

Pengendalian Gerak Robot Penghindar Halangan Menggunakan Citra dengan Kontrol PID

Basuki Winarno, S.T., M.T.
Jurusan Teknik
Politeknik Negeri Madiun
Madiun, Indonesia
basuki@pnm.ac.id

Meliyana Wahyu Putri Pratama
Jurusan Teknik
Politeknik Negeri Madiun
Madiun, Indonesia
meliyanaputri26@yahoo.com

Abstrak— Paper ini membahas tentang robot yang diintegrasikan dengan kamera sebagai pengolah citra sehingga memiliki kemampuan untuk menghindari halangan. Hasil citra yang di tangkap kamera akan diolah menjadi sebuah titik koordinat, kemudian dikirim ke Arduino Uno menggunakan komunikasi USART (*Universal Synchronous / Asynchronous Receiver/Transmitter*). Arduino Uno sebagai pusat pengaturan gerak menggunakan kontroler PID (*Proportional, Integral, Derivative*). Kontroler PID berfungsi untuk memuluskan pergerakan robot saat menghindari halangan dengan menghasikan nilai $K_p=4$, $K_i=0,00015$, $K_d=5$.

Kata kunci— Gerak Robot; PID; Arduino Uno; USART .

I. PENDAHULUAN

Robot merupakan suatu mekanik yang dapat diperintah secara otomatis untuk memproses suatu benda agar dapat bekerja sesuai dengan permintaan tertentu sehingga dapat meringankan pekerjaan manusia. Robot penghindar halangan merupakan kombinasi antara robotika dan kamera yang terintegrasi sehingga robot memiliki kemampuan menghindari halangan. Hasil citra yang di tangkap kamera akan diolah untuk menjadi sebuah titik koordinat, kemudian di kirim ke Arduino Uno menggunakan komunikasi USART. Arduino Uno sebagai pusat pengaturan gerak robot menggunakan sistem gerak differensial.[1]

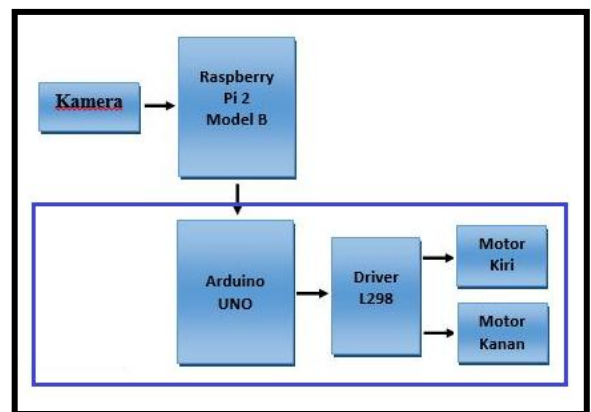
Kontroler PID merupakan control cerdas untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut. Kontroler PID (Proporsional, Integral, Differential) di tambahkan pada robot penghindar halangan untuk mengatur kecepatan motor sehingga pergerakan robot saat menghindari halangan menjadi lebih halus. Hal ini disebabkan karena hasil pembacaan citra memiliki nilai yang berubah di setiap robot bergerak.

II. METODOLOGI

A. Sistem Gerak Robot

Pada sistem gerak robot penghindar halangan, Arduino digunakan sebagai pusat pengolahan data saat dilakukan pengendalian gerak robot dengan masukan citra yang dihasilkan dari kamera. Hasil citra diolah menjadi sebuah

titik koordinat X yang selanjutnya dikirim ke Arduino menggunakan komunikasi USART (*Universal Synchronous / Asynchronous Receiver / Transmitter*). Setelah mendapat nilai koordinat, sistem robot akan menentukan nilai set point yang akan digunakan untuk menentukan arah gerak robot. Diagram kerja pengendalian gerak robot penghindar halangan di tampilan pada Gambar 1.

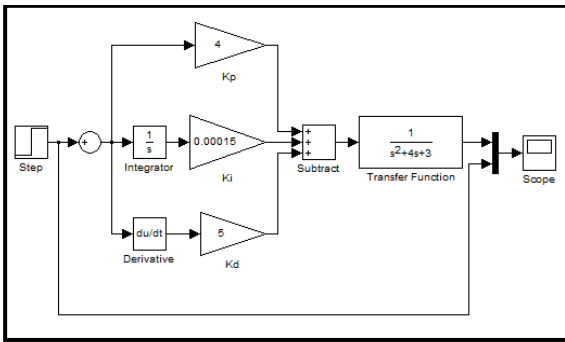


Gambar 1. Diagram kerja robot penghindar halangan

Kamera type Webcam WB100 5MP digunakan sebagai masukan untuk mendeteksi halangan. Raspberry Pi 2 Model B digunakan untuk pengolahan citra. Arduino UNO digunakan untuk pengendali sistem gerak robot. Aktuator untuk mengendalikan motor yang digunakan berupa motor DC dengan driver L298N.

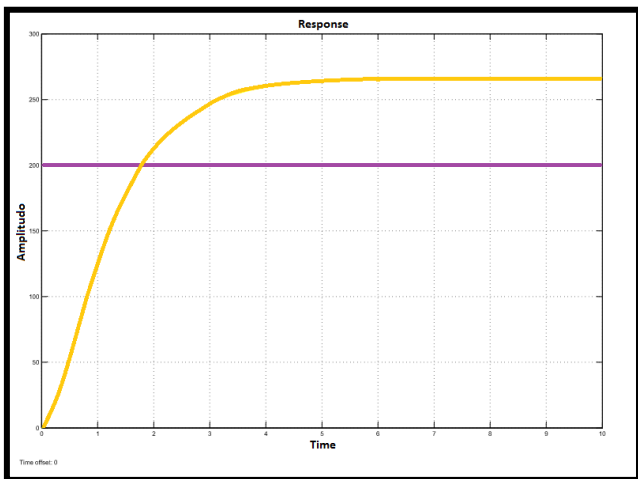
B. Kontroler PID

Kontroler PID berfungsi untuk memperhalus gerakan robot saat menghindari halangan dengan menggerakkan motor DC berdasarkan besar *error* yang diperoleh. Kontroler PID ini didapat menggunakan metode *trial and error* dengan *software* bantu dengan blok diagram ditampilkan pada Gambar 2.



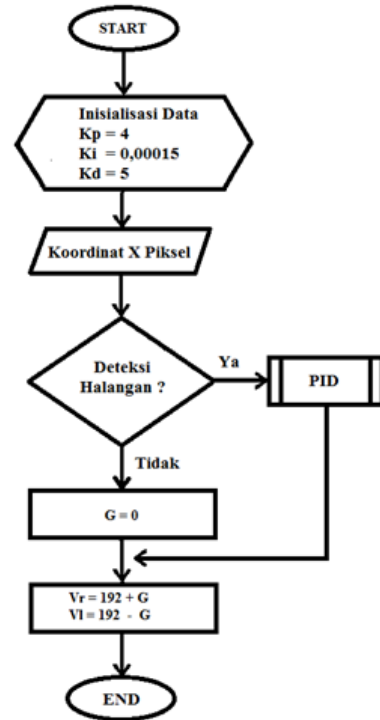
Gambar 2. Blok diagram PID

Gambar 2 adalah blok diagram pada matlab versi *student* untuk menentukan nilai K_p , K_i dan K_d pada kontroler PID yang akan digunakan. Berdasarkan *tunning* PID yang dilakukan, nilai PID yang diperoleh adalah $K_p=3$, $K_i=0,00015$ dan $K_d=5$. Namun saat diterapkan pada robot, nilai $K_p=3$ memiliki respon lambat pada robot sehingga nilai K_p diubah menjadi 4. Gambar 3 adalah respon saat nilai $K_p=4$, $K_i=0,00015$ dan $K_d=5$. Nilai kontroler PID ini sesuai saat digunakan pada robot untuk menghindari halangan.



Gambar 3. Grafik respon kontroler PID dengan nilai $K_p=4$, $K_i= 0,00015$ dan $K_d=5$

Diagram alir sistem pengendalian gerak robot penghindar halangan ditunjukkan pada Gambar 4. Inisialisasi data dilakukan dengan nilai K_p , K_i dan K_d yang telah didapatkan untuk menentukan nilai set point yang akan digunakan sebagai acuan pembacaan koordinat. Hasil dari pembacaan digunakan untuk mencari nilai gain (G) dengan akumulasi error dan perhitungan K_p , K_i dan K_d pada PID sehingga diperoleh nilai V_r dan V_l sebagai proses Bergeraknya robot saat menghindari halangan berupa balok kuning sehingga robot akan bergerak lurus, ke kanan atau ke kiri untuk menghindari halangan tersebut.

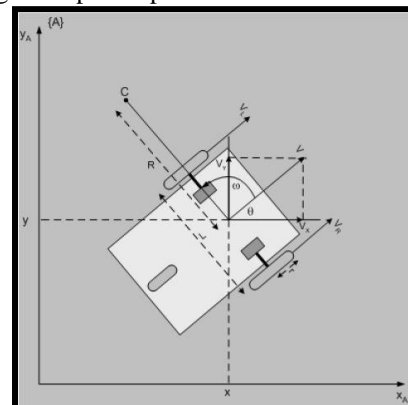


Gambar 4. Diagram alir sistem gerak robot penghindar halangan

Output dari pengolahan citra berupa koordinat dan diolah menjadi arah gerak. Apabila robot mendeteksi halangan, maka kontroler PID akan mengatur kecepatan motor DC agar bergerak menghindari halangan.

C. Metode Differential

Sistem kemudi yang digunakan adalah sistem penggerak *differential*. Robot penghindar halangan memiliki dua roda utama yang masing-masing digerakan oleh penggerak tersendiri dan satu buah roda *castor* yang ditempatkan dibagian belakang robot yang berfungsi sebagai penyeimbang ditampilkan pada Gambar 5.

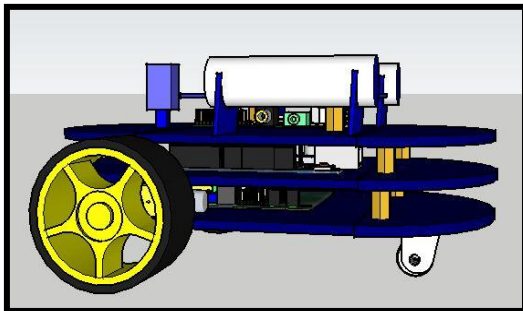


Gambar 5. Posisi dan orientasi mobile robot dalam sistem koordinat *cartesian*

Gambar 5 merupakan arsitektur robot dilihat dari bagian atas. Jika kedua roda penggerak tersebut berputar dengan kecepatan yang sama maka robot tersebut akan bergerak dengan arah yang lurus, sedangkan jika kecepatan salah satu roda lebih lambat maka robot akan bergerak membentuk kurva dengan arah lintasan menuju salah satu roda yang bergerak lebih lambat. Gerak kedua roda berdasar output dari motor DC dengan gerakan maju, ke kanan atau ke kiri. Sedangkan roda *castor* akan mengikuti arah gerak kedua roda utama.

D. Perancangan Perangkat Keras

Sistem mekanik yang baik dapat mendukung pergerakan robot menjadi lebih baik. Desain robot dari bahan dasar akrilik dengan dimensi lebar 13 cm, panjang 15 cm, dan tinggi 14 cm yang ditampilkan pada Gambar 6.



Gambar 6. Desain robot penghindar halangan

Robot memiliki 3 tingkatan. Tingkat pertama terdapat driver L298N, Motor DC dan roda. Tingkat kedua terdapat arduino, rangkaian buzzer dan button. Tingkatan ketiga terdapat kamera, kipas, baterai dan raspberry pi. Hasil rancangan robot penghindar halangan yang telah dibuat, di tampilkan pada Gambar 7 dan Gambar 8.

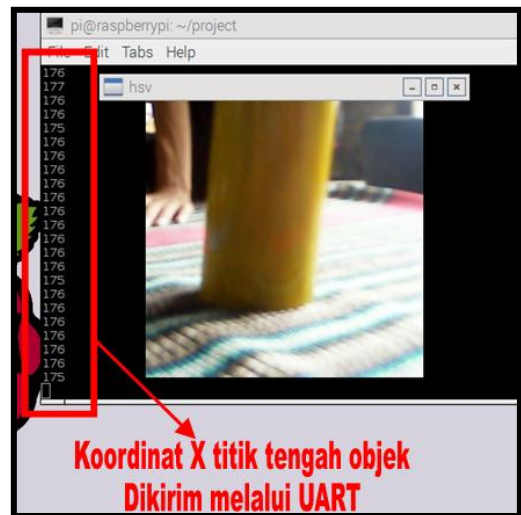


Gambar 7 Robot mendeteksi halangan

III. HASIL DAN ANALISA

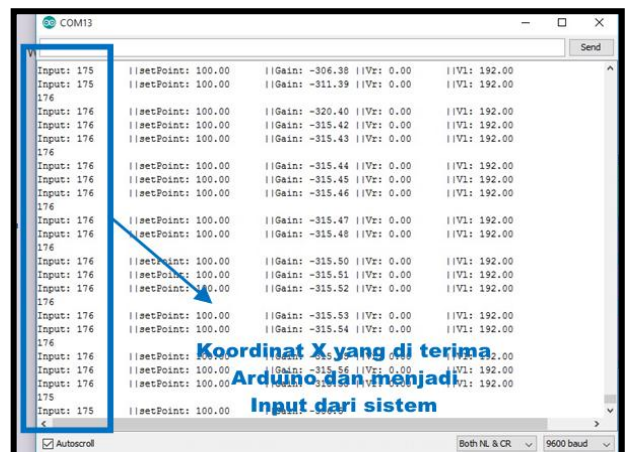
A. Penerimaan Hasil Pengolahan Citra Berupa Koordinat X Pikel Titik Tengah Objek Penghalang.

Pada saat dilakukan pengujian, dan sistem menjadi mode *Running* maka raspberry menampilkan citra asli yang di tangkap kamera dan *Serial Monitor* menunjukkan koordinat titik tengah objek. Selanjutnya data dikirim melalui komunikasi USART ke Arduino dengan nilai koordinat X titik tengah objek penghalang. Penghalang yang digunakan berupa berupa tabung berwarna kuning yang diletakkan secara acak di sekitar robot di tunjukkan pada Gambar 9.



Gambar 8. Tampilan koordinat X titik tengah halangan yang terbaca kamera

Gambar 8 merupakan hasil pengolahan citra pada tampilan *mode running* raspberry yang dikirim ke arduino berupa koordinat X titik tengah halangan yang terdeteksi, sedangkan koordinat yang di terima Arduino ditampikan pada Gambar 9.



Gambar 9. Tampilan koordinat X titik tengah halangan yang diterima arduino

Gambar 9 merupakan hasil pengolahan citra pada tampilan *mode running* arduino akan diubah menjadi arah gerak robot untuk menghindari halangan.

B. Perhitungan Kontroler PID

Nilai kontroler PID yang diperoleh berdasarkan metode *trial and error* adalah $K_p=4$, $K_i=0,00015$, $K_d=5$ yang digunakan sebagai masukan untuk perhitungan nilai error, Gain, V_r dan V_l sesuai Persamaan 1,2,3 dan 4.

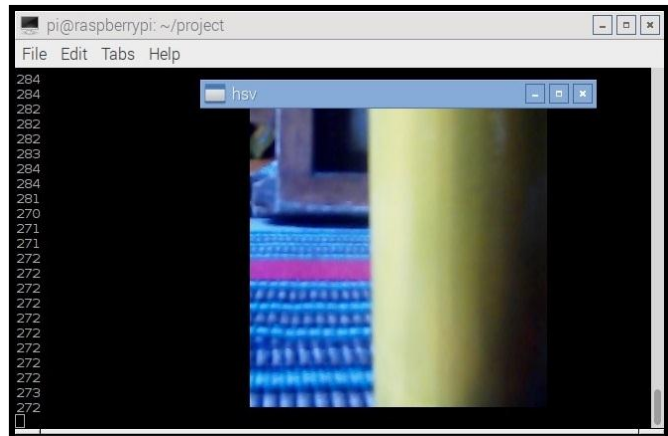
$$\begin{aligned} \text{error} &= \text{set point} - XG \dots\dots\dots (1) \\ G &= (K_p * \text{error}) + (K_d * (\text{error} - \text{previous error})) + \\ &\quad (K_i * \sum \text{error}) \dots\dots\dots (2) \\ V_r &= 192 + G \dots\dots\dots (3) \\ V_l &= 192 - G \dots\dots\dots (4) \end{aligned}$$

Keterangan :

- G = Gain (Konstanta PID)
- K_p = Konstanta *Proportional*
- K_i = Konstanta *Integral*
- K_d = Konstanta *Derivative*
- $\sum \text{error}$ = Total *error*
- Set Point* = 100 (jika halangan berada di kiri)
300(jika halangan berada di kanan)
- X = Koordinat halangan yang terbaca kamera berdasarkan nilai piksel pada sumbu X.
- V_r = Kecepatan Roda Kanan
- V_l = Kecepatan Roda Kiri
- 192 = Nilai PWM

Pengujian 1

Halangan Berada di S sebelah Kanan Robot



Gambar 10. Halangan di kanan robot kanan

Gambar 10 menampilkan bahwa halangan berada di sebelah kanan robot, maka nilai koordinat yang terbaca >200 seperti Gambar 10. Apabila halangan berada di sebelah kanan robot, maka nilai *set point* adalah 300. Kemudian robot akan mencari nilai error, gain, V_l dan V_r berdasarkan persamaan 1 sampai dengan 4.

Input: 272	setPoint: 300,00	Gain: 108,91	Vr: 192,00	Vl: 83,09
Input: 272	setPoint: 300,00	Gain: 108,92	Vr: 192,00	Vl: 83,08
272				
Input: 272	setPoint: 300,00	Gain: 108,92	Vr: 192,00	Vl: 83,08
Input: 272	setPoint: 300,00	Gain: 108,93	Vr: 192,00	Vl: 83,07
Input: 272	setPoint: 300,00	Gain: 108,93	Vr: 192,00	Vl: 83,07
272				
Input: 272	setPoint: 300,00	Gain: 108,94	Vr: 192,00	Vl: 83,06
Input: 272	setPoint: 300,00	Gain: 108,94	Vr: 192,00	Vl: 83,06
272				
Input: 272	setPoint: 300,00	Gain: 108,94	Vr: 192,00	Vl: 83,06
Input: 272	setPoint: 300,00	Gain: 108,95	Vr: 192,00	Vl: 83,05
Input: 272	setPoint: 300,00	Gain: 108,95	Vr: 192,00	Vl: 83,05
272				
Input: 272	setPoint: 300,00	Gain: 108,96	Vr: 192,00	Vl: 83,04
Input: 272	setPoint: 300,00	Gain: 108,96	Vr: 192,00	Vl: 83,04
272				
Input: 272	setPoint: 300,00	Gain: 108,97	Vr: 192,00	Vl: 83,03
Input: 272	setPoint: 300,00	Gain: 108,97	Vr: 192,00	Vl: 83,03
Input: 272	setPoint: 300,00	Gain: 108,97	Vr: 192,00	Vl: 83,03
273				
Input: 273	setPoint: 300,00	Gain: 99,98	Vr: 192,00	Vl: 92,02
Input: 273	setPoint: 300,00	Gain: 104,98	Vr: 192,00	Vl: 87,02
Input: 273	setPoint: 300,00	Gain: 104,99	Vr: 192,00	Vl: 87,01
272				

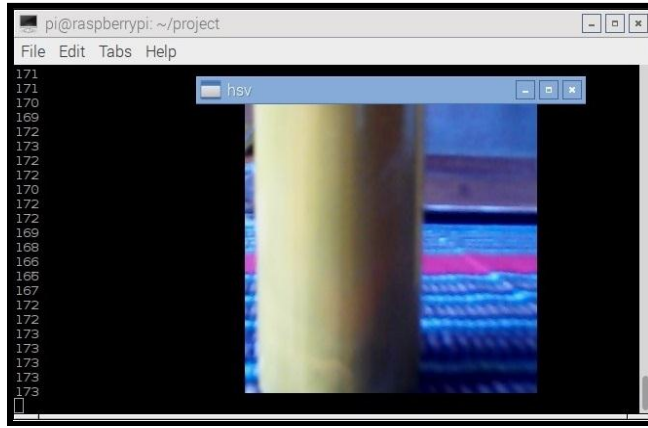
Gambar 11. Hasil monitoring robot bergerak ke kanan

Perhitungan pada robot saat halangan yang terdeteksi berada di sebelah kanan robot adalah

$$\begin{aligned} \text{previous error} &= \text{set point} - X \\ &= 300 - 272 \\ &= 28 \\ \text{error} &= \text{set point} - X \\ &= 300 - 273 \\ &= 27 \\ G &= (K_p * \text{error}) + (K_d * (\text{error} - \text{previous error})) + \\ &\quad (K_i * \sum \text{error}) \\ &= (4 * 27) + (5 * (27 - 28)) + (0,00015 * (27 + 28)) \\ &= 108 - 5 + 0,00825 \\ &= 103,00825 \\ V_r &= 192 + G \\ &= 192 + 103,00825 \\ &= 295,00825 \\ &= 192 \\ V_l &= 192 - G \\ &= 192 - 103,00825 \\ &= 89,000825 \end{aligned}$$

Pengujian 2

Halangan Berada di S sebelah Kiri Robot



Gambar 12. Halangan di kiri robot

jika <200 maka halangan berada di sebelah kiri robot dan set point menjadi 100 sehingga robot akan bergerak ke kanan. Jika nilai koordinat yang terbaca >200 maka halangan berada di sebelah kanan robot dan set point menjadi 300 sehingga robot akan bergerak ke kiri. Gerak robot untuk menghindari halangan akan berjalan lebih baik lagi jika menggunakan spesifikasi motor yang lebih baik. Selain itu dapat juga ditambahkan sensor *rotary encoder* serta sensor kompas untuk mengetahui posisi robot yang lebih akurat

DAFTAR PUSTAKA

- [1] S.Aditya, F.Ardilla, A.R.A. Besari "Robot Pendeteksi Wajah dan Penghindar Halangan", Paper, Jurusan Teknik Komputer, Politeknik Elektronika Negeri Surabaya, 1955.
- [2] J. Guilberto, "An Autonomous Robot with Reconfigurable Hardware and RT Linux For Fire-Fighting", Intelligent Systems and Robotics Group, Electrical Engineering Department, New Mexico Institute of Mining and Technology, New Meksiko
- [3] A. Dian, "Pengenalan Arduino", Yogyakarta, 2012
- [4] Atmel, Datasheet Arduino Uno R3
- [5] Atmel, IC L298N