

MONOGRAF

Penerbit Yayasan Peneliti Teknologi Teknik Indonesia

OPTIMASI PREDIKSI ARUS LALU LINTAS PERKOTAAN MENGGUNAKAN MODEL LSTM-KMA

IMAM AHMAD ASHARI
WAHYUL AMIEN SYAFEI
ADI WIBOWO



Optimasi Prediksi Arus Lalu Lintas Perkotaan Menggunakan Model LSTM-KMA

Penulis:

Imam Ahmad Ashari, S.Kom., M.Kom.
Dr. Ir. Wahyul Amien Syafei, S.T., M.T., IPM.
Dr. Eng. Adi Wibowo, S.Si., M.Kom

Penerbit Yayasan Peneliti Teknologi Teknik Indonesia

Optimasi Prediksi Arus Lalu Lintas Perkotaan Menggunakan Model LSTM-KMA

Penulis:

Imam Ahmad Ashari, S.Kom., M.Kom.
Dr. Ir. Wahyul Amien Syafei, S.T., M.T., IPM.
Dr. Eng. Adi Wibowo, S.Si., M.Kom.

Editor:

Soiful Hadi, S.T, M.Kom.

Desain Cover:

Imam Ahmad Ashari, S.Kom., M.Kom.

Tata Letak:

Ir. Purwono, S.Kom., M.Kom.

Ukuran:

UNESCO 15 x 23 cm

ISBN:

Sedang Proses Penerbitan

Cetakan Pertama:

Agustus 2025

Diterbitkan oleh

Penerbit Yayasan Peneliti Teknologi Teknik Indonesia

Jl. Empu Sedah No. 12, Pringwulung, Condongcatut, Kec. Depok,

Kabupaten Sleman, Daerah Istimewa Yogyakarta 55281, Indonesia

Hak Cipta 2025, Pada Penulis

Dilarang keras menerjemahkan, memfotokopi, atau memperbanyak sebagian atau seluruh isi buku ini tanpa izin tertulis dari Penerbit.

UU No 28 tahun 2014 tentang Hak Cipta

Fungsi dan sifat hak cipta Pasal 4
Hak Cipta sebagaimana dimaksud dalam Pasal 3 huruf a merupakan hak eksklusif yang terdiri atas hak moral dan hak ekonomi.

Sanksi Pelanggaran Pasal 113

1. Setiap Orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf i untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan/atau pidana denda paling banyak **Rp100.000.000** (seratus juta rupiah).
2. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf c, huruf d, huruf f, dan/atau huruf i, untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 3 (tiga) tahun dan/atau denda paling banyak **Rp500.000.000** (lima ratus juta rupiah).

Optimasi Prediksi Arus Lalu Lintas Perkotaan Menggunakan Model LSTM-KMA

Oleh:

Imam Ahmad Ashari, S.Kom., M.Kom.

Dr. Ir. Wahyul Amien Syafei, S.T., M.T., IPM

Dr.Eng. Adi Wibowo, S.Si., M.Kom.

Hak Cipta © 2025 pada penulis,

Editor: Soiful Hadi, S.T, M.Kom.

Hak Cipta dilindungi oleh undang-undang. Dilarang memperbanyak atau memindahkan Sebagian atau keseluruhan isi buku ini dalam bentuk apapun, secara elektronis maupun mekanis, termasuk mefotokopi, merekam, atau dengan Teknik perekaman lainnya, tanpa izin tertulis dari penerbit.

Diterbitkan oleh

Penerbit Yayasan Peneliti Teknologi Teknik Indonesia

Jl. Empu Sedah No. 12, Pringwulung, Condongcatur, Kec. Depok,
Kabupaten Sleman, Daerah Istimewa Yogyakarta 55281, Indonesia

Imam Ahmad Ashari, S.Kom., M.Kom., Dr. Ir. Wahyul Amien
Syafei, S.T., M.T., IPM., Dr.Eng. Adi Wibowo, S.Si., M.Kom.

Optimasi Prediksi Arus Lalu Lintas Perkotaan Menggunakan
Model LSTM-KMA

-edisi Pertama – Purwokerto: Penerbit Yayasan Peneliti
Teknologi Teknik Indonesia, 2025

LXXIX + 79, 15x23 cm

ISBN: In Process

DAFTAR ISI

DAFTAR ISI	v
DAFTAR TABEL	vii
DAFTAR GAMBAR	vii
KATA PENGANTAR	ix
RINGKASAN	x
BAB I Pendahuluan	1
1.1. Latar Belakang	1
1.2. Transportasi Cerdas.....	5
1.3. Peran Computer Vision dalam Transportasi Cerdas	8
1.4. Penerapan Deep Learning untuk Prediksi Lalu Lintas	10
1.5. Algoritma Metaheuristik dalam Optimasi	12
BAB II Model LSTM untuk Data Temporal	15
2.1. Konsep Dasar Long Short-Term Memory (LSTM).....	15
2.2. Arsitektur LSTM.....	16
2.3. Keunggulan LSTM dalam Analisis Data Temporal.....	18
2.4. Aplikasi LSTM untuk Prediksi Lalu Lintas.....	20
BAB III Komodo Mlipir Algorithm (KMA) sebagai Teknik Optimasi ...	22
3.1. Pengenalan Komodo Mlipir Algorithm	22
3.2. Konsep Komodo Mlipir Algorithm.....	25
3.3. Representasi Individu, Kelompok, dan Mekanisme Evolusi dalam KMA	26
3.4. Perbedaan KMA dibanding Algoritma Metaheuristik Lain	29
BAB IV Metodologi Pendekatan LSTM-KMA	31
4.1. Proses Data Lalu Lintas	31
4.2. Penggunaan YOLO untuk Deteksi Kendaraan	35
4.3. Integrasi LSTM dengan KMA	43

BAB V Eksperimen dan Hasil Penelitian.....48

5.1. Parameter untuk Model LSTM yang Dioptimalkan48

5.2. Analisis Hasil Evaluasi (RMSE, MAE).....49

5.3. Perbandingan dengan Model Lain51

BAB VI Kesimpulan dan Rekomendasi54

6.1. Kesimpulan54

6.2. Kontribusi Penelitian55

DAFTAR PUSTAKA.....56

DAFTAR TABEL

Tabel 1. Nilai Augmentasi 35

Tabel 2. Perbandingan Metrik Kinerja Model yang Diuji 38

Tabel 3. Kelas Metrik Evaluasi dari Yolov8n menggunakan Multi-Augmentasi 41

Tabel 4. Pengaturan Parameter dari Algoritma yang diuji..... 48

Tabel 5. Nilai RMSE dan MAE dari Masing-Masing Model yang Dievaluasi Secara Individual 51

DAFTAR GAMBAR

Gambar 1. Arsitektur Big Data Tiga Lapisan.....	6
Gambar 2. Sistem deteksi perilaku abnormal kota pintar	9
Gambar 3. Skema LSTM	16
Gambar 4. Arsitektur LSTM	17
Gambar 5. Konsep Algoritma Komodo Mlipir	22
Gambar 6. Tangkapan gambar CCTV di lampu lalu lintas Fatmawati, Kota Semarang.....	32
Gambar 7. Alur kerja deteksi kendaraan menggunakan YOLO	36
Gambar 8. Confusion Matrix	43
Gambar 9. Intergrasi LSTM-KMA	44
Gambar 10. Hasil prediksi dari masing-masing model	51
Gambar 11. Hasil evaluasi RMSE.....	52
Gambar 12. Hasil evaluasi MAE.....	52

KATA PENGANTAR

Assalamualaikum warahmatullahi wabarakatuh

Puji syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa atas terselesaikannya buku ini yang berjudul "Optimasi Prediksi Arus Lalu Lintas Perkotaan Menggunakan Model LSTM-KMA". Buku ini membahas solusi untuk menghadapi tantangan prediksi arus lalu lintas di perkotaan melalui integrasi model Long Short-Term Memory (LSTM) dan Komodo Mlipir Algorithm (KMA). Pendekatan ini dirancang untuk mendeteksi kendaraan secara akurat, bahkan dalam kondisi pencahayaan buruk dan ukuran objek yang kecil. Kami berharap buku ini dapat memberikan manfaat bagi akademisi, praktisi, dan pembuat kebijakan di bidang transportasi, terutama dalam meningkatkan efisiensi manajemen lalu lintas perkotaan. Terima kasih kami sampaikan kepada semua pihak yang mendukung penyusunan buku ini. Kami menyadari buku ini belum sempurna, sehingga saran dan kritik yang membangun sangat kami harapkan untuk penyempurnaan di masa mendatang.

Wassalamualaikum warahmatullahi wabarakatuh.

Purwokerto, Agustus 2025

Penulis

RINGKASAN

Buku ini menawarkan solusi untuk menghadapi tantangan deteksi multi-target kendaraan di lalu lintas perkotaan, seperti pencahayaan buruk, ukuran objek kecil, dan keragaman jenis kendaraan. Menggunakan model Long Short-Term Memory yang dioptimalkan dengan Komodo Mlipir Algorithm (LSTM-KMA), pendekatan ini bertujuan meningkatkan akurasi prediksi arus lalu lintas.

Data video lalu lintas diproses menggunakan YOLO untuk klasifikasi kendaraan dan perhitungan objek, sementara model LSTM dilatih untuk mengenali pola lalu lintas dengan parameter yang dioptimalkan oleh KMA. Algoritma ini memanfaatkan strategi mutasi dan crossover untuk memastikan seleksi adaptif dalam pencarian solusi global dan lokal. Melalui evaluasi pada dataset lalu lintas perkotaan dengan konfigurasi seragam, LSTM-KMA mengungguli model LSTM konvensional serta yang dioptimalkan dengan IDBO dan PSO.

Buku ini menyajikan model yang mampu meningkatkan efisiensi sistem transportasi perkotaan secara nyata. Dengan pendekatan yang andal, LSTM-KMA mendukung pengelolaan lalu lintas secara real-time, menjawab kebutuhan lingkungan perkotaan yang semakin dinamis dan kompleks.

BAB I

Pendahuluan

1.1. Latar Belakang

Dengan kemajuan teknologi komunikasi dan ilmu komputer, Sistem Transportasi Cerdas (ITS) telah mengambil peran yang semakin penting dalam kehidupan sehari-hari (W. Zhang et al. 2024). Transportasi pintar menjadi pilar utama dalam pengembangan sistem transportasi cerdas berbasis teknologi untuk mendukung kebutuhan masyarakat urban yang terus berkembang (Oladimeji et al. 2023). Transportasi cerdas mengacu pada pendekatan yang mengintegrasikan teknologi modern ke dalam sistem transportasi untuk meningkatkan efisiensi mobilitas perkotaan (Almatar 2024). Dalam konteks kota pintar, teknologi mutakhir seperti Internet of Things (IoT), analisis data, dan kecerdasan buatan menjadi fondasi utama untuk menciptakan ekosistem transportasi yang cerdas dan saling terhubung (Khan et al. 2024). Mobilitas cerdas kini menjadi bagian integral dari kehidupan sehari-hari, di mana 40% dari total populasi melakukan perjalanan setidaknya satu jam setiap hari (Jalil et al. 2024). Dengan mengintegrasikan teknologi seperti computer vision, kecerdasan buatan (AI), dan sistem transportasi cerdas (ITS), kota-kota dapat mendeteksi kondisi lalu lintas, mengidentifikasi jenis kendaraan, dan memprediksi kemacetan secara lebih akurat. Hal ini membantu menghadapi tantangan urbanisasi seperti polusi, kecelakaan lalu lintas, dan konsumsi sumber daya yang berlebihan (Dilek and Dener 2023).

Prediksi arus lalu lintas merupakan elemen penting dalam Sistem Transportasi Cerdas karena menyediakan wawasan berharga untuk kontrol lalu lintas, perencanaan rute, dan manajemen operasi (Chi et al. 2025). Model prediksi arus lalu lintas tradisional tidak dapat sepenuhnya mempertimbangkan karakteristik jaringan lalu lintas perkotaan yang kompleks dan dinamis (Xing, Chen, and Zhang 2023). Dengan percepatan urbanisasi dan perkembangan ITS, prediksi arus lalu lintas jangka pendek menjadi bidang penelitian yang semakin signifikan (Changxi, Yanming, and Xuecai 2024). Prediksi yang akurat memberikan manfaat besar, seperti optimalisasi perencanaan lalu lintas, peningkatan pemanfaatan jalan, pengurangan kemacetan, penurunan insiden kecelakaan, dan pengurangan polusi lingkungan (Zheng, Wang, and Huang 2024).

Untuk mencapai prediksi yang akurat, diperlukan ekstraksi dan analisis data lalu lintas perkotaan yang besar secara efisien, termasuk pemilihan ukuran sampel data yang tepat. Kemajuan teknologi seperti kamera CCTV pinggir jalan dan kendaraan udara tak berawak (UAV) menyediakan data video baru yang memungkinkan pengumpulan informasi lalu lintas secara lebih lengkap menggunakan teknik visi komputer (Abdel-aty et al. 2023). Hal ini mendukung analisis keselamatan berbasis kecelakaan serta mewujudkan kontrol lalu lintas real-time, panduan rute, perumusan kebijakan, dan alokasi lalu lintas yang lebih efektif untuk meningkatkan efisiensi lalu lintas dan kualitas hidup masyarakat (Gao et al. 2024). Dalam implementasinya, model visi komputer seperti YOLO dan pengembangannya banyak

digunakan untuk mendeteksi keadaan lalu lintas perkotaan (R. Zhao et al. 2024), (Xu et al. 2024), (Saputri et al. 2024), (Tang et al. 2024), (Ronariv et al. 2024), (Detection 2023).

Dalam sistem transportasi cerdas, algoritme deteksi objek tradisional menghadapi berbagai tantangan, terutama dalam menangani lingkungan yang kompleks dan kondisi pencahayaan yang bervariasi. Tantangan ini menjadi lebih signifikan ketika harus mendeteksi target berukuran kecil atau memproses data multimoda (Tang et al. 2024). Untuk mengatasi keterbatasan tersebut, salah satu pendekatan yang digunakan adalah meningkatkan kualitas dan keragaman data melalui teknik augmentasi (Shamta, Demir, and Demir 2024). Penelitian sebelumnya menunjukkan bahwa kombinasi deteksi objek dengan algoritme LSTM dapat digunakan untuk prediksi volume lalu lintas (K. Wang et al. 2021). Selain itu, berbagai studi juga mengusulkan pengembangan model baru yang memanfaatkan dan mengoptimalkan algoritme LSTM, yang telah terbukti efektif dalam menangani data deret waktu serta meningkatkan akurasi prediksi kepadatan lalu lintas perkotaan (K. Wang et al. 2021), (Lu 2023), (Naheliya, Redhu, and Kumar 2024), (J. D. Wang and Susanto 2023), (Sattarzadeh et al. 2023), (Luo et al. 2024b). Tren terbaru menunjukkan bahwa banyak penelitian berfokus pada pengoptimalan parameter LSTM menggunakan pendekatan metaheuristik untuk meningkatkan performa prediksi lalu lintas (K. Zhao et al. 2023), (Bharti, Redhu, and Kumar 2023), (Naheliya, Redhu, and Kumar 2024), (CHAOURA, LAZAR, and JARIR 2024). Pendekatan ini diharapkan mampu menjawab tantangan dalam

menghasilkan model prediksi yang lebih andal dan efisien di berbagai kondisi lalu lintas.

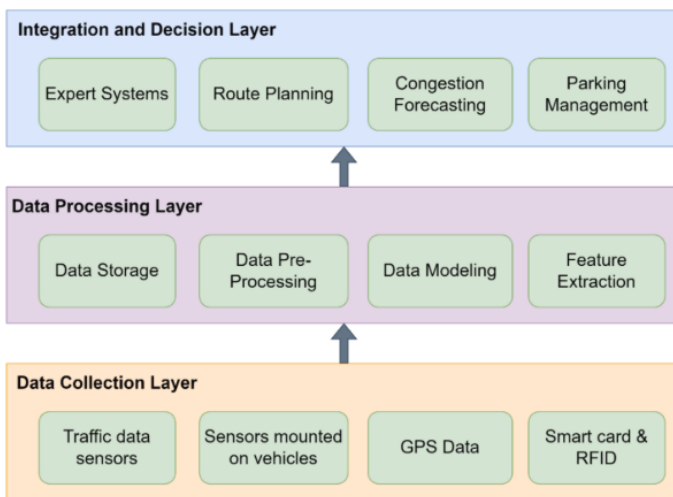
Algoritma Komodo Mlipir Optimization (KMA) adalah algoritme yang terinspirasi dari dua fenomena unik: perilaku komodo yang hidup di Nusa Tenggara Timur, Indonesia, serta gaya berjalan orang Jawa yang dikenal dengan istilah mlipir (Suyanto, Ariyanto, and Ariyanto 2022b). Dalam konteks penelitian Traveling Salesman Problem (TSP), KMA menunjukkan performa unggul dibandingkan algoritme lain seperti Dragonfly Algorithm (DKA), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Black Hole (BH), Dynamic Tabu Search Algorithm (DTSA), dan Discrete Jaya Algorithm (DJAYA) (Kusuma et al. 2023). Pada penelitian yang kami usulkan, Long Short-Term Memory (LSTM) dikombinasikan dengan KMA untuk prediksi volume lalu lintas. Kombinasi LSTM-KMA yang kami usulkan kemudian dibandingkan dengan LSTM standar serta kombinasi LSTM terkini, yaitu IDBO-LSTM dan LSTM-PSO. Penelitian sebelumnya menunjukkan bahwa IDBO-LSTM unggul dibandingkan metode lain seperti Gray Wolf Optimization (GWO), Sparrow Optimization Algorithm (SSA), Whale Optimization Algorithm (WOA), dan Nighthawk Optimization (NGO) (K. Zhao et al. 2023). Sementara itu, LSTM-PSO telah terbukti lebih baik daripada metode lainnya seperti LSTM standar, Random Forest Regression (RFR), K-Nearest Regression (KNR), dan Decision Tree Regression (DTR) (CHAOURA, LAZAR, and JARIR 2024).

Kontribusi penelitian ini adalah pengembangan model prediksi lalu lintas dengan menggabungkan algoritme LSTM dan KMA yang dirancang untuk meningkatkan akurasi prediksi. Proses prediksi diawali dengan deteksi kendaraan secara real-time menggunakan YOLO, yang memungkinkan pengambilan data kendaraan dengan cepat dan akurat dalam berbagai kondisi lalu lintas. Dengan memanfaatkan kekuatan KMA dalam optimasi parameter dan kemampuan LSTM dalam menangani data deret waktu, model ini diharapkan mampu mengatasi tantangan dalam memprediksi volume lalu lintas yang kompleks dan dinamis. Hasil dari penelitian ini diharapkan tidak hanya memperkuat fondasi teoritis dalam bidang optimasi dan prediksi, tetapi juga membuka peluang implementasi model ini dalam sistem transportasi cerdas untuk mendukung pengambilan keputusan berbasis data secara real-time.

1.2. Transportasi Cerdas

Perkembangan transportasi cerdas telah mengalami kemajuan pesat berkat integrasi teknologi Big Data dan Artificial Intelligence (AI) (S. Abirami et al. 2024a). Sistem Transportasi Cerdas (ITS) menggunakan teknologi canggih untuk mengatasi tantangan transportasi modern, menciptakan jaringan transportasi yang lebih pintar, cepat, dan aman. Peningkatan ketersediaan data dan munculnya AI serta Big Data telah memungkinkan ITS mendapatkan perhatian signifikan dalam beberapa tahun terakhir. Integrasi AI dan Big Data berkontribusi secara signifikan terhadap pengembangan ITS, mengoptimalkan perencanaan lalu lintas, peramalan, dan manajemen, serta

mengurangi biaya transportasi dengan meningkatkan kinerja transportasi umum, ride-sharing, dan parkir pintar (Lnenicka, Hervert, and Horak 2024). Secara umum, arsitektur Big Data untuk ITS memiliki arsitektur tiga lapis yang terdiri dari lapisan Pengumpulan Data, lapisan Analisis Data, dan lapisan Aplikasi seperti yang ditunjukkan pada Gambar 1.



Gambar 1. Arsitektur Big Data Tiga Lapisan untuk ITS (S. Abirami et al. 2024a)

ITS membantu mengurangi waktu idle kendaraan, memprioritaskan kendaraan darurat, mempercepat respons darurat, meningkatkan efisiensi lalu lintas, mencegah tabrakan tak terduga, dan mengurangi emisi karbon (Shaaban, Elamin, and Alsoub 2021). Integrasi komunikasi real-time 5G dalam teknologi ini dapat meningkatkan efektivitasnya lebih lanjut. Kota-kota pintar menggunakan ITS untuk mengoptimalkan aliran lalu lintas, mengurangi kemacetan, dan meningkatkan kualitas hidup

warganya (Ribeiro et al. 2023). Contoh penerapan ITS dalam kota pintar termasuk sistem parkir cerdas yang mengarahkan pengemudi ke tempat parkir yang tersedia dan sistem manajemen lalu lintas yang menyesuaikan sinyal lalu lintas berdasarkan kondisi lalu lintas saat itu (Elassy et al. 2024).

Algoritma AI memainkan peran penting dalam berbagai aplikasi ITS seperti kendaraan otonom, manajemen lalu lintas, dan optimasi transportasi umum (Preeti and Rana 2024)(S. Abirami et al. 2024b)(Shaygan, Ardecani, and Nejad 2025). Algoritma ini meningkatkan keselamatan, efisiensi, dan keandalan dalam sistem transportasi dengan memungkinkan kendaraan membuat keputusan yang terinformasi, mengoptimalkan aliran lalu lintas, dan meningkatkan layanan transportasi umum. Selain itu, algoritma AI memfasilitasi pemantauan real-time dan pemeliharaan prediktif infrastruktur transportasi, memastikan deteksi dini anomali dan bahaya (S. Abirami et al. 2024b).

Computer Vision memainkan peran penting dalam pengembangan ITS dengan menyediakan kemampuan untuk menganalisis dan menafsirkan data visual dari kamera dan sensor yang dipasang di jalan dan kendaraan (D. Wu et al. 2024). Algoritma Computer Vision dapat mendeteksi keberadaan kendaraan, mengklasifikasikan jenis kendaraan, dan memantau pergerakannya. Teknologi ini memungkinkan sistem sinyal lalu lintas cerdas untuk mengumpulkan data real-time tentang kondisi lalu lintas dan menyesuaikan pengaturan sinyal untuk mengurangi waktu tunggu kendaraan (Deng et al. 2023). Selain itu, Computer Vision dapat digunakan

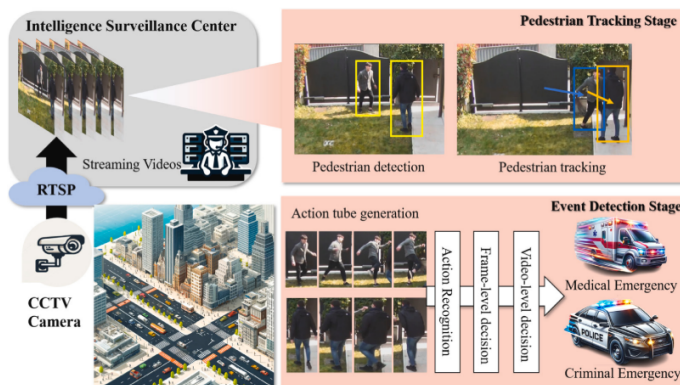
untuk mendeteksi pejalan kaki dan pengendara sepeda di persimpangan, memberikan peringatan kepada pengemudi, dan meningkatkan keselamatan jalan. Dengan integrasi IoT dan jaringan 5G, Computer Vision dapat memberikan informasi yang lebih akurat dan real-time, meningkatkan efisiensi dan keamanan sistem transportasi.

1.3. Peran Computer Vision dalam Transportasi Cerdas

Computer Vision (CV) memainkan peran penting dalam pengembangan sistem transportasi cerdas (Intelligent Transport Systems, ITS), terutama dalam meningkatkan efisiensi pengelolaan lalu lintas perkotaan. Teknologi CV memungkinkan pemantauan dan analisis lalu lintas secara real-time, yang mendukung pengambilan keputusan yang lebih cepat dan akurat dalam mengelola arus lalu lintas, meningkatkan keselamatan, serta mengurangi kemacetan dan polusi (Jeon et al. 2024).

Salah satu aplikasi utama CV dalam ITS adalah deteksi dan penghitungan kendaraan. Menggunakan teknologi seperti YOLO (You Only Look Once) dan Faster R-CNN, sistem CV dapat mendeteksi berbagai jenis kendaraan dalam gambar atau video yang diambil dari kamera CCTV yang terpasang di ruas-ruas jalan (Ganapathy and Ajmera 2024). Dengan menggunakan model deep learning, sistem ini dapat mengidentifikasi kendaraan secara otomatis dan menghitung jumlah kendaraan yang melintas, yang merupakan data penting untuk memprediksi arus lalu lintas dan memberikan informasi kepada pengendara atau sistem manajemen lalu lintas.

CV juga digunakan untuk menganalisis pola lalu lintas dan mengidentifikasi perilaku pengendara. Dengan menganalisis video lalu lintas secara real-time, teknologi ini dapat mengidentifikasi kemacetan, pelanggaran lalu lintas, serta perilaku abnormal seperti kendaraan yang melawan arus atau kecepatan tinggi (Kim et al. 2024). Hasil analisis ini dapat digunakan untuk mengoptimalkan lampu lalu lintas, memberikan informasi kepada pengendara, dan merencanakan jalur alternatif yang lebih efisien. Gambaran umum sistem deteksi perilaku abnormal yang diusulkan untuk kota pintar ditunjukkan pada gambar 2.



Gambar 2. Sistem deteksi perilaku abnormal kota pintar (Kim et al. 2024)

Selain itu, CV berperan penting dalam meningkatkan keamanan jalan dengan mendeteksi pelanggaran lalu lintas secara otomatis, seperti kendaraan yang melanggar lampu merah atau batas kecepatan. Sistem pengawasan yang dilengkapi dengan CV dapat mengidentifikasi pelanggaran ini secara otomatis dan memberikan bukti yang kuat dalam bentuk rekaman

video, yang kemudian dapat digunakan oleh pihak berwenang untuk tindakan lebih lanjut (Saravanan and Rajini 2024).

CV juga berintegrasi dengan sistem transportasi cerdas lainnya, seperti pengelolaan transportasi publik dan pemantauan kondisi jalan. Misalnya, analisis citra dari kamera dapat membantu mendeteksi kerusakan pada jalan atau infrastruktur lainnya yang dapat menghambat arus lalu lintas (Kaur et al. 2023). Data yang diperoleh dari teknologi CV ini dapat digunakan untuk merencanakan perbaikan atau pemeliharaan yang diperlukan, yang pada gilirannya meningkatkan kelancaran transportasi di perkotaan.

1.4. Penerapan Deep Learning untuk Prediksi Lalu Lintas

Deep learning telah menjadi teknologi unggulan dalam pengembangan sistem prediksi lalu lintas modern. Dengan kemampuan untuk memproses data besar dan menangkap pola non-linear yang kompleks, metode deep learning memberikan akurasi yang jauh lebih tinggi dibandingkan dengan metode prediktif tradisional (Yanming, Xuecai, and Science 2024). Dalam konteks prediksi lalu lintas, pendekatan ini memanfaatkan data real-time seperti volume kendaraan, kecepatan, dan pola perjalanan untuk menghasilkan prediksi yang lebih presisi (G. Zhang et al. 2024).

Metode deep learning seperti Recurrent Neural Network (RNN) dan Long Short-Term Memory (LSTM) sering digunakan untuk prediksi lalu lintas karena kemampuannya dalam menangani data deret waktu. Model LSTM, khususnya, dirancang untuk

mengatasi masalah long-term dependencies yang sering muncul dalam pola lalu lintas, seperti dampak lalu lintas pagi pada arus kendaraan di siang hari (Alzughairi, Karim, and Darwish 2024).

Selain itu, arsitektur Convolutional Neural Network (CNN) juga digunakan untuk analisis data lalu lintas berbasis citra, seperti penghitungan kendaraan melalui rekaman video atau citra drone. CNN dapat mengenali pola-pola penting dalam citra, seperti kepadatan kendaraan atau keberadaan hambatan jalan (Ibrahim Mohammad Ata et al. 2024).

Prediksi lalu lintas dengan deep learning tidak hanya bergantung pada data lalu lintas tradisional seperti volume kendaraan, tetapi juga mengintegrasikan data dari berbagai sumber lain (G. Zhang et al. 2024). Misalnya, data cuaca, jadwal transportasi publik, atau bahkan media sosial dapat dimasukkan sebagai input tambahan ke model deep learning. Teknik ini dikenal sebagai "fusion modeling," yang bertujuan untuk meningkatkan akurasi prediksi dengan memanfaatkan informasi kontekstual (Hu et al. 2024).

Deep learning mendukung prediksi lalu lintas baik untuk jangka pendek maupun jangka panjang. Dalam prediksi jangka pendek, data real-time digunakan untuk memprediksi kondisi lalu lintas beberapa menit atau jam ke depan. Pendekatan ini bermanfaat untuk pengendalian lampu lalu lintas adaptif dan pemberian informasi perjalanan kepada pengguna jalan. Di sisi lain, prediksi jangka panjang sering digunakan untuk perencanaan transportasi perkotaan, seperti mengidentifikasi lokasi kemacetan kronis atau

merencanakan jalur transportasi publik baru (Chen et al. 2024).

Deep learning menawarkan keunggulan dalam menangkap pola yang rumit dan tidak terstruktur dalam data lalu lintas (Selmy, Mohamed, and Medhat 2024). Namun, penerapannya juga menghadapi tantangan, seperti kebutuhan akan data yang besar, masalah overfitting, dan waktu pelatihan yang panjang. Selain itu, interpretabilitas hasil prediksi deep learning sering menjadi tantangan, terutama dalam pengambilan keputusan berbasis model ini.

1.5. Algoritma Metaheuristik dalam Optimasi

Algoritma metaheuristik merupakan pendekatan berbasis pencarian yang sering digunakan untuk menyelesaikan masalah optimasi yang kompleks dan berskala besar (Lara-montaña et al. 2024). Metaheuristik mampu memberikan solusi yang mendekati optimal dalam waktu komputasi yang relatif singkat, sehingga sangat cocok untuk diterapkan pada berbagai domain, termasuk optimasi parameter model deep learning seperti LSTM dalam prediksi lalu lintas (K. Zhao et al. 2024).

Metaheuristik memiliki beberapa karakteristik utama yang membuatnya fleksibel dan efektif untuk berbagai jenis masalah optimasi (Yanhang Li et al. 2024):

- **Heuristik Tingkat Tinggi:** Metaheuristik mengadopsi strategi pencarian umum yang dapat disesuaikan dengan berbagai jenis masalah.
- **Pendekatan Probabilistik:** Algoritma ini sering menggunakan elemen probabilistik untuk

mengeksplorasi ruang solusi secara luas, menghindari perangkap lokal optimum.

- Komputasi Iteratif: Solusi diperbarui secara iteratif hingga kriteria penghentian terpenuhi, seperti jumlah iterasi maksimum atau tidak ada peningkatan solusi dalam beberapa iterasi.

Beberapa algoritma metaheuristik yang banyak digunakan meliputi:

- Genetic Algorithm (GA): Terinspirasi oleh teori evolusi, algoritma ini menggunakan seleksi, crossover, dan mutasi untuk menghasilkan solusi baru (Putri et al. 2024).
- Particle Swarm Optimization (PSO): Meniru perilaku sosial kelompok seperti kawanan burung, PSO memanfaatkan interaksi antar partikel untuk mencari solusi optimal (Yan Li and Sun 2024).
- Simulated Annealing (SA): Berdasarkan prinsip annealing dalam fisika, algoritma ini menggunakan teknik pendinginan untuk menghindari perangkap lokal optimum (Khah et al. 2024).
- Ant Colony Optimization (ACO): Meniru perilaku semut dalam mencari makanan, algoritma ini efektif untuk masalah jalur terpendek (Blum 2024).
- Komodo Mlipir Optimization (KMA): Algoritma berbasis populasi yang terinspirasi oleh perilaku komodo. Algoritma ini menggabungkan eksplorasi (mlipir) dan eksploitasi untuk mencari solusi optimal secara adaptif (Suyanto, Ariyanto, and Ariyanto 2022a).

Dalam konteks optimasi model LSTM untuk prediksi lalu lintas, metaheuristik dapat digunakan untuk menyesuaikan parameter seperti learning rate, jumlah neuron, jumlah hidden layers, epochs, dan batch size. Proses optimasi ini bertujuan untuk meningkatkan akurasi prediksi sekaligus mengurangi waktu pelatihan (K. Zhao et al. 2023).

Sebagai contoh, penggunaan Particle Swarm Optimization (PSO) dalam pengaturan hyperparameter LSTM telah menunjukkan hasil yang signifikan dalam meningkatkan akurasi prediksi arus lalu lintas (CHAOURA, LAZAR, and JARIR 2024). Algoritma ini mampu mengeksplorasi berbagai kombinasi parameter dengan efisien, sehingga memberikan model yang lebih baik dibandingkan pendekatan trial-and-error manual.

Keunggulan metaheuristik meliputi (G. Li et al. 2024):

- Fleksibilitas untuk berbagai masalah optimasi.
- Kemampuan menangani fungsi tujuan yang tidak memiliki gradien.
- Efisiensi dalam mengeksplorasi ruang solusi yang besar.

Namun, metaheuristik juga menghadapi tantangan, seperti (G. Li et al. 2024):

- Kebutuhan untuk menyesuaikan parameter algoritma (misalnya, ukuran populasi, probabilitas mutasi).
- Ketergantungan pada waktu komputasi yang cukup untuk mencapai solusi yang optimal.

BAB II

Model LSTM untuk Data Temporal

2.1. Konsep Dasar Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) adalah salah satu jenis jaringan saraf tiruan yang dirancang untuk menangani data sekuensial dengan pola temporal yang kompleks. Berbeda dengan jaringan saraf tradisional, LSTM memiliki kemampuan untuk menyimpan informasi dalam jangka waktu panjang, memungkinkan model untuk mempertahankan konteks dari data yang bersifat serial (Al-selwi, Fadzil, and Jadid 2024).

LSTM dikembangkan untuk mengatasi masalah utama dalam jaringan saraf berulang (Recurrent Neural Network atau RNN) standar, yaitu vanishing gradient dan exploding gradient. Masalah ini membuat RNN sulit untuk belajar dari data dengan hubungan temporal yang panjang. LSTM mengatasi kendala ini dengan memperkenalkan struktur unik yang disebut sel memori dan gerbang-gerbang pengatur aliran informasi.

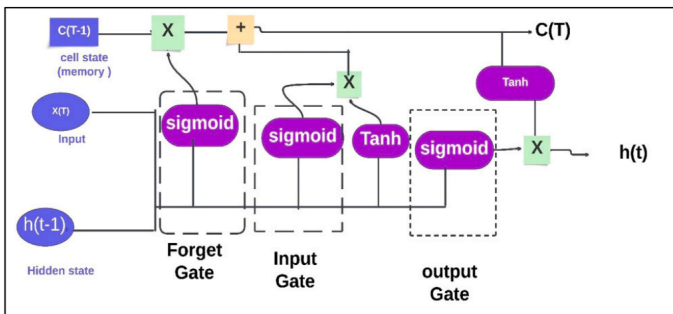
Komponen utama dalam LSTM meliputi (Al-selwi, Fadzil, and Jadid 2024):

1. **Gerbang Forget (Forget Gate):** Berfungsi untuk menentukan informasi mana yang tidak relevan dari memori sebelumnya dan harus dihapus. Proses ini dilakukan dengan fungsi aktivasi sigmoid:
di mana adalah output forget gate, adalah input

saat ini, adalah output dari waktu sebelumnya, dan, adalah parameter model.

2. **Gerbang Input (Input Gate):** Menentukan informasi baru yang akan ditambahkan ke memori. Kombinasi fungsi sigmoid dan tanh digunakan untuk memilih informasi baru:
3. **Gerbang Output (Output Gate):** Mengontrol informasi mana yang akan digunakan untuk output pada langkah waktu berikutnya. Output ditentukan oleh:

Proses ini memastikan bahwa LSTM dapat menangkap pola temporal jangka panjang tanpa kehilangan detail dari hubungan jangka pendek. Dengan struktur ini, LSTM menjadi salah satu metode yang paling efektif untuk data temporal, termasuk deret waktu, analisis sinyal, dan prediksi pola kompleks. Skema sel LSTM ditunjukkan pada gambar 3.

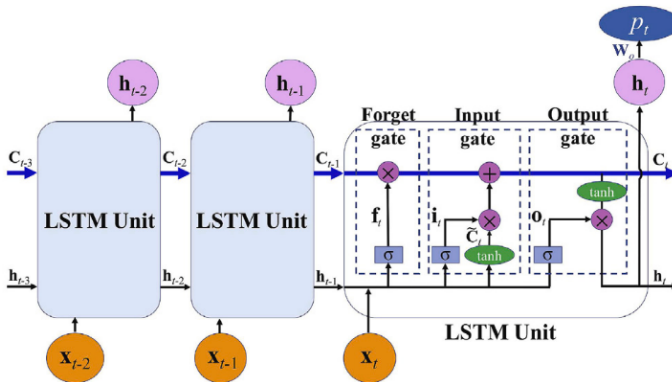


Gambar 3. Skema LSTM (Abdelkader, Zaineldin, and Saafan 2025)

2.2. Arsitektur LSTM

Long Short-Term Memory (LSTM) dirancang untuk memproses data sekuensial dengan struktur unik yang memungkinkan pembelajaran hubungan temporal

jangka panjang. Arsitektur ini memiliki beberapa elemen penting. Lapisan input menerima data sekuensial, di mana setiap elemen diproses secara berurutan sambil mempertahankan informasi dari elemen sebelumnya. Data input biasanya direpresentasikan dalam bentuk tensor dengan dimensi tertentu. Arsitektur LSTM ditunjukkan pada gambar 4.



Gambar 4. Arsitektur LSTM (Al-selwi, Fadzil, and Jadid 2024)

Bagian inti dari LSTM adalah sel memori yang berfungsi untuk menyimpan informasi selama proses pelatihan. Sel memori ini dilengkapi dengan tiga gerbang utama: gerbang forget untuk menghapus informasi yang tidak relevan, gerbang input untuk menambahkan informasi baru ke memori berdasarkan input saat ini, dan gerbang output yang menghasilkan output berdasarkan memori terkini.

Pada lapisan rekurensi, beberapa sel LSTM dihubungkan secara berurutan untuk memproses data sekuensial. Setiap sel menerima output dari langkah waktu sebelumnya sebagai input tambahan. Struktur

ini dapat diperluas menjadi beberapa tingkatan (stacked LSTM) untuk meningkatkan kapasitas pembelajaran fitur (Al-selwi, Fadzil, and Jadid 2024).

Output dari sel LSTM terakhir diteruskan ke lapisan output untuk menghasilkan hasil akhir, seperti prediksi atau klasifikasi. Aktivasi non-linear seperti sigmoid atau softmax sering digunakan pada lapisan ini untuk menghasilkan output yang sesuai.

Proses pelatihan LSTM melibatkan bobot dan bias sebagai parameter utama yang menghubungkan input, memori, dan output. Parameter ini dioptimalkan selama pelatihan menggunakan algoritma backpropagation through time (BPTT). Pada tahap forward pass, data input diproses secara berurutan melalui jaringan, menghasilkan output prediksi di setiap langkah waktu. Tahap backward pass kemudian menghitung error dari prediksi untuk memperbarui bobot.

Arsitektur LSTM sering divisualisasikan dalam bentuk diagram yang menunjukkan alur data melalui lapisan-lapisan dan gerbang-gerbang. Visualisasi ini membantu memahami interaksi antara berbagai komponen dalam jaringan, sehingga menunjukkan kemampuan LSTM dalam menangkap pola temporal yang kompleks.

2.3. Keunggulan LSTM dalam Analisis Data Temporal

LSTM memiliki sejumlah keunggulan yang membuatnya unggul dalam analisis data temporal, terutama dibandingkan dengan pendekatan tradisional seperti RNN standar atau model berbasis statistik. Salah satu keunggulan utamanya

adalah kemampuannya menangani masalah vanishing gradient. Dengan struktur unik yang melibatkan gerbang forget dan memori seluler, LSTM dapat mempertahankan gradien selama pelatihan, sehingga memungkinkan pembelajaran pola yang melibatkan dependensi jangka panjang tanpa kehilangan informasi (Luo et al. 2024a).

LSTM juga fleksibel untuk berbagai jenis data temporal, seperti deret waktu finansial, analisis cuaca, sinyal biomedis, dan data sekuensial lainnya. Kemampuannya untuk menangani data dengan ketidakpastian waktu (irregular time series) menjadikannya alat yang sangat serbaguna. Tidak seperti RNN standar yang sering kehilangan konteks karena keterbatasan kapasitas memori, LSTM mampu menyimpan informasi penting selama beberapa langkah waktu, memungkinkan pemahaman konteks jangka panjang.

Dalam hal generalisasi, LSTM memiliki kemampuan untuk mempelajari pola kompleks tanpa overfitting, terutama ketika digunakan dengan teknik regulasi seperti dropout. Selain itu, implementasi modern LSTM mendukung pemrosesan paralel menggunakan GPU atau TPU, sehingga mempercepat proses pelatihan dan inferensi.

Dalam aplikasi prediksi deret waktu, seperti forecasting cuaca, permintaan energi, dan volume lalu lintas, LSTM menunjukkan akurasi yang tinggi karena kemampuannya untuk menangkap pola non-linear (Thakkar, Kikani, and Geddam 2024). Model ini juga dapat dengan mudah digabungkan dengan model lain, seperti CNN (Convolutional Neural Network), untuk aplikasi yang melibatkan data

spasial dan temporal, misalnya analisis video (Amadeo, Gunawan, and Anindaputri 2024).

Dengan keunggulan-keunggulan tersebut, LSTM menjadi salah satu pilihan utama untuk aplikasi berbasis data temporal, memberikan hasil yang lebih baik dibandingkan metode konvensional.

2.4. Aplikasi LSTM untuk Prediksi Lalu Lintas

Prediksi lalu lintas merupakan salah satu aplikasi unggulan LSTM dalam menangani data temporal. Data lalu lintas biasanya mencakup informasi seperti volume kendaraan, kecepatan rata-rata, kondisi cuaca, waktu, dan lokasi, yang dikumpulkan dari berbagai sumber seperti sensor jalan atau kamera CCTV (CHAOURA, LAZAR, and JARIR 2024). Data ini perlu diproses terlebih dahulu untuk mengisi nilai yang hilang, menangani outlier, dan menormalisasi fitur agar sesuai dengan kebutuhan model LSTM.

Dalam penggunaannya, LSTM mempelajari pola dari data historis seperti volume atau kecepatan kendaraan pada rentang waktu tertentu. Model ini mampu menangkap hubungan temporal jangka pendek maupun jangka panjang yang penting untuk prediksi lalu lintas. Arsitektur model mencakup lapisan input yang menerima data deret waktu dengan berbagai fitur, lapisan tersembunyi yang terdiri dari beberapa lapisan LSTM untuk menangkap pola temporal kompleks, dan lapisan output yang menghasilkan prediksi volume lalu lintas pada waktu mendatang.

Model dievaluasi menggunakan metrik seperti Mean Absolute Error (MAE), Mean Squared Error (MSE), atau Root Mean Squared Error (RMSE) untuk

mengukur akurasi prediksi (J. D. Wang and Susanto 2023). Visualisasi antara prediksi dan nilai aktual digunakan untuk memvalidasi kemampuan model dalam menangkap pola temporal.

Keunggulan LSTM dalam prediksi lalu lintas meliputi adaptasi terhadap variasi temporal seperti jam sibuk atau kondisi cuaca ekstrem, serta kemampuan generalisasi yang memungkinkan model memberikan prediksi andal pada data yang belum pernah dilihat sebelumnya (Bharti, Redhu, and Kumar 2023). Dalam implementasi nyata, LSTM telah digunakan untuk memprediksi lalu lintas di kota-kota besar seperti Jakarta atau New York. Model ini memberikan wawasan berharga untuk pengelolaan lalu lintas, seperti pengaturan lampu lalu lintas atau perencanaan rute alternatif.

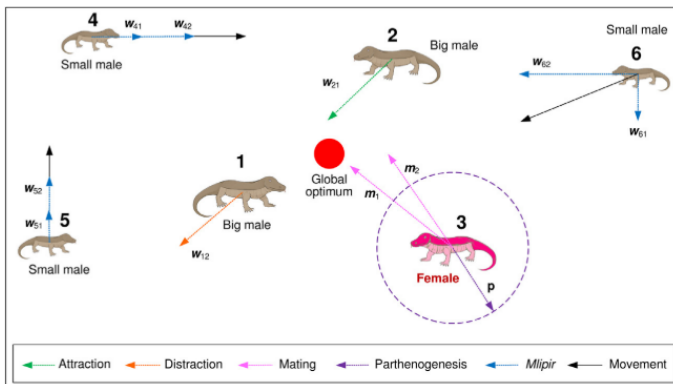
Hasil prediksi LSTM dapat diintegrasikan dengan sistem manajemen lalu lintas cerdas, seperti Adaptive Traffic Signal Control (ATSC), untuk mengoptimalkan aliran kendaraan secara real-time. Dengan kemampuan LSTM dalam menangkap pola temporal kompleks, model ini menjadi solusi yang andal untuk prediksi lalu lintas yang akurat, mendukung pengelolaan lalu lintas perkotaan yang lebih baik.

BAB III

Komodo Mlipir Algorithm (KMA) sebagai Teknik Optimasi

3.1. Pengenalan Komodo Mlipir Algorithm

Komodo Mlipir Algorithm (KMA) adalah algoritma metaheuristik yang terinspirasi dari perilaku komodo dalam berburu dan bertahan hidup. Istilah "mlipir" dalam bahasa Jawa menggambarkan gerakan perlahan namun pasti, merefleksikan pendekatan komodo saat menangkap mangsa, yang diterapkan untuk menyelesaikan masalah optimasi kompleks (Suyanto, Ariyanto, and Ariyanto 2022b). Konsep algoritma komodo mlipir dapat ditunjukkan pada gambar 5.



Gambar 5. Konsep Algoritma Komodo Mlipir (Suyanto, Ariyanto, and Ariyanto 2022a)

Pendekatan KMA mengintegrasikan kemampuan eksplorasi dan eksploitasi dengan cara yang unik. Eksplorasi dilakukan melalui pergerakan adaptif yang memungkinkan algoritma menjelajahi ruang solusi

secara luas. Eksploitasi, di sisi lain, dilakukan dengan fokus pada solusi terbaik yang telah ditemukan untuk meningkatkan kualitas hasil secara bertahap. Dengan pola ini, algoritma KMA efektif dalam menemukan solusi optimal dan menghindari jebakan solusi lokal (Kusuma et al. 2023).

Algoritma ini dirancang untuk efisiensi dan konsistensi. Proses iteratifnya tidak hanya memastikan bahwa solusi yang dihasilkan berkualitas tinggi, tetapi juga meminimalkan kebutuhan komputasi dibandingkan algoritma lain seperti Genetic Algorithm atau Particle Swarm Optimization. Dengan pendekatan yang inovatif ini, KMA menawarkan potensi besar dalam menyelesaikan berbagai tantangan optimasi di dunia nyata, termasuk dalam bidang prediksi lalu lintas berbasis data temporal.

Keunikan KMA terletak pada adaptasinya terhadap berbagai jenis masalah optimasi. Algoritma ini memanfaatkan mekanisme seleksi yang menyerupai cara komodo memilih mangsa berdasarkan peluang keberhasilan. Selain itu, gangguan acak yang terkontrol ditambahkan untuk memastikan diversifikasi solusi, sehingga memungkinkan pencarian solusi yang lebih luas tanpa kehilangan fokus pada target optimal. KMA menjadi sangat relevan dalam aplikasi pembelajaran mesin, khususnya dalam optimasi parameter model seperti LSTM, di mana performa model sangat bergantung pada konfigurasi parameter yang tepat. Algoritma Komodo Mlipir dapat dilihat pada algoritma 1.

Algoritma 1: Algoritma Komodo Mlipir

Hasil: k_{best} sebagai solusi optimum global
Tetapkan n, p , dan d sebagai jumlah individu Komodo, proporsi jantan besar, dan laju mlipir, masing-masing.

Inisialisasi n individu dengan dimensi m

While	<i>StoppingCriterion</i> = <i>false</i> do • Untuk setiap Komodo, hitung kualitasnya, lalu urutkan; • Berdasarkan peringkat mereka dan proporsi p, bagi populasi menjadi tiga grup: • q jantan besar dengan kualitas tertinggi, • 1 betina dengan kualitas sedang, • dan s jantan kecil dengan kualitas rendah menggunakan persamaan 1 dan 2. • Untuk setiap jantan besar, pindahkan menggunakan persamaan 4, dan pertahankan q jantan besar berkualitas tertinggi (posisi terbaik) untuk bertahan di generasi berikutnya. • Perbarui betina dengan cara: • kawin dengan jantan besar pemenang menggunakan persamaan 5, atau • melakukan partenogenesis menggunakan persamaan 7. • Untuk setiap jantan kecil, pindahkan menggunakan persamaan 8, dan pertahankan semua posisi baru mereka untuk bertahan di generasi berikutnya.
--------------	--

end	• Perbarui ukuran populasi n menggunakan persamaan 10. • Pilih Komodo dengan kualitas tertinggi dari ketiga grup sebagai solusi terbaik sejauh ini, k_{best}
------------	---

3.2. Konsep Komodo Mlipir Algorithm

Konsep Komodo Mlipir Algorithm (KMA) dapat diilustrasikan menggunakan masalah optimasi sederhana dengan fungsi dua dimensi $f(x_1, x_2) = x_1^2 + x_2^2$, di mana x_1 dan x_2 masing-masing adalah sumbu horizontal dan vertikal (Suyanto, Ariyanto, and Ariyanto 2022b). Solusi optimum global (target) adalah $f = 0$ pada $x_1 = 0$ dan $x_2 = 0$. Pertama, populasi awal yang terdiri dari enam individu Komodo (solusi kandidat) secara acak dihasilkan, masing-masing direpresentasikan sebagai vektor posisi tanpa kecepatan dalam ruang masalah. Populasi ini kemudian dibagi menjadi tiga kelompok: big males berkualitas tinggi, female berkualitas menengah, dan small males berkualitas rendah. Dalam konteks ini, big male dengan kualitas tertinggi merepresentasikan solusi terbaik sejauh ini.

Evolusi dilakukan dalam urutan tertentu. Pertama, big males berinteraksi satu sama lain menggunakan pendekatan high-exploitation low-exploration (HILE). Selanjutnya, female dapat memilih untuk kawin dengan big male terbaik atau melakukan partenogenesis (reproduksi aseksual) dengan probabilitas normal terdistribusi sebesar 0,5, menghasilkan eksploitasi atau eksplorasi tingkat menengah (MIME). Terakhir, small males melakukan gerakan mlipir untuk mengeksplorasi ruang

pencarian dengan eksplorasi tinggi dan eksploitasi rendah (LIHE).

Big males berinteraksi melalui mekanisme tarik-menarik dan pengalihan berdasarkan aturan tertentu. Big male berkualitas rendah tertarik ke arah big male berkualitas lebih tinggi, sedangkan big male berkualitas tinggi dapat tertarik atau terdistraksi oleh big male berkualitas lebih rendah secara acak dengan probabilitas 0,5. Proses ini memastikan eksploitasi lebih dominan dibandingkan eksplorasi. Female, di sisi lain, dapat menghasilkan dua keturunan melalui kawin dengan big male terbaik atau melakukan partenogenesis dengan menambahkan nilai acak kecil pada dimensinya. Small males, melalui gerakan mlipir, memilih sebagian dimensi secara acak untuk bergerak mendekati big males sambil tetap menjaga jarak untuk menghindari kanibalisme.

Seluruh gerakan ini, termasuk HILE, MIME, dan LIHE, dirancang agar dapat diperluas ke dimensi tinggi (m -dimensi), dengan small males mengikuti big males hanya dalam sebagian dimensi tertentu, berdasarkan tingkat mlipir. Proses ini membantu KMA mencapai keseimbangan antara eksploitasi dan eksplorasi, memastikan pencarian solusi optimal global dengan efisiensi tinggi.

3.3. Representasi Individu, Kelompok, dan Mekanisme Evolusi dalam KMA

Dalam KMA, setiap individu Komodo direpresentasikan sebagai vektor bilangan riil dengan m dimensi yang menunjukkan posisinya dalam ruang masalah. Misalnya, populasi yang terdiri dari enam Komodo ($k_1, k_2, \dots, k_6$) pada dimensi dua ($m = 2$) hanya memiliki posisi tanpa kecepatan, yang

membuatnya lebih dinamis dalam bergerak di ruang pencarian karena inersia diabaikan (Suyanto, Ariyanto, and Ariyanto 2022b).

Populasi individu Komodo (n) dibagi menjadi tiga kelompok: big males (berkualitas tinggi), female (berkualitas menengah), dan small males (berkualitas rendah). Proporsi untuk setiap kelompok ditentukan oleh parameter p (umumnya $p = 0.5$) dengan persamaan 1 dan 2:

$$q = \lfloor (p - 1) n \rfloor, \quad (1)$$

$$s = n - q, \quad (2)$$

di mana q adalah jumlah big males, s adalah jumlah small males, dan setidaknya setiap kelompok memiliki dua individu untuk interaksi.

Big males berinteraksi melalui mekanisme tarik-menarik dan pengalihan berdasarkan kualitasnya. Big male berkualitas lebih rendah bergerak menuju yang lebih tinggi, sedangkan yang berkualitas lebih tinggi dapat tertarik atau terdistraksi secara acak dengan probabilitas 0,5. Gerakan ini diformulasikan seperti persamaan 3 dan 4:

$$w_{ij} = \begin{cases} r_1(k_j - k_i), & \text{if } f(k_j) < f(k_i) \text{ or } r_2 < 0.5 \\ r_1(k_i - k_j), & \text{jika tidak} \end{cases} \quad (3)$$

$$k'_i = k_i + \sum_{j=1}^q w_{ij}, \text{ dimana } j \neq i, \quad (4)$$

Mekanisme ini dikenal sebagai eksploitasi tinggi dan eksplorasi rendah (HILE), yang memungkinkan pencarian intensif di sekitar solusi terbaik.

Female melakukan eksploitasi atau eksplorasi dengan probabilitas 0,5. Dalam eksploitasi, female kawin

dengan big male terbaik untuk menghasilkan dua keturunan menggunakan persamaan 5:

$$\begin{aligned} k'_{il} &= rlk_{il} + (1 - rl)k_{jl}, \\ k'_{jl} &= rlk_{jl} + (1 - rl)k_{il}, \end{aligned} \quad (5)$$

Sedangkan dalam eksplorasi, female melakukan partenogenesis dengan menambahkan nilai acak kecil menggunakan persamaan 6 dan 7:

$$\begin{aligned} (k_{i1}, k_{i2}, \dots, k_{im}) &\rightarrow (k'_{i1}, k'_{i2}, \dots, k'_{im}), \quad (6) \\ k'_{ij} &= k_{ij} + (2r - 1)\alpha|ub_j - lb_j|, \quad (7) \end{aligned}$$

di mana $\alpha = 0.1$ memastikan variasi berada dalam radius 10% dari ruang pencarian.

Small males bergerak menggunakan strategi mlipir untuk mengikuti big males dalam sebagian dimensi secara acak berdasarkan tingkat mlipir (d) menggunakan persamaan 8 dan 9:

$$w_{ij} = \begin{cases} \sum_{l=1}^m r_l(k_{jl} - k_{il}), & \text{jika } r_2 < d \\ 0, & \text{lainnya} \end{cases} \quad (8)$$

$$k'_i = k_i + \sum_{j=1}^q w_{ij}, \text{ dimana } j \neq i, \quad (9)$$

Gerakan ini memastikan eksplorasi tinggi dan eksploitasi rendah (LIHE), yang mendukung pencarian luas pada ruang solusi.

Ukuran populasi (n) diadaptasi secara dinamis selama evolusi untuk mengoptimalkan eksploitasi dan eksplorasi. Jika ada peningkatan fitness terbaik secara berturut-turut, n dikurangi dengan menghapus individu, sedangkan stagnasi meningkatkan n dengan menambahkan individu baru dengan persamaan 10:

$$n' = \begin{cases} n - a, & \text{if } \delta f_1 > 0 \text{ dan } \delta f_2 > 0 \\ n + a, & \text{if } \delta f_1 = 0 \text{ and } \delta f_2 = 0 \end{cases} \quad (10)$$

dengan a adalah jumlah individu yang dihapus atau ditambahkan.

Kombinasi mekanisme gerakan big males, reproduksi female, dan gerakan small males dalam KMA menghasilkan keseimbangan eksplorasi dan eksploitasi yang efektif. Adaptasi populasi juga memperkuat kemampuan algoritma untuk menangani berbagai jenis masalah optimasi.

3.4. Perbedaan KMA dibanding Algoritma Metaheuristik Lain

KMA yang diusulkan berbeda dari algoritma metaheuristik yang ada karena menggabungkan tiga strategi pencarian: HILE, MIME, dan LIHE untuk menemukan solusi optimum global dengan cepat dan menghindari jebakan solusi lokal. Gerakan HILE dan LIHE sepenuhnya berbeda dari yang digunakan dalam algoritma yang ada, sedangkan MIME serupa dengan proses crossover dan mutasi dalam GA (Suyanto, Ariyanto, and Ariyanto 2022b).

HILE dirancang menggunakan mekanisme tarik-menarik dan pengalihan antara individu berkualitas tinggi dengan probabilitas 0,5. Hal ini berbeda dengan mekanisme dalam DFA yang selalu (dengan probabilitas 1,0) dilakukan berdasarkan dua aturan: seekor capung tertarik ke individu yang lebih baik dan teralihkan dari individu yang lebih buruk (Hammouri et al. 2020). Sementara itu, LIHE melakukan intensifikasi rendah dan diversifikasi tinggi menggunakan gerakan mlipir individu berkualitas rendah untuk menghindari jebakan solusi

lokal. Tidak seperti mutasi diferensial dalam DE yang dihitung dari vektor tidak langsung dari dua hingga empat individu terbaik atau individu yang dipilih secara acak, gerakan mlipir terdiri dari vektor tarik langsung dari sebagian dimensi satu hingga tiga individu berkualitas lebih tinggi yang dipilih secara acak (Opara and Arabas 2018). Seperti yang diilustrasikan pada Gambar 6, gerakan mlipir ini secara logis memberikan strategi eksploitasi yang lebih tinggi dibandingkan dengan mutasi diferensial dalam DE.

Biaya komputasi KMA dapat dengan mudah diperkirakan berdasarkan analisis kompleksitas. Kompleksitas waktu KMA untuk satu generasi (iterasi) adalah $O(nm + nc + \log n)$, di mana n, m, c , dan $\log n$ masing-masing adalah jumlah individu dalam populasi, dimensi, perhitungan fungsi objektif, dan proses pengurutan nilai kebugaran. Dibandingkan dengan GA, EO, MPA, dan SMA, KMA yang diusulkan sedikit lebih rumit. KMA memiliki kompleksitas tambahan berupa $\log p$ yang berasal dari proses pengurutan. Selain itu, KMA juga memiliki kompleksitas lain dari skema adaptasi populasi, meskipun cukup rendah.

BAB IV

Metodologi Pendekatan LSTM-KMA

4.1. Proses Data Lalu Lintas

Dataset dalam penelitian ini diperoleh dari kamera pengawas (CCTV) yang terpasang di lampu lalu lintas Fatmawati, Kota Semarang. Video direkam menggunakan codec H.264/MPEG-4 AVC dengan resolusi 1280x960 piksel dan frame rate 25 FPS, menghasilkan kualitas gambar yang tajam dan halus. Pengumpulan data dilakukan pada 19 Desember 2023 hingga 15 Februari 2024, setiap hari antara pukul 06.00 hingga 07.00 WIB. Data yang dikumpulkan difokuskan pada kendaraan di sisi kanan jalan. Contoh gambar yang diambil dari kamera CCTV ditunjukkan pada Gambar 6. Proses pengumpulan data melibatkan ekstraksi frame dari video yang direkam setiap 5 menit. Dari seluruh proses ekstraksi, dihasilkan total 720 gambar. Dataset ini kemudian diklasifikasikan ke dalam empat kelas utama: sepeda motor, mobil, truk, dan bus. Dari dataset tersebut, 80% atau 576 gambar digunakan untuk pelatihan, sementara sisanya 20% atau 144 gambar digunakan untuk validasi. Pembagian ini dirancang untuk memastikan model menerima data yang representatif untuk pelatihan dan validasi, memungkinkan model mencapai kinerja optimal dalam mendeteksi berbagai jenis kendaraan.



Gambar 6. Tangkapan gambar CCTV di lampu lalu lintas Fatmawati, Kota Semarang

Pada tahap Data Pre-processing, video CCTV dikonversi menjadi gambar untuk analisis lebih lanjut. Langkah pertama adalah memotong video menjadi segmen frame dengan interval 5 menit. Proses ini dilakukan menggunakan Python, di mana skrip Python memuat video menggunakan pustaka yang sesuai untuk pemrosesan video. Program ini menggunakan beberapa pustaka Python, seperti cv2 untuk pemrosesan video, numpy untuk manipulasi array, os untuk operasi sistem file, dan PIL untuk menyimpan frame sebagai gambar. Setelah video dimuat, skrip menghitung total durasi video dan membaginya menjadi segmen 5 menit. Interval frame dihitung dengan mengalikan FPS (frame per second) dengan 60 detik dan jumlah menit yang ditentukan. Dalam video ini, dengan FPS 25, interval frame yang dihasilkan adalah $25 \times 60 \times 5 = 7500$. Artinya, setiap 7500 frame, program akan mengekstraksi satu gambar. Program ini tidak bekerja berdasarkan waktu yang tepat tetapi berdasarkan jumlah frame. Jika

interval 5 menit diterapkan, program akan mengambil frame pertama sekitar pukul 06:02:42, kemudian 5 menit berikutnya pada pukul 06:07:42, dan seterusnya, dengan variasi kecil tergantung pada FPS video dan penyesuaian internal Python saat menghitung interval frame.

Selanjutnya, skrip Python mengiterasi setiap segmen waktu. Pada setiap iterasi, skrip menangkap frame yang mewakili waktu pada interval 5 menit tersebut dan mengekstraknya sebagai gambar. Gambar yang dihasilkan kemudian disimpan dalam format yang sesuai di direktori yang telah ditentukan, dengan nama file yang menunjukkan stempel waktu gambar. Proses ini memastikan bahwa setiap gambar diambil secara konsisten pada interval waktu yang sama dari video asli, memungkinkan analisis data gambar yang terstruktur dan seragam.

Teknik augmentasi bertujuan untuk mengubah atau memodifikasi gambar asli untuk menghasilkan variasi baru yang berguna dalam melatih model. Teknik augmentasi yang digunakan meliputi Blur, Penyesuaian Kecerahan, Penyesuaian Kontras, Color Jitter, Cropping, Flipping, Penyuntikan Noise, Rotasi, Scaling, dan Zoom-In. Setiap teknik memiliki faktor augmentasi yang berbeda yang diterapkan pada gambar (S. Yang et al. 2022)(Oza et al. 2022).

Teknik Blur menggunakan ukuran kernel untuk mengaburkan gambar dengan tingkat augmentasi yang bervariasi. Penyesuaian Kecerahan dan Kontras memodifikasi faktor kecerahan dan alpha gambar untuk menghasilkan variasi dalam kecerahan dan kontras. Color Jitter mengubah kecerahan, kontras, saturasi, dan hue dengan faktor acak untuk

menciptakan lebih banyak variasi warna. Teknik Cropping memotong bagian atas atau bawah gambar pada proporsi tertentu untuk menghasilkan variasi perspektif. Flipping mencakup pembalikan horizontal dan vertikal, yang mengubah orientasi gambar. Penyuntikan Noise menambahkan noise Gaussian untuk meningkatkan variasi tekstur gambar. Rotasi mengubah orientasi gambar pada sudut tertentu, seperti 90 dan 270 derajat. Scaling mengubah ukuran gambar dalam rentang tertentu, sedangkan Zoom-In memperbesar gambar dengan f Semua teknik ini dirancang untuk meningkatkan keragaman data gambar dan membantu melatih model yang lebih andal dan tangguh. Nilai dari setiap teknik augmentasi yang diterapkan dapat dilihat pada Tabel 1.

Dalam Tabel 1, setiap gambar dalam dataset pelatihan diduplikasi menjadi tiga versi untuk meningkatkan variasi data yang digunakan dalam pelatihan model. Gambar pertama adalah versi asli tanpa augmentasi, sementara dua gambar lainnya dikenakan teknik augmentasi yang berbeda. Dengan menduplikasi setiap gambar menjadi tiga versi, total gambar yang digunakan dalam pelatihan mencapai 1.728, terdiri dari 576 gambar asli dan 1.152 gambar yang telah diaugmentasi. Teknik ini membantu model belajar dari data yang lebih beragam, meningkatkan kemampuannya untuk mendeteksi objek di bawah berbagai kondisi pencahayaan, sudut pandang, dan kualitas gambar (Yun 2019).

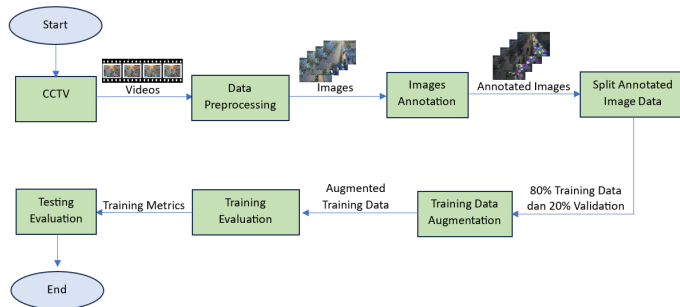
Tabel 1. Nilai Augmentasi

No	Aug	Value	Faktor Augmentasi		
			1	2	3
1	Blur	Kernel Size	-	1	2
2	Brightness Adjustment	Brightness Factor	-	0.8	1.2
3	Contrast Adjustment	Alpha	-	1.5	2.0
4	Color Jitter	Brightnes	-	Rand(0.6, 1.4)	Rand(0.6, 1.4)
		Contrast	-	Rand(0.6, 1.4)	Rand(0.6, 1.4)
		Saturation	-	Rand(0.6, 1.4)	Rand(0.6, 1.4)
		Hue	-	Rand(-25.5, 25.5)	Rand(-25.5, 25.5)
5	Cropping	Crop Height	-	Top Crop (0,0) to (width, height * 0.5)	Bottom Crop (0, height * 0.5) to (width, height)
6	Flipping	Horizontal and Vertical Flip	-	Horizontal Flip $x_center = 1.0 - x_center$	Vertical Flip $y_center = 1.0 - y_center$
7	Noise Injection	Gaussian Noise	-	Rand(0,0.1)	Rand(0,0.1)
8	Rotation	Rotation	-	90°	270°
9	Scaling	Scale Image	-	Rand(0.8, 1.2)	Rand(0.8, 1.2)
10	Zoom In	Zoom In	-	1.2	1.5

4.2. Penggunaan YOLO untuk Deteksi Kendaraan

Alur kerja untuk deteksi kendaraan multi-label secara real-time di lalu lintas menggunakan model YOLO yang ditingkatkan dengan teknik multi-augmentasi. Diagram di bawah ini mengilustrasikan setiap tahap,

mulai dari pengambilan data mentah hingga evaluasi hasil model, dengan penekanan pada penggunaan teknik augmentasi untuk meningkatkan performa model YOLO. Diagram alur kerja dapat dilihat pada Gambar 7.



Gambar 7. Alur kerja deteksi kendaraan menggunakan YOLO

Gambar 7 menunjukkan alur kerja deteksi kendaraan di lalu lintas menggunakan model YOLO yang ditingkatkan dengan teknik multi-augmentasi. Proses dimulai dengan perekaman video menggunakan kamera CCTV yang merekam lalu lintas kendaraan. Video ini kemudian diproses pada tahap Preprocessing Data untuk dipersiapkan dalam proses anotasi. Proses akuisisi gambar melibatkan pemecahan rekaman video menjadi frame dengan interval waktu 5 menit selama tahap preprocessing data. Setelah itu, setiap kendaraan yang muncul dalam gambar dianotasi berdasarkan jenisnya, termasuk informasi detail tentang objek, seperti lokasi dan koordinat kotak pembatas (bounding box). Proses ini memastikan bahwa data yang dihasilkan siap untuk analisis lebih lanjut atau pelatihan model.

Setelah proses anotasi, data dibagi menjadi dua bagian: 80% untuk pelatihan dan 20% untuk validasi, guna memastikan bahwa model dapat dilatih dan dievaluasi dengan baik menggunakan data yang tidak digunakan selama pelatihan (Ayana et al. 2023)(Yousif et al. 2024)(T. Wu and Dong 2023). Selanjutnya, berbagai teknik augmentasi diterapkan pada dataset pelatihan untuk meningkatkan variasi data pelatihan dan meningkatkan performa model. Teknik augmentasi yang digunakan meliputi Blur, Penyesuaian Kecerahan, Penyesuaian Kontras, Color Jitter, Cropping, Flipping, Penyuntikan Noise, Rotasi, Scaling, dan Zoom In.

Training Validation

Validasi pelatihan dilakukan menggunakan 32 skenario dengan model YOLOv8n dan YOLOv9t. Data pelatihan terdiri dari 1 dataset tanpa augmentasi, 10 dataset dengan augmentasi tunggal, dan 5 dataset dengan augmentasi ganda. Jumlah sampel pelatihan yang digunakan adalah 576 untuk dataset tanpa augmentasi, 1.728 untuk dataset dengan augmentasi tunggal, dan 6.336 untuk dataset dengan augmentasi ganda. Untuk validasi, digunakan 144 sampel, yang mewakili 20% dari dataset awal. Validasi dilakukan dengan mengukur beberapa metrik, yaitu Precision, Recall, mAP50, dan mAP50-95. Dari metrik tersebut, mAP50-95 menjadi metrik utama untuk menentukan nilai tertinggi atau hasil terbaik karena mencakup evaluasi pada berbagai ambang IoU, memberikan penilaian yang lebih komprehensif terhadap kinerja model dalam berbagai situasi. Hasil metrik kinerja model yang diuji ditampilkan pada Tabel 2, yang

mengilustrasikan performa setiap model dan augmentasi yang diterapkan.

Tabel 2. Perbandingan Metrik Kinerja Model yang Diuji

No	Augmentation	Models	mA P50- 95
1	Without Augmentation	YOLOv8n	0.39 0
2	Without Augmentation	YOLOv9t	0.37 8
3	Blur	YOLOv8n	0.46 5
4	Blur	YOLOv9t	0.44 3
5	Brightness Adjustment	YOLOv8n	0.46 2
6	Brightness Adjustment	YOLOv9t	0.44 9
7	Contrast Adjustment	YOLOv8n	0.20 3
8	Contrast Adjustment	YOLOv9t	0.16 7
9	Color Jitter	YOLOv8n	0.46 1
10	Color Jitter	YOLOv9t	0.41 1
11	Cropping	YOLOv8n	0.35 6
12	Cropping	YOLOv9t	0.34 9
13	Flipping	YOLOv8n	0.43 6

14	Flipping	YOLOv9t	0.41 1
15	Noise Injection	YOLOv8n	0.43 3
16	Noise Injection	YOLOv9t	0.42 2
17	Rotation	YOLOv8n	0.37 1
18	Rotation	YOLOv9t	0.36 6
19	Scaling	YOLOv8n	0.46 4
20	Scaling	YOLOv9t	0.45 9
21	Zoom-In	YOLOv8n	0.42 6
22	Zoom-In	YOLOv9t	0.42 1
23	Scaling + Cropping + Brightness Adjustment + Noise Injection + Blur	YOLOv8n	0.51 1
24	Scaling + Cropping + Brightness Adjustment + Noise Injection + Blur	YOLOv9t	0.50 3
25	Rotation + Flipping + Brightness Adjustment + Contrast Adjustment + Color Jitter	YOLOv8n	0.49 1
26	Rotation + Flipping + Brightness Adjustment + Contrast Adjustment + Color Jitter	YOLOv9t	0.48 0
27	Scaling + Zoom In + Brightness Adjustment +	YOLOv8n	0.52 6

	Color Jitter + Noise Injection		
28	Scaling + Zoom In + Brightness Adjustment + Color Jitter + Noise Injection	YOLOv9t	0.506
29	Cropping + Zoom In + Contrast Adjustment + Noise Injection + Blur	YOLOv8n	0.498
30	Cropping + Zoom In + Contrast Adjustment + Noise Injection + Blur	YOLOv9t	0.469
31	Scaling + Rotation + Brightness Adjustment + Contrast Adjustment + Noise Injection	YOLOv8n	0.511
32	Scaling + Rotation + Brightness Adjustment + Contrast Adjustment + Noise Injection	YOLOv9t	0.493

Berdasarkan nilai mAP50-95 yang disajikan dalam Tabel 2, model YOLO tanpa augmentasi mencapai nilai mAP50-95 tertinggi dalam uji YOLOv8n dengan skor 0,390. Sementara itu, model YOLO dengan augmentasi tunggal mencapai nilai mAP50-95 tertinggi dalam uji YOLOv8n menggunakan teknik augmentasi Blur dengan skor 0,465. Untuk model YOLO dengan augmentasi ganda, nilai mAP50-95 tertinggi diperoleh menggunakan kombinasi teknik Scaling, Zoom In, Brightness Adjustment, Color Jitter, dan Noise Injection, mencapai skor 0,526.

Dari hasil uji ini, jelas bahwa penerapan teknik augmentasi, baik tunggal maupun ganda, secara signifikan meningkatkan performa model YOLO dibandingkan dengan model tanpa augmentasi. Model dengan augmentasi ganda menghasilkan hasil terbaik, menunjukkan bahwa kombinasi beberapa teknik augmentasi dapat secara efektif meningkatkan kemampuan deteksi model. Dalam kasus deteksi lalu lintas perkotaan selama jam sibuk, di mana objek kendaraan sering berukuran kecil dan terkena pantulan sinar matahari, model YOLOv8n terbukti lebih baik dalam menangani tantangan ini. Uji menunjukkan bahwa model YOLOv8n mencapai akurasi yang lebih tinggi dibandingkan YOLOv9t dalam skenario tersebut. Oleh karena itu, dapat disimpulkan bahwa YOLOv8n lebih unggul dalam hal akurasi untuk mendeteksi objek dalam kondisi lalu lintas yang kompleks dan menantang.

Hasil evaluasi metrik untuk setiap kelas dari YOLOv8n menggunakan augmentasi ganda (Scaling, Zoom In, Brightness Adjustment, Color Jitter, dan Noise Injection) dapat dilihat pada Tabel 3.

Tabel 3. Kelas Metrik Evaluasi dari YOLOv8n menggunakan Multi-Augmentasi

Class	Precision	Recall	mAP50	mAP50-59
All	0.872	0.715	0.792	0.526
Bus	0.977	0.788	0.850	0.626
Car	0.866	0.763	0.834	0.538
Motorcycle	0.831	0.559	0.672	0.359
Truck	0.814	0.749	0.810	0.582

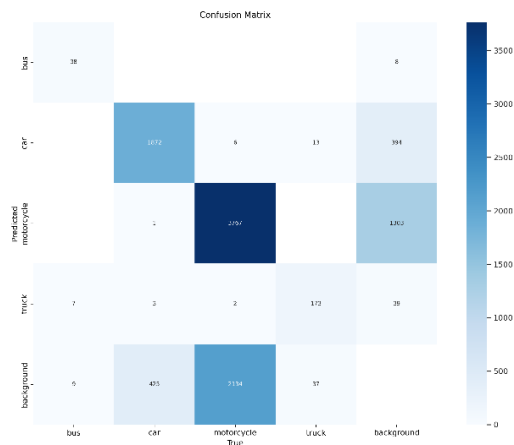
Tabel 3 menyajikan metrik evaluasi untuk berbagai kelas objek dalam model deteksi. Dari metrik yang ditampilkan, kelas "bus" memiliki nilai mAP50-95 terbaik sebesar 0,626, yang menunjukkan bahwa model memiliki kinerja deteksi yang konsisten di berbagai ambang IoU untuk kelas ini. Kelas "bus" juga memiliki precision tertinggi, yaitu 0,977, yang menunjukkan bahwa model jarang membuat kesalahan deteksi untuk kelas ini, kemungkinan karena karakteristik visual bus yang besar dan mudah dikenali.

Kelas "car" memiliki recall terbaik dengan nilai 0,763, menunjukkan bahwa model berhasil mendeteksi sebagian besar mobil dalam dataset, mungkin karena mobil lebih sering muncul dalam gambar dan model lebih terlatih untuk mendeteksinya. Kelas "bus" juga memiliki nilai mAP50 tertinggi, yaitu 0,85, yang menunjukkan akurasi yang sangat baik dalam mendeteksi bus pada ambang IoU sebesar 0,50, kemungkinan karena bus lebih mudah dikenali dan berukuran lebih besar dibandingkan objek lain seperti mobil atau sepeda motor.

Kelas "motorcycle" memiliki nilai mAP50-95 terendah sebesar 0,359, menunjukkan bahwa deteksi model untuk sepeda motor kurang optimal di berbagai ambang IoU. Precision untuk kelas ini juga relatif rendah, yaitu 0,831, yang menunjukkan lebih banyak kesalahan deteksi untuk sepeda motor dibandingkan kelas kendaraan lainnya. Hal ini kemungkinan disebabkan oleh ukuran sepeda motor yang lebih kecil dan kecenderungannya menciptakan bayangan

atau pantulan, sehingga lebih menantang bagi model untuk mendeteksinya secara konsisten.

Gambar 8 menunjukkan bahwa model berhasil mengidentifikasi sepeda motor sebanyak 3.767 kali dan mobil sebanyak 1.872 kali, tetapi sering salah mengidentifikasi sepeda motor sebagai mobil (1.303 kali) dan mobil sebagai latar belakang (394 kali). Kesalahan lain termasuk salah mengklasifikasikan latar belakang sebagai mobil (425 kali) dan sepeda motor (37 kali). Meskipun model menunjukkan performa yang baik, perbaikan diperlukan dalam menangani objek kecil dan kondisi pencahayaan.

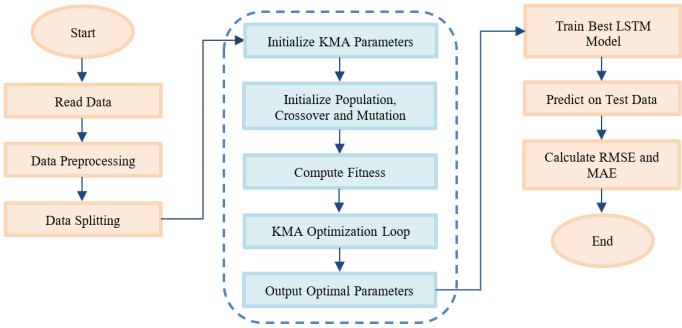


Gambar 8. Confusion Matrix

4.3. Integrasi LSTM dengan KMA

Integrasi Long Short-Term Memory (LSTM) dan Komodo Mlipir Algorithm (KMA) memanfaatkan keunggulan masing-masing metode dalam analisis data dan optimasi. LSTM sangat efektif dalam menangkap pola temporal pada data deret waktu, sehingga cocok untuk tugas prediksi jangka pendek

maupun panjang (Hamiane et al. 2024). Studi sebelumnya menunjukkan bahwa optimasi hiperparameter menggunakan pendekatan metaheuristik sering memberikan hasil yang lebih baik dibandingkan metode konvensional, yang semakin memperkuat keunggulan kombinasi teknik ini dalam meningkatkan performa model (C. Yang et al. 2023). Penerapan KMA dalam pendekatan ini diharapkan dapat melampaui kinerja algoritma metaheuristik lainnya. Integrasi LSTM dan KMA tidak hanya mempercepat proses optimasi, tetapi juga meningkatkan kemungkinan menemukan konfigurasi hiperparameter yang optimal, sehingga secara signifikan meningkatkan kinerja model LSTM dalam aplikasi prediksi arus lalu lintas. Pendekatan yang diusulkan ini ditunjukkan pada Gambar 9.



Gambar 9. Integrasi LSTM-KMA

Gambar 9 menunjukkan proses komputasi LSTM-KMA, yang dimulai dengan membaca dan preprocessing data, diikuti oleh pembagian dataset menjadi data pelatihan dan pengujian, masing-masing 80% untuk pelatihan dan 20% untuk pengujian (Halpern-wight, Konstantinou, and Charalambides n.d.), (Tan et al. 2022). Selanjutnya, Komodo Mlipir

Algorithm (KMA) diinisialisasi dengan parameter tertentu. Langkah ini melibatkan inisialisasi populasi kandidat solusi serta penerapan operasi crossover dan mutasi (A. Abirami and Kavitha 2023). Kelayakan setiap kandidat solusi dievaluasi untuk menentukan kesesuaian parameter dengan model LSTM menggunakan metrik MAE. Parameter LSTM yang dioptimasi meliputi jumlah neuron, tingkat pembelajaran (learning rate), dan jumlah epoch (K. Zhao et al. 2023). KMA secara iteratif memperbarui kandidat solusi melalui loop optimasi hingga parameter optimal ditemukan.

Parameter yang telah dioptimasi kemudian digunakan untuk melatih model LSTM akhir. Model yang telah dilatih ini selanjutnya diuji menggunakan data pengujian, dengan RMSE dan MAE dihitung sebagai ukuran akurasi prediksi. Konsep KMA dalam optimasi parameter LSTM diilustrasikan melalui pseudocode yang disajikan pada Algoritma 2.

Algorithm 2: Komodo MLIPIR for Optimizing LSTM Parameters

Input:

Jumlah iterasi maksimum (T), ukuran populasi (n).

Rentang parameter LSTM yang akan dioptimasi (jumlah neuron, learning rate, dan jumlah epoch).

Step 1: Initialization

Inisialisasi populasi sebanyak n individu (komodo) dengan kombinasi acak dari parameter LSTM.

Setiap individu q dalam populasi direpresentasikan sebagai: $Pq = [Xq, Yq, Zq]$, di mana Xq, Yq , dan

Z_q masing-masing menunjukkan jumlah neuron, learning rate, dan epoch.

Step 2: Fitness Evaluation

Evaluasi nilai fitness awal dari setiap kandidat solusi dengan mengukur kinerja LSTM pada data validasi.

Gunakan fungsi objektif: Minimalkan $F = \text{MAE}$ (Mean Absolute Error).

Urutkan individu berdasarkan skor fitness-nya, lalu kategorikan ke dalam tiga kelompok:

- Jantan besar (elite, performa terbaik)
- Betina (performa sedang)
- Jantan kecil (performa rendah)

Step 3: Main Loop

While ($t \leq T$):

1. Evaluasi ulang skor fitness setiap individu.
 2. Perbarui posisi mereka sebagai berikut:
 - Jantan besar: Sesuaikan posisi menggunakan strategi eksploitasi.
 - Betina:
 - Kawin dengan jantan besar terbaik menggunakan metode eksploitasi.
 - Bereproduksi secara aseksual (parthenogenesis) menggunakan strategi eksplorasi.
 - Jantan kecil: Menjelajahi ruang solusi secara acak menggunakan strategi eksplorasi.
 3. Terapkan proses seleksi:
 - Pertahankan individu dengan performa terbaik (elitisme).
-

-
- Perbaiki individu yang lemah menggunakan strategi pembaruan pada persamaan.

4. Tingkatkan jumlah iterasi ($t = t + 1$).

End While

Step 4: Output the Best Solution

Keluarkan parameter LSTM terbaik (P_{best}) dan nilai fitness terbaik (F_{best}).

Output: Parameter LSTM yang optimal.

BAB V
Eksperimen dan Hasil Penelitian

5.1. Parameter untuk Model LSTM yang Dioptimalkan

Dalam penelitian ini, parameter untuk metode metaheuristik distandardisasi dengan menetapkan nilai Populasi sebanyak 30 (K. Zhao et al. 2023). Rentang parameter yang dioptimalkan untuk model LSTM meliputi:

- Jumlah neuron: 300–500
- Learning Rate: 0,001–0,01
- Jumlah epoch: 1–150

Untuk model LSTM konvensional, analisis dilakukan dengan menggunakan nilai parameter tertinggi: 500 neuron, learning rate sebesar 0,01, dan 150 epoch. Konfigurasi detail parameter lainnya untuk setiap model dapat dilihat pada Tabel 4.

Tabel 4. Pengaturan Parameter dari Algoritma yang diuji

Algoritma	Parameter	Pengaturan	Referensi
LSTM-KMA	Probabilitas mutasi	20%	(A. Abirami and Kavitha 2023)
	Probabilitas crossover	0.8	
	Jumlah populasi	10	
IDBO-LSTM	Koefisien variasi	0.1 0.5	(K. Zhao et al. 2023)

	Skala atau parameter untuk pengaturan eksplorasi dan eksploitasi		
LSTM-PSO	Faktor pembelajaran mandiri Faktor pembelajaran kelompok	1.5 2	(CHAOUR A, LAZAR, and JARIR 2024)

5.2. Analisis Hasil Evaluasi (RMSE, MAE)

Metode evaluasi kinerja yang sesuai untuk data kontinu yang diperoleh secara real-time adalah fungsi loss regresi [45]. Oleh karena itu, metrik evaluasi kinerja yang digunakan dalam penelitian ini adalah RMSE dan MAE. RMSE mencerminkan tingkat deviasi antara nilai prediksi dengan nilai aktual. Rumus RMSE diberikan pada Persamaan 9 [46]. MAE mewakili rata-rata kesalahan absolut, di mana kesalahan absolut adalah selisih antara nilai prediksi dengan nilai aktual. Nilai MAE yang rendah menunjukkan bahwa model memprediksi nilai yang mendekati nilai aktual. Rumus MAE diberikan pada Persamaan 10 [26].

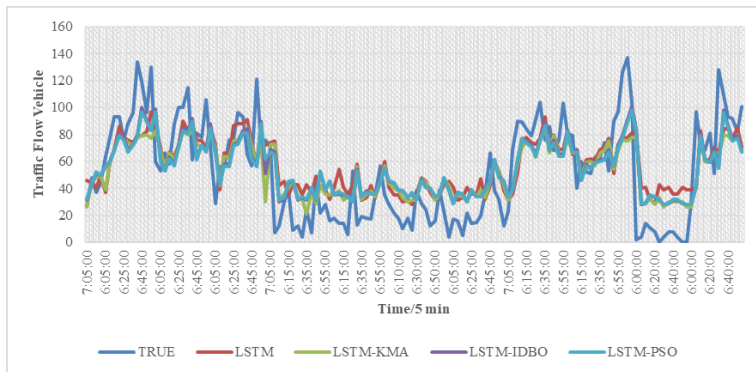
$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|^2} \quad (9)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (10)$$

RMSE dan MAE adalah dua metrik evaluasi yang digunakan untuk mengukur kesalahan prediksi sebuah model. Dalam rumus RMSE, selisih antara nilai aktual (y_i) dan nilai prediksi ($\hat{y}_i$) dikuadratkan untuk menghitung $(y_i - \hat{y}_i)^2$, sehingga memberikan bobot yang lebih besar pada kesalahan yang lebih besar, kemudian diambil akar kuadratnya. RMSE memberikan wawasan tambahan dengan mencerminkan tingkat deviasi antara nilai prediksi dengan nilai aktual, dan lebih sensitif terhadap kesalahan besar.

MAE memiliki rumus yang serupa, tetapi pendekatannya berbeda. Dalam rumus ini, n adalah jumlah total titik data atau observasi dalam dataset, yang menunjukkan jumlah data yang dianalisis. y_i adalah nilai aktual dari data ke- i , mewakili data sebenarnya yang diprediksi, seperti jumlah kendaraan aktual dalam prediksi lalu lintas. Sementara itu, $\hat{y}_i$ adalah nilai prediksi yang dihasilkan oleh model untuk data ke- i , mencerminkan perkiraan jumlah kendaraan. Selisih absolut antara nilai aktual dan prediksi dihitung sebagai $|y_i - \hat{y}_i|$, yang memberikan ukuran kesalahan tanpa memperhatikan arah kesalahan. Semua selisih absolut ini dijumlahkan dan dibagi dengan jumlah total titik data (n) untuk menghasilkan rata-rata kesalahan absolut.

Dengan demikian, RMSE dan MAE memberikan ukuran keseluruhan tentang seberapa dekat prediksi model dengan nilai aktual. Hasil prediksi dari model LSTM, LSTM-KMA, IDBO-LSTM, dan LSTM-PSO dibandingkan dengan data aktual. Hasil prediksi dari model yang digunakan ditunjukkan pada Gambar 10.



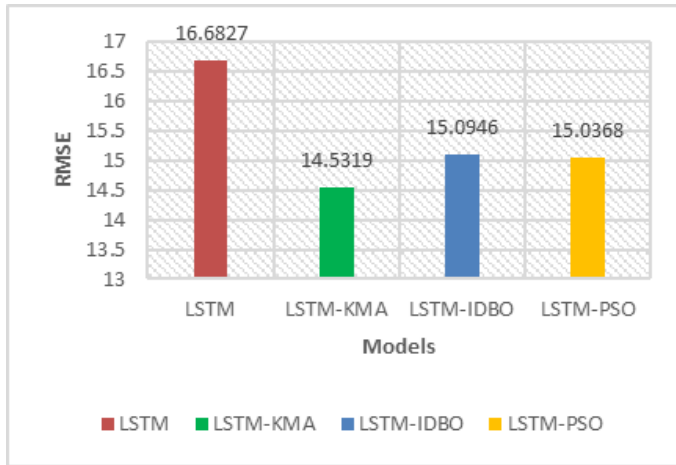
Gambar 10. Hasil prediksi dari masing-masing model

5.3. Perbandingan dengan Model Lain

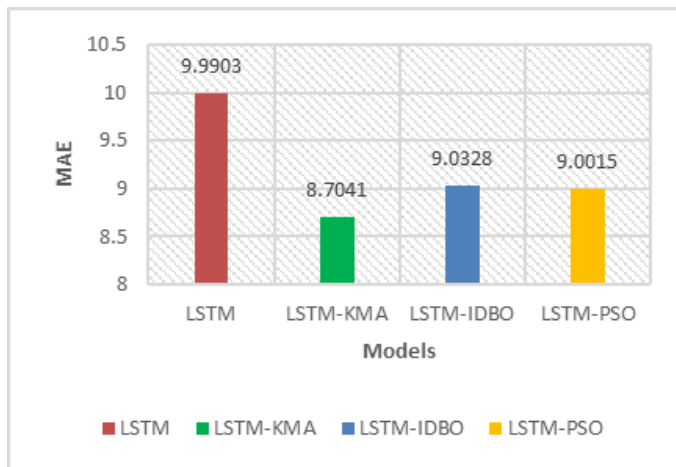
Model yang dikembangkan, yaitu LSTM-KMA, dibandingkan dengan model LSTM dasar serta beberapa model LSTM yang dioptimalkan menggunakan metode metaheuristik, yaitu LSTM-IDBO dan LSTM-PSO. Perbandingan kinerja masing-masing model berdasarkan metrik evaluasi RMSE dan MAE ditunjukkan pada Tabel 5, Gambar 11 dan 12.

Tabel 5. Nilai RMSE dan MAE dari Masing-Masing Model yang Dievaluasi Secara Individual

Model	RMSE	MAE
LSTM	16.6827	9.9903
LSTM-KMA	14.5319	8.7041
LSTM-IDBO	15.0946	9.0328
LSTM-PSO	15.0368	9.0015



Gambar 11. Hasil evaluasi RMSE



Gambar 12. Hasil evaluasi MAE

Gambar 11 menunjukkan bahwa model LSTM-KMA mencapai nilai RMSE terendah, yaitu 14.5319, yang menunjukkan kinerja terbaik dibandingkan dengan model lainnya. Sebaliknya, model LSTM dasar mencatat nilai RMSE tertinggi, yaitu 16.6827, yang

menunjukkan akurasi prediksi terendah. Model LSTM-IDBO dan LSTM-PSO masing-masing mencapai nilai RMSE sebesar 15.0946 dan 15.0368, yang menunjukkan peningkatan kinerja dibandingkan model LSTM dasar, tetapi masih kalah dibandingkan dengan LSTM-KMA.

Dari sisi MAE, LSTM-KMA juga menunjukkan kinerja terbaik dengan nilai terendah, yaitu 8.7041, dibandingkan dengan LSTM dasar (9.9903), LSTM-IDBO (9.0328), dan LSTM-PSO (9.0015). Berdasarkan analisis ini, optimasi menggunakan Komodo Mlipir Algorithm (KMA) terbukti menjadi metode paling efektif untuk meningkatkan kinerja model LSTM baik dalam hal RMSE maupun MAE, sehingga menjadi pendekatan yang direkomendasikan dalam penelitian ini.

BAB VI

Kesimpulan dan Rekomendasi

6.1. Kesimpulan

Deteksi kendaraan multi-target di lalu lintas perkotaan menghadapi tantangan signifikan, termasuk pencahayaan yang buruk, ukuran objek yang kecil, dan variasi jenis kendaraan, yang semuanya memengaruhi akurasi prediksi arus lalu lintas. Untuk mengatasi tantangan ini, penelitian ini mengusulkan penggunaan model Long Short-Term Memory (LSTM) yang dioptimalkan dengan Komodo Mlipir Algorithm (KMA).

Hasil analisis menunjukkan bahwa model LSTM-KMA mencapai Root Mean Square Error (RMSE) terendah sebesar 14.5319, mengungguli LSTM dasar (16.6827), LSTM-IDBO (15.0946), dan LSTM-PSO (15.0368). Selain itu, LSTM-KMA juga memberikan kinerja terbaik berdasarkan Mean Absolute Error (MAE), dengan nilai terendah sebesar 8.7041, lebih unggul dibandingkan LSTM dasar (9.9903), LSTM-IDBO (9.0328), dan LSTM-PSO (9.0015). Hal ini menunjukkan bahwa optimasi menggunakan KMA secara signifikan meningkatkan akurasi prediksi model LSTM dalam menghadapi kompleksitas deteksi kendaraan multi-target di lalu lintas perkotaan. Oleh karena itu, penelitian ini memberikan kontribusi yang signifikan terhadap pengembangan model prediktif yang tidak hanya mengatasi tantangan dalam deteksi kendaraan multi-target tetapi juga mendukung sistem manajemen lalu lintas secara real-time.

6.2. Kontribusi Penelitian

Kontribusi penelitian ini terletak pada pengembangan model prediksi lalu lintas yang mengintegrasikan algoritma Long Short-Term Memory (LSTM) dengan Komodo Mlipir Algorithm (KMA) untuk meningkatkan akurasi prediksi. Dengan menggabungkan kemampuan LSTM dalam menangani data deret waktu dan keunggulan KMA dalam optimasi, penelitian ini menjawab tantangan dalam memprediksi dinamika lalu lintas perkotaan yang kompleks. Integrasi deteksi kendaraan secara real-time menggunakan YOLO juga memastikan pengumpulan data yang cepat dan akurat di berbagai kondisi lalu lintas. Penelitian ini tidak hanya memperkaya kerangka teoretis dalam bidang optimasi dan prediksi, tetapi juga memberikan implikasi praktis untuk sistem transportasi cerdas, mendukung pengambilan keputusan berbasis data secara real-time guna meningkatkan efisiensi dan keberlanjutan manajemen lalu lintas perkotaan.

DAFTAR PUSTAKA

- Abdel-aty, Mohamed, Zijin Wang, Ou Zheng, and Amr Abdelraouf. 2023. "Advances and Applications of Computer Vision Techniques in Vehicle Trajectory Generation and Surrogate Traffic Safety Indicators." *Accident Analysis and Prevention* 191(July): 107191. <https://doi.org/10.1016/j.aap.2023.107191>.
- Abdelkader, Aml G, Hanaa Zaineldin, and Mahmoud M Saafan. 2025. "Sustainable Computing: Informatics and Systems Optimizing Wind Power Forecasting with RNN-LSTM Models through Grid Search Cross-Validation." 45(August 2024).
- Abirami, A., and R. Kavitha. 2023. "A Novel Automated Komodo Mlipir Optimization-Based Attention BiLSTM for Early Detection of Diabetic Retinopathy." *Signal, Image and Video Processing* 17(5): 1945–53. <https://doi.org/10.1007/s11760-022-02407-9>.
- Abirami, S., M. Pethuraj, M. Uthayakumar, and P. Chitra. 2024a. "A Systematic Survey on Big Data and Artificial Intelligence Algorithms for Intelligent Transportation System." *Case Studies on Transport Policy* 17(July 2023): 101247. <https://doi.org/10.1016/j.cstp.2024.101247>.
- . 2024b. "A Systematic Survey on Big Data and Artificial Intelligence Algorithms for Intelligent Transportation System." *Case Studies on Transport Policy* 17(July 2023).
- Al-selwi, Safwan Mahmood, Mohd Fadzil, and Said

- Jadid. 2024. "Journal of King Saud University - Computer and Information Sciences RNN-LSTM: From Applications to Modeling Techniques and beyond — Systematic Review ☆." *Journal of King Saud University - Computer and Information Sciences* 36(5): 102068. <https://doi.org/10.1016/j.jksuci.2024.102068>.
- Almatar, Khalid Mohammed. 2024. "Smart Transportation Planning and Its Challenges in the Kingdom of Saudi Arabia." *Sustainable Futures* 8(December 2023): 100238. <https://doi.org/10.1016/j.sftr.2024.100238>.
- Alzughairi, Arwa, Faten K. Karim, and Jumanah Ahmed Darwish. 2024. "Driven Traffic Flow Prediction in Smart Cities Using Hunter-Prey Optimization with Hybrid Deep Learning Models." *Alexandria Engineering Journal* 107(June): 625–33.
- Amadeo, Yohanes, Alexander A S Gunawan, and Irene Anindaputri. 2024. "ScienceDirect Enhancing Small Object Detection: Development on Car Logo Enhancing Small Object Development Car and Logo Identification Using the Detection: Method for on Speed Identification Using the YOLO-NAS Method for Speed and Accuracy." 00.
- Ayana, Gelan et al. 2023. "Vision-Transformer-Based Transfer Learning for Mammogram Classification." *Diagnostics* 13(2).
- Bharti, Poonam Redhu, and Kranti Kumar. 2023. "Short-Term Traffic Flow Prediction Based on

- Optimized Deep Learning Neural Network: PSO-Bi-LSTM.” *Physica A: Statistical Mechanics and its Applications* 625: 129001. <https://doi.org/10.1016/j.physa.2023.129001>.
- Blum, Christian. 2024. “Ant Colony Optimization: A Bibliometric Review.” *Physics of Life Reviews* 51(September): 87–95. <https://doi.org/10.1016/j.plrev.2024.09.014>.
- Changxi, Ma, Hu Yanming, and Xu Xuecai. 2024. “Na l P Re r F.” *Data Science and Management*. <https://doi.org/10.1016/j.dsm.2024.10.004>.
- CHAOURA, Chaimaa, Hajar LAZAR, and Zahi JARIR. 2024. “Traffic Flow Prediction at Intersections: Enhancing with a Hybrid LSTM-PSO Approach.” *International Journal of Advanced Computer Science and Applications* 15(5): 494–501.
- Chen, Qiucheng et al. 2024. “Empirical Dynamics of Traffic Moving Jams: Insights from Kerner’s Three-Phase Traffic Theory.” *Physica A: Statistical Mechanics and its Applications* 648(June): 129953. <https://doi.org/10.1016/j.physa.2024.129953>.
- Chi, Haiyang et al. 2025. “Engineering Applications of Artificial Intelligence Spatio-Temporal Attention Based Collaborative Local – Global Learning for Traffic Flow Prediction.” *Engineering Applications of Artificial Intelligence* 139(PB): 109575. <https://doi.org/10.1016/j.engappai.2024.109575>.
- Deng, Xiaoheng et al. 2023. “A Review of 6G

- Autonomous Intelligent Transportation Systems: Mechanisms, Applications and Challenges.” *Journal of Systems Architecture* 142(February).
- Detection, Object. 2023. “YOLO-Drone : An Optimized YOLOv8 Network for Tiny UAV Object Detection.”
- Dilek, Esma, and Murat Dener. 2023. “Computer Vision Applications in Intelligent Transportation Systems: A Survey.” *Sensors* 23(6).
- Elassy, Mohamed, Mohammed Al-Hattab, Maen Takruri, and Sufian Badawi. 2024. “Intelligent Transportation Systems for Sustainable Smart Cities.” *Transportation Engineering* 16(April).
- Ganapathy, Sannasi, and Devansh Ajmera. 2024. “An Intelligent Video Surveillance System for Detecting the Vehicles on Road Using Refined YOLOV4.” *Computers and Electrical Engineering* 113(November 2023).
- Gao, Heyao et al. 2024. “A Hybrid Deep Learning Model for Urban Expressway Lane-Level Mixed Traffic Flow Prediction.” *Engineering Applications of Artificial Intelligence* 133(PB): 108242.<https://doi.org/10.1016/j.engappai.2024.108242>.
- Halpern-wight, Naylani, Maria Konstantinou, and Alexandros G Charalambides. “Applied Sciences Training and Testing of a Single-Layer LSTM Network for Near-Future Solar Forecasting.” : 1–9.
- Hamiane, Sana, Youssef Ghanou, Hamid Khalifi, and

- Meryam Telmem. 2024. "Comparative Analysis of LSTM, ARIMA, and Hybrid Models for Forecasting Future GDP." *Ingenierie des Systemes d'Information* 29(3): 853–61.
- Hammouri, Abdelaziz I. et al. 2020. "An Improved Dragonfly Algorithm for Feature Selection." *Knowledge-Based Systems* 203: 106131. <https://doi.org/10.1016/j.knosys.2020.106131>.
- Hu, Zhiyuan et al. 2024. "Research on Vehicle Speed Prediction Model Based on Traffic Flow Information Fusion." *Energy* 292(September 2023): 130416. <https://doi.org/10.1016/j.energy.2024.130416>.
- Ibrahim Mohammad Ata, Karimeh et al. 2024. "A Multi-Layer CNN-GRUSKIP Model Based on Transformer for Spatial –TEMPORAL Traffic Flow Prediction." *Ain Shams Engineering Journal*.
- Jalil, Khurram et al. 2024. "Integrative Review of Data Sciences for Driving Smart Mobility in Intelligent Transportation Systems." *Computers and Electrical Engineering* 119(PB): 109624. <https://doi.org/10.1016/j.compeleceng.2024.109624>.
- Jeon, Hobeom, Hyungmin Kim, Dohyung Kim, and Jeahong Kim. 2024. "PASS-CCTV: Proactive Anomaly Surveillance System for CCTV Footage Analysis in Adverse Environmental Conditions." *Expert Systems with Applications* 254(March).
- Kaur, Archana et al. 2023. "Subsequent Risky Driving Behaviors, Recidivism and Crashes

among Drivers with a Traffic Violation: A Scoping Review.” *Accident Analysis and Prevention* 192(July).

Khah, Nasser Keshavarz Faraj, Bahare Salehi, Pooria Kianoush, and Shahab Varkouhi. 2024. “Estimating Elastic Properties of Sediments by Pseudo-Wells Generation Utilizing Simulated Annealing Optimization Method.” *Results in Earth Sciences* 2(February): 100024. <https://doi.org/10.1016/j.rines.2024.100024>.

Khan, Sohrab et al. 2024. “Deep Neural Network and Trust Management Approach to Secure Smart Transportation Data in Sustainable Smart Cities.” 10: 1059–65.

Kim, Hyungmin, Hobeom Jeon, Dohyung Kim, and Jaehong Kim. 2024. “Elevating Urban Surveillance: A Deep CCTV Monitoring System for Detection of Anomalous Events via Human Action Recognition.” *Sustainable Cities and Society* 114(20018513).

Kusuma, Gilang, Garry Kuwanto, Tahir Hashmi, and Herman Widjaja. 2023. “Discrete Komodo Algorithm for Traveling Salesman Problem.” 139.

Lara-montaña, Oscar Daniel, Fernando Israel Gómez-castro, Claudia Gutiérrez-antonio, and Elena Niculina Dragoi. 2024. “Jou Rna LP.” *Computers and Chemical Engineering*: 108987. <https://doi.org/10.1016/j.compchemeng.2024.108987>.

Li, Guanghui et al. 2024. “Review of the Metaheuristic Algorithms in Applications :

- Visual Analysis Based on Bibliometrics.” *Expert Systems With Applications* 255(PD): 124857.<https://doi.org/10.1016/j.eswa.2024.124857>.
- Li, Yan, and Bin Sun. 2024. “An Intelligent Indoor Fire Localization System Combining Dynamic Clustering Algorithm and Particle Swarm Optimization Algorithm.” *Journal of Building Engineering* 98(October): 111180. <https://doi.org/10.1016/j.jobbe.2024.111180>.
- Li, Yanhang et al. 2024. “Improved Control-Oriented Polarization Characteristic Modeling for Proton Exchange Membrane Water Electrolyzer with Adaptive Hunting Game Based Metaheuristic Optimization.” *Energy Conversion and Management* 305(February): 118264. <https://doi.org/10.1016/j.enconman.2024.118264>.
- Lnenicka, Martin, Petr Hervert, and Oldrich Horak. 2024. “Understanding Big Data and Data Protection Measures in Smart City Strategies: An Analysis of 28 Cities.” *Urban Governance* 4(4): 255–73. <https://doi.org/10.1016/j.ugj.2024.12.008>.
- Lu, Jingyi. 2023. “An Efficient and Intelligent Traffic Flow Prediction Method Based on LSTM and Variational Modal Decomposition.” *Measurement: Sensors* 28(June): 100843. <https://doi.org/10.1016/j.measen.2023.100843>.
- Luo, Yong et al. 2024a. “GT-LSTM: A Spatio-Temporal Ensemble Network for Traffic Flow Prediction.” 171(August 2023): 251–62.

- . 2024b. “GT-LSTM: A Spatio-Temporal Ensemble Network for Traffic Flow Prediction.” *Neural Networks* 171(August 2023): 251–62. <https://doi.org/10.1016/j.neunet.2023.12.016>.
- Naheliya, Bharti, Poonam Redhu, and Kranti Kumar. 2024. “MFOA-Bi-LSTM: An Optimized Bidirectional Long Short-Term Memory Model for Short-Term Traffic Flow Prediction.” *Physica A: Statistical Mechanics and its Applications* 634(November 2023): 129448. <https://doi.org/10.1016/j.physa.2023.129448>.
- Oladimeji, Damilola et al. 2023. “Smart Transportation: An Overview of Technologies and Applications.” *Sensors* 23(8): 1–32.
- Opara, Karol, and Jarosław Arabas. 2018. “Comparison of Mutation Strategies in Differential Evolution – A Probabilistic Perspective.” *Swarm and Evolutionary Computation* 39(January): 53–69. <http://dx.doi.org/10.1016/j.swevo.2017.12.007>.
- Oza, Parita et al. 2022. “Image Augmentation Techniques for Mammogram Analysis.” : 1–22.
- Preeti, and Chhavi Rana. 2024. “Artificial Intelligence Based Object Detection and Traffic Prediction by Autonomous Vehicles – A Review.” *Expert Systems with Applications* 255(June 2023).
- Putri, Endah R.M. et al. 2024. “A Deep-Genetic Algorithm (Deep-GA) Approach for High-Dimensional Nonlinear Parabolic Partial Differential Equations.” *Computers and Mathematics with Applications* 154(October

- 2023): 120–27.
<https://doi.org/10.1016/j.camwa.2023.11.022>.
- Ribeiro, David Augusto et al. 2023. “A Novel Deep Deterministic Policy Gradient Model Applied to Intelligent Transportation System Security Problems in 5G and 6G Network Scenarios.” *Physical Communication* 56: 101938.
<https://doi.org/10.1016/j.phycom.2022.101938>.
- Ronariv, Raien et al. 2024. “ScienceDirect Object Detection Algorithms for Car Tracking with Euclidean Object Detection Algorithms for Car Tracking Distance Tracking and YOLO with Euclidean Distance a Tracking and YOLO.” *Procedia Computer Science* 245: 627–36.
<https://doi.org/10.1016/j.procs.2024.10.289>.
- Saputri, Hanis Amalia et al. 2024. “ScienceDirect ScienceDirect Implementation of YOLO v7 Algorithm in Estimating Traffic Implementation of YOLO Flow v7 in Algorithm Malang in Estimating Traffic Flow in Malang.” *Procedia Computer Science* 245(2022): 117–26.
<https://doi.org/10.1016/j.procs.2024.10.235>.
- Saravanan, M., and G. K. Rajini. 2024. “Comprehensive Study on the Development of an Automatic Helmet Violator Detection System (AHVDS) Using Advanced Machine Learning Techniques.” *Computers and Electrical Engineering* 118(December 2023).
- Sattarzadeh, Ali Reza, Ronny J. Kutadinata, Pubudu N. Pathirana, and Van Thanh Huynh. 2023. “A Novel Hybrid Deep Learning Model with ARIMA Conv-LSTM Networks and Shuffle Attention Layer for Short-Term Traffic Flow

Prediction.” *Transportmetrica A: Transport Science*.

<https://doi.org/10.1080/23249935.2023.2236724>.

Selmy, Hend A., Hoda K. Mohamed, and Walaa Medhat. 2024. “Big Data Analytics Deep Learning Techniques and Applications: A Survey.” *Information Systems* 120(October 2023): 102318.
<https://doi.org/10.1016/j.is.2023.102318>.

Shaaban, Khaled, Mazen Elamin, and Mohammed Alsoub. 2021. “Intelligent Transportation Systems in a Developing Country: Benefits and Challenges of Implementation.” *Transportation Research Procedia* 55(2019): 1373–80.
<https://doi.org/10.1016/j.trpro.2021.07.122>.

Shamta, Ibrahim, Funda Demir, and Batikan Erdem Demir. 2024. “Predictive Fault Detection and Resolution Using YOLOv8 Segmentation Model: A Comprehensive Study on Hotspot Faults and Generalization Challenges in Computer Vision.” *Ain Shams Engineering Journal* (October).

Shaygan, Maryam, Fatemeh Banani Ardecani, and Mark Nejad. 2025. “Optimizing Mixed Traffic Environments with Shared and Private Autonomous Vehicles: An Equilibrium Analysis of Entrance Permit and Tradable Credit Strategies.” *Transportation Research Part E: Logistics and Transportation Review* 194(December 2024).

Suyanto, Suyanto, Alifya Aisyah Ariyanto, and Alifya Fatimah Ariyanto. 2022a. “Komodo

- Mlipir Algorithm.” *Applied Soft Computing* 114: 108043.
<https://doi.org/10.1016/j.asoc.2021.108043>.
- . 2022b. “Komodo Mlipir Algorithm ☆.” *Applied Soft Computing* 114: 108043.
<https://doi.org/10.1016/j.asoc.2021.108043>.
- Tan, Kian Long, Chin P O O Lee, Kalaiarasi Sonai, and Muthu Anbananthen. 2022. “RoBERTa-LSTM : A Hybrid Model for Sentiment Analysis With Transformer and Recurrent Neural Network.” *IEEE Access* 10: 21517–25.
- Tang, Jun, Caixian Ye, Xianlai Zhou, and Lijun Xu. 2024. “YOLO-Fusion and Internet of Things: Advancing Object Detection in Smart Transportation.” *Alexandria Engineering Journal* 107(August): 1–12.
<https://doi.org/10.1016/j.aej.2024.09.012>.
- Thakkar, Ankit, Nandish Kikani, and Rebakah Geddam. 2024. “Fusion of Linear and Non-Linear Dimensionality Reduction Techniques for Feature Reduction in LSTM-Based Intrusion Detection System.” 154(August 2023).
- Wang, Jing Doo, and Chayadi Oktomy Noto Susanto. 2023. “Traffic Flow Prediction with Heterogenous Data Using a Hybrid CNN-LSTM Model.” *Computers, Materials and Continua* 76(3): 3097–3112.
- Wang, Ke et al. 2021. “A Hybrid Deep Learning Model with 1DCNN-LSTM-Attention Networks for Short-Term Traffic Flow Prediction.” *Physica A: Statistical Mechanics and its Applications* 583: 126293.

<https://doi.org/10.1016/j.physa.2021.126293>.

- Wu, Duidi et al. 2024. "Computer Vision-Based Intelligent Elevator Information System for Efficient Demand-Based Operation and Optimization." *Journal of Building Engineering* 81(November 2023).
- Wu, Tianyong, and Youkou Dong. 2023. "YOLO-SE: Improved YOLOv8 for Remote Sensing Object Detection and Recognition." *Applied Sciences (Switzerland)* 13(24).
- Xing, Hang, An Chen, and Xuan Zhang. 2023. "RL-GCN: Traffic Flow Prediction Based on Graph Convolution and Reinforcement Learning for Smart Cities." *Displays* 80(September).
- Xu, Shizhou, Mengjie Zhang, Jingyu Chen, and Yiming Zhong. 2024. "YOLO-HyperVision: A Vision Transformer Backbone-Based Enhancement of YOLOv5 for Detection of Dynamic Traffic Information." *Egyptian Informatics Journal* 27(September 2023).
- Yang, Chao-lung et al. 2023. "LSTM-Based Framework with Metaheuristic Optimizer for Manufacturing Process Monitoring." 83(October): 43–52.
- Yang, Suorong et al. 2022. "Image Data Augmentation for Deep Learning: A Survey." <http://arxiv.org/abs/2204.08610>.
- Yanming, Hu, Xu Xuecai, and Data Science. 2024. "Na l P Re r F."
- Yousif, Ibrahim, Liam Burns, Fadi El Kalach, and Ramy Harik. 2024. "Leveraging Computer Vision towards High-Efficiency Autonomous

Industrial Facilities.” *Journal of Intelligent Manufacturing*. <https://doi.org/10.1007/s10845-024-02396-1>.

Yun, Sangdoo. 2019. “CutMix.” *Iccv*: 6023–32.

Zhang, Gongquan, Jieling Jin, Fangrong Chang, and Helai Huang. 2024. “Real-Time Traffic Conflict Prediction at Signalized Intersections Using Vehicle Trajectory Data and Deep Learning.” *International Journal of Transportation Science and Technology* (xxxx).

Zhang, Wensong et al. 2024. “Engineering Applications of Artificial Intelligence A Traffic-Weather Generative Adversarial Network for Traffic Flow Prediction for Road Networks under Bad Weather.” *Engineering Applications of Artificial Intelligence* 137(PA): 109125. <https://doi.org/10.1016/j.engappai.2024.109125>

Zhao, Ke et al. 2023. “Short-Term Traffic Flow Prediction Based on VMD and IDBO-LSTM.” *IEEE Access* 11(August): 97072–88.

———. 2024. “Short-Term Traffic Flow Prediction Based on Hybrid Decomposition Optimization and Deep Extreme Learning Machine.” *Physica A: Statistical Mechanics and its Applications* 647(June): 129870. <https://doi.org/10.1016/j.physa.2024.129870>.

Zhao, Ruixin et al. 2024. “Enhancing Autonomous Driving Safety: A Robust Traffic Sign Detection and Recognition Model TSD-YOLO.” *Signal Processing* 225(July): 109619. <https://doi.org/10.1016/j.sigpro.2024.109619>.

Zheng, Jianhu, Minghua Wang, and Mingfang Huang. 2024. "Exploring the Relationship between Data Sample Size and Traffic Flow Prediction Accuracy." *Transportation Engineering* 18(September): 100279. <https://doi.org/10.1016/j.treng.2024.100279>.