

# Region-Based Detection of Four-Wheeled Vehicle Density on Highways Using Euclidean Distance

Neng Sri Lathifah Zulfa<sup>1</sup>, Novita Siti Julaeaha<sup>2</sup>, Mina Ismu Rahayu<sup>3</sup>, Yus Jayusman<sup>4</sup>

<sup>1</sup> Universitas Islam Al-Ihya Kuningan, Jl. Mayasih No.11, Cigugur, Kuningan, Indonesia

<sup>2,3,4</sup> STMIK Bandung, Jl. Cikutra No. 113 Cikutra, Cibeunying Kidul, Bandung, Indonesia

Surel: [srilathifah@gmail.com](mailto:srilathifah@gmail.com)<sup>1</sup>, [novitasitijulaeha03@gmail.com](mailto:novitasitijulaeha03@gmail.com)<sup>2</sup>, [mina@stmik-bandung.ac.id](mailto:mina@stmik-bandung.ac.id)<sup>3</sup>, [yus@stmik-bandung.ac.id](mailto:yus@stmik-bandung.ac.id)<sup>4</sup>

[Dikirim: 26 Juni 2025] [Direview: 19 Juli 2025] [Diterima: 23 Juli 2025]

DOI: 10.58761/juristikstmikbandung.v14.i1.182

## ABSTRAK

Kemacetan lalu lintas di kawasan perkotaan seperti Kota Bandung merupakan permasalahan krusial yang memerlukan solusi cerdas dan efisien. Penelitian ini mengusulkan sistem deteksi kepadatan kendaraan roda empat berbasis citra menggunakan pendekatan region-based yang dikombinasikan dengan algoritma Euclidean Distance. Citra lalu lintas dianalisis untuk menghitung jarak antar kendaraan berdasarkan titik pusat (centroid), yang kemudian digunakan untuk mengklasifikasikan kondisi lalu lintas ke dalam tiga kategori: lancar, sedang, dan padat. Sistem dikembangkan menggunakan bahasa pemrograman Python dan antarmuka web Streamlit. Pengujian fungsional dilakukan dengan metode Black Box Testing. Hasil eksperimen menunjukkan bahwa sistem mampu mendeteksi dan mengklasifikasikan kepadatan lalu lintas secara otomatis dan real-time, sehingga berpotensi menjadi solusi praktis untuk mendukung pengelolaan lalu lintas perkotaan.

**Kata kunci:** deteksi kepadatan lalu lintas, kendaraan roda empat, metode region-based, Euclidean Distance, pengolahan citra, lalu lintas perkotaan

## ABSTRACT

Traffic congestion in urban areas such as Bandung has become a critical issue that demands intelligent and efficient solutions. This study proposes an image-based vehicle density detection system for four-wheeled vehicles using a region-based approach combined with the Euclidean Distance algorithm. Traffic images are analyzed to calculate inter-vehicle distances based on centroid points, and the results are used to classify traffic conditions into three categories: free-flowing, moderate, and congested. The system is developed using the Python programming language and the Streamlit web interface. Functional testing is conducted using the Black Box Testing method. Experimental results demonstrate that the system can automatically and reliably detect and classify traffic density in real time, offering a practical solution to support urban traffic management.

**Keywords:** traffic density detection, four-wheeled vehicles, region-based method, Euclidean Distance, image processing, urban traffic

## 1. PENDAHULUAN

Perkembangan teknologi digital telah memberikan dampak yang signifikan terhadap berbagai aspek kehidupan, termasuk sektor transportasi. Melalui kemajuan teknologi informasi dan komunikasi, sistem transportasi dapat dioptimalkan dengan pemanfaatan perangkat lunak dan perangkat keras yang canggih (Irawanto et al., 2022). Seiring dengan perkembangan urbanisasi, jumlah kendaraan bermotor meningkat setiap tahun, dan tren ini menyebabkan lingkungan lalu lintas menjadi lebih kompleks (Wang et al., 2025). Terutama pada jalan-jalan utama yang menjadi arteri vital pergerakan logistik dan mobilitas antarkota (Sun & Liu, 2025). Kondisi saat ini, banyak persimpangan perkotaan dilengkapi dengan kamera pengawas yang terhubung ke sistem manajemen lalu Lintas

(Ghahremannezhad et al., 2022). Dalam konteks ini, *Computer Vision* telah memainkan peran penting dalam *Intelligent Transportation Systems* (ITS) dan pengawasan lalu lintas. Seiring dengan pesatnya pertumbuhan kendaraan otomatis dan kepadatan kota, *advanced traffic management system* (ATMS) otomatis dan canggih yang menggunakan infrastruktur pengawasan video (Rezaei et al., 2022).

Salah satu tantangan utama dalam pengelolaan transportasi adalah kemacetan lalu lintas, yang sering kali disebabkan oleh distribusi lampu lalu lintas yang kurang efisien di persimpangan jalan. Ketidakseimbangan ini dapat menimbulkan akumulasi kendaraan yang melebihi kapasitas jalan, sehingga menghambat kelancaran arus lalu lintas (Wahyudi & H. Kartowisastro, 2021). Selain itu, keterbatasan sarana dan prasarana seperti sistem pemantauan lalu lintas, lebar jalan, serta ketersediaan rambu-rambu turut memperburuk situasi. Sistem pemantauan yang digunakan saat ini umumnya masih bersifat konvensional, yakni dengan mengandalkan kamera statis yang dipantau secara manual oleh petugas. Meskipun sistem ini telah berjalan sesuai dengan regulasi yang berlaku, proses validasi kondisi kemacetan masih mengalami kendala akibat keterbatasan data pembanding (Solihun et al., 2022).

Sistem pengelolaan lalu lintas di perkotaan sangat bergantung pada kemampuan untuk mendeteksi kejadian-kejadian tidak normal (anomali) pada lalu lintas secara efektif. Hal ini penting untuk mencegah kemacetan dan mengurangi angka kecelakaan. Dengan pesatnya pertumbuhan kota dan semakin rumitnya jaringan lalu lintas, metode statistik tradisional kini tidak lagi memadai untuk memprediksi kondisi lalu lintas dengan akurat (Giasemis & Sopasakis, 2025). Khususnya di lingkungan jalan utama, deteksi anomali dan kepadatan menjadi sangat penting mengingat kecepatan kendaraan yang tinggi dan dampak berantai yang dapat ditimbulkan oleh insiden kecil, seperti kecelakaan, kemacetan mendadak, atau perubahan kondisi jalan akibat cuaca buruk (Sun & Liu, 2025).

Lalu lintas merupakan aspek vital dalam manajemen transportasi perkotaan yang berdampak langsung pada efisiensi, keselamatan, dan kenyamanan pengguna jalan. Oleh karena itu, pengawasan dan pengelolaan kepadatan lalu lintas menjadi faktor kunci dalam meningkatkan mobilitas dan mengurangi kemacetan (Du et al., 2023). Seiring dengan meningkatnya jumlah kendaraan setiap tahunnya, volume lalu lintas pun mengalami kenaikan yang signifikan, sehingga menyebabkan kepadatan di sejumlah titik. Untuk mengantisipasi hal tersebut, diperlukan informasi statistik yang akurat mengenai pertumbuhan jumlah kendaraan pada wilayah tertentu (Irawanto et al., 2022).

Sebagai respon terhadap permasalahan tersebut, diperlukan pendekatan teknologi yang lebih adaptif, akurat, dan otomatis. Salah satunya adalah dengan memanfaatkan analisis citra menggunakan *metode region-based* dan pengukuran jarak spasial seperti *Euclidean Distance*. *Metode region-based* memungkinkan segmentasi citra ke dalam wilayah-wilayah terpisah yang dapat dianalisis untuk mendeteksi objek seperti kendaraan (Khan et al., 2024). Sementara itu, *Euclidean Distance* digunakan untuk mengukur kedekatan antar objek atau wilayah dalam citra guna mengidentifikasi tingkat kepadatan kendaraan. (Wahyudi & H. Kartowisastro, 2021).

### 1.1 Perumusan Masalah

Berdasarkan latar belakang tersebut didapatkan rumusan masalah dalam penelitian ini diantaranya:

1. Bagaimana *metode region-based* dengan menggunakan *euclidean distance* dapat diterapkan dalam deteksi kepadatan lalu lintas?
2. Bagaimana metode yang diusulkan dapat diaplikasikan untuk memprediksi kepadatan lalu lintas dari citra yang diunggah ke sistem?

### 1.2 Tujuan

Adapun tujuan dari penelitian deteksi rambu lalu lintas dengan :

1. Menerapkan metode *region-based* menggunakan *euclidean distance* dalam konteks lalu lintas.
2. Menciptakan sistem yang praktis dan dapat diandalkan untuk memprediksi dan menyimpulkan kondisi lalu lintas dari citra yang diunggah.

### 1.3 Objek Penelitian

Objek dalam penelitian ini adalah citra lalu lintas jalan raya yang menampilkan kendaraan roda empat, khususnya pada ruas Jalan Jakarta di Kota Bandung. Citra ini digunakan sebagai data input dalam sistem deteksi kepadatan lalu lintas berbasis metode *region-based* dan *Euclidean distance*.

## 1.4 Urgensi

Penelitian ini penting untuk dilakukan karena kepadatan lalu lintas di wilayah perkotaan seperti Kota Bandung semakin meningkat seiring pertumbuhan jumlah kendaraan, sementara sistem pemantauan lalu lintas yang ada masih bersifat manual dan belum efisien. Penerapan teknologi pengolahan citra digital dan algoritma Euclidean distance dapat memberikan solusi otomatis dalam mendeteksi kepadatan lalu lintas secara real-time, sehingga mendukung pengambilan keputusan yang lebih cepat dan akurat. Hasil dari penelitian ini diharapkan menjadi model awal pengembangan sistem pemantauan lalu lintas berbasis visual yang sederhana dan dapat diterapkan di berbagai wilayah dengan keterbatasan infrastruktur canggih.

## 1.5 Kebaruan Penelitian

Kebaruan dalam penelitian ini terletak pada:

1. Penerapan metode region-based yang dikombinasikan dengan algoritma Euclidean distance untuk mengklasifikasikan kepadatan lalu lintas berdasarkan citra kendaraan roda empat, yang jarang digunakan pada sistem pemantauan lalu lintas konvensional.
2. Pengembangan sistem deteksi kepadatan berbasis web yang dapat memproses input citra secara langsung, tanpa membutuhkan perangkat keras tambahan seperti sensor khusus atau sistem kamera canggih.
3. Penentuan klasifikasi kondisi lalu lintas (lancar, sedang, padat) dilakukan secara otomatis berdasarkan perhitungan jarak rata-rata antar kendaraan dalam citra, menawarkan pendekatan praktis dan efisien yang dapat diadaptasi untuk berbagai konteks jalan perkotaan.

## 2. METODOLOGI

Metodologi yang digunakan pada deteksi kepadatan kendaraan roda 4 pada jalan raya ini terbagi kedalam 2 sub-bab yaitu sebagai berikut:

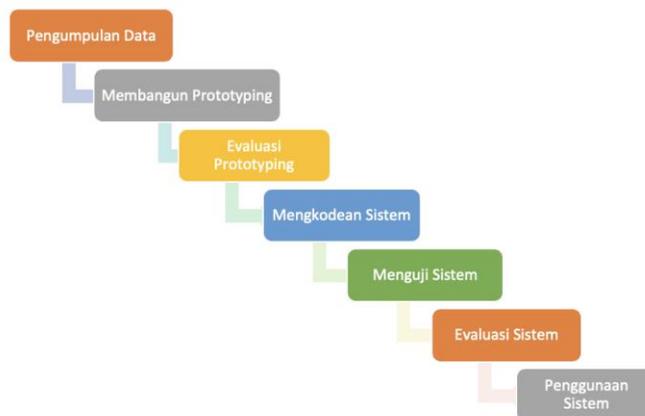
### 1.1 Metode Pengumpulan Data

Pada pengumpulan data terkait kepadatan lalu lintas ini dilakukan dengan 2 (dua) cara yaitu:

1. Penelitian Kepustakaan (*Library research*), Metode ini melibatkan pengumpulan dan analisis berbagai sumber kepustakaan yang relevan. Tujuannya adalah untuk memperoleh landasan teoritis yang komprehensif dan memadai yang mendukung penelitian. Sumber-sumber yang dimaksud meliputi buku-buku referensi, publikasi ilmiah, jurnal, serta materi perkuliahan yang memiliki keterkaitan erat dengan topik penelitian yang sedang dilakukan. (Sriharyani & Hadijah, 2023).
2. Observasi, pendekatan ini dilakukan dengan melaksanakan observasi secara langsung di lokasi lalu lintas yang menjadi objek penelitian. Tujuan dari observasi ini adalah untuk mengumpulkan data citra yang merepresentasikan kondisi kepadatan lalu lintas di lingkungan nyata. Data citra ini esensial untuk analisis empiris lebih lanjut terkait deteksi kepadatan kendaraan. (Sriharyani & Hadijah, 2023).

### 1.2 Metode Pengembangan Sistem

Penelitian ini menggunakan pendekatan rekayasa perangkat lunak (*software engineering*) dengan metode *Prototype* untuk merancang dan mengembangkan sistem deteksi kepadatan kendaraan roda empat berbasis citra digital (Bjarnason et al., 2023).



Gambar 1. Metode *Prototype*

Tahapan pengembangan sistem dalam metode Prototype pada penelitian ini meliputi:

- a. **Pengumpulan Kebutuhan (*Requirements Elicitation*)**

Tahap ini berfokus pada identifikasi komprehensif terhadap semua kebutuhan perangkat lunak dan sistem secara keseluruhan yang akan dikembangkan. Proses ini melibatkan pemahaman mendalam mengenai fungsionalitas yang diharapkan, batasan sistem, serta kebutuhan pengguna dan pemangku kepentingan.
- b. **Pembangunan Prototipe (*Prototyping*)**

Pada fase ini, prototipe sistem dibangun sebagai representasi awal atau model sementara dari aplikasi yang diinginkan. Pembangunan prototipe mencakup perancangan antarmuka masukan (*input*) dan keluaran (*output*) untuk memberikan gambaran fungsionalitas dasar sistem yang akan dihasilkan.
- c. **Evaluasi Prototipe (*Prototype Evaluation*)**

Tahap evaluasi prototipe merupakan proses krusial yang memerlukan peninjauan cermat sebelum melangkah ke tahapan pengembangan selanjutnya. Keberhasilan proyek sangat ditentukan oleh akurasi dan kelengkapan pada tahap pengumpulan kebutuhan dan pembangunan prototipe. Apabila terdapat kekurangan atau kesalahan substansial pada kedua tahap awal ini, akan timbul kesulitan signifikan dalam melanjutkan ke langkah-langkah pengembangan berikutnya.
- d. **Pengkodean Sistem (*System Coding*)**

Dalam tahap ini, kebutuhan yang telah didefinisikan dan disetujui diimplementasikan ke dalam bentuk kode program. Fase ini seringkali menjadi tantangan tersendiri karena memerlukan translasi spesifikasi desain ke dalam struktur dan logika pemrograman yang fungsional.
- e. **Pengujian Sistem (*System Testing*)**

Setelah proses pengkodean selesai, sistem akan menjalani tahap pengujian untuk memverifikasi fungsionalitas dan kinerjanya. Metode pengujian yang dapat diterapkan *Black Box Testing*, metode ini berfokus pada pengujian fungsionalitas sistem dari perspektif pengguna tanpa memperhatikan struktur internal kode, yakni dengan mengevaluasi keluaran berdasarkan masukan dan tampilan sistem.
- f. **Evaluasi Sistem (*System Evaluation*)**

Tahap ini melibatkan penilaian menyeluruh untuk memastikan apakah sistem yang telah selesai dikembangkan telah memenuhi ekspektasi dan sesuai dengan kebutuhan yang telah didefinisikan di awal proyek. Evaluasi ini mencakup verifikasi kepatuhan terhadap spesifikasi dan validasi bahwa sistem dapat menyelesaikan masalah yang dituju.
- g. **Implementasi Sistem (*System Deployment/Usage*)**

Setelah perangkat lunak berhasil melewati seluruh proses pengujian dan mendapatkan persetujuan dari pelanggan atau pengguna akhir, sistem siap untuk diimplementasikan dan digunakan dalam lingkungan operasional yang sebenarnya (Bjarnason et al., 2023).

### 3. HASIL DAN PEMBAHASAN

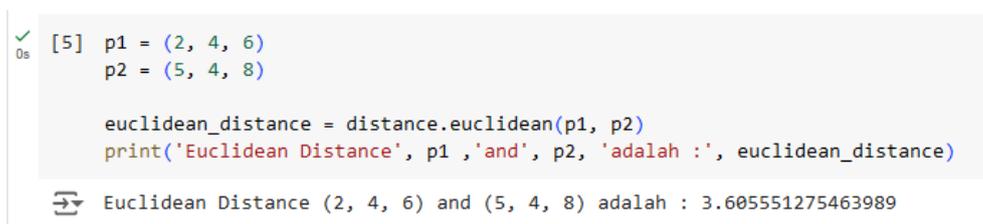
Bagian ini menyajikan hasil eksperimen yang telah dilakukan serta pembahasan mendalam mengenai implikasi dari temuan-temuan tersebut. Analisis difokuskan pada efektivitas metode deteksi *region-based* dengan pengukuran *Euclidean Distance* dalam mengidentifikasi kepadatan kendaraan roda empat di jalan raya, serta perbandingan kinerja dengan pendekatan yang relevan.

#### A. Pengumpulan Kebutuhan (*Requirements Elicitation*)

##### 1. Analisis Algoritma *Euclidean Distance* untuk Perhitungan Kepadatan lalu Lintas

Algoritma *Euclidean Distance* digunakan untuk mengukur jarak antara dua titik dalam ruang *Euclidean*. Dalam konteks kepadatan lalu lintas, dua titik tersebut merupakan pusat (*centroid*) dari dua kendaraan yang terdeteksi. Semakin dekat jarak antara dua kendaraan, maka diasumsikan kondisi lalu lintas semakin padat, dan sebaliknya.

Pada penelitian ini jarak *euclidean* menjadi bagian penting dalam proses deteksi kepadatan lalu lintas. Berikut gambaran algoritma euclidean distance yang digunakan pada proses deteksi kepadatan lalu lintas.



```

[5] p1 = (2, 4, 6)
    p2 = (5, 4, 8)

euclidean_distance = distance.euclidean(p1, p2)
print('Euclidean Distance', p1, 'and', p2, 'adalah :', euclidean_distance)
Euclidean Distance (2, 4, 6) and (5, 4, 8) adalah : 3.605551275463989

```

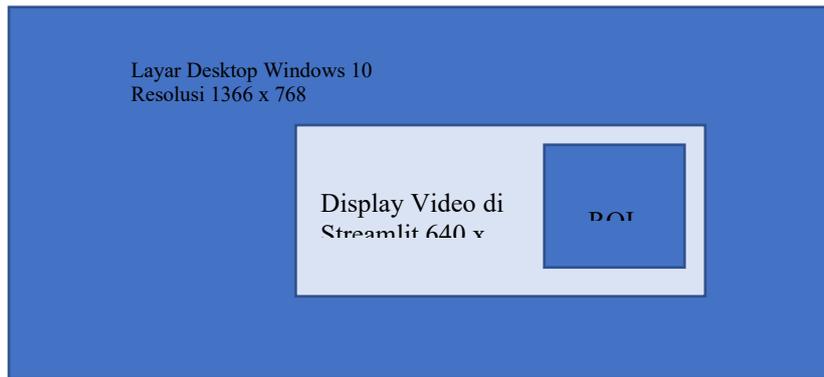
Gambar 2. Algoritma *euclidean distance*

*Euclidean Distance* dalam deteksi kepadatan lalu lintas ini bekerja dengan beberapa langkah, yaitu sebagai berikut:

- Penentuan pusat massa (*centroid*). Langkah awal ini dilakukan pada setiap kendaraan yang terdeteksi, centroid ini dianggap sebagai titik representatif dari kendaraan tersebut.
- Perhitungan jarak. Setelah centroid diketahui, algoritma *euclidean distance* digunakan untuk menghitung jarak antar setiap pasang *centroid* dari kendaraan yang terdeteksi yang dimana jarak ini merupakan jarak terdekat antara dua kendaraan.
- Penentuan ambang batas. Untuk penentuan kondisi lalu lintas padat atau tidak, maka ditetapkan sebuah ambang batas jarak, dimana jika jarak antara dua kendaraan lebih kecil dari ambang batas yang telah ditentukan maka dapat disimpulkan bahwa kepadatan lalu lintas di area tersebut tinggi dan sebaliknya jika jarak antara dua kendaraan lebih besar dari nilai ambang batasnya maka kepadatannya dikategorikan rendah.

##### 2. Analisis Klasifikasi Kepadatan Lalu Lintas

Sebelum masuk pada penghitungan jarak terlebih dahulu ditentukan batas wilayah untuk proses deteksi, disini digunakan ROI (*Region Of Interest*) (Hendrizal, 2023). Untuk menentukan nilai dari ROI ini menggunakan aplikasi bawaan dari desktop yaitu *paint* untuk mengetahui nilai pixel wilayah deteksi. Dari pixel desktop dengan nilai pixel 1366 x 764 diambil pixel display video dari streamlit dan didapat nilai pixel 640 x 400 kemudian disini ditentukan wilayah untuk deteksi. Berikut ilustrasi penentuan ROI dalam wilayah deteksi kepadatan lalu lintas.



Gambar 3. Ilustrasi penentuan *ROI (Region Of Interest)*

Dalam menentukan jenis kondisi lalu lintas pada citra yang di deteksi dilakukan proses klasifikasi dengan acuan jarak rata-rata antar kendaraan. Jarak rata-rata ini didapat dari proses perhitungan setiap jarak *centroid* yang telah ditentukan dalam analisis algoritma *euclidean distance* sebelumnya.

### 3. Analisis Kebutuhan Fungsional

Analisis fungsional ini berisi proses-proses apa saja yang dilakukan oleh sistem. Perancangan *website* ini merupakan pendeteksian kepadatan lalu lintas berdasarkan penghitungan kendaraan dilihat dari jarak antar kendaraan yang akan memberikan gambaran dan dapat menjadi parameter untuk pengambilan keputusan pada pengelolaan lalu lintas. Dalam pembangunan website ini diharapkan adanya fungsi yang terdapat dalam *website Traffic Density Detector* berdasarkan kebutuhan yang diperlukan, antara lain:

- a. Proses pemilihan citra yang digunakan sebagai proses awal dalam deteksi kepadatan lalu lintas
- b. Proses deteksi kepadatan lalu lintas berdasarkan jarak kendaraan yang terdeteksi dari citra yang dipilih.
- c. Proses menampilkan hasil deteksi berupa jumlah kendaraan yang terdeteksi dan klasifikasi kepadatan lalu lintas.

### 4. Analisis Kebutuhan Sistem

Bagian ini menjelaskan kebutuhan teknis utama dari sistem yang dikembangkan untuk mendeteksi kepadatan lalu lintas kendaraan roda empat berbasis citra. Dua aspek utama yang dianalisis meliputi penggunaan algoritma Euclidean distance untuk penghitungan kepadatan, serta proses klasifikasi kondisi lalu lintas berdasarkan nilai rata-rata jarak antar kendaraan.

Tabel 1. Analisis Algoritma *Euclidean Distance*

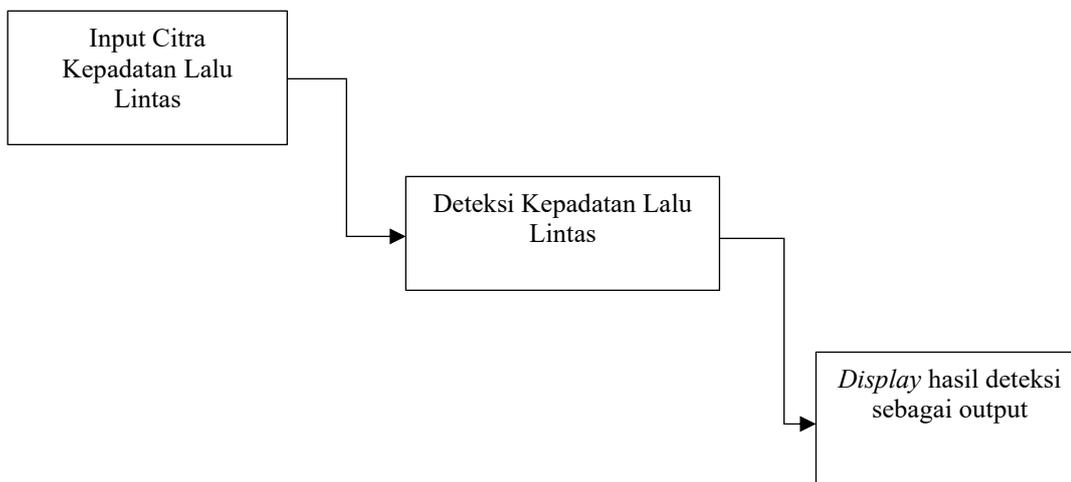
Tahapan	Deskripsi
Penentuan <i>Centroid</i>	Setiap kendaraan yang terdeteksi dalam citra ditentukan titik pusatnya ( <i>centroid</i> ) sebagai representasi posisi kendaraan.
Perhitungan Jarak	Setelah <i>centroid</i> diketahui, algoritma <i>euclidean distance</i> digunakan untuk menghitung jarak antar setiap pasang <i>centroid</i> dari kendaraan yang terdeteksi yang dimana jarak ini merupakan jarak terdekat antara dua kendaraan
Penentuan Ambang Batas	Jika rata-rata jarak antar kendaraan < ambang batas, maka lalu lintas dikategorikan padat. Jika > ambang batas, lalu lintas dikategorikan lancar.

**Tabel 2. Analisis Klasifikasi Kepadatan Lalu Lintas**

Tahapan	Deskripsi
Penentuan ROI	Wilayah deteksi dibatasi menggunakan Region of Interest (ROI) berdasarkan resolusi tampilan Streamlit ( $640 \times 400$ piksel).
Segmentasi Kendaraan	Sistem memisahkan objek kendaraan dari latar belakang citra.
Deteksi Objek	Sistem mendeteksi jumlah kendaraan yang berada dalam wilayah ROI.
Perhitungan Jarak <i>Euclidean</i>	Menghitung jarak antar <i>centroid</i> kendaraan yang terdeteksi menggunakan algoritma <i>Euclidean distance</i> .
Rata-rata Jarak	Sistem menghitung rata-rata jarak dari seluruh pasang kendaraan.
Klasifikasi Kepadatan	Menentukan kondisi lalu lintas berdasarkan rata-rata jarak sebagai berikut: a. $< 20$ piksel $\rightarrow$ Padat b. $20-49$ piksel $\rightarrow$ Sedang c. $\geq 50$ piksel $\rightarrow$ Lancar
<i>Output</i> Sistem	Sistem menampilkan jumlah kendaraan dan hasil klasifikasi kepadatan secara visual dan teks.

**B. Pembangunan Prototipe (*Prototyping*)**

Deteksi kepadatan kendaraan roda 4 dengan metode *region-based* menggunakan *euclidean distance* ini disebut dengan *Traffic Density Detector*, sebuah website yang dirancang untuk memberikan informasi mengenai kepadatan lalu lintas. *Traffic Density Detector* menggunakan teknologi citra digital dan konsep perhitungan jarak *euclidean* untuk secara efisien mengidentifikasi kepadatan lalu lintas berdasarkan jarak antar kendaraan dengan harapan hasil output dari sistem ini dapat memberikan informasi mengenai kondisi lalu lintas untuk dapat dijadikan parameter pengambilan keputusan para pemangku kepentingan pengelola lalu lintas.



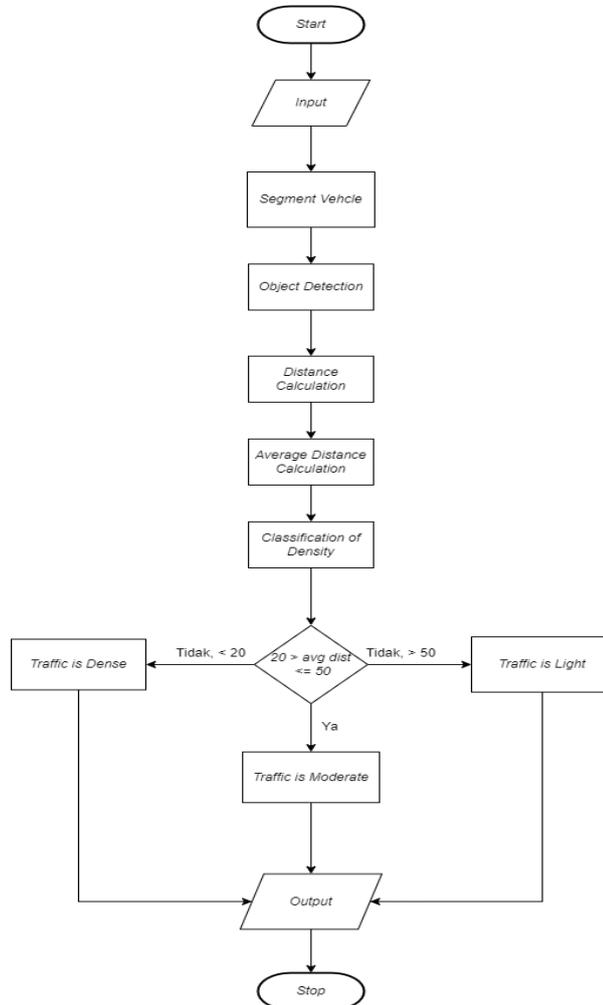
**Gambar 4. Gambaran Umum Sistem**

Gambaran umum sistem sebagai berikut:

1. Pengguna memasukkan cita kepadatan lalu lintas pada sistem *Traffic Density Detector*.
2. Sistem akan mendeteksi jumlah kendaraan dan mengklasifikasikan kondisi lalu lintas berdasarkan analisis algoritma *euclidean distance*.

3. Setelah sistem berjalan dan berhasil mendeteksi kepadatan lalu lintas maka akan menghasilkan output berupa display hasil deteksi berupa informasi jumlah kendaraan yang terdeteksi dan klasifikasi kondisi lalu lintasnya.

Berikut gambaran singkat proses klasifikasi kepadatan lalu lintas dengan menghitung rata-rata dari jarak *euclidean*.



Gambar 5. Flowchart Klasifikasi Kepadatan Lalu Lintas

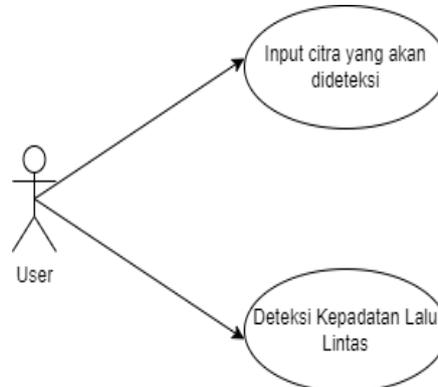
Penjelasan proses klasifikasi kepadatan lalu lintas berdasarkan jarak *euclidean*, sebagai berikut:

1. *Input*; Proses masukan berupa citra yang terdapat kendaraan untuk dideteksi
2. *Segment Vehicles*; Proses pemisahan objek kendaraan dengan latar belakang gambar
3. *Object Detection*; Proses deteksi objek kendaraan untuk menghitung jumlah kendaraan dalam citra inputan
4. *Distance Calculation*; Proses perhitungan jarak euclidean antar kendaraan
5. *Average Distance Calculation*; Proses perhitungan rata-rata jarak euclidean untuk menentukan kondisi lalu lintas.
6. *Classification of Density*; Proses pengklasifikasian untuk mengetahui kondisi lalu lintas dengan ketentuan sebagai berikut :
  - a. Jika rata-rata jarak euclidean  $< 20$  : Lalu lintas padat
  - b. Jika  $20 \leq$  jarak rata-rata euclidean  $< 50$  : Lalu lintas sedang
  - c. Jika rata-rata jarak euclidean  $\geq 50$  : Lalu lintas lancar

7. *Output*; Keluaran yang dihasilkan berupa informasi jumlah kendaraan yang terdeteksi dan hasil klasifikasi kepadatan lalu lintas

### C. Evaluasi Prototipe (*Prototype Evaluation*)

*Use Case* diagram menggambarkan hubungan antara aktor (pengguna) dengan sistem.



Gambar 6. *Use Case Diagram*

*Use case* diagram di atas menggambarkan interaksi antara seorang pengguna (*user*) dan sistem yang digunakan untuk mendeteksi kepadatan lalu lintas berdasarkan input citra yang diberikan (Oktavian, 2024). Berikut adalah penjelasan rinci mengenai elemen-elemen dalam diagram tersebut:

#### 1. Input Citra yang Akan Dideteksi

Tabel 3. Input Citra

Deskripsi	Interaksi
Pengguna memberikan input berupa citra atau video yang akan dianalisis oleh sistem. Input ini dapat berupa gambar statis atau frame video yang mengandung informasi mengenai kondisi lalu lintas.	Pengguna mengunggah citra atau video ke dalam sistem melalui antarmuka yang disediakan.

#### 2. Deteksi Kepadatan Lalu Lintas

Tabel 4. Deteksi Kepadatan

Deskripsi	Interaksi
Sistem melakukan analisis terhadap citra atau video yang telah diunggah untuk mendeteksi jumlah kendaraan dan menghitung kepadatan lalu lintas berdasarkan algoritma tertentu.	Sistem memproses input yang diberikan pengguna dan memberikan output berupa informasi mengenai kondisi kepadatan lalu lintas, misalnya apakah lalu lintas dalam keadaan lancar, sedang, atau padat

### D. Pengkodean Sistem (*System Coding*)

Pada tahap implementasi, sistem dibangun dengan OpenCV memfasilitasi implementasi metode region-based untuk segmentasi objek kendaraan, deteksi fitur, dan manipulasi citra lainnya yang diperlukan dalam identifikasi kendaraan. NumPy mendukung perhitungan matematis kompleks, termasuk komputasi yang diperlukan untuk metode *Euclidean Distance* dalam mengukur kedekatan antar objek kendaraan. *Streamlit* memungkinkan pembuatan prototipe dan aplikasi berbasis web dengan cepat menggunakan kode Python murni, memfasilitasi visualisasi hasil deteksi kepadatan secara *real-time* dan interaktif.

### E. Pengujian Sistem (*System Testing*)

Pengujian sistem dilakukan menggunakan metode *Black Box Testing*, yang bertujuan menguji apakah fitur-fitur dalam sistem bekerja sesuai dengan spesifikasi tanpa melihat struktur kode program (Raihan & Voutama, 2023). Pengujian *black box testing* disebut sebagai pengujian perilaku. Dimana struktur interior, logika perangkat lunak yang diuji tidak diketahui oleh penguji. Penguji didasarkan kepada spesifikasi kebutuhan dan tidak perlu dilakukannya analisis kode (Praniffa et al., 2023).

#### 1. Kasus dan Hasil Pengujian

##### a. Pengujian Input Citra

Pengujian penginputan citra ini dilakukan untuk mengetahui apakah citra yang dipilih berhasil masuk ke sistem atau tidak sebelum digunakan untuk dilakukan pendeteksian. Berikut pengujian yang yang dijelaskan dalam tabel dibawah.

**Tabel 5. Pengujian Input Citra**

<b>Kasus dan Hasil Uji (Data Normal)</b>			
<b>Data</b>	<b>Yang Diharapkan</b>	<b>Hasil Uji</b>	<b>Kesimpulan</b>
Input Citra	Sistem dapat menampilkan input citra	Sistem berhasil menampilkan input citra	OK
<b>Kasus dan Hasil Uji (Data Salah)</b>			
<b>Data</b>	<b>Yang Diharapkan</b>	<b>Hasil Uji</b>	<b>Kesimpulan</b>
Input Citra	Sistem tidak menampilkan input citra	Sistem tidak berhasil menampilkan input citra	OK

##### b. Pengujian Deteksi Kepadatan Lalu Lintas

Pengujian ini dilakukan untuk mengetahui apakah algoritma sudah dapat mendeteksi kendaraan dan menganalisis kondisi lalu lintas berdasarkan jarak euclidean pada citra.

**Tabel 6. Pengujian Deteksi Kepadatan Lalu Lintas**

<b>Kasus dan Hasil Uji (Data Normal)</b>			
<b>Data</b>	<b>Yang Diharapkan</b>	<b>Hasil Uji</b>	<b>Kesimpulan</b>
Deteksi Kepadatan Lalu Lintas	Algoritma dapat mendeteksi kendaraan dan menganalisis kondisi lalu lintas pada citra	Algoritma berhasil mendeteksi kendaraan dan menganalisis kondisi lalu lintas pada citra	OK
<b>Kasus dan Hasil Uji (Data Salah)</b>			
<b>Data</b>	<b>Yang Diharapkan</b>	<b>Hasil Uji</b>	<b>Kesimpulan</b>
Deteksi Kepadatan Lalu Lintas	Algoritma tidak dapat mendeteksi kendaraan dan menganalisis kondisi lalu lintas pada citra	Algoritma tidak berhasil mendeteksi kendaraan dan menganalisis kondisi lalu lintas	OK

**F. Evaluasi Sistem (System Evaluation)**

Bagian ini menjelaskan kebutuhan teknis utama dari sistem yang dikembangkan untuk mendeteksi kepadatan lalu lintas kendaraan roda empat berbasis citra. Dua aspek utama yang dianalisis meliputi penggunaan algoritma *Euclidean distance* untuk penghitungan kepadatan, serta proses klasifikasi kondisi lalu lintas berdasarkan nilai rata-rata jarak antar kendaraan (Uswah, 2024).

**Tabel 7. Analisis Algoritma Euclidean Distance**

Tahapan	Deskripsi
Penentuan <i>Centroid</i>	Setiap kendaraan yang terdeteksi dalam citra ditentukan titik pusatnya ( <i>centroid</i> ) sebagai representasi posisi kendaraan.
Perhitungan Jarak	Setelah <i>centroid</i> diketahui, algoritma <i>euclidean distance</i> digunakan untuk menghitung jarak antar setiap pasang <i>centroid</i> dari kendaraan yang terdeteksi yang dimana jarak ini merupakan jarak terdekat antara dua kendaraan
Penentuan Ambang Batas	Jika rata-rata jarak antar kendaraan < ambang batas, maka lalu lintas dikategorikan padat. Jika > ambang batas, lalu lintas dikategorikan lancar.

**Tabel 8. Analisis Klasifikasi Kepadatan Lalu Lintas**

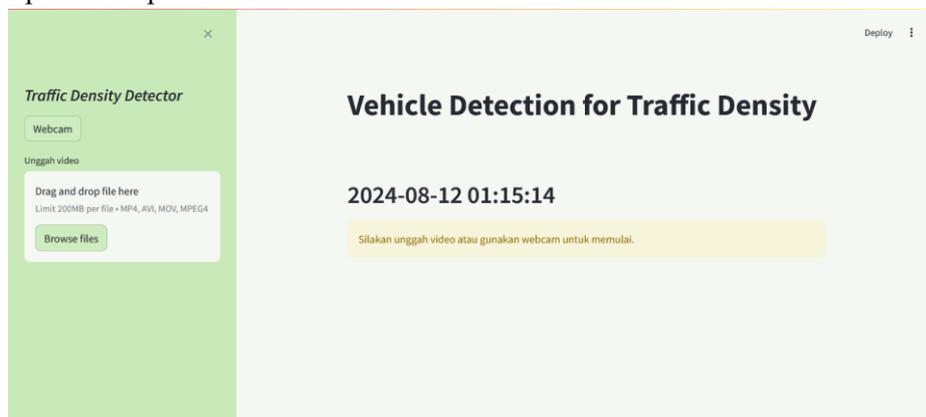
Tahapan	Deskripsi
Penentuan ROI	Wilayah deteksi dibatasi menggunakan Region of Interest (ROI) berdasarkan resolusi tampilan Streamlit (640 × 400 piksel).

**G. Implementasi Sistem (System Deployment/Usage)**

Pada bagian ini dijelaskan bagaimana sistem *Traffic Density Detector* diimplementasikan untuk mendeteksi dan mengklasifikasikan tingkat kepadatan lalu lintas kendaraan roda empat berbasis citra. Sistem dibangun menggunakan bahasa pemrograman Python dengan bantuan pustaka OpenCV, NumPy, dan framework antarmuka *Streamlit* yang berbasis web. Berikut beberapa tampilan antarmuka utama dari sistem yang telah dikembangkan:

1. Halaman Utama Sistem

Berikut merupakan tampilan halaman utama sistem.

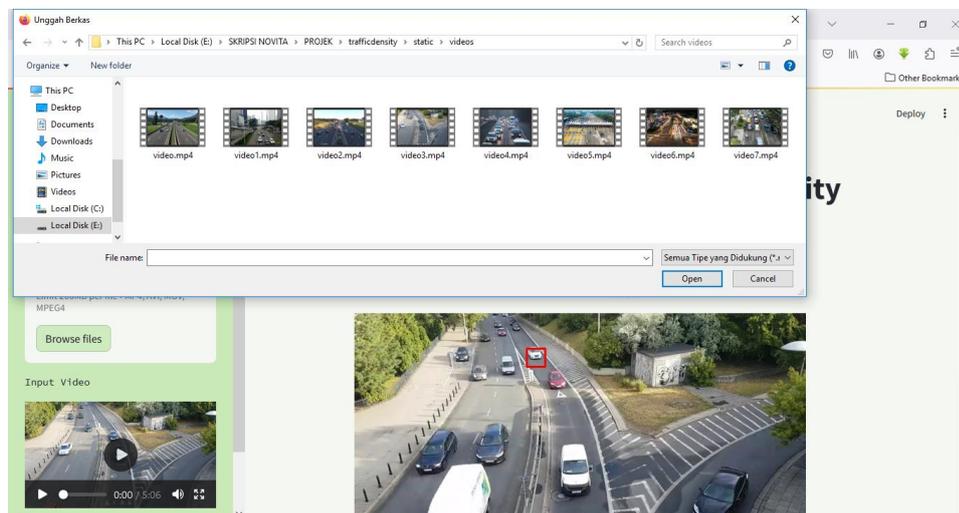


**Gambar 7. Halaman Utama Sistem**

- a. Terdapat tombol untuk mengunggah gambar lalu lintas.
- b. Opsi untuk menangkap gambar secara langsung dari kamera/webcam.

## 2. Tampilan Proses Upload Citra

Berikut merupakan tampilan proses upload citra.

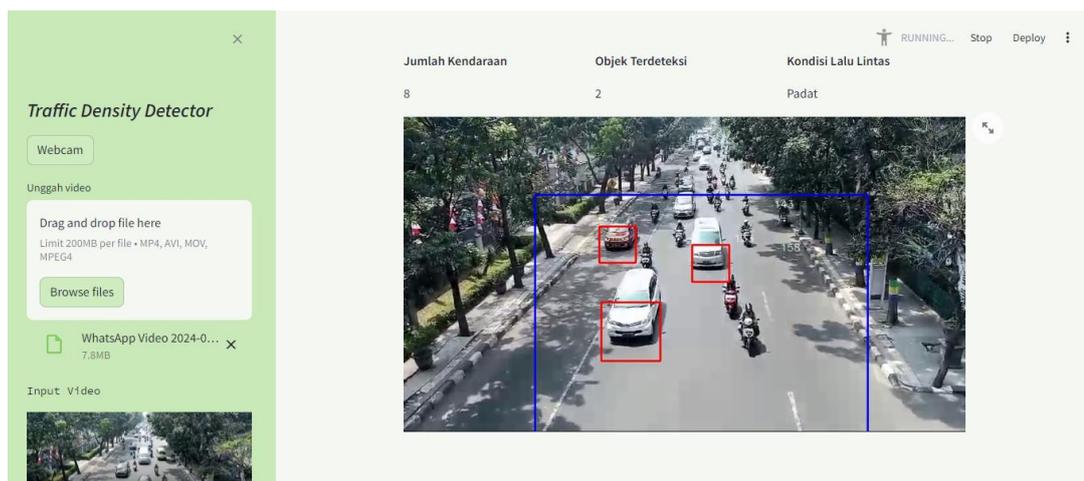


Gambar 8. Tampilan Proses Upload Citra

- a. Sistem menampilkan jendela pemilihan file dari direktori pengguna.
- b. Tersedia beberapa file video lalu lintas yang siap digunakan untuk uji deteksi kendaraan.
- c. Pratinjau video ditampilkan sebelum diproses.

## 3. Tampilan Proses Deteksi

Berikut adalah hasil visualisasi sistem saat mendeteksi kendaraan dalam kondisi lalu lintas padat.

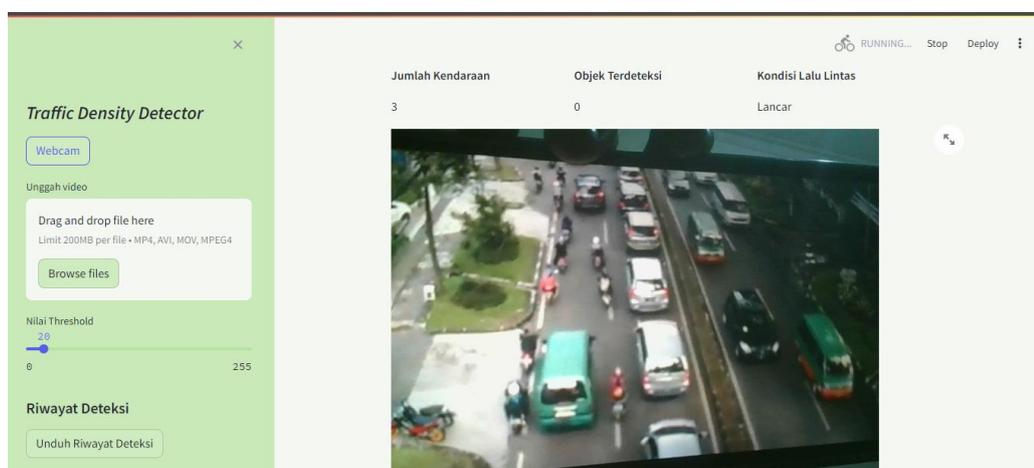


Gambar 9. Tampilan Proses Deteksi

- a. Sistem mendeteksi kendaraan dalam *Region of Interest* (ROI) yang dibatasi garis biru.
- b. Setiap kendaraan diberi *bounding box* merah sebagai hasil deteksi.
- c. Hasil klasifikasi menunjukkan kondisi lalu lintas padat, dengan jumlah kendaraan dan objek terdeteksi yang ditampilkan secara real-time.

#### 4. Tampilan Deteksi *Real-Time* via Webcam

Berikut adalah tampilan hasil klasifikasi sistem saat lalu lintas dalam keadaan lancar.



Gambar 10. Tampilan Deteksi Real-Time via Webcam

- Jika user ingin mendeteksi secara *real-time*, user dapat mengklik tombol *Webcam*.
- Sistem akan langsung terhubung dengan IP Camera dari perangkat yang digunakan.
- Deteksi dilakukan secara langsung dan hasil klasifikasi ditampilkan *real-time*.

#### 4. KESIMPULAN

Berdasarkan hasil penelitian dan implementasi sistem deteksi kepadatan kendaraan roda empat berbasis citra menggunakan metode *region-based* dan algoritma *Euclidean Distance*, dapat disimpulkan bahwa:

- Sistem yang dikembangkan mampu mendeteksi jumlah kendaraan dan menghitung tingkat kepadatan lalu lintas berdasarkan jarak antar kendaraan dalam sebuah citra, serta mengklasifikasikannya ke dalam kategori “lancar”, “sedang”, dan “padat” secara otomatis dan *real-time*.
- Sistem telah memenuhi kebutuhan fungsional dan non-fungsional sebagaimana diidentifikasi pada tahap analisis kebutuhan, termasuk proses unggah citra, pendeteksian objek kendaraan, perhitungan jarak menggunakan Euclidean Distance, serta visualisasi hasil klasifikasi kepadatan lalu lintas. Seluruh fitur telah diuji secara fungsional menggunakan metode black box dan memberikan hasil sesuai harapan.
- Hasil pengujian menunjukkan bahwa sistem mudah digunakan dan dapat dioperasikan melalui antarmuka web berbasis Streamlit, tanpa memerlukan perangkat keras tambahan. Sistem ini dinilai efektif sebagai alat bantu analisis visual kepadatan lalu lintas dan berpotensi untuk dikembangkan lebih lanjut dengan integrasi kamera real-time dan peningkatan akurasi melalui teknologi *deep learning*.

#### DAFTAR PUSTAKA

- Bjarnason, E., Lang, F., & Mjöberg, A. (2023). An empirically based model of software prototyping: a mapping study and a multi-case study. *Empirical Software Engineering*, 28(5), 115. <https://doi.org/10.1007/s10664-023-10331-w>
- Du, W., Dash, A., Li, J., Wei, H., & Wang, G. (2023). Safety in Traffic Management Systems: A Comprehensive Survey. *Designs*, 7(4), 100. <https://doi.org/10.3390/designs7040100>
- Gahremannezhad, H., Shi, H., & Liu, C. (2022). *Real-Time Accident Detection in Traffic Surveillance Using Deep Learning*. <https://doi.org/10.1109/IST55454.2022.9827736>
- Giasemis, F. I., & Sopsakis, A. (2025). *Learning Traffic Anomalies from Generative Models on Real-Time Observations*. <http://arxiv.org/abs/2502.01391>
- Hendrizal, S. (2023). *Penentuan Region Of Interest (ROI) Untuk Menghitung Jumlah Kendaraan Pada Jalan Raya Menggunakan Frame Substratcion*.

- Irawanto, M. A., Setianingsih, C., & Irawan, B. (2022). DETECTION OF TRAFFIC DENSITY WITH IMAGE PROCESSING USING PIN HOLE ALGORITHM. *IJUM Engineering Journal*, 23(1), 244–257. <https://doi.org/10.31436/iiumej.v23i1.2135>
- Khan, T. S., Pfoser, D., Ruan, S., & Züfle, A. (2024). Simplifying traffic simulation - from Euclidean distances to agent-based models. *Computational Urban Science*, 4(1). <https://doi.org/10.1007/s43762-024-00145-x>
- Oktavian, R. (2024). Pengembangan Sistem Pemetaan Kondisi Lalu Lintas Melalui Pengolahan Citra Digital Jalan. *Jurnal TICOM: Technology of Information and Communication*, 12(2), 70–74.
- Praniffa, A. C., Syahri, A., Sandes, F., Fariha, U., & Giansyah, Q. A. (2023). PENGUJIAN BLACK BOX DAN WHITE BOX SISTEM INFORMASI PARKIR BERBASIS WEB BLACK BOX AND WHITE BOX TESTING OF WEB-BASED PARKING INFORMATION SYSTEM. In *Jurnal Testing dan Implementasi Sistem Informasi* (Vol. 1, Issue 1).
- Raihan, H., & Voutama, A. (2023). Pengujian Black Box Pada Aplikasi Database Perguruan Tinggi dengan Teknik Equivalence Partition. *Antivirus: Jurnal Ilmiah Teknik Informatika*, 17(1), 1–18. <https://doi.org/10.35457/antivirus.v17i1.2501>
- Rezaei, M., Azarmi, M., & Mir, F. M. P. (2022). *Traffic-Net: 3D Traffic Monitoring Using a Single Camera*. <http://arxiv.org/abs/2109.09165>
- Solihun, S., Ruslianto, I., & Ristian, U. (2022). IMPLEMENTASI PERHITUNGAN KENDARAAN MOBIL DI JALAN RAYA MENGGUNAKAN METODE BACKGROUND SUBTRACTION DAN TEKNIK MORFOLOGI CITRA. *Coding Jurnal Komputer Dan Aplikasi*, 9(03), 424. <https://doi.org/10.26418/coding.v9i03.50865>
- Sriharyani, L., & Hadijah, I. (2023). KEPADATAN LALU LINTAS AKIBAT HAMBATAN SAMPING RUAS JALAN KI HAJAR DEWANTARA KOTA METRO. *TAPAK (Teknologi Aplikasi Konstruksi) : Jurnal Program Studi Teknik Sipil*, 12(2), 179. <https://doi.org/10.24127/tp.v12i2.2596>
- Sun, S., & Liu, M. (2025). A framework for detecting and predicting highway traffic anomalies via multimodal fusion and heterogeneous graph neural networks. *PLOS ONE*, 20(6 June). <https://doi.org/10.1371/journal.pone.0326313>
- Uswah, K. (2024). *Smart Traffic Light System Using Euclidean Distance Algorithm To Improve Traffic Efficiency In Small Villages*. <http://dx.doi.org/xx.xxxxx/jistel.vXiX>
- Wahyudi, D. A., & H. Kartowisastro, I. (2021). Menghitung Kecepatan Menggunakan Computer Vision. *Jurnal Teknik Komputer*, 19(2), 89–101.
- Wang, G., Jin, P., Qi, Z., & Li, X. (2025). Traffic sign detection method based on improved YOLOv8. *Scientific Reports*, 15(1). <https://doi.org/10.1038/s41598-025-03792-0>