

# Implementasi Sistem *Voice Recognition* pada Robot Pemindah Objek sebagai Sistem Navigasi

Jatra Kurnia Ardi<sup>1</sup>, Nurussa'adah, Ir., MT<sup>2</sup>, Mochammad Rif'an, ST., MT<sup>3</sup>.

<sup>1</sup>Mahasiswa Teknik Elektro UB, <sup>2,3</sup>Dosen Teknik Elektro UB

jatra.kurnia@gmail.com

**Abstrak** – Perkembangan ilmu pengetahuan dan teknologi khususnya di bidang robotika membawa dampak positif dalam kehidupan manusia yang pada saat ini telah sampai pada zaman *autonomous robot*. Perkembangan di bidang robotika ini juga mampu memberikan dampak positif di bidang industri. Salah satu contohnya adalah proses memindahkan sebuah barang/objek dari suatu tempat ke tempat lainnya. Penggunaan *mobile robot* yang dapat memindahkan barang/objek dapat menekan biaya produksi dan meningkatkan efisiensi waktu. Dalam kenyataannya, sering dijumpai robot pemindah barang/objek yang salah satu sistem kerjanya adalah dengan cara memberi perintah pada *mobile robot* tersebut menggunakan lebih dari satu tombol atau dalam bentuk *joystick*. Oleh karena itu, dalam skripsi ini akan dirancang Implementasi Sistem *Voice Recognition* pada Robot Pemindah Objek sebagai Sistem Navigasi. Metode pengambilan *sample* suara pada modul *EasyVR* dilakukan sebanyak dua kali pengambilan suara dengan variasi pengucapan yang relatif sama pada setiap kata. Hal ini dilakukan sesuai dengan kemampuan *EasyVR* yang tidak bisa menerima pengucapan variasi suara kedua jika berbeda dengan variasi pengucapan suara pertama. Selain itu, *EasyVR commander* memberi batas waktu selama 5 detik untuk setiap pengucapan suara yang akan dijadikan *sample*. Penerapan sistem *voice recognition* pada robot pemindah objek dapat digunakan sebagai sistem navigasi robot meskipun belum mampu menggantikan penggunaan perangkat berupa *joystick*.

**Kata Kunci:** sistem *voice recognition*, *EasyVR*, robot pemindah objek.

## I. PENDAHULUAN

Perkembangan ilmu pengetahuan dan teknologi khususnya di bidang robotika membawa dampak positif dalam kehidupan manusia yang pada saat ini telah sampai pada zaman *autonomous robot*. *Autonomous robot* adalah suatu robot yang mampu berperilaku secara mandiri (hanya sesekali membutuhkan perintah atau otomatis). [1]

*Autonomous robot* dapat dibagi menjadi dua yaitu *autonomous stationary robot* dan *autonomous mobile robot*. *Autonomous stationary robot* adalah robot mandiri yang dirancang untuk tidak berpindah posisi atau tetap (*station*). Salah satu aplikasi *autonomous stationary robot* yang sering dijumpai adalah robot lengan yang dilengkapi manipulator (*capit*, *bor*, dll.). Sedangkan *autonomous mobile robot* adalah robot mandiri yang dirancang untuk dapat berpindah posisi (*mobile*). Salah satu aplikasi *autonomous mobile robot* adalah robot yang dapat memindahkan sebuah objek dari suatu tempat ke tempat yang lainnya.

Perkembangan di bidang robotika ini juga mampu memberikan dampak positif di bidang industri. Salah

satunya adalah proses memindahkan sebuah barang/objek dari suatu tempat ke tempat lainnya. Jika zaman dahulu untuk memindahkan barang/objek dari satu tempat ke tempat lainnya membutuhkan tenaga manusia yang cukup banyak, maka untuk saat ini hal tersebut dirasa kurang efisien lagi. Penggunaan *mobile robot* yang dapat memindahkan barang/objek dapat menekan biaya produksi dan meningkatkan efisiensi waktu.

Dalam kenyataannya, sering dijumpai robot pemindah barang/objek yang salah satu sistem kerjanya adalah dengan cara memberi perintah pada *mobile robot* tersebut menggunakan lebih dari satu tombol atau dalam bentuk *joystick*. Hal tersebut masih memiliki kekurangan yaitu masih adanya bantuan fisik manusia yang dapat diartikan robot tersebut masih belum bisa mandiri. Oleh karena itu, dalam skripsi ini akan dirancang Implementasi Sistem *Voice recognition* pada Robot Pemindah Objek sebagai Sistem Navigasi. Fungsi dari sistem ini adalah untuk memberi perintah kepada robot dengan menggunakan suara manusia, sehingga manusia tersebut tidak perlu bersentuhan langsung dengan robot.

Pada penelitian yang dilakukan oleh Mohamed Fezari (2009) telah dirancang suatu sistem *voice recognition* dengan judul *New Speech Processor and Ultrasonic Sensors Based Embedded System to Improve the Control of a Motorised Wheelchair*. Namun, masih ada kekurangan dari penelitian tersebut yaitu tingkat keberhasilan menerima perintah suara dengan benar sebesar 80%, jarak yang digunakan dalam pemberian perintah hanya 5 cm sampai 20 cm, dan pengujian hanya dilakukan dengan satu jenis perintah

Perancangan ini menggunakan *mobile robot* berukuran kecil yang berfungsi sebagai *prototype* untuk *mobile robot* berukuran besar yang mampu memindahkan barang berukuran besar, misal barang-barang industri. Modul yang digunakan dalam perancangan ini adalah modul *EasyVR* yang berfungsi sebagai telinga pada robot agar mampu bernavigasi sesuai perintah yang diberikan.

## II. TINJAUAN PUSTAKA

### A. Modul *EasyVR*

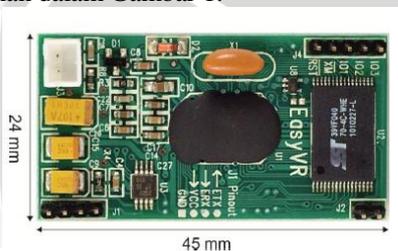
*EasyVR* merupakan modul *voice recognition* multi-fungsi yang dapat digunakan pada banyak aplikasi pengontrolan yang membutuhkan pendeteksian suara dan percakapan. Modul ini dapat digunakan/dihubungkan dengan board mikrokontroler *Arduino*. Cocok digunakan untuk beragam aplikasi,

seperti home automation (dimana kita dapat mengontrol nyala lampu, kunci pintu, televisi, atau perangkat lainnya) atau sebagai modul pelengkap sensor pendengaran robot yang dibuat sebagaimana robot-robot canggih yang dijual di pasaran yang harganya luar biasa mahal.

Secara umum, fitur dari *EasyVR* adalah sebagai berikut:

- Mendukung beberapa bahasa, yaitu: English(US), Italian, German, French, Spanish, Japanese.
- Mendukung hingga 32 *custom Speaker Dependent* (SD) trigger atau perintah, bahkan dapat digunakan pada bahasa apapun.
- GUI yang mudah digunakan.
- x GPIO (IO1, IO2, IO3) dapat dikontrol dengan perintah protokol baru.
- PWM audio output mendukung speaker 8 ohm.

Modul *EasyVR* dapat digunakan dengan antarmuka UART yang didukung pada rentang tegangan 3.3V - 5V, seperti PIC dan board *Arduino*. Beberapa contoh aplikasi termasuk otomatisasi rumah, seperti suara yang mengendalikan switch pada lampu, kunci, atau penambahan "pendengaran" untuk robot yang saat ini sedang berkembang. Bentuk fisik *EasyVR* ditunjukkan dalam Gambar 1.

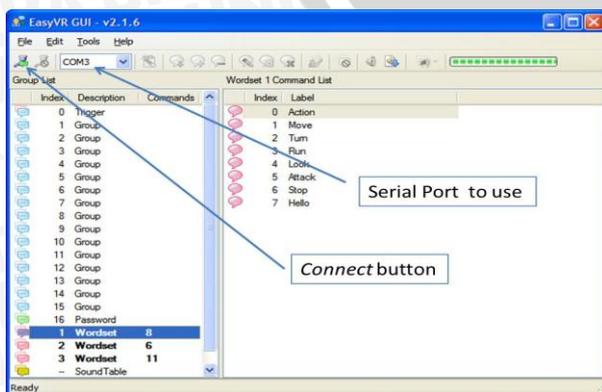


Gambar 1. Bentuk fisik *EasyVR*.

### B. *EasyVR Commander*

*EasyVR Commander* adalah software yang digunakan untuk mengkonfigurasi modul *EasyVR* yang terhubung ke dengan menggunakan mikrokontroler yang menyediakan program "bridge".

Pengguna dapat menentukan kelompok perintah atau password dan menghasilkan kode template untuk menangani mereka. Hal ini diperlukan untuk mengedit kode yang dihasilkan untuk mengimplementasikan aplikasi logika. Template berisi semua fungsi atau subrutin untuk menangani tugas-tugas pengenalan suara. Tampilan software *EasyVR Commander* ditunjukkan dalam Gambar 2.



Gambar 2. Tampilan pada *EasyVR Commander*.

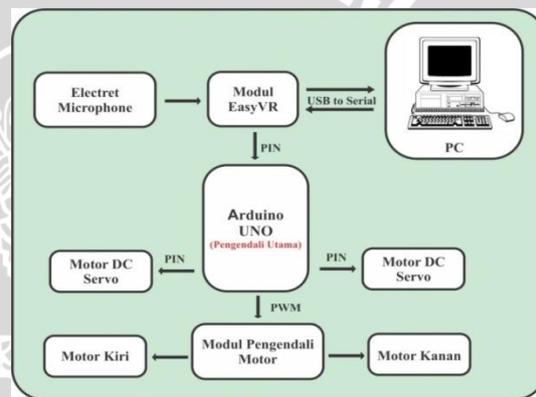
Terdapat empat jenis perintah dalam *EasyVR Commander*, yaitu:

- **Trigger** : kelompok khusus yang memiliki *built-in* memicu kata SI "Robot" dan dapat menambahkan satu user-defined pemacu kata SD. Kata-kata pemacu digunakan untuk memulai proses pengenalan.
- **Group** : untuk menambahkan perintah SD user-defined.
- **Password** : kelompok khusus untuk "password vokal" (sampai lima), menggunakan Speaker Verifikasi (SV) teknologi.
- **Wordset** : built-in set perintah SI (misalnya pada Gambar 1 di atas, Wordset 1 dipilih).

## III. PERANCANGAN DAN PEMBUATAN ALAT

### A. Perancangan Sistem

Blok diagram keseluruhan sistem yang dirancang dibagi menjadi dua bagian, yaitu blok mikrokontroler utama, blok pengendali motor DC, dan blok *EasyVR* seperti ditunjukkan dalam Gambar 3.



Gambar 3. Diagram Blok Sistem.

Fungsi masing-masing blok dalam diagram blok diatas adalah sebagai berikut :

#### 1). Blok Mikrokontroler Utama

Mikrokontroler yang digunakan adalah *Arduino UNO* yang berfungsi untuk mengolah data dari modul *EasyVR* dan mengakses servo untuk kendali arah.

#### 2). Blok Modul *EasyVR*

Pergerakan dari robot ini dibantu dengan modul *EasyVR* yang berfungsi sebagai sensor suara. Data-data suara akan di sampling melalui komputer menggunakan program *EasyVR Commander*. Data-data yang sudah disampling akan dimasukkan kembali ke database *EasyVR*. Mikrokontroler utama berupa *Arduino UNO* bekerja mengatur dan mengolah data dari modul *EasyVR*.

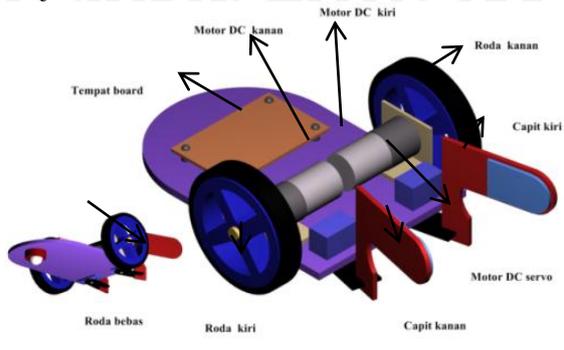
#### 3). Blok motor DC

Pada modul pengendali motor, data yang diterima dikonversi menjadi sinyal kendali arah. Motor kanan dan motor kiri berfungsi sebagai kemudi yang menggerakkan robot ketika harus maju, mundur, belok kanan atau belok kiri.

### B. Desain Mekanik

Sistem mekanik yang baik, mendukung pergerakan robot menjadi lebih baik, oleh karena itu perancangan mekanik dalam hal ini badan robot haruslah proporsional dengan panjang dan lebar.

Gambar perspektif dan rancangan ukuran Robot ditunjukkan dalam Gambar 4.

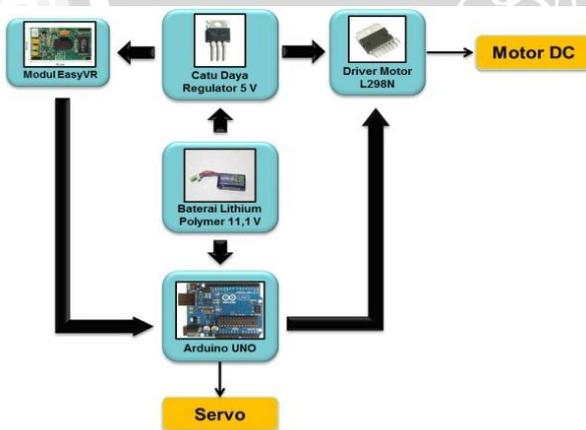


Gambar 4. Perspektif Desain Mekanik Robot Beroda.

Badan robot terbuat dari bahan mika *acrylic* dengan ketebalan 3 mm, ke dua buah roda depan berbahan *acrylic* dengan tebal 6 mm dan berdiameter 10 cm. Roda belakang menggunakan roda *castor* (roda bebas). Caput robot terbuat dari bahan *acrylic* dengan tebal 2 mm. Bentuk dan dimensi robot dirancang sesuai secara proporsional dengan harapan robot dapat bergerak sesuai dengan fungsinya yaitu memindahkan objek.

**C. Perancangan Desain Sistem Elektronik**

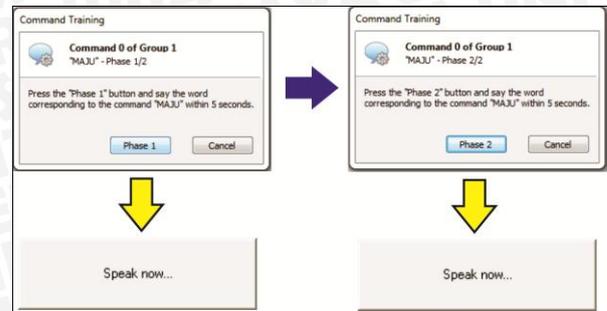
Diagram blok sistem elektronik, terdiri dari tiga bagian catu daya (baterai), masukan, bagian kendali, bagian keluaran. Pada bagian masukan berupa sebuah sensor microphone yang berfungsi untuk menerima masukan berupa suara manusia yang terhubung dengan *EasyVR* sebagai pengolah data sinyal suara. Pada bagian kendali menggunakan *Arduino UNO* sebagai mikrokontroler utama dan pengontrol gerak servo. Pada bagian keluaran berupa *driver* motor sebagai penggerak aktuator robot.



Gambar 5. Desain Sistem Elektronik Robot Beroda

**D. Perancangan Sistem Voice Recognition Menggunakan EasyVR**

Perancangan ini bertujuan untuk mengambil *sample* suara yang akan disimpan ke dalam modul *EasyVR*. Pengambilan *sample* suara dilakukan melalui PC dengan software bawaan dari *EasyVR* yaitu *EasyVR commander*. *Sample* suara yang akan digunakan adalah sebanyak tujuh kata.

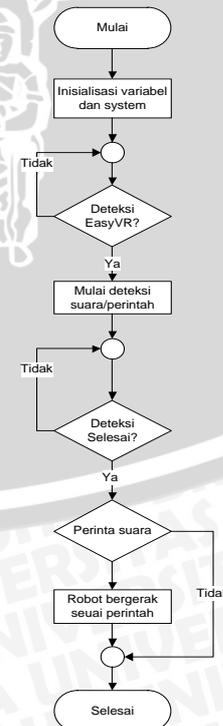


Gambar 6. Diagram blok pemberian *sample*.

Diagram blok pemberian *sample* ditunjukkan dalam Gambar 6. Pengambilan *sample* suara dilakukan dalam kondisi ideal atau tanpa adanya *noise* dan dilakukan sebanyak dua kali pengambilan suara dengan variasi pengucapan yang relatif sama pada setiap kata. Hal ini dilakukan sesuai dengan kemampuan *EasyVR* yang tidak bisa menerima pengucapan variasi suara kedua jika berbeda dengan variasi pengucapan suara pertama. Oleh karena itu, agar *EasyVR* dapat berfungsi atau dapat mengakses perintah suara, dibutuhkan variasi suara yang relatif sama dengan *sample*. Selain itu, *EasyVR commander* memberi batas waktu selama 5 detik untuk setiap pengucapan suara yang akan dijadikan *sample*. [2]

**E. Perancangan Susunan Perangkat Lunak**

Tahapan proses yang terdapat pada sistem ini meliputi proses pengolahan data dari modul *EasyVR* ke *Arduino* dan proses pengontrolan motor DC maupun servo. Semua proses tersebut dilakukan oleh perangkat lunak yang terdapat dalam mikrokontroler. Perangkat lunak ini tersusun dari instruksi-instruksi yang membentuk sebuah *listing* program atau *source code*.



Gambar 7. Diagram alir perancangan perangkat lunak pada robot.

Diagram alir perancangan perangkat lunak ditunjukkan dalam Gambar 7 yang ada pada mikrokontroler robot. Robot akan menginisialisasi

sistem dan akan mendeteksi keberadaan *EasyVR* sebelum menunggu input berupa suara. Data input berupa suara dibandingkan dengan data sampling yang ada pada *EasyVR*. Jika sesuai, board *Arduino UNO* akan menerima perintah dari board *EasyVR* dan akan mengakses. Setelah itu, *Arduino UNO* mengirimkan perintah kepada rangkaian driver motor dan servo. Setelah itu, robot akan mulai bergerak sesuai dengan perintah yang diberikan.

#### IV. PENGUJIAN DAN ANALISIS SISTEM

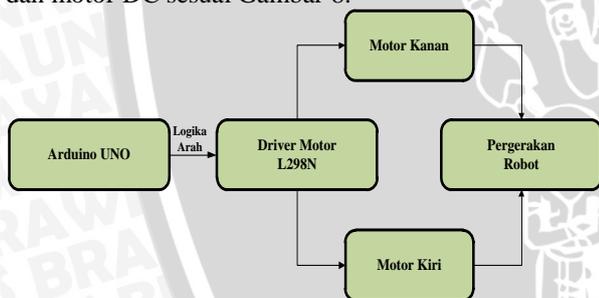
Pengujian dan analisis dilakukan untuk mengetahui apakah sistem telah bekerja sesuai perancangan yang telah dilakukan. Pengujian dilakukan per blok sistem kemudian secara keseluruhan.

##### A. Pengujian Rangkaian Driver Pengendali Motor DC

Pengujian ini bertujuan untuk mengetahui kinerja dan respon dari rangkaian *driver* motor L298N sebagai pengendali motor DC dengan membandingkan dan menguji sinyal keluaran dari *driver* motor L298N terhadap sinyal masukan arah dan sinyal masukan PWM yang diberikan oleh mikrokontroler.[3]

Pada pengujian ini ada dua prosedur yang harus dilakukan, yaitu prosedur untuk pengujian respon *driver* motor L298N terhadap sinyal masukan arah dari mikrokontroler.

Prosedur pengujian dilakukan dengan menghubungkan *driver* motor L298N, *Arduino UNO*, dan motor DC sesuai Gambar 8.



Gambar 8. Diagram Blok Pengujian Respon Sinyal Arah Rangkaian Driver Motor L298N

*Arduino UNO* akan memberikan instruksi berupa arah dan kecepatan pada *driver* motor L298N. Kecepatan dan arah putar motor DC diukur dengan menggunakan perangkat *timer 0* dan *timer 2*. Motor DC kemudian berputar sesuai intruksi arah yang diberikan oleh *Arduino UNO*. Dengan memberi masukan sinyal arah motor 1 atau 0 ke *driver* motor L298N maka di dapatkan hasil pengujian pada Tabel 1.

Tabel 1. Hasil pengujian respons driver motor L298N.

| Logika Arah Motor Kanan |         | Logika Arah Motor Kiri |         | Respon Robot          |                 |
|-------------------------|---------|------------------------|---------|-----------------------|-----------------|
| PIN D.5                 | PIN D.4 | PIN D.6                | PIN D.7 | Hasil yang diharapkan | Hasil Pengujian |
| 1                       | 0       | 1                      | 0       | Maju                  | Maju            |
| 0                       | 1       | 0                      | 1       | Mundur                | Mundur          |
| 1                       | 0       | 0                      | 0       | Belok ke kiri         | Belok ke kiri   |
| 0                       | 0       | 1                      | 0       | Belok ke kanan        | Belok ke kanan  |
| 0                       | 0       | 0                      | 0       | Stop                  | Stop            |

Pada Tabel 1 dapat diketahui bahwa respon *driver* motor L298N terhadap masukan sinyal arah pada mikrokontroler berjalan sesuai perancangan yang diharapkan. sehingga dapat disimpulkan bahwa *driver* motor L298N dapat bekerja dengan baik saat mendapatkan sinyal logika arah dari mikrokontroler.

##### B. Pengujian Jarak Ideal Pemberian Perintah Suara Terhadap Jarak Pemberian Sample Suara

Tujuan dari pengujian ini adalah untuk mengetahui berapa jarak ideal pada modul *EasyVR* untuk dapat menerima perintah dengan baik jika ditentukan variasi pemberian sample suara. Masing-masing perintah diucapkan 10 kali pada setiap jarak yang ditentukan. Sehingga, dapat diketahui keberhasilan pada setiap jarak dan diketahui range jarak ideal untuk memberi perintah suara dan memberi sample suara. Pegucapan perintah suara baik pemberian sample maupun pemberian perintah dilakukan dengan variasi pengucapan yang relatif sama.

Tabel 2. Hasil pengujian jarak ideal pemberian perintah suara

| Keberhasilan Pemberian Perintah Berdasarkan Jarak Pemberian Sample Suara |                          |      |      |      |      |       |       |        |    |
|--|--------------------------|------|------|------|------|-------|-------|--------|----|
| Kata MAJU  | Jarak Pemberian Perintah |      |      |      |      |       |       |        |    |
|  | 1 cm                     | 2 cm | 3 cm | 4 cm | 5 cm | 10 cm | 50 cm | 100 cm |    |
| Jarak Pemberian Sample   | 1 cm                     | 10   | 10   | 10   | 10   | 0     | 0     | 0      | 0  |
|  | 2 cm                     | 10   | 10   | 10   | 10   | 10    | 0     | 0      | 0  |
|  | 3 cm                     | 0    | 10   | 10   | 10   | 10    | 10    | 10     | 10 |
|  | 4 cm                     | 0    | 10   | 10   | 10   | 10    | 10    | 10     | 10 |
|  | 5 cm                     | 0    | 10   | 10   | 10   | 10    | 10    | 10     | 10 |
|  | 10 cm                    | 0    | 0    | 10   | 10   | 10    | 10    | 10     | 10 |
|  | 50 cm                    | 0    | 0    | 0    | 0    | 10    | 10    | 10     | 10 |
|  | 100 cm                   | 0    | 0    | 0    | 0    | 10    | 10    | 10     | 10 |

Tabel 2 menunjukkan tingkat keberhasilan pemberian perintah suara dengan kata "MAJU" dengan jarak yang ditentukan berdasarkan jarak pemberian *sample* yang ditentukan pula. Hasil pengujian membuktikan tingkat keberhasilan pemberian perintah suara berdasarkan jarak sangat bergantung dari jarak pemberian *sample* suara yang diberikan.

##### C. Pengujian Pemberian Perintah dari Orang yang Berbeda

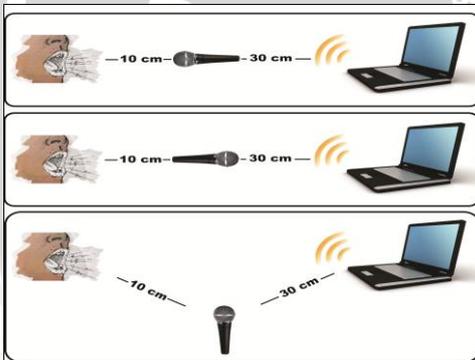
Tujuan dari pengujian ini adalah untuk mengetahui tingkat keberhasilan *EasyVR* dalam menerima perintah suara dari orang yang berbeda. Perintah diberikan oleh sepuluh orang dengan etnis Jawa dan berusia rata-rata 20 tahun hingga 23 tahun. Setiap pemberian perintah dilakukan 10 kali per perintah dengan jarak 10 cm dari sensor *microphone* pada setiap orang. Pengujian dilakukan dengan kondisi ideal atau *noise* yang sangat kecil.

Tabel 3. Hasil pengujian pemberian perintah dari orang yang berbeda

| Jumlah Keberhasilan Pemberian Perintah Suara |        |       |      |      |       |      |  |
|--|--------|-------|------|------|-------|------|--|
| Maju   | Mundur | Kanan | Kiri | Buka | Tutup | Stop |  |
| 10   | 10     | 10    | 10   | 10   | 10    | 10   |  |
| 9  | 10     | 9     | 10   | 9    | 10    | 9    |  |
| 10   | 10     | 9     | 10   | 10   | 10    | 10   |  |
| 10   | 9      | 10    | 10   | 10   | 8     | 9    |  |
| 9  | 8      | 10    | 10   | 10   | 9     | 10   |  |
| 9  | 10     | 10    | 9    | 10   | 9     | 10   |  |
| 8  | 9      | 10    | 10   | 10   | 8     | 9    |  |
| 10   | 8      | 9     | 10   | 9    | 9     | 10   |  |
| 9  | 7      | 9     | 10   | 9    | 8     | 10   |  |
| 9  | 9      | 9     | 9    | 10   | 10    | 10   |  |
| Rata-rata % Keberhasilan 94,37 %             |        |       |      |      |       |      |  |

Tabel 3 menunjukkan berapa kali perintah harus diucapkan agar *EasyVR* mampu mengolah perintah yang diberikan agar sama dengan hasil sampling. Hasil pengujian menunjukkan tingkat keberhasilan menerima perintah suara dari kondisi yang sudah ditentukan dan pengucapan perintah suara yang memiliki variasi sama dengan sample yang direkam mencapai 94,37 %. Kegagalan dalam pengolahan perintah suara disebabkan oleh pengucapan tutur kata yang kurang jelas dan terlalu besar atau terlalu kecil volume suara yang diberikan.

**D. Pengujian keberhasilan menerima perintah dengan adanya noise**



Gambar 9. Ilustrasi pengujian dengan posisi hadap microphone.

Tujuan dari pengujian ini adalah mengetahui tingkat keberhasilan menerima perintah dengan adanya *noise*. *Noise* yang diberikan berupa sebuah rekaman suara yang berisi kondisi bising pada saat Kontes Robot Indonesia Regional IV di gedung Robotika ITS. Jarak antara *noise* dengan *microphone* ditentukan, yaitu 30 cm dan jarak pemberian perintah adalah 10 cm. Pengujian dilakukan dengan posisi hadap *microphone* yang berbeda, yaitu menghadap pemberi perintah, menghadap sumber *noise*, dan diantara pemberi perintah dan sumber *noise*. Ilustrasi pengujian dengan posisi hadap microphone ditunjukkan dalam Gambar 9.

Tabel 4. Hasil pengujian keberhasilan menerima perintah dengan adanya *noise* dengan posisi *microphone* menghadap pemberi perintah.

| Noise (dB) | Keberhasilan menerima perintah terhadap noise |        |       |      |      |      |       |
|------------|---|--------|-------|------|------|------|-------|
|            | Maju  | Mundur | Kanan | Kiri | Stop | Buka | Tutup |
| 61 – 69    | 10  | 10     | 10    | 10   | 10   | 10   | 10    |
| 66 – 74    | 10  | 9      | 10    | 10   | 10   | 10   | 10    |
| 75 – 82    | 10  | 9      | 10    | 9    | 9    | 10   | 9     |
| 84 – 90    | 5   | 8      | 7     | 6    | 5    | 7    | 6     |
| 88 – 102   | 0   | 0      | 0     | 0    | 0    | 0    | 0     |

Tabel 5. Hasil pengujian keberhasilan menerima perintah dengan adanya *noise* dengan posisi *microphone* menghadap sumber *noise*.

| Noise (dB) | Keberhasilan menerima perintah terhadap noise |        |       |      |      |      |       |
|------------|---|--------|-------|------|------|------|-------|
|            | Maju  | Mundur | Kanan | Kiri | Stop | Buka | Tutup |
| 61 – 69    | 10  | 10     | 10    | 10   | 10   | 10   | 10    |
| 66 – 74    | 10  | 9      | 10    | 10   | 10   | 10   | 10    |
| 75 – 82    | 10  | 9      | 10    | 9    | 9    | 10   | 9     |
| 84 – 90    | 5   | 8      | 7     | 6    | 5    | 7    | 6     |
| 88 – 102   | 0   | 0      | 0     | 0    | 0    | 0    | 0     |

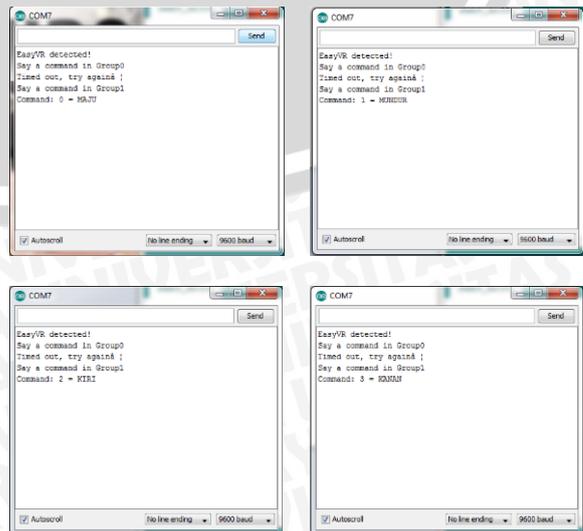
Tabel 6. Hasil pengujian keberhasilan menerima perintah dengan adanya *noise* dengan posisi *microphone* berada diantara pemberi perintah dan sumber *noise*.

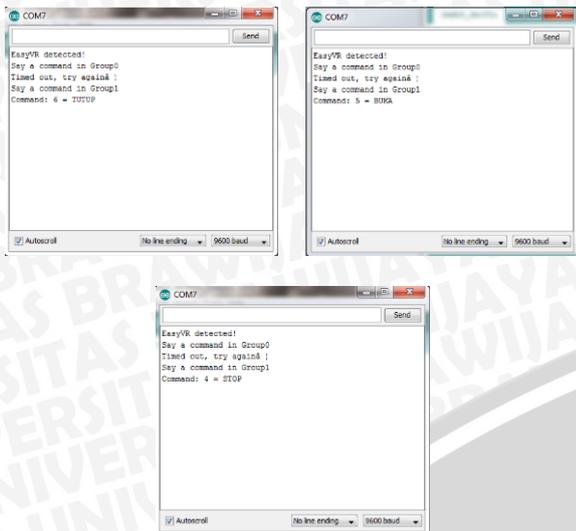
| Noise (dB) | Keberhasilan menerima perintah terhadap noise |        |       |      |      |      |       |
|------------|---|--------|-------|------|------|------|-------|
|            | Maju  | Mundur | Kanan | Kiri | Stop | Buka | Tutup |
| 61 – 69    | 6   | 7      | 5     | 6    | 5    | 7    | 7     |
| 66 – 74    | 3   | 5      | 4     | 4    | 5    | 5    | 6     |
| 75 – 82    | 0   | 1      | 0     | 0    | 0    | 2    | 0     |
| 84 – 90    | 0   | 0      | 0     | 0    | 0    | 0    | 0     |
| 88 – 102   | 0   | 0      | 0     | 0    | 0    | 0    | 0     |

Tabel 4, Tabel 5, dan Tabel 6 menunjukkan tingkat keberhasilan menerima perintah dengan adanya *noise* dengan posisi hadap *microphone* yang berbeda. Kegagalan dalam menerima perintah disebabkan oleh besarnya *noise* dalam keadaan tertentu. Jika dilihat dalam dapat disimpulkan bahwa semakin besar *noise* yang diberikan, maka akan semakin kecil tingkat keberhasilan dalam menerima perintah suara.

**E. Pengujian Melalui Serial Monitor pada Program Arduino**

Tujuan dari pengujian ini adalah untuk mengetahui apakah board *Arduino* UNO mampu menerima data ASCII dari *EasyVR* dan mengeksekusi data tersebut. Pengujian ini dilakukan dengan cara menghubungkan *Arduino* UNO yang sudah terpasang dengan perancangan sistem secara keseluruhan dengan PC atau laptop melalui kabel USB. Hasil pengujian dapat dilihat melalui Serial Monitor yang terdapat pada program *Arduino*.





Gambar 10. Tampilan pada Serial Monitor untuk pengujian navigasi atau pergerakan robot.

Gambar 4.1 menunjukkan respon dari setiap kata yang diucapkan. Dari hasil pengujian terlihat bahwa robot mampu menerima perintah berupa suara. Terbukti dari setiap pengucapan kata sesuai dengan command yang diberikan. Gambar 4.2 menunjukkan respon robot terhadap kata “maju” yang berarti bernavigasi jalan ke depan atau maju dengan adanya tanda yaitu LED berwarna merah menyala.

#### F. Pengujian Respon Robot Setelah Menerima Perintah dalam Satuan Waktu

Tujuan dari pengujian ini adalah untuk mengetahui berapa lama respons robot melakukan pergerakan atau navigasi terhadap perintah suara yang diberikan. Pengujian dilakukan dengan cara manual yaitu dengan menggunakan jam analog. Perhitungan waktu dimulai setelah perintah selesai diucapkan.

Tabel 7. Hasil pengujian rata-rata waktu respon robot setelah menerima perintah.

| Perintah | Waktu Respon |
|----------|--------------|
| Maju     | ≤ 1 detik    |
| Mundur   | ≤ 1 detik    |
| Kanan    | ≤ 1 detik    |
| Kiri     | ≤ 1 detik    |
| Buka     | ≤ 1 detik    |
| Tutup    | ≤ 1 detik    |
| Stop     | ≤ 1 detik    |

Tabel 7 menunjukkan bahwa rata-rata waktu respon dari setelah perintah diucapkan sampai pergerakan atau navigasi robot adalah kurang dari sama dengan 1 detik. Hal ini disebabkan berjalannya proses ketika *microphone* menerima perintah yang kemudian dicocokkan oleh *EasyVR* apakah perintah yang diberikan sesuai dengan *sample*. Jika cocok, *EasyVR* mengirim perintah ke *Arduino UNO* yang kemudian akan mengakses aktuator melalui *driver* motor L298N. Hal ini cukup bagus karena tidak membutuhkan waktu terlalu lama agar data dapat dieksekusi.

## VII. PENUTUP

Dari hasil perancangan dan pengujian yang telah dilakukan, maka dapat disimpulkan beberapa hal sebagai berikut:

1. Metode pengambilan *sample* suara pada modul *EasyVR* dilakukan dalam kondisi ideal atau tanpa adanya *noise* dan dilakukan sebanyak dua kali pengambilan suara dengan variasi pengucapan yang relatif sama pada setiap kata. Hal ini dilakukan sesuai dengan kemampuan *EasyVR* yang tidak bisa menerima pengucapan variasi suara kedua jika berbeda dengan variasi pengucapan suara pertama. Oleh karena itu, agar *EasyVR* dapat berfungsi atau dapat mengakses perintah suara, dibutuhkan variasi suara yang relatif sama dengan *sample*. Selain itu, *EasyVR commander* memberi batas waktu selama 5 detik untuk setiap pengucapan suara yang akan dijadikan *sample*.
2. Modul *EasyVR* memiliki tingkat keberhasilan sebesar 94,37 % jika menerima perintah suara dari orang yang berbeda dengan variasi pengucapan yang relatif sama. Kegagalan dalam pengolahan perintah suara disebabkan oleh pengucapan tutur kata yang kurang jelas dan terlalu besar atau terlalu kecil volume suara yang diberikan.
3. Keberhasilan modul *EasyVR* dalam menerima perintah suara dengan baik segi jarak ditentukan oleh jarak pemberian perintah suara yang sangat bergantung pada jarak pemberian *sample*.
4. Rata-rata waktu respon robot setelah pengucapan perintah ke pergerakan atau navigasi robot adalah kurang dari sama dengan 1 detik. Hal ini cukup bagus karena tidak membutuhkan terlalu waktu lama agar data dapat dieksekusi.
5. Penerapan sistem *voice recognition* pada robot pemindah objek dapat digunakan sebagai sistem navigasi robot meskipun belum mampu menggantikan penggunaan perangkat berupa *joystick*.

## DAFTAR PUSTAKA

- [1] Taufik, Ahmad Sul Khan. 2013. *Sistem Navigasi Waypoint pada Autonomous Mobile Robot*. Malang: Skripsi Jurusan Teknik Elektro FT-UB.
- [2] *EasyVR Datasheet*. <http://www.veear.eu/>. Diakses tanggal: 15 Juni 2013. Austria: TIGAL KG.
- [3] Akbar, Arnas Elmiawan. 2013. *Implementasi Sistem Navigasi Wall Following Menggunakan Kontroler PID dengan Metode Tuning pada Robot Kontes Robot Cerdas Indonesia (KRCI) Divisi Senior Beroda*. Malang. Skripsi Jurusan Teknik Elektro FT-UB.