

Robot Pengenal Warna dengan Penganturan Kecepatan Menggunakan Metode PID

Andira Mutiara^{1*}, Rahmi Nurhaliza¹, Indra Dwisaputra¹, Yudhi¹

¹Politeknik Manufaktur Negeri Bangka Belitung, Sungailiat

*E-mail : andiramutiara10@gmail.com¹

Received:9 Januari 2025; Received in revised form:10 Januari 2025; Accepted:14 Januari 2025

Abstract

This research focuses on the development of a robot equipped with a color recognition system and speed control using a PID controller. The robot employs a TCS3200 color sensor to detect object colors and determine its movement trajectory. The optimal distance between the sensor and the object was identified as 0.5 cm to ensure maximum accuracy. The PID controller is responsible for regulating the motor speed, ensuring the robot moves according to the specified velocity. At a setpoint of 100, the tuning parameters Kp 1.0, Ki 0.0012, and Kd 0.00045 produced a balance between fast response and stability. With a setpoint of 200, the tuning values Kp 1.0, Ki 0.0002, and Kd 0.0005 maintained both quick response and stability. For a setpoint of 300, the configuration of Kp 1.0, Ki 0.00035, and Kd 0.00070 achieved a balanced response with minimal overshoot. The test results demonstrate that the robot is capable of following color-based paths according to the predetermined speed settings.

Keywords: Color Recognition; PID Controller; Speed Control

Abstrak

Penelitian ini berfokus pada pengembangan robot dengan sistem pengenalan warna dan pengaturan kecepatan menggunakan PID *controller*. Robot ini menggunakan sensor warna TCS3200 untuk mendeteksi warna objek dan menentukan lintasan pergerakannya. Jarak optimal antara sensor dan objek ditemukan pada 0,5 cm untuk mencapai akurasi terbaik. PID *controller* berfungsi untuk mengendalikan kecepatan motor, menjaga pergerakan robot sesuai kecepatan yang telah ditentukan. Pada *setpoint* 100, konfigurasi Kp 1.0, Ki 0.0012, dan Kd 0.00045 menghasilkan keseimbangan antara respon cepat dan stabilitas. Dengan *setpoint* 200, tuning Kp 1.0, Ki 0.0002, dan Kd 0.0005 mempertahankan respon yang cepat dan stabil. Untuk *setpoint* 300, konfigurasi Kp 1.0, Ki 0.00035, dan Kd 0.00070 menghasilkan respon yang seimbang dengan *overshoot* minimal. Hasil pengujian menunjukkan bahwa robot ini mampu mengikuti lintasan berbasis warna sesuai pengaturan kecepatan yang telah ditentukan.

Kata kunci: Pengenalan Warna; PID *Controller*; Pengaturan Kecepatan

1. PENDAHULUAN

Kemampuan robot untuk mendeteksi warna dan menyesuaikan kecepatan gerakannya secara otomatis merupakan topik penting dalam pengembangan teknologi robotika. Robot yang mampu mendeteksi warna dengan baik dan mengatur kecepatan berdasarkan hasil deteksi tersebut memiliki potensi besar untuk diaplikasikan dalam berbagai bidang, seperti industri, otomasi, hingga pengembangan robot penjelajah. Salah satu tantangan utama dalam pengembangan robot pengenal warna adalah bagaimana mengintegrasikan sensor warna dengan sistem pengaturan kecepatan secara

presisi, sehingga robot dapat merespon perubahan warna secara *real-time* dengan gerakan yang sesuai. Hal ini menjadi penting dalam konteks aplikasi seperti transportasi otomatis, robot penjelajah, dan sistem penyortiran otomatis, di mana kemampuan untuk mengenali warna tertentu dan memberikan respon yang tepat sangat diperlukan [1].

Sensor TCS3200, yang merupakan sensor warna yang mampu mendeteksi berbagai jenis warna berdasarkan perubahan frekuensi, telah banyak digunakan dalam penelitian robotika untuk mendukung identifikasi warna [2] [3]. Beberapa penelitian

sebelumnya telah mengeksplorasi penggunaan sensor warna TCS3200 untuk mengidentifikasi objek berdasarkan warna dan mengendalikan pergerakan robot. Dalam pengaplikasiannya, sensor TCS3200 dapat membaca nilai warna dan mengonversinya menjadi nilai digital yang dapat diproses oleh mikrokontroler, sehingga robot dapat memberikan respons yang diharapkan [4]. Penelitian menunjukkan bahwa robot yang dilengkapi dengan sensor TCS3200 mampu memisahkan objek berdasarkan warna dengan tingkat akurasi yang cukup tinggi, mencapai 85% dalam pengujian [5].

Selain sensor warna, aspek penting lain dalam pengembangan robot ini adalah sistem kendali kecepatan. *PID Controller (Proportional-Integral-Derivative)* merupakan salah satu metode kontrol yang banyak digunakan karena kemampuannya dalam memberikan pengaturan kecepatan yang stabil dan responsif terhadap perubahan [6]. Dengan metode PID, robot dapat menyesuaikan kecepatannya secara otomatis berdasarkan data yang diperoleh dari sensor, serta menjaga keseimbangan roda belakang agar dapat bergerak lurus [7] [8]. Hal ini membuat metode PID sangat cocok untuk diaplikasikan pada robot pengenalan warna yang membutuhkan respons cepat terhadap perubahan warna yang terdeteksi [9]. Penggunaan *PID Controller* juga memungkinkan pengaturan kecepatan yang lebih halus, sehingga gerakan robot menjadi lebih presisi dan stabil dalam mengikuti lintasan yang ditentukan [10] [11].

Penelitian ini bertujuan untuk mengembangkan robot otomatis yang mampu mengenali warna menggunakan sensor TCS3200 dan mengatur kecepatannya dengan menggunakan *PID Controller*. Dengan teknologi ini, robot tidak hanya dapat mendeteksi warna secara akurat tetapi juga menyesuaikan kecepatan gerakannya secara dinamis sesuai dengan warna yang terdeteksi. Robot ini dirancang untuk dapat bergerak sesuai dengan instruksi yang diterima melalui *smartphone*, sehingga memberikan fleksibilitas dalam pengaturan gerakan sesuai dengan kebutuhan pengguna. Ketika robot

bertemu dengan kartu warna tertentu, robot akan bergerak mengikuti pengaturan yang telah diatur melalui *smartphone*, seperti berhenti, mempercepat, atau berbelok [12]. Aplikasi teknologi ini telah diterapkan dalam berbagai skenario praktis, seperti penyortiran benda di industri, pengenalan warna untuk tujuan pemisahan objek, dan sistem navigasi otomatis [13].

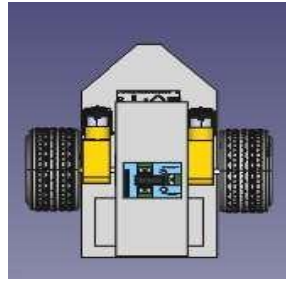
Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi terhadap pengembangan robot pengenalan warna yang lebih canggih dan responsif. Kombinasi antara sensor TCS3200 dan *PID Controller* dalam pengaturan kecepatan diharapkan dapat meningkatkan performa robot, baik dari segi akurasi deteksi warna maupun stabilitas gerakannya. Di sisi lain, integrasi dengan kontroler berbasis *smartphone* memberikan kemudahan dalam pengoperasian dan fleksibilitas dalam penyesuaian perintah, sehingga robot ini dapat diadaptasi untuk berbagai keperluan di lapangan [14].

2. METODE PENELITIAN

Pelaksanaan penelitian ini terdiri dari beberapa tahap yang terstruktur untuk memastikan tindakan yang diambil lebih terarah dan terkontrol. Setiap tahapan dirancang sebagai panduan dalam pelaksanaan penelitian untuk mencapai target yang diinginkan. Penjelasan mengenai tahapan-tahapan tersebut akan disampaikan lebih lanjut dalam sub-bab berikut ini.

2.1 Rancangan *Hardware*

Tahap rancangan *hardware* bertujuan untuk menentukan dimensi dan skala robot yang akan digunakan dalam pembelajaran. Dalam perencanaan desain, penulis menggunakan aplikasi *FreeCAD*. Robot ini dirancang dengan dimensi panjang 15 cm, lebar 9 cm, dan tinggi 8 cm. Jarak antara roda belakang dan roda depan pada robot diatur sejauh 6 cm untuk memastikan stabilitas saat bergerak. Desain gambar rancangan *hardware* robot yang dapat dilihat pada Gambar 1.

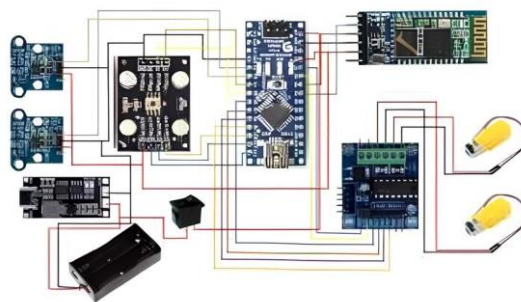


Gambar 1. Rancangan *hardware robot*

a. *Wiring Diagram Hardware Robot*

Berikut merupakan gambar wiring diagram *hardware robot* media pembelajaran

algoritma berbasis pengenalan warna dengan pengaturan kecepatan menggunakan *PID Controller* yang dapat dilihat pada Gambar 2.



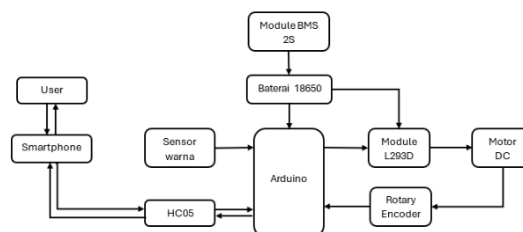
Gambar 2. *Wiring diagram hardware*

Diagram tersebut menunjukkan pengkabelan robot dengan Arduino Nano sebagai mikrokontroler pusat. Sensor HC 020K terhubung ke Arduino untuk data dan daya. Modul baterai dan pengisi daya juga terhubung. Modul *Bluetooth* HC-05 memungkinkan komunikasi dengan *smartphone*. *Driver* motor terhubung ke dua motor DC untuk daya dan arah, dengan kabel

berwarna yang menghubungkannya ke Arduino untuk kontrol motor.

b. *Flowchart Hardware Robot*

Berikut merupakan gambar *flowchart hardware robot* media pembelajaran algoritma berbasis pengenalan warna dengan pengaturan kecepatan menggunakan *PID Controller* yang disajikan pada Gambar 3.



Gambar 3. *Flowchart hardware robot*

Pada Gambar 3 *Flowchart hardware robot* dimulai dari interaksi pengguna melalui *smartphone* yang terhubung ke Arduino Nano via modul HC-05, modul komunikasi *Bluetooth*. Pengguna mengirim perintah ke *smartphone*, yang diteruskan ke Arduino Nano. Arduino Nano berfungsi sebagai pusat kendali, menerima data dari sensor warna untuk mendeteksi warna sebagai input.

Arduino juga mengontrol motor DC melalui modul L293D, yang memungkinkan pengaturan kecepatan dan arah motor berdasarkan sinyal kontrol. Motor DC diatur dengan sinyal PWM yang dihasilkan oleh *PID Controller* di dalam Arduino Nano. Kecepatan dan posisi motor dipantau oleh *rotary encoder* yang mengirimkan data kembali ke Arduino untuk memperbarui sinyal kontrol PID.

Sistem ini diberdayakan oleh baterai 18650 dengan modul BMS 2S yang mengatur pengisian dan pelepasan daya dari baterai, memastikan operasional yang aman dan efisien. Secara keseluruhan, sistem bekerja secara terintegrasi, memungkinkan pengguna mengontrol robot berdasarkan warna yang terdeteksi dengan presisi motor DC melalui *PID Controller*.

2.2 Rancangan Software

Tahap rancangan *Software* mencakup beberapa hal dan penulis membaginya menjadi beberapa aspek penting program yang akan mengontrol robot berdasarkan data dari sensor dan instruksi dari pengguna. Beberapa aspek penting dalam rancangan *software* meliputi :

a. Disain Antarmuka Aplikasi

Desain antarmuka aplikasi ini bertujuan untuk memberikan kontrol intuitif atas pengaturan kecepatan, warna, dan arah gerak. Di bagian atas, terdapat judul aplikasi "ALGOBOT" dengan huruf kapital. Di bawahnya, kolom status koneksi *Bluetooth* menunjukkan "Disconnected " jika belum terhubung dan "Connected" jika terhubung, ditampilkan dalam warna merah.

Tengah aplikasi terdiri dari tombol-tombol fungsional. Bagian "*SETTING SPEED*" memiliki tiga tombol hijau: "SLOW," "MEDIUM," dan "FAST" untuk mengatur kecepatan. Di bawahnya, terdapat tombol navigasi biru: "LEFT," "STRAIGHT," dan "RIGHT" untuk mengarahkan robot. Tombol merah "STOP" berfungsi menghentikan pergerakan. Ada juga tombol "WARNA" untuk pengaturan warna.

Di bagian bawah, terdapat dua tombol besar, biru dan kuning, bertuliskan "SAVE" dan "RESET" untuk menyimpan dan menghapus pengaturan aplikasi. Latar belakang aplikasi dihiasi gambar robot dan elemen grafis menarik, menciptakan antarmuka interaktif dan ramah pengguna.

b. Flowchart Kontrol PID

PID (Proportional-Integral-Derivative) Controller adalah metode pengendalian yang sering digunakan untuk mengatur kecepatan motor pada robot. *PID Controller* bekerja dengan menghitung tiga parameter utama: *proportional* (P), *integral* (I), dan *derivative* (D). Rumus dasar PID dapat dihitung menggunakan persamaan :

$$U(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \dots\dots\dots(1)$$

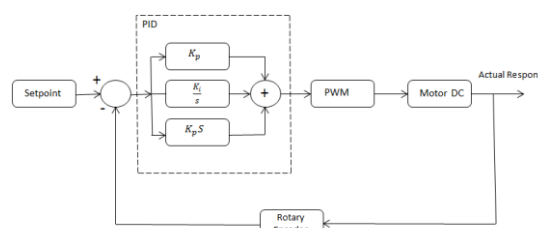
Penjelasan dasar PID:

- K_p adalah konstanta *proportional*,
- K_i adalah konstanta *integral*,
- K_d adalah konstanta *derivative*,
- $e(t)$ adalah *error* atau selisih antara nilai *setpoint* dan nilai aktual.

Dengan penggunaan *PID Controller*, robot mampu menyesuaikan kecepatan motor agar selalu mendekati nilai *setpoint*, sehingga menghasilkan pergerakan yang halus dan responif. Pengendalian PID memungkinkan untuk mengoreksi nilai *error* dengan mengolahnya menjadi sinyal kendali yang diteruskan ke motor, sehingga kecepatan motor dapat berubah-ubah sesuai dengan

pembacaan sensor [16]. Hal ini menunjukkan bahwa *PID Controller* sangat efektif digunakan sebagai alat pembelajaran dalam mengendalikan kecepatan robot.

Berikut merupakan gambar *flowchart control PID* robot media pembelajaran algoritma berbasis pengenalan warna dengan pengaturan kecepatan menggunakan *PID Controller* yang disajikan pada Gambar 4.



Gambar 4. Blok diagram control PID

Flowchart pada Gambar 4 menunjukkan diagram blok sistem kontrol PID untuk mengatur kecepatan atau posisi motor DC. Proses dimulai dengan menetapkan nilai target (*setpoint*) yang dibandingkan dengan nilai aktual (*actual response*) untuk menghasilkan sinyal *error*, yaitu selisih antara *setpoint* dan respon aktual. Sinyal *error* masuk ke PID Controller, yang terdiri dari tiga komponen: proporsional (K_p), integral (K_i/S), dan derivatif (K_dS). Komponen proporsional menghasilkan sinyal kontrol sebanding dengan *error* saat ini, komponen integral mengakumulasi *error* dari waktu ke waktu, dan komponen derivatif memperhitungkan laju perubahan *error*. Ketiga sinyal ini digabungkan untuk menghasilkan sinyal kontrol total [15].

Sinyal kontrol total kemudian digunakan untuk menghasilkan sinyal PWM (*Pulse Width Modulation*) yang mengatur daya motor DC. Motor menerima sinyal PWM dan menghasilkan respon aktual yang diukur oleh *rotary encoder*, yang mengirimkan informasi kembali ke sistem sebagai umpan balik.

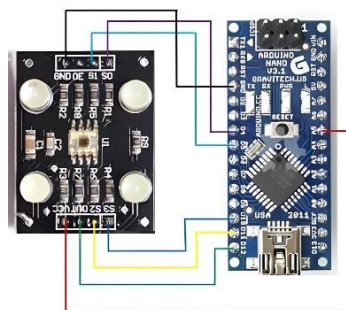
Respon aktual dibandingkan lagi dengan *setpoint* untuk menghasilkan *error* baru, dan proses ini berulang. Sistem ini memastikan kontrol presisi tinggi pada motor DC dengan mengurangi *error* dan menjaga respon sesuai *setpoint* yang diinginkan.

3. HASIL DAN PEMBAHASAN

Pada bab ini, dibahas hasil rancangan robot serta uraian dan analisis hasil pengambilan data yang telah dilakukan. Di bawah ini terdapat gambar hasil pembuatan "Robot Pengenal Warna dengan Pengaturan Kecepatan Menggunakan Metode PID"

3.1. Pengujian Kalibrasi Sensor Warna TCS3200

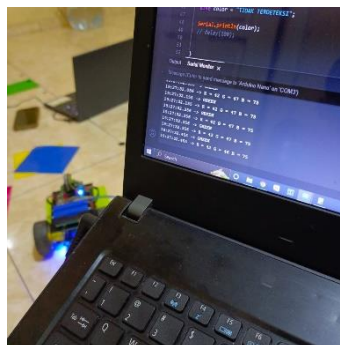
Pengujian ini dilakukan untuk memperoleh data dari kartu warna menggunakan sensor TCS3200. Data yang diperoleh akan dimasukkan ke dalam program Arduino untuk melakukan kalibrasi, agar sensor warna pada robot dapat membaca kartu warna dengan akurat.



Gambar 5. Wiring sensor warna TCS3200

Gambar 5 menunjukkan rangkaian antara sensor warna TCS3200 dan Arduino Nano. Sensor TCS3200 mendeteksi intensitas warna dan mengirimkan data frekuensi ke Arduino untuk diproses. Koneksi ini mencakup VCC yang terhubung ke 5V untuk suplai daya, GND ke GND untuk kesamaan tegangan,

serta pin S0, S1, S2, dan S3 yang terhubung ke pin digital Arduino untuk pengaturan frekuensi dan filter warna. Pin OUT terhubung ke input Arduino untuk mengirimkan data frekuensi deteksi warna. Data yang diperoleh akan dikalibrasi dalam program Arduino, agar sensor dapat membaca warna dengan akurat.



Gambar 6. Pengujian sensor warna TCS3200

Gambar 6 menunjukkan proses pengujian sensor warna TCS3200 menggunakan robot yang terhubung dengan komputer melalui program Arduino. Pengujian ini bertujuan untuk memperoleh nilai frekuensi yang dihasilkan oleh sensor saat mendeteksi berbagai kartu warna. Setiap kartu warna menghasilkan nilai frekuensi yang berbeda-beda, yang kemudian ditampilkan di layar komputer melalui serial monitor Arduino. Nilai-

nilai ini dicatat sebagai data acuan dan nantinya digunakan dalam proses kalibrasi program, agar robot dapat mengenali dan membedakan setiap warna dengan lebih akurat saat beroperasi. Hasil kalibrasi ini memungkinkan robot untuk memberikan respon yang tepat terhadap warna yang terdeteksi di lapangan.

Tabel 1. Data Pengujian Deteksi Sensor TCS3200

No	Object Color	Distance (cm)	RGB Value Reading					
			Sensor Value reading $\geq$			Sensor Value reading $\leq$		
			R	G	B	R	G	B
1	Red	0.5	30	127	65	41	138	75
	Green		38	37	80	50	47	85
	Blue		139	53	30	152	63	35
	Yellow		20	25	58	30	38	68
2	Red	1.5	50	137	80	56	150	90
	Green		50	50	90	65	65	99
	Blue		129	60	40	140	90	50
	Yellow		40	40	70	45	55	90
3	Red	2.5	60	155	100	80	165	110
	Green		70	70	100	80	75	105
	Blue		160	100	50	180	110	70
	Yellow		50	65	100	65	80	110

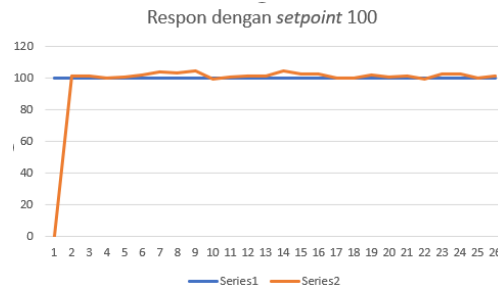
Berdasarkan Tabel 1 dapat disimpulkan bahwa jarak antara sensor warna dan objek mempengaruhi akurasi deteksi warna. Semakin dekat jarak sensor ke objek (dalam batas yang disarankan), semakin akurat deteksi yang diperoleh, dengan jarak optimal sekitar 0,5 cm. Jarak ini meningkatkan intensitas cahaya yang dipantulkan, mengurangi gangguan cahaya, dan meningkatkan resolusi. Penting untuk menemukan jarak optimal agar sensor beroperasi dalam kisaran intensitas cahaya yang sesuai, mencegah saturasi, dan mempertahankan akurasi tinggi, terutama untuk aplikasi yang memerlukan deteksi warna presisi dalam sistem pengendalian berbasis pengenalan warna.

3.2. Pengujian Sistem PID Controller ke Motor DC

Pengujian PID Controller terhadap motor DC bertujuan untuk memastikan bahwa PID Controller dapat mengatur kecepatan motor sesuai dengan nilai *setpoint* yang diinginkan.

- Pengujian Sistem PID Controller dengan *Setpoint* 100

Untuk *setpoint* 100 dengan parameter PID $K_p = 1,0$, $K_i = 0,0012$, dan $K_d = 0,00045$, sistem kontrol memberikan *respons* proporsional yang kuat, dengan cepat menangani kesalahan tetapi berpotensi mengalami *overshoot*. *Gain* integral yang kecil (K_i) memperbaiki kesalahan yang terakumulasi secara lambat, mengurangi *overshoot* tetapi mungkin memperlambat keadaan tunak. *Gain* derivatif yang minimal (K_d) menawarkan kontrol prediktif yang terbatas dan sedikit membantu mengurangi *overshoot*. Secara keseluruhan, pengaturan PID ini menyeimbangkan *respons* proporsional yang kuat dengan tindakan integral dan derivatif yang lebih lemah, menghasilkan sistem yang stabil yang mungkin memerlukan waktu lebih lama untuk stabil tetapi menghindari *overshoot* yang signifikan. Untuk hasil respon disajikan pada Gambar 7 dalam bentuk grafik.

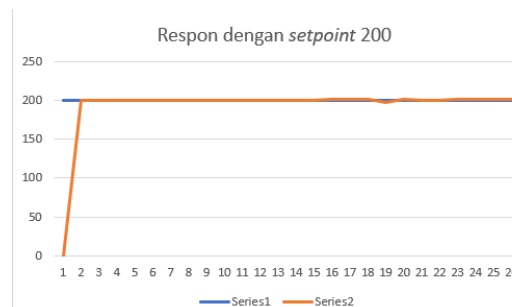


Gambar 7. Respon dengan setpoint 100

- Pengujian Sistem PID Controller dengan setpoint 200

Dengan setpoint 200 dan parameter PID $K_p = 1,0$, $K_i = 0,0002$, dan $K_d = 0,0005$, sistem kontrol memberikan respons proporsional yang kuat untuk koreksi kesalahan yang cepat tetapi dapat menyebabkan overshoot. Gain integral yang kecil (K_i) memperbaiki kesalahan yang terakumulasi secara lambat, mengurangi overshoot tetapi mungkin memperlambat

pencapaian keadaan tunak. Gain derivatif yang minimal (K_d) menawarkan prediksi kesalahan yang terbatas dan sedikit mengurangi overshoot. Pengaturan PID ini dioptimalkan untuk respons cepat dengan overshoot minimal, meskipun mungkin memerlukan waktu lebih lama untuk stabil dan beradaptasi dibandingkan dengan sistem yang memiliki nilai K_i dan K_d yang lebih tinggi. Untuk hasil respon disajikan pada Gambar 8 dalam bentuk grafik.

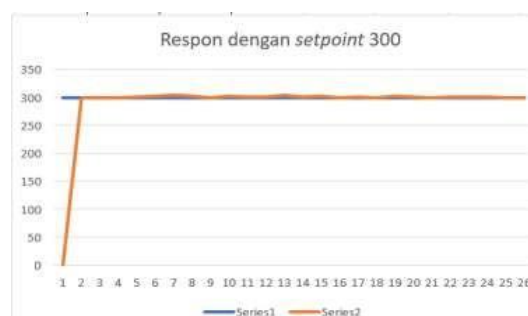


Gambar 8. Respon dengan setpoint 200

- Pengujian Sistem PID Controller dengan setpoint 300

Dengan setpoint 300, sistem kontrol PID menggunakan $K_p = 1,0$, $K_i = 0,00035$, dan $K_d = 0,00070$. Gain proporsional memastikan respons kesalahan yang seimbang, Gain integral secara bertahap mengurangi

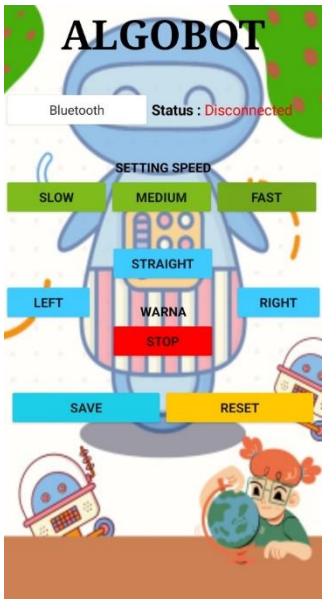
kesalahan keadaan tunak, dan Gain derivatif meminimalkan osilasi serta meningkatkan respons terhadap perubahan mendadak. Konfigurasi ini bertujuan untuk mencapai respons yang stabil dan seimbang dengan overshoot minimal. Untuk hasil respon disajikan pada Gambar 9 dalam bentuk grafik.



Gambar 9. Respon dengan setpoint 300

3.3. Pembuatan dan Pengujian Aplikasi
Berikut merupakan aplikasi dari robot
media pembelajaran algoritma berbasis

pengenalan warna dengan pengaturan
kecepatan menggunakan PID *Controller* yang
dapat dilihat pada Gambar 10.



Gambar 10. Desain antarmuka aplikasi

Gambar 10 menggambarkan antarmuka aplikasi ALGOBOT untuk mengontrol robot pengenalan warna. Antarmuka ini mencakup label status *Bluetooth*, tombol pengaturan kecepatan (*Slow, Medium, Fast*), serta kontrol navigasi

(*Left, Right, Straight, Stop*). Tombol warna muncul berdasarkan deteksi sensor. Di bagian bawah terdapat tombol Simpan dan *Reset*. Desain antarmuka ini ramah anak dengan elemen edukatif dan gambar robot.

Tabel 2. Tabel Pengujian aplikasi Algotob

Kecepatan	Warna	Printah Arah	Deskripsi Pengujian	Hasil
Slow	Merah	Right	Aplikasi Bergerak kekanan dengan kecepatan lambat	Berhasil
Slow	Hijau	Left	Aplikasi Bergerak kekiri dengan kecepatan lambat	Berhasil
Slow	Biru	Straight	Aplikasi Bergerak lurus dengan kecepatan lambat	Berhasil
Slow	Kuning	Stop	Aplikasi Bergerak berhenti dengan kecepatan lambat	Berhasil
Medium	Merah	Right	Aplikasi Bergerak kekanan dengan kecepatan sedang	Berhasil
Medium	Hijau	Left	Aplikasi Bergerak kekiri dengan kecepatan sedang	Berhasil
Medium	Biru	Stright	Aplikasi Bergerak lurus dengan kecepatan sedang	Berhasil
Medium	Kuning	Stop	Aplikasi Berhenti dengan kecepatan sedang	Berhasil
Fast	Merah	Right	Aplikasi Bergerak kekanan dengan kecepatan cepat	Berhasil
Fast	Hijau	Left	Aplikasi Bergerak kekiri dengan kecepatan cepat	Berhasil
Fast	Biru	Straight	Aplikasi Bergerak lurus dengan kecepatan cepat	Berhasil
Fast	Kuning	Stop	Aplikasi Berhenti dengan kecepatan cepat	Berhasil

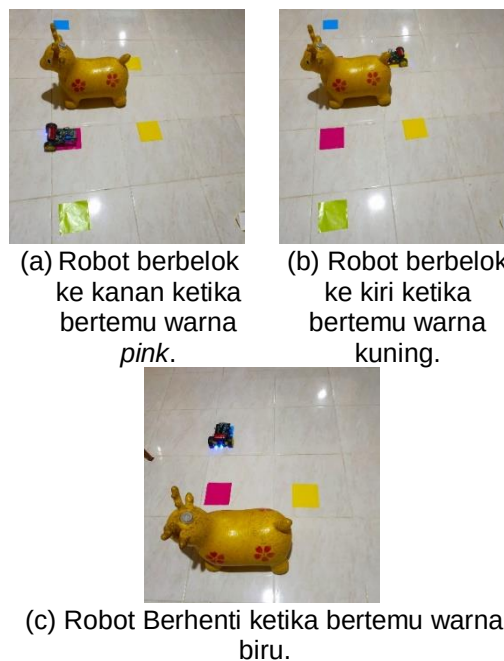
Berdasarkan Tabel 2 yang menampilkan hasil pengujian aplikasi "Algobot" dengan berbagai parameter, termasuk kecepatan, warna, arah perintah, deskripsi pengujian, dan hasilnya. Pengujian dilakukan dengan tiga kategori kecepatan, yaitu lambat (*slow*), sedang (*medium*), dan cepat (*fast*). Setiap kecepatan dikombinasikan dengan warna-warna tertentu yang tampaknya mewakili kondisi atau sinyal tertentu, seperti merah, hijau, biru, dan kuning.

Pengujian melibatkan instruksi arah gerakan, yaitu kanan (*right*), kiri (*left*), lurus (*straight*), dan berhenti (*stop*). Untuk setiap kombinasi kecepatan, warna, dan arah, dijelaskan bagaimana aplikasi berperilaku. Misalnya, pada kecepatan lambat dengan warna merah dan perintah "right", aplikasi bergerak ke kanan dengan kecepatan lambat. Hasil pengujian menunjukkan bahwa seluruh pengujian berhasil, tanpa adanya kesalahan atau kegagalan.

Secara keseluruhan, tabel ini menunjukkan bahwa aplikasi algobot mampu merespons berbagai kombinasi perintah dan kecepatan dengan baik, dengan hasil pengujian yang selalu berhasil.

3.4 Pengujian Keseluruhan Robot

Pengujian robot secara keseluruhan sangat penting untuk memastikan bahwa semua komponen dan sistem memenuhi spesifikasi. Fase ini mencakup pengujian modul perangkat keras dan perangkat lunak, seperti sensor, unit komunikasi (misalnya, *Bluetooth* HC-05), dan sistem *Control*, baik secara individual maupun dalam integrasi. Pengujian dilakukan dengan mensimulasikan berbagai kondisi operasional dan skenario kesalahan untuk memastikan ketahanan dan kinerja sistem. Pemeriksaan keselamatan juga dilakukan untuk menjamin bahwa perangkat beroperasi dengan andal dan aman. Untuk pengujian pergerakan robot dapat dilihat pada Gambar 11.



Gambar 11. Pergerakan Arah Robot

Berdasarkan Gambar 11 memperlihatkan rute yang diikuti oleh robot berdasarkan deteksi kartu warna yang telah ditempatkan di sepanjang jalur. Robot memulai perjalanan dari posisi awal dengan mendeteksi kartu *pink* yang menginstruksikan robot untuk berbelok ke kanan. Setelah itu, robot mendeteksi kartu

kuning yang menandakan bahwa robot harus berbelok ke kiri. Setelah berbelok ke kiri, robot kemudian mendeteksi kartu *pink* yang menginstruksikan robot untuk berbelok ke kanan hingga akhirnya robot mendeteksi kartu biru yang menunjukkan bahwa robot harus berhenti.

Tabel 3. Pengujian keseluruhan robot

Pengujian ke -	Deteksi warna	Navigasi	keterangan
1	Merah	Belok Kanan	Gagal
2	Merah	Belok Kanan	Berhasil
3	Merah	Belok Kanan	Berhasil
4	Hijau	Lurus	Berhasil
5	Hijau	Lurus	Berhasil
6	Biru	Stop	Berhasil
7	Biru	Stop	Gagal
8	Biru	Stop	Berhasil
9	Kuning	Belok Kiri	Berhasil
10	kuning	Belok Kiri	Berhasil

Tabel 3 menunjukkan hasil pengujian robot berdasarkan deteksi warna, navigasi, dan status keberhasilan atau kegagalan setiap pengujian. Pada pengujian pertama, robot mendeteksi warna merah dan seharusnya berhenti, namun hasilnya gagal. Sebaliknya, pengujian kedua dan ketiga yang juga melibatkan warna merah untuk perintah berhenti berhasil.

Pada pengujian keempat dan kelima, robot mendeteksi warna hijau yang menginstruksikan untuk bergerak lurus, dan keduanya berhasil. Untuk pengujian keenam hingga kedelapan, deteksi warna biru meminta robot belok kanan; pengujian keenam gagal, sedangkan ketujuh dan kedelapan berhasil.

Pengujian kesembilan dan kesepuluh menguji deteksi warna kuning untuk belok kiri, dengan kesembilan berhasil dan kesepuluh gagal. Meskipun terdapat beberapa kegagalan, sebagian besar pengujian menunjukkan hasil yang berhasil, menandakan bahwa sistem robot umumnya dapat mendeteksi warna dan melakukan navigasi sesuai instruksi.

4. SIMPULAN

Berdasarkan analisis penelitian berjudul Robot Pembelajaran Warna dengan Pengaturan Kecepatan PID disimpulkan bahwa sensor warna TCS3200 efektif mendeteksi warna dengan akurasi optimal pada jarak 0,5 cm, di mana intensitas cahaya tinggi dan gangguan minimal. PID Controller berhasil mengatur kecepatan motor DC dengan konfigurasi untuk *setpoint* 100 ($K_p = 1.0$, $K_i = 0.0012$, $K_d = 0.00045$), 200 ($K_p = 1.0$, $K_i = 0.0002$, $K_d = 0.0005$), dan 300 ($K_p = 1.0$, $K_i = 0.00035$, $K_d = 0.00070$), masing-masing mencapai *setpoint* dengan *overshoot* minimal. Aplikasi interaktif ALGOBOT yang dikembangkan dengan MIT App Inventor

memungkinkan kontrol robot berbasis warna melalui *smartphone*. Antarmuka aplikasi mencakup pengaturan kecepatan, kontrol navigasi, tampilan warna terdeteksi, serta opsi untuk menyimpan atau mereset pengaturan, dengan desain ramah pengguna dan elemen edukatif.

UCAPAN TERIMAKASIH

Penulis sampaikan kepada Politeknik Manufaktur Negeri Bangka Belitung atas dukungan, fasilitas, dan bimbingan yang telah diberikan dalam penyusunan jurnal ini. Terima kasih juga kepada dosen, staf, dan rekan-rekan yang telah membantu dalam proses penelitian. Semoga kerjasama ini dapat terus berkembang dan bermanfaat bagi pengembangan ilmu pengetahuan. sampaikan rasa terima kasih dukungan dari Politeknik Manufaktur Negeri Bangka Belitung yang sangat berarti bagi kami. Terima kasih atas kerja keras dan kontribusi yang berharga.

DAFTAR PUSTAKA

- [1] Lewenusa I. and W. A. Kusuma, "Autonomous mobile robot menggunakan metode formal logika temporal linier," *Seminar Nasional Informatika 2008 (SemnasIF 2008)*, Vols. ISSN: 1979-2328, 2008.
- [2] P. Risma, "Sensor Pemilih Warna," *Jurnal Teliska*, Vols. ISSN 2085-0786 , pp. 4(3), 29-34, 2012.
- [3] I. Farozzi, R. Maulana and W. Kurniawan, "Implementasi Sensor Warna pada Robot Lengan Pemindah Barang Menggunakan Inverse Kinematics," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Vols. e-ISSN: 2548-964X, p. 3(7), 2019.

- [4] N. L. Husni, S. Rasyad, M. S. Putra, Y. Hasan and J. Al Rasyid, "Pengaplikasian Sensor Warna pada Navigasi Line Tracking Robot Sampah Berbasis Mikrokontroler," *Jurnal AMPERE*, Vols. P-ISSN: 2477-2755, E-ISSN: 2622-2981, p. 4(2), 2019.
- [5] Fahmizal, E. A. K. R. and E. Iskandar, "Implementasi sistem navigasi behavior based dan kontroler PID pada manuver robot maze."
- [6] R. Solekha and U. Latifa, "Sistem Kendali Proportional Integral Derivative (PID) Menggunakan Mikrokontroler Arduino pada Thinkercad.," *Jurnal Ilmiah Teknik Elektro*, Vols. p-ISSN: 2830–523X, e-ISSN: 2622–6588, pp. 5(1), 89, 2024.
- [7] B. S. E. & S. D. Gusparentaqi, "Adaptasi Pergerakan pada Robot Beroda terhadap Jalan Halang Rintang dengan Metode Fuzzy.," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Vols. p-ISSN: 1873-1877, e-ISSN: 2548-964X, p. 6(4), 2022.
- [8] A. Najmurrokhman, K. Kusnandar, I. Irfansyah and A. Daelami, "Rancang Bangun Auto Balancing Robot Menggunakan Metode Kendali PID," *TELKA*, Vols. e-ISSN: 2540-9123, p-ISSN: 2502-1982, pp. 5(1), 15-23, 2019.
- [9] A. S. Priambodo, A. Nasuha and O. A. Dhewa, "Implementasi Integrasi dari Computer Vision dan Kendali PID untuk Kendali Kecepatan Dinamis pada Robot Line Follower.," *Jurnal Ilmiah Telekomunikasi Kendali dan Elektronika Terapan*, Vols. p-ISSN: 2303–2901, p. 21(1), 2024.
- [10] A. K. Zaky and E. Setiawan, "Sistem Pengatur Kecepatan Motor Robot Beroda untuk Mengurangi Selip Menggunakan Metode PID," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Vols. e-ISSN: 2548-964X, p. 1(1), 2017.
- [11] I. A. Wicaksono, T. Winarno and A. Komarudin, "Implementasi Kontrol PID pada Gerakan Robot Line Follower Berkaki Menggunakan Sensor Kamera," *JURNAL ELKOLIND*, p. 7(3), 2020.
- [12] A. J. Hulu, G. C. Setyawan and K. J. D. Lase, "Implementasi Kendali Google Voice pada Robot Aavoider HC-SR04 dengan Koneksi Bluetooth HC-05," *Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*, Vols. e-ISSN: 2685-0893, p-ISSN: 2089-3787.
- [13] J. Burdadi, Ridwan and I. Nugraha, "Aplikasi Sensor Warna TCS3200 pada Sistem Penyortiran Barang Berbasis Internet of Things (IoT)," *Seminar Nasional Inovasi Teknologi Terapan*, p. 141, 2022.
- [14] W. A. Dhimas, K. S. Santosob and R. Hartayud, "Pengembangan Kontrol Robot Mobil dengan Kerangka Kerja Robot Operating System," *Jurnal Elektro*, Vols. p-ISSN: 2527-6336, e-ISSN: 2656-7075, p. 6(1), 2024.
- [15] M. I. Marzuki, I. N. Farida and J. Sahertian, "Implementasi controller PID (proportional, integral, derivative) pada robot sepak bola beroda," *Jurnal Nasional Inovasi Teknologi*, Vols. 2549-7952, 2580-3336, , pp. 297-305., 2021.
- [16] Y. Suryadi, "Sistem Kendali Kecepatan Motor Pada Mobile Robot.," *TELKA : Jurnal Telekomunikasi, Elektronika, Komputasi, dan Kontrol.*, 2019.