

# Perancangan Jaringan Saraf Tiruan Untuk Pengenalan Pola Pada Navigasi *Mobile Robot* Berbasis Data Gambar

Pauzi Saputra, Ir. Rusdhianto Effendie A.K., M.T., dan Ir. Joko Susila, MT.

Departemen Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember  
e-mail: rusdhi@elect-eng.its.ac.id

**Abstrak**—Perkembangan ilmu teknologi khususnya pada sistem serba otomatisasi yang salah satunya melibatkan peran robotika pada kendaraan semakin terbuka lebar. *Mobile robot* merupakan salah satu bentuk prototype. Penerapan sistem cerdas pada robot dibutuhkan karena robot diberi pilihan sesuai navigasi. Navigasi sebagai petunjuk dari arah robot beroperasi. Untuk menerapkan sistem navigasi tersebut maka diterapkan pengenalan pola dari obyek yang dijadikan navigasi robot. Tujuan pengenalan pola adalah untuk mengenali karakteristik benda. Pengenalan pola benda dilakukan berkali-kali terhadap robot untuk melatih kecerdasan robot dalam melakukan pengidentifikasian obyek navigasi. Kemudian robot diberi tugas untuk mengelompokkan dari bentuk yang telah dikenali sebagai pembeda arah tiap navigasi. Navigasi berupa obyek yang direkam melalui kamera robot sebagai indra penglihatan buatan. Obyek bisa dibedakan dengan obyek lain dengan menentukan batas warna biru, dan selain biru tidak dijadikan bentuk navigasi. Pada penelitian studi sistem diimplementasikan pada *mobile robot Qbot* dan hasil tersebut disimulasikan dalam bentuk *file trajectory* luasan operasi robot menjadi titik koordinat robot beroperasi sebagai pembandingan maupun *interface* saat robot dioperasikan. Pergerakan *mobile robot* dapat lebih leluasa manuver sesuai jalur itu disebabkan sistem terhubung jaringan *wireless* dengan komputer sebagai *host-nya*.

**Kata kunci**—Pengenalan Pola, Sistem Cerdas, *Mobile robot*, Jalur, Navigasi, Posisi.

## I. PENDAHULUAN

SEMAKIN berkembangnya ilmu pengetahuan dan teknologi mendorong manusia menciptakan inovasi kreatif, canggih dan berguna. Penerapan inovasi tersebut bertujuan untuk mempermudah pekerjaan dan permasalahan yang dianggap memperlama tercapainya hasil yang diharapkan. Penerapan tersebut diantaranya menggunakan robot, karena sebagian besar khusus sistem otomatisasi membutuhkan teknologi robotik yang dapat beroperasi diberbagai kendala. Melengkapi kinerja robot maka dibutuhkan yang berupa *input* yaitu penggunaan sensor-sensor, motor listrik, controller dan agotihma untuk menyelesaikan berbagai macam problem.

Penerapan sistem otomatisasi pada kendaraan menggunakan kamera tanpa pengemudi pada kendaraan dalam mencapai tujuan disebut *Automatic guided vehicle (AGV)*. Pada 1989 diperkenalkan oleh Eberhardt bahwa tujuan belum diketahui, penerapan sistem dengan host tanpa kabel dan semacamnya. Pada kesempatan ini penerapan tersebut diimplementasikan pada *prototype mobile robot* yang memiliki sistem navigasi.

Sistem navigasi yang diterapkan pada *mobile robot* menggunakan data gambar yang diperoleh dari kamera robot yang kemudian disimulasikan dan diimplementasikan secara *online* menggunakan *wireless connection* antara robot dan komputer. Navigasi menentukan arah tujuan robot yang nantinya dianalisa diberbagai kondisi seperti saat belok maupun di pertigaan jalan. Jalur menentukan kecepatan roda pada *mobile robot* karena kondisi tersebut.

## II. LANDASAN TEORI

### A. Pengertian Robot

Robot pertama kali diperkenalkan kedalam kosa kata oleh dramawan Ceko Karel Capek (1920) yaitu "*robot*", dalam bahasa Ceko memiliki arti "bekerja". Sejak saat itu istilah tersebut telah diaplikasikan untuk bermacam – macam perangkat mekanik, seperti teleoperator, kendaraan bawah air, dan lain sebagainya. Hampir semua yang beroperasi di bawah kendali dari komputer dapat disebut dengan robot. RIA (*Robot Institute of America*) mendefinisikan bahwa sebuah robot merupakan manipulator multifungsi bisa di program (*reprogrammable*) ulang yang didesain untuk memindahkan material, suku cadang, peralatan atau perangkat khusus dengan variabel gerakan yang diprogram untuk berbagai tugas.

### B. Quanser Mobile Robot [1]

*Quanser mobile robot* merupakan *autonomus robot* buatan *Irobot* yang telah dilengkapi sistem kontrol terpusat pada sebuah komputer. *Mobile robot* ini diberi nama singkat yaitu robot *Qbot*. Robot ini dirancang untuk menguasai ground yang diintegrasikan dengan perangkat keras seperti sensor kamera, sensor *infra red*, sensor *sonar*, sensor posisi dan sensor jarak. Kemampuan robot mengirimkan data pada computer sebagai *host-nya* menggunakan perangkat lunak *QUARC (Quanser Controller Module)*. *QUARC* merupakan *software quanser* pada komputer. *Software* ini memfasilitasi pemrograman menggunakan *MATLAB* sehingga bisa dikembangkan menggunakan *Simulink* dan *M.File* sesuai kebutuhan pengguna. Komunikasi robot dengan robot menggunakan *Gumstix* yang memberikan sinyal dari komputer ke robot dan sebaliknya. Bentuk *Quanser Mobile Robot Qbot* seperti Gambar 1.

### C. Navigasi Pada Mobile Robot

Navigasi merupakan penentuan posisi dan arah terhadap tujuan dalam suatu pemetaan. Penentuan posisi *mobile robot*

berubah sehingga mempengaruhi ingatan yang baik agar tidak membingungkan. Navigasi kendaraan menggunakan metode metode garis diterapkan pada *mobile robot*. Batas tepi depan jalan digunakan sebagai kontrol kecepatan kendaraan. Langkah-langkah menggunakan metode garis adalah :

- 1) Menentukan titik awal  $X_0$  dan  $Y_0$  terlebih dahulu
- 2) Menentukan nilai  $Y_1$  berdasarkan sudut  $\theta$  yang diinginkan

$$Y_1 = X_0 \tan \theta \tag{1}$$

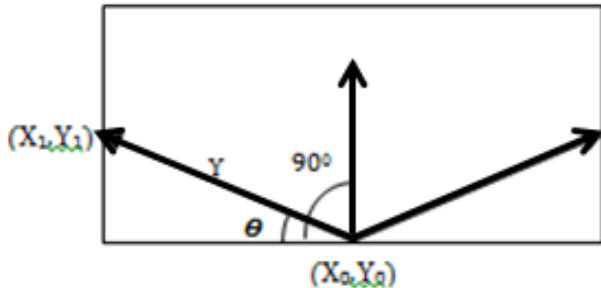
- 3) Setelah  $Y_1$  diketahui, maka menentukan panjang garis horisontal yang diinginkan  $X_1$  sehingga didapatkan nilai gradien  $m$

$$m = \frac{Y_0 - Y_1}{X_0 - X_1} \tag{2}$$

- 4) Setelah titik awal ( $X_0$  dan  $Y_0$  pada langkah 1) dan garis horisontal yang diinginkan ( $X_1$ ) ditentukan serta gradien ( $m$ ) diketahui, maka dimasukkan dalam rumus

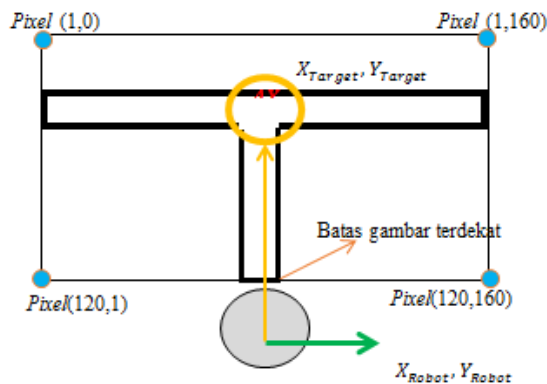
$$Y = X_0 - m (X_0 - X_1) \tag{3}$$

Sepanjang  $Y$  dicari batas jalan sehingga didapatkan Gambar 1.



Gambar 1. Navigasi Menggunakan Metode Garis

Metode ini diterapkan pada *mobile robot* seperti pada Gambar 2 berikut.



Gambar 2. Pengenalan Pertigaan

Keterangan:

- 1)  $X_{Robot}$  adalah robot berada pada posisi koordinat  $X$  robot saat sekarang
- 2)  $Y_{Robot}$  adalah robot berada pada posisi koordinat  $Y$  robot saat sekarang
- 3)  $X_{Target}$  adalah koordinat  $X$  yang akan dituju oleh robot pada pertigaan

- 4)  $Y_{Target}$  adalah koordinat  $Y$  yang akan dituju oleh robot pada pertigaan

#### D. Jaringan Saraf Tiruan [2]

Perhitungan *feedforward* adalah sebagai berikut.

Langkah 1 :

Masing-masing unit *input* ( $x_i$ ) untuk  $i = 1, \dots, n$ , menerima sinyal *input*  $x_i$ . Sinyal tersebut disebarkan ke tiap bagian unit lapisan *hidden layer*.

Langkah 2 :

Tiap unit *hidden layer* dikalikan dengan faktor pembobot, dijumlahkan dan ditambah dengan biasnya:

$$z\_in_j = v_{0j} + \sum_{i=1}^n x_i v_{ij} \tag{4}$$

Kemudian, Hitungan disesuaikan dengan fungsi aktivasi yang digunakan:

$$z_j = f(z\_in_j) \tag{5}$$

Langkah 4 :

Tiap unit *output* ( $y_k, k=1,2,3\dots m$ ) dikalikan dengan faktor pembobot dan dijumlahkan:

$$y\_in_k = w_{0k} + \sum_{j=1}^p z_j w_{jk} \tag{6}$$

Perhitung fungsi aktivasi  $y_k$ .

$$y_k = f(y\_in_k) \tag{7}$$

Kemudian dilanjutkan perhitungan *Backward* (*Backpropagation*) dan *error*-nya.

Langkah 5:

Masing-masing unit *output* ( $y_k, k=1, \dots, m$ ) menerima pola target sesuai dengan pola *input* saat pelatihan/*training* dan dihitung *error*nya:

$$\delta_k = (t_k - y_k) f'(y\_in_k) \tag{8}$$

Menghitung perbaikan faktor pembobot (kemudian untuk memperbaiki  $w_{jk}$ ):

$$\Delta w_{jk} = \alpha \delta_k z_j \tag{9}$$

Menghitung perbaikan bias koreksi bias:

$$\Delta w_{0k} = \alpha \delta_k \tag{10}$$

dan menggunakan nilainya pada semua unit lapisan sebelumnya.

Langkah 6 :

Masing-masing pembobot yang menghubungkan unit-unit lapisan *output* dengan unit-unit pada lapisan tersembunyi ( $z_j, j=1 \dots, p$ ) dikalikan delta dan dijumlahkan sebagai *input* ke unit-unit lapisan berikutnya.

$$\delta\_in_j = \sum_{k=1}^m \delta_k w_{jk} \tag{11}$$

Selanjutnya dikalikan dengan turunan dari fungsi aktivasinya untuk menghitung *error*nya.

$$\delta_j = \delta\_in_j f'(y\_in_j) \tag{12}$$

Kemudian menghitung perbaikan pembobot (digunakan untuk memperbaiki  $v_{ij}$ ).

$$\Delta v_{ij} = \alpha \delta_j x_i \tag{13}$$

Kemudian perbaikan bias (untuk memperbaiki  $v_{0j}$ )

$$\Delta v_{0j} = \alpha \delta_j \tag{14}$$

Langkah 7:

Tiap *output* unit  $w_k$ , untuk  $k=1, \dots, m$  diperbaiki bias dan pembobotnya ( $j=0, \dots, p$ )

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \quad (15)$$

Tiap unit tersembunyi  $z_j$ , untuk  $j: 1, \dots, p$  diperbaiki bias dan pembobotnya ( $j=0, \dots, n$ ).

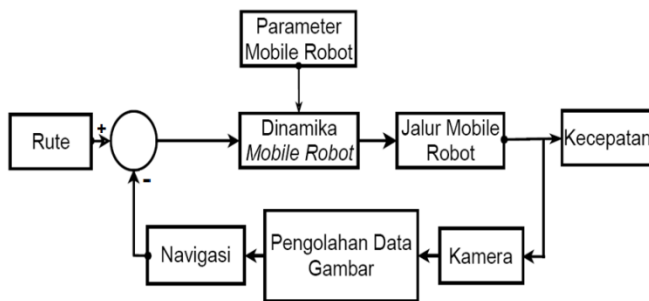
$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \quad (16)$$

### III. PERANCANGAN SISTEM

Pembahasan pada bab ini mengenai proses perancangan sistem, yaitu merancang simulasi dan implementasi perencanaan jalur *mobile robot* Qbot. Perancangan algoritma untuk perencanaan jalur dengan menggunakan metode Jaringan Saraf Tiruan sebagai pengenalan pola berbasis data gambar.

#### A. Sistem Penerapan Navigasi Pada Mobile Robot Qbot

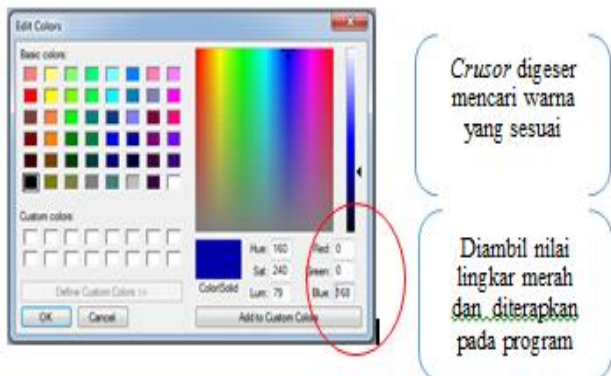
Diagram blok pada Gambar 3 berikut merupakan sistem keseluruhan dari pemrosesan penerapan navigasi pada mobile robot.



Gambar 3. Diagram Blok Sistem

#### B. Pengaturan Warna pada Line Image

Warna yang diambil pada line jalur mengambil warna biru, untuk mengetahui berapa range warna yang dideteksi menggunakan software paint *Tool Microsoft* pada Gambar 4.



Gambar 4. Pengambilan Warna identifikasi obyek

Untuk menganalisis karakteristik gambar yang telah di-*crop* maka dibutuhkan momen *invariant* untuk menyatakan obyek dengan memperhitungkan area obyek. Momen ini menggunakan momen spasial dan momen pusat kedua momen ini nantinya digunakan untuk menangani translasi, penekalaan, dan rotasi gambar yang dinamakan momenhu. Penciptanya teori ini yaitu Hu (Theodoridis dan Koutroubus, 2006), Dalam pengolahan citra momen spasial digunakan untuk menganalisis

orde dua dimensi untuk menyatakan jumlah baris dan kolom, dengan orde  $(m, n)$  yang didefinisikan sebagai berikut :

$$M_{ij} = \sum_{x=1}^M \sum_{y=1}^N x^i y^j I(x, y) \quad (3.21)$$

Keterangan,

- 1)  $i$  dan  $j$  menyatakan orde momen untuk  $i = 0, 1, 2, \dots$ , dan  $j = 0, 1, 2, 3, \dots$
- 2)  $M$  menyatakan jumlah kolom pada citra
- 3)  $N$  menyatakan jumlah baris pada citra.
- 4)  $X$  adalah ordinat piksel
- 5)  $Y$  adalah absis piksel
- 6)  $I(x, y)$  menyatakan intensitas piksel pada posisi  $(x, y)$

Adapun momen pusat adalah momen spasial yang dihitung relatif terhadap pusat massa. Jika pusat massa adalah  $(\bar{y}, \bar{x})$ , momen pusat ditulis sebagai berikut :

$$\mu_{ij} = \sum_{x=1}^M \sum_{y=1}^N (x - \bar{x})^i (y - \bar{y})^j I(x, y) \quad (17)$$

Agar momen pusat bersifat bebas terhadap translasi, penekalaan, dan rotasi maka momen perlu di dinormalisasi. Momen pusat ternormalisasi berupa:

$$\eta_{ij} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}}, \gamma = \frac{i+j+2}{2} \quad (18)$$

Momen yang dihasilkan dapat digunakan untuk menangani translasi, penekalaan, dan rotasi pada gambar. Dikenal sebagai momenhu. Momen ini memiliki 7 momen invariant seperti berikut.

$$\theta_1 = \eta_{20} + \eta_{02} \quad (19)$$

$$\theta_2 = (\eta_{20} + \eta_{02})^2 + 2\eta_{02}^2 \quad (20)$$

$$\theta_3 = (\eta_{30} + 3\eta_{12})^2 + (\eta_{03} - 3\eta_{21})^2 \quad (3.27)$$

$$\theta_4 = (\eta_{30} + 3\eta_{12})^2 + (\eta_{03} - 3\eta_{21})^2 \quad (21)$$

$$\theta_5 = (\eta_{30} + 3\eta_{12})(\eta_{03} - 3\eta_{21}) [(\eta_{30} + 3\eta_{12})^2 - 3(\eta_{21} - 3\eta_{03})^2] + (\eta_{03} - 3\eta_{21})(\eta_{03} + 3\eta_{21}) [(\eta_{30} + 3\eta_{12})^2 - 3(\eta_{12} - \eta_{30})^2] \quad (22)$$

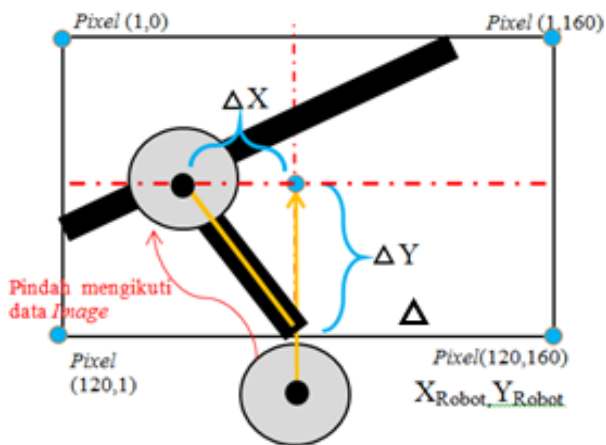
$$\theta_6 = (\eta_{20} + 3\eta_{02}) [(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{03} - 3\eta_{21}) \quad (23)$$

$$\theta_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + 3\eta_{12})(\eta_{30} + 3\eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2 (\eta_{30} + \eta_{12})(\eta_{21} + 3\eta_{03}) [(\eta_{03} - 3\eta_{21})^2 - 3(\eta_{30} + 3\eta_{12})^2] \quad (24)$$

#### C. Rancangan Navigasi

Rancangan navigasi yang digunakan adalah menggunakan metode koreksi koordinat baris dan kolom untuk mendeteksi adanya pertigaan atau tidak dan kemudian metode ini mengatur sudut kendali. Data Gambar dirubah ke 2-Dimensi *pixel* 160 x 120 untuk mendeteksi jalur.

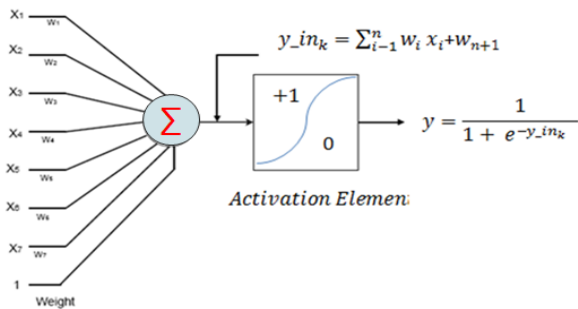
- 1) Rancangan Sistem Pendeteksian Pada Pertigaan Jalur seperti Gambar 5.



Gambar 5. Perpindahan Posisi Robot Ke Posisi Target

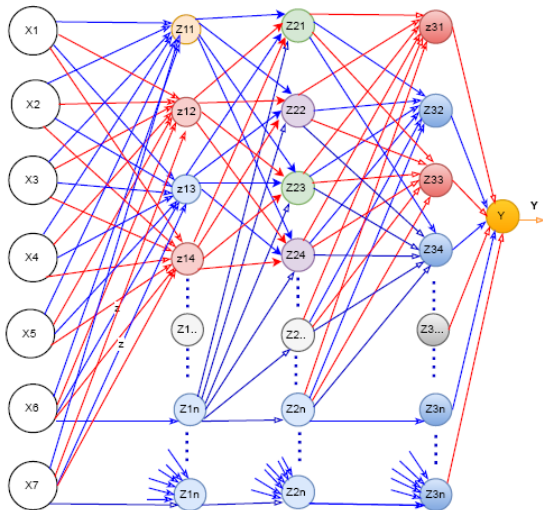
D. Struktur Jaringan Saraf Tiruan

Gambar 6 merupakan rancangan Jaringan Saraf Tiruan [3].



Gambar 6. Struktur Jaringan Tiruan

Pada aplikasinya direncanakan struktur jaringan saraf tiruan secara keseluruhan seperti pada Gambar 7 sebagai berikut.[4]



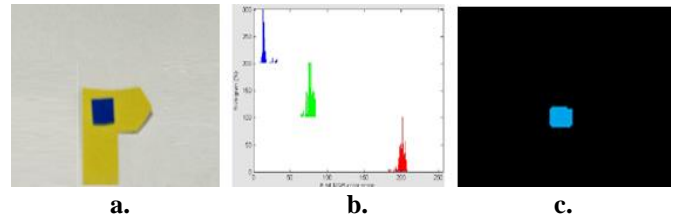
Gambar 7. Rancangan Jaringan Saraf Tiruan Secara Keseluruhan

Pada Gambar 7. Struktur jaringan saraf tiruan memiliki 7 input dan 50 node hidden Layer 1, 50 node hidden Layer 2 dan 10 node hidden layer 3 maka untuk perancangan jaringan saraf tiruan secara keseluruhan adalah seperti Gambar 7 berikut.

IV. IMPLEMENTASI DAN PENGUJIAN

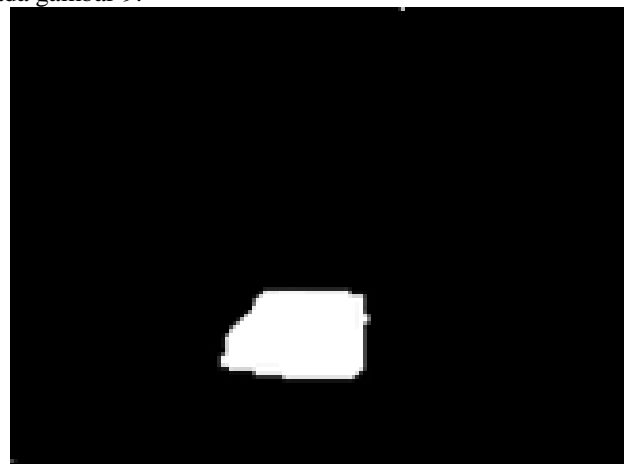
A. Training Pola

Proses analisa pola yang diambil pada gambar yang diambil seperti pada gambar 8. Untuk mendapatkan pola gambar dilakukan proses pemisahan background dengan Gambar bertujuan untuk hanya menganalisa bentuk gambar yang kita inginkan. [5]



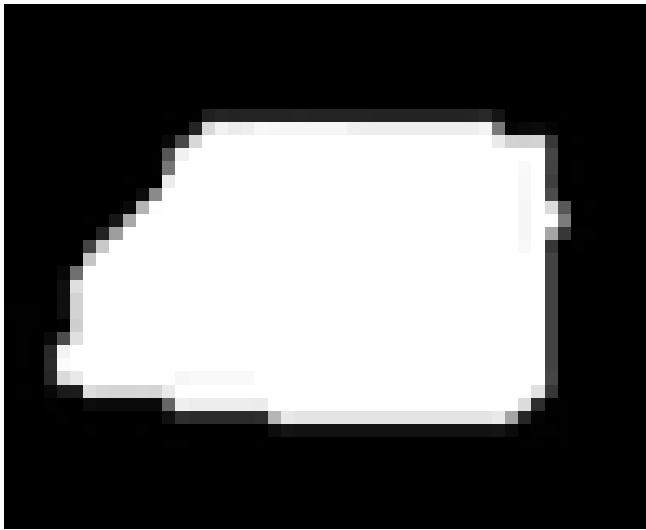
Gambar 8.1 (a) Gambar Asli (b) Pengambilan Warna RGB (c) Menampilkan Hanya Warna Biru

Kemudian mengubah gambar ke bentuk hitam putih seperti pada gambar 9.



Gambar 9. Gambar Obyek ditampilkan dan background dihitamkan

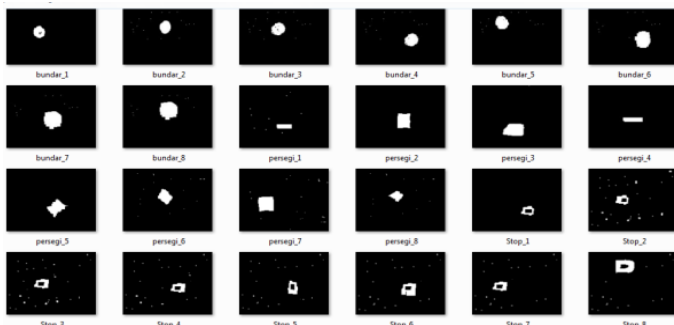
Pada pemrosesan gambar telah terpisahkan antara obyek dan background, maka gambar tersebut menjadi nilai input pengenalan pola untuk dijadikan navigasi. Kemudian melakukan proses crop gambar agar nilai benda semakin mudah dikenal dan cepat diakses. Pixel crop menjadi 46x46 pixel seperti pada Gambar 10. [2]



Gambar 10. Obyek Dalam Bentuk Data Gambar Yang Telah di-Crop

**B. Pengujian Navigasi**

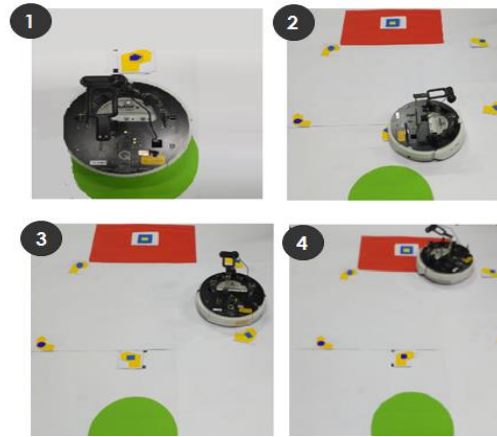
Pengujian navigasi mobile adalah pengujian terhadap kemampuan sistem yang diberikan pada robot untuk mendeteksi dalam menentukan navigasi. Untuk pengujian dimasukan beberapa data gambar, semakin banyak gambar yang dikenali dan memiliki karakteristik yang sama sama maka proses pengklasifikasian mampu dillakukan dengan baik dan cepat berikut data gambar yang akan diklasifikasi untuk data *training seperti* pada Gambar 11 merupan gambar-gambar yang akan dijadikan input untuk pengenalan pola. [2]



Gambar 11. Obyek-Obyek Yang Dijadikan Navigasi

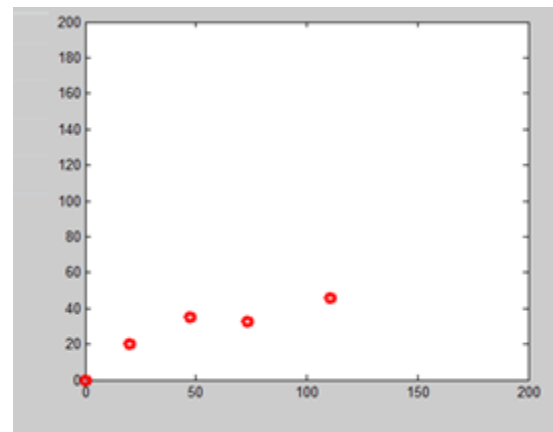
1) Kondisi Pertigaan Belok Kanan dan Robot Belok Kanan

Pengujian ketiga kondisi kendaraan berada di jalur belok kanan maka mobile robot akan bergerak belok ke kanan. Data terlihat pada Gambar 12.



Gambar 12. Kondisi Jalan Belok Kanan dan Robot Belok Kanan Kemudian Mengikuti Navigasi Selanjutnya

Untuk gambar sebenarnya seperti pada gambar 4.6 berikut. Hasil ditampilkan pada *Trajectory* diperoleh hasil seperti Gambar 13 berikut.

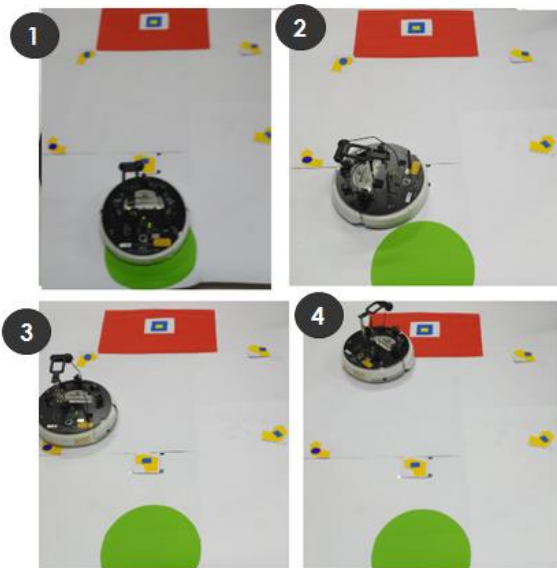


Gambar 13. Tampilan Trajectory Pertigaan Belok Kanan

Pada hasil tampilan *trajectory* miring bukan lurus 90 derajat karena titik X awal dan Y awal motor adalah 0,0. Maka jika robot jalan lurus artinya *trajectory* lurus menuju sudut 45 derajat menjauhi titik X,Y (0,0).

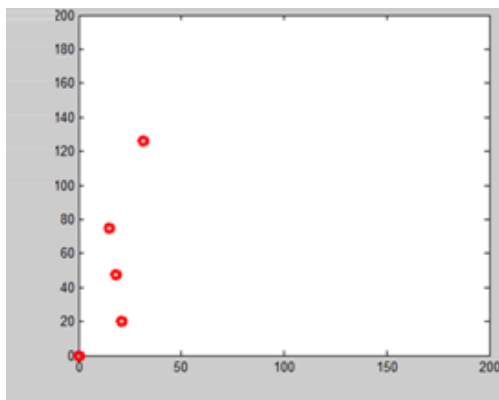
2) Kondisi Pertigaan Belok Kiri dan Robot Belok Kiri

Pengujian ketika mobile robot di jalur belok kiri dari posisi *mobile robot Qbot* maka robot akan berbeok ke kanan. Data terlihat pada Gambar 14.



Gambar 14. Kondisi Jalan Kiri dan Robot Belok Kiri

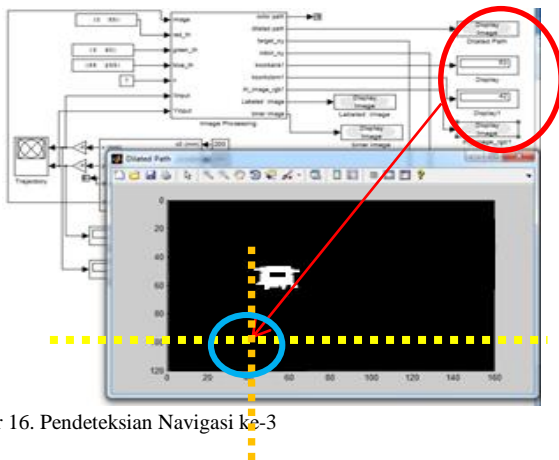
Pengujian kali untuk menampilkan mobile robot di *track* yang dilewati berupa tampilan *trajectory* seperti pada gambar 15 sebagai berikut [1].



Gambar 15. Trajectory yang ditampilkan Pertigaan Belok Kiri

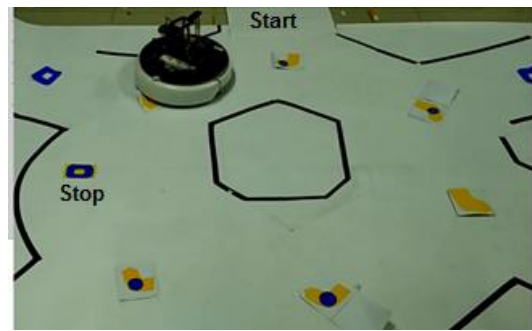
3) Pengujian Pada Pertigaan belok kiri dan saat melewati navigasi ke dua

Pada pengujian dan pengenalan pertigaan pada mobile robot Qbot menggunakan analisa piksel dalam bentuk matrik seperti pada gambar 16.



Gambar 16. Pendeteksian Navigasi ke-3

Gambar nyata dari gambar 16 adalah gambar 17.



Gambar 17. Tampilan jalur dilihat dari atas robot.

### V. KESIMPULAN

Berdasarkan data hasil pengujian dalam pembuatan studi ini, dapat ditarik kesimpulan sebagai berikut:

Sistem navigasi yang diterapkan pada mobile robot dengan jalur yang ditentukan merupakan pengambilan titik target yang akan dituju, disetiap ada kondisi maka mempertimbangkan arah tujuan. Sistem ini dihadapkan beberapa pilihan seperti pertigaan, perempatan dan peran navigasi seperti adanya obyek berupa penunjuk arah atau bentuk jalur sehingga *mobile robot* mampu mengikuti jalur tersebut.

Pada hasil tampilan *trajectory* terdapat perbedaan kordinat dengan tampilan pixel, yaitu koordinat pixel X,Y(1,1) beada pada kiri atas dimensi dua sedangkan pada koordinat medan motor untuk nilai koordinat X,Y (0,0) berada pada kiri bawah sehingga mempengaruhi dan gerak lurus menjadi menjauhi titik awal 0,0 ke arah susut 45 derajat sedangkan diambil data kamera lurus 90 derajat dalam bentuk pixel dibanding dari posisi koordinat robot.

### DAFTAR PUSTAKA

- [1] *User Manual Quanser Qbot*. 2012.
- [2] A. K. and A. Susanto, *Pengolahan Citra Teori dan Aplikasi, 1st ed*. Yogyakarta: Andi, 2013.
- [3] C. G. R. and E. W. Richard, *Digital Image Processing*. New Jersey: Tom Robbins, 2001.
- [4] Suyanto, *Artificial Intelligence*. Bandung: Informatika Bandung, 2014.
- [5] D. P. Tian, "A Review on Image Feature Extraction and Representation Techniques," *Int. J. Multimed. Ubiquitous Eng.*, vol. 8, p. 12, 2013.