

## Implementasi Metode *Complementary Filter* pada Pengendali Robot Mobil menggunakan Gestur Tangan Manusia

Samkhyha Aparigraha<sup>1</sup>, Wijaya Kurniawan<sup>2</sup>, Agung Setia Budi<sup>3</sup>

Program Studi Teknik Komputer, Fakultas Ilmu Komputer, Universitas Brawijaya  
Email: <sup>1</sup>samkhyapariagraha@gmail.com, <sup>2</sup>wjaykurnia@ub.ac.id, <sup>3</sup>agungsetiabudi@ub.ac.id

### Abstrak

Interaksi manusia dan komputer merupakan sebuah disiplin ilmu yang mempelajari perancangan, implementasi dan evaluasi sistem komputasi interaktif dari berbagai aspek terkait. Dalam perkembangan teknologi interaksi manusia dan komputer saat ini berkembang begitu sangat pesat, khususnya interaksi antara manusia dengan robot. Robotika adalah salah satu bidang teknologi yang masih berkembang sampai saat ini. Seiring perkembangannya, kemudian muncul istilah yang dikenal mobile robot, *humanoid*, *animaloid* dan lain sebagainya. Di dalam pengendaliannya, robot mobil dibagi menjadi dua jenis yaitu, robot otomatis yang dapat bergerak sendiri berdasarkan perintah-perintah yang telah dituliskan dalam program pengendalinya dan robot teleoperasi yang bergerak berdasarkan perintah-perintah yang dikirimkan secara manual, baik dengan kabel maupun tanpa kabel atau *Wireless*. Untuk membangun robot mobil yang dapat di kendalikan dengan menggunakan gestur tangan manusia, diperlukan modul MPU6050 GY-521 sebagai *input* data berupa *accelerometer* yang digunakan untuk mengukur percepatan gerak dan *gyroscope* yang digunakan untuk mengukur kecepatan sudut putaran, kemudian nantinya data tersebut akan di filter menggunakan metode *Complementary Filter* untuk menghilangkan *noise*, serta modul NRF24L01 sebagai komunikasi pertukaran data antara manusia dan robot mobil tersebut. Berdasarkan hasil dari penelitian yang telah di lakukan, dapat di simpulkan bahwa *complementary filter* dapat meredam *noise* dari nilai *roll* sebesar 21.8% dan *pitch* sebesar 26.5% sehingga dapat membuat sensor mpu6050 lebih stabil daripada tanpa menggunakan menggunakan filter dan metode *proporsional integratif derivatif* yang ditanamkan pada kontroller arduino nano dengan nilai  $K_p=6.98$ ,  $K_i=4.61$  dan  $K_d=1.15$  membuat sistem dapat berjalan sesuai yang diinginkan.

**Kata kunci:** *pengendali robot mobil, gestur tangan, robot mobil, complementary filter, PID*

### Abstract

*Human and computer interaction is one of the scientific disciplines that study the design, implementation, and evaluation of interactive computing systems from various related aspects. The current development of the technology of human and computer interaction is progressing very rapidly, especially the interaction between human and robot. Robotics is one area of technology that is still evolving today. Along with its development, more new terms have been introduced, such as mobile robots, humanoids, animaloids, et cetera. Within the control process, mobile robots are divided into two types, namely, automated robots and teleoperating robots. Automated robots can move themselves based on the commands that have been written in the control program, whereas teleoperating robots could move based on commands that are sent manually, either using wired or wireless. To develop a mobile robot that can be controlled using human-hand gestures, the MPU6050 GY-521 module is needed to input the data – in the form of an accelerometer and gyroscope. An accelerometer is used to measure the acceleration of motion, while a gyroscope is used to measure the angular velocity of the rotation. Afterwards, the data will be filtered using a complementary filter method to eliminate noise, which then will be processed using NRF24L01 module as a communication exchange of data between humans and the car robot. Based on the results of this research, it can be concluded that the complementary filter method can reduce the noise of the roll value of 21.8% and the pitch of 26.5% hence it can make the mpu6050 sensor more stable than without using a filter and integrative derivative proportional method which is implanted in the arduino nano controller with a value of  $K_p = 6.98$ ,  $K_i = 4.61$ , and  $K_d = 1.15$ , to make the system perfectly run as desired.*

**Keywords:** *car robot controller, hand-gestures, car robot, complementary filter, PID*

## 1. PENDAHULUAN

Interaksi manusia dan komputer merupakan sebuah disiplin ilmu yang mempelajari perancangan, implementasi dan evaluasi sistem komputasi interaktif dari berbagai aspek terkait. Dalam perkembangan teknologi interaksi manusia dan komputer saat ini berkembang begitu sangat pesat, khususnya interaksi antara manusia dengan robot (Siregar, 2017).

Robotika adalah salah satu bidang teknologi yang masih berkembang sampai saat ini. Robotika merupakan sebuah ilmu multidisiplin yang mempelajari tentang pembuatan sebuah mesin untuk meringankan pekerjaan manusia. Pada mulanya robot hanya digunakan pada aplikasi industri, sehingga terdapat istilah robot industri atau robot manipulator. Seiring perkembangannya, kemudian muncul istilah yang dikenal *mobile robot*, *humanoid*, *animaloid* dan lain sebagainya. Fungsi robot yang tadinya digunakan untuk keperluan industri pun mengalami perubahan, di mana robot dapat membantu dalam bidang lainnya seperti bidang kedokteran, militer bahkan juga sebagai alat hiburan (Umam, 2013).

Mobile robot merupakan salah satu kategori robot yang dapat menunjukkan adanya fungsi untuk berpindah tempat. Dalam kebutuhan penggunaannya robot bergerak dibagi menjadi tiga jenis yaitu, robot daratan, robot air (*Submarine Robot*) dan robot udara (*Aerial Robot*). Pada dasarnya robot bergerak menggunakan roda (*Wheel*) untuk dapat berpindah tempat (Iqbal & , 2013).

Dalam suatu pengendalian sistem robot manual, sebagai bentuk cara komunikasi atau interaksi manusia dengan robot, maka perlu adanya data-data yang di *input* kan kedalam suatu sistem robot untuk dibaca oleh robot tersebut (Sutisna & Sutisna, 2014).

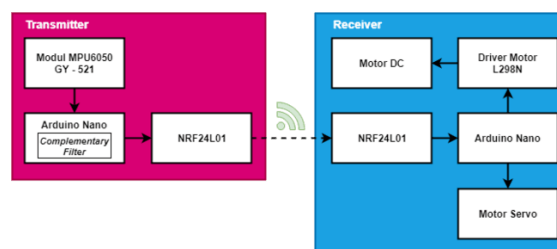
Pada umumnya, dalam pembuatan pengendalian robot menggunakan sensor MPU6050 GY-521 sebagai *input* data-nya. Sensor ini merupakan penggabungan dari *accelerometer* dan *gyroscope* dalam satu perangkat. Keluaran pada sensor ini berupa nilai *pitch*, *roll* dan *yaw* dengan cara memberikan nilai data elektrik yang di ukur dari perbedaan sinyal listrik pada tiap sisi di dalam perangkat sensor (Cahyo, 2018). Namun keluaran nilai *pitch*, *roll* dan *yaw* dari sensor MPU6050 GY-

521 pasti memiliki *noise*. Untuk meredam *noise* tersebut di butuhkan sebuah filter, agar hasilnya lebih stabil dengan *noise* dalam batas toleransi. Salah satu contoh dari filter penyaringan sinyal adalah Complementary filter. Complementary filter merupakan algoritma filter dari penggabungan *high pass filter* dan *low pass filter*. Complementary filter akan menyaring sinyal digital dan analog, kemudian memberikan nilai yang lebih stabil (Cahyo, 2018).

Terkait dengan pembahasan latar belakang diatas, maka penulis ingin mengimplementasikan sebuah robot bergerak, khususnya robot mobil yang dapat di kendalikan dengan menggunakan gestur tangan manusia. Modul MPU6050 GY-521 di perlukan sebagai *input* data, berupa *accelerometer* yang digunakan untuk mengukur percepatan gerak dan gyroscope yang digunakan untuk mengukur kecepatan sudut putaran, kemudian nantinya data tersebut akan di filter menggunakan metode Complementary Filter untuk menghilangkan *noise*, serta modul NRF24L01 sebagai komunikasi pertukaran data antara manusia dan robot mobil tersebut.

## 2. PERANCANGAN DAN IMPLEMENTASI

Dalam perancangan sistem ini menjelaskan tentang gambaran umum perancangan sistem seperti pada Gambar 1, perancangan perangkat keras dan perancangan perangkat lunak pada penelitian ini.



Gambar 1. Diagram Blok Perancangan Sistem

### 2.1. Perancangan Perangkat Keras

Perancangan perangkat keras disini, menjelaskan tentang bagaimana perancangan keseluruhan perangkat keras pada penelitian ini. Perancangan perangkat keras meliputi perancangan perangkat keras *transmitter*, perancangan desain sarung tangan, perancangan perangkat keras *receiver* dan perancangan desain

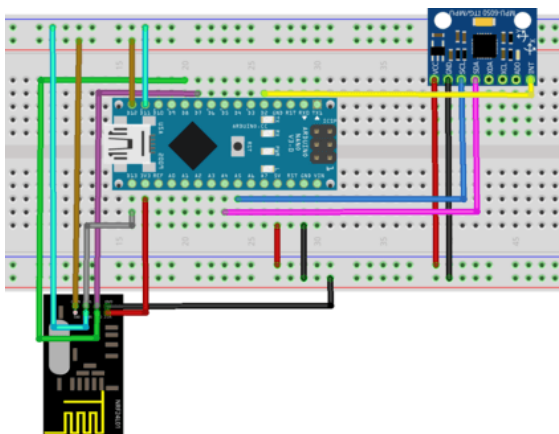
rangka robot mobil.

**2.1.1. Perangkat Keras Transmitter**

Perancangan perangkat keras *transmitter* ini menjelaskan tentang perancangan pada sensor MPU6050 GY-521 dan modul NRF24L01 dengan mikrokontroler Arduino Nano. Sensor MPU6050 GY-521 memerlukan 1 pin vcc 5V serta 1 pin ground, 2 pin analog dan 1 pin digital yang terhubung dengan mikrokontroler Arduino Nano. Modul NRF24L01 memerlukan 1 pin vcc 3,3V serta 1 pin ground dan 5 pin digital yang terhubung dengan mikrokontroler Arduino Nano. Untuk lebih jelasnya dapat di lihat pada Tabel 1 dan Gambar 2 untuk keseluruhan perancangan rangkaian pada *transmitter*.

Tabel 1. Perancangan Pin *Transmitter*

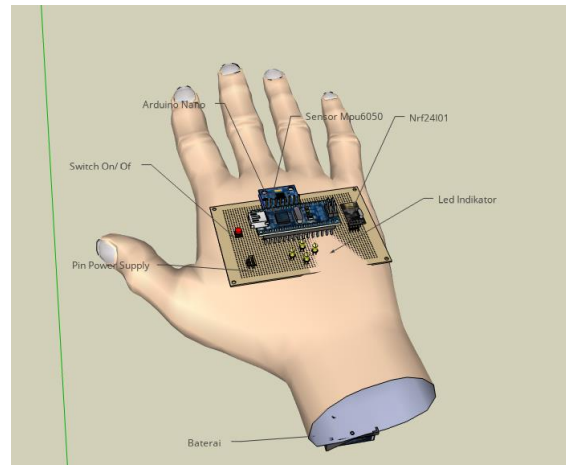
Arduino Nano	MPU6050 GY-521	NRF24L01
Pin 5V	Pin VCC	-
Pin A5	Pin SCL	-
Pin A4	Pin SDA	-
Pin D2	Pin INT	-
Pin 3,3V	-	Pin VCC
Pin D7	-	Pin CE
Pin D8	-	Pin CSN
Pin D13	-	Pin SCK
Pin D11	-	Pin MOSI
Pin D12	-	Pin MISO
Pin GND	Pin GND	Pin GND



Gambar 2. Perancangan Rangkaian *Transmitter*

**2.1.2. Desain Sarung Tangan**

Perancangan desain sarung tangan ini di sesuaikan dengan desain perancangan rangkaian *transmitter* sebelumnya. Rangkaian pcb akan di letakkan pada bagian atas sarung tangan manusia dan tempat baterai pada bagian bawahnya, seperti pada Gambar 3 di bawah ini.



Gambar 3. Perancangan Desain Sarung Tangan

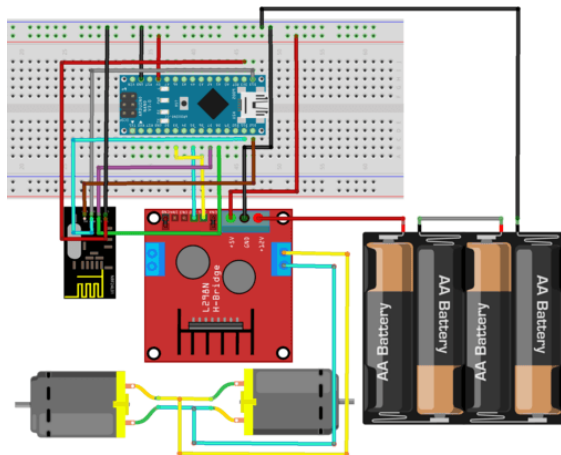
**2.1.3. Perangkat Keras Receiver**

Perancangan perangkat keras *receiver* ini menjelaskan tentang perancangan pin pada Driver Motor L298N, Motor Servo dan NRF24L01 dengan mikrokontroler Arduino Nano. Driver Motor L298N memerlukan 1 pin vcc 5V serta 1 pin ground, 2 pin digital yang terhubung dengan mikrokontroler Arduino Nano. Motor Servo memerlukan 1 pin vcc 5V serta 1 pin ground dan 1 pin digital yang terhubung dengan mikrokontroler Arduino Nano. Modul NRF24L01 memerlukan 1 pin vcc 3,3V serta 1 pin ground dan 5 pin digital yang terhubung dengan mikrokontroler Arduino Nano. Untuk lebih jelasnya dapat di lihat pada Tabel 2 dan Gambar 4 untuk keseluruhan perancangan rangkaian pada *receiver*.

Tabel 2. Perancangan Pin *Receiver*

Arduino Nano	Driver Motor L298N	Motor Servo	NRF24L01
Pin 5V	Pin Power 5V	Pin VCC	-
Pin D3	Pin IN1	-	-
Pin D5	Pin IN2	-	-
Pin 3,3V	-	-	Pin VCC
Pin D7	-	-	Pin CE

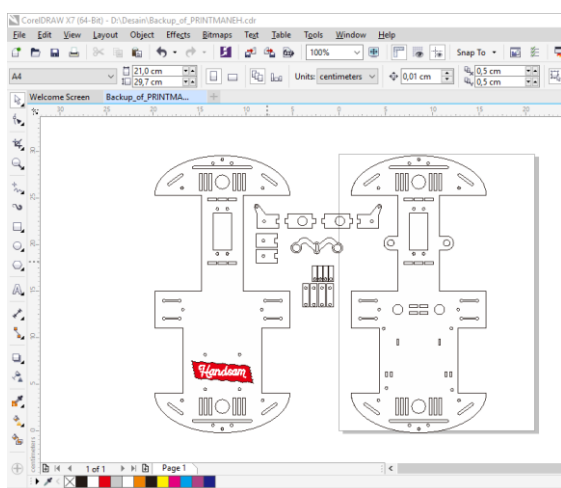
Pin D8	-	-	Pin CSN
Pin D13	-	-	Pin SCK
Pin D11	-	-	Pin MOSI
Pin D12	-	-	Pin MISO
Pin D10	-	Pin PWM	-
Pin GND	Pin GND	Pin GND	Pin GND



Gambar 4. Perancangan Rangkaian Receiver

### 2.1.4. Desain Rangka Robot Mobil

Perancangan desain rangka robot mobil ini di sesuaikan dengan desain perancangan rangkaian receiver di atas. Penulis menggunakan aplikasi CorelDraw X7 untuk mendesain rangka robot mobil, berikut adalah desain untuk kebutuhan mekanik rangka robot mobil yang di tunjukkan pada Gambar 5 di bawah ini.



Gambar 5. Perancangan Rangka Robot Mobil

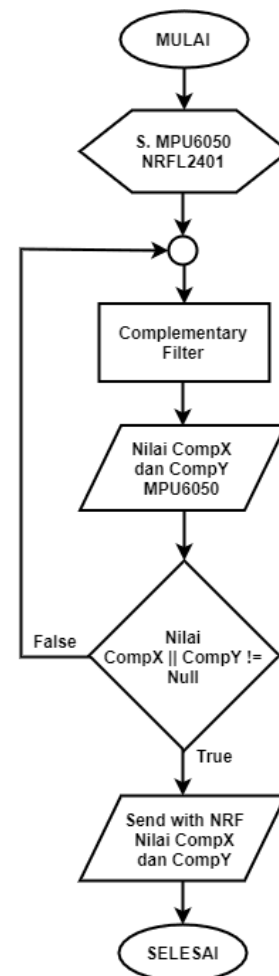
## 2.2. Perancangan Perangkat Lunak

Perancangan perangkat lunak disini, menjelaskan tentang bagaimana perancangan keseluruhan sistem dapat bekerja pada penelitian

ini. Perancangan perangkat lunak meliputi perancangan perangkat lunak transmitter dan perancangan perangkat lunak receiver.

### 2.2.1. Perangkat Lunak Transmitter

Perancangan perangkat lunak transmitter ini menjelaskan tentang alur sistem dapat bekerja pada transmitter. Pada Gambar 6 dapat di lihat diagram alir sistem. Proses di mulai dari pembacaan sensor MPU6050, data keluaran sensor ini berupa posisi pitch (sudut putaran sumbu y) dan roll (sudut putaran sumbu x). Proses berikutnya, data dari sensor MPU6050 di filter oleh arduino nano menggunakan algoritma complementary filter. Setelah data selesai di filter, maka hasil nya akan di kirim menggunakan NRF24L01.

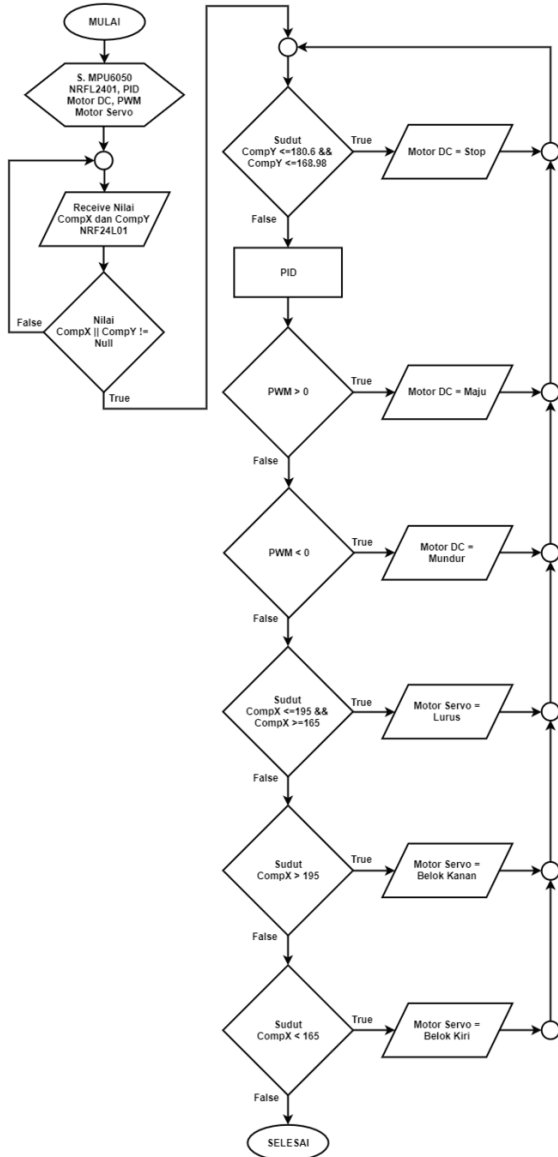


Gambar 6. Diagram Alur Sistem Transmitter

### 2.2.2. Perangkat Lunak Receiver

Perancangan perangkat lunak receiver ini menjelaskan tentang alur sistem dapat bekerja pada receiver. Pada Gambar 7 dapat di lihat diagram alir sistem. Proses di mulai dari

penerimaan data dengan bantuan modul NRF24L01. Data yang diterima berupa sudut *pitch* (sudut putaran sumbu y) akan di proses menggunakan algoritma PID sebelum menjadi masukan untuk Motor DC, sedangkan data dari sudut *roll* (sudut putaran sumbu x) akan langsung menjadi masukan untuk Motor Servo.



Gambar 7. Diagram Alur Sistem Receiver

### 2.3. Implementasi Perangkat Keras

Implementasi perangkat keras ini, di lakukan sebagaimana sesuai dengan perancangan mekanik perangkat keras yang telah di desain pada tahap sebelumnya. Implementasi perangkat keras meliputi implementasi peletakkan perangkat keras pada sarung tangan dan implementasi peletakkan perangkat keras pada rangka mobil.

#### 2.3.1. Implementasi Pada Sarung Tangan

Implementasi peletakkan perangkat keras pada sarung tangan ini di sesuaikan dengan desain sarung tangan. Terdapat sensor MPU6050 yang di letakkan horizontal dengan sumbu y menghadap ke arah jari, Arduino Nano di letakkan pada bagian tengah pcb, modul NRF24L01 yang berada pada bagian kiri, empat buah led indikator yang di letakkan pada bagian belakang serta tempat baterai yang di letakkan pada bagian bawah sarung tangan manusia, untuk lebih jelasnya dapat di lihat pada Gambar 8 di bawah ini.



Gambar 8. Implementasi Peletakkan Perangkat Keras Pada Sarung Tangan

#### 2.3.1. Implementasi Pada Rangka Mobil

Implementasi peletakkan perangkat keras pada rangka mobil ini di sesuaikan dengan desain rangka robot mobil, dimana rangka robot mobil ini menggunakan bahan akrilik ketebalan 3 mm dengan ukuran panjang 29,775 cm dan lebar 14,5 cm. Rangka robot mobil ini di desain khusus oleh penulis untuk dapat meletakkan rangkaian pcb, tempat baterai, motor servo, motor dc beserta empat ban pada rangka bagian atasnya dan driver motor L298N pada bagian bawahnya, seperti pada Gambar 9 berikut.



Gambar 8. Implementasi Peletakkan Perangkat Keras Pada Rangka Robot Mobil

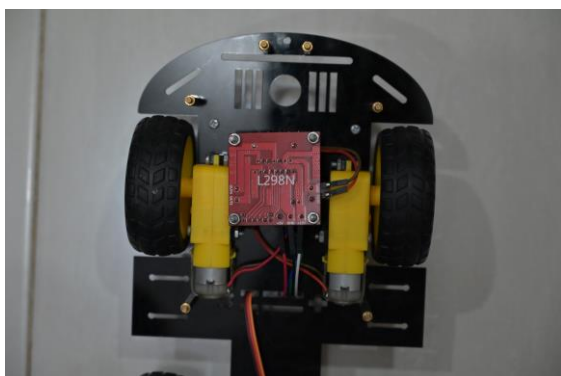
Pada Gambar 9 memperlihatkan bagaimana mekanik antara ban dengan motor servo pada

kaki depan robot mobil ini. Desain mekanik pada kaki depan mobil di rancang sedemikian rupa agar robot mobil ini dapat belok ke arah kanan dan kiri.



Gambar 9. Implementasi Peletakkan Motor Servo

Sedangkan untuk driver motor di letakkan pada rangka bagian bawah robot mobil ini, seperti pada Gambar 10 di bawah ini.



Gambar 10. Implementasi Peletakkan Driver Motor L298N

### 2.4. Implementasi Perangkat Lunak

Implementasi perangkat lunak ini, di lakukan sebagaimana sesuai dengan perancangan keseluruhan sistem perangkat lunak yang telah di desain dalam bentuk diagram alur.

Implementasi perangkat lunak meliputi implementasi sistem pada *transmitter* yang di dalamnya terdapat program algoritma complementary filter dengan *library* SPI.h, RF24.h dan Wire.h sedangkan untuk implementasi sistem pada *receiver* terdapat metode PID dengan *library* SPI.h, RF24.h dan Servo.h.

## 3. PENGUJIAN DAN ANALISA

Pada pengujian dan analisis ini, menjelaskan proses pada pengujian dan menganalisis untuk mendapatkan hasil apakah

sistem pada sistem yang di gunakan sudah berjalan sesuai dengan perancangan dan implementasi pada proses sebelumnya. Pengujian ini meliputi pengujian kalibrasi sensor MPU6050, pengujian akurasi sensor MPU6050, pengujian pengaruh Complementary Filter, pengujian mencari nilai Kp, Ki dan Kd, pengujian keseluruhan sistem dan pengujian *delay response time*.

### 3.1. Penulisan Kalibrasi Sensor MPU6050

Pengujian dari kalibrasi sensor MPU6050 ini, memiliki tujuan untuk menentukan nilai *offset* sensor. Nilai ini, berguna untuk mengatur nilai keluaran pada sensor berdasarkan tempat pengujian-nya.

Dari hasil pengujian kalibrasi sensor, nilai *offset* yang dihasilkan harus mendekati nilai *offset* keluaran pabrikan sensor MPU6050. Setiap pengujian memiliki nilai *offset* sensor-nya masing-masing, sesuai dengan tempat pengujian-nya. Untuk hasil pengujian, dapat di lihat pada Tabel 3 di bawah ini.

Tabel 3. Kalibrasi Sensor MPU6050

Nama Offset	Pembacaan Offset	Offset Sensor	Offset Pabrikan
accx	-6	-2587	0
accy	0	561	0
accz	16392	1354	16384
gyrox	0	94	0
gyroy	0	-87	0
gyroz	1	16	0

### 3.2. Pengujian Akurasi Sensor MPU6050

Pengujian akurasi sensor MPU6050 ini, memiliki tujuan untuk mendapatkan nilai keluaran sensor yang mendekati nilai kebutuhan sudut pada penelitian ini.

Pada Tabel 4 menunjukkan hasil pengujian keluaran sudut *pitch* dan *roll* di dapatkan analisis rata-rata sudut *error* sebesar 39%. Dari hasil rata-rata tersebut dapat di tarik kesimpulan sementara, penelitian ini memerlukan algoritma complementary filter untuk mengurangi sudut *error* tersebut.

Tabel 4. Akurasi Sensor MPU6050

Sudut Pengujian	Rata-rata Sudut Keluaran Sensor	Sudut Error
<b>Pitch 0°</b>	0.71°	0.71°
<b>Pitch 20°</b>	20.04°	0.04°
<b>Pitch -20°</b>	-20.04°	0.04°

Roll 0°	0.58°	0.58°
Roll 30°	30.48°	0.48°
Roll -30°	-30.54°	0.54°
Rata-rata Sudut Error Dalam Persen	<b>39%</b>	

### 3.3. Pengujian Pengaruh Complementary

Pengujian pengaruh complementary filter ini, memiliki tujuan untuk mengetahui seberapa besar pengaruh complementary filter pada sensor mpu6050.

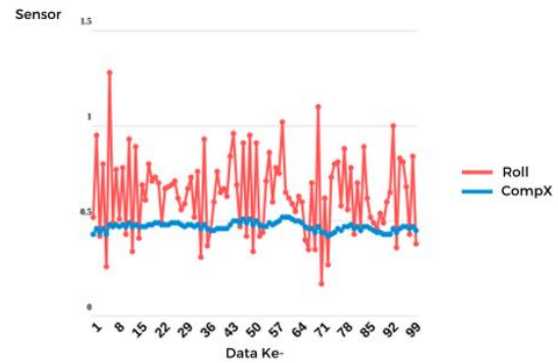
Dapat di lihat pada Tabel 5 merupakan data dari nilai *roll* dan *pitch*, tanpa dan dengan complementary filter. Dari 100 data uji diambil 10 data sampel dengan kegagalan 4 data nilai redam yang membuat akurasi pada pengujian ini sebesar 80%. Hasil standar deviasi CompX sebesar 0.02 dan CompY sebesar 0.09 yang membuktikan bahwa complementary filter lebih stabil daripada tanpa menggunakan filter dan untuk nilai redam *roll* sebesar 21.8% dan *pitch* sebesar 26.5%.

Tabel 5. Pengaruh Complementary Filter

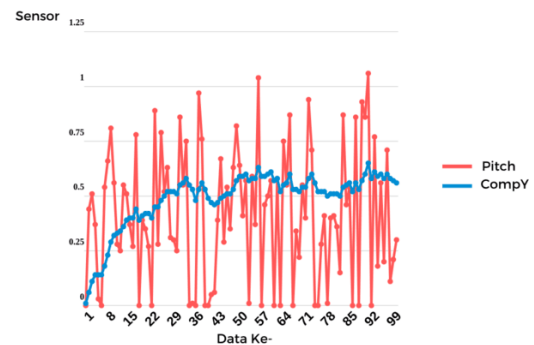
Sampel	Roll	Com pX	Nilai Redam	Pitch	Com pY	Nilai Redam
1	0.52	0.43	0.09	-	0.01	0.33
2	0.95	0.46	0.49	0.44	0.06	0.38
3	0.42	0.44	-0.02	0.51	0.11	0.4
4	0.80	0.46	0.34	0.37	0.14	0.23
5	0.26	0.43	-0.17	0.03	0.14	-0.11
6	1.28	0.48	0.8	-	0.14	-0.13
7	0.49	0.47	0.02	0.54	0.18	0.36
8	0.77	0.48	0.29	0.66	0.23	0.43
9	0.51	0.47	0.04	0.81	0.29	0.52
10	0.78	0.48	0.3	0.56	0.32	0.24
Standar Deviasi	0.29	0.02	<b>21.8</b> %	0.35	0.09	<b>26.5</b> %

Pada Gambar 11 dan Gambar 12 menunjukkan persebaran nilai yang di peroleh dari nilai *roll* dan *pitch* dengan jumlah 100 data sampel.

Dari kedua gambar tersebut dapat di simpulkan bahwa complementary filter dapat meredam *noise* dari nilai *roll* dan *pitch* dan membuat sensor mpu6050 lebih stabil daripada tanpa menggunakan menggunakan filter.



Gambar 11. Roll Sudut 0°

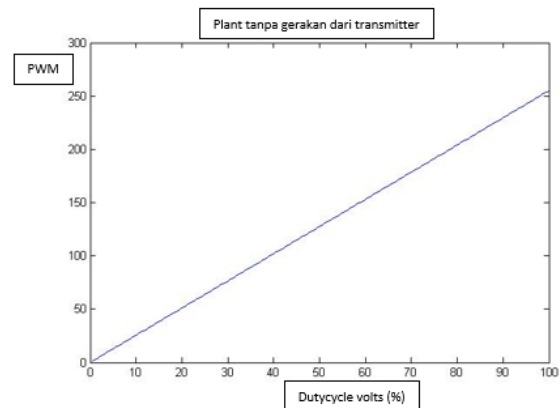


Gambar 12. Pitch Sudut 0°

### 3.4. Pengujian Mencari Nilai Kp, Ki dan Kd

Pengujian mencari nilai Kp, Ki dan Kd ini, memiliki tujuan untuk mendapatkan nilai karakteristik dari *static gain* pada keadaan mantap.

Plant yang berupa robot mobil untuk mengetahui respon *steadystate* dalam keadaan tanpa gerakan dari *transmitter*. Grafik karakteristik *static gain* keadaan tanpa gerakan ditunjukkan pada Gambar 13 di bawah ini.



Gambar 13. Karakteristik Static Gain Keadaan Tanpa Gerakan Transmitter

Kemudian dari percobaan *hard tuning* yang di lakukan beberapa kali, di dapatkan

nilai L dan T dengan grafik Kp, Ki dan Kd yang paling bagus.

Gunakan bias yang di dapatkan dari fungsi alih *plant* seperti yang ditunjukkan dalam Persamaan 1 di bawah ini.

$$G(s) = \frac{Ke^{-Ls}}{Ts + 1} = \frac{3.92e^{-0.33s}}{1.92s + 1} \quad (1)$$

L = 0.33 s  
T = 1.92 s

Dari hasil nilai L sebesar 0,33 dan T sebesar 1,92 tersebut maka, hitung nilai Kp, Ki, dan Kd menggunakan persamaan *Ziegler-Nichols* agar mendapat nilai akhir Kp pada Persamaan 2, Ki pada Persamaan 3 dan Kd persamaan 4 di bawah ini.

$$Kp = 1.2 \times \frac{T}{L} = 1.2 \times \frac{1.92}{0.33} = 6.98 \quad (2)$$

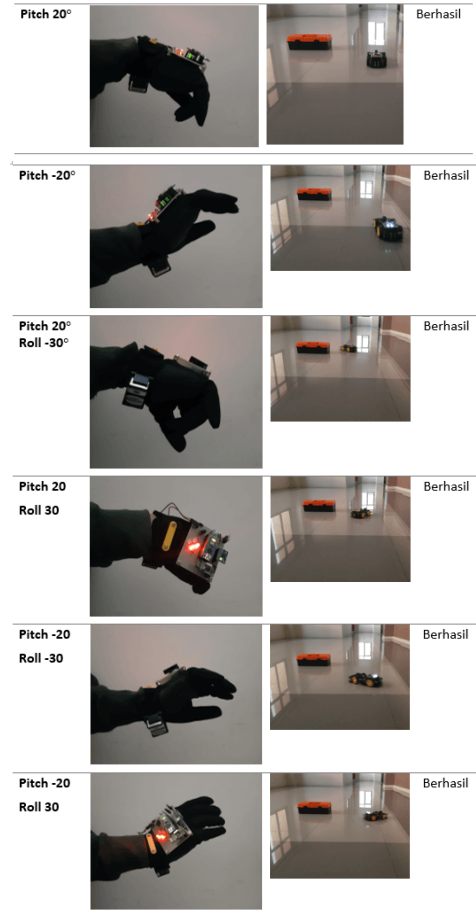
$$Ki = \frac{Kp}{Ti} = \frac{20}{2 \times L} = \frac{6.98}{0.66} = 4.61 \quad (3)$$

$$Kd = Kp \times Td = 6.98 \times (0.5 \times L) = 6.98 \times 0.165 = 1.15 \quad (4)$$

### 3.5. Pengujian Keseluruhan Sistem

Pengujian keseluruhan sistem ini, memiliki tujuan untuk mengetahui apakah sistem pada pengendalian robot mobil menggunakan gestur tangan manusia ini berjalan dengan baik.

Dapat di lihat pada Gambar 14 menunjukkan hasil dari pengujian keseluruhan sistem pengendali robot mobil dengan sembilan gestur atau gerakan tangan manusia untuk mengendalikan sebuah robot mobil. Nilai besar sudut hanyalah contoh saja, agar dapat mengerti bagaimana gestur gerakan tangan pada saat pengujian.



Gambar 14. Keseluruhan Sistem

Dari hasil pengujian keseluruhan sistem di atas, dapat disimpulkan bahwa sistem pengendali robot mobil menggunakan metode complementary filter pada sensor mpu6050 dapat berjalan dengan baik. Berikut merupakan presentase keberhasilan sistem yang di tunjukkan pada Persamaan 5 di bawah ini.

$$\frac{\text{Jumlah Keberhasilan}}{\text{Jumlah Data}} \times 100\% = \frac{9}{9} \times 100\% = 100\% \quad (5)$$

### 3.6. Pengujian Delay Response Time

Pengujian *delay response time* ini, memiliki tujuan untuk mengetahui berapa besar *delay* dan *response time* antara NRF24L01 pada *transmitter* dan NRF24L01 pada *receiver* yang di gunakan pada sistem pengendalian robot mobil menggunakan gestur tangan manusia.

#### 3.6.1. Pengujian Delay Response Time Awal

Dapat di lihat pada Gambar 15 menunjukkan proses pengujian *delay* dan *response time* yang di butuhkan dengan rata-rata sebesar 1.57 detik,

Sudut	Gestur Tangan	Hasil	Verifikasi
Pitch 0° Roll 0°			Berhasil
Roll 30°			Berhasil
Roll -30°			Berhasil



pada saat robot mobil di hidupkan dan kedua NRF24L01 mulai saling terhubung.



Gambar 15. Pengujian Delay Response Time Awal

Pada Tabel 6 menunjukkan data dari delay response time awal yang di lakukan sebanyak lima kali.

Tabel 6. Rata-rata Delay Response Time Awal

No	Waktu Awal	Waktu Akhir	Delay Response Time
1	01.19s	02.52s	1.33s
2	09.17s	10.87s	1.70s
3	10.56s	12.18s	1.62s
4	12.24s	13.83s	1.59s
5	14.23s	15.85s	1.62s
Rata-rata Delay Response Time			1.57s

### 3.6.1. Pengujian Delay Response Time 3m

Pada Gambar 16 menunjukkan proses pengujian delay dan response time yang di butuhkan dengan rata-rata sebesar 6.4 ms, pada saat mengendalikan robot mobil dengan jarak 3 meter.



Gambar 13. Pengujian Delay Response Time Jarak 3 Meter

Pada Tabel 7 menunjukkan data dari delay response time dengan jarak 3 meter yang di lakukan sebanyak lima kali.

Tabel 7. Rata-rata Dalam Jarak 3 Meter

No	Waktu Awal	Waktu Akhir	Delay Response Time
1	05.27ms	05.33ms	6ms
2	13.12ms	13.19ms	7ms
3	16.18ms	16.25ms	7ms

4	17.35ms	17.42ms	7ms
5	23.44ms	23.49ms	5ms
Rata-rata Delay Response Time			6.4ms

### 3.6.2. Pengujian Delay Response Time 7.5m

Pada Gambar 17 menunjukkan proses pengujian delay dan response time yang di butuhkan sebesar 7.2 ms pada saat mengendalikan robot mobil dengan jarak 7.5 meter.



Gambar 13. Pengujian Delay Response Time Jarak 7.5 Meter

Pada Tabel 8 menunjukkan data dari delay response time dengan jarak 7.5 meter yang di lakukan sebanyak lima kali.

Tabel 8. Rata-rata Dalam Jarak 7.5 Meter

No	Waktu Awal	Waktu Akhir	Delay Response Time
1	03.09ms	03.17ms	8ms
2	04.08ms	04.15ms	7ms
3	16.18ms	16.25ms	7ms
4	08.17ms	08.25ms	8ms
5	08.51ms	08.57ms	6ms
Rata-rata Delay Response Time			7.2ms

### 3.6.3. Pengujian Delay Response Time 10m

Pada akhir proses pengujian diperlihatkan pada Gambar 14 yang menunjukkan proses pengujian delay dan response time yang di butuhkan sebesar 7.4 ms pada saat mengendalikan robot mobil dengan jarak 10 meter.

Untuk lebih jelas nya dapat di lihat pada Tabel 9 yang menunjukkan data dari delay response time dengan jarak 10 meter yang di lakukan sebanyak lima kali.



Gambar 14. Pengujian Delay Response Time Jarak 10 Meter

Tabel 9. Rata-rata Dalam Jarak 10 Meter

No	Waktu Awal	Waktu Akhir	Delay Response Time
1	02.40ms	02.47ms	7ms
2	04.36ms	04.43ms	7ms
3	06.11ms	06.19ms	8ms
4	07.45ms	07.52ms	7ms
5	10.19ms	10.27ms	8ms
Rata-rata Delay Response Time			7.4ms

#### 4. KESIMPULAN

Berdasarkan hasil dari penyusunan, penerapan, pengujian serta analisis sistem yang telah penulis selesaikan dan dapat disimpulkan:

- a. Perancang alat pengendali robot mobil dapat menggunakan modul MPU6050 GY-521 sebagai *input* data *pitch* dan *roll*, kemudian di proses menggunakan mikrokontroler Arduino Nano yang terdapat metode complementary filter di dalamnya, modul NRF24L01 sebagai pengirim dan penerima data menggunakan gelombang radio, kemudian *output* motor DC dan motor Servo sebagai aktuator pada robot mobil.
- b. Dapat di simpulkan pengaruh yang di dapat dengan implementasi pengendali robot mobil menggunakan gestur tangan manusia dengan metode Complementary Filter lebih baik daripada tanpa menggunakan filter dapat di lihat pada hasil pengujian pengaruh complementary filter, di mana complementary filter dapat meredam *noise pitch* sebesar 29% dan *roll* sebesar 21% pada sensor mpu6050.
- c. Jarak antara perangkat keras pengendali

robot mobil dengan robot mobil mempengaruhi *delay response time*. Semakin jauh jarak antara perangkat keras *transmitter* dengan *receiver* semakin besar pula *delay response time* nya.

#### 5. DAFTAR PUSTAKA

S S, A. Z., Salahuddin, N. S. & Sari, S. P., 2017. Alat Pembaca Gerakan Lengan Menggunakan Accelerometer Dan Gyroscope Untuk Menggerakkan Robot Lengan.

Satriady, A., Alamsyah, W., Saad, A. H. & Hidayat, S., 2016. Pengaruh Luas Elektroda Terhadap Karakteristik Baterai LiFePo4.

Sayyid, A. R., WS, M. S. & P, Y. S., 2015. Kontrol Mobil Robot Menggunakan Hand Gesture Recognition Dengan Metode Adaptive Neuro-Fuzzy Interference System (ANFIS).

Siregar, R. F., 2017. Yogyakarta: Universitas Muhammadiyah Yogyakarta.

Sugiharto, A. & Windiyanti, S., 2017. Rancang Bangun Robot Pengintai Dengan Kendali Android.

Iqbal, M. & S., 2013. Perancangan Wheel Mobile Robot Sebagai Modul Pratikum Mikrokontroler.

Kahir, J. P., 2018. Implementasi Pengaturan Kekuatan Tendangan Prototipe Kaki Robot Pada Kaki Manusia Menggunakan Sensor Gyroscope Dan Accelerometer Dengan Metode Pid.

Musri, T., Herumurti, D. & Munif, A., 2017. Remote Control Komunikasi Robot Berbasis Pergerakan Tangan Pada Smartphone Menggunakan Metode Logika Fuzzy.

Umam, F., 2013. Pengembangan Sistem Kendali Pergerakan Autonomous Mobile Robot Untuk Mendapatkan Jalur Bebas Hambatan Menggunakan Fuzzy Logic Controller.

Z. & Sutisna, U., 2014. Perancangan Sistem Kontrol Wireless Pada Mobile Robot Manipulator Berbasis Mikrokontroler ATMega8.