

KOMPARASI HASIL PERANCANGAN SISTEM KEAMANAN SEPEDA MOTOR BERBASIS SMARTPHONE DAN KOMPUTER

COMPARATIVE RESULTS OF MOTORCYCLE SAFETY SYSTEM DESIGN BASED ON SMARTPHONE AND COMPUTER

Kalimas Henggar Sadewo¹, Raditiana Patmasari², Arif Indra Irawan³

^{1,2,3} Universitas Telkom, Bandung

¹kalimashenggar@student.telkomuniversity.ac.id ² raditiana@telkomuniversity.ac.id

³ arifirawan@telkomuniversity.ac.id

Abstrak

Keamanan alat transportasi menjadi penting untuk dimiliki mengingat banyaknya kejahatan yang dilakukan dari tahun ke tahun. Statistik kriminal tahun 2020 yang dikeluarkan oleh Badan Pusat Statistik Indonesia, menyatakan bahwa pada tahun 2020 telah terjadi banyak kasus jumlah kejadian kejahatan terhadap hak/milik tanpa penggunaan kekerasan. Kejahatan terhadap hak/milik tanpa penggunaan kekerasan merupakan jenis kejahatan yang paling banyak terjadi dalam segi jumlah setiap tahunnya. Kejahatan ini didominasi oleh kejahatan pencurian alat rumah dan pencurian kendaraan bermotor.

Pada penelitian tugas akhir ini dibuat sebuah aplikasi yang bertujuan untuk membantu pengguna kendaraan bermotor dalam meningkatkan keamanan kendaraan. Alat ini menggunakan *Voice Command* untuk mengatur semua perintah dan menjalankan fitur-fitur yang terdapat di aplikasi seperti menyalakan dan mematikan motor, memberikan notifikasi penggunaan sepeda motor, dan memastikan keberadaan sepeda motor. Pada penelitian ini juga dilakukan analisis perbandingan simulasi *Speech to Text* pada aplikasi *Smartphone* dan komputer secara *Real Time*. Hasil pengujian menunjukkan bahwa hasil perancangan pada aplikasi *Smartphone* dapat mencapai akurasi 92% dalam mengenali sinyal suara berupa kalimat perintah sedangkan pada komputer dapat mencapai akurasi 45% dalam mengenali sinyal suara berupa kalimat perintah dan pada kedua sistem, perintah yang diberikan dapat berjalan dan berfungsi pada sepeda motor seperti nyalakan dan matikan mesin, nyalakan dan matikan kunci, nyalakan dan matikan alarm, nyalakan dan matikan kunci rahasia, dan kirim lokasi.

Kata Kunci : Aplikasi Keamanan, KNN, MFCC, Sepeda Motor

Abstract

Transportation security becomes important to have, considering the number of crimes committed from year to year. Criminal Statistic from Indonesian Central Statistics Agency said, in 2020 there have been many cases of crimes against right ownership without violence. Crimes against right ownership without violence are the most common types of crime in terms of number each year. This crime is dominated by home appliance theft and motor vehicle theft.

In this final project, an application is made to help motorcycle user to secure their motorcycle. This application uses voice command to manage all commands and features like turning the engine on and off, giving notifications from the motorcycle, and ensuring the presence of the motorcycle. In this final project, a comparison analysis of Speech To Text simulations on Smartphone and computer applications was also carried out in real time.

The result showed that, the results of the design on the Smartphone application, it can achieve 92% accuracy in recognizing voice signals in the form of command sentences while on the computer, it can achieve 45% accuracy in recognizing voice signals in the form of command and on both system, the commands given can run and work on motorcycle such as turn on and turn off the machine, turn on and turn off the key, turn on and turn off the alarm, turn on and turn off the secret key, and get location.

Keywords : MFCC, KNN, Motorcycle, Security Application

1. Pendahuluan

1.1 Latar Belakang Masalah

Kendaraan bermotor merupakan salah satu jenis transportasi untuk memudahkan masyarakat berpindah tempat atau berpergian. Akan tetapi, sering sekali terjadi kesalahan seperti kehilangan kunci motor, kunci motor rusak, ataupun keteledoran pemilik dalam menyimpan kendaraan saat parkir diluar, maka yang terjadi tindak kriminalitas seperti pencurian kendaraan bermotor bisa terjadi. Dikutip dari [1] Statistik Kriminal tahun 2020 yang dikeluarkan oleh Badan Pusat Statistik Indonesia, menyatakan bahwa pada tahun 2020 telah terjadi banyak kasus jumlah kejadian kejahatan terhadap hak/milik tanpa penggunaan kekerasan. Kejahatan terhadap hak/milik tanpa penggunaan kekerasan merupakan jenis kejahatan yang paling banyak dalam segi jumlah setiap tahunnya. Pada tahun 2020 terjadi kejahatan sebanyak 80.450 kasus. Kejahatan ini didominasi oleh kejahatan pencurian alat rumah dan pencurian kendaraan bermotor.

Beberapa penelitian telah dilakukan untuk membuat sistem keamanan pada sepeda motor, seperti penelitian yang dilakukan oleh [2] Muhammad Irsyad yang melakukan pengimplementasian sistem keamanan sepeda motor menggunakan *RFID*, *GSM* Modul, *GPS* modul berbasis mikrokontroler yang dapat mematikan atau menyalakan sepeda motor, dan melacak lokasi sepeda motor. Namun, pada penelitian tersebut, tidak terdapat fitur *Voice Command* dan outputan tidak berupa aplikasi, dikhawatirkan penggunaan *RFID* masih membahayakan dikarenakan ukurannya yang kecil, sehingga sangat mudah untuk rusak ataupun hilang. Adapun penelitian sejenis yang pernah dilakukan berupa sistem keamanan kendaraan berbasis sensor nirkabel [3][4]. Dari sisi *Voice Command*, penelitian sebelumnya yang dilakukan oleh Azri Fathul, Azri Fathul melakukan pengujian *Speech Recognition* pada alat *Edu Braille* menggunakan Metode *Mel Frequency Cepstral Coefficient* dan *K-Nearest Neighbor* dan mendapat akurasi tertinggi sekitar 87,3% [5]. Namun pada pengujian tersebut, data pelafalan hanya berupa huruf alphabet.

Maka dari itu, dirancang suatu alat yaitu “Sistem Keamanan Sepeda Motor” yang bertujuan untuk membantu pengguna kendaraan bermotor dalam meningkatkan keamanan kendaraan. Proses kerja alat ini adalah ketika pemilik memberi perintah berupa suara, suara akan diterima oleh sepeda motor, dan sepeda motor akan menjalankan perintah yang diinputkan. Pada alat ini, terdapat fitur untuk menyalakan dan mematikan kunci, menyalakan dan mematikan mesin, fitur untuk membunyikan dan mematikan alarm, dan fitur menyalakan dan mematikan kunci rahasia, selain itu terdapat fitur kirim lokasi yang berfungsi untuk mengetahui posisi dari kendaraan bermotor. Pada penelitian ini juga dilakukan analisis perbandingan simulasi *Speech to Text* pada aplikasi *Smartphone* dan komputer. Untuk komputer, aplikasi yang digunakan adalah matlab. Pengujian ini dilakukan secara *Realtime* menggunakan suara berupa kalimat perintah seperti “nyalakan atau matikan kunci, nyalakan atau matikan mesin, nyalakan atau matikan alarm, nyalakan atau matikan kunci rahasia, dan kirim lokasi”.

Dengan dibuatnya aplikasi tersebut, diharapkan dapat membantu pengguna kendaraan bermotor untuk menjaga keamanan dari sepeda motor dan dapat meminimalisir tindak kriminal yang terjadi..

2. Dasar Teori

2.1 *Speech Recognition*

Speech Recognition atau yang biasa dikenal dengan *Automatic Speech Recognition (ASR)* merupakan proses identifikasi yang dilakukan komputer untuk mengenali kata yang diucapkan oleh seseorang tanpa memedulikan identitas orang terkait dengan melakukan konversi sebuah sinyal akustik, yang ditangkap oleh *Audio Device* (perangkat input suara) [6].

2.2 *Cropping Signal*

Cropping Signal atau pemotongan sinyal adalah suatu proses yang dilakukan dengan cara memotong sinyal yang bertujuan untuk menghilangkan sinyal yang tidak dibutuhkan dalam proses ekstraksi ciri [7].

2.3 *Mel Frequency Cepstral Coefficients*

Mel Frequency Cepstral Coefficients (MFCC) merupakan ekstraksi ciri berupa koefisien yang dapat mempresentasikan audio. Ini dibutuhkan ketika ingin melakukan pengenalan sinyal suara agar dapat diidentifikasi dan dianalisis lebih lanjut. Ekstraksi ciri dalam proses ini ditandai dengan cara mengubah suatu data suara menjadi data citra berupa spektrum gelombang[7].

2.3.1 *Frame Blocking*

Frame Blocking adalah proses yang bertujuan untuk membagi sampel sinyal nada pada beberapa frame tertentu atau slot dengan panjang tertentu, yang nantinya dapat memudahkan dalam perhitungan dan analisa sinyal. Setiap *Frame* terdiri dari *N* sampel suara, dengan jarak setiap *Frame* dipisahkan oleh *M*-sampel. Menyusun sinyal ke dalam bingkai yang lebih pendek. Panjang dari suatu *Frame* akan membagi setiap sample menjadi beberapa *Frame* berdasarkan waktunya [8].

2.3.2 *Windowing*

Windowing dilakukan untuk membobotkan terhadap setiap *Frame* yang telah dibentuk. Fungsi *Window* yang paling sering digunakan dalam *Voice Recognition* adalah *Hamming Window*. *Hamming Window* mempunyai *Side Lobe* yang paling kecil dan *Main Lobe* yang paling besar sehingga hasil *Windowing* akan lebih halus dalam menghilangkan efek diskontinuitas[8].

$$w_{Ham}(n) = \begin{cases} 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) & 0 < n < N - 1 \\ 0 & \text{Lainnya} \end{cases} \quad (2.1)$$

Dimana:

w_{Ham} : *Hamming Windowing*

n : Waktu diskrit

N : Jumlah data dari sinyal

2.3.3 Fast Fourier Transform

Proses FFT dimulai ketika sinyal yang berdomain waktu akan diubah menjadi sinyal berdomain frekuensi. FFT akan mengubah setiap *Frame* sejumlah N sampel dari domain waktu ke domain frekuensi. Dalam mengolah suatu sinyal suara FFT berguna dalam mengubah konvolusi getaran celah suara dan respon gelombang saluran suara dalam domain waktu[8].

2.3.4 Mel Frequency Wrapping

Persepsi telinga manusia tidak selalu *Linear*, karena itu untuk setiap frekuensi real diukur dalam Hz, nada yang subyektif akan diukur pada skala yang disebut dengan *Mel Scale* [9]. *Mel Scale* bertujuan untuk menyesuaikan resolusi setiap frekuensi terhadap properti pendengaran manusia. *Mel Scale* akan dikelompokkan menjadi beberapa *Critical Bank* menggunakan suatu *Filter Bank*. Jangkauan frekuensi dalam suatu spektrum sangatlah luas dan tidak mengikuti skala secara *Linear*, sehingga setelah spektrum dikomputasi akan dipetakan dalam *Mel Scale* menggunakan *Filter* segitiga.[10]

$$Mel(f) = 2595 \times \log_{10}(1 + f/100) \quad (2.2)$$

Dimana:

$Mel(f)$: *Mel Frequency Wrapping*

f : *Frequency*

2.3.5 Cepstrum

Pada tahap *Cepstrum* sinyal suara yang sudah dipetakan dalam *Mel Scale* akan diubah kembali kedalam domain waktu. *Mel Spectrum Coefficient* merupakan angka *Real* sehingga dapat diubah kedalam domain waktu menggunakan *Discrete Cosine Transform* (DCT) [10]. Konsep dasar DCT adalah dengan mendekorrelasikan dari suatu *Mel Spectrum* kemudian akan menghasilkan suatu representasi yang baik dari suatu properti *Spectral Local* [10].

$$C_n = \sum_{k=1}^K (\log \tilde{S}_k) \cos[n(k - \frac{1}{2}) \frac{\pi}{K}] \quad (2.3)$$

Dimana:

\tilde{S}_k : Keluaran dari proses *Filter Bank* pada indeks k

k : Jumlah koefisien yang diharapkan

n : Indeks *Cepstral Coefficients*

2.4 K-Nearest Neighbor

Algoritma *K-Nearest Neighbor* (KNN) adalah sebuah metode untuk melakukan klasifikasi terhadap objek yang berdasarkan dari data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Prinsip kerja dari KNN adalah mencari jarak terdekat antara data uji dengan *Database* [7].

$$d_i = \sqrt{\sum_{i=1}^p (x_{2i} - x_{1i})^2} \quad (2.4)$$

Dimana:

X_1 : Sampel data uji

X_2 : Data uji

d_i : Jarak *Euclidean*

p : Dimensi data

Adapun rumus yang digunakan dalam mencari jarak *Cityblock* terdekat adalah sebagai berikut [12].

$$d_{ij} = \sum_{k=1}^n |i_k - j_k| \quad (2.5)$$

Dimana:

i : Data uji

j : Data latih

k : Nilai atribut

n : Dimensi atribut

Adapun rumus yang digunakan dalam mencari jarak *Cosine* terdekat adalah sebagai berikut [13].

$$\cos a = \frac{A \cdot B}{|A||B|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}} \quad (2.6)$$

Dimana:

A_i : Data A

B_i : Data B

$A \cdot B$: *Dot Product* antara data A dan data B

$|A|$: Panjang vektor A

$|B|$: Panjang vector B

$|A||B|$: *Cross Product* antara data A dan data B

3. Perancangan Sistem

3.1 Desain Sistem

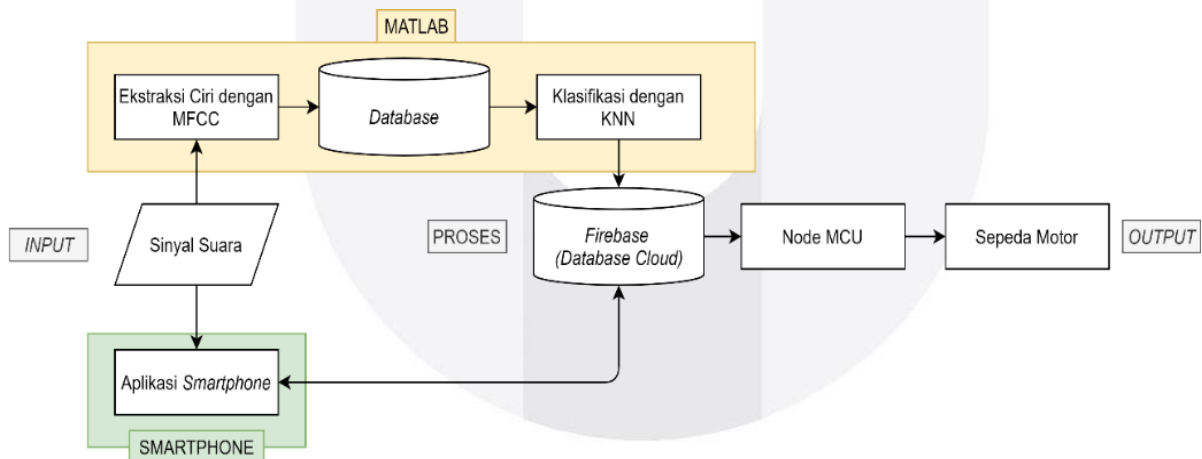
Pada perancangan dan implementasi sistem ini dibagi menjadi 4 tahap utama, yaitu pengumpulan data *Audio*, *Cropping Signal* yang bertujuan agar sinyal yang masuk ke sistem sesuai dengan spesifikasi yang diinginkan, kemudian data *Audio* tersebut di ekstraksi menggunakan MFCC. Tahap terakhir yaitu klasifikasi menggunakan *K-Nearest Neighbors*. Desain sistem pada tugas akhir ini dapat dilihat pada gambar 3.1:



Gambar 3.1 Desain Sistem

3.2 Perancangan Sistem Utama

Pada tugas akhir ini, dirancang suatu sistem agar sinyal suara manusia dapat diubah menjadi perintah *Text* untuk mengaktifkan fitur pada aplikasi sistem kendali sepeda motor. Terdapat dua sistem di dalam tugas akhir ini, sistem pada matlab dan sistem pada *Smartphone*. Untuk matlab. Inputan suara akan di ekstraksi menggunakan MFCC lalu suara akan disimpan pada *Database* dan diklasifikasi menggunakan KNN. Setelah itu, sinyal suara akan melakukan proses *Speech to Text* yang nantinya akan di terima oleh *Node MCU*. *Node MCU* dan *Firestore* akan mengkonfirmasi perintah yang diberikan matlab, jika berhasil, maka sepeda motor akan berjalan sesuai dengan perintah yang diberikan. Untuk aplikasi pada *Smartphone*, perintah suara akan diterima oleh aplikasi *Smartphone*, lalu akan mengirim ke *Firestore* melalui *Node MCU*. Jika berhasil, sepeda motor akan berjalan sesuai dengan perintah yang diberikan. Secara keseluruhan blok diagram yang akan dirancang dapat dilihat pada gambar 3.2

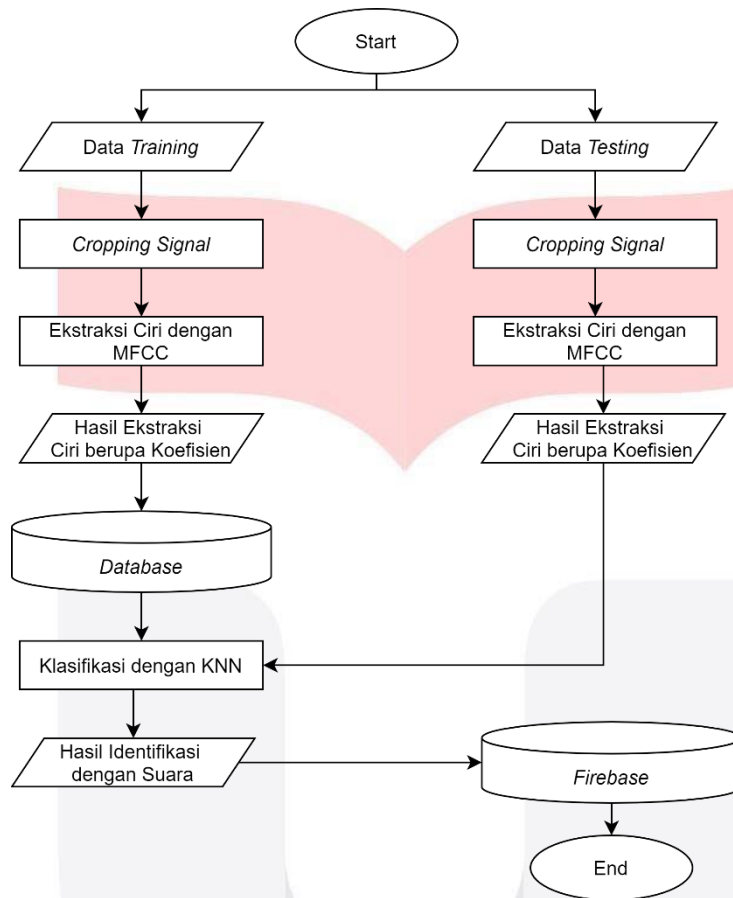


Gambar 3.2 Blok Diagram Perancangan Umum Sistem

Pada pengimplementasian perancangan ini, diharapkan ketika pengendara ingin menyalakan sepeda motor, pengendara dapat mengirimkan perintah berupa suara untuk menyalakan sepeda motor. Data perintah menyalakan motor akan diterima oleh aplikasi yang telah terhubung dengan *Firestore* untuk mengubah perintah suara menjadi perintah *Text*, lalu diteruskan ke motor oleh *Node MCU*. Apabila data yang diterima sesuai, maka motor akan menyala.

3.3 Diagram Alir Simulasi

Diagram alir simulasi bertujuan agar dalam pembuatan suatu simulasi perancangan lebih sistematis dan struktur prosesnya, sehingga simulasi dapat berjalan sesuai dengan yang diharapkan. Diagram alir pada sistem ini dapat dilihat pada gambar 3.3 :

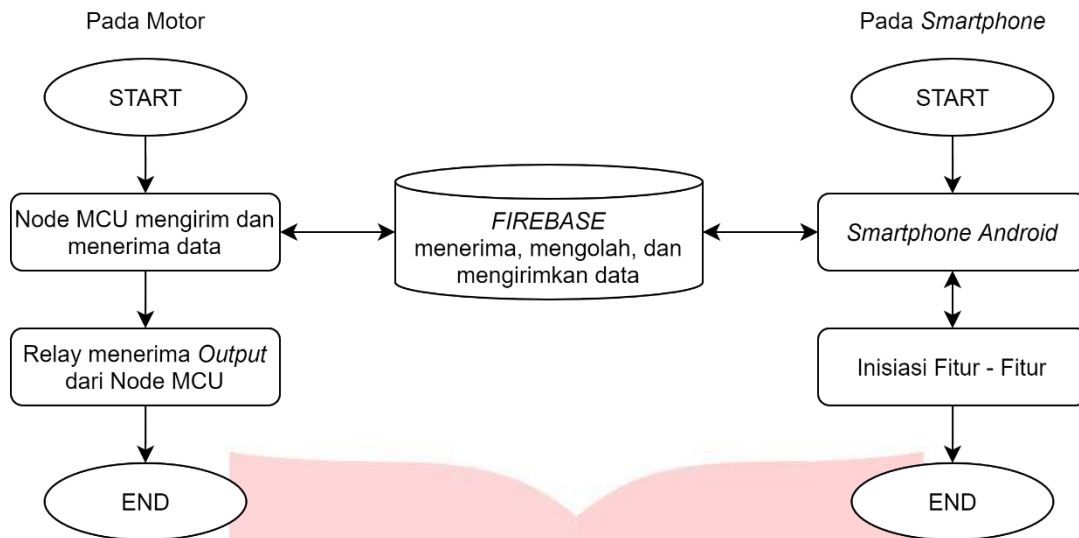


Gambar 3.3 Diagram Alir Simulasi

Dari blok diagram diatas secara garis besar dapat dijelaskan bahwa terdapat dua tahap, yaitu *Training* (data latih) dan *Testing* (data uji). Pada kedua tahap tersebut akan melalui tahap *Cropping Signal*, Ekstraksi ciri menggunakan *Mel Frequency Cepstral Coefficient*. Namun yang membedakan yaitu pada tahap data latih setelah didapat hasil dari ekstraksi ciri data tersebut akan disimpan kedalam *Database*, sedangkan pada tahap data uji dilakukan proses pengenalan suara untuk menentukan model referensi yang mirip dengan sinyal suara input menggunakan metode *K- Nearest Neighbors*. Setelah mendapatkan hasil identifikasi, maka data *Audio* diterima oleh *Firestore*, dari *Firestore* akan disambungkan ke motor.

3.4 Diagram Pembuatan Alat

Diagram alir simulasi pembuatan alat bertujuan agar dalam pembuatan suatu simulasi perancangan hasilnya akan lebih sistematis dan struktur prosesnya. Diagram alir pada sistem ini dapat dilihat pada gambar 3.4 :



Gambar 3.4 Diagram Alir Pembuatan Alat

Berdasarkan gambar 3.4 bahwa sistem dimulai dari membuka aplikasi di *Smartphone Android*. Lalu memasukkan suara menggunakan tombol pengenalan suara di aplikasi. Setelah direkam, ada 9 perintah yang bisa kita gunakan. Seperti nyalakan dan matikan kunci, nyalakan dan matikan mesin, nyalakan dan matikan alarm, nyalakan dan matikan kunci rahasia, dan kirim lokasi. Jika perintah yang diberikan sama dengan kodingan, maka perintah tersebut akan dikirim ke *Firebase*. *Firebase* berfungsi sebagai *Cloud Database* yaitu untuk mengirim data, mengolah data, menyimpan data, dan menerima data. Data yang masuk dari *Smartphone* akan diteruskan ke *Node MCU* oleh *Firebase*. *Node MCU* akan menerima data perintah dari *Smartphone* melalui *Firebase*. Jika perintahnya sama dengan kodingan, maka *Node MCU* akan menyalakan *Relay* dan *Relay* akan menghidupkan mesin.

3.5 Software (Perangkat Lunak)

Perangkat lunak yang digunakan pada tugas akhir ini yaitu sebagai berikut :

Tabel 3.1 Kebutuhan *Software*

Jenis	Versi	Kegunaan
Software Arduino IDE	1.6.12	Aplikasi yang digunakan untuk mengoperasikan Arduino UNO/mengoperasikan sistem <i>Microcontroller</i>
Android Studio		Aplikasi untuk membuat sebuah program aplikasi yang di pasang di <i>Smartphone</i> .
Matlab 2019		Aplikasi yang digunakan untuk menganalisa dan menguji data <i>Audio</i>

3.6 Hardware (Perangkat Keras)

Perangkat keras yang digunakan pada tugas akhir ini yaitu sebagai berikut :

Tabel 3.2 Kebutuhan *Hardware*

Jenis	Jumlah	Fungsi
Node MCU ESP8266	1	Berfungsi sebagai <i>Controller</i> utama pada sistem yang dibangun
Aki 12 volt	1	Sebagai sumber energi listrik
Relay 12v	4	Berfungsi sebagai penghubung <i>Controller</i> ke bagian komponen-komponen yang akan digunakan pada sistem yang akan dibangun
<i>DC Buck Converter 12V</i>	1	Berfungsi sebagai <i>Converter</i> daya dari <i>Accu</i> sepeda motor 12v yang di konversi ke 5v untuk menghidupkan komponen alat

3.7 Performansi Sistem

Sistem yang sudah dirancang akan diuji tingkat akurasi dalam menghasilkan *Output* yang sesuai dengan yang diinginkan. Perhitungan akurasi sistem yaitu adalah sebagai berikut :

$$\text{Akurasi} = \frac{\text{jumlah data benar}}{\text{jumlah data salah}} \times 100\% \quad (2.7)$$

4. Hasil Pengujian dan Analisis

4.1 Skenario Pengujian

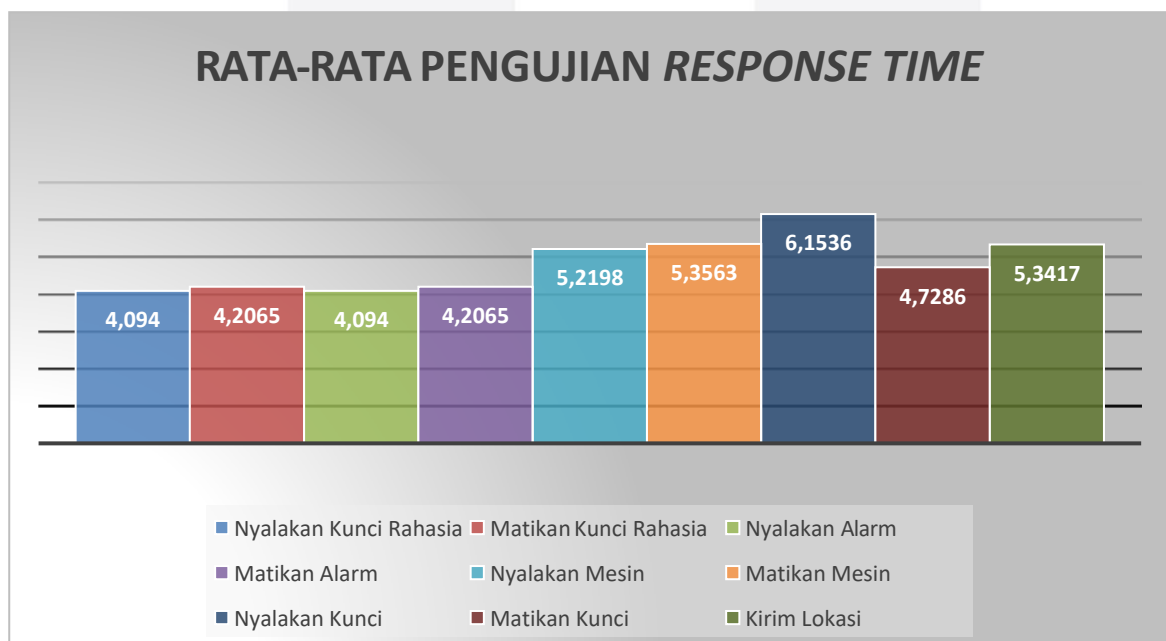
Pada skenario pengujian Tugas Akhir ini bagi menjadi 3 bagian yaitu skenario pengujian pada aplikasi *Smartphone*, skenario pengujian pada komputer dengan menggunakan aplikasi matlab, dan skenario pengujian antara aplikasi *Smartphone* dan komputer. Pengujian pada sistem ini yaitu menggunakan data *Audio* sebanyak 720 data yang dibagi menjadi 9 *Subclass*.

1. Skenario pengujian pada aplikasi *Smartphone*
 - a. Pengujian *Response Time* yang didapatkan ketika menjalankan suatu perintah
 - b. Pengujian akurasi pelafalan *Voice* dari aplikasi ke *Firebase*
2. Skenario pengujian pada komputer
 - a. Menganalisa pengaruh jumlah data terhadap tingkat akurasi yang didapatkan menggunakan ciri statistik.
 - b. Menganalisa pengaruh jumlah *Neighborhood* terhadap akurasi yang didapatkan
 - c. Menguji akurasi pelafalan *Voice* dari aplikasi matlab ke *Firebase*
3. Skenario pengujian pada aplikasi *Smartphone* dan komputer
 - a. Membandingkan akurasi sistem pada aplikasi *Smartphone* dan komputer

4.2 Hasil Pengujian Sistem

4.2.1 Pengujian *Response Time* pada Motor

Pengujian ini dilakukan ketika menekan tombol pada aplikasi *Smartphone* hingga sampai ke *Firebase*. Kemudian akan dilihat rata-rata *Response Time* sistem yang terjadi sebanyak 20 percobaan pada masing-masing perintah yang dibagi menjadi 9 *Subclass*. Adapun hasil pengujian *Response Time* terdapat pada gambar 4.1 :



Gambar 4.1 Pengujian *Response Time* pada Sistem Kendali Motor

Response Time tertinggi yaitu ketika sistem kendali motor diberikan inputan “nyalakan kunci” dengan *Response Time* sebesar 6.1536 detik dan *Response Time* terendah yaitu saat sistem kendali diberikan inputan “nyalakan kunci rahasia” dan “nyalakan alarm” dengan *Response Time* sebesar 4.094 detik. Sehingga rata-rata *Response Time* pada sistem saat diberikan 9 inputan perintah yaitu 4.367 detik. *Response Time* itu sendiri dipengaruhi oleh kualitas sinyal dari operator atau wifi, dalam pengujian ini menggunakan operator telkomsel.

4.2.2 Pengujian Akurasi Pelafalan Voice dari Aplikasi Smartphone ke Firebase

Pengujian ini bertujuan untuk melihat seberapa besar sistem dapat mengenali inputan berupa suara. Pengujian ini dilakukan dengan cara menginputkan perintah suara ke aplikasi *Smartphone* dan dilakukan sebanyak 20 kali percobaan yang dibagi menjadi 9 *Subclass*. Hasil akurasi dapat dilihat pada tabel 4.1 berikut :

Tabel 4.1 Akurasi Pelafalan Voice pada Motor

	Nyalakan Kunci Rahasia	Matikan Kunci Rahasia	Nyalakan Alarm	Matikan Alarm	Nyalakan Mesin	Matikan Mesin	Nyalakan Kunci	Matikan Kunci	Kirim Lokasi
Jumlah Benar	18	18	18	19	20	18	19	18	18
Jumlah Salah	2	2	2	1	0	2	1	2	2

Berdasarkan tabel 4.1 sistem dapat mengenali 9 inputan perintah dengan baik, Dan yang terbaik yaitu ketika diberikan inputan “nyalakan mesin”, dimana jumlah benar yang diperoleh yaitu 20. Rata rata Akurasi yang didapat pada pengujian ini yaitu sebesar 92%

4.2.3 Pengujian Jumlah Data Latih Terhadap Akurasi

Pengujian ini dilakukan perubahan pada jumlah data latih, untuk melihat apakah ada pengaruh perubahan pada akurasi sistem ketika jumlah data latih dirubah. Perubahan jumlah data latih dibagi menjadi 3 yaitu menggunakan 35 data latih 10 data uji, 70 data latih 10 data uji, dan 90 data latih 10 data uji dengan menggunakan ciri statistik (*Mean*, *Skew*, dan *Kurt*). Hasil nya tertera pada tabel 4.2 berikut :

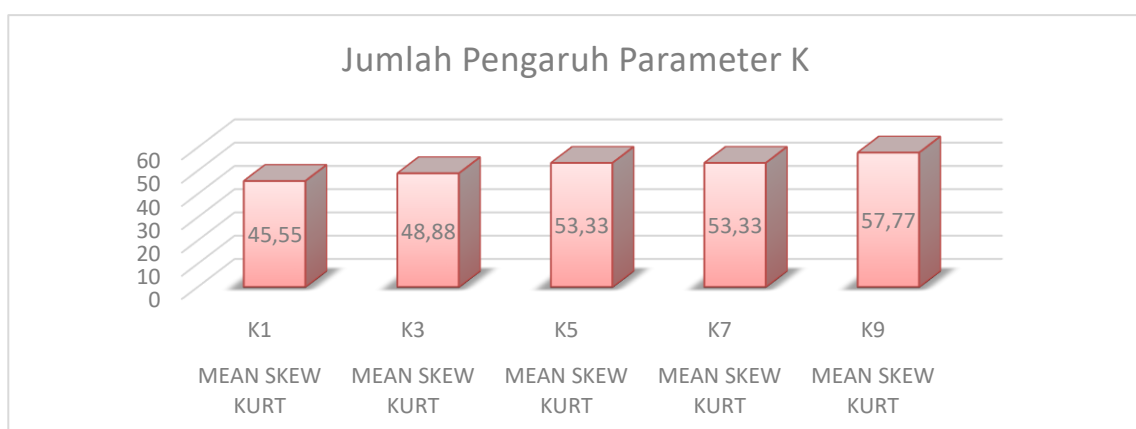
Tabel 4.2 Perubahan Jumlah Data Latih Terhadap Akurasi

Data Audio	Mean, Skew, Kurt (%)				
	K=1	K=3	K=5	K=7	K=9
35 data latih, 10 data uji	44.44	43.33	45.55	44.44	42.22
70 data latih, 10 data uji	45.55	48.88	53.33	53.33	57.77
90 data latih 10 data uji	44.44	51.11	55.55	56.66	51.11

Berdasarkan tabel 4.2 akurasi terendah terdapat pada saat menggunakan 35 data latih 10 data uji yaitu sebesar 42.22 % menggunakan K=9. Sedangkan akurasi tertinggi terdapat pada saat menggunakan 70 data latih 10 data uji yaitu sebesar 57.77 % menggunakan K=9. Semakin banyak jumlah data latih maka akan semakin baik akurasinya, namun faktor ketetangaan juga berpengaruh dalam menentukan sebuah akurasi. Harus seimbang dalam menentukan pemakaian jumlah data latih dan jumlah tetangga yang dibutuhkan, jika data latih terlalu banyak dan tidak menemukan tetangga yang sesuai, maka akan membuat ciri menjadi tidak jelas atau tidak dapat dikenali yang mengakibatkan akurasi akan menurun

4.2.4 Pengujian Jumlah Neighborhood Terhadap Akurasi

Pengujian ini dilakukan dengan menggunakan ciri statistik *Mean*, *Skew*, dan *Kurt* pada 70 data latih 10 data uji karena pada pengujian sebelumnya, ciri ini mendapatkan akurasi tertinggi diantara ciri statistik yang lainnya yaitu sebesar 57.77%. Pengujian ini dilakukan dengan mengganti nilai parameter K yang digunakan pada KNN. Parameter K yang diuji yaitu K=1, K=3, K=5, K=7, dan K=9, seperti yang tercantum pada gambar 4.2 berikut ini



Gambar 4.2 Pengaruh Jumlah Neighborhood Terhadap Akurasi

Berdasarkan gambar 4.2 akurasi tertinggi didapat pada saat menggunakan parameter $K = 9$ yaitu sebesar 57.77 %. Sedangkan untuk akurasi terendah didapat pada saat parameter $K=1$ yaitu sebesar 45.55 %. Dalam data ini, semakin banyak jumlah parameter K maka akan semakin baik akurasinya. Hal ini terjadi dikarenakan semakin banyak parameter K yang digunakan maka akan semakin banyak tetangga terdekat yang akan dibandingkan, jika semakin banyak tetangga terdekat yang dibandingkan, maka akan membuat sistem semakin terbantu dalam mengenali suatu sinyal suara

4.2.5 Pengujian Akurasi Pelafalan Voice dari Aplikasi Matlab ke Firebase

Pada skenario ini dilakukan pengujian akurasi pelafalan Voice dari aplikasi matlab hingga ke Firebase. Pengujian ini dilakukan untuk melihat apakah akurasi sistem dapat mengenali inputan berupa suara dan apakah inputan yang diberikan dapat berfungsi pada motor. Pengujian ini dilakukan sebanyak 20 kali percobaan yang dibagi menjadi 9 Subclass dengan menggunakan ciri statistik Mean, Skew dan Kurt pada 70 data latih 10 data uji, hasilnya dapat dilihat pada tabel 4.3 berikut :

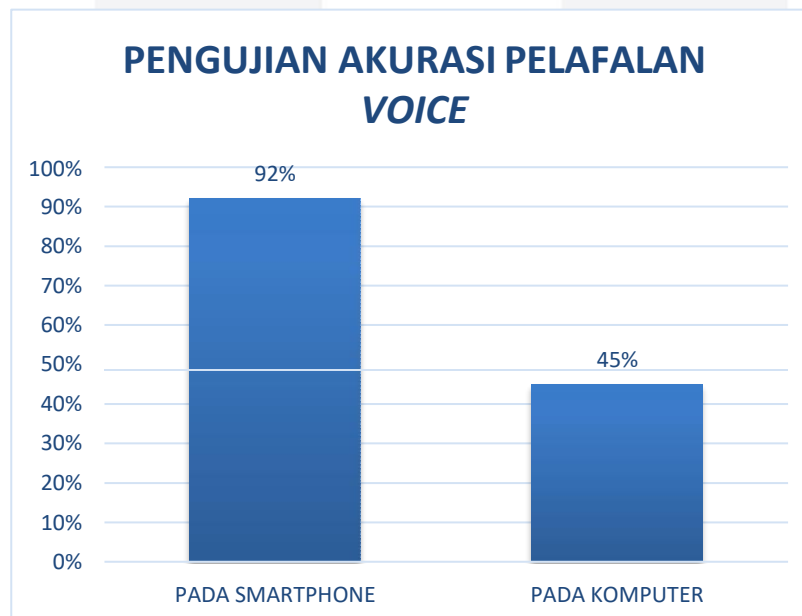
Tabel 4.3 Akurasi Pelafalan Voice Dari Aplikasi Matlab Pada Firebase

	Nyalakan Kunci Rahasia	Matikan Kunci Rahasia	Nyalakan Alarm	Matikan Alarm	Nyalakan Mesin	Matikan Mesin	Nyalakan Kunci	Matikan Kunci	Kirim Lokasi
Jumlah Benar	10	12	11	10	4	5	7	10	12
Jumlah Salah	10	8	9	10	16	15	13	10	8

Berdasarkan tabel 4.3 sistem dapat mengenali 9 inputan perintah. Dan yang terbaik yaitu ketika diberikan inputan “Matikan Kunci Rahasia” dan “Kirim Lokasi”, dimana jumlah benar yang diperoleh yaitu 12. Rata rata Akurasi yang didapat pada pengujian ini yaitu sebesar 45%.

4.2.6 Perbandingan Akurasi Antara Aplikasi Smartphone dan Komputer

Pengujian ini bertujuan untuk membandingkan hasil akurasi pelafalan Voice dari motor ke Firebase dengan akurasi pelafalan Voice dari aplikasi ke Firebase. Perbandingan akurasi antara pengujian pada aplikasi Smartphone dan komputer terdapat pada gambar 4.3 berikut :



Gambar 4.2 Pengujian Akurasi Pelafalan Voice

Berdasarkan gambar 4.3 didapatkan perbedaan antara pengujian akurasi pelafalan Voice pada aplikasi Smartphone dan pada komputer menggunakan aplikasi matlab. Pada aplikasi Smartphone didapatkan hasil 92% dikarenakan Library yang digunakan adalah milik Google dan sudah melalui berbagai macam Research yang sangat panjang, sehingga pengenalan suara dapat mencapai hasil yang maksimal. Sedangkan pada komputer yang menggunakan aplikasi matlab, didapatkan hasil 45 %, dikarenakan Library yang digunakan masih cukup minim sehingga akurasi yang didapat belum mendapatkan hasil yang maksimal. Pada perbandingan ini, aplikasi Smartphone memiliki akurasi yang lebih besar daripada akurasi pada komputer, maka ketika ada inputan suara yang masuk kedalam aplikasi Smartphone, akan lebih cepat untuk mendeteksi Voice Recognition dari suara kita dan memulai Controlling System.

5. Kesimpulan

Berdasarkan hasil pengujian penelitian sistem keamanan sepeda motor berbasis *Smartphone* dan komputer, maka ada beberapa hal yang dapat penulis simpulkan, yaitu :

1. Dari hasil penelitian pada tugas akhir ini, kedua sistem yang dibangun dapat diimplementasikan dan dapat dikoneksikan ke sepeda motor dengan lancar. Mulai dari menyalakan dan mematikan kunci, menyalakan dan mematikan alarm, menyalakan dan mematikan mesin. Dengan rata-rata *Response Time* dari semua perintah yaitu sebesar 4.367 detik pada aplikasi *Smartphone* dan 4.921 detik pada komputer
2. Teknologi *Speech to Text* pada sepeda motor dapat diimplementasikan pada sistem baik melalui inputan dari aplikasi *Smartphone* ataupun dari komputer.
3. Hasil pengujian *Speech Recognition* atau pengujian pelafalan *Voice* pada kedua sistem, pada aplikasi *Smartphone* mendapatkan tingkat akurasi sebesar 92% sedangkan pada komputer mendapatkan tingkat akurasi sebesar 45%

Referensi

- [1] BPS, "Statistik Kriminal 2019," *Badan Pus. Stat.*, pp. 1–218, 2019
- [2] M. Irsyad *et al.*, "Sistem Keamanan Sepeda Motor Menggunakan Rfid , Gsm Modul , Gps Modul Berbasis Mikrokontroler, Motorcycle Security System Using Rfid , Gsm Module , Gps Modules Based on Microcontroller," 2019.
- [3] Syamsudin, Muhamad. 2011. *Sistem Keamanan Kendaraan dari Pencurian Berbasis Jaringan Sensor Nirkabel*. Seminar Tugas Akhir. FTI- ITS.
- [4] Mahardika Raden Bayu Zaky. 2014. *Sistem Pengaman Kendaraan Bermotor Roda Dua Menggunakan Kunci Kontak Wireless Berbasis Mikrokontroler*. TELEKONTRAN, Vol. 2, No. 1, November 2014. UNIKOM
- [5] R. Patmasari, S. Saidah, A. F. Akbar, and R. Magdalena, "Design and Implementation of Learning Tools To Read the Braille Letters Based on Voice Processing and Arduino Using Mel Frequency Cepstral Coefficient and K- Nearest Neighbor Method," *J. Meas. Electron. Commun. Syst.*, vol. 6, no. 1, p. 28, 2020, doi: 10.25124/jmecs.v6i1.2019.
- [6] D.W.Akhmad, A.Andi, H.Dodi, "Perancangan Aplikasi Voice Command Recognition Berbasis Android Dan Arduino UNO". Mercubuana, 2016
- [7] A. Suroso, Y. Fitri, S. F. Retnowaty, and Nurkhamdi, "Aplikasi Pengenalan Ucapan dengan Ekstraksi Ciri Mel Frequency Cepstral Coefficient (MFCC) dan Jaringan Syaraf Tiruan (JST) Propagasi Balik untuk Buka dan Tutup Pintu," *Jurnal Komputer Terapan* Vol. 1, No. 2, November 2015
- [8] T. Chamidy, "Metode Mel Frequency Cepstral Coeffisients (MFCC) Pada klasifikasi Hidden Markov Model (HMM) Untuk Kata Arabic pada Penutur Indonesia," *Matics*, vol. 8, no. 1, pp. 36–39, 2016
- [9] S. Dhingra, G. Nijhawan, and P. Pandit, "Isolated speech recognition using MFCC and DTW," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 2, no. 8, pp. 4085–4092, 2013
- [10] D. P. Pamungkas, and F.R. Hariri, "Implementasi Metode PCA dan City Block Distance untuk Presensi Mahasiswa Berbasis Wajah," *Seminar Nasional Teknologi, Informasi, Komunikasi dan Aplikasinya*, vol. 4, pp. 190-194. des 2017.
- [11] R. T. Wahyuni, D. Prastiyanto, and E. Suprpto, "Penerapan Algoritma Cosine Similarity dan Pembobotan TF-IDF pada Sistem Klasifikasi Dokumen Skripsi," *Skripsi. Fakultas Teknik Elektro. Universitas Negeri Semarang*. 2018.
- [12] S. K. Gaikwad, B. W. Gawali, and P. Yannawar, "A Review on Speech Recognition Technique," *Int. J. Comput. Appl.*, vol. 10, no. 3, pp. 16–24, 2010."
- [13] R. M. Ikhsanudin, "Identifikasi Citra Pada Plat Nomor Kendaraan Mobil Pribadi Menggunakan Metode K-Nearest Neighbour," *Skripsi. Fakultas Ilmu Komputer. Universitas Dian Nuswantoro*. 2014.
- [14] Z. D. Dewi Lusita Hidayati Nurul, Rohmah F mimin, "Prototipe Smart Home Dengan Modul Nodemcu Esp8266 Berbasis Internet of Things (Iot)," p. 3, 2019.
- [15] A. Asnil and I. Husnaini, "Analisis riak keluaran buck converter," *Seminar Nasional, FORTEI 2015*, no. 561, pp. 58– 62, 201