

**AMARTO (DAMAGED ROAD DETECTOR):  
PURWARUPA SISTEM DETEKSI DAN ANALISATOR KERUSAKAN JALAN  
RAYA KOTA YOGYAKARTA BERBASIS CITRA DIGITAL DAN GPS**

**Tim Peneliti :**

Phisca Aditya Rosyady<sup>1</sup>

<sup>1</sup>Prodi Teknik Elektro, Fakultas Teknologi Industri, Universitas Ahmad Dahlan  
[phisca.aditya@te.uad.ac.id](mailto:phisca.aditya@te.uad.ac.id)

Faisal Fajri Rahani<sup>2</sup>

<sup>2</sup>Prodi Teknik Informatika, Fakultas Teknologi Industri, Universitas Ahmad Dahlan  
[faisal.fajri@tif.uad.ac.id](mailto:faisal.fajri@tif.uad.ac.id)

Ahmad Raditya Cahya Baswara<sup>1</sup>

<sup>1</sup>Prodi Teknik Elektro, Fakultas Teknologi Industri, Universitas Ahmad Dahlan  
[ahmad.baswara@te.uad.ac.id](mailto:ahmad.baswara@te.uad.ac.id)

**Abstrak**

Jalan raya merupakan salah satu aspek penunjang kehidupan bermasyarakat. Namun terkadang jalan raya tidak selalu dalam kondisi baik, terdapat beberapa jenis kerusakan jalan seperti lubang, retak, jalan tidak rata dan kerusakan-kerusakan lain yang dapat menyebabkan pengguna jalan terganggu. Untuk itu berdasarkan undang-undang nomor 22 tahun 2009, dimana pemerintah bertanggung jawab dalam melakukan perbaikan jalan, maka setiap adanya kerusakan yang terjadi pada suatu jalan raya, sudah menjadi kewajiban pemerintah untuk segera memperbaikinya. Permasalahan yang ada adalah proses deteksi dan pengecekan kerusakan jalan oleh Bina Marga Dinas Pekerjaan Umum dan Perumahan Rakyat Pemkot Yogyakarta masih berjalan secara manual.

Penelitian ini akan membuat solusi pemanfaatan teknologi dalam hal pendeteksi, klasifikasi, analisis dan pencatatan data kerusakan jalan raya, agar proses perbaikan jalan raya dapat dikerjakan dengan lebih cepat dan efisien. Metode penelitian ini adalah *Research and Development* yang digunakan untuk menghasilkan produk tertentu, dan menguji keefektifan produk atau sistem yang diteliti. Pendeteksian kerusakan lubang jalan raya ini digunakan sensor HC-SR04 dan Modul GPS GY-NEO6MV2, sedangkan untuk analisis beban kerusakan lubang jalan tersebut digunakan pengolahan citra dengan input kamera Webcam Logitech dan kedalaman menggunakan sensor HC-SR04. Pemrosesan utama menggunakan Raspberry Pi 4 dan sistem juga berbasis Internet of Things dengan ThinkSpeak.

Dalam penelitian ini telah terbuat purwarupa deteksi kerusakan lubang jalan berbasis citra digital dan GPS. Sistem ini mampu menganalisa dan menghitung luas jalan berlubang dengan tingkat error 0.358% - 4.19%, namun nilai parameter *lower*, *upper* HSV, *gaussian blur kernel*, *alpha*, *beta* pada fungsi masih beragam. Program yang berperan penting dalam proses pendeteksian jalan berlubang adalah *equalizer* histogram. Berdasarkan pengujian sensor jarak HC-SR04 dalam penelitian ini adalah *ping* sensor didapatkan bahwa untuk kedalaman lubang dengan pengukuran metode tidak bergerak maka didapatkan hasil *error* sebesar lebih kecil yaitu 1,66 % dan deteksi GPS dengan rata-rata selisih Latitude 0.0006144 dan Longitude 0.0000396. Implementasi IoT berbasis *Thinkspeak* sudah bisa dimanfaatkan sehingga monitoring bisa dipantau melalui web dan aplikasi android. Luaran penelitian ini berupa *paper* di Jurnal Nasional dan Hak Cipta.

Kata Kunci : *Lubang Jalan, Pengolahan Citra, GPS, Internet of Things*

## PENDAHULUAN

Jalan raya merupakan salah satu aspek penunjang kehidupan bermasyarakat. Keberadaannya sangat membantu masyarakat dalam transportasi maupun distribusi barang. Jalan raya dapat dibangun menggunakan konstruksi beton, aspal maupun *paving block*. Namun terkadang jalan raya tidak selalu dalam kondisi baik, terdapat beberapa jenis kerusakan jalan seperti lubang, retak, jalan tidak rata dan kerusakan-kerusakan lain yang dapat menyebabkan pengguna jalan terganggu. Untuk itu berdasarkan undang-undang nomor 22 tahun 2009, dimana pemerintah bertanggung jawab dalam melakukan perbaikan jalan, maka setiap adanya kerusakan yang terjadi pada suatu jalan raya, sudah menjadi kewajiban pemerintah untuk segera memperbaikinya. Apalagi menurut riset jalan menjadi salah satu penyebab utama kecelakaan. Faktor jalan ini seperti desain geometrik jalan dan layout yang tidak sesuai, kondisi permukaan jalan yang kurang memenuhi syarat (berlubang), fasilitas pejalan kaki tidak memadai, pencahayaan jalan.

Dalam sebuah jurnal disebutkan oleh Abadi Dwi Saputra dari Komite Nasional Keselamatan Transportasi (KNKT) bahwa prasarana sebagai faktor pendukung terselenggaranya transportasi jalan raya seharusnya dibangun dan dipelihara sehingga dapat memenuhi standar keamanan. Beberapa temuan kondisi prasarana transportasi yang memicu timbulnya kecelakaan adalah:

- a) Daerah rawan kecelakaan belum ditangani dengan baik oleh regulator;
- b) Kontruksi dan geometrik jalan yang kurang sempurna;
- c) Buruknya kondisi jembatan;
- d) Akses yang tidak dikontrol atau dikendalikan;
- e) Kurangnya rambu-rambu lalu lintas, alat penerangan jalan, marka jalan dan alat pemberi isyarat lalu lintas, serta alat pengendali dan pengaman bagi pengguna jalan

Proses perbaikan jalan raya dimulai dengan melakukan survei terlebih dahulu, guna memperoleh informasi yang diperlukan, seperti survei geometrik, struktur, kondisi jalan, pemanfaatan jalan, lalu lintas, dan sebagainya. Sistem deteksi kerusakan aspal jalan melalui video dapat membuat biaya lebih efektif, lebih cepat dan lebih aman dalam pelaksanaan pengamatan dan evaluasi kondisi jalan tersebut. Penelitian ini menggunakan rekaman video aspal jalan kemudian di ekstraksi menjadi frame-frame citra. Dengan memanfaatkan beberapa area hasil penjumlahan nilai *Fast Fourier Transform* pada citra digunakan sebagai fitur kerusakan untuk mengklasifikasi citra aspal jalan dengan kategori baik, sedang, rusak ringan, dan rusak berat. Selain proses survei yang masih manual, data yang dihasilkan juga hanya dicatat dalam sebuah tabel data survei. Proses pencatatan seperti ini seharusnya sudah dapat digantikan menggunakan perangkat-perangkat digital, sehingga akan membuat data menjadi tersinkronisasi, yang mana data tersebut dapat memudahkan proses-proses selanjutnya yang menggunakan data hasil survei sebagai acuan.

Sejauh ini terdapat beberapa penelitian yang telah mengembangkan sistem analisa kerusakan jalan yang memanfaatkan teknologi pengolahan citra, namun mayoritas masih menggunakan data citra 2 dimensi, sehingga data yang diperoleh hanya berupa luas dari lubang jalan . Untuk dapat memperoleh

data *volume* lubang jalan, maka diperlukan kamera yang dapat memberikan data citra *Red-Green-Blue Depth* (RGB-D).

Pada Januari 2020, peneliti juga berkoordinasi dengan Umi Akhsanti, S.T., M.T Kepala Bina Marga Dinas Pekerjaan Umum dan Perumahan Rakyat Pemkot Yogyakarta terkait permasalahan dinas terkait dalam hal kerusakan jalan hingga perbaikan jalan. Permasalahan yang ada adalah proses deteksi kerusakan jalan masih berjalan manual dengan petugas dan hanya menggunakan kamera biasa. Hal ini belum memungkinkan adanya system deteksi yang otomatis dan analisis kerusakan secara detail. Selain juga untuk proses perbaikan jalan dinas terkait belum bisa mengestimasi secara detail beban perbaikannya, seperti berapa lama perbaikan, berapa jumlah material, hingga perlu berapa orang yang memperbaiki.

Penelitian ini akan membuat solusi pemanfaatan teknologi dalam hal pendeteksi, klasifikasi, analisis dan pencatatan data kerusakan jalan raya, agar proses perbaikan jalan raya dapat dikerjakan dengan lebih cepat dan efisien serta detail dengan estimasi beban perbaikannya.

Dalam penelitian ini kami fokuskan pada pembuatan *prototype* untuk mendeteksi kerusakan jalan berbasis citra digital, menganalisisnya beban kerusakan tersebut dan menginformasikannya melalui *website* dan *android* dengan berbasis *Internet of Things*. Sehingga kami berharap akan membantu dinas terkait yaitu Dinas Pekerjaan Umum dan Perumahan Rakyat Pemkot Yogyakarta untuk melakukan pengecekan kerusakan lubang jalan di wilayahnya.

## **METODOLOGI**

Penelitian dilakukan selama enam bulan dari Maret – Agustus 2021 di Laboratorium Teknik Elektro, Universitas Ahmad Dahlan. Sedangkan untuk pengujiannya dilakukan di Jalan Pandeyan dan Jalan Warungboto, di wilayah Kota Yogyakarta. Dalam penelitian ini dilakukan suatu perancangan mulai dari identifikasi masalah yang ada pada objek penelitian hingga pemecahan masalah tersebut. Jenis penelitian yang digunakan dalam penelitian ini adalah R&D/*Research and Development*. Metode R&D adalah metode penelitian yang digunakan untuk menghasilkan produk tertentu, dan menguji keefektifan produk atau sistem yang diteliti.

Bahan-bahan yang dibutuhkan dalam pembuatan alat pendeteksi kerusakan jalan berbasis citra digital dan GPS adalah sebagai berikut :

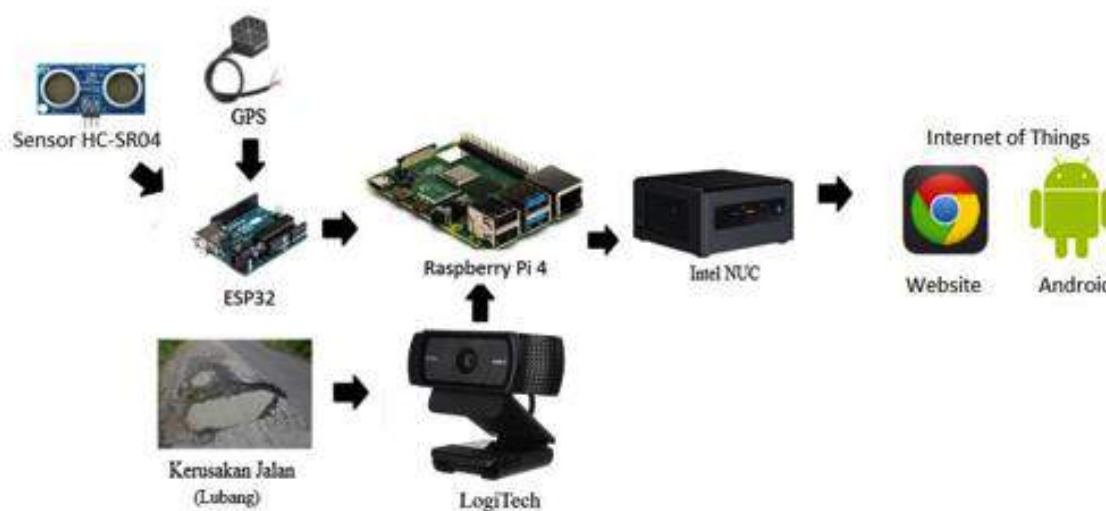
- a. Logitech C920 HD Pro Webcam digunakan sebagai sarana untuk mendapatkan informasi citra RGB-D dari suatu bagian jalan raya,
- b. Modul *Global Positioning System* (GPS) digunakan sebagai sarana untuk memperoleh informasi lokasi dimana sistem dijalankan,
- c. Komputer mini digunakan untuk mengolah dan menganalisa data citra dan data lokasi yang kemudian akan disimpan dalam server,
- d. Arduino UNO digunakan sebagai sarana untuk menghubungkan data yang diperoleh oleh GPS ke komputer mini,

- e. Baterai digunakan sebagai catu daya untuk menghidupkan perangkat-perangkat seperti komputer, Webcam, Arduino dan GPS
- f. *Acrylic* digunakan sebagai bahan mekanik untuk menyusun rangka yang dapat menopang semua komponen yang dibutuhkan

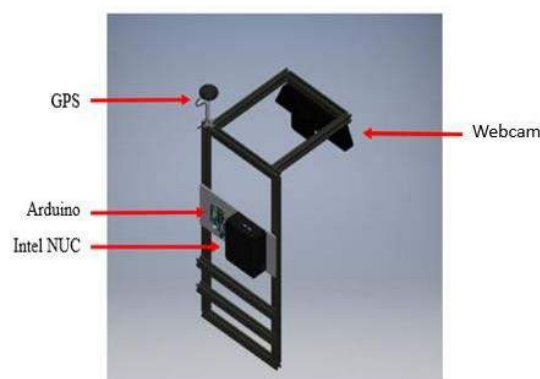
Penelitian ini memerlukan beberapa peralatan untuk, peralatann yang dibutuhkan antara lain :

- a. Laptop digunakan untuk merancang dan mensimulasikan program
- b. Perkakas elektronik digunakan untuk memasang komponen elektronik pada alat yang akan dibuat, perkakas tersebut diantaranya solder, tenol, atraktor, obeng dan lain – lain.
- c. Bor untuk melubangi PCB dan juga box alat
- d. Tang potong digunakan untuk memotong kabel

Dalam tahap ini penulis akan melakukan perancangan sistem yang akan dibuat. Dalam merancang sistem ini, peneliti merancang sistem dengan dua tahap yaitu perancangan perangkat keras dan perancangan perangkat lunak, untuk menghasilkan sistem yang sesuai perancangan sistem dilakukan dengan mengacu pada teori – teori dari penelitian sebelumnya. Berikut pada Gambar 4.1 merupakan blok diagram cara alat pendeteksi kerusakan jalan menggunakan kamera *Webcam* dan GPS. Sementara pada Gambar 4.2, memberikan gambaran secara menyeluruh mengenai bentuk dari purwarupa yang dibuat.



Gambar 4.1 Diagram sistem alat pendeteksi kerusakan jalan berbasis citra digital & GPS



Gambar 4.2 Rancangan Mekanik Sistem Detektor Kerusakan Lubang Jalan

Setelah alat berhasil dibuat, maka alat tersebut diuji coba terlebih dahulu. Pengujian sistem ini dilakukan untuk memastikan bahwa sistem yang telah dibuat sesuai dengan apa yang diharapkan oleh peneliti. Tahapan pengujian sistem dimulai dalam beberapa tahap sebagai berikut.

**a. Pengujian Perangkat Keras**

Pengujian perangkat keras sistem ini dimulai dengan melakukan pengecekan terhadap kondisi kamera *webcam*, *Arduino Uno*, komputer mini dan *GPS*, serta berbagai perangkat keras lainnya. Kemudian dilakukan juga pengujian terhadap kemasan dalam alat ini, seperti memaksimalkan sekrup, mur, serta baut yang digunakan pada alat ini. Memastikan pemasangan sistem ini di sepeda listriknya. Termasuk melakukan pengecekan pada *panel box* alat ini agar saat pengambilan data dapat sesuai dengan yang diinginkan.

**b. Pengujian Perangkat Lunak**

Pada pengujian tahap ini merupakan pengujian yang berkaitan dengan program yang sudah dibuat apakah berjalan dengan baik dan sesuai dengan apa yang diharapkan, pada pengujian perangkat lunak ini terbagi menjadi beberapa tahap sebagai berikut :

**a) Pengujian Program**

Pada tahap ini program yang sudah dituliskan didalam *software* *Arduino IDE* dan *Python* akan dilakukan pengujian apakah program yang sudah dibuat ini sudah benar dan tidak terdapat *error* yang dapat mengganggu kinerja dari sistem

**b) Pengujian Modul GPS**

Pada tahap ini modul *GPS* yang sudah dikoneksikan dengan *Arduino Uno* akan dilakukan pengujian apakah dapat menerima data sinyal *GPS* dalam keadaan normal dan apakah data yang diterima telah sesuai dengan lokasi yang sesungguhnya

**c) Pengujian Kamera Logitech**

Pada tahap ini, kamera *Webcam* diuji apakah dapat membaca data warna dan kedalaman sesuai dengan kondisi yang sesungguhnya

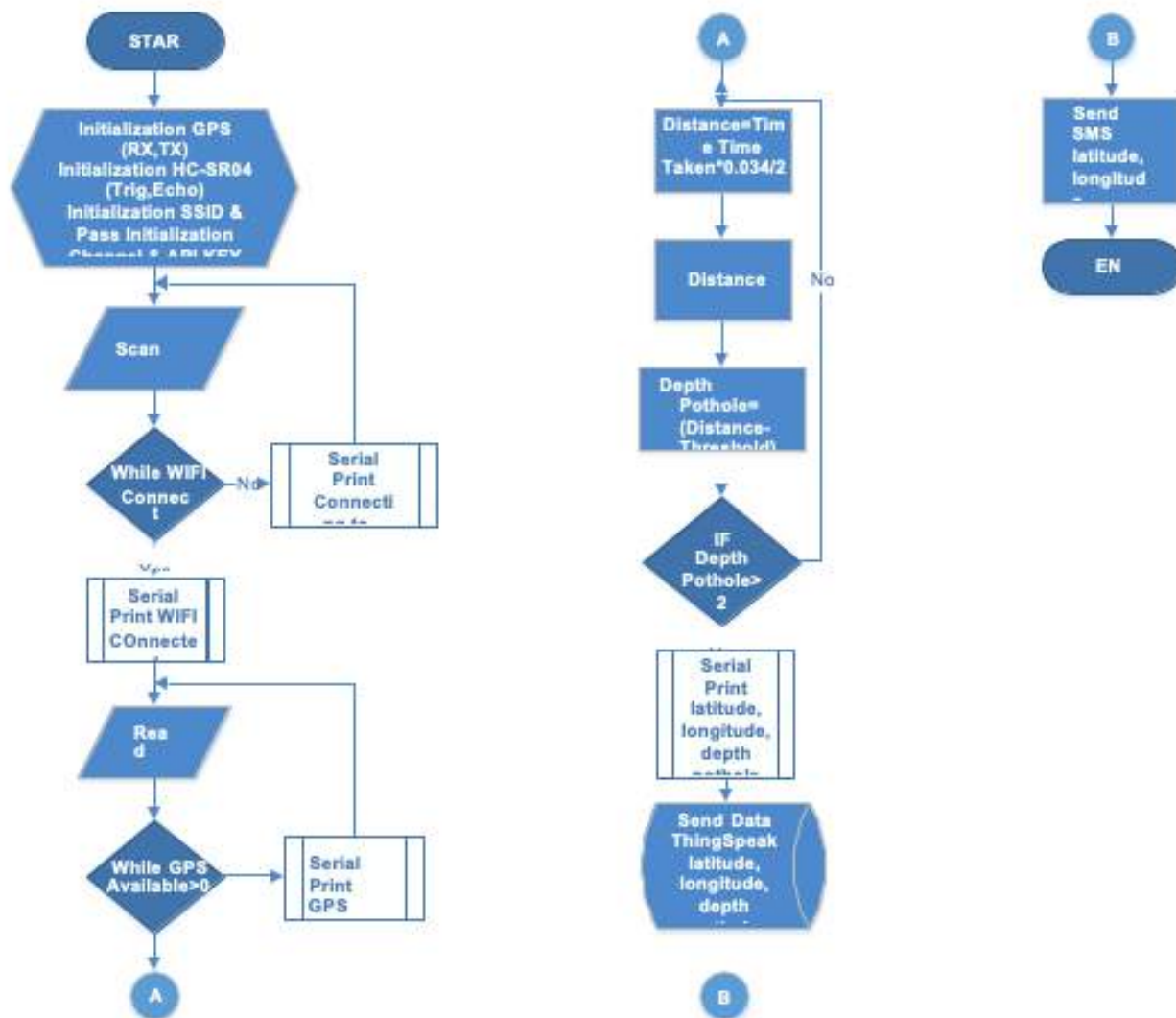
## HASIL DAN PEMBAHASAN

### 1. Sistem Deteksi Kedalaman dan Posisi Koordinat

Pada sistem ini digunakan *ESPDUINO-32* sebagai pemroses utama, sedangkan sensor ultrasonik *HC-SR04* berfungsi untuk mengukur ke dalam lubang. Pengukuran kedalaman lubang dengan sensor ultrasonik *HC-SR04* memanfaatkan prinsip pemantulan gelombang untuk mendapatkan jarak objek dengan frekuensi yang sudah ditentukan pada osilator dalam rangkaian modul sensor ultrasonik *HC-SR04*. Sedangkan modul *GPS Ublox Neo-6MV2* berfungsi sebagai penerima (*receiver*) sinyal *GPS (Global Positioning System Receiver)* dalam menangkap dan memproses sinyal dari satelit, sehingga dapat mendeteksi lokasi suatu objek. Data dari sensor akan dikirim ke internet dengan menggunakan *platform ThingSpeak*, karena *ESPDUINO-32* sudah didukung dengan *chip wifi* sehingga sistem sudah mendukung untuk terkoneksi dengan internet tanpa memerlukan modul tambahan. Dalam perancangan ini penulis memanfaatkan *platform ThingSpeak* untuk menyimpan data dan menampilkan dalam bentuk

grafik, untuk notifikasi ketika ada lubang yang terdeteksi sistem akan mengirimkan pesan dalam bentuk *tweet*. Isi *tweet* berupa lokasi dalam bentuk data lintang (Latitude) dan bujur (Longitude) serta data kedalaman lubang yang terdeteksi.

Diagram alir dari sistem deteksi kedalaman lubang dengan sensor ultrasonik HC-SR04 dan modul GPS Ublox Neo-6MV2 ini dapat dilihat pada gambar 3 berikut.



Gambar 4.3 Diagram Alir Sistem Pengukuran Kedalaman Lubang Jalan Raya

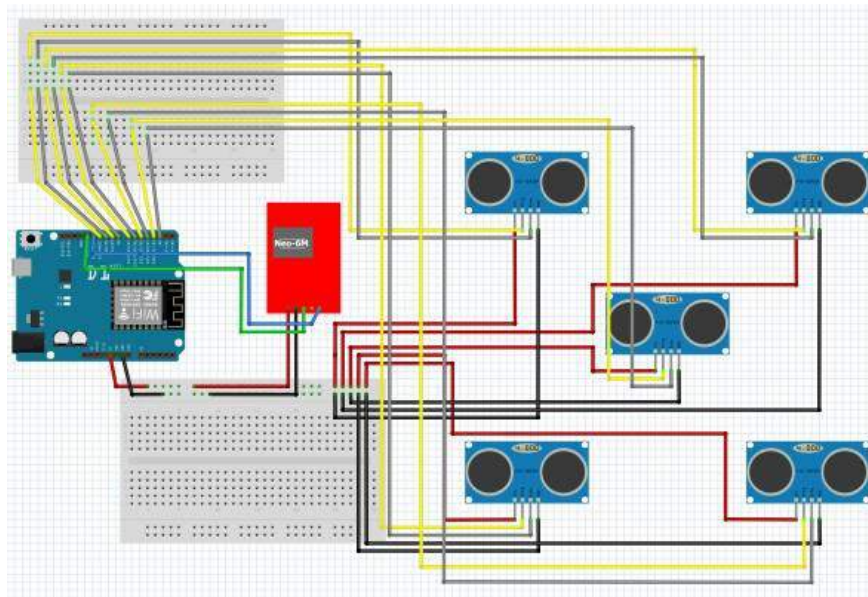
Berdasarkan diagram alir pada gambar 3 bahwasannya sistem ini menggunakan wifi untuk terkoneksi ke jaringan internet. Jika sistem tidak terkoneksi ke jaringan internet maka proses berikutnya tidak akan berjalan, sehingga ESPDUINO-32 akan melakukan *scanning wifi* sampai *wifi* terdeteksi dan terhubung. Pada saat ESPDUINO-32 sudah terkoneksi dengan *wifi* maka GPS Ublox Neo-6MV2 akan dibaca oleh mikrokontroler dalam hal ini adalah ESPDUINO32, ketika GPS sudah menerima sinyal yang ditandai dengan indikator led merah yang berkedip maka artinya GPS sudah bisa diakses oleh ESPDUINO-32. Pada saat *wifi* terhubung dan GPS aktif maka sistem akan memerintahkan sensor ultrasonik HC-SR04 untuk membaca jarak. Prinsipnya adalah sensor ultrasonik HC-SR04 akan mengukur jarak ketika pulsa *trigger* diberikan pada sensor, *transmitter* akan mulai memancarkan gelombang ultrasonik, pada saat yang sama sensor akan menghasilkan *output transistor-transistor logic* (TTL) transisi naik menandakan sensor mulai menghitung waktu pengukuran, setelah *receiver* menerima pantulan yang dihasilkan oleh suatu objek maka pengukuran waktu akan dihentikan dengan

menghasilkan *output transistor-transistor logic* (TTL) transisi turun. Proses pemantulan inilah yang dimanfaatkan untuk mendapatkan jarak antara sensor dengan objek tersebut. Persamaan untuk mendapatkan jarak adalah  $\text{Jarak (cm)} = \text{waktu tempuh}/29,4 * 2$ .

- a. Waktu tempuh adalah waktu yang dibutuhkan gelombang pada saat dipancarkan sampai gelombang tersebut diterima kembali.
- b. Sementara 0,034 adalah kecepatan gelombang Ultrasonik, di mana kecepatan Ultrasonik adalah 340 m/s sehingga untuk 1 cm memerlukan 0,0000294 s (29,4  $\mu$ s), karena proses tersebut terdapat *transmit* – *receive* atau memancarkan dan menerima maka waktu yang dibutuhkan adalah 2 kali, sehingga untuk memperoleh jarak 1 cm diperlukan waktu  $29,4 \mu\text{s} * 2 = 58,8 \mu\text{s}$

Selanjutnya data jarak yang diperoleh akan diproses lanjut dengan menggunakan algoritma kalman *filter*. Fungsi dari algoritma ini adalah agar data jarak yang diperoleh sensor bagus, sebab sensor ini diletakan pada sepeda motor untuk mengukur jarak, sehingga saat sepeda motor bergerak maka akan banyak guncangan yang dapat mengakibatkan data dari sensor kurang bagus nilainya. Setelah nilai jarak pada sensor di *filter* maka didapatkan nilai *Distance Kalman Filter*. Kemudian untuk memperoleh kedalaman lubang sensor dapat dicari dengan rumus nilai *Distance Kalman Filter*. Nilai *Threshold*. Nilai *threshold* pada penelitian ini adalah 13 cm, nilai ini didapatkan dari nilai jarak antara sensor dengan *ground* (aspal/tanah). Pada sitem ini akan mengirimkan data Latitude dan Longitude dan kedalaman lubang (*depth pothole*) ke *ThingSpeak* apabila terdeteksi kedalaman lubang lebih dari 2 cm ketika data tersebut tidak terpenuhi atau kurang maka sistem tidak akan mengirim data ke *ThingSpeak*. Selain mengirimkan data pada *Thingspeak* sistem juga akan mengirimkan data dalam hal ini mengirimkan data Latitude dan Longitude dan kedalaman lubang (*depth pothole*) melalui sms ke nomor yang sudah di set dalam program dan untuk notifikasi tidak hanya melalui sms, akan tetapi juga melalui *tweet* di aplikasi *twitter* dengan akun yang sudah dibuat khusus untuk penelitian ini.

Pada Gambar 4.4 sistem pengkabelan (*wiring*) modul GPS Neo-7M dihubungkan dengan ESP32 pada modul tersebut ada 4 pin (vcc, gnd, rx tx) pada vcc modul GPS dihubungkan pin vin ESP32, sedangkan untuk gnd dihubungkan ke pin ground dan untuk pin rx dihubungkan pada pin D16 kemudian pada pin tx dihubungkan pada pin D17. Modul sensor Ping terdapat 3 pin (gnd, 5v dan sig) penyambungan pin ini adalah untuk pin gnd dihubungkan ke pin *ground* pada ESP32, 5v dihubungkan pada pin vin, sedangkan untuk pin sig dapat disambungkan dengan pin D4 pada ESP32. Sistem ini membutuhkan sumber tegangan 5 volt yang dihubungkan menggunakan kabel micro USB. Pengujian sensor jarak dilakukan dengan menggunakan skenario 5 sensor dengan pengkabelan seperti pada Gambar 4.4.



Gambar 4.4 Wiring Sensor Ultrasonik HC-SR04 dan Modul GPS Neo-7M

Berikut adalah hasil pengujian dengan variasi jarak mulai dari 10 cm hingga 50 cm dengan pengambilan data masing-masing sebanyak 5 kali.

Tabel 4.1 Data Pengujian Sensor HC-SR04

Penggaris	Sensor 1	Sensor 2	Sensor 3	Sensor 4	Sensor 5	Rerata	Selisih (Cm)	Selisih (%)
10	10,1	9,54	10,06	9,9	10,76	10,072	0,072	0,72
10	10,98	9,54	10,16	9,32	10,83	10,166	0,166	1,66
10	10,39	9,54	10,06	9,32	10,74	10,01	0,01	0,10
20	21,57	19,06	20,86	19,04	21	20,306	0,306	1,53
20	20,11	19,01	20,81	19,45	20,51	19,978	0,022	0,11
20	21,57	19,06	20,86	19,04	21	20,306	0,306	1,53
30	30,75	28,87	30,45	28,82	30,21	29,82	0,18	0,60
30	30,34	28,87	30,46	28,82	30,19	29,736	0,264	0,88
30	30,29	28,87	30,4	28,75	30,62	29,786	0,214	0,71
40	41,6	38,81	41	38,42	40,31	40,028	0,028	0,07
40	41,65	38,86	41,06	38,42	40,78	40,154	0,154	0,39
40	41,19	39,27	41	38,42	40,73	40,122	0,122	0,31
50	51,61	49,87	50,9	48,04	50,88	50,26	0,26	0,52
50	51,54	48,5	50,92	48,09	50,42	49,894	0,106	0,21
50	51,61	48,5	50,49	48,47	50,47	49,908	0,092	0,18

Tabel 4.1 menjelaskan tentang pengujian awal deteksi jarak kedalaman menggunakan sensor HC-SR04. Dalam pengujian ini peneliti melakukan variasi jarak dari 10 cm hingga 50 cm dengan masing-masing pengujian sebanyak 3 kali. Alat ukur pembandingnya digunakan penggaris sehingga didapatkan selisih yang menunjukkan tingkat presisi dari sensor jarak HC-SR04 ini. Dari pengujian awal ini didapatkan selisih terkecil dari hasil pengukuran dengan referensi 0.07 % dan selisih terbesar dalam pengujian ini mencapai 1.66 %.

Sedangkan untuk pengujian Modul GPS dilakukan dengan pengambilan data pada posisi Modul GPS diam di suatu lokasi. Pengamatan dilakukan dengan posisi diam untuk menentukan Latitude suatu tempat berdasarkan koordinat yang diperoleh dengan menggunakan modul GPS. Berdasarkan data yang diperoleh, data GPS dibandingkan dengan data Google Maps sebagai referensi, sehingga dapat diketahui persentase kesalahan dari GPS yang diperlihatkan pada Tabel 4.2.



Tabel 4.2 Data GPS Latitude

No.	Latitude Modul GPS (°)	Latitude Google Maps (°)	Error (%)
1.	-7.770980	-7.7709685	0,000147987
2.	-7.762779	-7.7627124	0,000857948
3.	-7.762895	-7.7628832	0,000152005
4.	-7.763368	-7.7634264	0,000752245
5.	-7.763626	-7.7635872	0,000499769
6.	-7.763895	-7.7640313	0,001755531
7.	-7.764050	-7.7640784	0,000365787
8.	-7.767836	-7.7678577	0,000279356
9.	-7.767972	-7.7680329	0,000783982
10.	-7.768146	-7.7681887	0,000549678
Jumlah Rata-Rata Error = $\frac{\Sigma}{10}$			0,000614429

Berdasarkan pengamatan bahwasanya modul *GPS* untuk data latitude didapatkan persentase *error* atau kesalahan rata-rata yang sangat kecil yakni 0,000614429 %. Pengambilan data ini dilakukan dalam posisi diam dan meskipun pengambilan secara diam.

Pengamatan dilakukan dengan posisi diam untuk menentukan longitude suatu tempat berdasarkan koordinat yang diperoleh dengan menggunakan modul *GPS*. Berdasarkan data yang diperoleh, data *GPS* dibandingkan dengan data *Google Maps* sebagai referensi, sehingga dapat diketahui persentase kesalahan dari *GPS* yang di perlihatkan pada Tabel 4.3.

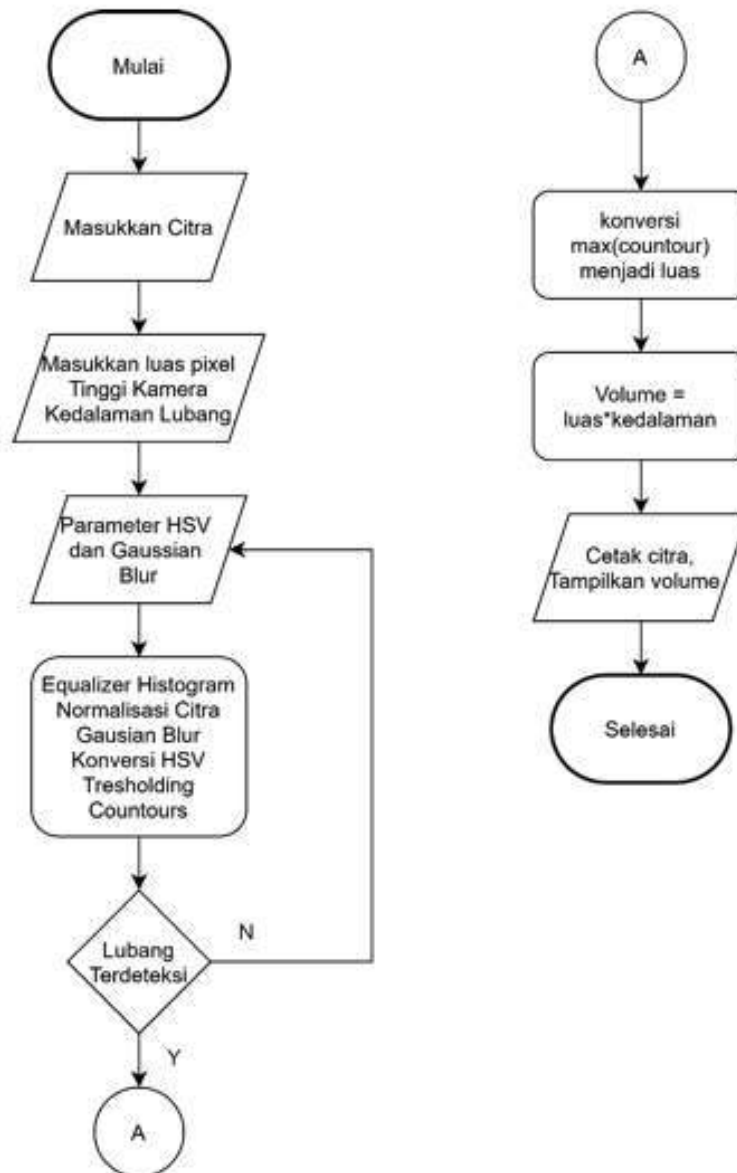
Tabel 4.3 Data GPS Longitude

No.	Longitude Modul GPS (°)	Latitude Google Maps (°)	Error (%)
1.	110.386638	110.3866263	0,0000105991
2.	110.381272	110.3812472	0,0000224676
3.	110.381775	110.381812	0,0000335200
4.	110.382396	110.3823827	0,0000120490
5.	110.382260	110.3822829	0,0000207461
6.	110.382106	110.3821904	0,0000764616
7.	110.382169	110.3821142	0,0000496457
8.	110.389445	110.3894042	0,0000369601
9.	110.389419	110.3894635	0,0000403118
10.	110.389386	110.3892826	0,0000936685
Jumlah Rata-Rata Error = $\frac{\Sigma}{10}$			0,0000396430

Berdasarkan data pengujian di atas bahwasanya modul *GPS* untuk data longitude didapatkan *persentase error* atau kesalahan rata-rata yang sangat kecil yakni 0,0000396430 %. Pengambilan data ini dilakukan dalam posisi diam dan meskipun pengambilan secara diam.

## 2. Sistem Analisator Luas Lubang Jalan Berbasis Citra Digital

Diagram alir untuk program citra digital pada Sistem Analisator Kerusakan Jalan Raya Berbasis Citra Digital yang digunakan pada penelitian ini tampak pada Gambar 4.5.



Gambar 4.5 Diagram Alir Sistem Pengukuran Kedalaman Lubang Jalan Raya

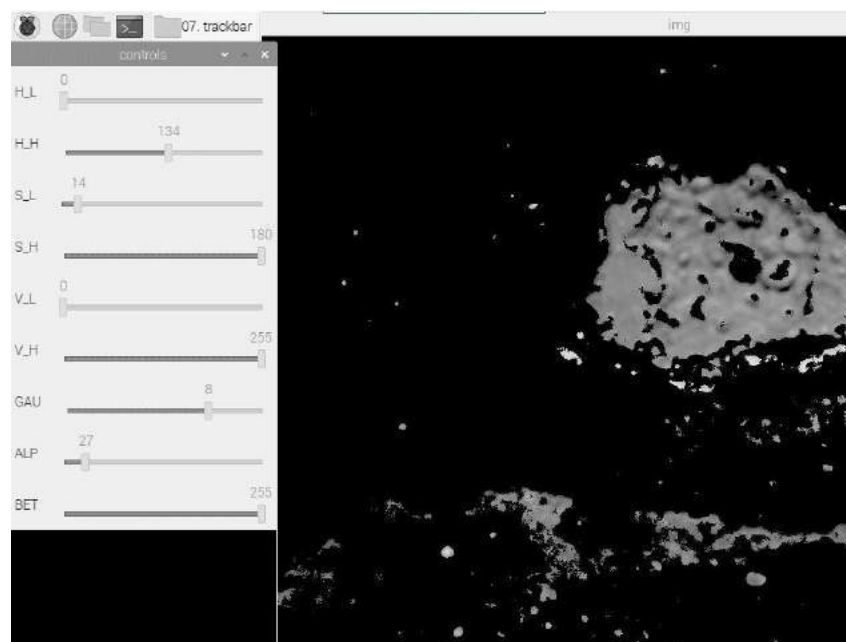
Pada diagram alir sistem diatas, menjelaskan proses pendeteksian objek dan pengambilan *contour* serta mendapatkan nilai luas dan diameter, pada kondisi awal diperlukan adanya proses pengambilan gambar dengan menggunakan kamera *smartphone*, kemudian gambar tersebut akan diproses pada *pycharm* dengan berbasis pengolahan citra digital. Pendeteksian objek berdasarkan *range* nilai minimum dan maksimum HSV, jika nilai minimum dan maksimum HSV pada suatu gambar memenuhi maka akan terjadi pendeteksian objek kemudian hasil deteksi tersebut *dicontour* dan mendapatkan nilai luas dan diameter, jika tidak maka akan melakukan proses pengambilan gambar kembali secara terus menerus sampai memenuhi nilai HSV.

Pengujian program merupakan kegiatan yang dimaksudkan untuk menguji program dengan menggunakan perangkat lunak *Thony Python IDE*. Program yang ditulis dengan perangkat lunak *Thony Python IDE* disebut sebagai *sketch*. *Sketch* ditulis dalam suatu *editor* teks dan disimpan dalam file dengan ekstensi *.py*. Format tersebut muncul ketika program yang dibuat disimpan pada suatu folder. Program pada *Thony Python IDE* digunakan untuk seluruh kegiatan pemrograman alat pada penelitian ini.

Program menjalankan Kamera Pi yang terhubung pada Raspberry Pi 3B dengan delay 5 detik. *Verify* dan *Running* program menunjukkan tidak terdapat *error* dan kamera berjalan sesuai perintah program. Adapun parameter yang dibuatkan trackbar para program di antaranya adalah:

1. *Lower hue*
2. *Lower saturation*
3. *Lower value*
4. *Upper hue*
5. *Upper saturation*
6. *Upper value*
7. Dimensi *kernel gaussian blur*
8. Nilai *alpha* normalisasi
9. Nilai *beta* normalisasi

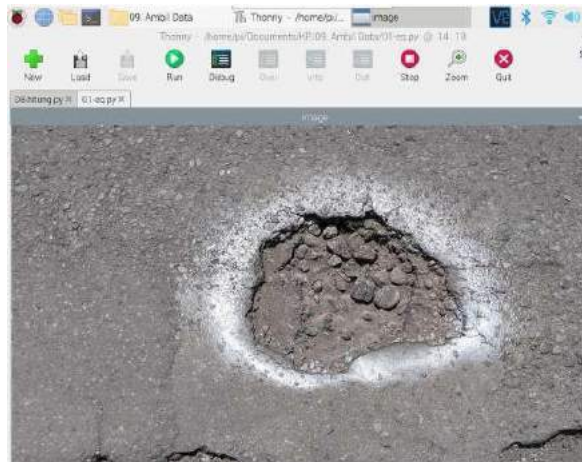
*Trackbar* untuk mencari nilai parameter deteksi jalan berlubang disajikan pada Gambar 4.6.



Gambar 4.6 *Trackbar* parameter deteksi jalan berlubang

- **Pengujian mengakses gambar citra dari file**

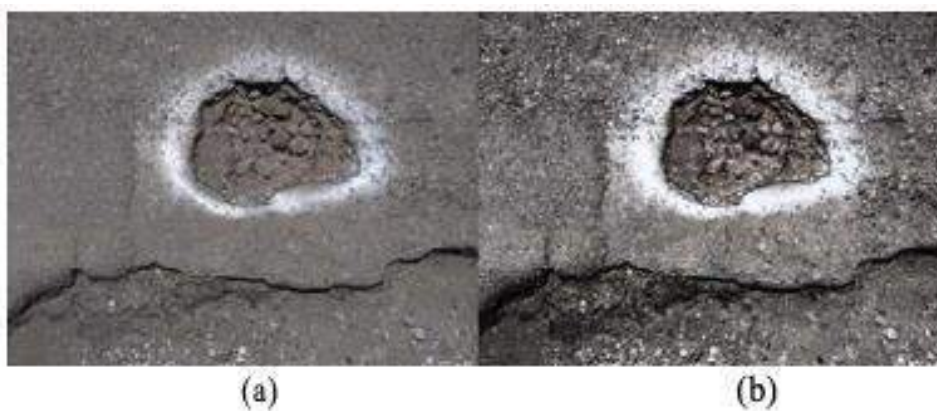
Pengujian mengakses gambar citra melalui *file* yang sudah disediakan berjalan dengan baik, hasil pengujian akses citra melalui *file* disajikan pada Gambar 4.7.



Gambar 4.7 Pengujian akses citra melalui file

- **Pengujian program *equalizer histogram***

Pengujian alat dengan mengubah citra asli dengan menggunakan metode *equalize histogram* yang dimaksudkan untuk menyeimbangkan histogram persebaran warnanya. sehingga lebih mudah untuk diproses. Hasil dari eksekusi program *equalize histogram* pada citra asli disajikan pada Gambar 4.8



Gambar 4.8 Gambar Analisis Equalize Histogram

- **Pengujian program normalisasi**

Pengujian alat dengan mengubah citra hasil dari *equalize histogram* dengan menggunakan metode *normalize minmax* yang memiliki dua parameter yaitu *alpha* dan *beta*. di mana fungsi dari *alpha* dimaksudkan untuk meningkatkan kecerahan dan *beta* untuk menurunkan kecerahan. nilai *default alpha* adalah 0, dan *beta* adalah 255. Pada proses *thresholding* metode *normalize* bertujuan untuk mengatur besar rongga pada hasil *thresholding*. hasil eksekusi dari metode *normalize* disajikan pada Gambar 4.9.



Gambar 4.9 (a) hasil equalize histogram, (b) hasil normalisasi

- **Pengujian program gaussian blur**

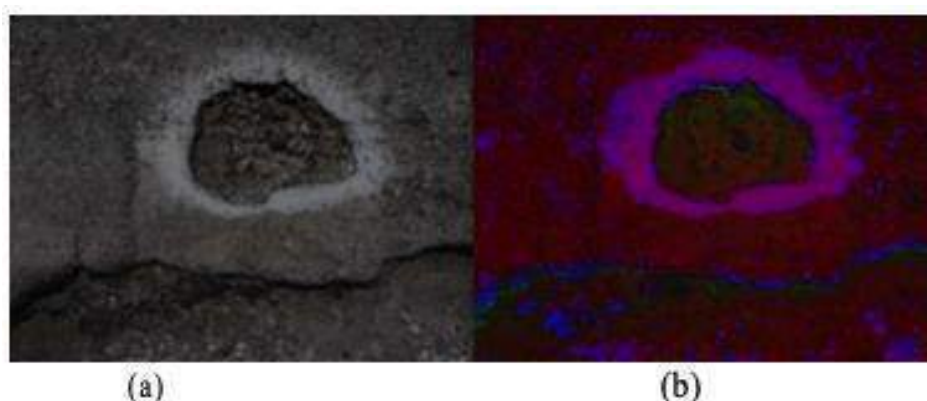
Pengujian alat pada bagian filter *gaussian blur* dimaksudkan untuk mengurangi ketajaman gambar. Bila ada perbedaan warna yang tinggi pada jarak yang kecil akan diburamkan sehingga memudahkan proses *thresholding*. hasil eksekusi program *filter gaussian blur* disajikan pada Gambar 4.10.



Gambar 4.10 (a) hasil normalisasi, (b) hasil filter gaussian blur

- **Pengujian program konversi HSV**

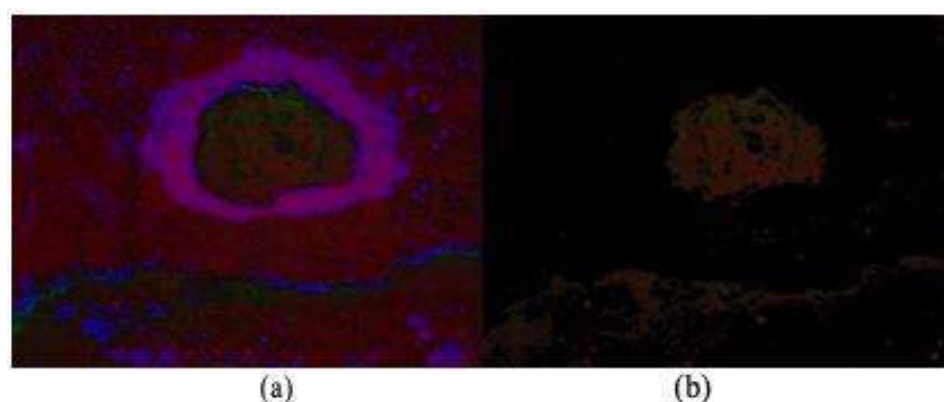
Pengujian alat pada bagian ini yaitu mengubah ruang warna RGB menjadi ruang warna HSV. Ruang warna HSV dipilih karena dapat mengubah warna yang awalnya gabungan dari beberapa warna primer menjadi warna spektrum pada cahaya, sehingga dapat meningkatkan akurasi pada metode *thresholding*. hasil pengujian konversi warna HSV disajikan pada Gambar 4.11.



Gambar 4.11 (a) Hasil filter gaussian, (b) Hasil konversi HSV

- **Pengujian program thresholding**

Pengujian alat pada program *thresholding* dimaksudkan untuk memisahkan objek lubang jalan. Pada pengujian *thresholding*, lubang jalan tidak benar-benar terpisah. melainkan ada objek lain yang memiliki ciri yang sama dengan lubang jalan. hasil pengujian program *thresholding* disajikan pada Gambar 4.12.



Gambar 4.12 (a) Hasil filter gaussian, (b) Hasil thresholding

Langkah berikutnya adalah mencari nilai setiap parameter dengan menggunakan program mencari nilai HSV. Parameter yang dicari di antaranya adalah:

- a. *Normalize*
- b. *Lower inRange HSV*
- c. *Upper InRange HSV*
- d. *Kernel filter gaussian*
- e. *Threshold biner.*

Hasil pengujian parameter dari 9 sampel yang sudah ditentukan disajikan pada Tabel 4.4.

Tabel 4.4 Nilai parameter HSV dan Gaussian

Pothole	Lower inRange HSV			Upper inRange HSV			Gaussian
	H	S	V	H	S	V	
1	0	62	0	179	225	225	5
2	0	0	69	37	255	99	10
3	80	0	0	179	255	42	10
4	0	5	0	30	31	130	10
5	0	0	0	43	23	91	10
6	5	10	0	179	255	117	10
7	0	0	92	41	255	251	10
8	96	0	0	179	255	255	10
9	7	0	26	106	9	136	10
10	0	10	99	45	42	255	10
11	0	0	111	26	255	255	10
12	0	12	109	179	255	255	10
13	0	33	92	179	255	176	10

Pothole	Lower inRange HSV			Upper inRange HSV			Gaussian
	H	S	V	H	S	V	
14	0	10	76	18	194	253	7
15	0	21	76	18	186	204	10
16	11	2	47	15	255	119	10
17	0	0	59	23	255	255	10
18	0	3	47	44	54	125	10
19	0	9	0	28	255	119	10
20	0	0	132	108	255	255	10
21	0	10	0	45	255	255	0

Dari 21 sampel yang diuji, bisa dicari beberapa parameternya seperti yang terlihat pada Tabel 4.4 Parameter yang di butuhkan yaitu minimum dan maksimum HSV dan Nilai *Gaussian Blur*nya.

Pengujian untuk memverifikasi dengan pendeteksian luas gambar bidang datar dengan luas yang sudah ditentukan dengan bidang datar berupa lingkaran, persegi, dan segitiga. Ada tiga objek yang disajikan pada Tabel 4.5

Tabel 4.5 Perbandingan Luas Estimasi dan Luas Hitung

No.	Bidang	Luas Estimasi (cm <sup>2</sup> )	Luas Hitung (cm <sup>2</sup> )
1	Lingkaran	660cm <sup>2</sup>	649.09 cm <sup>2</sup>
2	Persegi	400cm <sup>2</sup>	401.43 cm <sup>2</sup>
3	Segitiga	230cm <sup>2</sup>	220.36 cm <sup>2</sup>

Hasil pengujian pada tiga buah objek bidang datar yang telah ditentukan luasnya menghasilkan error berkisar antara 0.358% - 4.19%.

Pengambilan data diambil pada 5 titik di jalan di wilayah Kota Yogyakarta. 5 data lubang jalan tersebut diantaranya adalah sebagai berikut :

**1. Jl. Pandeyan, Umbulharjo, Kota Yogyakarta, Daerah Istimewa Yogyakarta 55164**

Tabel 4.6 Data Pengujian Koordinat Pertama

Hasil Pengujian Sistem (°)		Hasil Pengujian Google Maps (°)	
Latitude	Longitude	Latitude	Longitude
-7.809645	110.389289	-7.809646	110.389398



Gambar 4.13 Gambar Jalan Lokasi Pertama

**2. Jl. Warungboto I, Warungboto, Kec. Umbulharjo, Kota Yogyakarta, Daerah Istimewa Yogyakarta 55164**

Tabel 4.7 Data Pengujian Koordinat Kedua

Hasil Pengujian Sistem (°)		Hasil Pengujian Google Maps (°)	
Latitude	Longitude	Latitude	Longitude
-7.809677	110.389627	-7.809689	110.389651



Gambar 4.14 Gambar Jalan Lokasi Kedua

**3. Jl. Warungboto I, Warungboto, Kec. Umbulharjo, Kota Yogyakarta, Daerah Istimewa Yogyakarta 55164**

Tabel 4.8 Data Pengujian Koordinat Ketiga

Hasil Pengujian Sistem (°)		Hasil Pengujian Google Maps (°)	
Latitude	Longitude	Latitude	Longitude
-7.809684	110.389528	-7.809694	110.389574



Gambar 4.15 Gambar Jalan Lokasi Ketiga



4. Jl. Pandeyan, Umbulharjo, Kota Yogyakarta, Daerah Istimewa Yogyakarta 55164

Tabel 4.9 Data Pengujian Koordinat Keempat

Hasil Pengujian Sistem (°)		Hasil Pengujian Google Maps (°)	
Latitude	Longitude	Latitude	Longitude
-7.809645	110.389289	-7.809646	110.389398

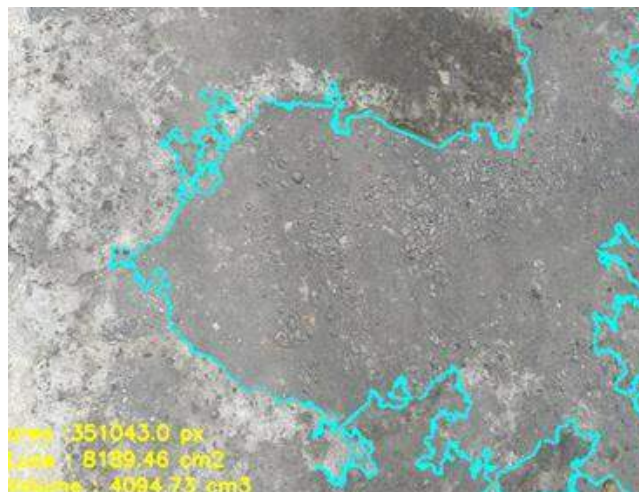


Gambar 4.16 Gambar Jalan Lokasi Keempat

5. Gg. Melati, Warungboto, Kec. Umbulharjo, Kota Yogyakarta, Daerah Istimewa Yogyakarta 55164

Tabel 4.10 Data Pengujian Koordinat Kelima

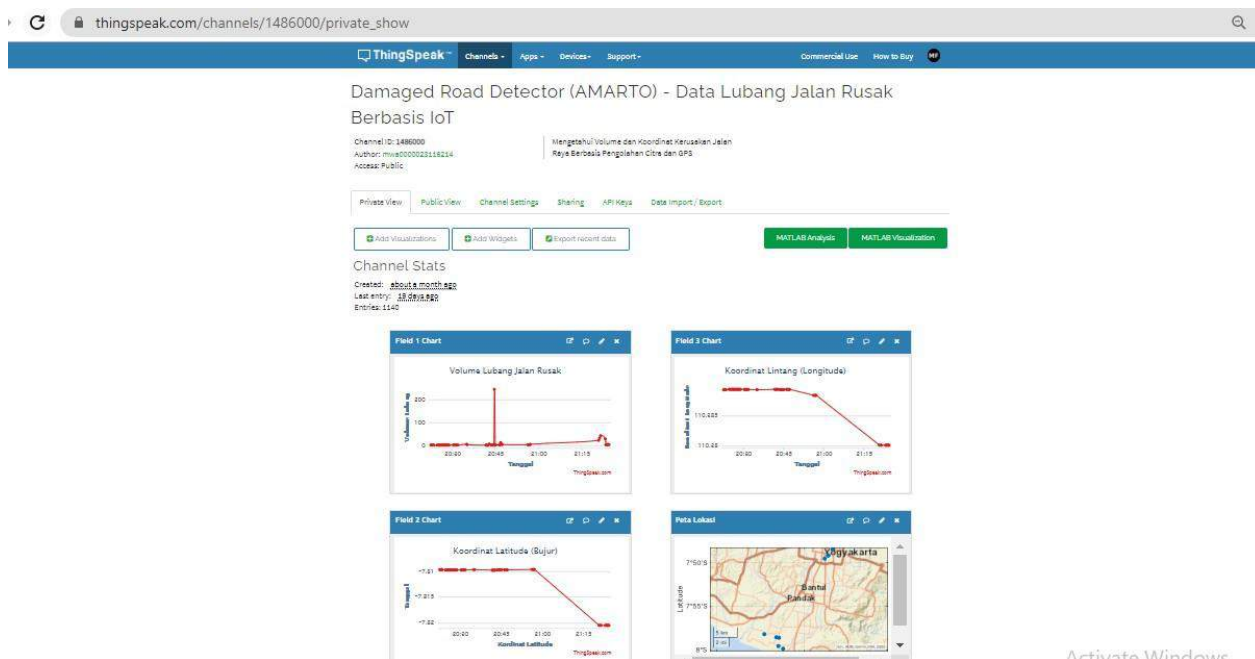
Hasil Pengujian Sistem (°)		Hasil Pengujian Google Maps (°)	
Latitude	Longitude	Latitude	Longitude
-7.809652	110.388371	-7.809644	110.388326



Gambar 4.17 Gambar Jalan Lokasi Kelima

• Implementasi Penampilan Data Berbasis *Internet of Things*

*Platform IoT ThingSpeak* juga dapat digunakan untuk menampilkan map suatu lokasi berdasarkan data-data yang telah dikumpulkan oleh *ThingSpeak*. Dalam penelitian ini penulis memanfaatkan fitur ini untuk menampilkan data persebaran lubang dengan data yang diterima dalam sistem ini. Sehingga data bisa diakses melalui *platform website* dan aplikasi *android* secara *realtime* dan terdokumentasikan dengan rapi dengan visualisasi data yang menarik seperti pada Gambar 4.18 dan Gambar 4.19.



Gambar 4.18 Tampilan Thingspeak Berbasis IoT di Website



Gambar 4.19 Tampilan Thingspeak Berbasis IoT di Aplikasi Android

## KESIMPULAN DAN REKOMENDASI

### 1. KESIMPULAN

- a. Telah terbuat purwarupa deteksi kerusakan lubang jalan berbasis citra digital dan *GPS*.
- b. Sistem ini mampu menganalisa dan menghitung luas jalan berlubang dengan tingkat error 0.358% - 4.19%, namun nilai parameter *lower*, *upper* HSV, *gaussian blur kernel*, *alpha*, *beta* pada fungsi masih beragam. Program yang berperan penting dalam proses pendeteksian jalan berlubang adalah *equalizer histogram*.
- c. Berdasarkan pengujian sensor jarak HC-SR04 dalam penelitian ini adalah didapatkan bahwa

untuk kedalaman lubang dengan pengukuran metode tidak bergerak maka didapatkan hasil *error* sebesar lebih kecil yaitu 1,66 % dan deteksi GPS dengan rata-rata selisih Latitude 0.0006144 dan Longitude 0.0000396.

- d. Implementasi IoT berbasis *Thinkspeak* sudah bisa dimanfaatkan sehingga monitoring bisa dipantau melalui *web* dan aplikasi *android*.

## 2. REKOMENDASI

- a. Bagi Dinas terkait penanggungjawab program AMARTO nantinya, dapat menindaklanjuti hasil penelitian ini untuk bisa diimplementasikan dalam sistem yang sudah ada sebagai *upgrade* dari sistem deteksi dan analisis kerusakan lubang jalan di wilayah Kota Yogyakarta.
- b. Bagi Peneliti selanjutnya, dapat melakukan penelitian lanjutan terkait implementasi dan evaluasi sistem AMARTO ini. Perlu dilakukan riset yang lebih mendalam tentang kebutuhan dari dinas terkait yang lebih luas dan relevan berkaitan dengan jalan raya. Misalnya deteksi kerusakan tidak hanya berupa lubang jalan bisa retakan atau juga analisis tingkat kenyamanan jalan.

## DAFTAR PUSTAKA

- Alwi K, Ismail E, Palupi IR (2019). Pengetahuan Keamanan Pangan Penjamah Makanan dan Mutu Keamanan Pangan di Pondok Pesantren. *Darussalam Nutrition Journal*, 3(2): 72 – 83
- P. Indonesia, “Undang-Undang Republik Indonesia Nomor 38 Tahun 2004 tentang jalan.” Sekretariat Negara, 2004.
- ASTM International, “Standard Practice for Roads and Parking Lots Pavement Condition Index Surveys.” West Conshohocken.
- D. Jenderal, B. Marga, D. Pembinaan, and J. Kota, “Tata Cara Penyusunan Program Pemeliharaan Jalan Kota,” no. 018, 1990.
- Irawan, Agus. Perancangan Sistem Deteksi Kerusakan Aspal Jalan Melalui Video Menggunakan Fast Fourier Transform, Prosiding SNRT, Politeknik Negeri Banjarmasin, 2016.
- A. Suswandi, W. Sartono, and H. C. H, “Evaluasi Tingkat Kerusakan Jalan Dengan Metode Pavement Condition Index ( PCI ) Untuk Menunjang Pengambilan Keputusan ( Studi Kasus : Jalan Lingkar Selatan , Yogyakarta ),” *Forum Tek. Sipil*, vol. XVIII, pp. 934–946, 2008.
- D. A. Saputro, L. Djakfar, and A. Rachmansyah, “Evaluasi Kondisi Jalan Dan Pengembangan Prioritas Penanganannya,” vol. 5, no. 2, pp. 76–83, 2011.
- T. K. H. Slamet Riyadi, Riza Azyumarridha Azra, Ramadoni Syahputra, “Kombinasi Teknik Thresholding, Median Filter Dan Morphological Closing,” *Simp. Nas. Teknol. Terap. (SNTT)*2, 2014.
- B. D. Idestio, T. Agung, and B. Wirayuda, “Alternative of Pothole Area Measurement Based-on Video using Threshold-based Marking and GLCM Alternatif Pengukuran Luas Lubang Jalan Berbasis Data Video Menerapkan Threshold-based Marking dan GLCM,” vol. 7, no. 2, 2013.
- D. Joubert, A. Tyatyantsi, J. Mphahlehle, and V. Manchidi, “Pothole Tagging System,” 4th

- Robot. Mechatronics Conf. South Africa, 2011.
- S. Lee, S. Kim, K. E. An, S. Ryu, and D. Seo, "Image Processing-based Pothole Detecting System for Driving Environment," *IEEE Int. Conf. Consum. Electron.*, pp. 0–1, 2018.
- A. Choudhury, R. Ramchandani, M. Shamoan, A. Khare, and K. Kaushik, "An Efficient Algorithm for Detecting and Measure the Properties of Pothole," *Pengguna. Pengolah. citra untuk mendeteksi Luban. jalan*, pp. 447–457, 2019.
- P. Wang, Y. Hu, Y. Dai, and M. Tian, "Asphalt Pavement Pothole Detection and Segmentation Based on Wavelet Energy Field," *Math. Probl. Eng.*, vol. 2017, 2017.
- I. M. Wahyu Hidayat Ibrahim, "Sistem Informasi Pelayanan Publik Berbasis Web Pada Dinas Pekerjaan Umum Kabupaten Kampar," *J. Ilm. Rekayasa dan Manaj. Sist. Inf.*, vol. 3, no. 2, pp. 17–22, 2017.
- D. O. Sihombing, "Perancangan Sistem Informasi Pelaporan Masyarakat Untuk Kerusakan Jalan Di Pontianak Menggunakan Google Maps Api," *J. Khatulistiwa Inform.*, vol. 4, no. 1, pp. 12–22, 2016.
- M. Abu et al., "Arduino-Uno Based Mobile Data Logger with GPS Feature," *TELKOMNIKA*, vol. 13, no. 1, pp. 250–259, 2015.
- R. G. H. Clarkson H. Oglesby, *Teknik jalan raya*, 1st ed. Jakarta: Gramedia, 1999.
- P. R. Indonesia, "Peraturan Pemerintah Republik Indonesia Nomor 44 Tahun 1993 Tentang Kendaraan Dan Pengemudi," vol. 8. 1993.
- P. R. Indonesia, "Peraturan Pemerintah Republik Indonesia Nomor 43 Tahun 1993 Tentang Prasarana Dan Lalu Lintas Jalan Presiden Republik Indonesia." 1993.
- P. Y. Budiarto, "Deteksi Objek Lubang pada Citra Jalan Raya menggunakan Pengolahan Citra Digital," *J. Politek. Caltex Riau*, vol. 3, no. 2, pp. 109–118, 2017.
- C. Dong, M. C. Leu, and Z. Yin, "American Sign Language Alphabet Recognition Using Microsoft Kinect," *Comput. Vis. Pattern Recognit. Work.*, 2015.
- Surface Distress Index ( SDI ) Dan Pavement Condition Index ( PCI ) Untuk Penilaian Kondisi Jalan Di," *Semin. Nas. Sains dan Teknol.*, no. November, pp. 1–9, 2016.
- L. M. N. Arsyad and S. N. Ahmad, "Penilaian Kondisi Struktur Kerusakan Perkerasan Jalan Berbasis Metode Pavement Condition Indeks (PCI)," *J. STABILITA*, vol. 6, no. 3, pp. 35–42, 2018.