

## Implementasi *Controller* PID (*Proportional, Integral, Derivative*) pada Robot Sepak Bola Beroda

Moh. Ismail Marzuki<sup>1</sup>, Intan Nur Farida<sup>2</sup>, Julian Sahertian<sup>3</sup>

<sup>1,2</sup>Teknik Informatika, Fakultas Teknik, Universitas Nusantara PGRI Kediri

E-mail: <sup>1</sup>ismail.zuxi@gmail.com, <sup>2</sup>in.nfarida@gmail.com, <sup>3</sup>juliansahertian@unpkediri.ac.id

**Abstrak** – salah satu wadah untuk perkembangan robotika dibidang pendidikan adalah adanya Kontes Robot Indonesia (KRI) dimana tahun 2020 dilaksanakan secara online dikarenakan adanya wabah Covid-19. Salah satu divisi KRI yaitu Kontes Robot Sepak Bola Indonesia (KRSBI) Beroda. Konsep pertandingannya yaitu robot dituntut untuk mencetak gol ke gawang sebanyak-banyaknya dengan waktu yang singkat. Robot sebelumnya tidak menerapkan sistem *controller* yang mampu membuat robot bergerak cepat mencari bola dan menendang bola kearah gawang, sehingga kesulitan dalam mencetak gol. Penelitian ini menggunakan sistem *controller* PID (*Proportional, Integral, Derivative*) dengan menggunakan sistem *controller* PID ini robot mampu berjalan mencari bola dengan lebih cepat dan lebih baik dibanding tanpa menggunakan sistem *controller* sehingga robot mampu mencetak gol ke gawang lebih cepat dan banyak. Dari hasil pengujian *tunning* PID didapatkan nilai  $k_P$ ,  $k_I$ , dan  $k_D$  terbaik yaitu dengan nilai  $k_P = 5$ ,  $k_I = 5$ ,  $k_D = 10$  dengan waktu 03.15 detik dengan hasil ini respon pergerakan robot lebih cepat dan *error* yang kecil.

**Kata Kunci** — *Controller, KRSBI, Mencari Bola, PID*

### 1. PENDAHULUAN

Perkembangan teknologi robotika saat ini berkembang begitu pesat. Salah satu wadah perkembangan robotika dibidang pendidikan adalah dengan adanya Kontes Robot Indonesia (KRI). Pada tahun 2020 KRI diselenggarakan oleh Pusat Prestasi Nasional (PUSPRESNAS) yang pelaksanaannya secara daring atau *online* dikarenakan adanya wabah penyakit Covid-19. Pada KRI terdapat beberapa divisi salah satunya divisi Kontes Robot Sepak Bola Indonesia (KRSBI) Beroda. Konsep pertandingannya yaitu setiap tim menampilkan 2 robot dimana robot mampu mencari bola dan saling mengumpan antara robot satu dan dua, setelah berhasil mengumpan robot mampu mencetak gol dengan menendang bola kearah gawang. Pada lapangan juga terdapat dummy robot sebagai halangan atau rintangan yang diletakkan pada sisi lapangan dan sisi gawang.

Robot dituntut untuk mencetak gol sebanyak-banyaknya sampai waktu pertandingan habis. Pada robot terdapat kamera untuk mendeteksi obyek tertentu diantaranya mendeteksi bola, mendeteksi robot lain, mendeteksi gawang. Pada pertandingan tersebut robot harus mampu mencari bola dan menendang bola ke gawang dengan cepat, apabila robot lambat bergerak dalam mencari bola maka akan kehabisan waktu sehingga tidak bisa mencetak gol ke gawang. Pada KRSBI Beroda terdapat 3 sesi dimana masing-masing tim diberi waktu penampilan 3 menit dalam tiap sesi. Untuk menentukan juara pada divisi KRSBI Beroda adalah tim yang paling banyak mencetak gol ke gawang dan meminimalisir pelanggaran. Pada robot sepak bola sebelumnya

belum menerapkan sistem *controller* yang mampu membuat robot bergerak cepat. Sehingga robot masih lambat dalam mencari dan menendang bola.

Pada penelitian sebelumnya yang membahas tentang sistem pendeteksian bola pada robot sepakbola beroda berbasis filter warna. pada penelitian tersebut menghasilkan robot sepak bola beroda yang dibuat untuk mengikuti ajang Kontes Robot Sepak Bola Indonesia Beroda (KRSBI) di Universitas Brawijaya Malang. Dari hasil pengujian pada penelitian tersebut memperoleh nilai akurasi uji coba jarak 100% [1].

Terdapat penelitian lain yang membahas tentang sistem kendali PID untuk pencarian arah gawang lawan pada robot sepak bola beroda dari hasil *tunning* PID dengan nilai  $k_p = 0,1$  ;  $k_i = 0,2$  ;  $k_d = 0,2$  didapatkan respon yang stabil dengan jumlah osilasi = 1, *setling time* = 1,320 detik, persen *overshoot* sebesar 78,26% dan *rise time* sebesar 0,04 detik [2].

Pada penelitian lain membahas tentang perancangan navigasi pada robot sepak bola beroda menggunakan metode PID (*Proportional, Integer, Derivatif*) penelitian tersebut menghasilkan nilai PID yang terbaik yaitu  $P = 12$ ,  $I = 2$ ,  $D = 6$  [3].

Penelitian lain membahas tentang Implementasi pergerakan robot penjaga gawang KRSBI (Beroda) dengan metode fuzzy PID menghasilkan nilai PID yang terbaik yaitu  $K_p = 2,4$  ,  $K_i = 0,53$  ,  $K_d = 2,7$  dan menghasilkan nilai *risetime* sebesar 3 ms dan *settling time* sebesar 11 ms [4].

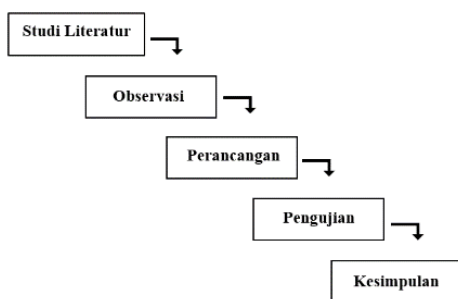
Berdasarkan permasalahan tersebut maka penelitian ini menerapkan *controller* PID (*Proportional, Integral, Derivative*) pada Robot Sepak Bola Beroda. Dimana sistem kontrol ini digunakan agar robot mampu mencari bola dan berjalan kearah bola dengan lebih cepat dan lebih baik.

## 2. METODE PENELITIAN

Prosedur penelitian dilakukan dengan 5 tahapan antara lain: studi literatur, observasi, perancangan, pengujian dan kesimpulan.

### 2.1 Prosedur Penelitian

Prosedur penelitian dapat dilihat pada gambar 1.



Gambar 1. Prosedur Penelitian

#### 2.1.1 Studi Literatur

Pada tahap ini dilakukan pengumpulan referensi sebagai dasar teori yang diambil dari buku-buku literatur, jurnal, artikel-artikel dari internet dan sumber yang lain tentang sistem controller dengan menggunakan metode PID (*Proportional, Integral, Derivative*).

#### 2.1.2 Observasi

Melakukan pengamatan secara langsung tentang permasalahan pada robot sepak bola beroda pada saat mencari bola dan menendang bola kearah gawang.

#### 2.1.3 Perancangan

Pada tahapan ini terdapat dua aktivitas perancangan, yaitu perancangan software dan hardware. Perancangan software dilakukan dengan pembuatan kode program dan penerapan metode PID pada kode program. Perancangan hardware yaitu perancangan elektronika pada robot, pemasangan sensor kamera, pemasangan motor, pemasangan penendang.

#### 2.1.4 Pengujian

Pada tahap ini terdapat beberapa pengujian yang dilakukan pada robot antara lain pengujian *tuning* PID, pengujian penendang, dan pengujian gerakan dasar pada robot.

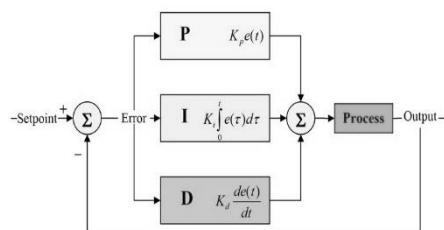
#### 2.1.5 Kesimpulan

Melakukan analisa terhadap hasil pengujian dengan menggunakan metode PID apakah sudah berjalan dengan baik.

## 2.2 Controller PID

*Controller* PID (*Proportional Integral Derivative*), merupakan gabungan dari tiga sistem kendali yang bertujuan untuk mendapatkan keluaran dengan *risetime* yang tinggi dan galat yang kecil. Sistem kendali Proporsional memiliki keunggulan yaitu *risetime* yang cepat tetapi sangat rentan dengan *overshot/undershot*, sistem kendali Integral memiliki keunggulan untuk meredam galat, sedangkan sistem kendali Derivative memiliki keunggulan memperkecil delta *error* atau meredam *overshot/undershot*. PID berdasarkan implementasinya dibedakan menjadi 2 yaitu analog dan digital, PID analog diimplementasikan dengan komponen elektronika yang berupa *resistor, capasitor, dan operational amplifier*, sedangkan PID digital diimplementasikan secara program [5].

Ketika membuat suatu sistem *controller*, tentunya ada suatu hal yang menjadi tujuan. Tujuan dari pengontrolan suatu sistem tersebut biasa disebut dengan *set point*. Sistem akan terus berusaha agar nilai dari *set point* tersebut selalu terpenuhi. Simpangan nilai antara *set point* dengan nilai kenyataan disebut dengan *error*.



Gambar 2. Blok Diagram PID

Dalam membuat suatu sistem *controller* yang ditanamkan pada robot tentunya harus diimplementasikan kedalam suatu sintak program. Suatu rumus nantinya akan sedikit berubah penampilan jika telah diimplementasikan ke dalam program, karena pada pemrograman, hanya diperlukan sifat-sifat dan algoritma dari rumus tersebut. Prancangan sistem PID

ini, merupakan penerapan rumus menjadi suatu sintak program sebagai berikut :

a) Proporsional

Rumus dasar dari *controller* Proporsional adalah :

$$P = K_p e(t) \dots\dots\dots(1)$$

Dengan:

- P : adalah variabel Proporsional
- K<sub>p</sub> : adalah nilai konstanta Proporsional
- e(t) : adalah nilai error yang selalu akan berubah

Nilai error didapat dari:

$$e(t) = Sp - Pv \dots\dots\dots(2)$$

Dengan:

- Sp : adalah *Set pointt*
- Pv : adalah *Process Variable*

b) Integral

Rumus dasar dari *controller* Integral adalah:

$$I = K_i \int_0^t e(t)dt \dots\dots\dots(3)$$

Dengan:

- I : adalah variabel Integral
- K<sub>i</sub> : adalah nilai konstanta Integral
- e(t) : adalah nilai error
- dt : adalah nilai perubahan waktu (*second*)

*Controller* Integral tidak dapat digunakan sendiri, maka dari itu harus digabungkan dengan *controller* Proporsional.

c) *Derivative*

Rumus dasar dari *controller* *Derivative* adalah:

$$D = K_d \frac{de(t)}{dt} \dots\dots\dots(4)$$

Dengan:

- D : adalah variabel *Derivative*
- K<sub>d</sub> : adalah nilai konstanta *Derivative*
- de(t) : adalah nilai perubahan *error*
- dt : adalah nilai perubahan waktu (*second*)

Sama juga halnya dengan *controller* Integral, *controller* *Derivative* juga tidak bisa digunakan sendirian, melainkan juga harus digabungkan dengan *controller* Proporsional. Apabila tidak, maka *controller* *Derivative* tidak akan dapat melakukan aksi apapun, karena tidak ada perubahan *error* [6].

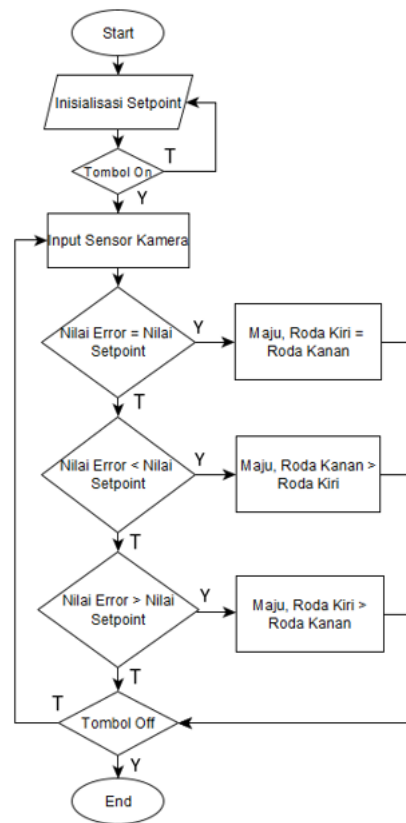
Kemudian variabel Proporsional, Integral, dan Derivative digabungkan menjadi variabel PID yaitu dengan menjumlahkannya.

$$PID = P + I + D \dots\dots\dots(5)$$

Keluaran dari perhitungan *controller* PID ini adalah nilai PWM (*Pulse Width Modulation*) yang mengatur kecepatan putaran motor kanan dan motor kiri pada robot sepak bola beroda.

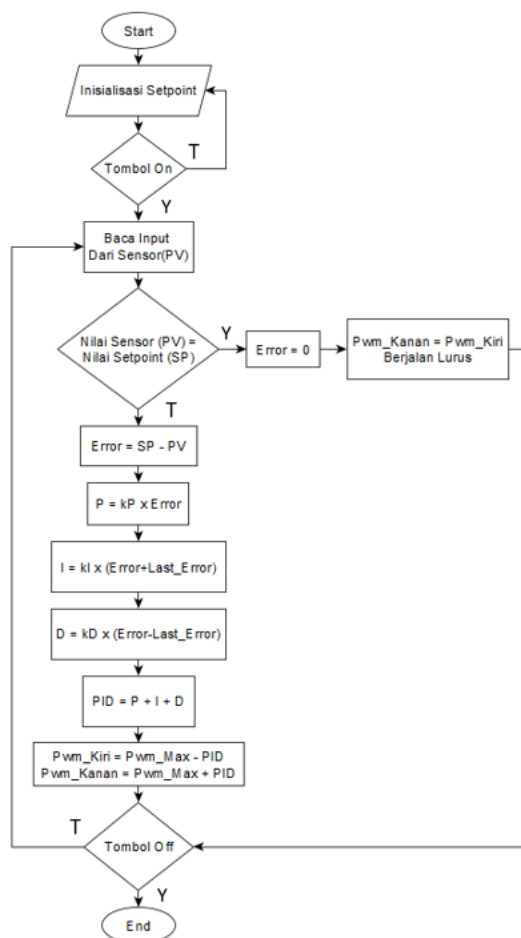
2.3 Perancangan *Controller* PID pada robot

Pada perancangan *controller* PID terdapat 2 *flowchart* antara lain *flowchart* algoritma pergerakan pada robot dan *flowchart* sistem *controller* PID. Seperti yang terlihat pada gambar 3 dan gambar 4.



Gambar 3. *Flowchart* Algoritma Pergerakan pada Robot

Pada gambar 3 menjelaskan tentang *flowchart* algoritma pergerakan pada robot. Jika input pada sensor kamera mendeteksi nilai *Error* sama dengan nilai *Setpoint* maka robot akan melaju dengan kecepatan roda kiri dan kanan sama. Jika nilai *Error* kurang dari nilai *Setpoint* maka robot akan melaju dengan kecepatan roda kanan lebih cepat dibanding dengan roda kiri. Jika nilai *Error* lebih besar dari nilai *Setpoint* maka robot akan melaju dengan kecepatan roda kiri lebih cepat dari pada roda kanan.



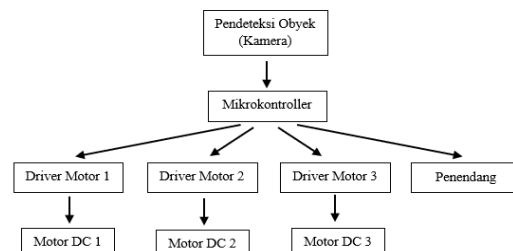
Gambar 4. Flowchart Sistem Controller PID pada Robot

Pada gambar 4 menjelaskan tentang *flowchart* sistem *controller* PID pada robot. Apabila nilai sensor (PV) sama dengan nilai *Setpoint* (SP) maka nilai *Error* sama dengan 0 dan nilai Pwm\_kanan akan sama besar dengan nilai Pwm\_Kiri dengan begitu robot akan melaju berjalan lurus. Jika nilai sensor (PV) tidak sama dengan nilai *Setpoint* (SP) maka akan melakukan penghitungan PID. Nilai *Error* diperoleh dari pengurangan nilai *Setpoint* (SP) dengan nilai pembacaan sensor kamera (PV). Kemudian nilai *Error* tersebut yang akan menjadi acuan dalam penghitungan PID. Selanjutnya penghitungan nilai P atau Proporsional yaitu nilai  $K_p$  dikalikan dengan nilai *Error*. Penghitungan nilai I atau Integral yaitu nilai  $K_i$  dikalikan dengan hasil penjumlahan antara *Error* dan *Last\_error*. Penghitungan nilai D atau Derivative yaitu nilai  $K_d$  dikalikan dengan hasil pengurangan antara *Error* dan *Last\_error*. Untuk memperoleh nilai PID yaitu dengan menjumlahkan antara P, I dan D. Untuk mengatur kecepatan motor kiri, kecepatan maksimum pwm motor kiri dikurangi dengan nilai PID sehingga nilai output pwm akan berkurang. Sedangkan untuk

mengatur kecepatan motor kanan, kecepatan maksimum pwm motor kanan ditambah dengan nilai PID sehingga nilai output pwm akan bertambah. Dengan kondisi demikian robot akan berusaha bergerak agar nilai *Error* sama dengan 0.

#### 2.4 Perancangan *Hardware* pada robot

Secara umum perancangan *hardware* atau perangkat keras yaitu unit masukan dan keluaran pada *microcontroller*. Perancangan *hardware* dapat dilihat pada gambar 5.



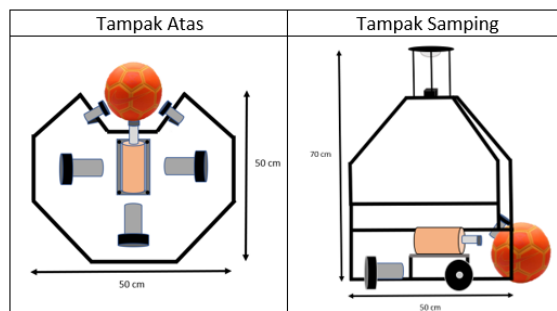
Gambar 5. Diagram Perangkat Keras pada Robot

Penjelasan dari diagram perangkat keras dapat dilihat pada tabel 1.

Tabel 1. Penjelasan dari diagram perangkat keras

No	Nama	Fungsi dan Kegunaan
1	Pendeteksi Obyek (Kamera)	Sebagai pendeteksi Obyek Bola dan sebagai inputan dari <i>microcontroller</i> .
2	<i>Microcontroller</i>	Sebagai pengontrol rangkaian elektronik.
3	<i>Driver Motor</i>	Sebagai pengontrol arah putaran dan kecepatan motor DC.
4	Penendang	Sebagai penendang pada robot.

Desain rancangan perangkat keras pada robot sepak bola beroda dapat dilihat pada gambar 6.



Gambar 6. Desain Rancangan Robot

### 3. HASIL DAN PEMBAHASAN

Pada bagian ini menjelaskan tentang hasil dan pembahasan pada pengujian robot sepak bola beroda.

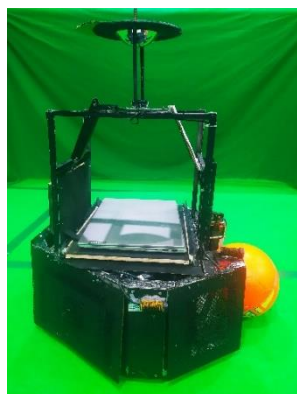
#### 3.1 Bentuk Fisik Robot Sepak Bola Beroda

Bentuk fisik robot ini adalah hasil implementasi dari perancangan perangkat keras dan desain robot. Berikut adalah hasil bentuk fisik robot sepak bola beroda.



Gambar 7. Robot Tampak Depan

Pada gambar 7 robot terlihat tampak depan. Pada bagian depan terdapat penendang dan penggiring bola.



Gambar 8. Robot Tampak Samping

Pada gambar 8 robot terlihat tampak samping. Pada bagian samping terdapat laptop.

Robot terlihat tampak belakang ditunjukkan oleh gambar 9. Pada bagian belakang terdapat tombol untuk menyalakan robot.



Gambar 9. Robot Tampak Belakang

#### 3.2 Pengujian *Tunning* PID

Pengujian *tunning* PID dengan merubah parameter kP, kI, kD sampai menemukan hasil yang terbaik. Pengujian dilakukan dengan menempatkan objek disebelah kanan robot yang berjarak 3 meter.

Pengujian ke	Parameter			Jarak Objek (m)	Waktu (detik)	Hasil
	kP	kI	kD			
1	10	0	0	3	04.44	Berhasil
2	0	10	0	3	05.65	Berhasil
3	0	0	10	3	04.88	Berhasil
4	10	10	0	3	05.60	Berhasil
5	10	0	10	3	-	Gagal
6	0	10	10	3	-	Gagal
7	12	0	0	3	04.30	Berhasil
8	15	0	0	3	05.72	Berhasil
9	17	0	0	3	-	Gagal
10	12	5	0	3	06.10	Berhasil
11	12	7	0	3	06.48	Berhasil
12	12	10	0	3	-	Gagal
13	12	5	5	3	-	Gagal
14	5	5	5	3	04.05	Berhasil
15	5	5	7	3	03.44	Berhasil
16	5	5	10	3	03.15	Berhasil
17	5	5	12	3	04.71	Berhasil

Hasil pengujian dapat dilihat pada tabel 2.

Tabel 2. Pengujian *tunning* PID

Pada tabel 2 menunjukkan bahwa dari 17 aktivitas pengujian, keberhasilan robot mencari bola sebanyak 12 kali sedangkan gagal sebanyak 5 kali. Waktu tercepat robot dalam mendapatkan bola yaitu 03.15 detik dengan menggunakan nilai kP = 5, kI = 5, kD = 10. Dari pengujian ini dapat diketahui bahwa hasil *tunning* PID yang terbaik yaitu pengujian yang ke 16.

### 3.3 Pengujian Penendang

Pengujian penendang dilakukan dengan cara bola diletakkan didepan robot dengan jarak 50 cm. Robot diuji untuk menangkap dan menggiring bola setelah itu menendang bola. Hasil pengujian penendang dapat dilihat pada tabel 3.

Tabel 3. Pengujian Penendang pada robot

Pengujian ke	Jarak Bola dengan Robot (m)	Dribble	Jarak Lurus (m)	Jarak Samping (m)	Hasil
1	0.5	Berhasil	4	0.7	Berhasil
2	0.5	Berhasil	4	0.6	Berhasil
3	0.5	Berhasil	4	0.55	Berhasil
4	0.5	Gagal	2	0.8	Berhasil

Pada tabel 3 menunjukkan terdapat 4 aktivitas pengujian. Berdasarkan empat pengujian diketahui bahwa robot mampu menangkap bola sebanyak 3 kali dengan sempurna dan 1 kali gagal. Sedangkan untuk penendang, robot telah berhasil menendang dalam seluruh aktivitas pengujian yaitu dengan jarak lurus 4 meter dan jarak samping paling sedikit yaitu 55 cm yaitu pada pengujian ke 3.

### 3.4 Pengujian Gerakan Dasar

Pada pengujian gerakan dasar dilakukan dengan cara robot diuji untuk melakukan gerakan dasar antara lain maju, mundur, kanan dan kiri dimana robot melakukan gerakan dasar sambil menggiring bola. Hasil pengujian dapat dilihat pada tabel 4.

Tabel 4. Pengujian gerakan dasar robot

Pengujian ke	Gerakan	Jarak Tempuh (m)	Jarak Error (cm)	Posisi Bola Lepas / Tidak
1	Maju	3	6	Tidak
2	Mundur	3	10	Lepas
3	Kanan	3	18	Tidak
4	Kiri	3	12	Tidak

Pada tabel 4 menunjukkan hasil bahwa dari ke 4 pengujian robot mampu melakukan gerakan dasar dengan jarak tempuh 3 meter. Jarak *error* paling sedikit yaitu pada pengujian ke 1 dengan jarak *error* 6 cm. sedangkan untuk menggiring bola, dari ke 4 pengujian robot mampu menggiring bola sebanyak 3 kali dengan bola tidak terlepas pada robot dan 1 kali bola terlepas pada pengujian yang ke 2.

## 4. SIMPULAN

Pada penelitian ini telah dibuat sistem controller PID (*Proportional, Integral, Derivative*) pada robot sepak bola beroda. Pengujian yang dilakukan antara lain pengujian *tunning* PID yaitu mencari nilai parameter kP, kI, dan kD yang terbaik. Pengujian penendang yaitu dengan cara robot diuji untuk menangkap, menggiring dan menendang bola. Pengujian gerakan dasar yaitu robot diuji untuk melakukan gerakan antara lain maju, mundur, kanan dan kiri sambil menggiring bola. Dari beberapa pengujian yang dilakukan dapat disimpulkan beberapa hal yaitu hasil *tunning* PID yang terbaik dengan nilai kP = 5, kI = 5, kD = 10 dengan waktu 03.15 detik. Pada pengujian penendang robot berhasil menendang bola. Pengujian gerakan dasar pada robot sudah mampu melakukan gerakan dengan baik, dari keempat pengujian robot mengalami sekali kegagalan dalam menggiring bola.

## 5. SARAN

Untuk pengembangan lebih lanjut pada penelitian ini, maka penulis memberikan beberapa saran diantaranya pada pemberian nilai-nilai parameter masih secara manual sehingga diperlukan banyak percobaan, untuk mendapatkan nilai secara otomatis yaitu menggunakan metode *tunning* Ziegler Nichols. Selain itu juga dapat mengganti sensor kamera yang akurasi lebih baik, agar robot mampu mendeteksi objek lebih akurat.

## DAFTAR PUSTAKA

- [1] Sahertian, J., Verlianto, M.M. 2018. Sistem Pendeteksian Bola Pada Robot Sepakbola Beroda Berbasis Filter Warna. *SENATIK 2018*. Program Studi Teknik Informatika. Universitas PGRI Madiun
- [2] Sya'ban, D.P., Sujono. 2018. Sistem Kendali PID untuk Pencarian Arah Gawang Lawan pada Robot Sepak Bola Beroda. *Jurnal MAESTRO*. Vol.01, No.02.
- [3] Budianto, E.A., Jasmir, Kisbianty, D. 2019. Perancangan Navigasi pada Robot Sepak Bola Beroda menggunakan Metode PID (Proposional, Integer, Delivatif). *Jurnal Ilmiah Mahasiswa Teknik Informatika*. Vol.1, No.1.
- [4] Adi, C.M. 2020. *Implementasi Pergerakan Robot Penjaga Gawang KRSBI (Beroda) dengan Metode Fuzzy PID*. Program Studi Teknik Elektro. Universitas Jember.
- [5] Mas'ud, M. I., Hafidz, M., Harianto, & Wibowo, M. C. 2012. Kendali PID dan Logika Fuzzy Untuk Optimalisasi. *Journal of Control and Network Systems*, 2-3.
- [6] Mas Pratama, I. P., Suwedan, I. N., & Swamardika, I. B. 2013. Sistem Kontrol Pergerakan Pada Robot Line Follower Berbasis Hybrid PID-Fuzzy Logic. *PROSIDING CSGTEIS 2013*, 160-161.