

FIRE AND WALL DETECTION IN FIRE FIGHTING ROBOT USING SEGMENTATION AND DIFFERENTIAL OF INTEGRAL PROJECTION METHOD

ABSTRACT

Fire fighting robot is like a fireman working simulation. Robot can move automatically through the way and enter the room to find the source of fire. Robot has a task to find and extinguish the fire as the rule of 2006 Indonesia Intelligent Robot Contest. Robot has some abilities, first robot can avoid the obstacle that represented by the wall, and the second robot can find and extinguish the fire that represented by the candle. Camera used for wall detecting sensor and fire detector. Detecting the wall, used integral projection and differential method, and detecting the fire, used segmentation method. The camera sending data to computer to be processed and then the computer sending data to microcontroller in order to move motors in robot. The success of wall and fire detection is depend on lightening level when the object was captured.

Key word: integral projection, differentiation, and segmentation.

Deteksi Api dan Pemetaan pada Robot Pemadam Api dengan Metode Segmentasi dan Diferensial - Integral Proyeksi

Adnan Rachmat Anom Besari, Riyanto Sigit

Jurusan Teknologi Informasi - Politeknik Elektronika Negeri Surabaya, Institut Teknologi Sepuluh Nopember
Kampus ITS Keputih Sukolilo Surabaya 60111 Telp. 031-5947280, E-mail: moozle554@gmail.com

ABSTRAK

Robot pemadam api merupakan sebuah simulasi dari petugas pemadam api. Robot bergerak secara otomatis menyusuri jalan dan memasuki ruangan untuk mencari sumber api. Robot mempunyai tugas menemukan dan mematikan api sebagaimana peraturan Kontes Robot Cerdas Indonesia (KRCI) 2006. Robot mempunyai beberapa kemampuan antara lain dapat menghindari halangan yang diwakili oleh dinding, mendeteksi dan mematikan sumber api yang diwakili oleh api pada lilin. Digunakan kamera sebagai sensor pendeteksi dinding sekaligus pendeteksi ada atau tidaknya api. Untuk mendeteksi dinding digunakan metode integral proyeksi dan diferensial sedangkan untuk mendeteksi api digunakan metode segmentasi. Kamera mengirimkan data ke komputer untuk diproses selanjutnya komputer mengirimkan data ke mikrokontroler untuk menggerakkan motor pada robot. Keberhasilan deteksi dinding dan deteksi api banyak ditentukan oleh tingkat pencahayaan saat dilakukan pengambilan objek.

Kata kunci : integral proyeksi, deferensial dan segmentasi.

1. PENDAHULUAN

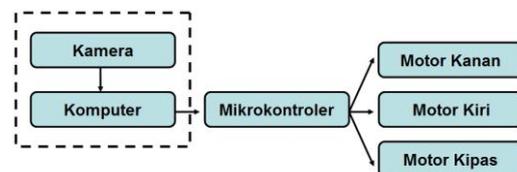
Dewasa ini, perkembangan ilmu pengetahuan dan teknologi sangat pesat. Salah satu akibatnya adalah digantikannya peran manusia dalam mengerjakan suatu pekerjaan oleh mesin yang kita lebih sering menyebutnya dengan istilah robot. Tujuan dibuatnya robot adalah untuk mengatasi permasalahan yang dihadapi manusia. Salah satu masalah manusia yang paling sering dihadapi adalah masalah keamanan dalam pekerjaan. Dalam hal ini, fungsi robot sebagai pengganti peran manusia dalam menjalankan pekerjaan yang sangat rawan bagi keselamatan manusia.

Proyek akhir ini mengambil contoh permasalahan kebakaran rumah untuk disimulasikan karena permasalahan ini merupakan permasalahan sehari-hari yang paling sering dihadapi saat ini. Fungsi robot disini adalah mengantisipasi kebakaran secara dini dengan mematikan sumber api. Dalam simulasi proyek akhir ini, sumber api diganti dengan lilin yang menyala, sedangkan ruangan-ruangan di rumah diganti dengan labirin atau ruangan bersekat. Simulasi di atas diadopsi dari Kontes Robot Cerdas Indonesia 2006.

Pembuatan robot cerdas pemadam api selama ini kebanyakan menggunakan sensor analog untuk antarmuka dengan dunia luar seperti sensor ultrasonik untuk menghindari halangan dan sensor cahaya untuk mencari sumber api. Proyek akhir ini berupaya mengembangkan sensor kamera agar dapat menggantikan fungsi sensor-sensor analog di atas dengan menggunakan komputer sebagai pemroses data.

2. DISAIN SISTEM

Secara garis besar sistem yang dibangun ditunjukkan pada blok diagram sebagai berikut :



Gambar 1. Blok diagram rancangan sistem

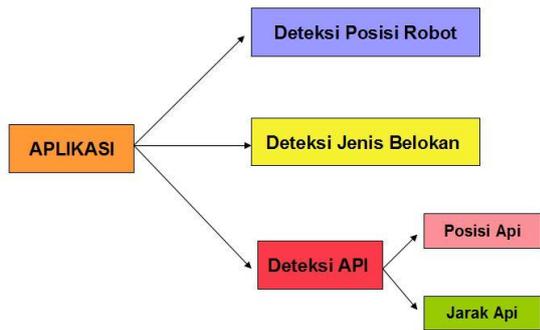
Pembahasan pada makalah ini terbatas pada proses pengambilan input (gambar) oleh kamera, yang pemrosesannya dilakukan di dalam komputer, output yang dihasilkan ditampilkan dalam bentuk simulasi yang ada pada aplikasi yang dibuat.

Algoritma program yang dibuat seperti digambarkan pada diagram alur di bawah ini :



Gambar 2. Diagram alur program aplikasi

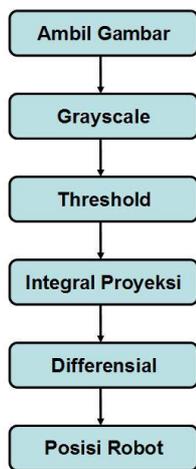
Dari alur diatas, pembuatan sistem dibagi kedalam 3 bagian besar, seperti pada bagan di bawah ini :



Gambar 3. Pembagian sistem kerja aplikasi

2.1 Deteksi Posisi Robot

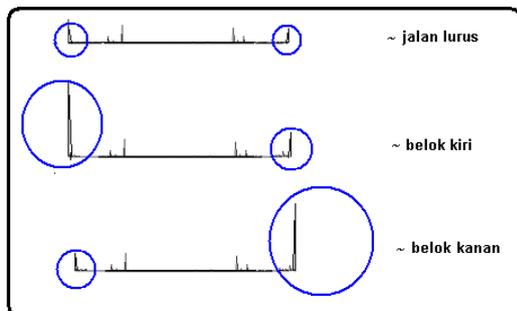
Untuk mendapatkan posisi robot secara umum dapat dilihat pada blok diagram berikut ini:



Gambar 4. Blok diagram deteksi posisi robot

Deteksi posisi robot digunakan oleh robot agar dapat menghindari dari dinding, deteksi posisi robot menentukan kapan robot harus berjalan lurus, serong kiri atau serong kanan agar tidak menyentuh atau menabrak dinding.

Secara umum posisi robot dapat dibagi kedalam 3 kondisi, yaitu : lurus, miring ke kiri, dan miring ke kanan. Masing-masing kondisi tersebut memiliki perbedaan jika dilihat dari hasil diferensial integral proyeksinya.

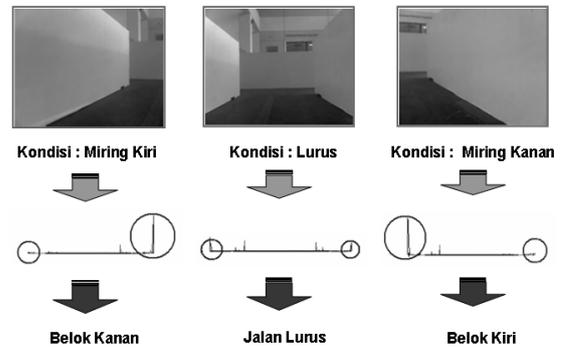


Gambar 5. Diferensial untuk deteksi posisi robot
Deteksi posisi robot dilakukan dengan mengambil dari hasil deferensial setelah sebelumnya dilakukan

integral proyeksi. Data yang diambil adalah array paling awal (ke-0) dan array paling akhir (ke-n). Kedua data tadi dibandingkan berdasarkan rumusan

Jalan lurus	: Array ke 0 = Array ke-n
Belok kiri	: Array ke 0 > Array ke-n
Belok kanan	: Array ke 0 < Array ke-n(1)

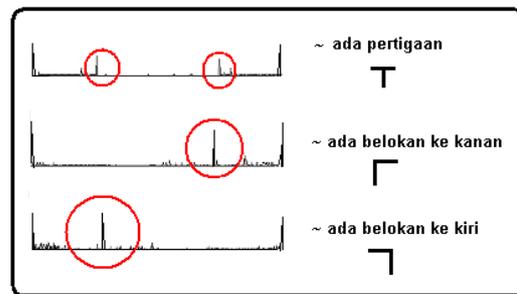
berikut :



Gambar 6. Contoh deteksi posisi robot

2.2 Deteksi Jenis Belokan

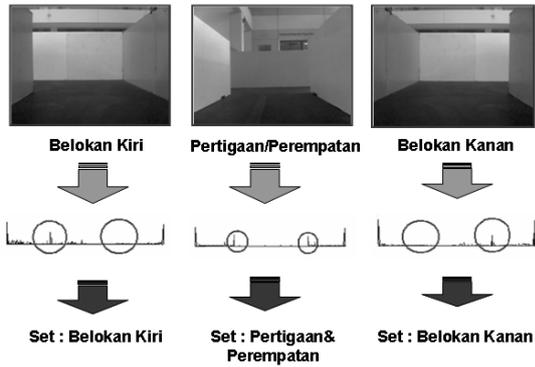
Deteksi jenis belokan digunakan agar dapat mengetahui jenis belokan yang ada di depan robot. Secara umum ada 3 jenis belokan, yaitu : belokan kanan, belokan kiri, dan pertigaan atau perempatan. Masing-masing jenis belokan memiliki perbedaan jika dilihat dari hasil diferensial integral proyeksinya.



Gambar 7. Diferensial untuk deteksi jenis belokan

Deteksi jenis belokan juga dilakukan dengan mengambil hasil deferensial integral proyeksinya. Namun, data yang diambil adalah array tertinggi pada kuadran kiri tetapi bukan array paling awal dan array tertinggi pada kuadran kanan tetapi bukan array paling akhir. Kedua data tadi dibandingkan berdasarkan rumusan berikut :

Pertigaan	: kuadran kanan = kuadran kiri
Belokan kanan	: kuadran kanan > kuadran kiri
Belokan kiri	: kuadran kanan < kuadran kiri(2)

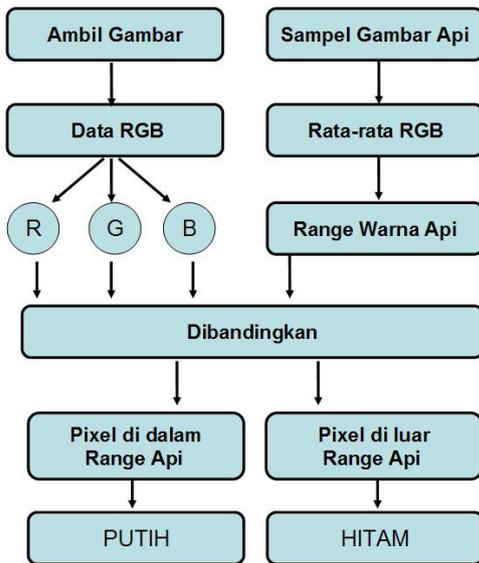


Gambar 8. Contoh deteksi jenis belokan

2.3 Deteksi Api

2.3.1 Segmentasi Warna Api

Segmentasi warna api secara umum dijelaskan pada blok diagram berikut ini.



Gambar 9. Blok diagram segmentasi api

Terlebih dulu diperlukan sampel warna api. Kemudian didapatkan rata-rata dari warna api.



Gambar 10. Sampel api dan rata-rata RGB-nya

Setelah didapatkan rata-rata dari warna api, ditentukan range atau daerah nilai warna api. Range warna api ditentukan dengan menentukan batas bawah (x1) dan batas atas (x2).

$$x1 < \text{nilai_api} < x2 \dots\dots\dots(3)$$

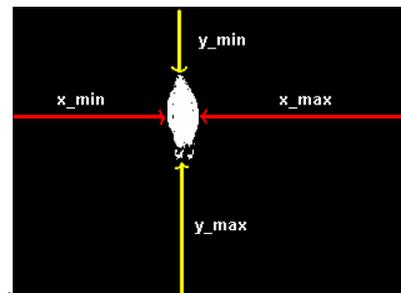
Kemudian setiap pixel pada gambar dibandingkan dengan range warna api. Jika nilai pixel berada di dalam range warna api maka nilai pixel diubah ke nilai maksimal (warna putih). Jika nilai pixel berada di luar range warna api maka nilai pixel diubah ke nilai minimal (warna hitam). Dengan demikian akan terlihat gambar hasil segmentasi warna api yang memisahkan objek api dengan objek lainnya. Api akan berwarna putih dan selain api akan berwarna hitam.



Gambar 11. Hasil segmentasi warna api.

2.3.2 Mencari Posisi Api

Posisi api digunakan untuk menjaga letak api agar tetap berada di tengah-tengah bidang gambar, hal ini dimaksudkan agar dapat menjaga posisi robot berada tepat di depan api. Penentuan daerah hasil segmentasi warna api dilakukan dengan mencari pixel berwarna putih pertama pada 4 sisi bidang gambar. Sehingga didapatkan nilai koordinat x,y terkecil dan koordinat x,y terbesar.

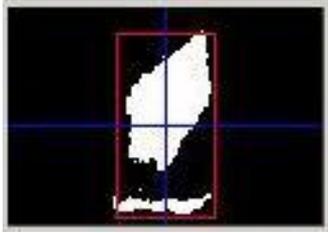


Gambar 12. Pencarian posisi api

Setelah posisi api didapatkan, dilakukan perhitungan koordinat pusat dari daerah luasan api dengan rumusan :

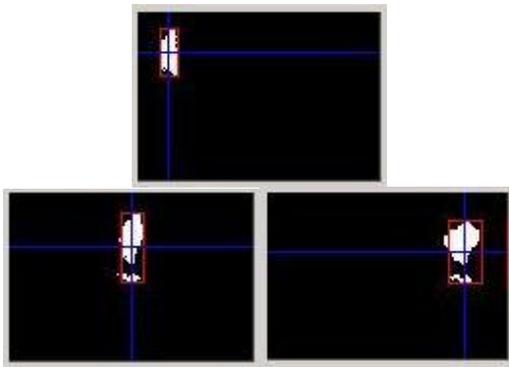
$$\begin{aligned} X \text{ pusat} &= X \text{ terkecil} + (X \text{ terbesar} - X \text{ terkecil}) / 2 \\ Y \text{ pusat} &= Y \text{ terkecil} + (Y \text{ terbesar} - Y \text{ terkecil}) / 2 \dots\dots\dots(4) \end{aligned}$$

Pusat koordinat dari daerah luasan api digunakan untuk menentukan fokus api. Fokus api disini dimaksudkan untuk menempatkan posisi api agar selalu berada di tengah-tengah gambar (persis di depan kamera pada robot).



Gambar 13. Penentuan pusat koordinat api

Jika api berada di posisi kiri maka robot akan bergerak ke kiri, dan sebaliknya jika api berada di posisi kanan maka robot akan bergerak ke kanan. Robot akan bergerak lurus jika posisi api berada tepat di tengah gambar.



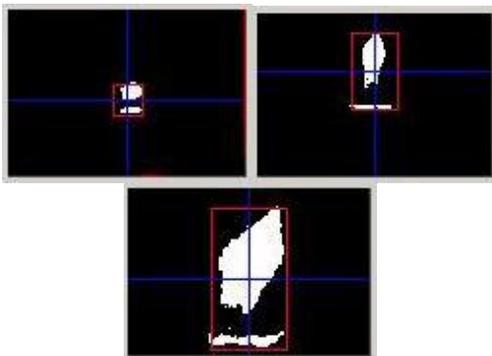
Gambar 14. Contoh deteksi posisi api

2.3.3 Menentukan Jarak Api dengan Robot

Jarak api terhadap robot digunakan untuk mengetahui posisi robot sudah dekat dengan api atau belum sehingga dapat diambil keputusan kapan saatnya mematikan api.

Jarak api terhadap robot didapatkan dengan cara menghitung daerah luasan api. Sebelumnya harus ditentukan range luasan api untuk kategori sedang.

Jauh	: $L_{api} < range_sedang$
Sedang	: $L_{api} = range_sedang$
Dekat	: $L_{api} > range_sedang$(5)



Gambar 15. Contoh deteksi jarak api

3. PENGUJIAN DAN ANALISA

Pengujian dilakukan sebanyak 30 kali pada masing-masing deteksi, yang dibagi ke dalam 3 tingkat pencahayaan.

1. Terang
2. Redup
3. Gelap dengan lampu kamera

3.1 Pengujian Deteksi Posisi Robot



Gambar 16. Tampilan deteksi jenis belokan

Dari hasil pengujian **deteksi posisi robot** didapatkan data sebagai berikut :

- Cahaya terang : 90 % benar.
- Cahaya redup : 70 % benar.
- Cahaya gelap + lampu kamera : 40 % benar.
- **Rata-rata : 66 % benar**

Analisa :

- Hasil yang maksimal didapatkan dengan menggunakan pencahayaan yang terang dibandingkan dengan cahaya yang redup ataupun gelap dengan lampu kamera.
- Bayangan dari dinding akan sangat mengganggu proses threshold, sehingga diperlukan tingkat pencahayaan yang homogen, maksudnya seluruh sisi objek mendapatkan banyak cahaya yang sama. Jika objek yang didapatkan adalah dinding saja atau lantai saja, data tidak dapat di proses karena nilai threshold yang didapat maksimal atau minimal.
- Sumber cahaya yang terlalu dekat dengan objek akan mengakibatkan warna objek tidak dapat didefinisikan dengan jelas, karena warnanya berubah menjadi putih. Hal ini sangat mengganggu ketika objek yang dimaksud adalah lantai, sehingga mengakibatkan nilai threshold yang didapat tidak valid.

3.2 Pengujian Deteksi Jenis Belokan



Gambar 17. Tampilan deteksi jenis belokan

Dari hasil pengujian **deteksi jenis belokan** didapatkan data sebagai berikut :

- Cahaya terang : 80 % benar.
- Cahaya redup : 70 % benar.
- Cahaya gelap + lampu kamera : 30 % benar.
- **Rata-rata : 60 % benar**

Analisa :

- Hasil yang maksimal didapatkan dengan menggunakan pencahayaan yang terang dibandingkan dengan cahaya yang redup ataupun gelap dengan lampu kamera.
- Bayangan dari dinding akan sangat mengganggu proses threshold, sehingga diperlukan tingkat pencahayaan yang homogen, maksudnya seluruh sisi objek mendapatkan banyak cahaya yang sama.
- Sumber cahaya yang terlalu dekat dengan objek akan mengakibatkan warna objek tidak dapat didefinisikan dengan jelas, karena warnanya berubah menjadi putih. Hal ini sangat mengganggu ketika objek yang dimaksud adalah lantai, sehingga mengakibatkan nilai threshold yang didapat tidak valid.
- Letak kamera yang paling baik untuk menentukan jenis belokan adalah pada posisi di tengah, diantara dinding dalam keadaan lurus.

3.3 Pengujian Deteksi Api



Gambar 18. Tampilan deteksi api

Dari hasil pengujian **deteksi posisi api** didapatkan data sebagai berikut :

- Cahaya terang : 100 % benar.
- Cahaya redup : 100 % benar.
- Cahaya gelap + lampu kamera : 60 % benar.
- **Rata-rata : 86 % benar**

Dari hasil pengujian **deteksi jarak api** didapatkan data sebagai berikut :

- Cahaya terang : 100 % benar.
- Cahaya redup : 100 % benar.
- Cahaya gelap + lampu kamera : 70 % benar.
- **Rata-rata : 90 % benar**

Analisa :

- Untuk deteksi api secara umum tidak berpengaruh pada tinggi atau rendahnya tingkat pencahayaan, karena api sendiri merupakan objek yang memancarkan cahaya sendiri. Sehingga tidak didapatkan hasil yang berbeda dalam pencahayaan terang ataupun redup.
- Jika digunakan pencahayaan dari lampu kamera, didapatkan hasil yang kurang maksimal. Cahaya yang dipancarkan oleh lampu kamera berpadu dengan cahaya api sehingga kamera menangkap cahaya api berpadu. Posisi api dan jarak api tidak dapat didapatkan dengan maksimal ketika digunakan lampu kamera.

Dari hasil pengujian diatas di dapatkan data keseluruhan dalam tabel sebagai berikut :

Tabel 1: Data hasil percobaan

Deteksi	Pencahayaannya			Rata -Rata
	Terang	Redup	Gelap+Lampu	
Posisi Robot	90%	70%	40%	66%
Jenis Belokan	80%	70%	30%	60%
Posisi Api	100%	100%	60%	86%
Jarak Api	100%	100%	70%	90%

4. KESIMPULAN

Setelah dilakukan pengujian dan analisisnya, dapat diambil beberapa kesimpulan antara lain :

1. Tingkat keberhasilan deteksi posisi robot sangat ditentukan oleh objek yang diambil kamera yaitu lantai dan dinding. Jika salah satu tidak terpenuhi (misal dinding saja atau lantai saja) posisi robot tidak dapat dideteksi.
2. Tingkat keberhasilan deteksi jenis belokan sangat ditentukan oleh posisi robot. Posisi yang paling baik untuk mendeteksi jenis belokan adalah posisi lurus terhadap jalan (robot berada tepat di tengah).
3. Tingkat keberhasilan deteksi api (posisi api da jarak api) sangat ditentukan oleh rata-rata sampel gambar api dan pemberian nilai toleransi pada warna api saat proses segmentasi api.

5. DAFTAR PUSTAKA

- Sigit, Riyanto., *Teori Pengolahan Citra Digital*, Surabaya : Politeknik Elektronika Negeri Surabaya – ITS, 2002.
- Sigit, Riyanto., *Sistem Pengenalan Wajah Secara Real Time*, Surabaya : Institut Teknologi Sepuluh November Surabaya, 2005.
- Basuki, Achmad., *Pengolahan Citra Digital menggunakan Visual Basic*, Yogyakarta : Graha Ilmu, 2005
- Munir, Rinaldi. *Pengolahan Citra Digital dengan Pendekatan Algoritmik*, Bandung : Informatika. 2004.
- Ardila, Fernando., *Robot Pemadam Api PENSA mobile*, Surabaya : Politeknik Elektronika Negeri Surabaya – ITS, 2006.
- Gonzalez, Rafael C., Woods, Richard C., *Digital Image Processing*, Prentice Hall, 2nd edition, 2002
- Kadir, Abdul. *Pemrograman Visual C++*, Yogyakarta : Andi Offset, 2004.
- _____, *DirectX Documentation for C++ (StillCap)*, Microsoft DirectX Help, 2004.
- _____, *Kontes Robot Cerdas Indonesia 2006 : Robot Cerdas Pemadam Api*, Jakarta : DIKTI. 2005.