Mulyawan, Hendi. 2011. *Identifikasi dan Tracking Obyek Berbasis Image Processing Secara Real Time*. Jurnal Teknik Telekomunikasi. Politeknik Elektronika Negeri Surabaya
- [9]Pradana, Ferdian. 2014. *Perancangan Dan Implementasi Visual Servoing Pada Robot Beroda Menggunakan Kamera Berbasis Beaglebone Black*. Skripsi Teknik Elektro Politeknik Negeri Malang