

Sistem Pemetaan Kerusakan Jalan Dengan Pengembangan *Internet Of Things*

1st William Wafi Wisnutama

Fakultas Teknik Elektro

Universitas Telkom

Bandung, Indonesia

wafiwisnutama@student.telkomuniversit
y.ac.id

2nd Aloysius Adya Pramudita

Fakultas Teknik Elektro

Universitas Telkom

Bandung, Indonesia

pramuditaadya@telkomuniversity.ac.id

3rd Harfan Hian Ryanu

Fakultas Teknik Elektro

Universitas Telkom

Bandung, Indonesia

harfanhr@telkomuniversity.ac.id

Abstrak - Pesatnya perkembangan teknologi saat ini telah mengubah berbagai aspek dalam kehidupan masyarakat, salah satunya adalah transportasi. Transportasi menjadi kebutuhan masyarakat tiap harinya. Namun, tingkat kepadatan masyarakat yang meningkat juga mempengaruhi jumlah transportasi. Sehingga, meningkatnya jumlah transportasi tentunya akan mempengaruhi kondisi jalan raya yang dilalui. Kondisi jalan yang rusak akan memperlambat dan membatasi dalam aktivitas, sehingga menimbulkan kemacetan pada jalan.

Untuk memberikan penyelesaian atas permasalahan yang terjadi, *Capstone Design* menghadirkan inovasi baru yaitu *Obstac Map* yang merupakan bentuk pengembangan sistem deteksi dan pemetaan berbasis *Internet of Things*. Sistem ini bertujuan untuk efisiensi dan keamanan berkendara di jalan raya. Sistem ini menggabungkan sensor dengan mikrokontroler sehingga dapat mendeteksi anomali jalan raya dan melakukan pemetaan lokasi jalan yang rusak. *Obstac Map* juga didukung oleh *mobile application* dan kamera yang dapat digunakan oleh pengguna selama berkendara untuk memperoleh informasi terkait jalan yang dilaluinya dalam bentuk koordinat dan gambar.

Berdasarkan pengujian yang telah dilakukan, *Obstac Map* bekerja secara baik dengan ketepatan data yang diperoleh mencapai 90%. *Obstac Map* mampu menampilkan titik lokasi jalan rusak dengan kelas data yang menyesuaikan tingkat kerusakan jalan. Sehingga *Obstac Map* berhasil melewati pengujian yang ditentukan, menunjukkan kinerja yang optimal dan memenuhi kebutuhan ekspektasi para pengguna. Namun terdapat kendala pada performa pemetaan lokasi karena terkadang mendeteksi lokasi bangunan terdekat.

Kata kunci : *Obstac Map*, Pemetaan Kerusakan Jalan, *Internet of Things*, *Mobile Application*, *UserExperience*.

I. PENDAHULUAN

Kerusakan jalan telah menjadi salah satu masalah yang sering kita hadapi dalam menjalankan kehidupan sehari-hari. Salah satu penyebab adanya jalan berlubang adalah penggunaan jalan yang berlebihan atau *overused*, membuat beban jalan melewati batas yang

direncanakan. Saat beban jalan melewati batasnya, jalan akan mengalami retakan dan pada akhirnya menimbulkan lubang pada jalan. Jalan yang berlubang yang memiliki dampak yang buruk untuk beberapa hal yang dapat menjadi *butterfly effect* kedepannya. Oleh karena itu, pengguna jalan harus memiliki kenyamanan yang tinggi untuk membuat proses berkendara menjadi aman, baik dari pengemudi maupun pengguna jalan lainnya.

Sebagai negara berkembang, Indonesia memiliki aktivitas masyarakat yang tinggi. Diambil dari Databoks, 37,02% masyarakat Indonesia bekerja sebagai buruh / karyawan / pegawai, pekerjaan ini mengharuskan pegawai untuk datang ke tempat kerja agar dapat melaksanakan pekerjaannya. Jumlah persentase pekerja yang tinggi akan berpengaruh terhadap beban penggunaan jalan. Penelitian [1] membahas bahwa desain geometrik jalan sangat berpengaruh terhadap keamanan dan kenyamanan bagi pengguna jalan yang merupakan prioritas utama serta syarat pokok pada perencanaan jalan. Dengan persentase masyarakat yang tinggi dalam menggunakan jalan, keadaan jalan menjadi prioritas utama demi kenyamanan dan keselamatan dalam penggunaan jalan. Jumlah pengguna jalan raya yang tinggi menyebabkan tingginya peluang terjadi kecelakaan lalu lintas. Dilansir dari Badan Pusat Statistik Nasional, jumlah kecelakaan lalu lintas dari tahun 2020 cenderung meningkat pada tahun 2021 kecelakaan mencapai 103.645 kecelakaan yang menyebabkan 25.266 jiwa terenggut. Banyak faktor yang berpengaruh terhadap kecelakaan lalu lintas, salah satunya adalah faktor lingkungan / kondisi jalan. Pada sejumlah 63 kasus kecelakaan lalu lintas di kota metro didominasi oleh infrastruktur jalan yang tidak memadai. Menurut (Radik M

F. & Widowati, 2021) dalam [2] perencanaan dan kondisi jalan sangat berpengaruh untuk menjadi penyebab kecelakaan lalu lintas. Perencanaan jalan yang tidak sesuai dengan hasil analisis fungsi jalan, volume dan komposisi lalu lintas, kecepatan rencana, topografi serta dimensi kendaraan dan lingkungan

maupun dana akan mengakibatkan turunnya nilai fungsi jalan dan nilai aman ruas jalan. Selain menjadi hal yang bahaya terhadap faktor keamanan, kondisi jalan yang buruk akan berdampak terhadap faktor ekonomi.

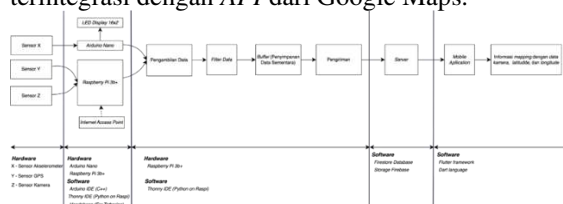
II. DASAR TEORI /MATERIAL DAN METODOLOGI/PERANCANGAN

A. Desain Rancangan Solusi

Sistem deteksi kerusakan jalan akan memanfaatkan sensor akselerometer dengan *Internet of Things*. Pada akselerometer, memiliki tiga sumbu yang dimanfaatkan untuk mengecek kerusakan jalan. Sumbu x dan y akan menunjukkan pergerakan kendaraan secara horizontal, dimana sumbu x akan menunjukkan arah kendaraan apakah bergerak ke depan atau ke belakang, sumbu y menunjukkan apakah kendaraan akan berbelok ke kanan dan ke kiri. Sedangkan sumbu z akan menunjukkan pergerakan kendaraan secara vertikal. Akselerometer merupakan sensor yang berfungsi untuk mengukur percepatan hingga mendeteksi getaran terhadap objek yang dipasangkannya [3]. Apabila kendaraan yang digunakan melewati jalan berlubang hingga mengakibatkan adanya guncangan, maka akselerometer akan aktif bekerja dipicu oleh guncangan tersebut. Guncangan memberikan perubahan terhadap ketiga sumbu pada akselerometer.

Akselerometer yang digunakan akan diberikan *filter* untuk membantu dalam mendeteksi jalan yang berlubang. *Filter* tersebut berfungsi untuk memberikan indikator terkait kondisi minimal guncangan hingga akselerometer dapat melakukan *mapping* pada kondisi guncangan tertentu. Untuk memastikan data *mapping* adalah data yang valid, maka akan ditambahkan GPS. Dalam pengiriman informasi dari akselerometer ke *server*, akan menggunakan perangkat *Raspberry Pi* yang dilengkapi *transmitter* modem untuk mengakses *internet*. Dengan adanya akses *internet*, maka dapat mengakses *server* untuk menyimpan datayangdiperoleh dari akselerometer dan GPS. Pengiriman data ke *server* akan bergantung pada sinyal yang dapat diakses oleh perangkat.

Apabila terjadi kehilangan sinyal, data akan disimpan terlebih dahulu. Lalu, setelah kembali mendapatkan sinyal, semua data akan dikirimkan ke aplikasi. Data informasi jalan berlubang yang telah didapatkan akan langsung dikirim ke *server* untuk disimpan dan diteruskan ke *mobile application*. Data akan divisualisasikan ke dalam *mobile application* yang terintegrasi dengan *API* dari Google Maps.

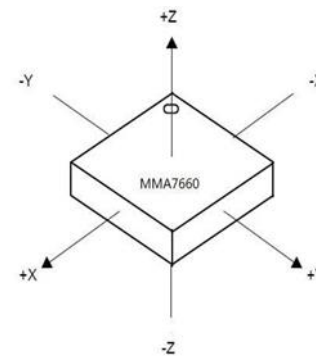


GAMBAR 1
Diagram Blok Sistem

Perangkat yang akan dikembangkan adalah perangkat deteksi guncangan untuk kerusakan jalan pada kendaraan dengan menggunakan sensor sebagai pendeteksi utama dan beberapa tambahan *software* sebagai pendukung dan perangkat lunak. Sensor akan dipasang pada kendaraan sebagai penerima respons dari deteksi yang terjadi pada sensor di kendaraan. Perangkat keras yang akan dipasang pada kendaraan berfungsi sebagai penerima dan proses pengiriman data dari data sensor yang diterima oleh kendaraan. Sedangkan perangkat lunak akan berfungsi sebagai penampil informasi terkait kerusakan yang terdeteksi pada kendaraan untuk *user* dengan bentuk data berupa *latitude* dan *longitude*. Selain *latitude* dan *longitude* pada deteksi kerusakan yang parah data yang akan ditampilkan adalah sebuah foto dari kerusakan jalan.

B. Detail Rancangan Akselerometer

Akselerometer adalah sensor yang dapat digunakan untuk mengukur percepatan dari suatu objek. Akselerometer dapat mengukur percepatan secara statis maupun dinamis dengan pengukuran gravitasi bumi dan percepatan objek bergerak. Pada jurnal ini, akselerometer digunakan sebagai sensor untuk melakukan deteksi dari suatu guncangan dengan menambahkan batasan pada beberapa kondisi untuk melakukan proses klasifikasi data. Pada akselerometer, terdapat tiga sumbu ortogonal yang mengacu pada sumbu X, Y, dan Z yang digunakan untuk mengukur percepatan dalam tiga dimensi. Selain itu, tiga sumbu ortogonal juga dapat mengacu terhadap arah dan mata angin:



GAMBAR 2
Sumbu Akselerometer

1. Sumbu X dan Y dapat diorientasikan untuk mengukur suatu percepatan pada bentuk arah horizontal. Pada sumbu X mengukur arah percepatan dari kiri ke kanan atau sebaliknya dengan arah mata angin timur dengan nilai positif X dan barat dengan nilai negatif X. Pada sumbu Y mengukur arah percepatan dari arah depan ke belakang atau sebaliknya dengan arah mata angin utara dengan nilai positif Y dan selatan dengan nilai negatif Y.
2. Sumbu Z dapat diorientasikan untuk mengukur percepatan dalam arah vertikal, dari bawah ke atas atau sebaliknya dengan arah atas bernilai positif Z

dan bawah bernilai negatif Z.

Sebelum kami menggunakan akselerometer untuk mendeteksi jalan berlubang, terlebih dahulu kami memastikan bahwasanya akselerometer yang akan digunakan berfungsi dengan baik. Tahap pertama yang perlu dilakukan adalah mencari tahu nilai awal dari setiap sumbu pada akselerometer. Hal ini untuk memastikan bahwasanya sensor meng-*outputkan* nilai yang sama setiap akan digunakan. Pengujian dilakukan selama 5 kali dan menghasilkan *output* pada tabel 1.

TABEL 1
Nilai Awal Akselerometer

Perobaan	Sumbu X (m/s ²)	Sumbu Y (m/s ²)	Sumbu Z (m/s ²)
1	-0.12	0.08	11.30
2	-0.16	0.08	11.34
3	-0.16	0.12	11.26
4	-0.16	0.12	11.34
5	-0.12	0.08	11.26

Tabel di atas menunjukkan hasil percobaan mencari nilai awal yang diberikan oleh sensor akselerometer. Angka di masing-masing sumbu menggunakan satuan m/s². Pengujian dilakukan selama 5 kali dengan rentang waktu antar pengujian yang berbeda beda. Rentang waktu percobaan satu ke dua memakan waktu 2 menit untuk *cooling down* sensor. Rentang waktu selanjutnya dinaikkan kelipatan 2. sehingga pada percobaan terakhir menghabiskan waktu 8 menit. Hasil dari percobaan tersebut menunjukkan bahwasanya dalam kondisi diam, sensor akselerometer meng-*outputkan* nilai yang cenderung stagnan dan sama setiap kali melakukan percobaan. Dengan demikian sensor akselerometer yang diuji sudah sesuai dan dapat digunakan untuk mendeteksi jalan berlubang.

Untuk menentukan klasifikasi kedalaman lubang tersebut, kami menggunakan klasifikasi yang sudah digunakan pada penelitian terdahulu [4]. Berikut tabel menunjukkan klasifikasi kerusakan jalan berdasarkan kedalaman dan panjang lubang.

TABEL 2
Klasifikasi Kerusakan Jalan

Kedalaman Maksimum	Rata-Rata Panjang Lubang		
	102 – 203 mm	203 – 457 mm	457 – 762 mm
12.7 - 25.4 mm	L	L	M
25.4 - 50.8 mm	L	M	H
> 50.8 mm	M	M	H
L: Belum perlu diperbaiki M: Penambalan parsial H: Penambalan di seluruh kedalaman			

Berdasarkan tabel tersebut, penulis membagi klasifikasi kondisi jalan menjadi tiga kelas, yaitu jalan yang belum perlu diperbaiki, penambalan parsial, serta penambalan di seluruh kedalaman dengan didukung aspek lainnya seperti kedalaman jalan berlubang hingga diameter besarnya kerusakan jalan. Tiga klasifikasi

kondisi jalan tersebut juga mengikuti standar yang dibuat oleh U.S. Army Corp of Engineer. Dimana mereka telah membuat penilaian kondisi kerusakan perkerasan yang diberi nama *Pavement Condition Index* (PCI) untuk kebutuhan perkerasan bandara, jalan, dan tempat parkir telah dipakai secara luas di Amerika. Berdasarkan klasifikasi tersebut, kami membuat tiga klasifikasi kondisi jalan mengikuti parameter yang sudah ada. Kondisi jalan baik mengikuti parameter jalan dengan kedalaman 12.7 - 25.4 mm, kondisi kerusakan ringan dengan kedalaman 25.4 - 50.8 mm, kondisi kerusakan besar dengan kedalaman lebih dari 50.8 mm.

Dalam tabel ditunjukkan bahwasanya terdapat 3 kondisi panjang lubang dengan rentang yang berbeda-beda. Kami menggunakan 3 rentang panjang jalan tersebut untuk dapat membantu menentukan klasifikasi kondisi jalan yang dilalui oleh alat. Dalam menentukan panjang lubang menggunakan akselerometer, kami mengaplikasikan rumus pythagoras dalam algoritma yang digunakan untuk mendapatkan percepatan dari sumbu x dan y. Untuk mengubah nilai percepatan ke dalam nilai jarak yang diinginkan kami menggunakan rumus berikut:

$$\Delta s = 0.5 x \sqrt{\Delta x^2 + \Delta y^2} x \Delta t^2 x \quad (1)$$

Rumus di atas menunjukkan bagaimana kami dapat menentukan panjang yang dilalui oleh sensor akselerometer menggunakan rumus perpindahan. X dan Y diambil dari perubahan sumbu yang terdeteksi oleh sensor akselerometer. Satuan dari variabel x dan y sendiri merupakan m/s², sehingga untuk mendapatkan satuan jarak kami mengalikan hasilnya dengan Δt^2 (waktu tempuh) untuk menghilangkan satuan second pada variabel tersebut. Kemudian hasil dari akar penjumlahan tersebut dikalikan dengan 1000. Fungsinya untuk mengubah hasil yang diberikan ke dalam satuan mm. Dengan demikian, kami dapat lebih mudah menggunakan delta S karena satuan yang dikonversi sudah sesuai dengan tabel pengklasifikasian kondisi jalan.

Setelah membuat tiga klasifikasi jalan yang dapat ditentukan berdasarkan kedalaman dan panjang lubang yang dilalui, kami menggunakan akselerometer untuk mendapatkan data sumbu X, Y, dan Z ketika melewati masing-masing klasifikasi jalan yang ada. Berikut merupakan tabel penjelasan dari masing-masing klasifikasi jalan disertai dengan perubahan sumbu X, Y, dan Z sertagambar kondisi jalan yang dilalui.

TABEL 3
Data Awal Kondisi Jalan

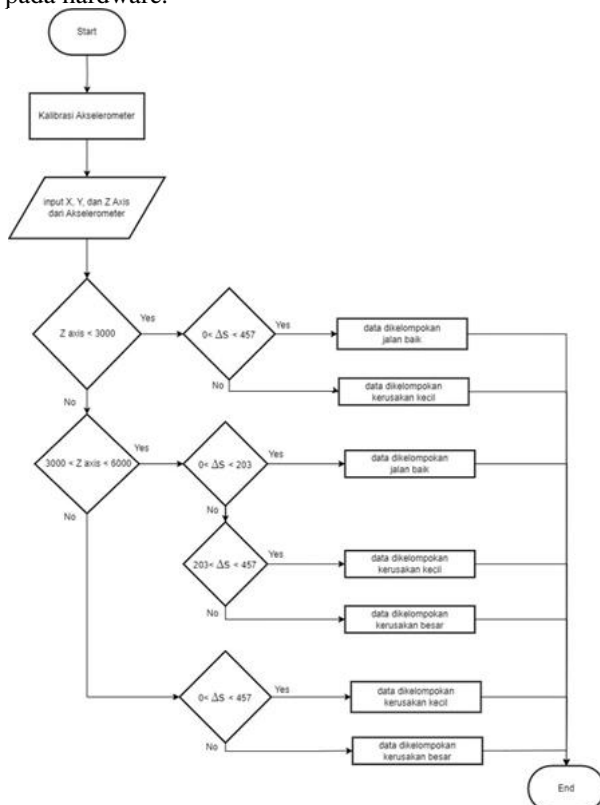
Gambar Jalan	Klasifikasi	X, Y, Z Axis (i)	X, Y, Z Axis (i+1)	S (Jarak Tempuh)	$\Delta(Z)$
--------------	-------------	------------------	--------------------	------------------	-------------

	L	356, 152, 17592	436, 120, 17644	100.5 mm	52
	M	-1108, 368, 13144	996, - 12, 9352	120.8 mm	3792
	H	-580, - 616, 13228	-4392, - 240, 19788	823.9 mm	6560

C. Detail Rancangan Arduino

Arduino merupakan *board* mikrokontroler untuk mengoperasikan perangkat dasar atau sensor dengan melatih sesuai dengan modul yang diberikan. Jenis Arduino yang digunakan, yaitu Arduino Nano. Arduino Nano memiliki 14 digital *input* dan *output*, dengan 8 *input* analog, *port* USB, dan *input* tegangan 6-20V [5]. Dalam melakukan proses *input source code* pada Arduino, diperlukan aplikasi Arduino IDE. Arduino yang digunakan membantu kami dalam mengintegrasikan komponen yang ada. Arduino sendiri memiliki peran sebagai komponen penyalur informasi berbentuk *sourcecode* menjadi perintah kepada perangkat yang terhubung.

Dalam pengaplikasiannya, arduino menghubungkan sensor akselerometer, *led*, hingga *LCD* untuk memberikan informasi output dari arduino. berikut merupakan flowchart dari sistem rancangan arduino pada hardware.



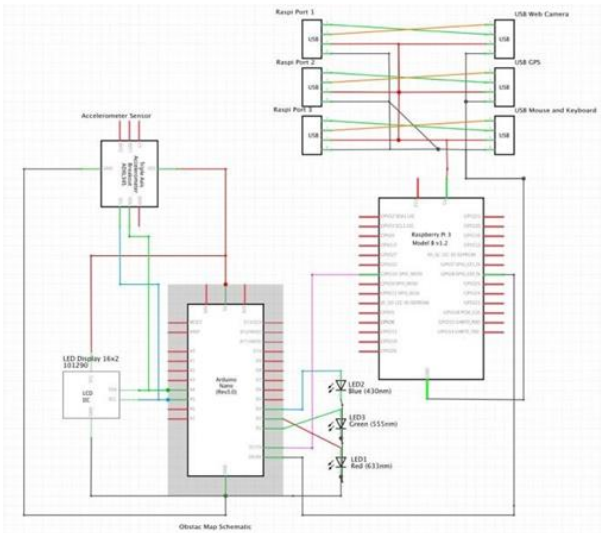
GAMBAR 3

Flowchart Klasifikasi Data Akselerometer

Flowchart pada gambar menunjukkan alur dari sistem rancangan arduino yang didukung oleh komponen-komponen lain. Awal dari *flowchart* menunjukkan bahwasanya arduino memerintahkan akselerometer untuk melakukan kalibrasi terlebih dahulu, sehingga didapat nilai awal dari kondisi stabil dan ideal pada akselerometer. Kemudian arduino memerintahkan sensor akselerometer untuk melakukan pengambilan data berupa sumbu x, y dan z yang terdeteksi. Proses pengambilan data ini dilakukan selama 1.5 detik. Setelah mendapatkan data berupa perubahan sumbu yang ada, arduino melakukan klasifikasi pada sumbu z terlebih dahulu. Pada arduino, kami memasang 2 nilai *threshold*. Nilai *threshold* ini membantu kami dalam melakukan klasifikasi kedalam lubang berdasarkan sumbu z yang diterima oleh sensor. Setelah data yang diterima oleh arduino dibagi berdasarkan nilai z-nya, tahap berikutnya adalah pengklasifikasian kondisi jalan yang dilalui berdasarkan nilai delta S yang didapat. Apabila data yang diperoleh sudah diklasifikasi, tahap berikutnya arduino akan memberikan perintah kepada 2 komponen lain. Komponen pertama adalah *LED*. *LED* yang terhubung pada arduino akan menyala menyesuaikan tipe klasifikasi data yang diterima pada arduino. Komponen kedua adalah *LCD*. *LCD* akan memberikan output berupa tipe kondisi jalan yang terdeteksi oleh sensor. berdasarkan alur dari flowchart tersebut, dapat dilihat bahwasanya arduino mengambil peran penting pada proses pengambilan hingga pengklasifikasian data.

D. Detail Rancangan Raspberry Pi

Raspberry Pi adalah mikrokomputer yang dikembangkan oleh Raspberry Pi Foundation untuk dapat mengoperasikan dan mengontrol perangkat serta berbasis *Internet of Things*. Raspberry Pi juga memiliki beberapa *port* untuk *input* dan *output* yang berfungsi untuk mengontrol sistem atau perangkat yang terintegrasi dengannya [6]. Dalam implementasi terhadap perangkat yang dikembangkan, kami menggunakan Raspberry Pi 3b+ untuk mengontrol antara Arduino, GPS, kamera, daya, serta *transmit* data ke *database*. Sistem wiring yang kelompok kami gunakan dapat dilihat pada gambar di bawah ini:



GAMBAR 4
Wiring Diagram Sistem

Sistem wiring yang kami gunakan akan menggunakan USB pada raspberry pi 3b+ untuk mendapatkan data dari kamera, GPS, dan juga arduino. Pengambilan data yang dilakukan oleh raspberry pi akan dilakukan setiap 10 menit. Setelah melakukan penerimaan data pada Raspberry selanjutnya akan dilakukan pengiriman data JSON yang telah disimpan ke *database* yang digunakan yaitu *firebase*. Untuk tahap pengiriman memanfaatkan dua fitur dari *firebase* yaitu *firestore* dan *storage*, tahapan akan dilakukan setiap 10 menit setelah data berhasil dibaca dan disimpan. Setelah melakukan pengiriman data selanjutnya akan dilakukan pengambilan data kembali. [8-10]

III. PEMBAHASAN

A. Akselerometer

Dari hasil penentuan kedalaman jalan rusak tersebut, kami mendapati *threshold* sebesar 3000 untuk jalan dengan kerusakan kecil, dan 6000 untuk klasifikasi jalan dengan kerusakan besar. Kedua kondisi *threshold* tersebut digunakan pada akselerometer supaya dapat mengklasifikasikan perubahan sumbu menjadi tiga kedalaman. Kedalaman jalan tersebut akan memberikan pengaruh terhadap sumbu z di akselerometer. Apabila sumbu z sudah menentukan klasifikasi kedalaman, tahap berikutnya adalah penentuan klasifikasi jalan berdasarkan panjang dari lubang yang dilalui. Dalam tahap ini, data yang digunakan adalah sumbu X dan Y dari akselerometer. Dengan menggunakan rumus *pythagoras*, maka diperoleh perpindahan dari sensor yang kemudian akan dikonversi menjadi satuan mm sesuai dengan ketentuan tabel pada .

Setelah mengaplikasikan nilai *threshold* pada akselerometer, kami melakukan pengujian pengambilan data untuk mengetahui presentase akurasi dari akselerometer. Pengujian ini dilakukan di sepanjang jalan radio. Pengujian dilakukan menggunakan kendaraan motor dengan kecepatan rata-rata 25 km/jam. Selama pengujian, kami mendapatkan 336 data

terdeteksi oleh akselerometer. Kami mengambil 3 data acak dari masing-masing klasifikasi jalan yang sudah ditentukan. Berikut merupakan tabel dari masing-masing sampel beserta dengan gambar kondisi jalan yang dilalui.

TABEL 4
Hasil Sampel Pengujian Data

Gambar	Z Axis (i)	Z Axis (i+1)	ΔZ	S (Jarak Tempuh)
Klasifikasi Jalan Baik				
	17436	17596	160	100.0 mm
	17640	17828	188	200.0 mm
	17828	17568	260	224.0003 mm
Klasifikasi Kerusakan Ringan				
	24120	14448	9672	439.3 mm
	16480	22900	6420	392.9 mm
	32767	16508	16259	200.0 mm
Klasifikasi Kerusakan Besar				
	5924	17492	11568	931.9 mm
	17812	6932	10880	504.9 mm
	6932	19228	12296	922.01 mm

Data pada tabel di atas menunjukkan klasifikasi kondisi jalan berdasarkan guncangan yang diterima oleh sensor akselerometer. Nilai *minus* pada sumbu akselerometer menunjukkan arah serta mata angin dari

guncangan yang terdeteksi oleh sensor. Selama pengambilan data sumbu dari sensor, data yang diterima akan disimpan di dalam JSON. Klasifikasi tersebut memberikan *output* akselerometer berupa judul kondisi klasifikasi serta sumbu X, Y, dan Z. Ketika akselerometer digunakan, sensor akan mengambil data guncangan yang terdeteksi setiap 1.5 detik. Hal ini dilakukan untuk memastikan tidak ada jalan rusak yang terlewat pada saat pendeteksian dilakukan.

B. GPS

Pada pengujian pertama yang dilakukan bertujuan untuk memastikan kemampuan dari GPS pada kondisi tertentu. Pengujian dilakukan di dalam gedung *Telkom University Landmark Tower (TULT)* dan di luar ruangan dilakukan di lokasi parkir TULT, tepatnya di tangga menuju lantai dua. Pada gambar 5, menunjukkan hasil dari pengujian yang dilakukan di dalam ruangan, GPS tidak dapat menunjukkan data *latitude* dan *longitude* sama sekali. Data yang ditunjukkan di dalam ruangan tidak memiliki nilai dan tidak dapat dibaca. Pada gambar 6, menunjukkan hasil pengambilan data di luar ruangan, hasil pengambilan ini menunjukkan bahwa GPS dapat memberikan data di barisan dengan kode \$GPGGA (GP dapat diartikan sebagai data yang diterima berasal dari GPS dan GGA mengartikan bahwa data di dalam baris memiliki data mengenai *latitude*, *longitude*, dan arah dari *latitude* dan *longitude*). GPGGA adalah kode yang digunakan oleh National Marine Electronics Association (NMEA). Hasil dari pengujian juga menunjukkan bahwa data yang didapat dari GPS berada di dalam bentuk DDM (Degree, Decimal Minutes).

GAMBAR 5
Pengujian In-Door

GAMBAR 6
Pengujian Out-Door

Pada pengujian kedua yang telah dilakukan bertujuan untuk memastikan keakuratan antara GPS dengan lokasi jalan rusak yang sebenarnya. Pengujian kedua dilakukan dengan menjalankan bersamaan perangkat *Obstac Map* di jalan Radio Palasari, Telkom University. Pengujian kedua dilakukan dengan mengambil sepuluh data dari pengumpulan data yang dilakukan. Dari hasil pengujian diambil lima data kerusakan besar, tiga data kerusakan ringan, dan dua data jalan baik. Dari hasil pengujian, dilakukan dengan membandingkan data GPS dan data sebenarnya menghasilkan perbedaan akurasi data. Data dari GPS dan data sebenarnya ditampilkan dalam tabel berikut:

TABEL 5

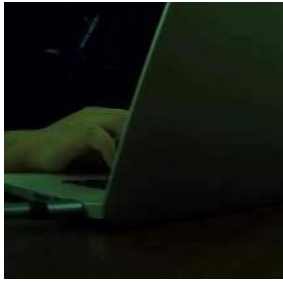
Data GPS Sebelum dan Sesudah Konversi

No.	Latitude dan Longitude dari GPS	Latitude dan Longitude Sebenarnya	Perbedaan Jarak
1	-6.9729801666666668, 107.62658583333332	-6.97293, 107.62658	5 m
2	-6.972972, 107.62638783333333	-6.97292, 107.62639	5 m
3	-6.9729351666666668, 107.62553766666666	-6.97288, 107.62553	5 m
4	-6.9729241666666667, 107.6252595	-6.97287, 107.62527	6 m
5	-6.9728995, 107.62479183333333	-6.97284, 107.62479	5 m

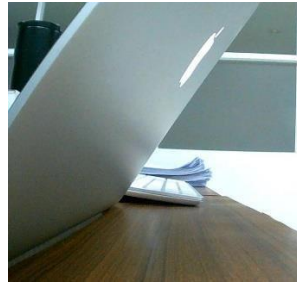
Pada tabel 13, hasil menunjukkan bahwa akurasi antara titik GPS dengan titik lokasi sebenarnya memiliki nilai toleransi antara dua hingga enam meter. Nilai tersebut dipengaruhi oleh banyaknya bangunan di sepanjang lokasi pengujian dengan waktu pengambilan data dilakukan pada siang hari dan kondisi lalu lintas yang cukup ramai. Data pada tabel 13 masih dalam bentuk DDM (*Degree, Decimal Minutes*) dimana hanya menampilkan nilai koordinat. Maka dari itu dilakukan konversi kedalam format DD (*Decimal Degree*) untuk menentukan arah mata angin, nilai derajat, serta menit pada lokasi pengujian. Hasil data yang telah dikonversi, akan dikirim ke *Firestore* melalui *Raspberry Pi* untuk dapat ditampilkan di *mobile application*.

C. Kamera

Pengujian yang dilakukan tanpa memberikan setting pada dan langsung melakukan *capture* pada akan menampilkan gambar yang kehijauan seperti diberikan sebuah *filter* tersendiri. Hal ini disebabkan oleh kurangnya cahaya yang didapatkan oleh kamera menyebabkan hasil pengambilan gambar yang dilakukan menjadi tidak maksimal. Hasil pengujian pertama dapat dilihat pada gambar 7. Untuk memperbaiki hal ini dilakukan pengujian kedua dengan menambahkan pengaturan pada *brightness* sebesar 0.8, *exposure* sebesar 0.5, dan *frame* sebesar 20. Selain itu pengujian kedua juga memberikan *delay* selama 3 detik untuk kamera sebelum mengambil gambar, hal ini membuat kamera dapat mengumpulkan cahaya terlebih dahulu sebelum mengambil gambar. Hasil pengujian kedua ini dapat dilihat pada gambar 8.



GAMBAR 7
Hasil Pengambilan Awal



GAMBAR 8
Hasil Pengambilan Setelah Perbaikan

Dalam proses pengambilan data menggunakan OpenCV akan menggunakan penyesuaian pada pengujian kedua yang menghasilkan gambar yang lebih baik dibandingkan dengan pengujian pertama. Kemampuan kamera untuk memberikan gambar yang jernih menjadi hal yang sangat penting, untuk dapat melakukan peningkatan pada kamera dapat digunakan jenis kamera yang lebih baik agar dapat melakukan pengambilan gambar yang lebih cepat dan hasil gambar akan menjadi lebih baik.

D. Software

Obstac Map merupakan produk inovasi dari tim *capstone design* kami yang ingin mengembangkan suatu sistem untuk mengurangi risiko kecelakaan serta memudahkan komunikasi dan informasi bagi para pengguna kendaraan. Dengan memberikan kemudahan akses informasi terkait kondisi jalan, Obstac Map dibangun menggunakan framework flutter dan menggunakan bahasa pemrograman swift, bahasa ini dipilih karena flutter adalah framework yang dibangun dan dikembangkan oleh Google dan bahasa yang dipakai bersifat native[7]. Obstac Map memiliki tujuan untuk membuat pengalaman para pengguna kendaraan lebih aman dan efisien. Berikut adalah bagian-bagian dari aplikasi Obstac Map:

1. Halaman Utama

Ketika memasuki aplikasi, akan diarahkan ke halaman utama yang terdiri dari dua pilihan, yaitu *Map* dan *About Us*. Anda dapat memilih salah satu dari kedua opsi tersebut. Ketika anda memilih *Map*, maka akan diarahkan ke jendela berikutnya untuk mengakses tampilan peta beserta informasinya. Sedangkan, ketika anda memilih *About Us*, maka akan diarahkan ke jendela berikutnya untuk diperkenalkan terkait *hardware* dan *software* dari Obstac Map.

2. Halaman Map

Ketika anda memilih untuk mengakses *Map*, maka akan diarahkan ke jendela berikutnya untuk melihat peta dengan beberapa indikator. Indikator jalan pada peta dapat berbeda tergantung tinggi atau rendahnya kerusakan jalan. Selain itu, anda dapat memperoleh *visual data* berupa gambar terkait kondisi jalan secara langsung.

3. Halaman About Us

Ketika anda memilih untuk mengakses *About Us*, maka akan diarahkan ke jendela berikutnya untuk melihat daftar *hardware* dan *software* dari Obstac Map. Anda dapat menekan dari tiap daftar *hardware* dan

software untuk mengetahui detail informasi terkait masing-masing perangkatnya. Selain itu, *About Us* juga akan memperkenalkan *profile* dari masing-masing pengembang.

IV. KESIMPULAN

Pengujian pada sistem yang kelompok kami telah kerjakan menjadi sebuah langkah yang penting untuk pengambilan keputusan pada pengerjaan selanjutnya. Pengujian pada *hardware* yang digunakan pada sistem ini seperti akselerometer, GPS, dan kamera menunjukkan bahwa sistem yang dibangun memberikan hasil yang dapat digunakan untuk mendeteksi kerusakan jalan. Akselerometer menunjukkan persentase keberhasilan sistem mencapai 93.33% dari total 45 sampel yang diambil. pada pengujian *software* sistem menunjukkan kemampuan untuk mengirimkan data ke *cloud database* sesuai dengan rencana yang telah dibuat pada awal pengerjaan solusi dari sistem.

Setelah melakukan hasil pengujian dapat dilihat beberapa kekurangan pada sistem yang dikerjakan, beberapa sistem yang dapat diperbaiki antara lain:

1. Peningkatan nilai toleransi akurasi pada GPS yang digunakan dengan menggunakan jenis GPS yang lebih baik sehingga data yang dihasilkan menjadi lebih akurat dengan toleransi dari GPS menjadi lebih kecil.
2. Pemakaian jenis kamera yang lebih baik untuk mempersingkat waktu *delay* yang ada pada saat kerusakan besar terjadi.
3. Pengiriman data yang lebih cepat dengan menggunakan internet yang lebih baik dengan penyusunan struktur data yang lebih baik untuk mempercepat pengiriman data.

Peneliti selanjutnya dapat menambahkan sistem yang dapat digunakan untuk melakukan pengecekan hasil deteksi secara *remote* sehingga peneliti tidak harus selalu melakukan pengecekan di lapangan. Dengan memperbaiki dan mengembangkan hal-hal yang telah disebutkan di atas dapat meningkatkan kinerja dari Sistem Deteksi Kerusakan Jalan menjadi lebih efisien dan akurat. Pengembangan yang dilakukan diharapkan tidak mengurangi tujuan awal dari dibuatnya solusi yang kelompok kami ajukan.

REFERENSI

- [1] M. Lubis, N. M. Rangkuti, and M. Ardan, "EVALUASI GEOMETRIK JALAN PADA TIKUNGAN LAOWOMARU," 2019, Accessed: Jun. 14, 2024. [Online]. Available: <https://jurnal.uisu.ac.id/index.php/semnastek/article/view/1350/1043>
- [2] F. Z. Aryatama and H. Widhiarto, "Analisis Penyebab Kecelakaan Lalu Lintas Di Jalan Empunala Kota Mojokerto," 2022. Accessed: Jun. 14, 2024. [Online]. Available: 10.33506/rb.v8i2.1805

- [3] I. A. Faisal, T. W. Purboyo, A. Siswo, and R. Ansori, "A Review of Accelerometer Sensor and Gyroscope Sensor in IMU Sensors on Motion Capture," 2020. Accessed: Jun. 14, 2024. [Online]. Available: 10.36478/jeasci.2020.826.829
- [4] W. Widodo, A. Rahmawati, and E. Adly, "MODEL ANTISIPATIF MENGATASI KERUSAKAN PERKERASAN JALAN DI KABUPATEN BANTUL YOGYAKARTA," 2017. Accessed: Jul. 04, 2024. [Online]. Available: <https://repository.umsida.ac.id/handle/123456789/14066>
- [5] I. A. Faisal, T. W. Purboyo, A. Siswo, and R. Ansori, "A Review of Accelerometer Sensor and Gyroscope Sensor in IMU Sensors on Motion Capture," 2020. Accessed: Jun. 14, 2024. [Online]. Available: 10.36478/jeasci.2020.826.829
- [6] H. Ghael, H. Dipak Ghael, L. Solanki, G. Sahu, and A. Professor, "A Review Paper on Raspberry Pi and its Applications," International Journal of Advances in Engineering and Management (IJAEM), vol. 2, p. 225, 2008, doi: 10.35629/5252-0212225227.
- [7] E. Windmill, "Flutter in Action," New York, 2020. [Online]. Available: www.allitebooks.com
- [8] A. Trimbakrao Gaikwad Bharati Vidyapeeth, P. Chougale, V. Yadav, A. Gaikwad, and B. Vidyapeeth, "FIREBASE-OVERVIEW AND USAGE," Article in Journal of Engineering and Technology Management, 2022, [Online]. Available: www.irjmets.com
- [9] M. M. Engel, "Android Based Thesis Mentoring System Using Google Firebase," ComTech: Computer, Mathematics and Engineering Applications, vol. 9, no. 2, p. 73, Dec. 2018, doi: 10.21512/comtech.v9i2.4992.
- [10] C. Khawas and P. Shah, "Application of Firebase in Android App Development-A Study," Int J Comput Appl, vol. 179, no. 46, pp. 49–53, Jun. 2018, doi: 10.5120/ijca2018917200.