

Penerapan Euclidean Distance dan Manhattan Distance untuk Penentuan Jarak Kendaraan dengan GPS

Hadi Juanda Fani
Universitas Buana Perjuangan
Karawang, Indonesia
if17.hadifani@mhs.ubpkarawang.ac.id

Yana Cahyana
Universitas Buana Perjuangan
Karawang, Indonesia
yana.cahyana@ubpkarawang.ac.id

Tohirin Al Mudzakir
Universitas Buana Perjuangan
Karawang, Indonesia
tohirin@ubpkarawang.ac.id

Abstract—

Kemajuan teknologi kini berkembang luas di hampir semua bidang, berkembang hingga merambah pada sebuah sistem keamanan kendaraan. Banyaknya kasus pencurian kendaraan, khususnya di Kabupaten Karawang, mendorong perlunya sistem keamanan pada kendaraan dengan menggunakan GPS (Global Positioning System). Dengan penambahan fungsi penyimpanan data lokasi kendaraan yang dapat tersimpan di Firebase sebagai penyimpanan data yang bersifat pribadi, maka diperlukan unit tambahan seperti modul sensor GPS Ublox Neo 6 yang terhubung dengan modul Wemos D1 R1. Penghitungan jarak yang memiliki variabel latitude dan longitude dapat dilakukan dengan metode Euclidean Distance, sedangkan metode Manhattan Distance menghasilkan deviasi paling besar karena konsep perhitungan jaraknya menerapkan pencarian selisih murni antar data. Hal tersebut kurang cocok untuk perhitungan jarak yang menggunakan variabel koordinat latitude dan longitude.

Kata kunci — Euclidean Distance, Firebase, GPS, Keamanan Kendaraan, Manhattan Distance, Ublox Neo 6, Wemos D1 R1

I. PENDAHULUAN

Kemajuan teknologi kini berkembang luas di hampir semua bidang, terus berkembang hingga merambah pada sebuah sistem keamanan kendaraan. Banyaknya kasus pencurian sepeda motor, khususnya di Kabupaten Karawang, berawal dari kelalaian pemilik hingga sistem keamanan kunci kendaraan yang tidak sepenuhnya aman.

Hasil data yang didapat dari situs website resmi Pusat Informasi Kriminal Nasional pada tautan https://pusiknas.polri.go.id/peta_kriminalitas diakses pada tanggal 14 April tahun 2022. Kasus pencurian kendaraan, khususnya di Kabupaten Karawang, memiliki 2.811 kasus terhitung dari tahun 2018 sampai dengan tahun 2022.

Sistem pemantauan saat ini merupakan salah satu solusi untuk masalah yang sudah dijelaskan di atas. Dengan menggunakan sistem pemantauan, maka dapat diketahui posisi kendaraan. Teknologi real-time memungkinkan pembangunan sistem pengawasan; informasi lokasi kendaraan akan langsung diketahui saat kendaraan bergerak dan dapat diakses melalui situs website.

Adapun penelitian yang berhubungan dengan pengembangan sistem pemantauan keamanan kendaraan yang dapat dikendalikan melalui internet, hasil penelitian tersebut memberikan informasi keakuratan rata-rata daya lebih dari 85%. Semakin banyak sensor yang digunakan, semakin efektif sistem keamanan sepeda motor dalam mengurangi pencurian sepeda motor [1].

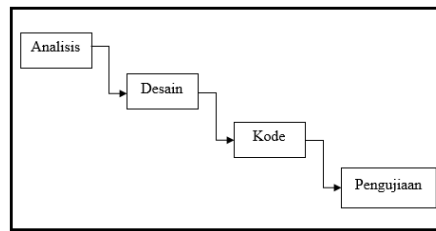
Penelitian sistem keamanan sepeda motor berbasis Internet of Things (IoT) menggunakan smartphone dan NodeMCU, objek dicari di Google Maps berdasarkan koordinat lintang dan bujur yang dikonversi ke angka desimal. Google Maps digunakan untuk memantau objek, dan aplikasi Blynk menggunakan konsep Internet of Things (IoT) untuk kontrol dan penerimaan data melalui media komunikasi yang dapat diuji melalui integrasi modul NodeMCU, sensor getaran SW-420, dan relay 4 channel yang dilakukan dan ditampilkan secara bersamaan [2].

Berdasarkan permasalahan keamanan kendaraan sebelumnya, maka diperlukan sistem keamanan kendaraan dengan menggunakan GPS (Global Positioning System), dan ditambahkan fungsi penyimpanan data lokasi kendaraan yang dapat disimpan pada layanan Firebase sebagai penyimpanan data pribadi. Alat pelacak GPS yang dipasang pada kendaraan menyediakan penyimpanan data GPS sebagai riwayat keberadaan kendaraan. Kemudian, dibutuhkan unit tambahan seperti modul sensor GPS Ublox Neo 6 yang terhubung dengan modul Wemos D1 R1 sebagai penunjang pencarian untuk peralatan kendaraan. Untuk menjawab permasalahan tersebut, penulis memilih judul “Penerapan Euclidean Distance dan Manhattan Distance untuk Penentuan Jarak Kendaraan dengan GPS.”

II. METODE PENELITIAN

A. Prosedur Penelitian

Proses penelitian dengan menggunakan metode Waterfall merupakan salah satu tahapan dalam proses pengembangan. Langkah-langkah tersebut dapat dilihat pada gambar di bawah ini.



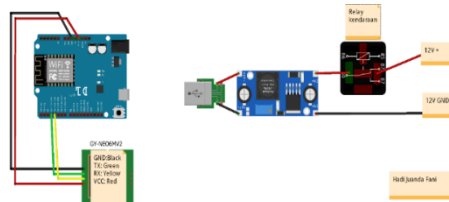
Gambar 1 Model *Waterfall*

1. Analisis

Analisis merupakan proses untuk menentukan alat dan data yang dibutuhkan. Analisis berupa studi pustaka jurnal yang dianggap relevan dengan proses pembuatan perangkat GPS dengan metode analisis deskriptif, serta landasan dasar dalam pembuatan perangkat dan laporan dengan melakukan pendekatan penelitian.

2. Desain

Dalam laporan ini, penulis menggunakan tiga desain, yaitu desain alat, desain website, dan desain sistem. Berikut merupakan tampilan desain alat dari perancangan alat menggunakan aplikasi Fritzing:

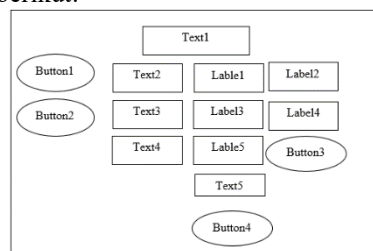


Gambar 2 Desain Alat

Penghubungan kabel dari setiap modul adalah sebagai berikut :

- Sumber tegangan positif (+) 12V DC diambil dari relay motor yang berfungsi sebagai stop kontak kendaraan untuk memberikan tegangan pada modul step-down LM2599.
- Modul step-down LM2599 diatur agar menghasilkan tegangan 10V DC sebagai sumber tegangan pada modul Wemos D1 R1.
- Modul GPS Ublox Neo 6 membutuhkan tegangan 5V DC yang bersumber dari modul Wemos D1 R1. Modul ini mempunyai 4 (empat) pin, yaitu VCC ke 5V DC, GND pada GND, RX pada pin ke-4, dan TX pada pin ke-5.

Tampilan desain *website*, adalah sebagai berikut:



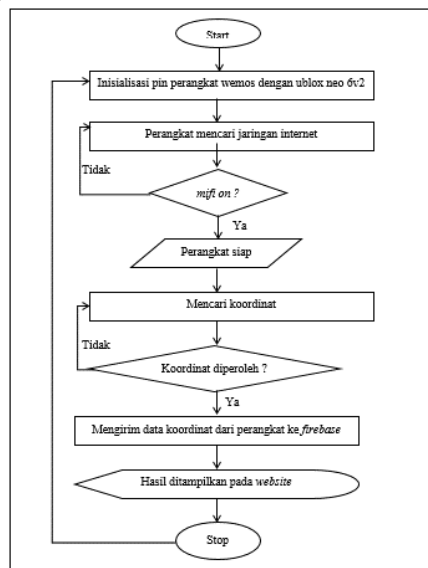
Gambar 3 Tampilan *Website*

Fungsi dari bagian yang ada pada tampilan utama:

- Text1* diperuntukan sebagai judul dari nama *website*.
- Text2* diperuntukan sebagai nama koordinat pemilik.
- Lable1* diperuntukan sebagai mengisi nilai *latitude*.
- Lable2* diperuntukan sebagai mengisi nilai *longitude*.
- Text3* diperuntukan sebagai nama koordinat kendaraan.
- Lable3* diperuntukan sebagai menampilkan nilai *latitude*.
- Lable4* diperuntukan sebagai menampilkan nilai *longitude*.
- Text4* diperuntukan sebagai nama jarak.
- Lable5* diperuntukan sebagai menampilkan jarak.
- Text5* diperuntukan sebagai kalimat petunjuk.
- Button1* diperuntukan sebagai penampil modal *component*, yang isinya informasi dari *website*.
- Button2* diperuntukan sebagai penampil modal *component*, yang isinya cara penggunaan dari *website*.

- m) *Button3* diperuntukan sebagai penampil jarak.
n) *Button4* diperuntukan sebagai penampil peta yang terkoneksi ke *google maps*.

Desain sistem digunakan untuk menggambarkan proses kerja sistem pada cara kerja alat pelacak kendaraan dengan menggunakan GPS dalam pengembangan atau dalam bentuk flowchart.



Gambar 4 Flowchart Proses Kerja

Flowchart menunjukkan keputusan mana yang dapat dikonfirmasi dalam situasi tertentu. Sistem ini membuat diagram aktivitas yang menggambarkan aktivitas pengguna saat menggunakan sistem. Cara kerja dari alat ini terdiri dari rangkaian Wemos D1 R1 sebagai penghubung antara alat GPS dengan MiFi.

Tahapan proses kerja perangkat, adalah sebagai berikut :

- Menginisialisasi perangkat Wemos dengan Ublox Neo 6.
- Perangkat mencari koneksi internet melalui MiFi. Setelah perangkat terkoneksi dengan internet, maka perangkat siap digunakan. Apabila perangkat tidak mendapatkan koneksi internet, maka proses kembali mengidentifikasi koneksi internet.
- Membaca data koordinat saat perangkat dinyalakan. Jika data sudah didapat berupa longitude dan latitude, maka perangkat melalui modul Wemos mengirim data ke Firebase. Apabila perangkat tidak mendapatkan koordinat, maka proses kembali mencari sinyal internet.
- Pencarian website dengan alamat hosting yang dimiliki oleh layanan Firebase, dan hasil data dapat dilihat melalui website.

3. Kode

Kode merupakan proses pengkodean perangkat GPS. Aplikasi IDE (Integrated Development Environment) diperlukan saat merancang alat ini. Proyek akan menggunakan bahasa pemrograman mikrokontroler, yaitu bahasa C, yang dikompilasi saat membuat perangkat lunak.

Proses koding membutuhkan library untuk keperluan yang dibutuhkan dalam proses pembuatan perangkat dengan software Arduino IDE agar bisa saling berkomunikasi dengan perangkat lainnya.

4. Pengujian

Proses pengujian pada laporan ini menggunakan metode pengujian White Box dan Black Box, di mana hasil pengujian akan menampilkan nilai akhir dan fungsionalitas dari perangkat lunak yang sudah dibuat. Pengujian perangkat akan dievaluasi oleh penulis. Jika perangkat pertama yang dievaluasi mengalami kegagalan, maka proses kembali ke tahap desain dan kode.

Proses pengujian dilakukan untuk memeriksa apakah perangkat memenuhi persyaratan yang diharapkan dan berhasil. Proses pengujian ini dijalankan untuk menguji keakuratan pencarian GPS yang digunakan.

Hasil dari data yang diperoleh akan dihitung keakuratannya dengan membandingkan hasil perhitungan jarak Google Maps dengan algoritma Euclidean Distance dan Manhattan Distance.

Euclidean Distance adalah metode perhitungan jarak yang mengukur jarak antara dua titik dalam ruang Euclidean, meliputi 2D, 3D, dan bahkan lebih banyak bidang Euclidean [3]. Metode Euclidean Distance memiliki rumus sebagai berikut:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Ket:

d = Jarak

x^2 = Latitude modul GPS

x^1 = Latitude google maps

y^2 = Longitude modul GPS

y^1 = Longitude google maps

Manhattan distance digunakan untuk menghitung perbedaan absolut (mutlak) antara koordinat sepasang objek [4], metode *manhattan distance* dengan rumus sebagai berikut:

$$d(x, y) = \sum_{i=1}^n |x_i - y_i|$$

Ket:

d(x,y) = Jarak

x = Koordinat lokasi 1

y = Koordinat lokasi 2

Algoritma 1 hitung jarak dengan metode *euclidean distance* :

Input : x1, y1, x2, y2

Algoritma :

a) Baca koordinat awal kendaraan (x1, y1)

b) Baca koordinat akhir kendaraan (x2, y2)

c) Hitung jarak = $\text{SQRT}((x2-x1)^2 + (y2-y1)^2)$

Output : Jarak

Algoritma 2 hitung jarak dengan metode *manhattan distance* :

Input : x1, y1, x2, y2

Algoritma :

a) Baca koordinat awal kendaraan (x1, y1)

b) Baca koordinat akhir kendaraan (x2, y2)

c) Hitung jarak = $\text{Math.abs}(X2 - X1) + \text{Math.abs}(Y2 - Y1)$

Output : Jarak

III. HASIL DAN PEMBAHASAN

A. Analisis

Hasil dari analisis yang diperoleh berdasarkan studi jurnal mengenai penghitungan jarak pada kendaraan menggunakan GPS dengan metode analisis deskriptif dapat dilakukan pada perangkat atau modul dari IoT yang terkoneksi dengan jaringan internet dan fitur pendukung berupa database.

Kebutuhan yang harus dipersiapkan dalam pembuatan perangkat GPS di antaranya yaitu Wemos D1 R1, Ublox Neo 6, step-down LM2599, perangkat MiFi. Untuk proses kode dilakukan pada aplikasi Arduino IDE, Visual Studio Code, dan sebagai media database menggunakan layanan Firebase.

B. Desain

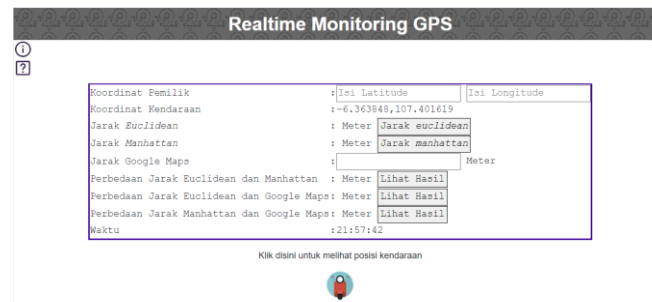
Instalasi aplikasi Fritzing diperuntukkan sebagai desain prototipe elektronika yang dibutuhkan dalam pembuatan perangkat GPS, dengan memasukkan komponen yang diperlukan pada lembar kerja aplikasi Fritzing.

Penjelasannya sudah dijelaskan pada Gambar 2. Berikut merupakan perangkat GPS yang sudah didesain sebelumnya, dengan tampilan seperti gambar di bawah ini:



Gambar 5 Perangkat GPS

Tampilan website dengan data yang diperoleh dari Realtime Database dapat diakses melalui web browser pada laptop. Namun, pada perangkat telepon genggam, layanan Google Maps bawaan harus dihentikan terlebih dahulu untuk menampilkan koordinat yang ditampilkan pada website.



Gambar 6 Tampilan Website Halaman Utama

C. Kode

Proses kode untuk perangkat GPS dilakukan pada aplikasi Arduino IDE, dan untuk pembuatan Realtime Database beserta hosting yang dimiliki oleh Firebase harus menggunakan akun Google. Pembuatan HTML dilakukan pada aplikasi Visual Studio Code.

Instalasi Arduino IDE dimaksudkan sebagai proses penulisan kode dengan modul Wemos D1 R1. Papan Wemos menggunakan chip pemrograman CH340, sehingga perlu menginstal driver CH340 agar dapat menghubungkan Wemos ke laptop. Tambahkan papan Wemos ke Arduino IDE dengan cara masuk ke Berkas lalu Preferences. Ketika Preferences terbuka, masukkan tautan berikut ini: http://arduino.esp8266.com/stable/package_esp8266com_index.json pada bagian Additional Board Manager URLs, lalu klik OK.

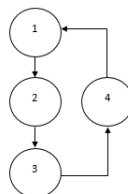
Selanjutnya, masuk ke Alat, klik Papan, kemudian klik Boards Manager. Setelah Boards Manager terbuka, masuk ke kotak pencarian dan ketik ESP8266.

Library yang dibutuhkan untuk proses kode di antaranya adalah ESP8266WiFi, TinyGPS++, SoftwareSerial, TimeLib, dan FirebaseESP8266.

Untuk membuat proyek Firebase dengan pengolahan website, hal yang harus dilakukan adalah mendaftar ke Google Firebase untuk mendapatkan fasilitas Realtime Database dan hosting. Kemudian, pilih Realtime Database sampai mendapatkan alamat hosting yang diberikan oleh Firebase. Selanjutnya, masukkan key, child, dan value.

D. Pengujian

Pengujian White Box adalah menguji dan menganalisis kesalahan kode program serta mengabaikan indikasi [5]. Pengujian dilakukan pada kode HTML yang melakukan proses penghitungan jarak antara koordinat kendaraan dan koordinat pemilik. Oleh karena itu, dilakukan pemetaan dari beberapa kode yang dapat ditampilkan, serta susunan flowgraph dari kode penghitungan Euclidean Distance dan Manhattan Distance, sebagai berikut:



Gambar 7 Flowgraph Kode Penghitungan

Pada flowgraph di atas dijelaskan bahwa:

- Nomor 1 adalah proses memasukkan nilai x1 dan y1 yang diperoleh dari koordinat pemilik kendaraan, serta menampilkan nilai x2 dan y2.
- Nomor 2 merupakan metode yang akan menghitung jarak antara x1, x2, y1, dan y2.
- Nomor 3 merupakan hasil dari proses penghitungan koordinat.
- Nomor 4 menunjukkan bahwa jika nilai x1 dan y1 kosong, maka akan ditampilkan pop-up dan hasil penghitungan akan bernilai nol.

Penghitungan Cyclomatic Complexity menggunakan rumus $V(G) = E - N + 2$, di mana E (Edge) = 4 dan N (Node) = 4. Maka, hasil yang diperoleh berdasarkan rumus adalah sebagai berikut:

$$V(G) = E - N + 2$$

$$V(G) = 4 - 4 + 2$$

$$V(G) = 0 + 2$$

$$V(G) = 2$$

Berdasarkan penghitungan dari *cyclomatic complexity* maka menghasilkan nilai sebanyak 2 *independent path* yaitu :

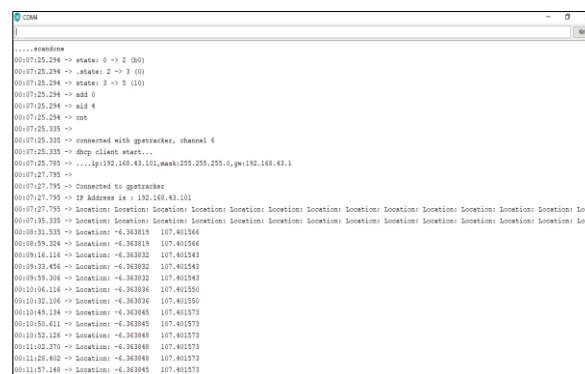
Path 1 : 1-2-3

Path 2 : 1-2-3-4

Pengujian Black Box merupakan salah satu metode pengujian yang berfokus pada spesifikasi fungsional perangkat lunak. Pengujian ini memberikan gambaran atas sekumpulan kondisi masukan dan melakukan pengujian pada uraian fungsional program [6].

Pengujian perangkat GPS dilakukan dengan menjalankan monitor serial pada Arduino IDE. Perangkat GPS mulai menginisialisasi jaringan MiFi yang sudah diatur sebagai sumber jaringan internet dan berhasil mendapatkan jaringan MiFi pada waktu 00:07:25.335, serta mulai menginisialisasi titik koordinat pada waktu 00:07:27.795. Titik koordinat berhasil didapatkan pada waktu 00:08:31.535.

Dari setiap percobaan, penginisialisasian titik koordinat selalu berubah karena perangkat GPS terus mencoba mencari titik koordinat yang tepat. Rata-rata, perangkat GPS memerlukan waktu sekitar 5 detik untuk menemukan titik koordinat yang sesuai.



Gambar 8 Tampilan *Monitor Serial*

Menguji database pada layanan Firebase dilakukan untuk memastikan apakah data yang dikirim oleh perangkat GPS telah terkonfirmasi oleh Firebase atau belum. Jika database siap, maka akan menampilkan titik koordinat yang diperoleh dari perangkat GPS. Contoh tampilannya seperti di bawah ini:



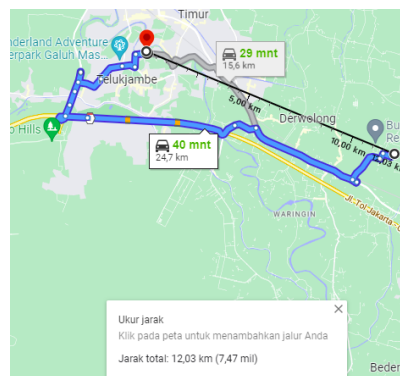
Gambar 9 Tampilan *Database Firebase*

Tampilan Firebase pada Realtime Database menunjukkan hasil dari pencarian posisi koordinat yang diperoleh dari perangkat GPS. Setiap perubahan data yang diterima oleh perangkat GPS akan secara otomatis mengubah data yang terdapat pada Realtime Database. Begitu pula tampilan website yang sudah memiliki hosting dari layanan Firebase akan menampilkan data yang sama dengan data yang diperoleh dari Realtime Database.

Pengujian dilakukan untuk mengukur kinerja sistem yang telah dibuat dan mengetahui tingkat akurasi. Pengujian dilakukan dengan cara memasukkan nilai latitude dan longitude, yaitu dengan memasukkan koordinat yang diperoleh dari posisi pemilik kendaraan dan koordinat kendaraan yang diperoleh dari perangkat GPS. Selanjutnya, koordinat tersebut dihitung menggunakan rumus Euclidean Distance dan Manhattan Distance, kemudian dibandingkan dengan jarak yang dihitung melalui Google Maps sebagai acuan standar kelayakan metode pengukuran jarak. Berikut merupakan hasil uji coba perangkat yang ditampilkan pada website dan Google Maps.

Koordinat Pemilik	: -6.323510	107.301380
Koordinat Kendaraan	: -6.363848, 107.401619	
Jarak Euclidean	: 12028.22 Meter	Jarak euclidean
Jarak Manhattan	: 15649.02 Meter	Jarak manhattan
Jarak Google Maps	: 12030	Meter
Perbedaan Jarak Euclidean dan Manhattan	: 3620.80 Meter	Lihat Hasil
Perbedaan Jarak Euclidean dan Google Maps	: 1.78 Meter	Lihat Hasil
Perbedaan Jarak Manhattan dan Google Maps	: 3619.02 Meter	Lihat Hasil
Waktu	: 23:04:19	

Gambar 10 Tampilan Website Proses Perhitungan Koordinat



Gambar 11 Tampilan Google Maps Menunjukkan Jarak Garis Lurus

Berdasarkan Gambar 10, koordinat pemilik akan dihitung dengan koordinat kendaraan untuk menghasilkan jarak menggunakan metode Euclidean dan Manhattan. Perolehan nilai jarak dari Google Maps ditampilkan pada Gambar 11.

Perbedaan nilai jarak Euclidean dan Manhattan diperoleh dengan cara mengurangkan hasil jarak Euclidean dengan jarak Manhattan. Begitu pula perbedaan antara jarak Euclidean dan jarak Google Maps, serta antara jarak Manhattan dan jarak Google Maps, dihitung dengan cara pengurangan nilai.

Hasil uji penghitungan jarak antara koordinat kendaraan dan koordinat pemilik menggunakan metode Euclidean Distance dan Manhattan Distance dapat dilihat pada tabel di bawah ini.

Tabel 1 Hasil Pengujian Penghitungan

Uji ke-	Koordinat Pemilik	Koordinat Kendaraan	Jarak Berdasarkan Google Map	Jarak Menurut Euclidean	Jarak Menurut Manhattan	Hasil Perbedaan Euclidean dan Manhattan
1.	-6.363848 107.401619	-6.323510 107.301380	12,030 m	12,028.22 m	15,649.02 m	3,620.80 m
2.	-6.363848 107.401619	-6.248048 107.581805	23,680 m	23,843.42 m	32,949.13 m	9,105.71 m
3.	-6.363848 107.401619	-6.041800 107.345181	36,290 m	36,396.69 m	42,133.02 m	5,736.33 m
4.	-6.363848 107.401619	-6.147679 107.334138	25,300 m	25,209.16 m	31,575.89 m	6,366.73 m
5.	-6.363848 107.401619	-6.279935 107.460594	11,420 m	11,417.45 m	15,906.28 m	4,488.83 m
6.	-6.363848 107.401619	-5.990182 107.131275	51,180 m	51,341.54 m	71,691.12 m	20,349.8 m
7.	-6.363848 107.401619	-6.198428 107.610590	29,530 m	29,668.92 m	41,677.16 m	12,008.24 m
8.	-6.363848 107.401619	-6.436631 107.409796	8,140 m	8,153.17 m	9,012.46 m	859.29 m
9.	-6.363848 107.401619	-6.314649 107.241421	18,530 m	18,655.28 m	23,310.05 m	4,654.77 m

10.	-6.363848 107.401619	-6.108023 107.277154	31,600 m	31,670 m	42,333.84 m	10,663.78 m
11.	-6.363848 107.401619	-6.162219 107.430341	22,640 m	22,671.90 m	25,642.65 m	2,970.75 m
12.	-6.363848 107.401619	-5.978338 107.307268	44,120 m	44,181 m	53,418.07 m	9,236.54 m
13.	-6.363848 107.401619	-6.314587 107.407788	5,520 m	5,526.56 m	6,170.46 m	634.90 m
14.	-6.363848 107.401619	-6.393792 107.469092	8,170 m	8,217.53 m	10,844.45 m	2,626.92 m
15.	-6.363848 107.401619	-6.165283 107.478287	23,650 m	23,694.68 m	30,638.91 m	6,944.23 m
16.	-6.363848 107.401619	-6.369422 107.395527	915 m	919.19 m	1,298.66 m	379.47 m
17.	-6.363848 107.401619	-6.125664 107.374071	26,660 m	26,691 m	29,581.26 m	2,889.89 m
18.	-6.363848 107.401619	-6.284120 107.462961	11,160 m	11,198.25 m	15,703.90 m	4,505.65 m
19.	-6.363848 107.401619	-6.278026 107.389397	9,640 m	9,650.09 m	10,914.25 m	1,264.16 m
20.	-6.363848 107.401619	-6.329819 107.475222	8,970 m	9,026.78 m	11,981.58 m	2,954.80 m
Hasil rata – rata			20,457 m	20,508.04 m	26,121.61 m	5,613.06 m

Pengujian penghitungan jarak dengan metode Euclidean Distance dan metode Manhattan Distance diperoleh nilai jarak rata-rata dari 20 kali percobaan. Perolehan dari metode Euclidean Distance menghasilkan nilai rata-rata sebesar 20.508,04 meter, sedangkan metode Manhattan Distance menghasilkan nilai rata-rata sebesar 26.121,61 meter. Maka, hasil uji coba menunjukkan bahwa jarak rata-rata yang diperoleh dengan metode Euclidean Distance lebih mendekati jarak sebenarnya, yaitu jarak rata-rata dari Google Maps sebesar 20.457 meter.

Perbedaan hasil dari setiap metode perhitungan jarak dapat disebabkan oleh konsep dasar masing-masing metode. Metode Euclidean Distance menerapkan konsep Pythagoras dan dapat digunakan pada bidang 1, 2, dan 3 dimensi. Sementara itu, metode Manhattan Distance menghasilkan bias terbesar karena menggunakan konsep pencarian selisih murni antar data, yang kurang sesuai untuk penghitungan jarak menggunakan variabel koordinat latitude dan longitude.

IV. KESIMPULAN DAN SARAN

Setelah dilakukan percobaan dengan beberapa tahapan, maka dapat disimpulkan bahwa perangkat GPS berhasil menampilkan koordinat kendaraan yang dapat ditampilkan pada website dan tersimpan pada Firebase. Hasil penelitian menunjukkan bahwa metode jarak Euclidean memiliki akurasi yang lebih baik dibandingkan dengan metode jarak Manhattan.

Perangkat GPS ini sangat bergantung pada jaringan internet, karena jika jaringan internet hilang secara tiba-tiba, maka perangkat GPS tidak dapat mengirimkan data koordinat ke layanan Firebase. Oleh karena itu, sangat dianjurkan agar perangkat MiFi menggunakan provider GSM yang kuat dan stabil sebagai sumber jaringan internet.

Penghitungan jarak yang menggunakan variabel latitude dan longitude dapat dilakukan dengan metode Euclidean Distance. Untuk memperoleh tingkat akurasi jarak yang lebih baik, dapat dipertimbangkan penggunaan metode lain selain metode yang digunakan dalam laporan ini.

PENGAKUAN

Naskah ilmiah ini merupakan bagian dari penelitian pada Tugas Akhir milik Hadi Juanda Fani dengan judul Penerapan Euclidean Distance dan Manhattan Distance untuk Penentuan Jarak Kendaraan dengan GPS, yang dibimbing langsung oleh Bapak Yana Cahyana, M.Kom., dan Bapak Tohirin Al Mudzakir, M.Kom.

DAFTAR PUSTAKA

- [1] H. Sujadi, T. F. Prasetyo, and P. Paisal, "Pengembangan Sistem Monitoring Keamanan Sepeda Motor Berbasis Internet of Things," *J-Ensitem*, vol. 5, no. 01, pp. 226–231, 2018, doi: 10.31949/j-ensitem.v5i01.1209.
- [2] A. P. Putra, "Sistem Keamanan Sepeda Motor Berbasis Iot (Internet of Things) Dengan Smartphone Menggunakan Nodemcu," *JIT (Jurnal Teknol. Terpadu)*, vol. 9, no. 1, pp. 77–87, 2021, doi: 10.32487/jtt.v9i1.1112.
- [3] M. Nishom, "Perbandingan Akurasi Euclidean Distance, Minkowski Distance, dan Manhattan Distance pada Algoritma K-Means Clustering berbasis Chi-Square," *J. Inform. J. Pengemb. IT*, vol. 4, no. 1, pp. 20–24, 2019, doi: 10.30591/jpit.v4i1.1253.
- [4] Y. Miftahuddin, S. Umaroh, and F. R. Karim, "Perbandingan Metode Perhitungan Jarak Euclidean, Haversine, Dan Manhattan Dalam Penentuan Posisi Karyawan," *J. Tekno Insentif*, vol. 14, no. 2, pp. 69–77, 2020, doi: 10.36787/jti.v14i2.270.

- [5] C. T. Pratala, E. M. Asyer, I. Prayudi, and A. Saifudin, "Pengujian White Box pada Aplikasi Cash Flow Berbasis Android Menggunakan Teknik Basis Path," *J. Inform. Univ. Pamulang*, vol. 5, no. 2, p. 111, 2020, doi: 10.32493/informatika.v5i2.4713.
- [6] M. Nurudin, W. Jayanti, R. D. Saputro, M. P. Saputra, and Y. Yulianti, "Pengujian Black Box pada Aplikasi Penjualan Berbasis Web Menggunakan Teknik Boundary Value Analysis," *J. Inform. Univ. Pamulang*, vol. 4, no. 4, p. 143, 2019, doi: 10.32493/informatika.v4i4.3841.