

## **Sistem Remote Control Robot Beroda Menggunakan Teknologi Leap Motion**

**Supria<sup>1</sup>, Nurul Fahmi<sup>2</sup>**

<sup>1,2</sup>Teknik Informatika, Politeknik Negeri Bengkalis

<sup>1,2</sup>Jl. Bathin Alam, Sei. Alam Bengkalis, RIAU, telp. 0766012021

e-mail : <sup>1</sup>phiya@polbeng.ac.id, <sup>2</sup>nurulfahmi@polbeng.ac.id

### **Abstrak**

*Interaksi manusia dan komputer (IMK) adalah ilmu yang mempelajari bagaimana manusia bisa berinteraksi dengan komputer. Robot merupakan sebuah mesin komputer yang dapat membantu manusia dalam melakukan pekerjaan. Tetapi robot harus mendapatkan perintah atau instruksi dari manusia agar bisa melaksanakan sebuah aksi (action). Remote control adalah sebuah alat yang sering digunakan manusia dalam mengontrol robot untuk melakukan sebuah aksi. Banyak peneliti yang membuat remote control untuk mengontrol sebuah robot dengan menggunakan smartphone. Leap motion adalah sebuah teknologi baru yang dapat mendeteksi posisi titik koordinat tangan dan gerakan tangan, sehingga dapat digunakan sebagai remote control pada robot. Pada penelitian ini diusulkan sistem remote control robot beroda menggunakan teknologi leap motion. Leap motion akan digunakan untuk mendeteksi posisi titik-titik koordinat tangan. Titik-titik koordinat yang didapatkan dari leap motion akan digunakan untuk membentuk sebuah fitur (feature extraction). Bentuk fitur tersebut yang akan digunakan untuk mengontrol robot beroda, sehingga robot akan melakukan aksi berdasarkan bentuk fitur. Dari hasil uji coba yang telah dilakukan menunjukkan bahwa sistem ini dapat melakukan kontrol terhadap robot beroda dengan perintah maju, mundur, berhenti, belok kanan dan belok kiri. Dari hasil pengukuran akurasi terhadap sistem yang diusulkan didapatkan akurasi yaitu instruksi forward (90%), backward (90%), stop (100%), right (90%) dan left (90%), sehingga memiliki akurasi rata-rata 92%.*

**Kata Kunci:** remote control, robot beroda, leap motion, koordinat tangan, pembentukan fitur.

### **Abstract**

*Human and computer interaction is the study of how humans can interact with computers. Robot is a computer machine that can help humans in doing the job. But the robot must get orders or instructions from humans in order to carry out an action. Remote control is a tool often used by humans in controlling robots to perform an action. Many researchers create remote controls to control a robot using a smartphone. Leap motion is a new technology that can detect the position of hand coordinate points and hand movements, so it can be used as a remote control on the robot. In this research, it is proposed remote control system for robot wheel using leap motion technology. Leap motion will be used to detect the position of hand coordinate points. The coordinate points obtained from leap motion will be used to form a feature (feature extraction). The shape of the feature will be used to control the wheeled robot, so that the robot will perform the action based on the feature form. From the results of trials that have been done indicate that this system can control the wheeled robot with the command forward, backward, stop, turn right and turn left. From the measurement of accuracy to the proposed system is obtained the accuracy of forward 90%, backward 90%, stop 100%, right 90% and left 90%, so has an average accuracy of 92%.*

**Keyword :** remote control, whelled robot, leap motion, hand coordinates, feature extraction.

## 1. Pendahuluan

Interaksi manusia dan komputer adalah ilmu yang mempelajari bagaimana manusia berinteraksi dengan komputer. IMK adalah disiplin ilmu yang mempelajari tentang desain, evaluasi dan implementasi sistem komputer interaktif yang digunakan manusia dengan memperhatikan fenomena di lingkungan sekitarnya [1]. Seperti halnya yang terjadi pada sistem *remote* kontrol seperti *remote* kontrol mainan menggunakan *joystick* dan *remote* kontrol robot menggunakan *joystick*.

Beberapa penelitian telah mengembangkan sistem *remote control* menggunakan aplikasi android dengan komunikasi *bluetooth* [2]. Sistem ini dapat mengontrol robot beroda dengan instruksi maju, mundur, berhenti, belok kanan dan belok kiri. Adapun akurasi dari hasil uji coba terhadap sistem tersebut sangat baik. Sistem ini menggunakan *mikrocontroller* sebagai kontrolnya [3][4][5][6].

*Leap motion* adalah sebuah sensor yang berfungsi sebagai alat untuk mendeteksi tangan dan jari. Telah banyak yang membahas *leap motion* pada penelitian [7][8][9]. *Leap motion* menggunakan sensor optik dan cahaya inframerah. Sensor ini diarahkan sepanjang arah sumbu-Y yang memiliki bidang pandang 150 derajat. Rentang yang efektif untuk deteksi adalah antara 25 mm sampai dengan 600 mm. *Leap motion* sebagai sistem yang dapat mendeteksi tangan, jari dan alat yang berada pada bidang pandang *leap motion*, memberikan perubahan data pada sebuah data atau *frame* data. Setiap objek *frame* menunjukkan *frame* yang mengandung daftar data deteksi seperti tangan, jari dan alat yang berbentuk seperti jari, serta deteksi gerakan dan faktor yang menggambarkan gerakan selama waktu proses deteksi. Objek *frame* merupakan akar dari model data *leap motion*. Pada *motion tracking data* terdapat beberapa objek yaitu *hands*, *arms*, *fingers*, *tools*, *gestures* dan *motions*.

Beberapa penelitian telah menggunakan *leap motion* untuk mendeteksi koordinat tangan seperti pengenalan bahasa isyarat menggunakan fitur statis dan dinamis berbasis LGCNN[10], Pengenalan bahasa isyarat Amerika menggunakan *leap motion*[11], Pengenalan bahasa isyarat Arab menggunakan *leap motion*[12], Deteksi gerakan tangan menggunakan *leap motion*[13].

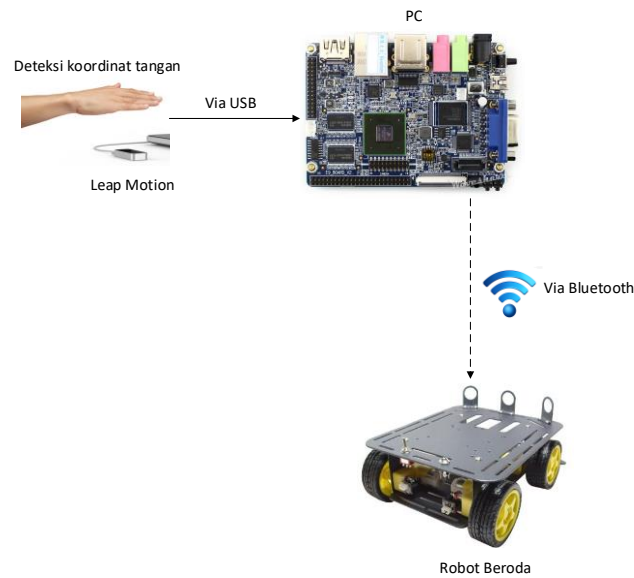
Robot beroda adalah sebuah robot yang dapat berjalan dengan menggunakan roda. Robot ini umumnya memiliki gerak maju, mundur, berhenti, belok kanan dan belok kiri. Robot ini dibuat dengan menggunakan *mikrocontroller* sebagai alat pengontrolnya. Robot tidak dapat melakukan aksinya jika tidak dikendalikan oleh manusia.

Pada penelitian ini diusulkan Sistem *Remote Control Robot Beroda Menggunakan Teknologi Leap Motion*. *Remote* kontrol akan dibuat dengan menggunakan *leap motion*. *Leap motion* digunakan untuk mendeteksi posisi titik-titik koordinat (XYZ) tulang-tulang pada tangan dan jari (*Metacarpal*, *Proximal Phalanx*, *Intermediate Phalanx*, *Distal Phalanx* dan *Palm*). Data koordinat tersebut akan digunakan untuk membentuk fitur. Fitur yang telah dibentuk akan diambil nilai rata-ratanya. Nilai rata-rata fitur tersebut akan dijadikan sebagai nilai untuk menentukan range atau area, dimana ada 5 range atau area (nilai range *forward*, *backward*, *stop*, *right* dan *left*). Setiap nilai uji coba akan dicocokkan dengan nilai range yang ada untuk menjalankan robot beroda. Sistem ini diharapkan dapat mengontrol robot beroda dengan instruksi maju, mundur, berhenti, belok kanan, dan belok kiri. Adapun untuk mengukur keberhasilan dari sistem ini akan dilakukan uji coba dengan mengukur tingkat akurasi.

## 2. Metode Penelitian

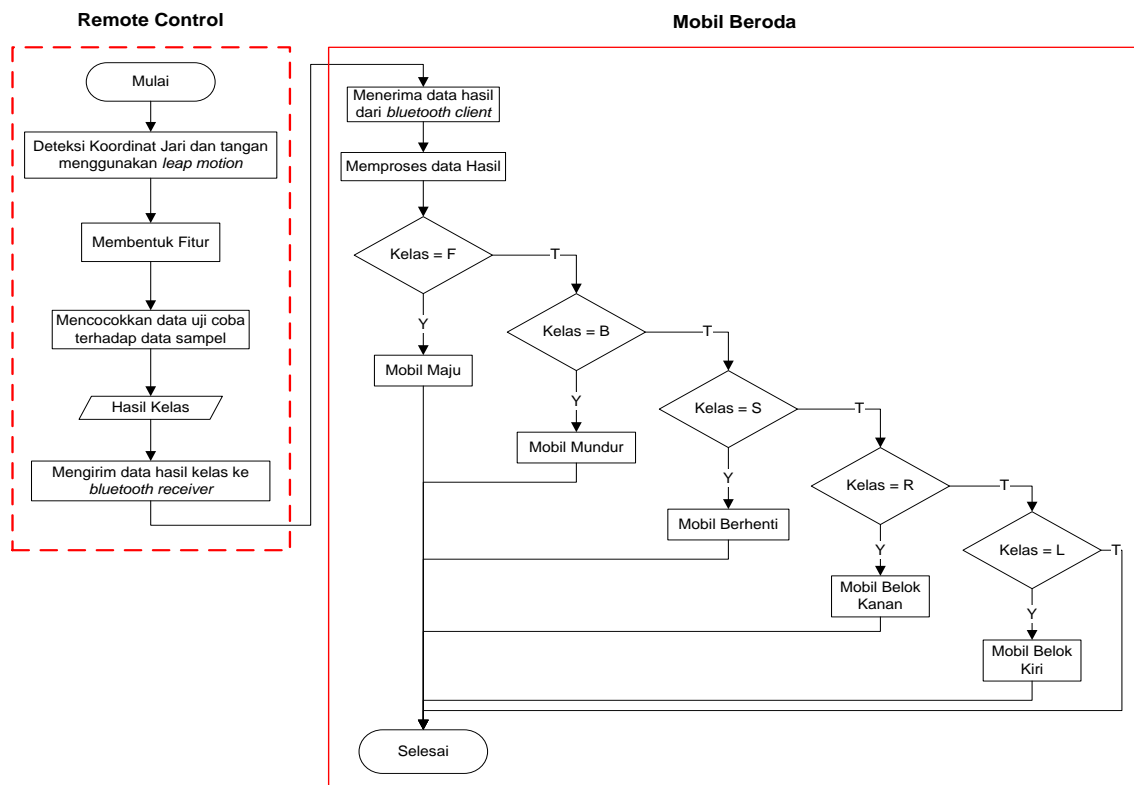
Pada tahap ini akan dipaparkan gambaran sistem yang diusulkan. Ada beberapa tahap pada sistem yang diusulkan, yaitu tahap deteksi koordinat tangan, perancangan pembentukan fitur, perancangan penentuan range, perancangan pengiriman data dari PC ke robot beroda melalui *bluetooth*, dan perancangan robot beroda. Secara umum gambaran sistem yang diusulkan dapat dilihat pada Gambar 1. Ada beberapa komponen yang digunakan dalam penelitian ini yaitu *leap motion*, *mini PC* dan Robot beroda.

---



Gambar 1. Gambaran sistem yang diusulkan.

Secara umum ada dua bagian alur sistem dari sistem yang diusulkan yaitu alur sistem pada bagian *remote control* dan alur sistem pada bagian robot beroda. Adapun alur kedua bagian alur sistem pada penelitian ini dapat dilihat pada Gambar 2.



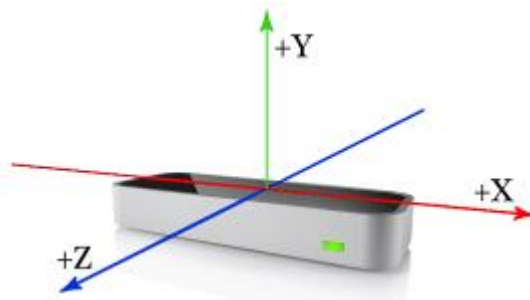
Gambar 2. Alur sistem yang diusulkan.

### Remote Control

Pada remote control terdapat beberapa tahapan untuk mencapai tujuan. Adapun tahapan-tahapan pada remote control adalah deteksi koordinat tangan, pembentukan fitur, pencocokan data uji coba dengan data sampel, hasil kelas, dan pengiriman data kelas ke *Bluetooth receiver*.

#### Deteksi koordinat tangan dan jari.

Untuk mendapatkan titik-titik koordinat tangan dilakukan dengan menggunakan *leap motion*. *Leap motion* akan mendeteksi titik-titik koordinat (bidang x, y, z) pada tangan seperti ditunjukkan pada Gambar 3. Titik-titik koordinat yang akan digunakan adalah ujung jari (*distal palanx*) dan titik tengah tangan (*palm position*).



Gambar 3. Sistem koordinat leap motion.

#### Pembentukan fitur

Data koordinat tangan dan jari yang didapatkan dari deteksi menggunakan *leap motion* akan dijadikan fitur. Adapun fitur yang dibentuk adalah hasil pengukuran jarak antara ujung pada setiap jari (*distal palanx*) terhadap titik tengah tangan (*palm*). Untuk mengukur jarak tersebut dengan menggunakan *Euclidean distance*.

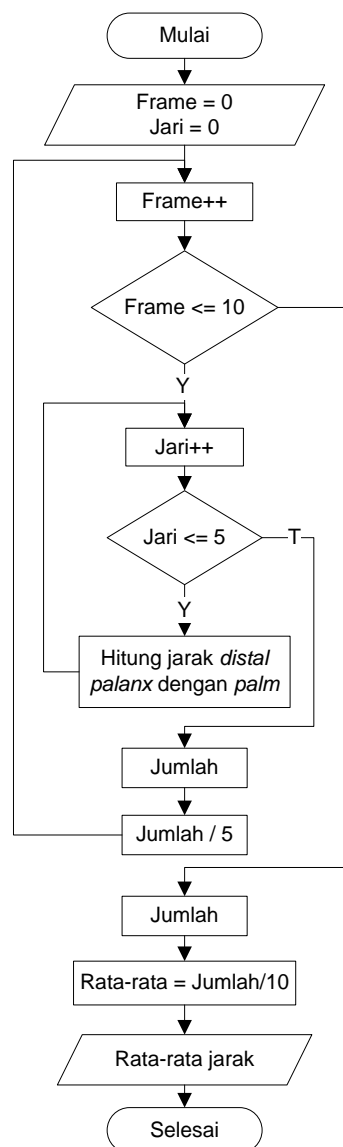
*Euclidean distance* adalah perhitungan jarak dari 2 buah titik dalam [Euclidean space](#). Euclidean space diperkenalkan oleh [Euclid](#), seorang matematikawan dari Yunani sekitar tahun 300 B.C.E. untuk mempelajari hubungan antara sudut dan jarak. Euclidean ini berkaitan dengan Teorema Phytagoras dan biasanya diterapkan pada 1, 2 dan 3 dimensi. Tapi juga sederhana jika diterapkan pada dimensi yang lebih tinggi. Persamaan *Euclidean distance* dapat dilihat pada persamaan 1.

$$d(xyz) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad (1)$$

dimana :

- $d(xyz)$  = jarak ruang tiga bidang x, y, dan z.
- x = nilai x pada bidang x, y, dan z.
- y = nilai y pada bidang x, y, dan z.
- z = nilai z pada bidang x, y, dan z.

Proses pembentukan fitur dilakukan dengan mengukur jarak setiap titik koordinat *distal palanx* terhadap *palm position*. Data yang dignakan untuk membentuk fitur adalah sebanyak 10 frame. Setiap frame akan dilakukan pengukuran jarak antara setiap titik koordinat *distal palanx* terhadap *palm position*. Dari hasil perhitungan jarak pada 10 frame tersebut akan diambil nilai rata-ratanya. Adapun proses perhitungan dapat ditunjukkan pada Gambar 4.



Gambar 4. Proses perhitungan jarak *distal palanx* terhadap *palm* pada 10 *frame*.

#### Pencocokan data ujicoba dengan data sampel.

Hasil dari pembentukan fitur berupa nilai rata-rata dari perhitungan jarak antara *distal palanx* dengan *palm position* dari 10 *frame*. Nilai rata-rata tersebut akan dicocokkan dengan data range hasil dari proses penentuan range. Data range berupa data-data hasil penentuan range yang berjumlah 5 kelas yaitu maju, mundur, berhenti, belok kanan, dan belok kiri. Hasil dari pencocokan data uji coba dengan data range adalah seperti ditunjukkan pada tabel 1.

Tabel 1 Data kelas

No	Kelas	Keterangan
1	F	<i>Forward</i> / maju
2	B	<i>Backward</i> / mundur
3	S	<i>Stop</i> / berhenti
4	R	<i>Right</i> / belok kanan
5	L	<i>Left</i> / belok kiri

Adapun hasil penentuan range nilai rata-rata dapat dilihat pada Algoritma 1.

Algoritma 1. Penentuan range 5 instruksi.

```

if (Rata < 48) {
    Kirim data ke bluetooth "R"
} else if (Rata > 48 && Rata < 52) {
    Kirim data ke bluetooth "L"
} else if (Rata > 52 && Rata < 56) {
    Kirim data ke bluetooth "B"
} else if (Rata > 56 && Rata < 70) {
    Kirim data ke bluetooth "F"
} else if (Rata > 70) {
    Kirim data ke bluetooth "S"
}

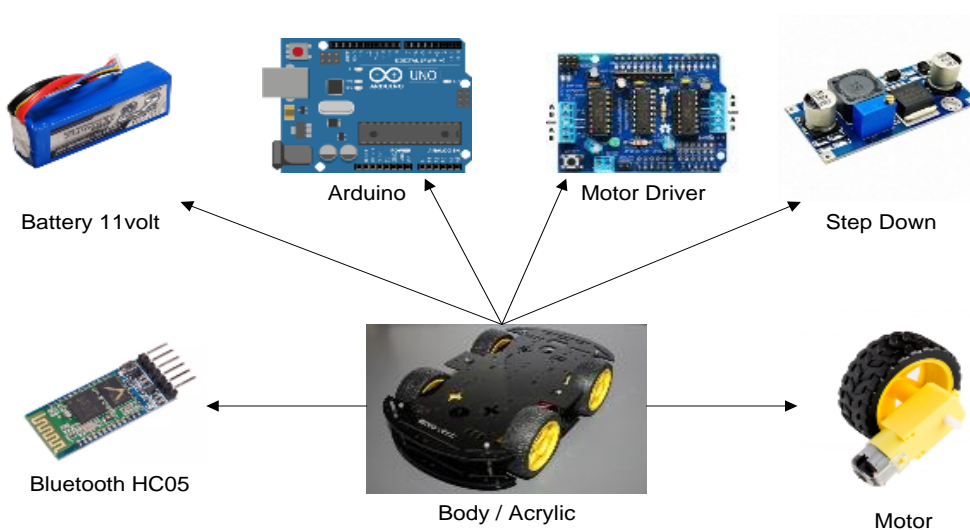
```

#### Pengiriman ke *bluetooth receiver*.

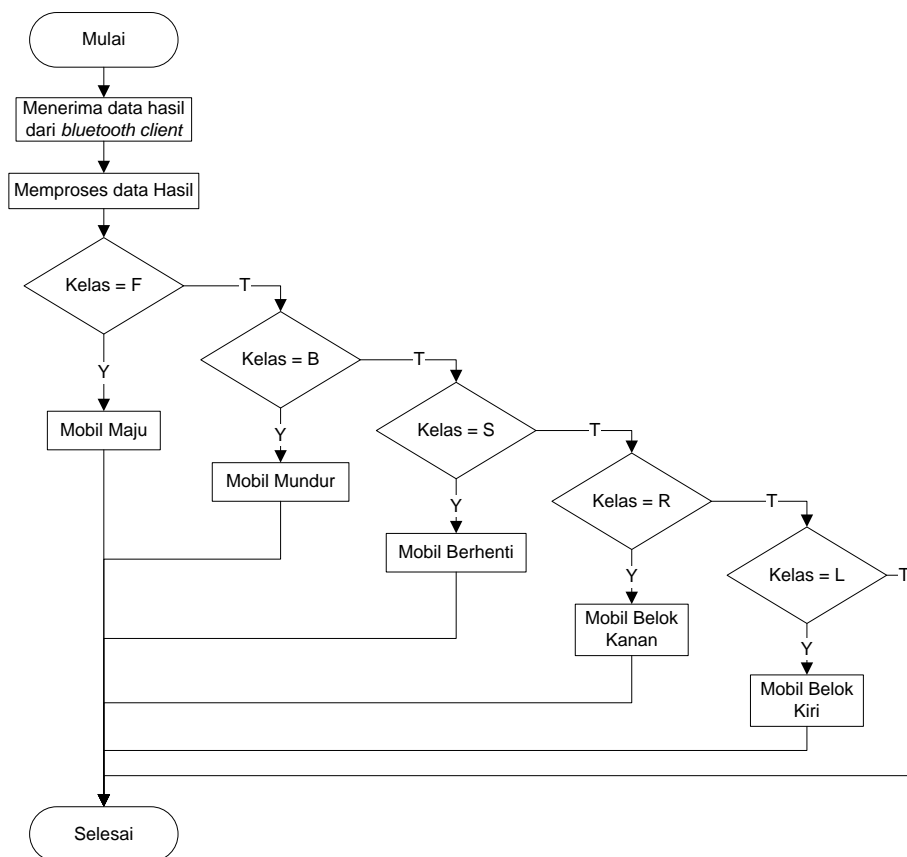
Hasil kelas yang didapatkan dari proses pencocokan akan dikirim ke robot. Proses pengiriman dilakukan melalui komunikasi *bluetooth*. Dalam hal ini ada dua sisi *bluetooth* yaitu sebagai *bluetooth transfer* dan *bluetooth receiver*. *Bluetooth transfer* adalah *bluetooth* yang terdapat pada PC, sedangkan *bluetooth receiver* terdapat pada robot. Jadi PC akan mengirimkan kelas (F, B, S, R, atau L) ke robot.

#### Robot Beroda

Pada robot beroda terdapat beberapa komponen yaitu *bluetooth receiver*, *mikrokontroller*, *motor driver*, *power*, dan *motor*. *Bluetooth receiver* berfungsi sebagai penerima data yang dikirim dari *bluetooth transfer* yang ada pada PC. *Mikrokontroller* digunakan sebagai pengontrol pada robot, sehingga semua komponen yang ada pada robot di kontrol oleh *mikrokontroller*. *Motor driver* digunakan untuk mengontrol *motor* berdasarkan instruksi yang diberikan oleh *mikrokontroller*. *Motor* berfungsi sebagai roda pada robot. Beberapa komponen-komponen yang digunakan untuk membuat robot beroda dapat ditunjukkan pada Gambar 5. Adapun alur proses yang ada pada robot dapat dilihat pada Gambar 6.



Gambar 5. Komponen-komponen robot beroda.



Gambar 6. Alur sistem pada robot beroda.

### 3. Hasil dan Pembahasan

Pada tahap ini akan dilakukan uji coba terhadap sistem yang diusulkan. Ujicoba dilakukan dengan menggunakan data nilai range. Data nilai range didapat dari proses perekaman pada saat penentuan range. Penentuan range pada setiap range ditentukan dengan menentukan nilai range minimal dan nilai range maksimal. Data hasil penentuan range dapat dilihat pada Tabel 2.

Tabel 2. Nilai range atau *mapping area*

No	Simbol	Keterangan	Min	Mak
1	F	<i>Forward</i> / maju	40	48
2	B	<i>Backward</i> / mundur	48	52
3	S	<i>Stop</i> / berhenti	52	56
4	R	<i>Right</i> / belok kanan	56	70
5	L	<i>Left</i> / belok kiri	70	90

Adapun spesifikasi computer yang digunakan dalam pengujian ini adalah Laptop *Lenovo G40 AMD A8*.

#### Implementasi

Hasil uji coba dilakukan dengan mengontrol robot beroda dengan menggunakan sistem yang diusulkan. Adapun uji coba dilakukan dengan 5 instruksi yaitu *forward*, *backward*, *stop*, *right* dan *left*.



Gambar 7. Hasil implementasi sistem yang diusulkan

1. *Forward* / maju, hasil uji coba dengan instruksi *forward* dapat dilihat pada Gambar 8.



Gambar 8. Hasil implementasi dengan instruksi *forward*.

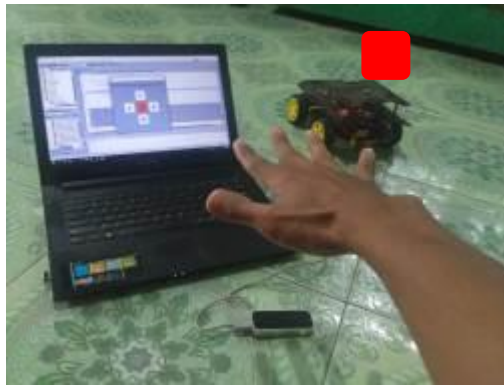
2. *Backward* / mundur, hasil uji coba dengan instruksi *backward* dapat dilihat pada Gambar 9.



Gambar 9. Hasil implementasi dengan instruksi *backward*.

3. *Stop* / Berhenti, hasil uji coba dengan instruksi *stop* dapat dilihat pada Gambar 10.





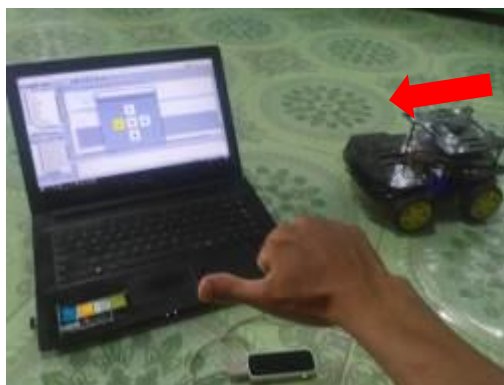
Gambar 10. Hasil implementasi dengan instruksi *stop*.

4. *Right* / belok kanan, hasil uji coba dengan instruksi *right* dapat dilihat pada Gambar 11.



Gambar 11. Hasil implementasi dengan instruksi *right*.

5. *Left* / belok kiri, hasil uji coba dengan instruksi *left* dapat dilihat pada Gambar 12.



Gambar 12. Hasil implementasi dengan instruksi *left*.

### Hasil Uji Coba

Untuk mengukur keberhasilan sistem yang diusulkan, maka dilakukan uji coba terhadap sistem yang dibuat. Untuk mengukur tingkat keberhasilan dilakukan dengan menggunakan akurasi dengan persamaan akurasi yang dapat dilihat pada persamaan 2.

$$Akurasi = \frac{\text{jumlah ujicoba benar}}{\text{total ujicoba}} \times 100\% \quad (2)$$

Setiap instruksi dilakukan uji coba sebanyak 10 kali ( 5 instruksi dengan jumlah total uji coba 50 kali). Adapun hasil uji coba yang dilakukan dapat dilihat pada Tabel 3.

**Tabel 3 Hasil uji coba**

Kelas	Total ujicoba	Ujicoba benar	Akurasi (%)
F	10	9	90
B	10	8	90
S	10	10	100
R	10	9	90
L	10	9	90
Akurasi rata-rata			92

### Pembahasan

Dari beberapa hasil pengujian yang telah dilakukan pada sistem yang diusulkan menunjukkan bahwa sistem yang diusulkan dapat bekerja sesuai dengan yang diharapkan. Hasil ujicoba dengan instruksi *forward* memiliki akurasi 90% dengan 10 kali percobaan, yaitu 9 kali percobaan benar dan 1 kali percobaan dideteksi sebagai instruksi *backward*. Hasil ujicoba dengan instruksi *backward* memiliki akurasi 90% dengan 10 kali percobaan, yaitu 9 kali percobaan benar dan 1 kali percobaan dideteksi sebagai instruksi *forward*. Hasil ujicoba dengan instruksi *stop* memiliki akurasi 100% dengan 10 kali percobaan. Hasil ujicoba dengan instruksi *right* memiliki akurasi 90% dengan 10 kali percobaan, yaitu 9 kali percobaan benar dan 1 kali percobaan dideteksi sebagai instruksi *left*. Hasil ujicoba dengan instruksi *left* memiliki akurasi 90% dengan 10 kali percobaan, yaitu 9 kali percobaan benar dan 1 kali percobaan dideteksi sebagai instruksi *right*. Adapun hasil akurasi rata-rata percobaan dari semua instruksi memiliki akurasi 92%.

*Leap Motion* merupakan perangkat yang menggunakan sensor *infrared*. Sensor mendeteksi posisi titik koordinat pada setiap tulang pada tangan dan jari. Untuk mendeteksi titik koordinat tersebut, harus dalam kondisi pencahayaan yang stabil dan tidak ada halangan. Ketika tulang yang dideteksi terhalang oleh suatu benda atau jari lain maka hasil proses deteksi tidak optimal, sehingga pembentukan fitur tidak mendapatkan hasil dengan baik. Seperti halnya yang terjadi pada uji coba sistem yang diusulkan. Ketika uji coba dengan instruksi *forward* maka akan terdeteksi sebagai instruksi *backward*. Hal ini terjadi karena nilai rata-rata fitur *forward* hampir sama dengan nilai rata-rata fitur *backward*. Begitu juga dengan instruksi *right* dan *left*. Nilai rata-rata fitur *right* dan *left* hampir sama. Hal ini terjadi karena beberapa kemungkinan seperti spesifikasi komputer yang digunakan rendah sehingga kinerja dari *leap motion* tidak optimal. *Leap motion* merupakan sensor yang bekerja secara *real time*, sehingga membutuhkan spesifikasi komputer yang tinggi.

#### 4. Kesimpulan

Berdasarkan penelitian dan perancangan sistem yang diusulkan, dan hasil uji coba yang telah dilakukan maka dapat ditarik kesimpulan bahwa sistem *remote control* yang diusulkan adalah sistem untuk mengontrol robot beroda dengan menggunakan *leap motion*. *Leap motion* digunakan untuk mendeteksi posisi titik-titik koordinat tangan dan jari. Pembentukan fitur dilakukan dengan mengukur jarak dari *palm position* terhadap *distal palanx* pada setiap jari tangan dengan menggunakan persamaan *euclidean distance*. Sistem yang diusulkan dapat mengontrol robot beroda dengan 5 instruksi yaitu maju, mundur, berhenti, belok kanan, dan belok kiri. Dari hasil uji coba yang telah dilakukan menunjukkan bahwa sistem ini memiliki akurasi rata-rata 92%.

Adapun saran untuk pengembangan kedepan dari sistem yang diusulkan adalah ditambahnya pembentukan fitur baru, sehingga dapat meningkatkan akurasi pengenalan. Menggunakan metode klasifikasi untuk menentukan kelas dari sebuah fitur yang telah dibentuk berdasarkan data *sampel*. Menambah instruksi lain yang dibutuhkan robot beroda selain instruksi yang diusulkan (maju, mundur, berhenti, belok kanan, dan belok kiri).

#### Daftar Pustaka

- [1] I. S. MacKenzie, *Human-Computer Interaction*, 1st ed. United States America: Elsevier, 2013.
- [2] A. . Arvindan and K. D, "Experimental Investigation of Remote Control Via Android Smart Phone of Arduino-Based Automated Irrigation System Using Moisture Sensor," pp. 168–175, 2016.
- [3] Y. Hendriana and R. Hardi, "Remote Control System as Serial Communications Mobile using a Microcontroller," 2016.
- [4] E. D. Tica, L. A. Perisoara, and A. Vasile, "An Arduino Platform for Remote Control and Bus Testing of Programmable Instruments." .
- [5] J. P. Ventura, N. A. Cruz, and F. P. Lima, "A remote monitoring and control system for ecosystem replication experiments," 2016.
- [6] A. Widiyanto and N. N, "Rancang Bangun Mobil Remote Control Android dengan Arduino," pp. 50–61.
- [7] Y. Curiel-raza and O. Icasio-hern, "Leap Motion Controller Three Dimensional Verification and Polynomial Correction," *Measurement*, 2016.
- [8] R. Katahira and M. Soga, "Development and Evaluation of a System for AR enabling Realistic Display of Gripping Motions using Leap Motion Controller," *Procedia - Procedia Comput. Sci.*, vol. 60, pp. 1595–1603, 2015.
- [9] I. Oropesa, T. L. de Jong, P. Sánchez-González, J. Dankelman, and E. J. Gómez, "Feasibility of tracking laparoscopic instruments in a box trainer using a Leap Motion Controller," *Measurement*, vol. 80, pp. 115–124, 2016.
- [10] S. S, D. HeruMurti, and W. Nurul Khotimah, "Pengenalan sistem isyarat bahasa indonesia menggunakan kombinasi fitur statis dan fitur dinamis lmc berbasis l-gcnn," pp. 217–230.
- [11] K. Fok, N. Ganganath, C. Cheng, and C. K. Tse, "A Real-Time ASL Recognition System Using Leap Motion Sensors," 2015.
- [12] M. Mohandes, S. Aliyu, and M. Deriche, "Arabic Sign Language Recognition using the Leap Motion Controller," 2014.
- [13] N. C K and U. Gudukbay, "A hand gesture recognition technique for human – computer interaction q," vol. 28, pp. 97–104, 2015.