



Implementasi Robot *Line Follower* Sebagai Pengantar Makanan / Minuman Di Restoran Modern

Ariski Sapni Putra¹, Tresna Dewi², Pola Risma³

¹Teknik Elektro, Politeknik Negeri Sriwijaya

ariskspaniputra08@gmail.com, tresna_dewi@polsri.ac.id, polarisma@polsri.ac.id

Abstract

This report discusses the implementation of Line Follower Robots to deliver food or drinks in modern restaurants. This research aims to design and implement a robot that can follow lines and deliver orders automatically to customer tables. This robot uses line sensors to detect trajectories, ultrasonic sensors to avoid obstacles, and color sensors to detect tables. The control system on this robot uses an ESP32 microcontroller. The research results show that the robot can function well in delivering food to two different tables and returning to the starting point after the order is taken by the customer. However, robots still require improvements to sensors and programming to increase their performance. This research is expected to contribute to the development of automation technology in the food service industry.

Key words : Line Follower Robot, ESP32, line sensor, color sensor, ultrasonic sensor, PID controller, restaurant automation.

Abstrak

Laporan ini membahas implementasi Robot Line Follower untuk mengantarkan makanan atau minuman di restoran modern. Penelitian ini bertujuan untuk merancang dan mengimplementasikan robot yang dapat mengikuti garis dan mengantarkan pesanan secara otomatis ke meja pelanggan. Robot ini menggunakan sensor garis untuk mendeteksi lintasan, sensor ultrasonik untuk menghindari rintangan, dan sensor warna untuk mendeteksi meja. Sistem kontrol pada robot ini menggunakan mikrokontroler ESP32. Hasil penelitian menunjukkan bahwa robot dapat berfungsi dengan baik dalam mengantarkan makanan ke dua meja yang berbeda dan kembali ke titik awal setelah pesanan diambil oleh pelanggan. Namun, robot masih memerlukan perbaikan pada sensor dan pemrograman untuk meningkatkan kinerjanya. Penelitian ini diharapkan dapat berkontribusi pada pengembangan teknologi otomasi di industri layanan makanan.

Kata kunci : Robot Line Follower, ESP32, sensor garis, sensor warna, sensor ultrasonik, PID controller, otomasi restoran.

Diterima Redaksi : 15-05-2024 | Selesai Revisi : 28-06-2024 | Diterbitkan Online : 30-06-2024

1. Pendahuluan

Kebutuhan akan layanan pengiriman makanan meningkat karena gaya hidup modern yang sibuk dan mobilitas yang tinggi. Banyak orang, terutama di kota-kota, membutuhkan pengiriman makanan yang handal, cepat, dan efisien. Namun, masalah logistik seperti lalu lintas dan kekurangan sumber daya manusia seringkali memperlambat dan memperlambat pengiriman.

Penggunaan robot sekarang mencakup seluruh sendi atau pekerjaan manusia. Dengan perkembangan teknologi dan otomasi industri yang semakin canggih, manusia dapat memenuhi kebutuhan hidupnya dengan cepat, tepat, dan efisien. Oleh karena itu, di masa mendatang, teknologi robot akan membantu dan mempermudah pekerjaan manusia.

Salah satu robot yang dapat membantu pegawai restoran modern adalah robot pengikut garis, yang merupakan jenis robot bergerak atau robot mobile yang bertugas mendeteksi dan mengikuti garis pandu.

Untuk mengirimkan makanan, robot dapat meningkatkan efisiensi pengiriman dan mengurangi waktu dan biaya. Salah satu ide yang menarik adalah robot pengikuti jalur, yang dapat secara otomatis mengikuti jalur tertentu dan mengirimkan makanan dari A ke B tanpa bantuan manusia.

2. Metode Penelitian

Penelitian menggunakan metode eksperimental dan metode *trial and error*. Implementasi Robot Pengikuti Jalur untuk Pengiriman Makanan dan Minuman di Restoran Modern menggunakan metode *PID*. *PID*

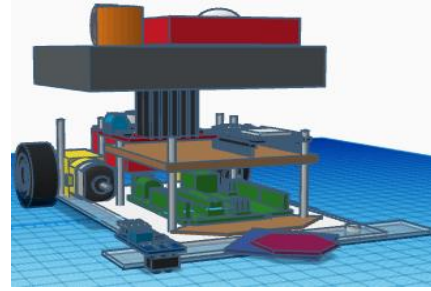
adalah sistem kontrol yang digunakan untuk mengukur gerak presisi sistem instrumentasi dengan adanya umpan balik (*feedback*). Nilai kesalahan dihitung secara konsisten oleh kontroler *PID* sebagai perbedaan antara *set point* yang diinginkan dan variabel proses yang diukur.

Kontroler berusaha setiap saat untuk meminimalkan nilai kesalahan dengan menyetel variabel kontrol. *PID* menggabungkan tiga elemen penting. Pada *Robot Line Follower*, *Proportional (P)* memberikan output kontrol yang *proporsional* terhadap selisih antara nilai *setpoint*, atau nilai yang diinginkan, dan nilai aktual variabel proses. Selanjutnya, *Integral (I)* mengumpulkan selisih antara nilai *setpoint* dan nilai aktual dari waktu ke waktu, yang membantu menghilangkan kesalahan statis yang disebabkan oleh selisih ini. Terakhir, turunan variabel proses digunakan oleh *derivative (D)* untuk merespon perubahan nilai variabel proses yang cepat. Ini membantu mengantisipasi perubahan cepat dan memperbaiki *overshoot*.

PID bekerja dengan menggabungkan ketiga komponen ini untuk menghasilkan sinyal kontrol output yang sesuai dengan tujuan untuk menjaga variabel proses sesuai dengan *setpoint*. Parameter *P*, *I*, dan *D* biasanya disesuaikan dan disetel secara menyeluruh dan menyeluruh untuk mendapatkan kinerja kontrol yang optimal sesuai dengan karakter sistem yang diharapkan.

2.1 Desain Mekanik

Desain mekanik dikembangkan dalam perancangan mekanik ini, prinsip-prinsip fisika, matematika, dan ilmu material digunakan untuk membuat solusi praktis yang dapat digunakan dalam penggunaan Robot Pengikuti Jalur sebagai Penghantar Makanan dan Minuman di Restoran Modern. Desain Robot *Line Follower* untuk penghantar makanan dan minuman di restoran modern berukuran 40x40 cm. Terdapat tiga sensor dalam perancangan ini: sensor garis, sensor *ultrasonik*, dan sensor warna. Semua sensor ini berfungsi sebagai pengirim sinyal ke mikrokontroler untuk mendeteksi gerak robot. Robot digerakkan oleh motor dan baterai 12V sebagai sumber daya.



Gambar 1. 3D Desain Robot Keseluruhan

Robot ini memiliki 2 alas pertama tampak bawah yang terdiri dari sensor, motor, driver motor, dan baterai. Mikrokontroler ESP 32 berfungsi sebagai otak dari robot line follower ini dan mengatur semua komponen ini. Lalu alas tampak atas merupakan desain robot yang berfungsi sebagai nampan untuk robot line follower ini. Selain itu, ada tombol *ON/OFF*, tombol meja, dan LCD. Mikrokontroler ESP32 dapat mengaturnya.

2.2 Tahap Perancangan Sistem

Robot *Line Follower* ini dirancang untuk mengantarkan makanan dan minuman. Pada bagian input, terdapat tombol tekan untuk mengontrol fungsi *ON/OFF*. Sensor garis digunakan untuk mendeteksi jalur yang akan dilalui oleh robot, sementara sensor warna membedakan meja tujuan pengantaran. Selain itu, sensor *ultrasonik* mendeteksi keberadaan objek, seperti piring di nampan, untuk menentukan apakah pelanggan telah mengambil makanan. Pada blok proses, sistem kontrol menggunakan Node MCU ESP32 yang telah disesuaikan dengan modul yang digunakan. Pada bagian output, Node MCU ESP32 berperan sebagai pusat kontrol, sementara Motor *Driver BTS7960*, Motor *Power Window*, dan speaker bertindak sebagai perangkat output.



Gambar 2. Diagram Blok

3. Hasil

Perangkat keras yang dirancang untuk gerak robot terdiri dari mikrokontroler ESP32, sensor garis, sensor

warna, sensor ultrasonic, motor power window, driver motor *BTS7690*, *DC step down XL4005*, 1 tombol *ON/OFF*, 2 tombol meja, 1 LCD, 1 *DF Player Mini MP3*, 1 speaker, dan 1 baterai. Mikrokontroler *ESP32* bertindak sebagai pusat pemrosesan dan menjalankan algoritma pemrosesan data untuk seluruh sistem. Sensor garis (Hitam-Putih) mendeteksi jalur robot, sedangkan Sensor Warna (Merah-Biru) mendeteksi meja yang akan dikirim, dan Sensor Ultrasonic mendeteksi objek di depan robot, yang kemudian dianalisis oleh sistem selain itu, Driver *BTS7690* dapat mengatur kecepatan gerak motor, dan *DC Step Down XL4005* berfungsi sebagai penurun tegangan dari baterai untuk komponen yang ada pada *ESP32*, LCD, dan Speaker.



Gambar 3. Robot Line Follower

3.1 Pengujian Gerak Lurus Robot

Pengujian dilakukan untuk mengukur gerak lurus robot dengan mengukur posisi target yang akan dicapai oleh robot, dengan jalur atau jalur yang telah ditentukan. Jarak pergerakan robot diukur dari posisi awal robot, yaitu $M1 = 0$ dan $M2 = 0$, untuk mengetahui besarnya error yang terjadi ketika robot bergerak maju ke jalur track untuk menuju posisi yang ditentukan. Tabel berikut menunjukkan hasil tes:

Per cob aan	Posisi Awal (cm)		Posisi Target (cm)		Posisi Tercapai (cm)		Error (cm)	
	M1	M2	M1	M2	M1	M2	M1	M2
	1	0	0	120	220	90	150	30
2	0	0	120	220	100	170	20	50
3	0	0	120	220	110	196	10	24
4	0	0	120	220	121	220	0	0
Nilai Error Keseluruhan Robot Line Follower							60	144

Hasil pengujian yang dilakukan untuk mengukur ketepatan posisi yang dicapai oleh robot setelah mengikuti jalur yang ditetapkan ditunjukkan dalam Tabel 4.1. Pengujian dilakukan empat kali (Percobaan 1 hingga 4) dengan dua jalur berbeda, yaitu jalur Meja 1 dan jalur Meja 2.

Berikut rumus dasar pengujian :

$$Error (M1) = \text{Posisi Target (M1)} - \text{Posisi Tercapai (M1)}$$

$$Error (M2) = \text{Posisi Target (M2)} - \text{Posisi Tercapai (M2)}$$

Rumus untuk tiap percobaan:

Percobaan 1:

- $Error (M1) = 120 - 90 = 30 \text{ cm}$
- $Error (M2) = 220 - 150 = 70 \text{ cm}$

Percobaan 2:

- $Error (M1) = 120 - 70 = 50 \text{ cm}$
- $Error (M2) = 220 - 170 = 50 \text{ cm}$

Percobaan 3:

- $Error (M1) = 120 - 110 = 10 \text{ cm}$
- $Error (M2) = 220 - 196 = 24 \text{ cm}$

Percobaan 4:

- $Error (M1) = 120 - 120 = 0 \text{ cm}$
- $Error (M2) = 220 - 220 = 0 \text{ cm}$

Total *error* dapat dihitung dengan menjumlahkan *error* pada tiap percobaan untuk masing-masing M1 dan M2.

- $Total Error (M1) = 30 + 50 + 10 + 0 = 60 \text{ cm}$
- $Total Error (M2) = 70 + 50 + 24 + 0 = 144 \text{ cm}$

3.2 Pengujian Gerak Belok Robot

Tujuan dari pengujian sudut belok adalah untuk mengetahui seberapa besar error yang dihasilkan oleh robot ketika berbelok. Pengujian sudut belok dilakukan untuk mengukur seberapa besar *error* yang dihasilkan oleh robot ketika berbelok. Tabel berikut menunjukkan hasil tes :

Tabel 2. Hasil Pengujian Gerak Belok Robot

Percobaan Ke-	Sudut Target	Gerak Robot Saat Berbelok (cm)	Error (cm)
1	90	200	110
2	90	250	160
3	90	190	100
4	90	180	90
5	90	110	20
6	90	90	0
Robot Line Follower			480

Hasil tes sudut belok robot yang dilakukan dengan beberapa titik sudut target dan sudut gerak robot saat berbelok yang sebenarnya, serta kesalahan yang disebabkan oleh perbedaan antara keduanya. Pada percobaan pertama, robot bergerak dengan sudut 200° sehingga menghasilkan *error* 110° . Dalam percobaan kedua, robot menghasilkan *error* terbesar, yaitu 160° dengan sudut target 90° dan sudut gerak 250° . Dalam percobaan ketiga, robot menghasilkan *error* 100° dengan sudut target 90° dan sudut gerak 190° . Dalam percobaan keempat, robot menghasilkan *error* terkecil,

yaitu 20° dengan sudut target 90° dan sudut gerak 180°. Dari rata-rata *error* yang dihasilkan, terlihat bahwa robot *line follower* ini memiliki rata-rata *error* sebesar 480°.

Berikut adalah rumusnya:

$|Error| = |Set Point - Gerak Robot Saat Berbelok|$
 Di mana *Set Point* adalah sudut target Gerak Robot Saat Berbelok adalah sudut yang diukur sebenarnya. Dari tabel kesalahan-kesalahannya adalah:

- Percobaan 1: $|90^\circ - 200^\circ| = 110^\circ$
- Percobaan 2: $|90^\circ - 250^\circ| = 160^\circ$
- Percobaan 3: $|90^\circ - 190^\circ| = 100^\circ$
- Percobaan 4: $|90^\circ - 180^\circ| = 90^\circ$
- Percobaan 5: $|90^\circ - 110^\circ| = 20^\circ$
- Percobaan 6: $|90^\circ - 90^\circ| = 0^\circ$

Total kesalahan kemudian dijumlahkan untuk memberikan total kesalahan dari semua percobaan:

$$Total Error = 110^\circ + 160^\circ + 100^\circ + 90^\circ + 20^\circ + 0^\circ = 480^\circ.$$

Rata-rata kesalahan dapat dihitung dengan membagi total kesalahan dengan jumlah percobaan:

$$Rata - rata Error = \frac{Total Error}{Jumlah Percobaan} = \frac{480^\circ}{6} = 80^\circ$$

Rata-rata kesalahan adalah 480°.

Rumus yang diturunkan dari tabel untuk menghitung kesalahan dan rata-rata kesalahan adalah:

$$|Error| = |Set Point - Gerak Robot Saat Berbelok|$$

$$Total Error = \sum \binom{n}{i} = iError_i$$

$$Total Error = \frac{Total Error}{n}$$

Di mana *n* adalah jumlah percobaan.

3.3 Pengujian Sensor Pada Robot

Pengujian dilakukan untuk mencari nilai error pada sensor dengan memanfaatkan dua sensor garis yang dihubungkan ke mikrokontroler ESP32 untuk membaca garis lintasan, menggunakan sensor garis untuk mengidentifikasi pembacaan garis hitam dengan latar putih, menggunakan sensor warna untuk membedakan meja yang akan dihantar, dan menggunakan sensor ultrasonic untuk mendeteksi objek di depan. Hasil pengujian sensor disajikan di bawah ini dalam Tabel 3.

Tabel 3. Pengujian Sensor pada robot

Percobaan Ke	Sensor	Meja	Posisi Awal Cm	Posisi Target Cm	Posisi Tercapai Cm	Error
1	Garis Kanan	1	0	250	240	10
1	Garis Kiri	1	0	250	245	5
1	Warna	1	0	250	242	8
1	Ultrasonic	1	0	250	238	12
2	Garis Kanan	2	0	250	248	2
2	Garis Kiri	2	0	250	250	0

2	Warna	2	0	250	249	1
2	Ultrasonic	2	0	250	247	3

Berdasarkan Tabel 4.4 Hasil pengujian sensor pada robot yang digunakan menggunakan sensor seperti Garis Kanan, Garis Kiri, Warna, dan *Ultrasonic*, pengujian di lakukan selama dua sesi, setiap sensor diuji dengan posisi awal 0 cm dan posisi target 250 cm. Hasil pengujian menunjukkan posisi robot yang sebenarnya, yang kemudian dibandingkan dengan posisi target untuk menghitung *error*, yaitu jarak antara posisi tercapai dan target berikut merupakan rumus pengujiannya :

Error untuk setiap percobaan :

$$Error \text{ Percobaan} = \text{Posisi Target} - \text{Posisi Tercapai}$$

Untuk tiap sensor pada setiap percobaan , rumor *error*nya, adalah : *Error* Percobaan = Posisi Target – Posisi Tercapai Nilai *Error* Keseluruhan :

Menjumlahkan semua *error* dari tiap percobaan untuk masing-masing sensor dan meja:

$$Total Error \text{ Sensor Meja} = \sum Error \text{ Sensor}$$

Berikut perhitungan *Error* untuk tabel di atas :

Meja 1

- Sensor Garis Kanan (Percobaan 1, Meja 1) :
- *Error* Garis Kanan = 250–240 = 10
- Sensor Garis Kiri (Percobaan 1, Meja 1) :
- *Error* Garis Kiri=250–245=5
- Sensor Warna (Percobaan 1, Meja 1):
- *Error* Warna = 250 – 242 = 8
- Sensor *Ultrasonic* (Percobaan 1, Meja 1) :
- *Error* *Ultrasonic* = 250 – 238 = 12

Meja 2

- Sensor Garis Kanan (Percobaan 2, Meja 2) :
- *Error* Garis Kanan = 250 – 248 = 2
- Sensor Garis Kiri (Percobaan 2, Meja 2) :
- *Error* Garis Kiri = 250–250 = 0
- Sensor Warna (Percobaan 2, Meja 2) :
- *Error* Warna = 250 – 249 = 1
- Sensor *Ultrasonic* (Percobaan 2, Meja 2) :
- *Error* *Ultrasonic* = 250 – 247 = 3

Total *Error*:

Menjumlahkan *Error* untuk setiap sensor di masing-masing meja :

Meja 1 :

- Total *Error* Sensor Garis Kanan, Meja 1 = 10
- Total *Error* Sensor Garis Kiri, Meja 1 = 5
- Total *Error* Sensor Warna, Meja 1 = 8
- Total *Error* Sensor *Ultrasonic*, Meja 1 = 12

Meja 2 :

- Total *Error* Sensor Garis Kanan, Meja 2 = 2
- Total *Error* Sensor Garis Kiri, Meja 2 = 0
- Total *Error* Sensor Warna, Meja 2 = 1
- Total *Error* Sensor *Ultrasonic*, Meja 2 = 3

4. Kesimpulan

Kesimpulan dari implementasi robot *line follower* sebagai penghantar makanan / minuman di restoran modern yaitu dapat mengikuti garis lintasan dengan stabil dan menghantarkan makanan ke meja tujuan dengan baik. Robot terbukti efisien dalam waktu perjalanan, mampu membawa beban hingga 3 kg, dan andal dalam sensor serta motor yang digunakan. Penggunaan *PID controller* meningkatkan stabilitas dan akurasi gerak robot, mengurangi kesalahan navigasi pada robot

5. Daftar Rujukan

- [1] E. Susanto, F. T. Elektro, and U. Telkom, "Prototipe Robot Pelayan Berbasis Line Follower Untuk Menu Untuk Pelanggan Prototype of Waiters Robot Based on Line Follower To Replace Waiters in Restaurant With Function To Writes a Menu," vol. 4, no. 2, pp. 1552–1559, 2017.
- [2] M. A. Khairul, P. Pangaribuan, and R. Nugraha, "Sistem Pengantaran Makanan Menggunakan Augmented Guided Vehicle (AGV) Line Follower Dilengkapi Barcode," *Agustus*, vol. 7, no. 2, p. 3091, 2020.
- [3] S. Anam, M. Iqbal, and I. A. Rozaq, "Prototype Robot Pengantar Pesanan Otomatis Berbasis Arduino," *J. Elektro Kontrol*, vol. 2, no. 2, pp. 18–31, 2022, doi: 10.24176/elkon.v2i2.8612.