IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 14, No. 4, August 2025, pp. 3343~3353
ISSN: 2252-8938, DOI: 10.11591/ijai.v14.i4.pp3343-3353

Traffic flow prediction using long short-term memory-Komodo Mlipir algorithm: metaheuristic optimization to multi-target vehicle detection

Imam Ahmad Ashari[1,4], Wahyul Amien Syafei[2], Adi Wibowo[3]
[1]Doctoral Program of Information Systems, Universitas Diponegoro, Semarang, Indonesia
[2]Faculty of Engineering, Universitas Diponegoro, Semarang, Indonesia
[3]Faculty of Science and Mathematics, Universitas Diponegoro, Semarang, Indonesia
[4]Faculty of Science and Technology, Universitas Harapan Bangsa, Purwokerto, Indonesia

Article Info

Article history:

Received Dec 12, 2024
Revised Jun 24, 2025
Accepted Jul 13, 2025

Keywords:

Komodo Mlipir algorithm
Long short-term memory
Metaheuristic optimization
Multi-target vehicle detection
Traffic flow prediction

ABSTRACT

Multi-target vehicle detection in urban traffic faces challenges such as poor lighting, small object sizes, and diverse vehicle types, impacting traffic flow prediction accuracy. This study introduces an optimized long short-term memory (LSTM) model using the Komodo Mlipir algorithm (KMA) to enhance prediction accuracy. Traffic video data are processed with YOLO for vehicle classification and object counting. The LSTM model, trained to capture traffic patterns, employs parameters optimized by KMA, including learning rate, neuron count, and epochs. KMA integrates mutation and crossover strategies to enable adaptive selection in global and local searches. The model's performance was evaluated on an urban traffic dataset with uniform configurations for population size and key LSTM parameters, ensuring consistent evaluation. Results showed LSTM-KMA achieved a root mean square error (RMSE) of 14.5319, outperforming LSTM (16.6827), LSTM-improved dung beetle optimization (IDBO) (15.0946), and LSTM-particle swarm optimization (PSO) (15.0368). Its mean absolute error (MAE), at 8.7041, also surpassed LSTM (9.9903), LSTM-IDBO (9.0328), and LSTM-PSO (9.0015). LSTM-KMA effectively tackles multi-target detection challenges, improving prediction accuracy and transportation system efficiency. This reliable solution supports real-time urban traffic management, addressing the demands of dynamic urban environments.

Corresponding Author:

Imam Ahmad Ashari
Doctoral Program of Information Systems, Universitas Diponegoro
Semarang, Indonesia
Email: imamahmad@students.undip.ac.id

1. INTRODUCTION

With advancements in communication technology and computer science, intelligent transportation systems (ITS) have assumed an increasingly significant role in daily life [1]. Smart transportation has become a cornerstone in the development of technology-based ITS to meet the evolving needs of urban societies [2]. It refers to an approach that integrates modern technology into transportation systems to enhance urban mobility efficiency [3]. In the context of smart cities, cutting-edge technologies such as the internet of things (IoT), data analytics, and artificial intelligence (AI) serve as foundational pillars for creating intelligent and interconnected transportation ecosystems [4]. Smart mobility has become an integral part of daily life, with 40% of the global population traveling for at least one hour each day [5]. By integrating technologies such as computer vision, AI, and ITS, cities can more accurately detect traffic conditions, identify vehicle types, and predict congestion. This integration helps address urbanization challenges, such as pollution, traffic accidents, and excessive resource consumption [6].

Traffic flow prediction is a critical element in ITS as it provides valuable insights for traffic control, route planning, and operational management [7]. Traditional traffic flow prediction models often fail to adequately account for the complex and dynamic characteristics of urban traffic networks [8]. With the acceleration of urbanization and advancements in ITS, short-term traffic flow prediction has emerged as an increasingly significant area of research [9]. Accurate predictions offer substantial benefits, including optimized traffic planning, improved road utilization, reduced congestion, fewer traffic accidents, and decreased environmental pollution [10].

Accurate traffic flow prediction requires the efficient extraction and analysis of large-scale urban traffic data, including the appropriate selection of data sample sizes. Technological advancements, such as roadside closed-circuit television (CCTV) cameras and unmanned aerial vehicles (UAVs), provide new video data that enable more comprehensive traffic information collection through computer vision techniques [11]. These advancements support accident-based safety analysis and facilitate real-time traffic control, route guidance, policy formulation, and more effective traffic allocation. Together, these efforts enhance traffic efficiency and improve the quality of urban life [12]. In practice, computer vision models such as YOLO and its advancements are widely applied to detect and analyze urban traffic conditions [13]–[18].

In ITS, traditional object detection algorithms face various challenges, particularly in dealing with complex environments and varying lighting conditions. These challenges become more significant when detecting small objects or analyzing multimodal data [16]. To address these limitations, enhancing data quality and diversity through augmentation techniques is a common approach [19]. Previous research has demonstrated that combining object detection with long short-term memory (LSTM) algorithms can effectively predict traffic volume [20]. Additionally, studies have proposed the development of new models leveraging and optimizing LSTM, which has proven effective in handling time-series data and improving the accuracy of urban traffic density predictions [20]–[25]. Recent trends suggest an increasing focus on optimizing LSTM parameters through metaheuristic approaches to improve traffic prediction performance [22], [26]–[28]. Such an approach is anticipated to tackle the challenges of creating more reliable and efficient predictive models for various traffic conditions.

The Komodo mlipir optimization algorithm (KMA) draws inspiration from two unique phenomena: the behavior of Komodo dragons native to East Nusa Tenggara, Indonesia, and the traditional Javanese walking style known as *mlipir* [29]. In the context of the traveling salesman problem (TSP), KMA has exhibited superior performance compared to algorithms like the dragonfly algorithm (DKA), ant colony optimization (ACO), particle swarm optimization (PSO), genetic algorithm (GA), black hole (BH), dynamic tabu search algorithm (DTSA), and discrete jaya algorithm (DJAYA) [30]. In our proposed research, LSTM is combined with KMA for traffic volume prediction. The LSTM-KMA model is then compared with the standard LSTM and other state-of-the-art combinations, namely LSTM-improved dung beetle optimization (IDBO) and LSTM-PSO. Previous studies have shown that LSTM-IDBO outperforms methods such as gray wolf optimization (GWO), sparrow optimization algorithm (SSA), whale optimization algorithm (WOA), and nighthawk optimization (NGO) [26]. Similarly, LSTM-PSO has proven superior to methods like standard LSTM, random forest regression (RFR), k-nearest regression (KNR), and decision tree regression (DTR) [28].

The main problem addressed in this study is the low accuracy in predicting complex and dynamic traffic volumes, particularly under real-world conditions that often involve challenges such as poor lighting, occlusions, and diverse vehicle types. To address this issue, the study aims to develop a traffic prediction model that integrates the LSTM algorithm with the KMA as an optimization method, supported by real-time vehicle detection data using YOLO. This research specifically focuses on how the integration of KMA can improve the predictive accuracy of LSTM in modeling dynamic traffic volumes, and evaluates the potential implementation of the YOLO-LSTM-KMA system under real traffic conditions. The main contribution of this study is the development of an intelligent predictive model capable of improving traffic flow prediction accuracy, offering both theoretical contributions in the field of optimization and time-series forecasting, and practical contributions in supporting data-driven decision-making within ITS.

2. METHOD

2.1. Vehicle object detection

Data collection was conducted using YOLO as the object detection model for identifying vehicles in traffic. Specifically, the YOLOv8n model was used because of its high-performance ability to detect vehicles in complex traffic conditions. To improve detection accuracy, multi-augmentation techniques were applied, combining scaling, zoom-in, brightness adjustment, color jitter, and noise injection. Table 1 presents the specific values for each augmentation technique used in the study.

Table 1. Augmentation values

No	Aug	Value	Augmentation factor (image) 1	2	3	References
1	Brightness adjustment	Brightness factor	-	0.8	1.2	[31]
2	Color jitter	(Brightness, contrast, saturation) and hue	-	Rand (0.6,1.4) and Rand (-0.1,0.1)	Rand (0.6,1.4) and Rand (-0.1,0.1)	[32]
3	Noise injection	Gaussian noise	-	Rand (0, 0.1)	Rand (0, 0.1)	[33]
4	Scaling	Scale image	-	Rand (0.8, 1.2)	Rand (0.8, 1.2)	[34]
5	Zoom in	Zoom in	-	1.2	1.5	[35]

In the image augmentation process summarized in Table 1, brightness adjustment was performed with a brightness factor of 0.8 for image 2 and 1.2 for image 3. For the color jitter technique, the brightness, contrast, and saturation factors were randomized within the range of 0.6 to 1.4, while the hue factor was randomized between -0.1 and 0.1. Noise injection utilized Gaussian noise with values randomized between 0 and 0.1 for both images. The scaling technique was applied with a factor range of 0.8 to 1.2 for both image 2 and image 3, while the zoom-in technique utilized a factor of 1.2 for image 2 and 1.5 for image 3. This combination of values was designed to create significant image variations, thereby improving the model's performance under diverse conditions.

Based on the conducted experiments, YOLOv8n outperformed YOLOv9t, achieving the highest mAP50-95 value of 0.536. A detailed performance analysis is presented in a manuscript titled "boosting real-time vehicle detection in urban traffic using a novel multi-augmentation". The experimental results identified the best-performing model, named best.pt, as the foundation for the vehicle detection process in this study. The model workflow is depicted in Figure 1, detailing the steps from data preprocessing to numeric feature extraction.

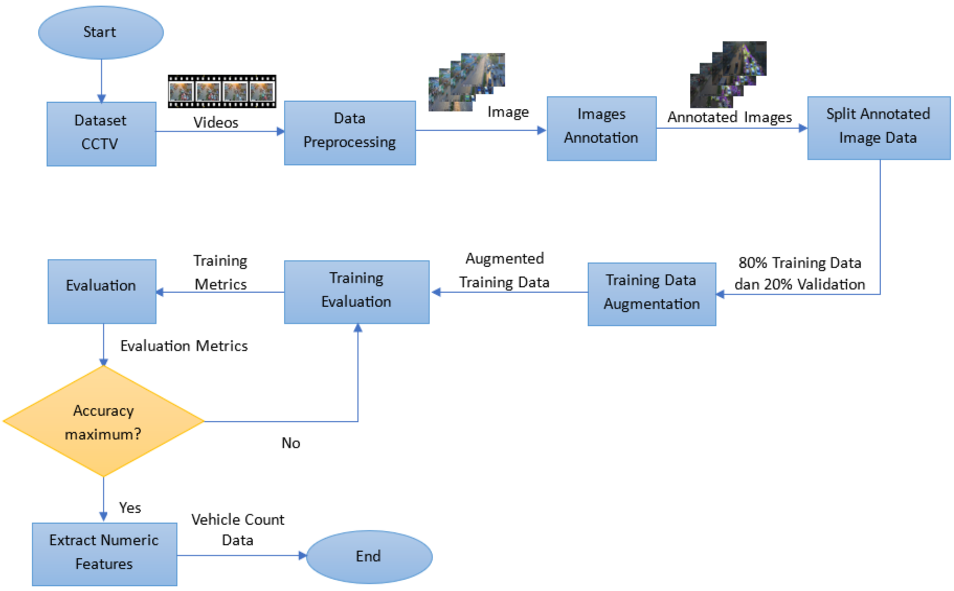

Figure 1. Numerical feature extraction process from YOLO model

The model workflow, as illustrated in Figure 1, begins with the collection of video data from traffic CCTV recordings. This video data is processed through a preprocessing stage where it is converted into individual frames for further analysis. Each frame is manually annotated using the Roboflow application to label vehicle objects, which include motorcycles, cars, trucks, and buses. The annotation process involved creating four vehicle classes and drawing bounding boxes around each object in every frame. In total, the dataset contains 720 images with 45,347 annotations, consisting of 31,481 motorcycles, 12,402 cars, 1,184 trucks, and 280 buses. The dataset is divided into two parts: 80% for training and 20% for validation [36], [37]. This 80:20 split is commonly used in machine learning experiments to ensure that the model has sufficient data to learn patterns during training while maintaining an adequate portion of unseen data for unbiased validation, allowing for accurate evaluation of the model’s generalization ability. The training subset includes 22,136 motorcycles, 8,804 cars, 839 trucks, and 199 buses, while the validation subset contains 9,345 motorcycles, 3,598 cars, 345 trucks, and 81 buses.

To increase data diversity and improve model generalization, augmentation techniques were applied to the training dataset. These techniques include scaling, zoom-in, brightness adjustment, color jitter, and noise injection. Each technique was applied using two parameter values, resulting in a tenfold increase in the amount of training data. The original training dataset consists of 576 images without augmentation, while the augmented dataset consists of 5,760 images, as shown in Table 1. The YOLO model was trained using this enhanced dataset, and its performance was evaluated periodically using the mAP50-95 metric. If the model did not meet the desired accuracy threshold, training was continued. Once the best-performing model was obtained, it was used to detect vehicles in each frame and predict their classes. The detection results were then converted into numerical features, such as vehicle counts by type, which were further processed into traffic flow data for subsequent analysis.

2.2. Long short-term memory-Komodo Mlipir algorithm

The integration of LSTM and KMA leverages the strengths of each method in data analysis and optimization. LSTM is highly effective at capturing temporal patterns in time-series data, making it suitable for both short-term and long-term prediction tasks [38]. Previous studies have shown that hyperparameter optimization using metaheuristic approaches often yields better results compared to conventional methods, further reinforcing the advantage of combining these techniques to improve model performance [39]. The incorporation of KMA in this approach is anticipated to surpass the performance of other metaheuristic algorithms. The integration of LSTM and KMA not only accelerates the optimization process but also enhances the likelihood of identifying optimal hyperparameter configurations, thereby significantly improving the performance of the LSTM model in traffic flow prediction applications. This proposed approach is depicted in Figure 2.

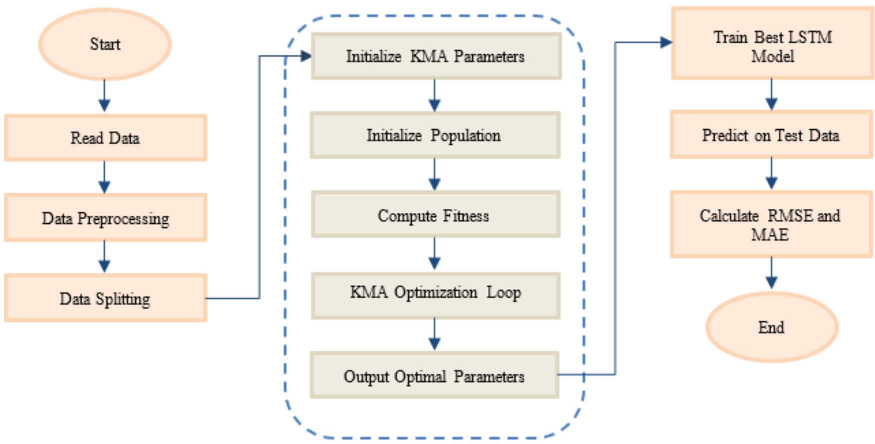

Figure 2. Proposed model

Figure 2 illustrates the LSTM-KMA computation process, beginning with data reading and preprocessing, followed by dividing the dataset into training and testing sets, allocating 80% for training and 20% for testing [40], [41]. The KMA is then initialized with specific parameters. This step involves initializing a population of candidate solutions and applying crossover and mutation operations [42]. The fitness of each candidate solution is evaluated to determine the suitability of the parameters for the LSTM model using the mean absolute error (MAE) metric. The LSTM parameters being optimized include the number of neurons, learning rate, and epochs [26]. KMA iteratively updates the candidate solutions through an optimization loop until the optimal parameters are identified. The optimized parameters are then applied to train the final LSTM model. The trained model is subsequently tested using the test data, with root mean square error (RMSE) and MAE calculated as accuracy measures for the predictions. The concept of KMA in LSTM parameter optimization is illustrated through the pseudocode presented in Algorithm 1.

Algorithm 1: Komodo Mlipir for optimizing LSTM parameters
```
Input:
   Maximum number of iterations (T), population size (n).
   Range of LSTM parameters to be optimized (neurons, learning rate, and epochs).
Step 1: Initialization
   Initialize a population of n individuals (komodo) with random combinations of LSTM
   parameters.
```

```
    Each individual q in the population is represented as Pq = [Xq,Yq,Zq], where Xq,Yq, and Zq
    respectively denote neurons, learning rate, and epochs.
Step 2: Fitness Evaluation
    Evaluate the initial fitness of each candidate solution by measuring the LSTM's
    performance on the validation dataset.
    Use the objective function: Minimize F=MAE.
    Sort the individuals based on their fitness scores and categorize them into three
    groups:
      -  Large males (elite, top performers)
      -  Females (moderate performance)
      -  Small males (low performers)
Step 3: Main Loop
    While (t ≤ T):
      1.  Reassess each individual's fitness score.
      2.  Update their positions as follows:
          -  Large males: Adjust positions using exploitation strategies.
          -  Females:
             -  Mate with the top-performing large male using exploitation method.
             -  Reproduce asexually via parthenogenesis using exploration strategies.
             -  Small males: Explore the solution space randomly using exploration
                strategies.
      3.  Apply selection process:
          -  Retain the best-performing individual (elitism).
          -  Improve weaker individuals using update strategy in equation.
      4.  Increment the iteration count (t = t + 1).
    End While
Step 4: Output the Best Solution
    Output the best LSTM parameters (P_best) and the best fitness value (F_best).
Output:
    Optimal LSTM parameters
```

Algorithm 1 is the pseudocode of the KMA used to optimize the parameters of the LSTM model. This algorithm aims to find the best combination of neurons, learning rate, and number of epochs by minimizing the MAE. The step-by-step procedure is outlined in the pseudocode above. To facilitate understanding, the workflow of this algorithm is also illustrated in the flowchart, as shown in Figure 3.

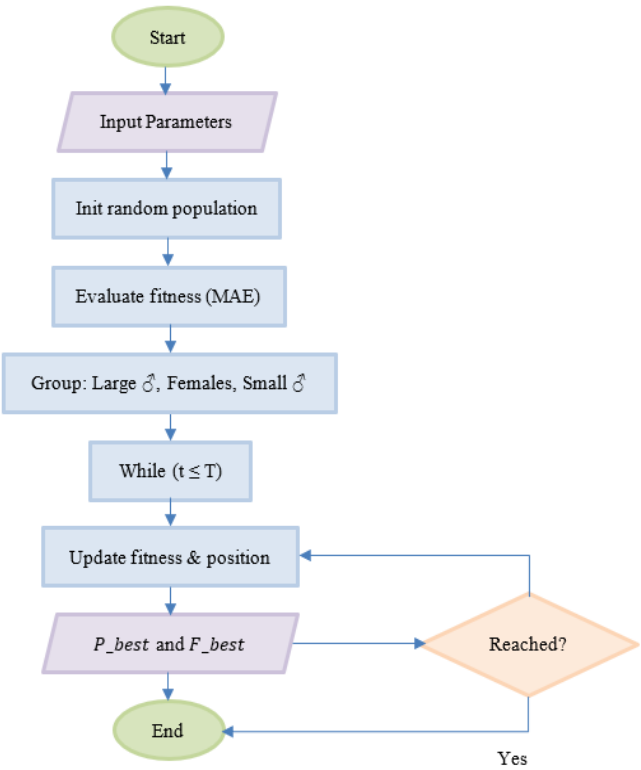

Figure 3. Optimization workflow of the LSTM model using KMA

3. RESULTS AND DISCUSSION

3.1. Data and environment

In this study, the data used for traffic flow prediction was collected from CCTV cameras installed in Fatmawati, Semarang City. The data collection period spanned from December 19, 2023, to February 15, 2024. Data was gathered by extracting images from recorded videos at 5-minute intervals. The frame interval was determined by multiplying the frames per second (FPS) by 60 and the specified number of minutes. With an FPS of 25, the resulting frame interval was 25×60×5=7,500 frames. This means the program extracted one image for every 7,500 frames. From this extraction process, a total of 720 images were obtained. A total of 45,347 annotations were generated from these images, comprising 31,481 motorcycles, 12,402 cars, 1,184 trucks, and 280 buses. In addition to vehicle types, date and time information was also extracted from the dataset. The dataset was then divided into two subsets: a training subset and a validation subset, to facilitate model training and evaluation. The distribution of vehicle annotations in the training set includes 22,136 motorcycles, 8,804 cars, 839 trucks, and 199 buses. The validation set consists of 9,345 motorcycles, 3,598 cars, 345 trucks, and 81 buses. This structured data collection and preprocessing process provides a solid foundation for developing traffic flow prediction models, ensuring that the dataset is representative and well-annotated for effective model training and evaluation.

This study utilized Google Colab Pro for experimental configuration. Google Colab offers cloud-based and open-source computing services to handle the extensive processing requirements needed for model training [43]. The runtime environment included Python 3 and an NVIDIA T4 GPU. The programming language utilized was Python 3.10.12, and the PyTorch framework version 2.3.0 was implemented with CUDA version 12.1 support.

3.2. Parameter settings and model optimization

In this study, the parameters for the metaheuristic method were standardized by setting the population size to 30, as referenced in previous studies [26]. The parameter ranges optimized for the LSTM model include the number of neurons (300-500), learning rate (0.001-0.01), and number of epochs (1-150). These ranges were initially adopted based on prior literature and then further refined through multiple trial-and-error experiments to obtain optimal performance. For the conventional LSTM model, the analysis was conducted using the highest values in each range-500 neurons, a learning rate of 0.01, and 150 epochs. Detailed configurations of other parameters used for each model can be found in Table 2.

Table 2 presents the parameter settings used for various algorithms in optimizing the LSTM model. For LSTM-KMA, the size of the population involved in the selection process is set to 10. In the LSTM-IDBO algorithm, the coefficient of variation is set to 0.1, and the scaling parameter for balancing exploration and exploitation is set to 0.5. The LSTM-PSO algorithm uses a self-learning factor of 1.5 and a group learning factor of 2.

Table 2. Parameter setting of the various algorithms

Algorithm	Parameters	Settings	Reference
LSTM-KMA	Size of population involved in selection	10	[42]
IDBO-LSTM	Coefficient of variation	0.1	[26]
	Scale or parameter for setting exploration and exploitation	0.5	
LSTM-PSO	Self-learning factor	1.5	[28]
	Group learning factor	2	

3.3. Evaluation criteria

The appropriate performance evaluation metrics for continuous data obtained in real-time are regression loss functions [44]. Therefore, the performance evaluation metrics used in this study are RMSE and MAE. RMSE reflects the degree of deviation of predicted values from actual values. The formula for RMSE is provided in (1) [45]. MAE represents the mean of absolute errors, where absolute error is the difference between predicted and actual values. A low MAE value indicates that the model predicts values close to the actual values. The formula for MAE is provided in (2) [26].

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

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

RMSE and MAE are two evaluation metrics used to measure the prediction errors of a model. In the RMSE formula, the difference between the actual value (y_i) and the predicted value ($\hat{y}_i$) is squared to calculate $(y_i - \hat{y}_i)^2$, giving greater weight to larger errors, and then the square root is taken. RMSE provides additional insights by reflecting the degree of deviation between predicted values and actual values, being more sensitive to large errors. MAE has a similar formula but with a different approach. In this formula, n represents the total number of data points or observations in the dataset, indicating the number of data points analyzed. y_i is the actual value of the i-th data point, representing the true data to be predicted, such as the actual number of vehicles in traffic prediction. On the other hand, $\hat{y}_i$ is the predicted value generated by the model for the i-th data point, reflecting the estimated number of vehicles. The absolute difference between actual and predicted values is calculated as $|y_i - \hat{y}_i|$, providing an error measure without regard to error direction. All these absolute differences are summed and divided by the total number of data points (n) to yield the MAE.

Therefore, RMSE and MAE provide an overall measure of how close the model's predictions are to the actual values. The prediction results of the LSTM, LSTM-KMA, LSTM-IDBO, and LSTM-PSO models are compared with the actual data. The prediction outcomes of the utilized models are shown in Figure 4. Figure 4 illustrates the comparison between the actual traffic flow data (TRUE) and the predicted results from several models, namely LSTM, LSTM-KMA, LSTM-IDBO, and LSTM-PSO. The graph shows how each model's predictions align with or deviate from the actual traffic flow values over time, highlighting the accuracy and performance differences among the models.

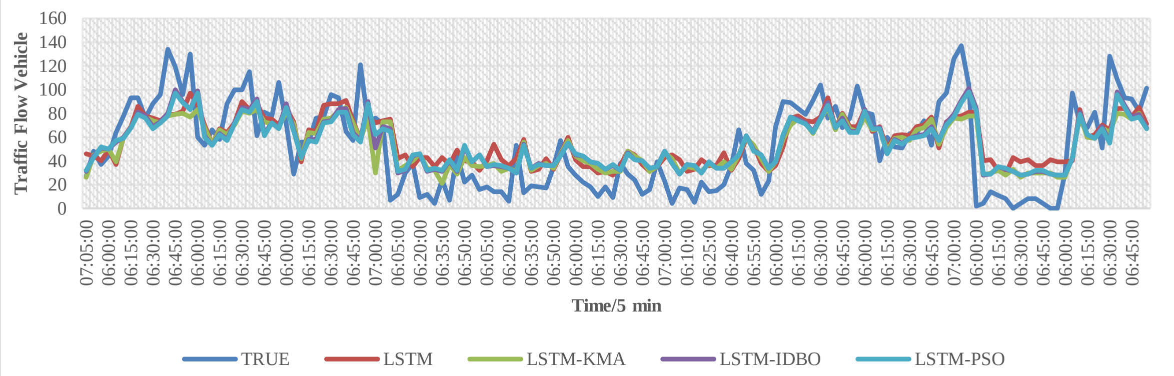

Figure 4. Prediction results of each model

3.4. Results and performance analysis

The developed model, LSTM-KMA, is compared with the baseline LSTM model. In addition, two other LSTM models optimized using metaheuristic algorithms, namely LSTM-IDBO and LSTM-PSO, are also included in the comparison. The performance of each model based on the RMSE is shown in Figure 5. A separate comparison using the MAE metric is presented in Figure 6. This figure highlights the average prediction error for each model. The lower the MAE value, the closer the model's predictions are to the actual data. To support the visual comparison, both RMSE and MAE values are summarized in Table 3. This table provides a clearer view of each model’s numerical performance. It complements the graphical results shown in the Figures 5 and 6.

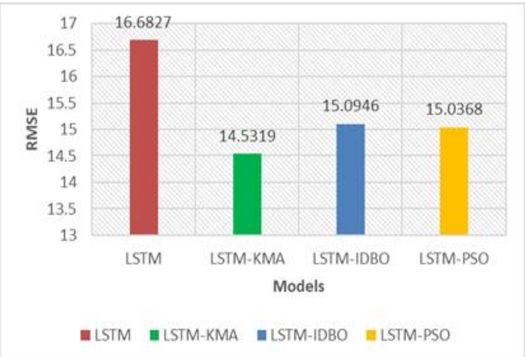

Figure 5. RMSE comparison of models

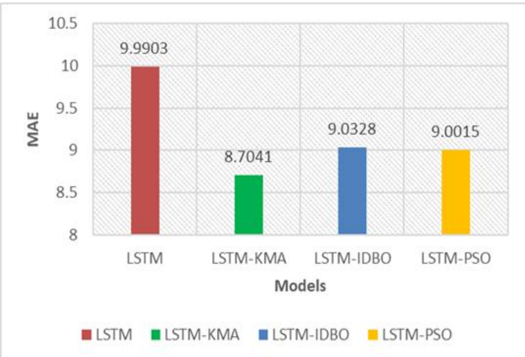

Figure 6. MAE comparison of models

Table 3. The RMSE and MAE values of the models were evaluated individually

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

Table 3 shows that the LSTM-KMA model achieves the lowest RMSE value of 14.5319, indicating the best performance compared to the other models. In contrast, the baseline LSTM model records the highest RMSE value of 16.6827, indicating the lowest prediction accuracy. The LSTM-IDBO and LSTM-PSO models achieve RMSE values of 15.0946 and 15.0368, respectively, demonstrating improved performance compared to the baseline LSTM but still falling short of LSTM-KMA. In terms of MAE, LSTM-KMA also demonstrates the best performance with the lowest value of 8.7041, compared to the baseline LSTM (9.9903), LSTM-IDBO (9.0328), and LSTM-PSO (9.0015). Based on this analysis, optimization using the KMA has proven to be the most effective method for enhancing the performance of the LSTM model in terms of both RMSE and MAE, making it the recommended approach in this study.

3.5. Challenges

Object detection using YOLO on this dataset faces several key challenges, primarily due to variations in lighting and traffic density. Light reflections, poor illumination, and high traffic congestion significantly reduce detection accuracy. Additionally, the presence of multiple object types in a single frame such as motorcycles, cars, trucks, and buses makes it difficult for the model to distinguish overlapping objects, especially for less dominant classes like trucks and buses. Meanwhile, the use of metaheuristic algorithms to optimize LSTM parameters yields better prediction accuracy. However, the drawback lies in the longer runtime compared to conventional methods, as the search for optimal parameters involves complex and iterative processes.

3.6. Practical implications for intelligent transportation systems deployment

The proposed YOLO-LSTM-KMA framework demonstrates promising potential for real-world deployment in ITS. By integrating real-time object detection with time-series traffic prediction, this approach supports automated traffic monitoring and data-driven decision-making. However, several practical aspects must be considered:

– Scalability: the framework is designed to handle large volumes of traffic video data, making it suitable for deployment in urban environments with high traffic density. However, the annotation and training process still require considerable effort, which may need automation or semi-supervised techniques for broader scalability.
– Computational requirements: real-time detection using YOLO and prediction with LSTM-KMA demands sufficient computational resources, particularly during model training and optimization. Deployment in the field would require edge computing or cloud-based infrastructure to meet latency constraints, especially for continuous traffic flow analysis.
– Integration challenges: integrating this model into existing ITS infrastructure may involve challenges such as data compatibility, synchronization across sensors and cameras, and ensuring reliability in variable conditions (e.g., weather, lighting, and occlusion). Robust preprocessing and adaptive retraining strategies could help mitigate these issues.

3.7. Comparison with related studies

The results of this study were compared with several existing approaches in the literature:

– Baseline and conventional models: traditional LSTM models often struggle with optimizing hyperparameters effectively, leading to suboptimal predictions. The integration of KMA in this study outperforms the standard LSTM by achieving higher accuracy and better generalization, particularly under complex traffic scenarios.
– Metaheuristic-based models: compared to other metaheuristic-integrated models such as LSTM-IDBO and LSTM-PSO, the proposed LSTM-KMA model provides competitive or superior performance in terms of prediction accuracy. However, like other metaheuristic approaches, it incurs a higher computational cost due to its iterative search mechanism.
– Advancements and differences: unlike previous works that focus either on detection or prediction alone, this study presents a complete pipeline from real-time vehicle detection to traffic flow prediction. This integrated design contributes to improved performance and practical applicability for ITS, aligning with

the direction of recent studies while introducing a novel optimization algorithm tailored to traffic data characteristics.

Overall, the study contributes to bridging the gap between academic models and real-world ITS implementation by addressing both detection accuracy and prediction robustness, while acknowledging the trade-offs in computation and integration complexity.

4. CONCLUSION

Multi-target vehicle detection in urban traffic faces significant challenges, including poor lighting, small object sizes, and variations in vehicle types, all of which affect the accuracy of traffic flow predictions. To address these challenges, this study proposes the use of a LSTM model optimized with the KMA. The analysis results show that the LSTM-KMA model achieves the lowest RMSE of 14.5319, outperforming the baseline LSTM (16.6827), LSTM-IDBO (15.0946), and LSTM-PSO (15.0368). Furthermore, LSTM-KMA also delivers the best performance based on the MAE, with the lowest value of 8.7041, superior to the baseline LSTM (9.9903), LSTM-IDBO (9.0328), and LSTM-PSO (9.0015). This demonstrates that optimization using KMA significantly improves the accuracy of the LSTM model's predictions when addressing the complexity of multi-target vehicle detection in urban traffic. Thus, this research makes a significant contribution to the development of predictive models that not only address the challenges in multi-target vehicle detection but also support real-time traffic management systems.

ACKNOWLEDGEMENTS

We would like to express our gratitude to the Department of Transportation of Semarang City for their assistance and support in providing the valuable dataset used in this research.

FUNDING INFORMATION

The authors affirm that this research was conducted with personal funding and did not receive any financial support from external institutions, funding agencies, or grant programs. All expenses related to the execution of this study were borne solely by the authors.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Imam Ahmad Ashari	✓	✓	✓	✓	✓	✓		✓	✓		✓		✓	
Wahyul Amien Syafei	✓	✓			✓	✓	✓			✓		✓		✓
Adi Wibowo		✓								✓				✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare that they have no known competing financial interests, personal relationships, or non-financial competing interests that could have appeared to influence the work reported in this paper. Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study were obtained from the Department of Transportation of Semarang City. Restrictions apply to the availability of these data, which were used under permission for this study. Data are available from the corresponding author upon reasonable request and with the permission of the Department of Transportation of Semarang City.

REFERENCES

[1] W. Zhang, R. Yao, Y. Yuan, X. Du, L. Wang, and F. Sun, "A traffic-weather generative adversarial network for traffic flow prediction for road networks under bad weather," *Engineering Applications of Artificial Intelligence*, vol. 137, 2024, doi: 10.1016/j.engappai.2024.109125.

[2] D. Oladimeji, K. Gupta, N. A. Kose, K. Gundogan, L. Ge, and F. Liang, "Smart transportation: an overview of technologies and applications," *Sensors*, vol. 23, no. 8, pp. 1–32, 2023, doi: 10.3390/s23083880.

[3] K. M. Almatar, "Smart transportation planning and its challenges in the Kingdom of Saudi Arabia," *Sustainable Futures*, vol. 8, no. December 2023, 2024, doi: 10.1016/j.sftr.2024.100238.

[4] S. Khan, S. Khan, A. Sulaiman, M. S. Al Reshan, H. Alshahrani, and A. Shaikh, "Deep neural network and trust management approach to secure smart transportation data in sustainable smart cities," *ICT Express*, vol. 10, no. 5, pp. 1059–1065, 2024, doi: 10.1016/j.icte.2024.08.006.

[5] K. Jalil, Y. Xia, Q. Chen, M. N. Zahid, T. Manzoor, and J. Zhao, "Integrative review of data sciences for driving smart mobility in intelligent transportation systems," *Computers and Electrical Engineering*, vol. 119, 2024, doi: 10.1016/j.compeleceng.2024.109624.

[6] E. Dilek and M. Dener, "Computer vision applications in intelligent transportation systems: a survey," *Sensors*, vol. 23, no. 6, 2023, doi: 10.3390/s23062938.

[7] H. Chi, Y. Lu, C. Xie, W. Ke, and B. Chen, "Spatio-temporal attention based collaborative local–global learning for traffic flow prediction," *Engineering Applications of Artificial Intelligence*, vol. 139, 2025, doi: 10.1016/j.engappai.2024.109575.

[8] H. Xing, A. Chen, and X. Zhang, "RL-GCN: traffic flow prediction based on graph convolution and reinforcement learning for smart cities," *Displays*, vol. 80, 2023, doi: 10.1016/j.displa.2023.102513.

[9] C. Ma, Y. Hu, and X. Xu, "Hybrid deep learning model with VMD-BiLSTM-GRU networks for short-term traffic flow prediction," *Data Science and Management*, Nov. 2024, doi: 10.1016/j.dsm.2024.10.004.

[10] J. Zheng, M. Wang, and M. Huang, "Exploring the relationship between data sample size and traffic flow prediction accuracy," *Transportation Engineering*, vol. 18, 2024, doi: 10.1016/j.treng.2024.100279.

[11] M. Abdel-Aty, Z. Wang, O. Zheng, and A. Abdelraouf, "Advances and applications of computer vision techniques in vehicle trajectory generation and surrogate traffic safety indicators," *Accident Analysis and Prevention*, vol. 191, 2023, doi: 10.1016/j.aap.2023.107191.

[12] H. Gao *et al.*, "A hybrid deep learning model for urban expressway lane-level mixed traffic flow prediction," *Engineering Applications of Artificial Intelligence*, vol. 133, 2024, doi: 10.1016/j.engappai.2024.108242.

[13] R. Zhao, S. H. Tang, J. Shen, E. E. Bin Supeni, and S. A. Rahim, "Enhancing autonomous driving safety: a robust traffic sign detection and recognition model TSD-YOLO," *Signal Processing*, vol. 225, 2024, doi: 10.1016/j.sigpro.2024.109619.

[14] S. Xu, M. Zhang, J. Chen, and Y. Zhong, "YOLO-HyperVision: a vision transformer backbone-based enhancement of YOLOv5 for detection of dynamic traffic information," *Egyptian Informatics Journal*, vol. 27, 2024, doi: 10.1016/j.eij.2024.100523.

[15] H. A. Saputri, M. Avrillio, L. Christofer, V. Simanjaya, and I. N. Alam, "Implementation of YOLO v7 algorithm in estimating traffic flow in Malang," *Procedia Computer Science*, vol. 245, pp. 117–126, 2024, doi: 10.1016/j.procs.2024.10.235.

[16] J. Tang, C. Ye, X. Zhou, and L. Xu, "YOLO-fusion and internet of things: advancing object detection in smart transportation," *Alexandria Engineering Journal*, vol. 107, pp. 1–12, 2024, doi: 10.1016/j.aej.2024.09.012.

[17] R. Ronariv, R. Antonio, S. F. Jorgensen, S. Achmad, and R. Sutoyo, "Object detection algorithms for car tracking with euclidean distance tracking and YOLO," *Procedia Computer Science*, vol. 245, pp. 627–636, 2024, doi: 10.1016/j.procs.2024.10.289.

[18] X. Zhai, Z. Huang, T. Li, H. Liu, and S. Wang, "YOLO-drone: an optimized YOLOv8 network for tiny UAV object detection," *Electronics*, vol. 12, no. 17, 2023, doi: 10.3390/electronics12173664.

[19] I. Shamta, F. Demir, and B. E. Demir, "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*, vol. 15, no. 12, 2024, doi: 10.1016/j.asej.2024.103148.

[20] K. Wang, C. Ma, Y. Qiao, X. Lu, W. Hao, and S. Dong, "A hybrid deep learning model with 1DCNN-LSTM-attention networks for short-term traffic flow prediction," *Physica A: Statistical Mechanics and its Applications*, vol. 583, 2021, doi: 10.1016/j.physa.2021.126293.

[21] J. Lu, "An efficient and intelligent traffic flow prediction method based on LSTM and variational modal decomposition," *Measurement: Sensors*, vol. 28, 2023, doi: 10.1016/j.measen.2023.100843.

[22] B. Naheliya, P. Redhu, and K. Kumar, "MFOA-Bi-LSTM: an optimized bidirectional long short-term memory model for short-term traffic flow prediction," *Physica A: Statistical Mechanics and its Applications*, vol. 634, 2024, doi: 10.1016/j.physa.2023.129448.

[23] J. D. Wang and C. O. N. Susanto, "Traffic flow prediction with heterogenous data using a hybrid CNN-LSTM model," *Computers, Materials and Continua*, vol. 76, no. 3, pp. 3097–3112, 2023, doi: 10.32604/cmc.2023.040914.

[24] A. R. Sattarzadeh, R. J. Kutadinata, P. N. Pathirana, and V. T. Huynh, "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*, vol. 21, no. 1, pp. 388–410, 2025, doi: 10.1080/23249935.2023.2236724.

[25] Y. Luo, J. Zheng, X. Wang, Y. Tao, and X. Jiang, "GT-LSTM: a spatio-temporal ensemble network for traffic flow prediction," *Neural Networks*, vol. 171, pp. 251–262, 2024, doi: 10.1016/j.neunet.2023.12.016.

[26] K. Zhao, D. Guo, M. Sun, C. Zhao, and H. Shuai, "Short-term traffic flow prediction based on VMD and IDBO-LSTM," *IEEE Access*, vol. 11, pp. 97072–97088, 2023, doi: 10.1109/ACCESS.2023.3312711.

[27] Bharti, P. Redhu, and K. Kumar, "Short-term traffic flow prediction based on optimized deep learning neural network: PSO-Bi-LSTM," *Physica A: Statistical Mechanics and its Applications*, vol. 625, 2023, doi: 10.1016/j.physa.2023.129001.

[28] C. Chaoura, H. Lazar, and Z. Jarir, "Traffic flow prediction at intersections: enhancing with a Hybrid LSTM-PSO approach," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 5, pp. 494–501, 2024, doi: 10.14569/IJACSA.2024.0150549.

[29] S. Suyanto, A. A. Ariyanto, and A. F. Ariyanto, "Komodo Mlipir algorithm," *Applied Soft Computing*, vol. 114, 2022, doi: 10.1016/j.asoc.2021.108043.

[30] G. K. Jati, G. Kuwanto, T. Hashmi, and H. Widjaja, "Discrete komodo algorithm for traveling salesman problem," *Applied Soft Computing*, vol. 139, May 2023, doi: 10.1016/j.asoc.2023.110219.

[31] D. Raimondo *et al.*, "Detection and classification of hysteroscopic images using deep learning," *Cancers*, vol. 16, no. 7, Mar. 2024, doi: 10.3390/cancers16071315.

[32] M. Pei, N. Liu, B. Zhao, and H. Sun, "Self-supervised learning for industrial image anomaly detection by simulating anomalous samples," *International Journal of Computational Intelligence Systems*, vol. 16, no. 1, 2023, doi: 10.1007/s44196-023-00328-0.

[33] N. E. Khalifa, M. Loey, and S. Mirjalili, “A comprehensive survey of recent trends in deep learning for digital images augmentation,” *Artificial Intelligence Review*, vol. 55, no. 3, pp. 2351–2377, 2022, doi: 10.1007/s10462-021-10066-4.

[34] B. A. Awaluddin, C. T. Chao, and J. S. Chiou, “Investigating effective geometric transformation for image augmentation to improve static hand gestures with a pre-trained convolutional neural network,” *Mathematics*, vol. 11, no. 23, 2023, doi: 10.3390/math11234783.

[35] M. Firdaus, K. Kusrini, and M. R. Arief, “Impact of data augmentation techniques on the implementation of a combination model of convolutional neural network (CNN) and multilayer perceptron (MLP) for the detection of diseases in rice plants,” *Journal of Scientific Research, Education, and Technology (JSRET)*, vol. 2, no. 2, pp. 453–465, 2023, doi: 10.58526/jsret.v2i2.94.

[36] M. M. Rafi *et al.*, “Performance analysis of deep learning YOLO models for South Asian Regional vehicle recognition,” *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 9, pp. 864–873, 2022, doi: 10.14569/IJACSA.2022.01309100.

[37] A. Betti and M. Tucci, “YOLO-S: A lightweight and accurate YOLO-like network for small target selection in aerial imagery,” *Sensors*, vol. 23, no. 4, pp. 1–22, 2023, doi: 10.3390/s23041865.

[38] S. Hamiane, Y. Ghanou, H. Khalifi, and M. Telmem, “Comparative analysis of LSTM, ARIMA, and hybrid models for forecasting future GDP,” *Ingenierie des Systemes d'Information*, vol. 29, no. 3, pp. 853–861, 2024, doi: 10.18280/isi.290306.

[39] C. L. Yang, A. A. Yilma, H. Sutrisno, B. H. Woldegiorgis, and T. P. Q. Nguyen, “LSTM-based framework with metaheuristic optimizer for manufacturing process monitoring,” *Alexandria Engineering Journal*, vol. 83, pp. 43–52, 2023, doi: 10.1016/j.aej.2023.10.006.

[40] N. Halpern-Wight, M. Konstantinou, A. G. Charalambides, and A. Reinders, “Training and testing of a single-layer LSTM network for near-future solar forecasting,” *Applied Sciences*, vol. 10, no. 17, pp. 1–9, 2020, doi: 10.3390/app10175873.

[41] K. L. Tan, C. P. Lee, K. S. M. Anbananthen, and K. M. Lim, “RoBERTa-LSTM: A hybrid model for sentiment analysis with transformer and recurrent neural network,” *IEEE Access*, vol. 10, pp. 21517–21525, 2022, doi: 10.1109/ACCESS.2022.3152828.

[42] A. Abirami and R. Kavitha, “A novel automated Komodo mlipir optimization-based attention BiLSTM for early detection of diabetic retinopathy,” *Signal, Image and Video Processing*, vol. 17, no. 5, pp. 1945–1953, 2023, doi: 10.1007/s11760-022-02407-9.

[43] B. L. Lawrence and E. de Lemmus, “Using computer vision to classify, locate and segment fire behavior in UAS-captured images,” *Science of Remote Sensing*, vol. 10, 2024, doi: 10.1016/j.srs.2024.100167.

[44] R. Kablaoui, I. Ahmad, S. Abed, and M. Awad, “Network traffic prediction by learning time series as images,” *Engineering Science and Technology, an International Journal*, vol. 55, Jul. 2024, doi: 10.1016/j.jestch.2024.101754.

[45] N. S. Chauhan and N. Kumar, “Confined attention mechanism enabled recurrent neural network framework to improve traffic flow prediction,” *Engineering Applications of Artificial Intelligence*, vol. 136, 2024, doi: 10.1016/j.engappai.2024.108791.

BIOGRAPHIES OF AUTHORS

Imam Ahmad Ashari received his bachelor's degree in Informatics from Universitas Negeri Semarang in 2016 and his master's degree in Information Systems from Universitas Diponegoro in 2019. He is currently a lecturer at the Informatics Study Program, Universitas Harapan Bangsa. His research interests include the internet of things (IoT), computer vision, and artificial intelligence. He can be contacted at email: imamahmadashari@uhb.ac.id.

Wahyul Amien Syafei received a bachelor's degree from Diponegoro University, Indonesia in 1995, a master's degree from Sepuluh Nopember Institute of Technology, Indonesia in 2002, and a doctoral degree from Kyushu Institute of Technology, Japan in 2010. Currently, he is an associate professor at the Master Program of Electrical Engineering, Faculty of Engineering, Diponegoro University. He is also a lecturer at the Master Program of Electrical Engineering, Postgraduate School, Diponegoro University. His research interests include transmission channels, digital communication systems, multimedia, multimedia networks, computer graphics, data communication, wireless and mobile networks, multimedia telecommunication, information systems, and signal transformation processing. He can be contacted at email: wasyafei@elektro.undip.ac.id.

Adi Wibowo received the B.Sc. degree in Mathematics from Universitas Diponegoro, Indonesia, in 2005, the M.Sc. degree in Computer Science from Universitas Indonesia in 2011, and the Ph.D. degree in Engineering from Nagoya University, Japan, in 2016. He has been an Assistant Professor with the Department of Informatics, Universitas Diponegoro, Indonesia, since 2006. His research interests primarily focus on artificial intelligence, deep learning, computer vision, bioinformatics, and data mining. He has authored or coauthored several publications in these areas. He can be contacted at email: adiwibowo@lecturer.undip.ac.id.