

Perancangan Deteksi Objek Kendaraan Bermotor Berbasis OpenCV Python menggunakan Metode HOG-SVM untuk Analisis Lalu Lintas Cerdas

Eligius Transparan Putra Zebua^{1*}, Perani Rosyani²

^{1,2}Fakultas Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Jl. Raya Puspipetek No. 46, Kel. Buaran, Kec. Serpong, Kota Tangerang Selatan. Banten 15310, Indonesia

Email: ^{1*}terazebua0604@gmail.com, ²dosen00837@unpam.ac.id

(* : coressponding author)

Abstrak—Pada saat ini jumlah pengguna kendaraan bermotor semakin banyak sehingga sangat membutuhkan sistem Lalu Lintas Cerdas untuk dapat mengontrol pengguna jalan. Penelitian ini bertujuan untuk mengembangkan dan mengimplementasikan sistem deteksi objek kendaraan bermotor menggunakan OpenCV Python dengan metode HOG-SVM untuk analisis lalu lintas cerdas. Deteksi objek kendaraan bermotor adalah komponen penting dalam sistem manajemen lalu lintas modern yang memungkinkan pemantauan dan analisis real-time dari arus lalu lintas. Metodologi yang digunakan dalam penelitian ini melibatkan penggunaan Histogram of Oriented Gradients (HOG) dan Support Vector Machine (SVM) untuk mendeteksi kendaraan dalam citra dan video. Studi kasus ini menyoroti kelebihan dan kekurangan metode tersebut serta memberikan wawasan tentang aplikasi praktis dari teknologi ini dalam manajemen lalu lintas. Dengan penelitian menunjukkan bahwa implementasi deteksi kendaraan dengan metode HOG-SVM dapat memberikan akurasi yang baik dalam berbagai kondisi pencahayaan dan lalu lintas.

Kata Kunci: Deteksi Objek, Kendaraan Bermotor, OpenCV Python, HOG-SVM, Analisis Lalu Lintas, Sistem Manajemen Lalu Lintas

Abstract— Currently, the number of motorized vehicle users is increasing, so there is a real need for an Intelligent Traffic system to be able to control road users. This research aims to develop and implement a motor vehicle object detection system using OpenCV Python with the HOG-SVM method for intelligent traffic analysis. Motor vehicle object detection is a critical component in modern traffic management systems that enables real-time monitoring and analysis of traffic flows. The methodology used in this research involves the use of Histogram of Oriented Gradients (HOG) and Support Vector Machine (SVM) to detect vehicles in images and videos. This case study highlights the advantages and disadvantages of the method and provides insight into the practical application of this technology in traffic management. Research shows that implementing vehicle detection using the HOG-SVM method can provide good accuracy in various lighting and traffic conditions.

Keywords: Object Detection, Motor Vehicles, OpenCV Python, HOG-SVM, Traffic Analysis, Traffic Management Systems

1. PENDAHULUAN

Kendaraan bermotor adalah alat transportasi yang menggunakan mesin sebagai sumber tenaganya. Jenis kendaraan bermotor sangat bervariasi, mulai dari sepeda motor, mobil penumpang, bus, hingga truk. Pertumbuhan jumlah kendaraan bermotor di berbagai negara, terutama di daerah perkotaan, mengalami peningkatan yang signifikan setiap tahunnya. Hal ini tidak hanya menunjukkan peningkatan mobilitas dan aksesibilitas masyarakat, tetapi juga menimbulkan berbagai tantangan baru, khususnya dalam manajemen lalu lintas. Lalu lintas yang padat dapat menyebabkan berbagai masalah seperti kemacetan, polusi udara, dan peningkatan risiko kecelakaan. Oleh karena itu, penting untuk mengembangkan sistem analisis lalu lintas cerdas yang dapat membantu memantau, mengelola, dan mengoptimalkan lalu lintas secara efektif.

Deteksi objek adalah proses mengidentifikasi dan menemukan instansi objek tertentu dalam gambar atau video. Dalam konteks pengolahan citra dan visi komputer, deteksi objek melibatkan penggunaan algoritma dan teknik khusus untuk mengenali berbagai jenis objek, seperti manusia, kendaraan, hewan, dan lain-lain. Deteksi objek tidak hanya terbatas pada menemukan objek dalam gambar, tetapi juga mencakup pelacakan dan klasifikasi objek tersebut. Dalam berbagai aplikasi praktis, seperti pengawasan keamanan, analisis lalu lintas, dan sistem bantuan pengemudi, deteksi objek memainkan peran yang sangat penting. Keakuratan dan kecepatan deteksi objek merupakan

faktor krusial yang menentukan keberhasilan implementasi sistem-sistem ini. Perkembangan *machine learning* dan pendeteksi objek dapat dimanfaatkan untuk membantu peran manusia dalam melakukan pengawasan (NURYASIN et al., 2023)

Deteksi objek kendaraan bermotor adalah sub-kategori dari deteksi objek yang khusus berfokus pada identifikasi dan pelacakan kendaraan dalam gambar atau video. Metode ini sangat relevan untuk aplikasi analisis lalu lintas cerdas, di mana informasi tentang jenis, jumlah, dan pergerakan kendaraan dapat digunakan untuk mengoptimalkan aliran lalu lintas dan meningkatkan keselamatan jalan. Deteksi kendaraan bermotor dapat dilakukan menggunakan berbagai teknik dan algoritma, baik yang berbasis pembelajaran mesin (*machine learning*) maupun yang berbasis pembelajaran mendalam (*deep learning*). Tantangan utama dalam deteksi kendaraan termasuk variasi dalam jenis kendaraan, kondisi pencahayaan yang berubah-ubah, dan latar belakang yang kompleks.

Histogram of Oriented Gradients (HOG) dan Support Vector Machine (SVM) adalah kombinasi metode yang sering digunakan dalam deteksi objek. HOG adalah teknik pengolahan citra yang mengekstraksi fitur bentuk dan tepi dari sebuah gambar. Algoritma Histogram of Oriented Gradients (HOG) digunakan sebagai salah satu Solusi praktis yang memiliki efisiensi kerja dan keamanan yang baik dalam mendeteksi dan pengenalan wajah. (Laia et al., 2023). Teknik ini bekerja dengan membagi gambar menjadi sel-sel kecil, menghitung histogram orientasi gradien untuk setiap sel, dan kemudian menggabungkan hasilnya menjadi deskriptor fitur yang kaya informasi. HOG sangat efektif dalam menangkap pola visual yang penting untuk mendeteksi bentuk objek tertentu, seperti kendaraan.

Metode analisis sentiment arus utama diwakili oleh dukungan Mesin Vektor (Han et al., 2020). Support Vector Machine (SVM) adalah algoritma pembelajaran mesin yang digunakan untuk klasifikasi. SVM bekerja dengan mencari hyperplane optimal yang memisahkan data ke dalam kelas-kelas yang berbeda. Dalam konteks deteksi kendaraan, fitur-fitur yang diekstraksi oleh HOG digunakan sebagai input untuk SVM, yang kemudian melatih model untuk mengenali dan mengklasifikasikan objek sebagai kendaraan atau bukan. Kombinasi HOG-SVM ini terkenal karena keakuratannya dalam berbagai aplikasi deteksi objek, termasuk deteksi kendaraan bermotor, karena mampu menangani variasi dalam tampilan dan latar belakang dengan baik. SVM merupakan salah satu algoritma yang menghasilkan nilai akurasi baik (Irmanda & Ria Astriratma, 2020)

Dengan menggabungkan HOG dan SVM, sistem deteksi objek kendaraan bermotor berbasis OpenCV Python dapat dibangun untuk menyediakan solusi yang efektif dan efisien dalam analisis lalu lintas cerdas. Sistem ini tidak hanya meningkatkan pemahaman tentang kondisi lalu lintas saat ini, tetapi juga memungkinkan pengambilan keputusan yang lebih baik untuk manajemen lalu lintas dan peningkatan keselamatan di jalan raya. Salah satu cara untuk meningkatkan kecepatan adalah dengan menerapkan algoritme ini pada susunan gerbang yang dapat diprogram di lapangan, untuk mendapatkan manfaat dari desain fleksibel seperti komputasi paralel (Ghaffari et al., 2020)

OpenCV (Open Source Computer Vision Library) adalah perpustakaan perangkat lunak yang dirancang khusus untuk pengolahan citra dan visi komputer. Dikembangkan oleh Intel dan sekarang bersifat open-source, OpenCV menyediakan berbagai alat dan fungsi yang memudahkan pengembang untuk membuat aplikasi berbasis pengolahan citra dan visi komputer. Perangkat lunak pendeteksi gerakan lebih disukai sebagai salah satu fitur keamanan terpenting dalam beberapa hari terakhir. Ini digunakan untuk meningkatkan perangkat keamanan yang ada seperti pencahayaan sensor gerak pada kamera keamanan dalam dan luar ruangan (SHARMA, Ayushi, 2021). Python, sebagai salah satu bahasa pemrograman yang didukung oleh OpenCV, menjadi pilihan populer karena sintaksnya yang mudah dipahami dan kemampuannya untuk integrasi dengan berbagai pustaka lain, seperti NumPy, SciPy, dan scikit-learn. Dengan OpenCV Python, pengembang dapat mengakses berbagai algoritma canggih untuk tugas-tugas seperti deteksi objek, pelacakan, pengenalan wajah, dan segmentasi citra.

Salah satu aplikasi penting dari OpenCV Python adalah dalam deteksi objek kendaraan bermotor menggunakan metode Histogram of Oriented Gradients (HOG) dan Support Vector Machine (SVM). HOG adalah teknik deskriptor fitur yang digunakan untuk mengekstrak informasi tepi dan bentuk dari gambar. Dalam konteks deteksi kendaraan, HOG membantu dalam mengidentifikasi kontur dan struktur visual yang khas dari kendaraan bermotor. OpenCV

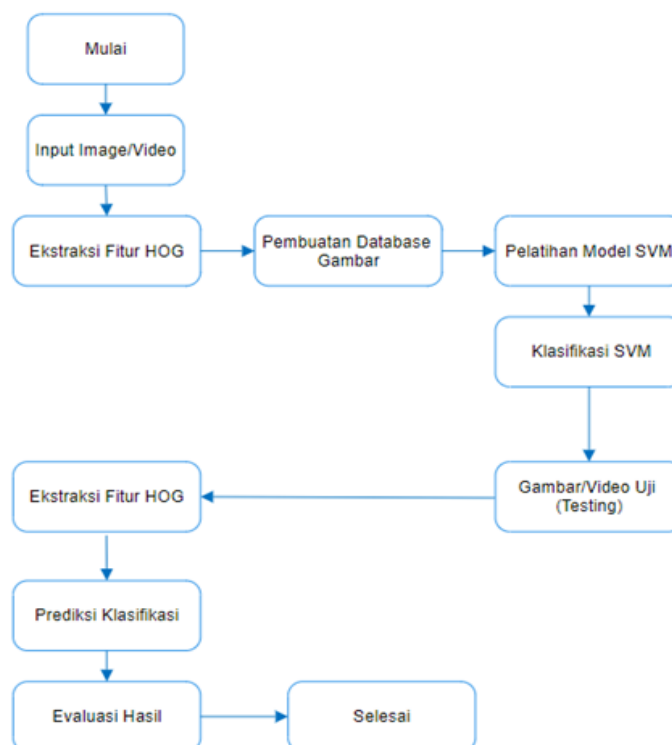
menyediakan fungsi untuk menghitung fitur HOG dari gambar, memudahkan proses ekstraksi fitur secara efisien.

Setelah fitur HOG diekstraksi, tahap berikutnya adalah menggunakan SVM untuk klasifikasi. SVM adalah algoritma pembelajaran mesin yang bekerja dengan mencari hyperplane optimal untuk memisahkan data ke dalam kelas-kelas yang berbeda. Dalam deteksi kendaraan, SVM dilatih menggunakan fitur HOG untuk membedakan antara kendaraan dan objek lainnya dalam gambar. Kombinasi HOG-SVM ini sangat efektif karena HOG mampu mengekstrak fitur yang signifikan dari gambar, sementara SVM menawarkan keakuratan yang tinggi dalam proses klasifikasi. OpenCV Python menyediakan implementasi SVM yang dapat diintegrasikan dengan mudah dengan deskriptor HOG, memungkinkan pengembangan sistem deteksi kendaraan yang kuat dan efisien.

Keterkaitan antara OpenCV Python dan metode HOG-SVM dalam deteksi kendaraan bermotor tidak hanya terbatas pada ketersediaan alat dan fungsi yang mendukung, tetapi juga mencakup kemudahan pengembangan dan penerapan aplikasi real-time. Dengan OpenCV Python, pengembang dapat memanfaatkan kekuatan pemrosesan citra HOG dan keakuratan klasifikasi SVM untuk membangun sistem analisis lalu lintas cerdas yang dapat beroperasi dalam berbagai kondisi lingkungan. Hal ini menjadikan OpenCV Python sebagai platform yang ideal untuk implementasi metode HOG-SVM dalam deteksi kendaraan bermotor, mendukung peningkatan efisiensi dan keamanan lalu lintas melalui solusi teknologi yang canggih.

2. METODOLOGI PENELITIAN

HOG-SVM adalah metode deteksi objek yang menggabungkan kekuatan deskripsi visual dari Histogram of Oriented Gradients (HOG) dengan kemampuan klasifikasi dari Support Vector Machine (SVM). HOG menganalisis perubahan intensitas cahaya pada citra untuk menghasilkan deskripsi numerik yang unik, sementara SVM menggunakan deskripsi ini untuk membedakan objek satu dengan lainnya. Kombinasi ini terbukti efektif dalam mendeteksi berbagai jenis objek dengan akurasi tinggi, bahkan dalam kondisi yang menantang. Implementasinya relatif mudah, menjadikannya pilihan menarik bagi peneliti di bidang visi komputer (XU, Panfeng; HUANG, Lidong; SONG, 2022).



Gambar 1. Flowchart Penelitian

2.1 Mulai

Ini adalah tahap inisiasi proses deteksi kendaraan bermotor menggunakan metode HOG-SVM. Pada tahap ini, semua persiapan untuk memulai proses sudah dilakukan, termasuk setup lingkungan kerja, mengimpor pustaka yang dibutuhkan (seperti OpenCV dan scikit-learn), dan memastikan data yang diperlukan sudah siap.

2.2 Input Image/Video

Pada tahap ini, gambar atau video yang berisi kendaraan bermotor diinput ke dalam sistem. Data ini akan digunakan sebagai bahan utama untuk proses deteksi. Gambar atau video bisa berasal dari berbagai sumber, seperti rekaman CCTV, kamera lalu lintas, atau dataset publik yang telah tersedia.

2.3 Ekstraksi Fitur HOG

Histogram of Oriented Gradients (HOG) adalah metode untuk mengekstraksi fitur visual penting dari gambar. Pada tahap ini, setiap gambar yang diinput akan dianalisis untuk menghitung vektor fitur HOG. Proses ini melibatkan pembagian gambar menjadi sel-sel kecil, menghitung histogram gradien arah untuk setiap sel, dan menggabungkannya untuk membentuk deskriptor fitur yang merepresentasikan kontur dan tepi objek dalam gambar.

2.4 Pembuatan Database Gambar

Fitur-fitur HOG yang telah diekstraksi dari gambar-gambar input kemudian digunakan untuk membuat database gambar. Database ini akan berisi vektor-vektor fitur HOG beserta label kelas masing-masing (misalnya, kendaraan atau bukan kendaraan). Database ini penting untuk tahap pelatihan model, karena menjadi dasar bagi algoritma pembelajaran mesin untuk belajar mengenali pola-pola yang ada.

2.5 Pelatihan Model SVM

Pada tahap ini, Support Vector Machine (SVM) dilatih menggunakan fitur HOG yang telah diekstraksi dan disimpan dalam database gambar. SVM adalah algoritma pembelajaran mesin yang sangat efektif untuk klasifikasi. Proses pelatihan ini melibatkan penggunaan data latih untuk menemukan hyperplane optimal yang memisahkan berbagai kelas objek dalam fitur ruang. Hasil dari tahap ini adalah model SVM yang terlatih, siap untuk digunakan dalam klasifikasi objek.

2.6 Klasifikasi SVM

Model SVM yang telah dilatih digunakan untuk mengklasifikasikan objek dalam gambar baru. Pada tahap ini, sistem akan mengaplikasikan model pada data uji untuk memprediksi kelas dari setiap objek yang terdeteksi. Klasifikasi ini berdasarkan vektor fitur HOG yang diekstraksi dari gambar uji.

2.7 Gambar/Video Uji (Testing)

Pada tahap ini, gambar atau video baru yang belum pernah dilihat oleh model digunakan untuk menguji kinerja model. Data uji ini penting untuk mengevaluasi seberapa baik model dapat menggeneralisasi dan mengklasifikasikan objek yang tidak ada dalam data latih.

2.8 Ekstraksi Fitur HOG (Testing)

Sama seperti pada tahap ekstraksi fitur sebelumnya, fitur HOG diekstraksi dari gambar atau video uji. Proses ini memastikan bahwa data uji juga direpresentasikan dalam bentuk vektor fitur HOG yang konsisten dengan data latih.

2.9 Prediksi Klasifikasi

Fitur HOG yang diekstraksi dari data uji kemudian dimasukkan ke dalam model SVM yang telah dilatih untuk memprediksi kelas dari setiap objek. Hasil prediksi ini menunjukkan apakah model dapat mengenali kendaraan dalam gambar uji dengan benar atau tidak.

2.10 Evaluasi Hasil

Tahap evaluasi melibatkan pengukuran kinerja model menggunakan metrik evaluasi seperti akurasi, precision, recall, F1-score, dan confusion matrix. Evaluasi ini memberikan gambaran tentang seberapa baik model dapat mendeteksi dan mengklasifikasikan objek kendaraan bermotor dalam kondisi nyata. Confusion matrix, khususnya, memberikan informasi detail tentang prediksi yang benar dan salah untuk setiap kelas.

3. ANALISA DAN PEMBAHASAN

3.1 Landasan Teori

3.1.1 Kendaraan

Kendaraan merupakan suatu perpanjangan dari kemampuan manusia untuk berpindah dan menjelajah. Dengan kendaraan manusia mampu mengatasi keterbatasan fisik dalam hal jarak dan kecepatan. Kendaraan tidak hanya berfungsi sebagai alat transportasi, tetapi juga mencerminkan perkembangan teknologi dan budaya suatu masyarakat. Desain, fungsi, dan penggunaan kendaraan terus berevolusi seiring dengan perubahan kebutuhan dan nilai-nilai manusia, mulai dari kereta kuda sederhana hingga mobil otonom canggih. Dengan demikian, kendaraan bukan hanya sekadar benda mati, melainkan bagian integral dari kehidupan manusia yang terus membentuk dan dipengaruhi oleh dinamika sosial, ekonomi, dan lingkungan (Ayuni Finda Rika & Nuzul Hikmah, 2021).

3.1.2 Lalu Lintas

Lalu lintas adalah sebuah tarian kompleks antara manusia, kendaraan, dan infrastruktur. Lebih dari sekadar pergerakan, lalu lintas mencerminkan interaksi sosial, ekonomi, dan lingkungan suatu masyarakat. Kelancaran dan keselamatan lalu lintas tidak hanya bergantung pada kepatuhan terhadap aturan, tetapi juga pada kesadaran bersama akan pentingnya berbagi ruang dan waktu. Lalu lintas yang efisien adalah cerminan dari perencanaan kota yang baik, teknologi transportasi yang maju, dan perilaku pengguna jalan yang bertanggung jawab. Sebaliknya, kemacetan dan kecelakaan lalu lintas adalah gejala dari ketidakseimbangan dalam sistem ini, yang dapat menimbulkan kerugian ekonomi, sosial, dan lingkungan (Iskandar Mulyana & Rofik, 2022).

3.1.3 Video

Video adalah jendela dinamis yang menangkap realitas dalam bingkai bergerak, merekam momen-momen berharga, dan menyampaikan informasi secara visual dan auditori. Lebih dari sekadar rangkaian gambar, video adalah medium ekspresi yang kuat, mampu membangkitkan emosi, menginspirasi tindakan, dan menghubungkan individu dari berbagai latar belakang. Video tidak hanya merepresentasikan dunia sebagaimana adanya, tetapi juga memungkinkan interpretasi, imajinasi, dan dialog. Dalam era digital, video telah menjadi bahasa universal yang melampaui batas geografis dan budaya, membentuk cara kita berkomunikasi, belajar, dan memahami dunia di sekitar kita.

3.1.4 Python

Python adalah bahasa pemrograman serbaguna yang mengutamakan kesederhanaan dan keterbacaan kode. Filosofi desainnya yang elegan memungkinkan pengembang, baik pemula maupun ahli, untuk mengekspresikan ide-ide kompleks dengan sintaksis yang ringkas dan intuitif. Python tidak hanya alat untuk membangun perangkat lunak, tetapi juga sarana untuk memecahkan masalah, mengotomatiskan tugas, dan menjelajahi dunia data. Fleksibilitasnya yang luar biasa menjadikannya pilihan ideal untuk berbagai aplikasi, mulai dari pengembangan web dan analisis data hingga kecerdasan buatan dan pembelajaran mesin. Dengan komunitas yang aktif dan dukungan pustaka yang luas, Python terus berkembang sebagai bahasa pemrograman yang relevan dan berpengaruh di era digital.

3.1.5 Metode HOG-SVM

Metode HOG-SVM (Histogram of Oriented Gradients - Support Vector Machine) adalah pendekatan ampuh dalam mendeteksi kendaraan pada citra digital. HOG berperan sebagai ekstraktor

fitur visual, mengubah gambar menjadi representasi numerik berdasarkan distribusi orientasi gradien. Fitur-fitur ini kemudian menjadi input bagi SVM, sebuah algoritma pembelajaran mesin yang mampu mengklasifikasikan objek berdasarkan pola-pola yang dipelajari dari data pelatihan. Keunggulan HOG-SVM terletak pada kemampuannya mengenali bentuk kendaraan yang khas, bahkan dalam kondisi pencahayaan dan sudut pandang yang bervariasi. Kombinasi antara deskripsi fitur yang kuat dan kemampuan klasifikasi yang akurat menjadikan HOG-SVM sebagai salah satu metode yang handal dan populer dalam bidang pengolahan citra dan visi komputer.

3.2 Analisa Sistem

Analisa sistem adalah proses yang komprehensif dan sistematis untuk mengevaluasi suatu sistem guna memahami bagaimana sistem tersebut beroperasi dan untuk mengidentifikasi kebutuhan atau masalah yang harus diselesaikan. Dalam konteks penelitian "Perancangan Deteksi Objek Kendaraan Bermotor Berbasis OpenCV Python menggunakan Metode HOG-SVM untuk Analisis Lalu Lintas Cerdas", analisa sistem bertujuan untuk merancang sistem deteksi objek kendaraan bermotor yang efektif. Langkah pertama dalam analisa sistem adalah memahami tujuan utama dari sistem, yaitu mendeteksi dan mengklasifikasikan kendaraan bermotor dalam rekaman video atau gambar lalu lintas secara akurat dan efisien. Deteksi objek merupakan salah satu tugas penting dalam bidang visi komputer yang bertujuan untuk mengidentifikasi dan menentukan lokasi objek dalam citra atau video (Nafis Alfarizi et al., 2023).

Setelah tujuan ditetapkan, langkah berikutnya dalam analisa sistem adalah mengidentifikasi komponen utama dari sistem. Ini mencakup modul ekstraksi fitur menggunakan Histogram of Oriented Gradients (HOG), algoritma klasifikasi dengan Support Vector Machine (SVM), serta integrasi dan pengujian menggunakan OpenCV Python. Setiap komponen dianalisis secara mendalam untuk memastikan bahwa mereka bekerja bersama secara harmonis dan efektif dalam mencapai tujuan sistem. Pengumpulan data lalu lintas, pengkajian metode deteksi objek yang sudah ada, dan pemetaan kebutuhan teknis serta fungsional dari sistem menjadi bagian penting dalam tahap ini.

3.3 Evaluasi Akurasi dan Efisiensi

Melalui analisa sistem yang teliti, dapat ditemukan berbagai cara untuk meningkatkan akurasi dan efisiensi deteksi kendaraan bermotor. Analisa ini tidak hanya membantu dalam perancangan sistem yang lebih baik, tetapi juga memastikan bahwa solusi yang dikembangkan benar-benar memenuhi kebutuhan pengguna dan dapat mengatasi tantangan di lapangan. Hasil dari analisa sistem ini menjadi dasar dalam proses perancangan dan pengembangan, yang pada akhirnya menghasilkan sistem deteksi kendaraan bermotor yang cerdas dan efektif untuk analisis lalu lintas cerdas. Dengan demikian, analisa sistem berfungsi sebagai fondasi yang kuat untuk menciptakan solusi teknologi yang inovatif dan bermanfaat dalam pengelolaan lalu lintas (Damarcha, 2021).

4. IMPLEMENTASI

4.1 Spesifikasi Kebutuhan

Tahap ini merupakan hal yang paling penting dalam menjalankan suatu studi kasus atau dalam mengembangkan suatu sistem. Dalam hal ini, semua kebutuhan dan perlengkapan harus terpenuhi.

4.2 Spesifikasi Perangkat Keras (*Hardware*)

Tabel 1. Spesifikasi Kebutuhan Perangkat Keras

Komponen	Spesifikasi
Laptop	MSI Modern 14
Processor	Intel Core i3
Penyimpanan	SSD Kingston 512GB
Ram	8GB

4.3 Spesifikasi Perangkat Lunak (*Software*)

Tabel 2. Spesifikasi Kebutuhan Perangkat Lunak

Komponen	Spesifikasi
Sistem Operasi	Windows 11
Code Editor	Visual Code Studio
Application Design	SmartDraw
Bahasa Pemrograman	Python
Library	OpenCV

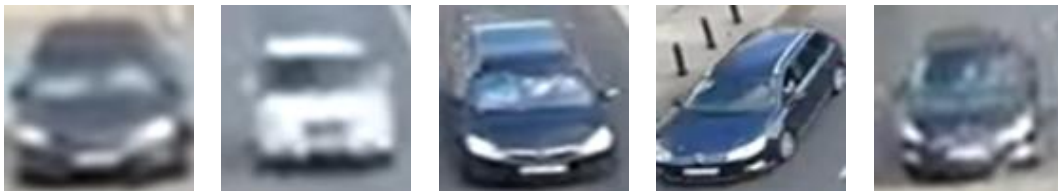
4.4 Video Rekaman Lalu Lintas

Dalam melakukan implementasi ini dibutuhkan sebuah video mp4 yang berdurasi lebih kurang 5 menit yang menampilkan arus lalu lintas dan kendaraan. Dengan video ini kita akan melakukan membaca dan memproses frame.

4.5 Sampel Deteksi

Deteksi objek merupakan salah satu tugas penting dalam bidang visi komputer yang bertujuan untuk mengidentifikasi dan menentukan lokasi objek dalam citra atau video (Nafis Alfarizi et al., 2023). Langkah awal yang harus dilakukan adalah kita harus menyiapkan sampel gambar-gambar objek dimana kita akan menyiapkan dua folder untuk menyimpan gambar. Gambar tersebut terdiri dari dua jenis objek yaitu gambar kendaraan (positif) dan gambar non-kendaraan (negative). Penyiapan gambar ini berfungsi untuk melakukan pelatihan Model SVM. Jadi kita perlu mengumpulkan dataset. Untuk pengambilan gambar objeknya kita bisa melakukan dengan screenshot objek pada video yang telah ada. Berikut contoh gambar objek untuk menyiapkan dataset Model SVM.

4.5.1 Gambar Objek Kendaraan



Gambar 2. Objek Kendaraan

4.5.2 Gambar Objek Non-Kendaraan



Gambar 3. Objek Non-Kendaraan

4.6 Model SVM

Setelah Objek Kendaraan dan Non-Kendaraan sudah disiapkan, selanjutnya kita akan melakukan pelatihan pendeteksi. Pada proses ini kita akan membuat sebuah dataset untuk menyimpan data objek yang akan disimpan dalam bentuk .joblib dimana dengan dataset ini proses pendeteksi kendaraan akan bisa dilakukan dengan membedakan kendaraan dengan non-kendaraan. Berikut skrip untuk menyiapkan dataset.

```
1 import cv2
2 import numpy as np
3 import os
4 from skimage.feature import hog
5 from sklearn.svm import LinearSVC
6 from joblib import dump
7
8 # Fungsi untuk membaca gambar dari folder dan mengubah ukurannya
9 def load_images_from_folder(folder, size=(64, 64)):
10     images = []
11     for filename in os.listdir(folder):
12         img = cv2.imread(os.path.join(folder, filename))
13         if img is not None:
14             img = cv2.resize(img, size)
15             images.append(img)
16     return images
17
18 # Fungsi untuk mengekstrak fitur HOG dari gambar
19 def get_hog_features(image):
20     image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY) # Ubah gambar menjadi grayscale
21     features, _ = hog(image, pixels_per_cell=(8, 8),
22                      cells_per_block=(2, 2), visualize=True)
23     return features
24
25 # Fungsi untuk melatih model SVM
26 def train_svm(positive_folder, negative_folder):
27     positive_images = load_images_from_folder(positive_folder)
28     negative_images = load_images_from_folder(negative_folder)
29
30     X = []
31     y = []
32
33     for img in positive_images:
34         X.append(get_hog_features(img))
35         y.append(1)
36
37     for img in negative_images:
38         X.append(get_hog_features(img))
39         y.append(0)
40
41     X = np.array(X)
42     y = np.array(y)
43
44     svm = LinearSVC()
45     svm.fit(X, y)
46
47     dump(svm, 'vehicle_detector_svm.joblib')
48     print("Model SVM telah dilatih dan disimpan.")
49
50 # Jalankan pelatihan model
51 if __name__ == "__main__":
52     positive_folder = 'positives'
53     negative_folder = 'negatives'
54     train_svm(positive_folder, negative_folder)
```

Gambar 4. Skrip Model SVM

4.7 Deteksi Objek

Setelah mendapatkan dataset model SVM, selanjut kita membuat skrip untuk mendeteksi objek sebagai berikut:

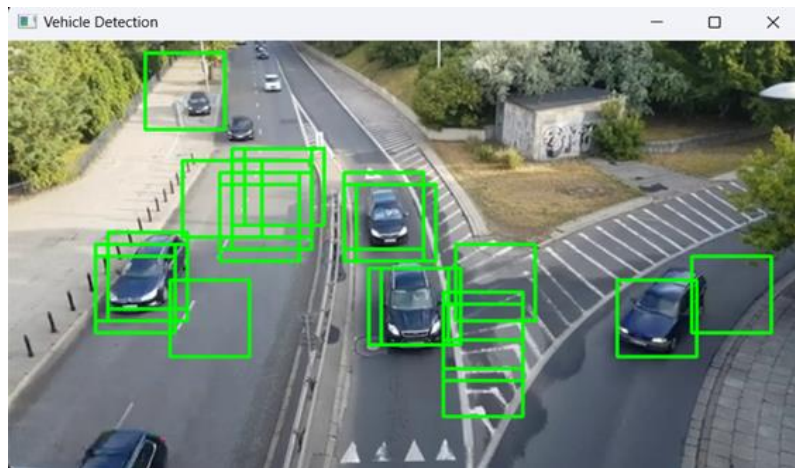
```

1 import cv2
2 import numpy as np
3 from skimage.feature import hog
4 from joblib import load
5
6 # Fungsi untuk mengekstrak fitur HOG dari gambar
7 def get_hog_features(image):
8     image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY) # Ubah gambar menjadi grayscale
9     features, _ = hog(image, pixels_per_cell=(8, 8),
10                      cells_per_block=(2, 2), visualize=True)
11     return features
12
13 # Fungsi untuk mendeteksi kendaraan di frame menggunakan model SVM
14 def detect_vehicles(frame, svm):
15     detected_vehicles = []
16     window_size = (64, 64)
17     step_size = 10
18
19     for y in range(0, frame.shape[0] - window_size[1], step_size):
20         for x in range(0, frame.shape[1] - window_size[0], step_size):
21             window = frame[y:y + window_size[1], x:x + window_size[0]]
22             if window.shape[0] != 64 or window.shape[1] != 64:
23                 continue
24
25             features = get_hog_features(window).reshape(1, -1)
26             prediction = svm.predict(features)
27
28             if prediction == 1:
29                 detected_vehicles.append((x, y, x + 64, y + 64))
30
31     return detected_vehicles
32
33 # Fungsi utama untuk membaca video dan melakukan deteksi kendaraan
34 def main(video_path):
35     svm = load('vehicle_detector_svm.joblib')
36
37     cap = cv2.VideoCapture(video_path)
38     while cap.isOpened():
39         ret, frame = cap.read()
40         if not ret:
41             break
42
43         detected_vehicles = detect_vehicles(frame, svm)
44
45         for (x1, y1, x2, y2) in detected_vehicles:
46             cv2.rectangle(frame, (x1, y1), (x2, y2), (0, 255, 0), 2)
47
48         cv2.imshow('Vehicle Detection', frame)
49
50         if cv2.waitKey(1) & 0xFF == ord('q'):
51             break
52
53     cap.release()
54     cv2.destroyAllWindows()
55
56 # Jalankan program
57 if __name__ == "__main__":
58     video_path = 'D:/Bahan Kuliah/Jurnal/tst/Road traffic.mp4'
59     main(video_path)
    
```

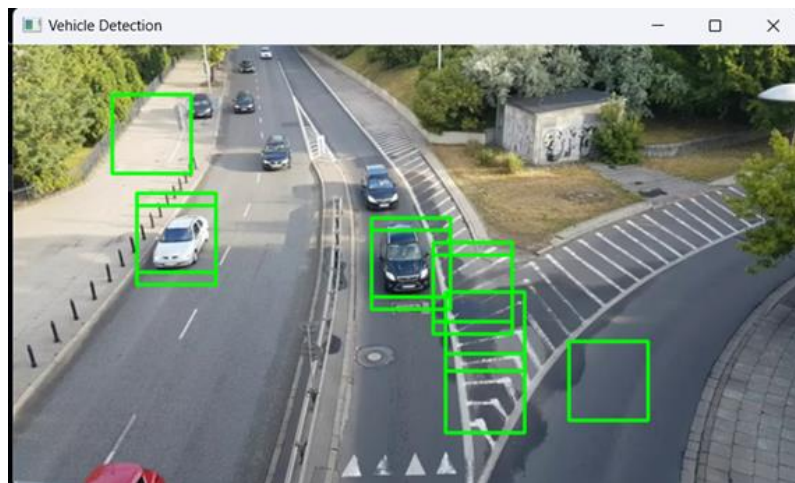
Gambar 5. Skrip Deteksi Objek

4.8 Hasil

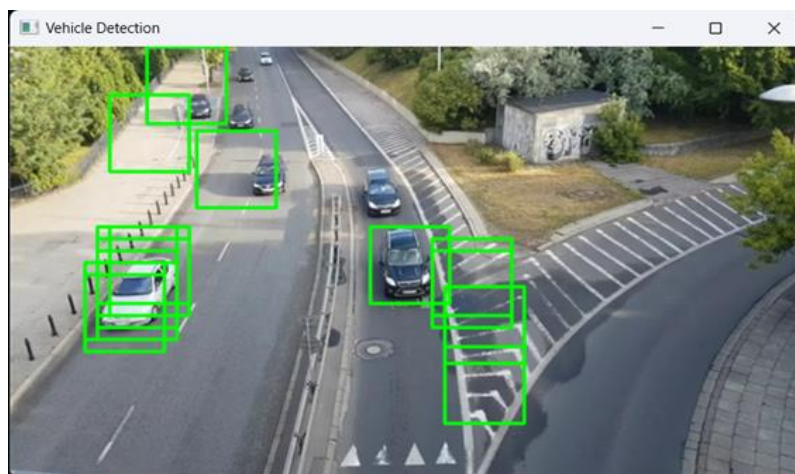
Setelah melakukan testing untuk mendeteksi objek kendaraan dan non kendaraan. Maka kita mendapatkan hasil seperti pada gambar berikut:



Gambar 6. Hasil Deteksi Objek 1



Gambar 7. Hasil Deteksi Objek 2



Gambar 8. Hasil Deteksi Objek 3

4. KESIMPULAN

Berdasarkan hasil penelitian dan analisis yang telah dilakukan, dapat disimpulkan bahwa metode Histogram of Oriented Gradients (HOG) yang dikombinasikan dengan Support Vector Machine (SVM) pada OpenCV Python merupakan pendekatan yang efektif dan akurat dalam mendeteksi kendaraan bermotor. Kinerja yang stabil dalam berbagai kondisi lingkungan, termasuk perubahan pencahayaan dan skala objek, menjadikan metode ini sangat menjanjikan untuk diterapkan pada sistem manajemen lalu lintas cerdas (Intelligent Transportation System/ITS).

Selain ITS, potensi penerapan teknologi deteksi objek berbasis HOG-SVM ini juga sangat luas, mencakup pemantauan keamanan, analisis video cerdas, dan berbagai aplikasi lain yang membutuhkan identifikasi kendaraan secara otomatis dan akurat. Kemampuannya dalam mendeteksi kendaraan dengan tingkat akurasi yang tinggi dapat memberikan kontribusi signifikan dalam meningkatkan efisiensi, keamanan, dan kenyamanan dalam berbagai sektor kehidupan.

Penelitian lebih lanjut dapat diarahkan pada pengembangan metode HOG-SVM yang lebih canggih, seperti optimasi parameter, penggunaan dataset yang lebih besar dan beragam, serta integrasi dengan teknologi lain seperti deep learning. Dengan demikian, diharapkan teknologi deteksi objek berbasis HOG-SVM dapat terus berkembang dan memberikan manfaat yang lebih besar bagi masyarakat.

REFERENCES

- Ayuni Finda Rika, & Nuzul Hikmah. (2021). Rancang Bangun Alat Pendeteksi Kendaraan Dari Arah Berlawanan Pada Tikungan Tajam Berbasis Arduino UNO. *Jurnal JEETech*, 2(1), 34–38. <https://doi.org/10.48056/jeetech.v2i1.158>
- Damarcha, A. N. (2021). *Contrast Stretching To Enhance Parking Lot Detection At Night Based on Convolutional Neural Network (Cnn) and Support ...* 2021. <https://etd.repository.ugm.ac.id/penelitian/detail/202514>
- Ghaffari, S., Soleimani, P., Li, K. F., & Capson, D. W. (2020). Analysis and Comparison of FPGA-Based Histogram of Oriented Gradients Implementations. *IEEE Access*, 8, 79920–79934. <https://doi.org/10.1109/ACCESS.2020.2989267>
- Han, K. X., Chien, W., Chiu, C. C., & Cheng, Y. T. (2020). Application of support vector machine (SVM) in the sentiment analysis of twitter dataset. *Applied Sciences (Switzerland)*, 10(3). <https://doi.org/10.3390/app10031125>
- Irmanda, H. N., & Ria Astriratma. (2020). Klasifikasi Jenis Pantun Dengan Metode Support Vector Machines (SVM). *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 4(5), 915–922. <https://doi.org/10.29207/resti.v4i5.2313>
- Iskandar Mulyana, D., & Rofik, M. A. (2022). Implementasi Deteksi Real Time Klasifikasi Jenis Kendaraan Di Indonesia Menggunakan Metode YOLOV5. *Jurnal Pendidikan Tambusai*, 6(3), 13971–13982. <https://doi.org/10.31004/jptam.v6i3.4825>
- Laia, F. H., Rosnelly, R., Naswar, A., Buulolo, K., & Christin Lase, M. (2023). Deteksi Pengenalan Wajah Orang Berbasis AI Computer Vision. *Teknologi Informasi Mura*, 15(1), 61–71.
- Nafis Alfarizi, D., Agung Pangestu, R., Aditya, D., Adi Setiawan, M., & Rosyani, P. (2023). Penggunaan Metode YOLO Pada Deteksi Objek: Sebuah Tinjauan Literatur Sistematis. *Jurnal Artificial Intelligent Dan Sistem Penunjang Keputusan*, 1(1), 54–63. <https://jurnalmahasiswa.com/index.php/aidanspk>
- NURYASIN, M. F., MACHBUB, C., & YULIANTI, L. (2023). Kombinasi Deteksi Objek, Pengenalan Wajah dan Perilaku Anomali menggunakan State Machine untuk Kamera Pengawas. *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 11(1), 86. <https://doi.org/10.26760/elkomika.v11i1.86>
- SHARMA, Ayushi, et al. (2021). *Object detection using OpenCV and python*. <https://ieeexplore.ieee.org/abstract/document/9725638>
- XU, Panfeng; HUANG, Lidong; SONG, Y. (2022). *An optimal method based on HOG-SVM for fault detection*. <https://link.springer.com/article/10.1007/s11042-022-12020-0>