

# IMPLEMENTASI *AUTO THRESHOLD* PADA SENSOR KAMERA UNTUK *WALL FOLLOWING* ROBOT KRPAI BERKAKI

Dyky Agustiono Lius, Totok Winarno, Sidik Nurcahyo

**Abstrak** –Pada pengolahan citra (*Image Processing*) salah satu metode yang sederhana untuk digunakan adalah metode segmentasi hitam putih (*threshold*). *Threshold* adalah citra digital yang hanya memiliki dua kemungkinan nilai *pixel* yaitu hitam dan putih. Namun dalam penentuan nilai ambang *threshold* akan berbeda-beda tergantung pada intensitas cahaya yang ditangkap oleh kamera. Oleh karena itu, diperlukan proses lanjutan dari *threshold* untuk mendapatkan nilai ambang *threshold* secara otomatis pada kondisi cahaya yang berbeda-beda yaitu yang disebut dengan *Auto Threshold*. *Auto Threshold* merupakan algoritma untuk mendapatkan nilai ambang *threshold* secara otomatis dengan cara mencari nilai *threshold* awal sebagai referensi lalu memperbaikinya menggunakan informasi dari sebaran intensitas warna abu-abu. Salah satu objek yang memiliki intensitas cahaya yang berbeda adalah arena robot KRPAI berkaki. Arena tersebut berbentuk seperti labirin yang memiliki dinding berwarna putih dan lantai berwarna hitam, namun intensitas warna tersebut berbeda-beda tiap ruangnya. Dengan menggunakan *beaglebone black* sebagai *embedded system* untuk melakukan proses pengolahan citra didapatkan bahwa ketika kondisi cahaya redup pada intensitas cahaya 42 lux diperoleh nilai ambang *threshold* 52, sedangkan untuk kondisi cahaya terang dengan intensitas 50 lux diperoleh nilai ambang *threshold* 85 dan untuk kondisi cahaya sangat terang dengan intensitas 53,5 lux diperoleh nilai ambang *threshold* 94. Untuk mengontrol pergerakan robot dalam *wall following* menggunakan kontrol PID dan diperoleh nilai dari parameter PID adalah  $kp=5$ ,  $ki=0.3$ ,  $kd=0,5$  dimana robot untuk dapat stabil dalam menemukan jalan setelah berhadapan dengan dinding memerlukan waktu selama 8 detik.

**Kata Kunci :** *Image Processing*, *Threshold*, *Auto Threshold*, Kontrol PID.

Dyky Agustiono Lius adalah Mahasiswa D4 Teknik Elektronika Jurusan Teknik Elektro Politeknik Negeri Malang, email: [dyky\\_agustiono@yahoo.co.id](mailto:dyky_agustiono@yahoo.co.id).

Totok Winarno adalah dosen Jurusan Teknik Elektro Politeknik Negeri Malang

Sidik Nurcahyo adalah dosen Jurusan Teknik Elektro Politeknik Negeri Malang.

## I. PENDAHULUAN

Pengolahan citra adalah setiap bentuk pengolahan sinyal dimana input adalah gambar, seperti foto atau video bingkai, sedangkan output dari pengolahan citra dapat berupa gambar atau sejumlah karakteristik atau parameter yang berkaitan dengan gambar.

Banyaknya informasi yang diperoleh serta tingkat fleksibilitas yang tinggi membuat pengolahan citra menjadi salah satu alternatif dalam pengembangan sensor pada sebuah robot.

Oleh karena itu, pada kesempatan kali ini penulis membuat suatu pengembangan pada robot berkaki yaitu menerapkan kamera sebagai sensor untuk manuver robot dalam menelusuri dinding (*wall following*). Pada konsep pengolahan citra salah satu metode yang dapat digunakan untuk mendeteksi dinding adalah dengan menggunakan metode *threshold*. *Threshold* adalah suatu metode yang digunakan untuk memisahkan antara obyek dan backgroundnya.

Agar robot dapat menjaga jarak dengan dinding (tidak menabrak) dinding maka digunakan kontrol PID yang mengatur pergerakan dari robot agar tetap pada jarak yang aman dengan dinding. Pada kontrol PID diperlukan pengaturan dalam menentukan nilai dari konstanta *proportional* ( $kp$ ), konstanta *integral* ( $ki$ ), dan konstanta *derivative* ( $kd$ ) sehingga diperlukan proses pengaturan (*tuning*).

Pada sistem robot ini sensor citra menggunakan kamera dan proses pengolahan citra dilakukan pada *embedded system beaglebone black*. Selain kamera digunakan pula sensor ultrasonik untuk mengukur jarak pada sisi kanan dan kiri robot karena pada sisi-sisi robot kamera tidak dapat mengaksesnya. Sedangkan proses kontrol PID dilakukan pada mikrokontroler atmega128 dan aktuator menggunakan motor servo yang dikendalikan oleh servo controller.

## II. TINJAUAN PUSTAKA

### 2.1 Pengolahan Citra

Pengolahan citra merupakan proses pengolahan dan analisis citra yang banyak melibatkan persepsi visual. Proses ini mempunyai ciri data masukan dan informasi keluaran yang berbentuk citra.[1]

Salah satu metode dalam pengolahan citra adalah *threshold*. *Threshold* adalah suatu metode yang digunakan untuk memisahkan antara obyek

dan backgroundnya. 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).[7]

2.2 Auto Threshold

Binerisasi otomatis dalam hal penentuan nilai *threshold* diperlukan untuk mengetahui karakteristik citra yang meliputi: Sifat intensitas dari citra, Ukuran citra, Bagian citra yang di tempati objek, Jumlah dan jenis yang berbeda dari objek-objek yang muncul dalam citra.[2]

Cara menentukannilai *threshold* yaitudengan di ulangulangsampainilai yang cocok di temukan.Nilaidimulaidenganmenentukannilaikira-kirauntuk *threshold* sebagainilaiawal.Kemudiansecaraberturut-turutmemperbaikinyamenggunakaninformasisebaranintensitas citra yang bersangkutan.Langkah :

1. Penentuannilaiawal *threshold* yaitudari nilai rata ratadariintensitas citra.
2. Citra di bagidua daerhdengan menggunakan nilai awal *threshold* padalangkah 1. (R1 dan R2).
3. Hitung nilai rata-rata intensitas padatiapdaerahnya ( $\mu_1$  dan  $\mu_2$ ).
4. Hitung nilai *threshold* barudari penjumlahannilai rata-rata intensitas yang di bagi 2.
5. Ulangilangkah 2 hingga 4 sampainilai rata-rata intensitas ( $\mu_1$  dan  $\mu_2$ ) tidak berubah. [2]

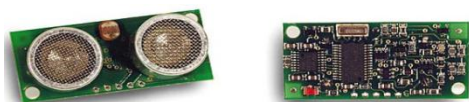
2.3 Sensor Kamera



Gambar 1 Sensor Kamera

Sensor kamera *logitech webcam C525* merupakan kamera dengan kualitas gambar 8MP (Mega Pixel). Kamera ini memiliki kualitas resolusi *Full HD* (1280x720) *webcam*. Selain itu, kamera ini dapat menyimpan 30 *frames* per detiknya. Dilengkapi kabel USB, sehingga dapat dikoneksikan dengan Beaglebone Black ataupun ke *embedded system*.

2.4 Sensor Ultrasonik SRF-08

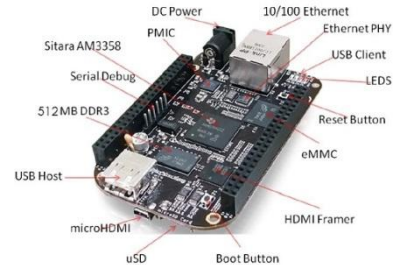


Gambar 2 Sensor Ultrasonik

Sensor SFR08 adalah sensor ultrasonik yang diproduksi oleh Devantech. Sensor ini merupakan sensor jarak yang presisi. Dapat melakukan

pengukuran jarak 3 cm sampai 6 meter dan sangat mudah untuk dihubungkan ke mikrokontroler menggunakan sebuah pin Input dan pin Output.[3]

2.5 Beaglebone Black



Gambar 3 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 *boardnya* yang dapat digunakan sebagai GPIO (*General Purpose Input Output*).[4]

2.6 Mikrokontroler Atmega128



Gambar 4 Mikrokontroler Atmega128

ATmega128 banyak digunakan untuk sistem yang kompleks. Termasuk pada robot ini, dimana atmel yang digunakan juga atmega128. Karena memiliki input sinyal analog, dan membutuhkan memori yang cukup besar.

2.7 Servo Controller



Gambar 5 Modul Arduino Nano

*Servo Controller* adalah controller motor servo yang telah didesain untuk digunakan dalam mengatur sudut dari motor servo yang akan digunakan sebagai aktuator pada robot berkaki, dimana motor servo dikendalikan dengan sinyal PWM standar, seperti pada robot *hexapod*.

### 2.8 Motor Servo

Motor Servo, disebut juga *control motors*, adalah motor listrik yang dibuat dan didesain khusus digunakan untuk *feedback* pada sistem control, sebagai keluaran *actuator*. Motor servo memiliki kecepatan respon yang tinggi, yang membutuhkan kelembaman rotor yang rendah. Oleh karena itu, motor ini diameternya kecil dan panjang pada ukuran panjangnya. Normalnya kecepatan motor servo rendah atau lambat dan torsi yang besar. Motor servo bisa digunakan pada berbagai aplikasi misalkan robot, radar, computer, dll. [5]

Pada robot berkaki ini menggunakan 2 jenis motor servo yaitu motor servo HS-225MG dan motor servo HS-645MG.



Gambar 6 Motor Servo HS-225MG dan HS-645MG

### 2.9 Kontrol PID

PID (darisingkatan bahasa Inggris: Proportional-Integral-Derivative controller) merupakan sistem kontroler untuk menentukan presisi suatu sistem instrumentasi dengan adanya umpan balik pada sistem tersebut.

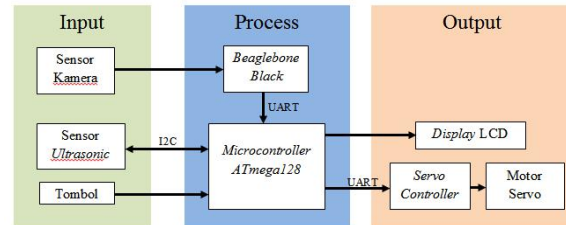
Set point merupakan keadaan tertentu yang ingin dicapai. Sensor berfungsi mendeteksi keluaran plant dan mengkonversikannya menjadi besaran dengan satuan yang sama seperti satuan set point. Keluaran sensor merupakan sinyal umpan balik (feedback) yang akan dikurangkan dengan set point menghasilkan sinyal error. Jika suatu dan dari keluaran sudah samadengan set point maka blok sensor dapat dihilangkan sehingga sinyal umpan balik (feedback) adalah keluaran plant. Sinyal error diproses oleh kontroler lalu menghasilkan sinyal kontrol yang diumpankan ke plant, dengan tujuan agar keluaran dari plant samadengan set point, yang berarti sinyal error bernilai (atau mendekati) nol.

Karakteristik kontroler PID sangat dipengaruhi oleh kontribusi nilai dari ketiga parameter P, I dan D. Penyetelan konstanta  $K_p$ ,  $K_i$ , dan  $K_d$  akan mengakibatkan penonjolan sifat dari masing-masing elemen. Satu atau dua dari ketiga konstanta tersebut dapat disetel lebih menonjol dibanding yang lain. Konstanta yang menonjol itulah akan memberikan kontribusi pengaruh pada respons sistem secara keseluruhan [6]

$$u(k) = K_p e_k + K_i T \sum_0^k e_k + \frac{1}{T} K_D (e_k - e_{k-1}) \quad (1)$$

## III. METODOLOGI

### 3.1 Blok Diagram Sistem



Gambar 7 Blok Diagram Sistem

### 3.2 Prinsip Kerja

Sebelum robot dapat melusuri dinding, robot diletakkan pada jarak tertentu terhadap dinding untuk dilakukan proses pengambilan visual oleh kamera webcam yang kemudian dilakukan pengolahan citra (*image processing*) oleh *beaglebone black*. Menggunakan metode *threshold* citra visual akan menjadi visual biner (1 dan 0) yang kemudian proses selanjutnya adalah mencari nilai *threshold* yang tepat berdasarkan kondisi pencahayaan pada lingkungan robot, penentuan jarak antara robot dengan dinding dimaksudkan sebagai tolak ukur untuk mencari nilai *threshold* yang tepat agar robot dapat mendeteksi adanya dinding dengan baik.

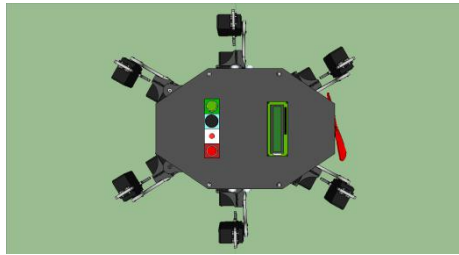
Setelah robot dapat mendeteksi keberadaan dinding, kemudian *beaglebone black* akan mengirimkan data serial melalui UART menuju mikrokontroler ATmega128. Setelah data keberadaan dinding dapat diterima, mikrokontroler melakukan proses kontrol pergerakan robot dengan mengirimkan perintah pergerakan ke *servo controller*, pengiriman perintah ke *servo controller* dilakukan dengan serial menggunakan UART. Selain sebagai kontrol pergerakan robot mikrokontroler juga menampilkan data berupa posisi dinding pada LCD 16x2 dan juga digunakan untuk menentukan *start* dan *stop* pada robot.

### 3.3 Spesifikasi dan Perancangan Mekanik

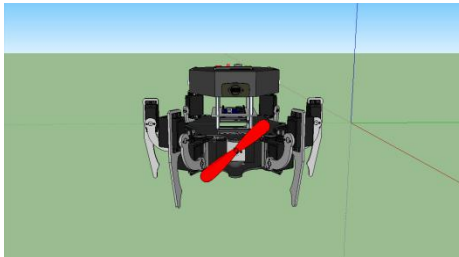
#### 3.3.1 Spesifikasi Mekanik

- Panjang: 30 cm
- Lebar: 25 cm
- Tinggi: 20 cm
- Berat ± 4 kg

#### 3.3.2 Perancangan Mekanik

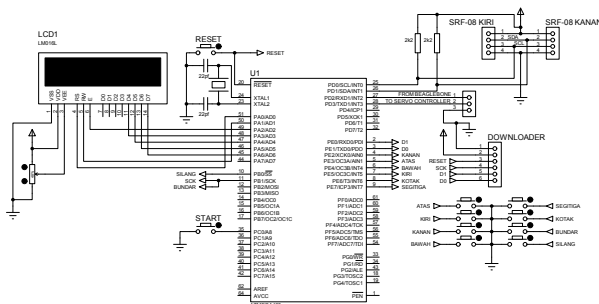


Gambar 8 Desain Robot (Tampak Atas)



Gambar 9 Desain Robot (Tampak Depan)

3.4 Perancangan Sistem Elektrik



Gambar 10 Skematik Rangkaian *Minimum System* Atmega128

Pada rangkaian *minimum system* atmega128 diatas juga terdapat beberapa rangkaian lain yaitu :

1. Rangkaian *display* LCD 16x2.
2. Rangkaian *oscillator eksternal*.
3. Rangkaian *pull-up* untuk sensor ultrasonik SRF-08.
4. Rangkaian *downloader*.
5. Rangkaian tombol.

3.5 Perancangan Kontrol PID

Pada robot berkaki ini setpoint untuk kontrol PID adalah titik tengah sumbu x pada kamera yaitu nilai dari resolusi kamera secara horizontal dibagi 2. Resolusi yang digunakan pada kamera adalah 320x240 dimana nilai resolusi ini adalah nilai resolusi maksimal yang dapat diolah beaglebone black sehingga nilai setpoint adalah.

$$\text{Setpoint} = \text{Resolusi}_X / 2$$

Sehingga :  $\text{Setpoint} = 320 / 2 = 160$

Sedangkan untuk nilai error merupakan selisih dari setpoint (SP) dengan process variabel (PV). Nilai PV merupakan titik tengah pada lantai

arena yang diperoleh setelah melakukan proses pengolahan citra (image processing) oleh beaglebone black. Pada kontroller PID terdapat 3 parameter konstanta yang mempengaruhi kinerja pada kontroller ini yaitu konstanta proporsional (kp), konstanta integral (ki), dan kontanta derivative (kd).

Pada kontrol proporsional berfungsi sebagai memperkuat sinyal error sehingga akan mempercepat respon sistem untuk mencapai kestabilan. Penerapannya pada sistem adalah.

$$P = k_p * \text{error};$$

Sedangkan kontrol integral bertujuan untuk mengurangi kesalahan keadaan tunak (offset) yang ditimbulkan oleh kontrol proporsional yaitu dengan cara mengintegalkan nilai error atau dengan kata lain menjumlahkan nilai error mulai dari error pertama hingga error. Sehingga dapat disimpulkan bahwa error yang sebelumnya dijumlahkan dengan error-error yang sebelumnya hingga error yang sekarang. Penerapannya pada sistem adalah.

$$\text{sum\_error} = \text{sum\_error} + \text{error};$$

$$I = k_i * \text{sum\_error};$$

Kontrol derivative bertujuan sebagai pengendali laju karena output kontroller sebanding dengan laju dari perubahan sinyal error, kontrol derivative juga sering disebut sebagai fungsi turunan sehingga dapat disimpulkan bahwa error yang sekarang dikurangi error yang sebelumnya. Penerapannya pada sistem adalah.

$$D = k_d * (\text{error} - \text{last\_error});$$

$$\text{last\_error} = \text{error};$$

Sehingga penerapan rumus PID pada sistem adalah.

$$\text{PID} = P + I + D;$$

IV. PENGUJIAN DAN ANALISA

4.1 Pengujian Sensor Ultrasonik

Pengujian pada sensor ultrasonik menggunakan mikrokontroller atmega128 yang telah ditanamkan program bahasa C, pengujian dilakukan pada 2 buah sensor yaitu SRF-08 kiri dan SRF-08 kanan.

Tabel. 1 Hasil Pengujian SRF-08 Kiri

Jarak Aktual (cm)	Jarak Terukur (cm)	Error (%)
10	10	0
15	15	0
20	20	0
25	25	0
30	30	0
35	36	2,8
40	40	0
45	45	0
50	50	0
Rata-rata Error		0,3

Tabel 2 Hasil Pengujian SRF-08 Kanan

Jarak Aktual (cm)	Jarak Terukur (cm)	Error (%)
10	11	10
15	15	0
20	20	0
25	25	0
30	30	0
35	35	0
40	40	0
45	45	0
50	49	2
Rata-rata Error		1,3

Dari hasil pengujian sensor ultrasonik SRF-08 terlihat masih adanya *error* dengan rata-rata *error* 0,3% untuk SRF-08 kiri dan 1,3% untuk SRF-08 kanan namun nilai *error* masih terbilang kecil sehingga sensor ultrasonik SRF-08 masih layak untuk digunakan.

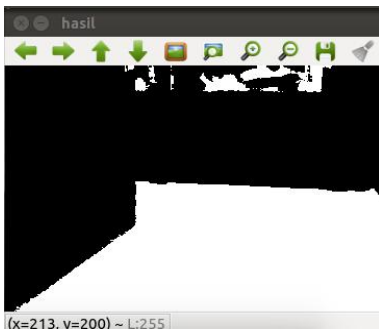
4.2 Pengujian Kamera

Pengujian pada kamera dilakukan untuk mengetahui apakah kamera tersebut mampu untuk dilakukan proses pengolahan citra. Pengujian menggunakan program bahasa C++ dengan menggunakan *library* tambahan khusus untuk *image processing* yaitu OpenCV.



Gambar 11 Hasil Pengujian Citra Asli Kamera

Setelah kamera dapat menangkap citra asli / citra aktual maka proses selanjutnya adalah melakukan proses pengolahan citra *gray* lalu *threshold* agar citra yang dihasilkan adalah citra biner.



Gambar 12 Hasil Pengujian Citra Biner Kamera

Metode *threshold* yang digunakan adalah metode *THRESH\_BINARY\_INV* yang akan menghasilkan citra seperti pada gambar 19.

Lantai yang awalnya berwarna hitam setelah dilakukan proses *threshold* akan menjadi citra biner 1 (putih) sedangkan untuk dinding yang awalnya berwarna putih berubah menjadi citra biner 0 (hitam) setelah dilakukan proses *threshold*.

Setelah citra biner didapatkan maka proses selanjutnya adalah mencari titik tengah dari lantai yang menjadi referensi pergerakan dari robot. Proses ini menggunakan kontur untuk mencari titik tengah atau *center of gravity* dari lantai.

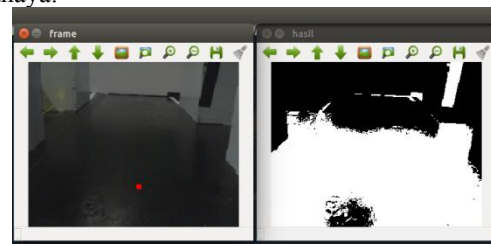


Gambar 13 Hasil Pengujian Kontur Kamera

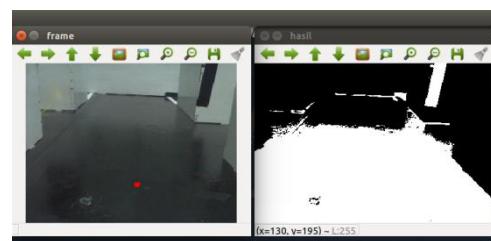
Titik yang berwarna merah pada gambar 4.3 adalah titik tengah atau *center of gravity* dari lantai, koordinat X dari titik tersebut yang digunakan sebagai acuan pergerakan dari robot.

4.3 Pengujian *Auto Threshold*

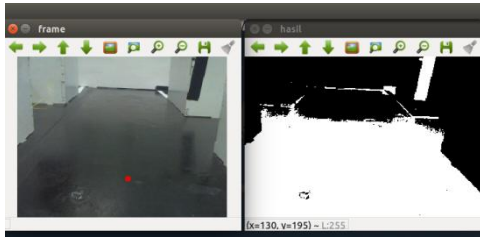
Penentuan nilai *threshold* akan berubah-ubah sesuai dengan kondisi pencahayaan pada lingkungan sekitar robot. Oleh karena itu, diperlukan algoritma untuk proses binerisasi otomatis (*Autothreshold*) dengan metode ini nilai *threshold* akan berubah tergantung dengan kondisi cahaya.



Gambar 14 Hasil Pengujian *Autothreshold* Kondisi Cahaya Redup



Gambar 15 Hasil Pengujian *Autothreshold* Kondisi Cahaya Terang



Gambar 16 Hasil Pengujian *Autothreshold* Kondisi Cahaya Sangat Terang

Tabel 3 Perbandingan Kondisi Cahaya, Intensitas Cahaya dan Nilai *Threshold*

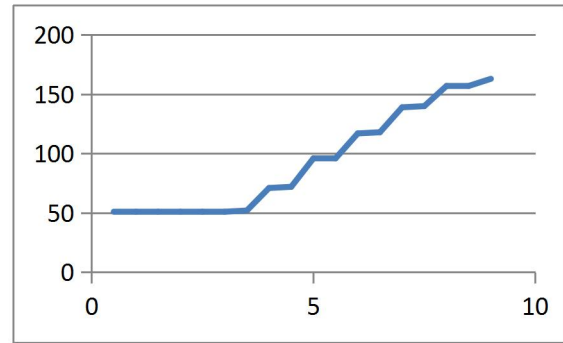
Kondisi Cahaya	Intensitas Cahaya (Lux)	<i>Threshold</i>
Sangat Terang	53,5	94
Terang	50	85
Redup	42	52

Berdasarkan hasil pengujian pada gambar 14, gambar 15, gambar 16, dan tabel 3 dapat disimpulkan bahwa kondisi cahaya berbanding lurus dengan nilai lux dan berbanding lurus pula dengan nilai threshold. Ketika kondisi pencahayaan redup nilai intensitas cahaya adalah yang terkecil dari ketiga kondisi yaitu 42 lux begitu pula dengan nilai threshold yang dihasilkan yaitu 52. Dan ketika kondisi pencahayaan terang dapat dilihat bahwa nilai intensitas cahaya naik dari sebelumnya yaitu menjadi 50 lux dan nilai threshold yang naik secara signifikan menjadi 85. Sedangkan pada kondisi cahaya sangat terang nilai intensitas cahaya adalah yang tertinggi yaitu 53,5 lux dan nilai threshold yang naik menjadi 94.

#### 4.4 Pengujian dan Analisa Kontrol PID

Pengujian kontrol PID dilakukan dengan tujuan agar pergerakan robot dapat stabil pada setpoint yang telah ditentukan sebelumnya yaitu robot dapat menelusuri dinding dengan baik serta respon yang cepat untuk menemukan jalan yang aman untuk dilewati oleh robot. Pada perancangan kontrol PID sebelumnya telah dilakukan tuning nilai konstanta proportional, integral, dan derivative, dimana tuning PID menggunakan metode *Trial and Error*

- $K_p = 5$  ;  $K_i = 0,3$  ;  $K_d = 0,5$
- Respon Sistem



Gambar 17 Grafik Respon PID

- Waktu menuju kestabilan : 8 detik

Oleh karena itu, hasil dari pengujian kontrol PID nilai  $k_p$ ,  $k_i$ , dan  $k_d$  yang terbaik untuk robot berkaki ini adalah  $K_p = 5$  ;  $K_i = 0,3$  ; dan  $K_d = 0,5$  yang diperoleh dengan menggunakan metode tuning PID trial and error.

## V. PENUTUP

### 5.1 Kesimpulan

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

1. Berdasarkan hasil pengujian Auto Threshold dapat dilihat pada gambar 14, 15, dan 16 robot tetap dapat mendeteksi posisi dinding dan lantai meskipun dalam kondisi cahaya yang berbeda-beda yaitu ketika kondisi cahaya redup dengan intensitas cahaya = 42 lux, nilai threshold yang dihasilkan adalah 52. Sedangkan ketika kondisi cahaya terang dengan intensitas cahaya = 50 lux menghasilkan nilai threshold = 85 dan ketika kondisi cahaya sangat terang dengan intensitas cahaya 53,5 menghasilkan nilai threshold = 94.
2. Berdasarkan hasil pengujian kontrol PID sebagai kontrol pergerakan robot berkaki, robot dapat kembali menemukan jalan yang tepat dengan menggunakan metode tuning trial and error dengan nilai  $K_p = 5$ ,  $K_i = 0,3$ ,  $K_d = 0,5$  dan memerlukan waktu untuk stabil selama 8 detik dari posisi awal robot menghadap dinding..

### 5.2 Saran

1. Sebagai aktuator pada robot berkaki dapat menggunakan servo jenis lain yang pergerakan sudutnya lebih cepat seperti menggunakan jenis servo digital.
2. Menggunakan *embedded system* yang lain untuk proses pengolahan citra (*image processing*) seperti *Raspberry-pi*.

## DAFTAR PUSTAKA

- [1] Setyawan, Nizar. 2015. Pengaplikasian Human Gesture Recognition pada Lengan Robot Humanoid dengan Metode K-Means Clustering.

- Skripsi Program Studi Teknik Elektronika,  
Jurusan Teknik Elektro. Malang : Politeknik  
Negeri Malang.
- [2] Astuti, Setia. 2010. *Binerisasi Otomatis pada  
Citra Bergradasi dengan Metode Variabel dan Metode Iterasi*. Jurnal Techno.com, Vol.9 No.3, Agustus  
2010. Semarang : UDINUS Semarang
- [3] Aydan, Habib. 2015. *Implementasi Kontrol PID  
Pada Motor DC Sebagai Penggerak Robot  
Beroda 4WD Omni Wheels dengan Metode Wall  
Following*. Skripsi Program  
Studi Teknik Elektronika, Jurusan Teknik Elektro.  
Malang : Politeknik Negeri Malang
- [4] *Datasheet Beaglebone  
Black*, [http://www.adafruit.com/datasheets/BBB  
\\_SRM.pdf](http://www.adafruit.com/datasheets/BBB_SRM.pdf), diakses tanggal 24 Oktober 2015
- [5] Yudi, Taufiq. 2012. *Perencanaan dan Pembuatan Robot  
Cerdas Berkaki dengan Teknik Wall Following*.  
Laporan Akhir Program Studi Elektronika,  
Jurusan Teknik Elektro.  
Malang : Politeknik Negeri Malang
- [6] Wikipedia, 2016. PID, ([https://id. wikipedia.  
org/ wiki /PID](https://id.wikipedia.org/wiki/PID)), (Online), Diakses 5 Agustus  
2016.
- [7] Putra ,Darma. 2016. *Pengolahan Citra Digital*.  
Yogyakarta : Penerbit Andi