

Robot *Line Follower* Berbasis Kendali Proportional-Integral-Derivative (PID) Untuk Lintasan Dengan Sudut Ekstrim

Koko Joni¹, Miftachul Ulum², Zainal Abidin³

^{1,2,3}Teknik Elektro, Universitas Trunojoyo Madura

^{1,2,3}Jl. Raya Telang PO BOX 2 Kamal, Madura

Email korespondensi : kokojoni@gmail.com

Dikirim 31 Oktober 2016, Direvisi 12 November, Diterima 16 November 2016

Abstrak – Robot *line follower* merupakan robot otonom yang dapat mengikuti jalur. Jalurnya bisa berupa garis berwarna hitam diatas permukaan putih atau sebaliknya. Selama ini kebanyakan peneliti merancang *robot line follower* kendali PID (*Proportional-Integral-Derivative*) untuk lintasan dengan tikungan yang relatif tumpul, namun pada penelitian ini, sistem kendali PID diterapkan pada robot *line follower* untuk menelusuri tiga jenis lintasan dengan sudut yang ekstrim. Robot ini menggunakan 8 buah sensor dengan asumsi 1 atau 2 sensor menyentuh garis lintasan dengan tebal 2 cm. Pada percobaan *max. speed* PWM 100 didapat *tuning* PID terbaik yaitu $K_p = 3$, $K_i = 3$ dan $K_d = 0$, menghasilkan waktu mencapai *finish* untuk lapangan A, B dan C masing-masing sebesar 3,59 s, 3,10 s dan 3,69 s. Sedangkan untuk *max. speed* PWM 255 didapat *tuning* PID terbaik yaitu $K_p = 5$, $K_i = 5$ dan $K_d = 5$, menghasilkan waktu mencapai *finish* untuk lapangan A, B dan C masing-masing sebesar 2,67 s, 2,46 s dan 2,78 s. Kecepatan maksimum sangat berpengaruh terhadap hasil *tuning* PID untuk lintasan ekstrim, hal ini dikarenakan ketika kecepatan tinggi membutuhkan sistem PID dengan respon *time* yang relatif cepat dan stabil. Sehingga pengaturan PID untuk kecepatan PWM tertentu tidak bisa diterapkan untuk kecepatan PWM yang lain.

Kata kunci – Robot, *Line*, PID

Abstract - Line follower robot is an autonomous robot that can follow the paths. The trail can be a black line on a white surface, or vice versa. So far, most researchers designed a line follower robot PID (Proportional-Integral-Derivative) control to the track with a relatively blunt corner, but in this study, PID control system applied to the line follower robot to discover three types of trajectories with extreme angles. This robot uses eight sensors assuming 1 or 2 sensor touch that line the track with 2 cm thick. In the experiment *max. speed* PWM 100 *tuning* PID that is best obtained $K_p = 3$, $K_i = 3$ and $K_d = 0$, generate the time to reach the finish to field A, B and C respectively by 3.59 s, 3.10 s and 3.69 s. As for the *max. 255* PWM speed that is best obtained *tuning* PID $K_p = 5$, $K_i = 5$ and a $K_d = 5$, resulting in time to the finish line to field A, B and C respectively of 2.67 s, 2.46 s and 2.78 s. Maximum speed heavily influence the PID tuning for extreme track, this is because when the high speed require the PID system response time is relatively fast and stable. So PID for some PWM speed cannot be implemented to other PWM speed.

Keywords – Robot, Line, PID

I. PENDAHULUAN

Dunia robotika di Indonesia sudah mengalami perkembangan yang sangat pesat dibanding beberapa tahun yang lalu. Kompetisi mengenai robot hampir tiap tahun diadakan baik oleh Kemenristek DIKTI, sekolah-sekolah ataupun perguruan tinggi negeri maupun swasta. Dengan adanya berbagai kompetisi ini membuat penelitian dibidang robotika juga semakin banyak. Perlombaan yang sering diadakan adalah robot pengikut garis (*robot line follower*), robot ini dirancang agar bisa mengikuti lintasan yang telah dibuat.

Salah satu jenis robot dengan kemampuan istimewa yang belakangan ini banyak menarik minat para ahli untuk dikembangkan adalah *mobile robot* [1]. Robot *line follower* merupakan salah satu bentuk *mobile robot* yang bergerak otonom yang banyak dirancang baik untuk penelitian, industri, maupun kompetisi robot [2]. Sesuai dengan namanya, misi yang harus dilakukan oleh suatu robot pengikut garis adalah mengikuti garis pandu. Dalam perancangan dan implementasi suatu robot bergerak otomatis, banyak masalah-masalah yang harus dipecahkan. Masalah-masalah itu adalah sistem penglihatan robot, arsitektur perangkat keras yang meliputi perangkat

elektronik dan mekanik, organisasi perangkat lunak untuk basis pengetahuan dan pengendalian robot [3].

Sistem kendali yang sering digunakan untuk robot *line follower* adalah *Proportional-Integral-Derivative* (PID), dengan cara memperbaiki nilai error dari *feedback* hingga menjadi nol [4], kondisi *set point* akan tercapai dari perpaduan nilai PID yang tepat. Nilai set point dan error berasal dari pembacaan posisi robot *line follower* terhadap garis. Dengan menentukan *set point* robot *line follower* akan bermanuver memposisikan robot berada di tengah jalur.

Telah banyak penelitian maupun artikel yang membahas tentang robot *line follower*, dan berikut sumber acuan yang mendukung penelitian ini. Penelitian pertama tentang pergerakan robot *line follower* dengan 7 sensor menggunakan *hybrid PID – Fuzzy* yang menganalisis perbandingan respon robot menggunakan Kendali PID dan *hybrid PID – Fuzzy* pada lintasan berbentuk ‘L’ dengan sudut tikungan yang tidak terlalu tajam. Dari hasil penelitian didapat algoritma *hybrid PID – Fuzzy* dengan konstanta yang sesuai untuk robot pada lintasan uji berbentuk ‘L’ [4].

Pada penelitian lainnya mengembangkan tentang robot *line follower* berbasis LEGO dengan 3 buah sensor menggunakan pengendali PID untuk menyelesaikan lintasan berbentuk lingkaran seperti kapsul dengan sudut tikungan yang relatif tumpul. Dari hasil penelitian, robot mampu menyelesaikan satu lintasan penuh hanya menggunakan 3 buah sensor warna dengan metode PID [5].

Terdapat pula penelitian tentang memecahkan lintasan lancip dan lengkung pada *robot line follower* menggunakan algoritma *line maze solving*. Dari hasil penelitian tersebut, robot mampu melewati jalur lengkung lebih dari 255° dan jalur lancip kurang dari 45° menggunakan modifikasi dari algoritma *line maze solving* [6].

Pada penelitian sebelumnya yang juga menggunakan sistem kendali PID, keduanya masih terbatas pada lintasan dengan tikungan yang relatif tumpul. Terdapat pemecahan untuk lintasan lancip dan lengkung akan tetapi masih menggunakan algoritma *line maze solving*. Hal tersebut menjadi celah menarik untuk dikembangkan, robot *line follower* dengan kendali PID untuk lintasan dengan sudut yang ekstrim, seperti lintasan berbentuk ‘S’, ‘M’ dan ‘U’ bergelombang.

Pada penelitian ini akan dibahas tentang “Rancang Bangun Robot *Line Follower* Berbasis Kendali *Proportional-Integral-Derivative* (PID) untuk Lintasan dengan Sudut Ekstrim”, dilakukan untuk membuat sebuah robot pengikut garis secara dinamis dengan respon yang cepat dan akurat pada lintasan yang ekstrim berbasis PID. Fokus penelitiannya adalah analisa sistem PID terhadap kestabilan

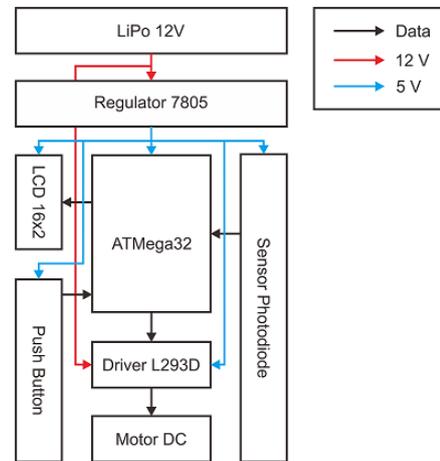
manuver robot dengan menggunakan 8 buah sensor dengan asumsi terdapat 2 sensor dalam garis.

II. METODOLOGI PENELITIAN

Metode penelitian meliputi perancangan perangkat keras, kendali PID, dan metode pengujian.

A. Perancangan Perangkat Keras

Secara garis besar skema robot *line follower* mikro seperti pada Gambar 1.



Gambar 1. Skema Robot Line Follower

Pada Gambar 1, dapat diketahui bahwa Sensor *Photodiode* akan mengirimkan data ADC yang kemudian diolah oleh ATmega 32 untuk menentukan posisi robot terhadap garis lalu menampilkannya pada LCD 16x2 dan menggerakkan motor DC agar posisi robot tepat ditegah garis.

Driver L293D digunakan sebagai pengontrol motor DC dengan inputan sinyal *direction* dan PWM dari mikro ATmega16. Sinyal *direction* digunakan untuk mengontrol arah putaran motor DC sedangkan PWM digunakan untuk mengontrol kecepatan putar motor DC dengan level 0 - 255.

LCD 16x2 digunakan untuk menampilkan beberapa data dari robot, seperti nilai ADC, posisi robot berdasarkan garis, kecepatan motor DC dan setting nilai PID. *Push Button* digunakan sebagai masukan untuk mengganti mode pada robot, misalnya untuk mode scan, mode start dan mode setting nilai PID.



Gambar 2. Tampilan Robot Line Follower



Gambar 3. Susunan Sensor Photodiode

Robot ini menggunakan 8 buah sensor *photodiode* yang disusun secara sejajar dengan jarak antara sensor sebesar 1,5 cm, hal ini memungkinkan dalam satu garis lintasan (tebal garis 2 cm) terdapat 1 atau 2 sensor terdeteksi, sehingga pilihan kondisi *set point* robot menjadi lebih banyak dan bervariasi.

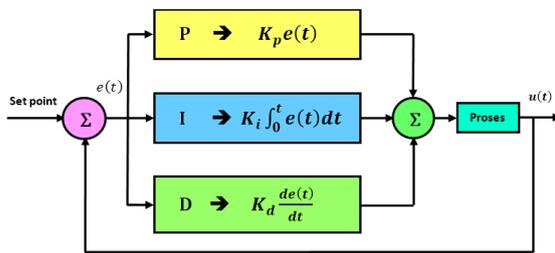
B. Kendali PID

Kontroler PID (dari singkatan bahasa Inggris *Proportional Integral Derivative controller*) merupakan suatu sistem kontrol yang digunakan untuk menentukan kontrol presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik (*feedback*) pada sistem tersebut [7]. Sebuah kontroler PID secara kontinu menghitung nilai error sebagai beda antara *set point* yang diinginkan dan variabel proses terukur. Kontroler mencoba untuk meminimalkan nilai error setiap waktu dengan penyetelan variabel kontrol .

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \quad (1)$$

K_p merupakan kontrol proporsional yang dapat memperbaiki respon transien khususnya *rise time* dan *settling time*. K_i adalah kontrol integral yang memiliki sifat lebih lambat dari kontrol proporsional. K_d adalah kontrol derivatif dimana hanya akan berubah ketika ada perubahan error, sehingga kontrol derivatif harus digunakan dengan kontrol lainnya[8].

Berikut merupakan blok diagram kontroler PID



Gambar 4. Blok Diagram Kontroler PID [8]

Apabila persamaan diatas diimplementasikan dalam bentuk program menjadi:

```

error = SP - PV;
Prop = Cp * error;
Integ = ((error+last_error) * (Ci/10)) * Ct;
Derif = (error-last_error) * (Cd/Ct);
last_error = error;
MV = Prop + Integ + Derif;

ka = ((MAXPWM-MINPWM)/2) + MV;
ki = ((MAXPWM-MINPWM)/2) - MV;
if (ki > MAXPWM) ki = MAXPWM;
if (ka > MAXPWM) ka = MAXPWM;

if (error == 0)      {maju(MAXPWM, MAXPWM);}
else if (ka<0)      {kanan(ki, 0-ka);}
    
```

```

else if (ki<0)      {kiri(0-ki, ka);}
else                {maju(ki, ka);}
    
```

Dari program diatas dapat diketahui hasil dari perhitungan PID (MV) digunakan untuk menentukan kecepatan dari motor DC kanan dan kiri, dengan asumsi jika error = 0 maka robot akan melaju dengan kecepatan penuh, sedangkan jika hasil dari perhitungan PWM (ka dan ki) terdapat nilai dibawah 0 itu digunakan untuk untuk manuver belok dengan kecepatan belok sesuai dengan hasil perhitungan PID.

Selain *tuning* konstanta P, I dan D, penentuan *set point error* yang tepat juga diperlukan agar respon dari robot cepat ketika mendapatkan gangguan (tikungan) dan kembali ke keadaan *steady state*, berikut merupakan program untuk menentukan *set point error* pada robot dengan asumsi angka '1' adalah deteksi garis dan angka '0' untuk deteksi *background*.

```

case 0b00000001: PV=28; break;
case 0b00000011: PV=21; break;
case 0b00000010: PV=15; break;
case 0b00000110: PV=10; break;
case 0b00000100: PV=6; break;
case 0b00001100: PV=3; break;
case 0b00001000: PV=1; break;
case 0b00011000: PV=0; break;
case 0b00010000: PV=-1; break;
case 0b00110000: PV=-3; break;
case 0b00100000: PV=-6; break;
case 0b01100000: PV=-10; break;
case 0b01000000: PV=-15; break;
case 0b11000000: PV=-21; break;
case 0b10000000: PV=-28; break;

case 0b00001111: PV=35; break;
case 0b00011111: PV=35; break;
case 0b00111111: PV=35; break;
case 0b11111100: PV=-35; break;
case 0b11111000: PV=-35; break;
case 0b11110000: PV=-35; break;

case 0b01111110: PV=0; break;
case 0b00000000: if(PV>0) PV=35;
                  else PV=-35; break;
    
```

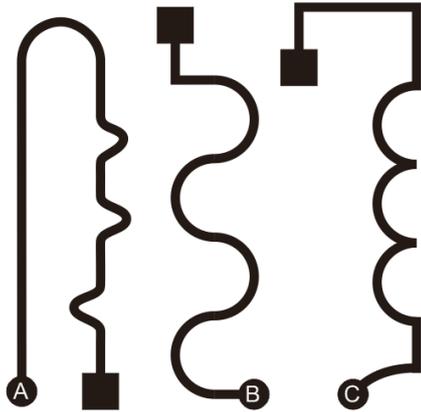
'PV' merupakan *set point* untuk error, umumnya ketika robot hampir mencapai *steady state* nilai 'PV' menjadi lebih kecil, itu digunakan agar tidak terjadi osilasi dan manuver robot lebih halus. Nilai 'PV' kemudian akan diolah dengan konstanta PID untuk mendapatkan kecepatan putaran motor yang sebenarnya.

C. Metode Pengujian

Pada pengujian, robot *line follower* yang sudah siap akan dihadapkan pada tiga macam lintasan berbentuk 'U' bergelombang, 'S' dan 'M', seperti pada Gambar 6, dengan lebar lintasan 2 cm dan warna garis hitam di atas *background* putih.

Sistem tuning PID yang dilakukan adalah *trial and error* untuk mendapatkan kondisi stabil robot sehingga dapat mencapai titik *finish* dengan cepat. *Tuning* PID dianggap baik jika 3 atau 2 dari 3 percobaan menghasilkan waktu mencapai *finish* tercepat dengan pergerakan robot yang halus. Dalam pengujian juga dilakukan perubahan kecepatan maksimum putaran motor DC dengan variasi PWM

100 dan PWM 255, hal ini dilakukan untuk mengetahui apakah perubahan kecepatan mempengaruhi terhadap kestabilan manuver robot dan *tuning* PID yang telah dilakukan.



Gambar 5. Lintasan Pengujian

III. HASIL PENELITIAN

Pengujian hasil perbandingan *tuning* nilai PID dan maksimum kecepatan terhadap waktu tempuh robot *line follower* seperti yang ditunjukkan pada Tabel 1 dan Tabel 2.

Tabel 1. Hasil Pengujian Untuk Kecepatan Maksimum PWM 100

Maksimum Kecepatan PWM 100					
Tuning			Lintasan		
P	I	D	A	B	C
1	0	0	Gagal	Gagal	Gagal
3	0	0	3,64 s	3,02 s	3,74 s
3	3	0	3,59 s	3,10 s	3,69 s
3	3	1	3,82 s	3,16 s	3,74 s
3	3	3	3,71 s	3,12 s	3,94 s

Tabel 2. Hasil Pengujian Untuk Kecepatan Maksimum PWM 255

Maksimum Kecepatan PWM 255					
Tuning			Lintasan		
P	I	D	A	B	C
3	3	0	Gagal	Gagal	Gagal
5	3	1	2,82 s	2,60 s	2,99 s
10	3	1	3,74 s	3,93 s	3,37 s
5	5	1	3,07 s	2,52 s	2,97 s
5	5	5	2,67 s	2,46 s	2,78 s

Waktu tercepat mencapai finish

Data pada Tabel 1 dan Tabel 2 hanya sebagai sampel yang mewakili dari hasil percobaan secara keseluruhan, setiap percobaan dipastikan menggunakan perlakuan yang sama untuk mendapatkan data yang akurat, seperti pengecekan daya dan kondisi robot secara berkala ketika pengujian serta posisi *start* dan *finish* yang sama pada setiap pengujian.

IV. PEMBAHASAN

Dari hasil pengujian pada Tabel 1 dan Tabel 2 dapat diketahui bahwa untuk pengujian pada maksimum kecepatan PWM 100 didapat *tuning* PID dengan respon sistem tercepat yaitu $K_p = 3$, $K_i = 3$ dan $K_d = 0$, menghasilkan waktu mencapai *finish* untuk lapangan A, B dan C masing-masing sebesar 3,59 s, 3,10 s dan 3,69 s. Robot sempat gagal menyelesaikan lintasan ketika nilai K_p terlalu kecil, ini dikarenakan sistem PID tidak mampu menghilangkan *error* dengan *gain* yang kecil sehingga robot tidak mampu mencapai posisi *steady state*.

Untuk percobaan kedua dengan maksimum kecepatan PWM 255 didapat *tuning* PID dengan respon sistem tercepat yaitu $K_p = 5$, $K_i = 5$ dan $K_d = 5$, menghasilkan waktu mencapai *finish* untuk lapangan A, B dan C masing-masing sebesar 2,67 s, 2,46 s dan 2,78 s. Robot sempat gagal menyelesaikan lintasan ketika nilai *tuning* PID terlalu kecil, robot juga lama mencapai *finish* ketika nilai PID terlalu besar, ini dikarenakan ketika nilai PID tidak sesuai, maka pergerakan robot akan menjadi kasar dan berisilasi sehingga membutuhkan waktu yang relatif lama untuk mencapai *finish*.

Dari dua percobaan diatas, ternyata kecepatan maksimum sangat berpengaruh terhadap hasil *tuning* PID, hal ini dibuktikan ketika hasil *tuning* PID terbaik dari percobaan pertama diterapkan pada percobaan kedua dengan maksimum kecepatan PWM 255, robot menjadi tidak stabil bahkan gagal untuk mencapai *finish*. Hal ini dikarenakan ketika kecepatan tinggi membutuhkan sistem PID dengan *respon time* yang relatif cepat dan stabil. Sehingga pengaturan PID untuk PWM tertentu tidak bisa diterapkan untuk pengaturan PWM yang lain.

V. PENUTUP

A. Kesimpulan

Robot *line follower* yang dirancang dengan *tuning* PID yang tepat dapat menyelesaikan lintasan dengan sudut ekstrim, untuk maksimum kecepatan PWM 100 didapat *tuning* PID dengan respon sistem tercepat yaitu $K_p = 3$, $K_i = 3$ dan $K_d = 0$, menghasilkan waktu mencapai *finish* untuk lapangan A, B dan C masing-masing sebesar 3,59 s, 3,10 s dan 3,69 s. Sedangkan untuk maksimum kecepatan PWM 255 didapat *tuning* PID dengan respon sistem tercepat yaitu $K_p = 5$, $K_i = 5$ dan $K_d = 5$, menghasilkan waktu mencapai *finish* untuk lapangan A, B dan C masing-masing sebesar 2,67 s, 2,46 s dan 2,78 s. Terakhir, maksimum kecepatan sangat berpengaruh terhadap hasil *tuning* PID, Hal ini dikarenakan ketika kecepatan tinggi dibutuhkan sistem PID dengan *respon time* yang relatif cepat dan stabil.

B. Saran

Selain sistem kontrol PID yang baik, susunan sensor juga berpengaruh pada manuver robot *line follower* pada lintasan dengan sudut yang ekstrim, diperlukan penelitian lebih lanjut tentang susunan sensor terhadap sistem PID yang dibuat.

DAFTAR PUSTAKA

- [1] Aripin M., dkk, "Analisa Algoritma Proporsional Integral Derivatif dan Algoritma Path Planning pada Robot Line Follower", STMIK Cirebon, Jurnal Digit, Vol.6, No.1 Mei 2016.
- [2] Zakki A., Budi Setyawati, "Robot Line Follower Berbasis Mikrokontroler Atmega 16 dengan Menampilkan Status Gerak pada LCD", Universitas Narotama Surabaya, E-Jurnal NARODROID, Vol.1 No.1 Januari 2015.
- [3] Amri Djuril, "Komparasi Rangkaian Sensor Garis Dengan LM 741 Dan TLC 274 Pada Robot Mobil Pengikut Garis (Line Follower) Dengan Menggunakan Mikrokontroler Atmega 8535", Universitas Sriwijaya, Jurnal Rekayasa Sriwijaya No. 1 Vol. 19, Maret 2010.
- [4] N. Gilang, D. Andi, A. Catur, "Implementasi Kendali Logika Fuzzy pada Robot Line Follower", UGM, IJEIS, Vol.4, No.1, April 2014, pp. 45~56.
- [5] Romadhon A., Fuad M., "Perancangan Sistem Kontrol Gerakan Pada Robot Line Tracer", Universitas Trunojoyo Madura, Jurnal Ilmiah Mikrotek Vol. 1, No.1 2013.
- [6] Yanto F., Welly Irma, "Analisa dan Perbaikan Algoritma Line Maze Solving Untuk Jalur Loop, Lancip, dan Lengkung pada Robot Line Follower (LFR)", UIN Sultan Syarif Kasim Riau, Jurnal CoreIT, Vol.1, No.2, Desember 2015.
- [7] Prinandika, Bayu E., "Sistem Pengaturan Kecepatan Motor pada Robot Line Follower Berbeban Menggunakan Controller PID", Tugas Akhir .
- [8] Putu I., dkk, "Sistem Kontrol Pergerakan pada Robot Line Follower Berbasis Hybrid PID-Fuzzy Logic", Universitas Udayana, Prosiding Conference on Smart-Green in Electrical and Information Systems, November 2013.