

Makalah Seminar Tugas Akhir

KONTROL KURSI RODA CERDAS MENGGUNAKAN PERGERAKAN KEPALA

Dwi Afiat Abrianto^[1], Iwan Setiawan, ST, MT^[2], Achmad Hidayatno, ST, MT^[2]
Jurusan Teknik Elektro, Fakultas Teknik, Universitas Diponegoro
Jln. Prof. Sudharto, Tembalang, Semarang, Indonesia

ABSTRAK

A wheelchair is a chair with wheels, designed to be a replacement for walking. When some individuals with the limitation of movement (disability) are helped by the usage of standard wheelchair, some other disabled people found difficulties or almost impossible to operate this standard wheelchair. To facilitate this group of disabled individuals, researchers make use the technology which initially developed for mobile robot to create a smart wheelchair, which then dramatically reduces the need of physical gestures, observabilities and skills needed to operate the standard wheelchair.

The purpose of this final project is to implement the control system for a smart wheelchair in which could be controlled using head movement that is captured by the webcam, as well as to avoid any obstacles/obstructions with minimum fault. The control system for this smart wheelchair adopts Visual C# 2008 and EmguCV software which act as a library to indentify the head movement. The head movement then will be sent and implemented into a microcontroller system that organize the movement of the motor in the wheelchair. The ultrasonic sensor is used to detect the presence of any obstacles/obstructions.

The result of this study concluded that the position and the direction of the ultrasonic sensor is having an influence with the detection of the objects in front of the wheelchair. This smart wheelchair, in addition to joystick, could also be controlled by using head movements. The movement of the wheelchair is in accordance with the movement of the user's head captured by the webcam. This smart wheelchair is able to detect the presence of any obstacles/obstructions and route to avoid them or to stop in front of them with 45 cm measured in front of the wheelchair.

Keywords : Smart Wheel Chair, Head Movement, Webcam, Visual C#, EmguCV, Ultrasonic Sensor

I PENDAHULUAN

1.1 Latar Belakang

Kursi roda adalah alat bantu yang digunakan oleh orang yang mengalami kesulitan berjalan menggunakan kaki, baik dikarenakan oleh penyakit, cedera, maupun cacat. Alat ini bisa digerakan dengan didorong oleh pihak lain, digerakan dengan menggunakan tangan, atau dengan menggunakan mesin otomatis.

Seiring dengan kemajuan teknologi, beberapa peneliti mempergunakan teknologi yang mula-mula dikembangkan untuk *mobile robot* untuk menciptakan kursi roda cerdas (*smart wheelchair*), yang dapat mengurangi kebutuhan fisik, keterampilan, dan keterampilan yang diperlukan untuk mengoperasikan kursi roda standar.

Pada penelitian sebelumnya, kursi roda cerdas sudah mempunyai kemampuan bernavigasi secara *semiautonomous* dengan menggunakan kontrol gerakan *joystick*. Namun hal ini bagi sebagian pengguna yang memiliki keterbatasan gerak tangan akan mengalami kesulitan untuk mengoperasikannya. Untuk mengatasi hal tersebut, perlu adanya inovasi dalam cara menggerakkan kursi roda selain menggunakan *joystick*, salah satunya dengan menggunakan pergerakan kepala (*head movement*).

1.2 Tujuan

Tujuan dari tugas akhir ini adalah mengembangkan sistem pengontrolan kursi roda cerdas dengan menggunakan pergerakan kepala (*head movement*) sehingga pemakai kursi roda cerdas yang memiliki keterbatasan gerak tangan dapat mengoperasikannya.

1.3 Pembatasan Masalah

Pembatasan masalah pada Tugas Akhir yaitu:

1. Ruang gerak kursi roda berada pada daerah yang datar sehingga kursi roda dapat bergerak dengan baik.
2. Mikrokontroler yang digunakan adalah mikrokontroler AVR ATmega128.
3. Bahasa pemrograman yang digunakan adalah bahasa C yang diadaptasikan pada *software Code Vision AVR*.
4. Sensor yang digunakan adalah sensor ultrasonik yang diproduksi oleh Parallax (Sensor PING)))TM Ultrasonik *Range Finder*).
5. Pengendalian kecepatan motor DC dilakukan melalui PWM.
6. Komunikasi antara laptop dengan mikrokontroler yang digunakan adalah komunikasi serial.

¹ Mahasiswa Jurusan Teknik Elektro UNDIP

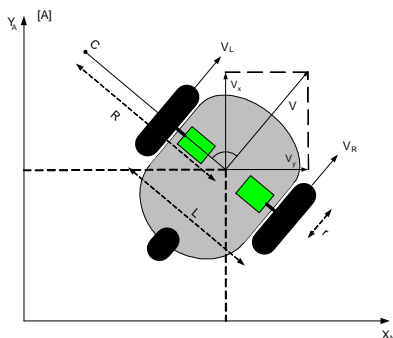
² Staf Pengajar Jurusan Teknik Elektro UNDIP

7. *Webcam* yang digunakan adalah *webcam Prolink PCC5020*.
8. Jumlah pemakai yang tertangkap oleh kamera hanya 1 orang (1 kepala).
9. Pembuatan program antarmuka kursi roda cerdas pada komputer menggunakan fungsi-fungsi dari *library* yang sudah disediakan *Visual Studio 2008* dan *EmguCV* dengan menggunakan bahasa pemrograman *C#*.
10. Tidak membahas proses pembuatan data *training haarcascade_mcs_eyepair_big.xml*.

II DASAR TEORI

2.1 Kinematika Kursi Roda Penggerak Differensial

Kursi roda tipe penggerak diferensial memiliki 2 buah roda penggerak yang terpisah (kanan dan kiri). Kedua roda ini digerakkan oleh motor DC yang ditempatkan pada satu sumbu secara terpisah. Sehingga kedua roda ini berfungsi sebagai penggerak sekaligus sebagai kemudi kursi roda. Sebagai penyeimbang umumnya kursi roda ini dilengkapi juga dengan satu atau dua buah roda *castor* yang ditempatkan dibagian belakang kursi roda tersebut



Gambar 1 Posisi dan Orientasi kursi roda dalam Sistem Koordinat Cartesian.

Secara matematis kecepatan linier dan kecepatan angular kursi roda dirumuskan sesuai persamaan (2.1).

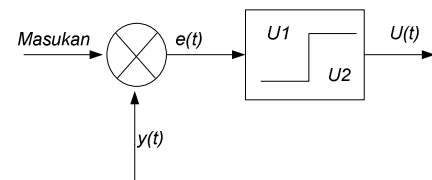
$$\begin{bmatrix} V(t) \\ \omega(t) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ \frac{1}{2} & \frac{1}{2} \\ \frac{1}{L} & -\frac{1}{L} \end{bmatrix} \begin{bmatrix} V_R(t) \\ V_L(t) \end{bmatrix} \quad (2.1)$$

Persamaan (2.1) di atas pada dasarnya memperlihatkan relasi kinematika langsung antara kecepatan linier roda-roda kursi roda terhadap kecepatan linier dan angular kursi rodanya, sedangkan Persamaan (2.2) berikut memperlihatkan relasi sebaliknya (Kinematika Bcalik).

$$\begin{bmatrix} V_R(t) \\ V_L(t) \end{bmatrix} = \begin{bmatrix} 1 & \frac{L}{2} \\ 1 & -\frac{L}{2} \end{bmatrix} \begin{bmatrix} V(t) \\ \omega(t) \end{bmatrix} \quad (2.2)$$

2.2 Kontrol On-Off

Salah satu metode pengendalian *plant* secara otomatis adalah metode kontrol *On-Off*. Pada sistem kontrol *on-off*, elemen pembangkit hanya memiliki dua posisi tertentu yaitu *on* dan *off*. Kontrol *on-off* memiliki karakteristik sinyal keluaran dari kontroler $u(t)$ tetap pada salah satu nilai maksimum atau minimum tergantung dari sinyal pembangkit kesalahan positif atau negatif. Diagram blok kontroler *on-off* yang memiliki masukan $e(t)$ dan keluaran $u(t)$, ditunjukkan pada Gambar 2.2.



Gambar 2 Blok kontrol On-Off

Aksi kontrol *On-Off* ditunjukkan pada persamaan dibawah ini:

$$U(t) = \begin{cases} U1, & e(t) > 0 \\ U2, & e(t) < 0 \end{cases} \quad (2.3)$$

Pada persamaan 2.3 $U1$ dan $U2$ adalah konstanta. Nilai minimum $U2$ dapat sebesar nol atau $-U1$. Pada sistem kontrol *lup* tertutup (*close loop*), sinyal $e(t)$ merupakan sinyal kesalahan (*error*) dari selisih antara sinyal *input* dengan sinyal umpan balik.

2.3 Computer Vision

Computer vision merupakan proses otomatis yang mengintegrasikan sejumlah besar proses untuk persepsi visual, seperti akuisisi citra, pengolahan citra, klasifikasi, pengenalan (*recognition*), dan membuat keputusan. *Computer vision* terdiri dari teknik-teknik untuk mengestimasi ciri-ciri objek di dalam citra, pengukuran ciri yang berkaitan dengan geometri objek, dan menginterpretasi informasi geometri tersebut.

Proses-proses di dalam *computer vision* dapat dibagi menjadi tiga aktivitas, yaitu:

1. Memperoleh atau mengakuisisi citra digital.
2. Melakukan teknik komputasi untuk memproses atau memodifikasi data citra (operasi-operasi pengolahan citra).
3. Menganalisis dan menginterpretasi citra dan menggunakan hasil pemrosesan untuk tujuan

tertentu, misalnya memandu robot, mengontrol peralatan, memantau proses manufaktur, dan lain-lain

2.4 EmguCV

EmguCV merupakan sebuah .NET *cross platform* yang berguna agar *image processing library* yang ada pada OpenCV dapat digunakan dalam berbagai *platform* bahasa pemrograman seperti C#, VB, IronPhyton, dan lain-lain. OpenCV adalah singkatan dari *Open Computer Vision*, yaitu *library-library open source* yang dikembangkan oleh Intel Corporation yang dikhususkan untuk melakukan *image processing*. Bahasa pemrograman yang dipakai dalam tugas akhir ini adalah C#.

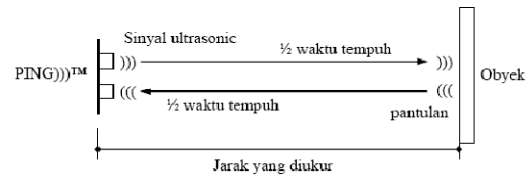
2.5 Deteksi Objek (*Object Detection*)

Salah satu metode deteksi objek yang ada pada openCV adalah metode *Haar Classifier* yaitu metode yang dipublikasikan oleh Paul Viola dan Michael Jones pada tahun 2001. Metode *Haar Classifier* merupakan sebuah *boosted rejection cascade*, yang akan membuang data training negatif sehingga didapat suatu keputusan untuk menentukan data positif. *Haar Classifier* merupakan metode *supervised learning* (pembelajaran terawasi), yaitu algoritma pembelajaran yang memerlukan target *output* yang diharapkan untuk diketahui sebelumnya dalam proses pembelajarannya. Dalam proses pembelajaran, *Haar Classifier* membutuhkan data training untuk dapat mendeteksi objek-objek tertentu yaitu data positif (objek yang akan dideteksi) dan data negatif (bukan objek yang akan dideteksi).

Pada openCV telah disediakan beberapa *library default* deteksi objek seperti deteksi wajah, mulut, hidung, mata, dan tubuh. Pada tugas akhir deteksi objek yang dipakai adalah deteksi sepasang mata yang menggunakan data training parojos `haarcascade_mcs_eyepair_big.xml`.

2.6 Sensor Jarak

Sensor yang digunakan adalah “PING)))TM *Ultrasonic Range Finder*”, buatan Parallax. Agar sensor ini dapat digunakan untuk mengukur jarak dibutuhkan sebuah mikrokontroler untuk mengukur waktu tempuh sinyal ultrasonik dari sensor, memantul pada benda yang akan diukur, dan diterima kembali oleh sensor. Ilustrasi cara kerja sensor ditunjukkan pada Gambar 4.



Gambar 4 Ilustrasi cara kerja PING)))TM.

III PERANCANGAN SISTEM

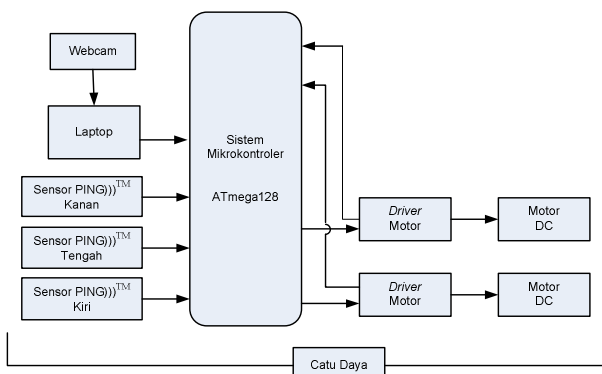
3.1 Perancangan Perangkat Keras

Perangkat keras dari sistem kursi roda cerdas yang akan dibangun meliputi sistem minimum mikrokontroler Atmega128, sensor PING)))TM, *webcam*, *driver* motor DC, dan laptop. Secara umum perancangan sistem yang akan dibuat dapat dilihat pada Gambar 5.

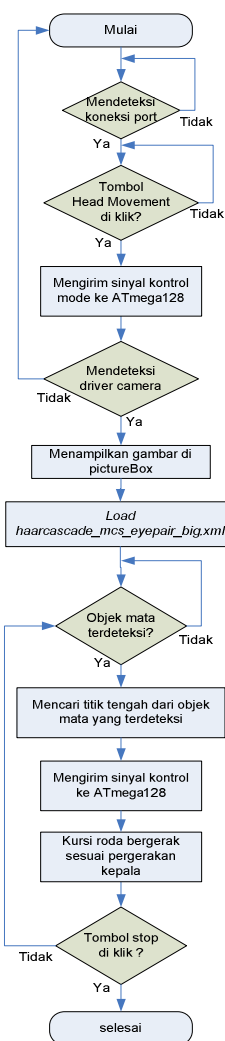
Pada Tiap-tiap bagian dari diagram blok sistem di atas dapat dijelaskan sebagai berikut:

1. Sensor jarak PING)))TM merupakan sensor yang akan mendeteksi jarak kursi roda terhadap halangan/rintangan yang ada di depannya. Keluaran sensor ini berupa tegangan digital, yaitu memiliki logika 1 saat transmitter memancarkan gelombang ultrasonik dan memiliki logika 0 saat receiver menerima gelombang ultrasonik yang telah dipancarkan.
2. *Webcam* digunakan sebagai media perekam video yang digunakan untuk mendapatkan koordinat atau posisi mata pengendali kursi roda cerdas. *Webcam* dipasang di depan pengendali kursi roda.
3. Laptop digunakan sebagai pusat pengolahan citra dan pengiriman data serial ke mikrokontroler.
4. Mikrokontroler ATmega128 berfungsi untuk melaksanakan tugas-tugas yang meliputi mengatur kecepatan motor, pembacaan sensor ping, dan penerimaan data serial sinyal kontrol posisi mata dari laptop.
5. Driver motor berfungsi untuk menjalankan motor DC. Driver ini menerima masukan PWM yang berasal dari sistem mikrokontroler ATmega128.
6. Motor DC berfungsi sebagai sistem kemudi kursi roda cerdas.
7. Catu daya berfungsi sebagai suplai sistem keseluruhan

$$\text{jarak} = \left[\frac{t_{IN}(s) \times 344(m/s)}{2} \right] \text{meter}$$



Gambar 5 Blok diagram sistem.

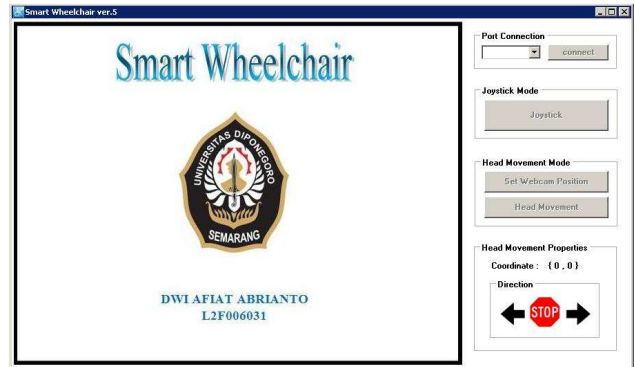


Gambar 6 Diagram alir program utama kursi roda cerdas..

3.2 Perancangan Perangkat Lunak pada Laptop

Perancangan perangkat lunak pada laptop menggunakan bahasa pemrograman Visual C# 2008 dan *library* dari emguCV. Perangkat lunak ini selain sebagai tampilan visual (*interface*) juga digunakan

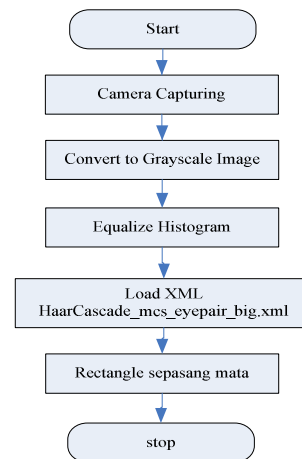
untuk mengolah hasil *capture* dari *webcam* agar didapat posisi mata dari pengendali kursi roda. Posisi mata ini akan digunakan sebagai acuan gerakan motor kursi roda. Dalam mengatur besar kecepatan linear dan kecepatan angular motor kursi roda ini digunakan metode kontrol *on-off*. Gambar 6 menunjukkan diagram alir program utama kursi roda cerdas.



Gambar 7 Interface program utama kursi roda cerdas.

3.2.1 Perancangan Program Pejejak Mata

Pada tugas akhir ini, untuk proses pengolahan citra dalam mendeteksi area mata menggunakan *library* dari EmguCV. Dalam mendeteksi mata, *class* yang dibutuhkan antara lain *class Emgu.CV.Capture*, *class Emgu.CV.Image*, dan *class Emgu.CV.HaarCascade*. Pengolahan citra yang dilakukan antara lain konversi hasil *capture webcam* ke dalam bentuk *grayscale*, normalisasi tingkat kecerahan (*brightness*) dan peningkatan *contrast* gambar (*histogram equalization*). Gambar 7 merupakan diagram alir program penjejak mata.

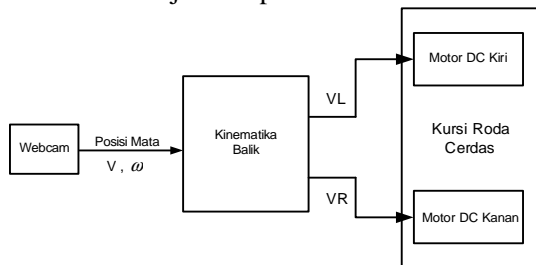


Gambar 8 Diagram alir program penjejak mata.

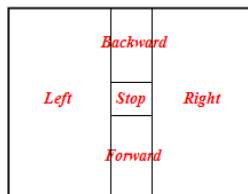
3.2.2 Perancangan Variabel Kinematika Balik Pejejak Mata

Pada tugas akhir ini digunakan untuk menjadwalkan besaran variabel kinematika balik

kecepatan linear (v) dan kecepatan angular (w) berdasarkan kondisi posisi objek mata pengendali kursi roda. Tujuannya adalah agar pada proses penentuan arah gerak (maju, mundur, kanan, kiri, atau berhenti) dan pengiriman sinyal kontrol kepada mikrokontroler berjalan cepat.



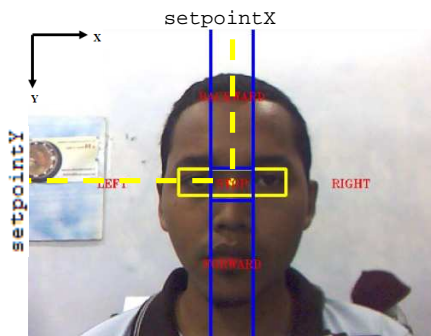
Gambar 7 Diagram alir program penjejak mata



Gambar 7 Pembagian daerah arah gerak kursi roda pada bingkai *pictureBox*.

Penentuan arah gerak ini dibagi berdasarkan posisi koordinat titik tengah mata pada 5 daerah pada bingkai *pictureBox* yaitu *forward* (maju), *backward* (mundur), *right* (kanan), *left* (kiri), dan *stop* (berhenti) seperti pada Gambar 7. Untuk mencari koordinat titik tengah mata dilakukan dengan menambah titik koordinat *rectangle* deteksi mata dengan setengah lebar atau tinggi *rectangle* tersebut.

```
setpointX = eye.rect.X + eye.rect.Width / 2
setpointY = eye.rect.Y + eye.rect.Height / 2
```



Gambar 8 Titik koordinat *setpoint*.

Koordinat *setpoint* ini digunakan sebagai acuan untuk pengontrolan variabel kecepatan linear (v) dan kecepatan angular (w) motor kursi roda. Pembagian besaran variable tersebut dapat dilihat pada tabel tengok (*look up table*) pada Tabel 2.

Tabel 2 Tabel tengok variabel kinematika balik berdasarkan posisi mata

Kondisi Posisi Titik Setpoint	Kinematika Balik	
	Kecepatan Linear (v)	Kecepatan Angular (w)
<i>Stop</i> $287 < \text{setpoint.X} < 353$ $215 \leq \text{setpoint.Y} \leq 265$	0	0
<i>Forward</i> $287 < \text{setpoint.X} < 353$ $\text{setpoint.Y} > 265$	80	0
<i>Backward</i> $287 < \text{setpoint.X} < 353$ $\text{setpoint.Y} < 215$	-78	0
<i>Left</i> $\text{setpoint.X} \leq 287$	0	80
<i>Right</i> $\text{setpoint.X} \geq 353$	0	-80

3.3 Perancangan Perangkat Lunak pada Mikrokontroler ATmega128

Tugas akhir ini merupakan pengembangan atau bagian dari sistem kursi roda cerdas secara keseluruhan. Tiap tugas dalam program dikelompokkan ke dalam fungsi-fungsi secara terpisah. Tiap fungsi dipanggil sesuai dengan pengaturan waktu pemanggilannya yang ditetapkan pada saat memrogram. Penjadwalan tugas atau pemanggilan fungsi pada tugas akhir ini menggunakan *template soft timer*. Fungsi-fungsi tersebut antara lain:

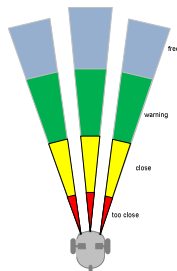
- a. *behaviour_mode()*
Fungsi ini digunakan untuk mengaktifkan program penjejak mata agar kursi roda dapat bergerak sesuai pergerakan kepala. Fungsi ini dipanggil tiap 50 ms sekali.
- b. Hitung Jarak Rintangan dan *Set Flag* Rintangan
Merupakan tugas yang berisi perintah untuk menghitung jarak objek yang terdeteksi oleh sensor PING)))TM dan mengaktifkan *flag_obs* jika jarak terdeteksi kurang dari batas yang ditentukan. Pada program, tugas menghitung jarak rintangan dan *set flag* rintangan dipisah menjadi dua *task* dengan tujuan supaya program lebih rapi. *Task* tersebut adalah *cek_ping* dan *read_ping*.
- d. *update_command()*
Fungsi ini digunakan untuk *update* posisi mata yang digunakan sebagai perintah pergerakan kursi roda. Fungsi ini dipanggil tiap 100 ms sekali.

3.3.1 Obstacle Avoidance

Merupakan subrutin program yang berisi perintah untuk mengatur kursi roda agar bergerak

menghindari halangan yang berada di depannya, dengan cara berbelok ke kiri atau kanan.

Perancangan tingkah laku *obstacle avoidance* dilakukan dengan membagi *range* dari tiap sensor menjadi 4 bagian yaitu *too close*, *close*, *warning* dan *free*. Daerah *too close* adalah daerah yang berada pada jarak kurang dari 45 cm. Daerah *close* adalah daerah yang berada pada jarak diantara 45 cm sampai 100 cm. Daerah *warning* adalah daerah yang berada pada jarak diantara 100 cm sampai 150 cm. Sedangkan daerah *free* adalah daerah yang berada pada jarak lebih dari 150 cm. Gambar 9 merupakan gambar pembagian area dari sensor jarak.



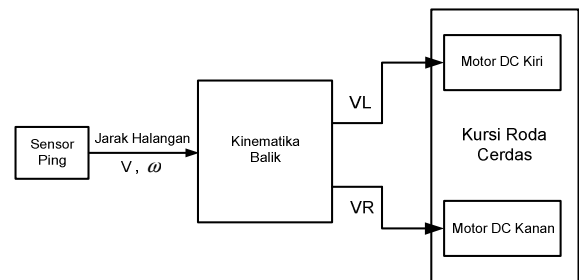
Gambar 9 Gambar pembagian area dari sensor jarak.

Dari keempat daerah tersebut, maka dapat disusun 5 keadaan berdasarkan kemungkinan posisi halangan seperti pada Tabel 3 berikut.

Tabel 3 Tabel kemungkinan posisi halangan.

Keadaan 1	
Keadaan 2	
Keadaan 3	
Keadaan 4	
Keadaan 5	

Fungsi *obstacle avoidance* ini menggunakan metode kontrol *on-off* untuk mengatur besaran variabel kinematika balik dari masing-masing *rule*.



Gambar 10 Diagram blok pengendalian motor kursi roda berdasar pada jarak halangan pada sensor ping.

Pembagian besaran variable tersebut dapat dilihat pada tabel tengok (*look up table*) pada Tabel 4.

Tabel 4 Tabel tengok variabel kinematika balik berdasarkan jarak halangan.

Pembacaan Sensor Ping	Kinematika Balik	
	v	w
Keadaan 1	0	0
Keadaan 2	$\frac{(\text{jarak halangan sensor tengah}) \times 100}{175}$	Jika jarak halangan sensor kiri > sensor kanan: $w = v$ Jika jarak halangan sensor kiri < sensor kanan: $w = -v$
Keadaan 3	Jika jarak halangan sensor kiri > sensor kanan: $\frac{(\text{jarak halangan sensor kanan}) \times 100}{120}$ Jika jarak halangan sensor kiri < sensor kanan: $\frac{(\text{jarak halangan sensor kiri}) \times 100}{120}$	Jika jarak halangan sensor kiri > sensor kanan: $w = v$ Jika jarak halangan sensor kiri < sensor kanan: $w = -v$
Keadaan 4	$\frac{(\text{jarak halangan sensor tengah}) \times 100}{150}$	0
Keadaan 5	$\frac{(\text{jarak halangan sensor tengah}) \times 100}{150}$	Jika jarak halangan sensor kiri > sensor kanan: $w = v/2$ Jika jarak halangan sensor kiri < sensor kanan: $w = -v/2$

IV PENGUJIAN DAN ANALISA

4.1 Pengujian Perangkat Keras

4.1.1 Pengujian Driver Motor DC

Driver motor yang digunakan adalah IC Rangkaian driver motor EMS (*Embedded Module Series*) 5 A H-Bridge, driver ini mendapat masukan dari PWM yang dihasilkan oleh timer 1 mikrokontroler AVR. Hasil pengujian driver motor dapat dilihat pada Tabel 5 berikut.

Tabel 5 Hasil Pengujian Driver motor.

Nilai OCR1A	Nilai OCR1B	V _{keluaran} channel 1 (volt) terukur	V _{keluaran} channel 2 (volt) terukur	V _{keluaran} (volt) Perhitungan
100	100	2,3	2,1	1,17
200	200	3,9	3,8	2,35
300	300	4,9	4,8	3,52
400	400	5,9	5,6	4,69
500	500	6,8	6,9	5,87
600	600	7,7	8,0	7,04
700	700	8,5	8,8	8,21
800	800	9,3	9,7	9,38
1023	1023	11,8	12,1	12,0

Pengujian driver motor DC menggunakan tegangan 12 V. Dari tabel 4.1 di atas terlihat bahwa semakin besar nilai OCR1A dan OCR1B yang

diberikan maka akan semakin besar tegangan yang dihasilkan.

4.1.2 Pengujian Sensor PING)))™

Pengujian terhadap sensor PING)))™ dilakukan dengan mengukur jarak suatu benda terukur yang diletakkan dihadapan sensor, pembacaan sensor dibatasi pada jarak 10 cm – 150 cm. Data hasil pengukuran ditunjukkan pada Tabel 6.

Tabel 6 Hasil pengukuran jarak dengan menggunakan PING)))™.

No.	Jarak Terdeteksi PING)))™ (cm)	Jarak Pengukuran Sebenarnya		
		PING)))™ kiri (cm)	PING)))™ tengah (cm)	PING)))™ kanan (cm)
1	10	8	9	9
2	20	18	19	19
3	30	31	31	30
4	40	41	39	39
5	50	51	50	51
6	70	71	70	71
7	90	91	90	91
8	110	111	109	109
9	130	131	129	129
10	150	151	150	151

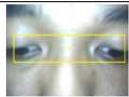




Dari hasil pengujian terdapat *error* rata-rata antara jarak yang terdeteksi dengan jarak pengukuran sebenarnya sebesar 0,9 cm. Hal ini disebabkan karena pengukuran atau adanya pembulatan dalam pemrograman.

4.2 Pengujian Perangkat Lunak

4.2.1 Pengujian Karakteristik Program Penjejak Mata

Pengujian ini meliputi pengujian pengaruh jarak, cahaya dan posisi kepala pengendali kursi roda.

Tabel 7 Pengujian jangkauan jarak deteksi mata.

Jarak webcam dengan kepala pengendali (cm)	Hasil Deteksi	Tampilan Software
8	terdeteksi	
45	terdeteksi	
70	terdeteksi	
90	terdeteksi	
100	Tidak terdeteksi	

Berdasarkan hasil pengujian Tabel 7 maka *webcam* dipasang di depan kepala pengendali dengan jarak 60 cm.

Tabel 8 Pengujian posisi kepala berdasarkan sudut orientasi.

Posisi Kepala	Sudut Orientasi			
	10°	20°	30°	45°
Menoleh ke kanan	terdeteksi	terdeteksi	terdeteksi	Tidak terdeteksi
Menoleh ke kiri	terdeteksi	terdeteksi	terdeteksi	Tidak terdeteksi
Serong ke kanan	terdeteksi	terdeteksi	terdeteksi	Tidak terdeteksi
Serong ke kiri	terdeteksi	terdeteksi	terdeteksi	Tidak terdeteksi
Menunduk	terdeteksi	terdeteksi	terdeteksi	terdeteksi
Menengadiah	terdeteksi	terdeteksi	terdeteksi	terdeteksi

Tabel 9 Pengujian pengaruh kondisi pencahayaan.

Posisi Kepala	Kondisi Pencahayaan		
	Redup	Terang	Sangat Terang
Menoleh ke kanan	terdeteksi	terdeteksi	terdeteksi
Menoleh ke kiri	Tidak terdeteksi	terdeteksi	terdeteksi
Serong ke kanan	Tidak terdeteksi	terdeteksi	terdeteksi
Serong ke kiri	Tidak terdeteksi	terdeteksi	terdeteksi
Menunduk	terdeteksi	terdeteksi	terdeteksi
Menengadiah	Tidak terdeteksi	terdeteksi	terdeteksi

Dari Tabel 8 dan 9, program hanya mampu mendeteksi pergerakan kepala dengan sudut orientasi maksimal 30° dan dengan kondisi cahaya terang/sangat terang.

4.2.2 Pengujian Tingkah Laku Kursi Roda Cerdas Tanpa Halangan

Pada pengujian ini, ketiga sensor jarak tidak mendeteksi adanya halangan, dan perubahan pergerakan berawal dari kondisi diam (*stop*). Hasil pengujian dapat dilihat pada Tabel 10. Dari hasil pengujian didapat bahwa waktu respon rata-rata dalam setiap perubahan pergerakan kepala dengan respon berputarnya roda kursi roda sebesar 0,53 s. Waktu tunda (*delay*) ini dikarenakan lama proses pendeteksian mata dan komunikasi serial antara laptop dengan mikrokontroler. Secara keseluruhan pergerakan kursi roda telah sesuai dengan pergerakan kepala.

Tabel 10 Hasil pengujian tingkah laku kursi roda cerdas tanpa halangan.

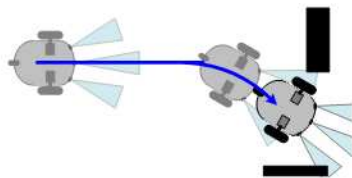
Pergerakan Kepala	Kinematika balik		Pergerakan motor kursi roda		Arah Kursi Roda	Waktu Respon (s)
	v	w	Motor Kiri	Motor Kanan		
Menunduk	80	0	maju	maju		0,53
Menengadahkan	-78	0	mundur	mundur		0,55
Menoleh ke kiri	0	80	mundur	maju		0,50
Menoleh ke kanan	0	-80	maju	mundur		0,56

4.2.3 Pengujian Tingkah Laku Kursi Roda Cerdas dengan Halangan

Pada tahap ini, pengujian dilakukan dengan meletakkan benda di beberapa keadaan. Tabel 11 merupakan hasil pengujian tingkah laku kursi roda cerdas dengan halangan. Besar variable kinematika balik telah sesuai dengan *rule* yang ada.

Tabel 11 Hasil pengujian tingkah laku kursi roda cerdas dengan halangan.

Pergerakan Kepala	Jarak Terdeteksi (PING)) TM (cm)			v	w	PWM	
	Kiri	Tengah	Kanan			Kiri	Kanan
Menunduk	170	40	157	0	0	0	0
Menengadahkan	139	35	138	-78	0	-797	-797
Menoleh ke kiri	23	70	135	0	0	0	0
Menoleh ke kanan	180	80	31	0	0	0	0
Menunduk	169	82	160	46	46	0	941
Menunduk	81	54	169	67	-33	1023	347
Menunduk	192	86	84	70	35	358	1023
Menunduk	169	109	237	90	0	920	920
Menunduk	147	156	239	104	-52	1023	531
Menunduk	169	129	120	87	43	450	1023



Gambar 11 Pengujian tingkah laku *obstacle avoidance*.

Gambar 11 menunjukkan kursi roda dapat mendeteksi adanya halangan di sebelah kiri, kursi roda bergerak menghindari halangan dengan berbelok ke arah kanan.

V PENUTUP

5.1 KESIMPULAN

Berdasarkan perancangan, pengujian dan analisa yang telah dilakukan, maka dapat disimpulkan beberapa hal sebagai berikut:

1. Pendeteksian gerakan kepala berdasarkan titik tengah area mata hasil *capture* yang ditangkap oleh *webcam*. Pendeteksian ini sangat dipengaruhi oleh besar intensitas cahaya dan jarak antara wajah dengan *webcam*. Jarak minimal yang dapat dideteksi sebesar 8 cm sedangkan jarak maksimal yang dapat dideteksi adalah sebesar 90 cm.
2. Kursi roda cerdas dapat dikendalikan dengan menggunakan pergerakan kepala. Gerakan kursi roda sesuai dengan perubahan gerakan kepala, jika di sekitar kursi roda tidak terdeteksi adanya halangan. Saat kepala menunduk, maka kursi roda akan maju ke depan sedangkan pada saat kepala menengadahkan ke atas, kursi roda akan bergerak mundur. Ketika kepala menoleh ke kanan atau kiri, maka kursi roda akan bergerak berotasi ke kanan atau kiri. Besar sudut orientasi pergerakan kepala ini maksimal 30°.
3. Perubahan gerakan kepala dengan perubahan gerakan motor kursi roda terdapat waktu tunda (*delay*) rata-rata sebesar 0,53 detik. Waktu tunda ini dikarenakan adanya waktu proses pendeteksian mata dan waktu komunikasi serial antara laptop dengan mikrokontroler.
4. Kursi roda cerdas dapat mendeteksi adanya halangan yang berada di depan dan di samping, jika kursi roda digerakkan maju ke depan maka kursi roda akan menghindari halangan tersebut dengan berbelok secara perlahan-lahan dan semakin berbelok tajam jika halangan semakin dekat. Tingkah laku ini dinamakan sebagai *obstacle avoidance*.
5. Kursi roda cerdas akan berhenti bergerak saat mendeteksi adanya halangan di depan atau di samping pada jarak kurang dari sama dengan 45 cm dan hanya dapat digerakkan mundur ke belakang.

5.2 Saran

Untuk pengembangan sistem lebih lanjut, maka penulis memberikan saran-saran sebagai berikut:

1. Penambahan sensor jarak pada kursi roda cerdas ini akan memperluas area pendeteksian halangan.
2. Penambahan sensor *limit switch* pada sekeliling kursi roda dapat mendeteksi letak terjadinya benturan/tabrakan.
3. Proses pendeteksian gerakan kepala dapat menggunakan metode lain seperti *matching template*, *Hidden Markov Model* (HMM), atau menggunakan sensor seperti *accelerometer*.
4. Kursi roda cerdas dapat dikembangkan dengan menambahkan kemampuan tingkah laku yang lain, seperti otomatisasi mengantarkan pemakai ke posisi tertentu (*go to goal*).
5. Kursi roda cerdas dapat dikembangkan untuk dikendalikan menggunakan media lain selain *joystick* atau gerakan kepala, seperti dengan menggunakan suara atau gerakan otot.

DAFTAR PUSTAKA

- | | |
|--|--|
| <p>[1] Aditama, <i>Pengenalan Citra Wajah dengan Metode Adaptive Hidden Markov Model</i>, Tugas Akhir, Universitas Kristen Petra, Surabaya, 2006.</p> <p>[2] Arihutomo, Mukhlas, <i>Rancang Bangun Sistem Penjejakan Objek Menggunakan Metode Viola Jones untuk Aplikasi Eyebot</i>, Tugas Akhir PENS, Institut Teknologi Sepuluh Nopember, Surabaya, 2010.</p> <p>[3] Fauzan, Isnain, <i>Pemetaan Posisi dan Orientasi Kursi Roda Cerdas Berbasis Dead Reckoning</i>, Skripsi S-1, Universitas Diponegoro, Semarang, 2010.</p> <p>[4] Kurniawan, Eddy, <i>Pengenalan Citra Wajah dengan Menggunakan Principal Common Analysis dan Local Feature Analysis</i>, Tugas Akhir, Universitas Kristen Petra, Surabaya, 2003.</p> <p>[5] Lienhart, R. and Maydt, J. <i>An extended set of Haar-like features for rapid object detection</i>. IEEE ICIP 2002, Vol. 1, pp. 900-903, Sep. 2002.</p> <p>[6] Miftahurrozaq, <i>Perancangan Sistem Kursi Roda Cerdas Menggunakan Pendekatan Model Fungsional dan Model Tingkah Laku dan Perancangan Perilaku Obstacle Avoidance</i>, Skripsi S-1, Universitas Diponegoro, Semarang, 2010.</p> <p>[7] Ongkodjojo, R. Paska, <i>Perancangan dan Pembuatan Aplikasi Pelacakan Pergerakan Wajah secara Real Time dengan Menggunakan</i></p> | <p><i>Facial Feature Extraction</i>, Tugas Akhir, Universitas Kristen Petra, Surabaya, 2006.</p> <p>[8] Open Computer Vision Library Reference Manual, Intel Corporation, http://software.intel.com/sites/oss/pdfs/OpenCVreferencemanual.pdf</p> <p>[9] Prasetyo, Rudy, <i>Implementasi Sistem Kontrol Navigasi Reaktif Subsumption Pada Kursi Roda Cerdas</i>, Skripsi S-1, Universitas Diponegoro, Semarang, 2008.</p> <p>[10] Setiawan, Iwan, <i>Fungsi Soft Timer Untuk Keperluan Operasi Tundaan dan Penjadwalan (Scheduling) Pada Sistem Embedded</i>, http://iwan.blog.undip.ac.id. Juni 2009.</p> <p>[11] Umam, Fatkhul, <i>Penjejak Objek pada Kursi Roda Cerdas Autonomous</i>. Skripsi S-1, Universitas Diponegoro, Semarang, 2011.</p> <p>[12] Viola, P. and Jones, M. <i>Rapid object detection using boosted cascade of simple features</i>. IEEE Conference on Computer Vision and Pattern Recognition, 2001.</p> <p>[13] -----, <i>Atmega128 Data Sheet</i>, http://www.atmel.com.</p> <p>[14] -----, <i>Download And Installation</i>, http://www.emgu.com.</p> <p>[15] -----, <i>PING)))™ Ultrasonic Distance Sensor Data Sheet</i>, http://www.parallax.com.</p> |
|--|--|

Dwi Afiat Abrianto (L2F006031)



Lahir di Surabaya, 5 Oktober 1987. Saat ini sedang melanjutkan studi pendidikan strata I di Jurusan Teknik Elektro, Fakultas Teknik Universitas Diponegoro Konsentrasi Kontrol.

Mengetahui dan mengesahkan,

Dosen Pembimbing I

Dosen Pembimbing II

Iwan Setiawan, ST, MT
NIP. 197309262000121001

Achmad Hidayatno, ST, MT
NIP. 196912211995121001

Tanggal: _____

Tanggal: _____