Leveraging Digital Twin Technology for Traffic Optimization: A Pathway to Sustainable Urban Transportation

Zaman Basheer[1], Sharan Karthick[2], Hari Chandana Pichhika[2], Raja Vara Prasad Yerra[2],
Hrishikesh Venkataraman[2], Purav Shah[1], Giacomo Nalli[3], Ramona Trestian[1] and Ioan-Sorin Comşa[4],
[1]London Digital Twin Research Centre, Middlesex University, NW4 4BT, London, U.K.
[2]Department of Electronics and Communication Engineering,
Indian Institute of Information Technology, Sri City, India.
[3]Computer Science Department, Middlesex University, NW4 4BT, London, U.K.
[4]Institute for Research in Open-, Distance- and eLearning,
Swiss Distance University of Applied Sciences, 3900 Brig, Switzerland,
E-mails: {z.kunnothChundangapoyil, p.shah, g.nalli, r.trestian}@mdx.ac.uk,
{sharankarthick.bl22, harichandana.p, yrv.prasad, hvraman}@@iiits.in, ioan-sorin.comsa@ffhs.ch

***Abstract*—Urbanization and rapid population growth have intensified vehicular traffic, leading to congestion, increased travel times, and environmental challenges, particularly carbon emissions. Addressing these issues requires innovative approaches that can optimize traffic flow while minimizing environmental impacts. Digital twin technology, a virtual replication of physical systems utilizing real-time data, has emerged as a promising solution in urban transportation management. This paper presents DT-STOP (Digital Twin for Sustainable Traffic Optimization and Planning), a novel framework leveraging computer vision and edge computing to optimize traffic management in unstructured traffic environments. DT-STOP is designed specifically for the complex and often chaotic traffic scenarios seen in countries like India, where lane discipline is minimal and vehicle diversity is high. Through the integration of digital twins and advanced AI-driven computer vision, DT-STOP enables dynamic traffic management, and improves urban mobility. The effectiveness of our approach is demonstrated through a case study in an urban Indian setting, showcasing the efficiency of integrating computer vision in unstructured traffic environments.**

***Index Terms*—Digital Twin, Intelligent Transportation Systems, Traffic Management**

I. Introduction

The unprecedented rise in urbanization and population growth has resulted in severe congestion in urban road networks, increasing travel times, and exacerbating environmental challenges such as air pollution and climate change. Transportation emissions, a major contributor to poor air quality and global warming, have become a growing concern for cities worldwide [1]. Consequently, there is an urgent need for innovative technologies that can enhance traffic management and promote sustainability. Digital twin technology, initially conceptualized by Grieves in 2003 [2] and further refined in 2010 [3], has emerged as a transformative tool for urban transportation. By creating virtual replicas of physical systems, digital twins enable real-time simulation and optimization, providing insights for efficient traffic management and emission reduction [4].

Several existing works have demonstrated the potential of digital twin systems for traffic management. Kusic *et al.* [5] developed a Digital Twin for the Geneva motorway (DT-GM) using the SUMO microscopic traffic simulator and real-time traffic data, enabling dynamic flow calibration and rerouting. Similarly, Tettamanti *et al.* [6] employed a Vehicle-In-The-Loop (VIL) approach for autonomous vehicle testing, integrating real-world and virtual traffic scenarios. Szalai *et al.* [7] expanded on this with a mixed-reality framework combining SUMO and Unity 3D for autonomous vehicle simulations, providing valuable insights into vehicle behavior and system performance. However, these studies predominantly focus on structured traffic systems.

The Indian Driving Dataset (IDD) [8], [9] tailored for the unique challenges of Indian roads, has been instrumental in developing and validating these models. Varma *et al.* [10] introduced IDD as a comprehensive resource for road scene understanding, while Paranjape & Naik *et al.* [11] extended its application with the DATS-2022, a dataset designed for object detection in unstructured traffic conditions in India. Baheti *et al.* [12] proposed Eff-UNet, a deep-learning model trained on IDD for enhanced semantic segmentation, and Kolekar *et al.* [13] applied Explainable AI (XAI) techniques to improve scene interpretation.

Vehicular communication plays a crucial role in ensuring reliable and efficient wireless connectivity in Intelligent Transportation Systems (ITS). The development and validation of small-scale Dedicated Short-Range Communication (DSRC) systems have been explored in previous studies. For example, Kamal *et al.* [14] and Rayamajhi *et al.* [15] demonstrated the feasibility of Dedicated Short-Range Communication (DSRC) and Cellular Vehicle-to-Everything (C-V2X) technologies for real-time data exchange. Ali *et al.* [16] explored 5G and massive MIMO systems to optimize vehicular networks, highlighting the role of RSUs in managing vehicle interactions. Additionally, Norazmi *et al.* [17] emphasized the importance

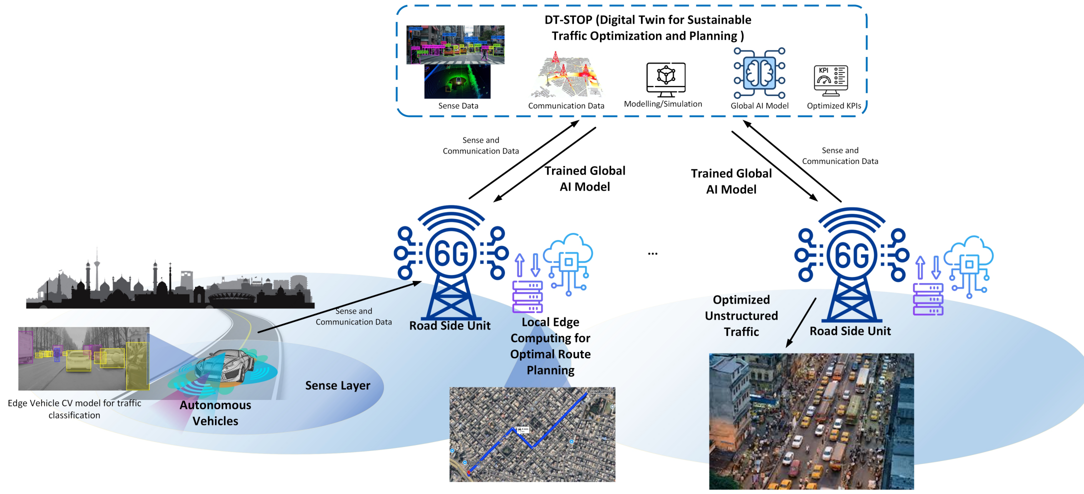

Fig. 1: Proposed DT-STOP Framework

of RSU placement for accurate vehicle localization.

In this context, we propose DT-STOP (**D**igital **T**win for **S**ustainable **T**raffic **O**ptimization and **P**lanning) to address the complexities of unstructured traffic using domain-specific computer vision models. Unlike traditional traffic management systems, which often rely on static data and centralized decision-making, DT-STOP leverages real-time data from autonomous vehicles, roadside units (RSUs), and sensors to dynamically optimize traffic flows. This approach is particularly well-suited for unstructured traffic environments like those in India, characterized by unpredictable vehicle behavior, diverse vehicle types, and limited lane discipline. Moreover, by integrating digital twins, computer vision, and edge computing, DT-STOP offers a comprehensive solution for sustainable traffic management in unstructured environments.

II. DT-STOP Framework

The proposed DT-STOP framework is illustrated in Figure 1 and consists of three main layers: the Sense Layer, the Communication Layer, and the Digital Twin Layer. Each layer works collaboratively to ensure real-time traffic analysis, optimization, and decision-making, specifically tailored to manage unstructured traffic conditions.

A. Sense layer

It is essential to monitor and measure different parameters of the vehicles in real time for the possible recommendations of DT-STOP framework. Vehicles of different levels of autonomy are equipped with different types and ranges of sensors [18], like RGB camera, LiDAR, RADAR, IMU (Inertial Measurement Unit)s, GPS etc. Number of above sensors required for a particular vehicle depends on different parameters like the level of autonomy, structured/unstructured traffic, type of traffic environment (urban, semi-Urban, Highway, congested, space etc.), geography, environmental conditions etc. The Sense Layer of DT-STOP is planned to accommodate most data collection from a heterogeneous nature of sensors. The sensor layer of DT-STOP assumes the preliminary data preprocessing and auto calibration of acquired data at the edge node before the data is sent for any further processing to the local AI/ML model or the communication layer to connect to the cloud DT-STOP platform.

B. Communication Layer

The Communication Layer is powered by 6G-enabled Road-Side Units (RSUs) that facilitate ultra-low-latency data transmission. RSUs act as intermediate nodes for communication between autonomous vehicles, edge devices, and the cloud. This enables real-time information sharing, allowing rapid updates on traffic conditions within local areas. Additionally, local edge AI-based models deployed at RSUs provide preliminary traffic analysis and send optimized route planning data to the vehicles and processed data to the digital twin layer for further analysis.

C. Digital Twin layer

At the core of the DT-STOP framework is the digital twin, a virtual representation of the urban traffic environment. It continuously receives data from the Communication Layer, simulating traffic conditions using AI-driven predictive models. The digital twin is equipped with a trained global AI model that aggregates insights from various cities to improve its predictions. By running multiple simulations, the digital twin generates optimized traffic management strategies, including real-time route adjustments and traffic light timing optimization. These strategies are relayed back to the Sense Layer for immediate implementation.

Unlike traditional traffic management systems, DT-STOP operates on a feedback loop that enables continuous learning and refinement. The local edge models periodically update based on insights from the global AI model, ensuring accurate and timely decision-making. Through this holistic and scalable approach, DT-STOP not only aims at reducing congestion and travel times but also aims at minimizing emissions, contributing to sustainable urban development.

III. Edge Vehicle Computer Vision

A key innovation of DT-STOP is its adaptive computer vision algorithms, specifically trained on unstructured traffic datasets such as the IDD. These models efficiently detect, classify, and track vehicles under challenging conditions. This section presents a detailed exploration of YOLOv8's architecture [19], focusing on its fundamental components, including its backbone network, detection head, and YOLOv8-Seg for semantic segmentation [20]. It thoroughly discusses various aspects of model training, such as dataset selection, preprocessing steps, and training methodologies aimed at enhancing model efficiency. The approach begins with real-time image capture from a camera sensor, which is subsequently processed by YOLOv8 to identify and classify objects within the captured frames. The model extracts critical visual features, such as shape, texture, and color, to distinguish different objects, ensuring high-precision obstacle detection.

1) Input data acquisition: A camera sensor was utilized to acquire real-time image or video data, mounted on an ego vehicle platform (e.g., an autonomous vehicle) as part of an IDD dataset.

2) Frame Extraction from Video: Frames from the IDD dataset are extracted for a specific road stretch in Hyderabad. This process utilizes OpenCV or similar video processing libraries to convert the video stream into individual frames, enabling object detection.

3) Object Detection and Classification: YOLOv8 Nano is optimized for edge computing, striking a balance between efficiency and accuracy. Its lightweight design reduces computational demands, making it ideal for low-power devices while maintaining real-time detection capabilities. Unlike larger models, it delivers fast inference without compromising detection quality, ensuring reliable object recognition in dynamic environments. This makes it well-suited for applications like autonomous driving and smart surveillance, where real-time processing on resource-limited hardware is crucial. Unlike traditional object detection pipelines, YOLOv8 processes full images in a single evaluation, bypassing the need for region proposal networks, which speeds up real-time inference. Built upon prior YOLO versions, YOLOv8 improves feature extraction by employing a Darknet-based backbone. This ensures robust detection in complex environments, such as Indian roads, where diverse vehicle types, high pedestrian density, and fluctuating lighting conditions create detection challenges. The model incorporates PANet (Path Aggregation Network) to enhance multi-scale feature extraction, allowing better recognition of objects regardless of size variations and occlusions. Feature fusion improves the detection of small or partially obscured objects, which is critical in dense urban traffic scenarios. YOLOv8 optimizes real-time performance through: Feature Pyramid Networks (FPNs), Anchor-free detection, Improved bounding box regression, High-speed inference techniques These enhancements significantly reduce computational overhead, enabling fast object detection in real-world driving conditions.

YOLOv8 offers multiple model variants, ranging from high-capacity large models for detailed feature extraction to lightweight Nano models, which are optimized for low-memory, low-power devices. Performance metrics commonly used to evaluate these models include Parameter count (Params): indicates model complexity, inference time and detection accuracy. Large and medium-sized YOLOv8 models excel in controlled environments but may struggle with efficiency and real-time execution in real-world scenarios. Nano and small YOLOv8 models demonstrate higher robustness in real-life testing, particularly in challenging, unstructured Indian road conditions, where rapid detection is critical. Once objects are identified, bounding boxes are placed over detected elements in the scene. Confidence scores accompany the bounding boxes, indicating the reliability of the detection results. This ensures that the autonomous system can process, interpret, and respond to real-time road environments within critical time constraints, enabling edge-based inference without relying on cloud models in certain scenarios.

4) Semantic Segmentation: For this task, we utilized the Masked YOLO framework for semantic segmentation, integrating alternating convolutional layers and a coarse-to-fine strategy to enhance feature extraction and refinement. To optimize performance, we applied transfer learning, initializing the model with pre-trained COCO weights and fine-tuning it on the IDD Segmentation dataset. The model was trained for 10 epochs with a learning rate of 0.001, ensuring efficient adaptation to road segmentation tasks while maintaining high pixel-wise accuracy.

5) GIS Mapping: Geographic Information System (GIS) - based mapping techniques are employed to integrate object detection results with geolocation data, enabling spatial visualization of traffic patterns. Folium, a Python-based interactive mapping library, is used to overlay detected objects onto a base map, providing an intuitive representation of traffic congestion along a predefined route. Polyline representations are utilized to illustrate vehicle movement, offering a structured way to depict changes in traffic density across different locations.

To classify traffic congestion levels, the total count of important road objects within a given spatial region was analyzed. Based on these counts, traffic density was categorized into distinct levels, ensuring a clear distinction between low, medium, and high congestion zones. This grouping method provided a structured approach to assess traffic flow variations, allowing for more effective congestion analysis and management.

IV. Experimental Settings and Evaluations

A. Computer Vision

In this study, computer vision (CV) plays a crucial role in capturing and analyzing real-time traffic data, including vehicle detection, and congestion monitoring. By leveraging CV techniques, the proposed DT-STOP framework can dynamically optimize traffic flow, reduce emissions, and enhance urban mobility.

1) Dataset: The Indian Driving Dataset (IDD) [10] comprehensively captures the complexities of unstructured traffic environments prevalent in India, encompassing diverse road conditions, heterogeneous traffic compositions, and pedestrian interactions across both urban and rural regions of Bangalore and Hyderabad. IDD is structured into multiple subsets, each curated to facilitate specific computer vision tasks. The IDD-Detection subset comprises 46,588 images, annotated with

15 traffic-related classes, designed to support object detection applications. The IDD-Segmentation subset consists of 10,003 images extracted from 182 driving sequences, featuring polygon-based annotations to enable precise semantic segmentation. Additionally, the IDD-Multimodal subset integrates GIS location data with sequential frames, facilitating advanced spatial analysis.

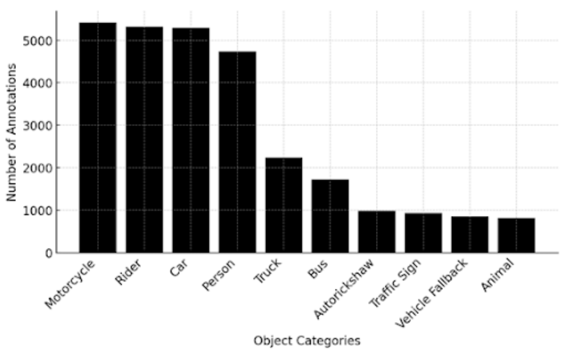

Fig. 2: Class Distribution for Detection

A total of 5,052 images from the IDD-Detection dataset were used for object detection. Annotations covered 10 traffic-related classes: animals, autorickshaws, buses, cars, motorcycles, persons, riders, traffic signs, trucks, and vehicle fallbacks, with class distribution shown in Fig. 2. The dataset was processed in Roboflow and split into 70% training, 20% validation, and 10% testing.

For IDD-Segmentation, only the road class was extracted from the polygon-based annotations. Binary masks were generated for each image and converted to YOLO .txt format. The dataset was partitioned into 80% training and 20% validation.

2) System Configuration: All training and implementation were conducted on a dedicated server built by Middlesex University, running Debian 6.1. The system is equipped with an Intel Xeon Platinum 8268 processor and an NVIDIA Quadro RTX 8000 GPU with 46GB VRAM. CUDA 12.8, cuDNN, and PyTorch were installed to enable high-performance deep learning training and inference.

3) Model Training and Implementation: Both object detection and semantic segmentation models were trained using YOLOv8. The Ultralytics YOLOv8 package was utilized for training, employing the SGD optimizer with an initial learning rate of 0.01. The models were trained on the dataset and evaluated on the validation set.

For object detection, an extensive simulations was carried out to compare the performances of different versions of YOLOv8 were trained, including nano, small, medium, and large. Each variant differed in complexity, balancing detection accuracy and inference speed. For segmentation, YOLOv8s-seg was used for training, and the model was trained for 10 epochs. The training process for object detection was initially conducted for 50 epochs and later extended to 100 epochs. A batch size of 16 and an input image size of 640 × 640 were used to optimize computational performance while ensuring high detection and segmentation accuracy.

Once trained, the models were applied to the IDD multi-modal dataset for inference. Object detection produced multi-class predictions, identifying various objects within each frame, while semantic segmentation generated a binary road mask, isolating road pixels. For traffic route optimization, only on-road objects were considered important, as they directly influenced vehicle movement and congestion, while off-road objects were excluded from the analysis.

To filter objects based on road presence, a bounding-box-based overlap filtering approach was used. The proportion of road pixels within each detected object's bounding box was computed as:

$$IoU = \left(\frac{A_{overlap}}{A_{obj}}\right) \times 100 \tag{1}$$

where $A_{overlap}$ represents the number of road pixels inside the bounding box and A_{obj} is the total area of the bounding box. If overlap percentage exceeded a predefined threshold τ, the object was classified as important. This method ensured that only objects physically present on the road were retained for further analysis.

The final output from the object filtering layer, consisting of frame-wise detections and the count of important objects, was mapped using Folium to visualize traffic conditions. The mapping process involved overlaying object detection results onto a base map centred on the mean latitude and longitude of the dataset. To represent the vehicle's trajectory, GPS coordinates were extracted and connected using polylines. Each polyline segment was color-coded to indicate traffic density, determined by summing the number of detected objects at each GPS coordinate. The classification thresholds for traffic density were predefined, with low-traffic areas displayed in green, medium-density areas in orange, and high-traffic zones in red. This visualization provided a spatial overview of congestion levels and vehicle movement, aiding in route optimization and traffic analysis.

4) Evaluation Metrics: To assess the detection performance of our enhanced model, we utilize several evaluation metrics: precision, recall, Mean Average Precison (mAP), and the inference time. The specific formulas for these metrics are provided in this section. mAP represents the average AP value across all categories, indicating the model's overall detection performance across the entire dataset. Precision is the metric that represents the ratio of true positives to the total predicted positives. Recall is a measure of the ratio of correctly predicted positive samples to all actual positive samples.

B. Results and Observations

1) Computer Vision: Fig. 3 illustrates the process of object detection, road segmentation, and filtering road-only objects in a traffic scene. Fig. 3 (a) showcases YOLO-based object detection, where various objects such as vehicles, motorcycles, and pedestrians are identified using bounding boxes. Fig. 3 (b) applies road segmentation, highlighting the road area while darkening the rest of the scene to distinguish objects positioned on the road. Finally, Fig. 3 (c) combines both processes, retaining only objects within the road boundaries while filtering out background elements like pedestrians on

(a) (b) (c)

Fig. 3: (a) Object Detection, (b) Road Segmentation and (c) Filtering road-only objects

sidewalks. This approach enhances traffic analysis by ensuring only road-relevant objects are considered, improving autonomous driving, smart traffic monitoring, and accident prevention systems.

The evaluation of YOLOv8 variants for object detection, as shown in Table I, highlights the trade-offs between accuracy and inference time. Larger models provided higher accuracy but required more processing time, whereas smaller models offered faster inference with a slight reduction in precision. YOLOv8s emerged as the most balanced option, optimizing both speed and accuracy for real-time detection.

TABLE I: Experiment Results

Object Detection				
Model	**mAP50**	**Precision**	**Recall**	**Inference Time(ms)**
YOLOv8n	0.593	0.774	0.526	1.0
YOLOv8s	0.636	0.775	0.581	1.9
YOLOv8m	0.649	0.809	0.574	4.3
YOLOv8l	0.648	0.791	0.580	6.5
Semantic Segmentation				
YOLOv8s-seg	0.975	0.983	0.965	2.5

For semantic segmentation, YOLOv8s-seg demonstrated high precision and recall while maintaining efficient processing. It achieved strong segmentation performance while ensuring computational feasibility. The identified important objects from live detection were utilized for traffic route optimization, allowing real-time analysis of congestion patterns and vehicle movement. This approach enables more effective navigation strategies by considering dynamic traffic conditions and road occupancy. The results suggest that a model like YOLOv8s is well-suited for real-time object detection in dynamic traffic environments and is particularly advantageous for deployment on edge devices due to its balance of computational efficiency and accuracy.

V. DT-STOP for RSU Placement Optimization

This section presents a use case scenario to demonstrate how the DT-STOP framework can optimize the placement of RSUs for efficient vehicular communication using digital twin technology. By leveraging communication network data, DT-STOP ensures seamless connectivity with minimal infrastructure deployment, especially in urban environments with unstructured traffic.

A. Proposed Methodology

1) Virtual Environment Generation: DT-STOP creates a Unity-based Digital Twin that models real-world urban environments using GIS data. The virtual environment enables detailed simulation of traffic movement and wireless connectivity scenarios.

2) Path Planning: Vehicle trajectory data is collected using sensors and edge devices. Real-time data from the communication layer is integrated into DT-STOP to simulate various traffic scenarios and determine optimal RSU placements. DT-STOP's adaptive models ensure accurate path representation, accommodating unstructured traffic characteristics.

3) RSU Placement: The DT-STOP framework employs a combination of computer vision-based vehicle classification and network connectivity analysis. Utilizing the Friis transmission equation, it evaluates wireless coverage under varying transmission powers, frequencies, and antenna gains. The iterative algorithm simulates multiple RSU configurations to achieve maximum Line-of-Sight (LOS) coverage with minimal overlap, ensuring reliable communication and reducing infrastructure costs.

4) Visualization and Output: The DT-STOP generates visual heatmaps to display coverage quality. LOS regions are highlighted in cyan, handoff zones in yellow, and Non-Line-of-Sight (NLOS) regions in red. Through continuous analysis, DT-STOP recommends optimal RSU placements and suggests real-time adjustments based on emerging traffic patterns.

B. Experimental Settings

A simulation environment was created using DT-STOP to evaluate RSU placement along a selected urban road segment. Real-time vehicle path data sourced from the IDD Multimodal dataset was utilized to reflect actual driving behaviors in India. The simulation area covered 100 hectares with Class-C RSUs, operating at 5920 MHz with a nominal range of 400 meters. Each simulation run tested different RSU placements, adjusting for dynamic NLOS scenarios and maintaining a 10% overlap between adjacent RSUs to mitigate link failures. The best placement strategy was identified by minimizing the number of RSUs while maximizing coverage.

C. Results and Analysis

The DT-STOP framework effectively optimized RSU placement using communication data and computer vision insights. By analyzing LOS, NLOS, and handoff scenarios, it identified the most efficient network configuration. To visualize coverage, heat maps are used to determine LOS, NLOS and optimal signal thresholds to better understand the effect of buildings on the wireless link as demonstrated in Fig. 4. An example path of vehicle in the considered IDD dataset is shown with placement of the RSU and vehicle trajectory as shown in Fig. 5. In a representative simulation, the system achieved full coverage using four RSUs after evaluating ten configurations. Visualizations generated by DT-STOP illustrated improved network reliability and reduced deployment costs.

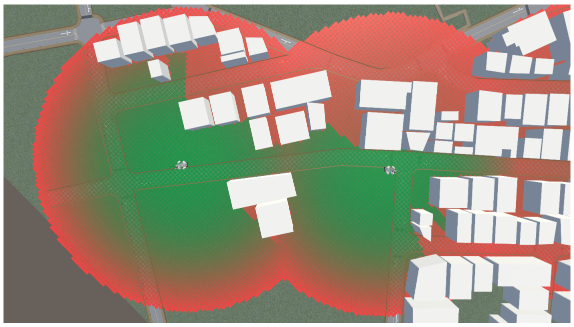

Fig. 4: Coverage Visualization in the form of Heatmap

Fig. 5: An Instance of RSU Placement

VI. Conclusions

This paper introduces DT-STOP (Digital Twin for Sustainable Traffic Optimization and Planning) as an advanced solution for addressing the complexities of unstructured traffic. A key aspect of DT-STOP is its use of computer vision, which plays a crucial role in vehicle detection and congestion monitoring. By leveraging datasets like the Indian Driving Dataset (IDD), DT-STOP can process complex traffic scenes, ensuring that only relevant road objects are analyzed, enhancing traffic optimization and vehicle movement management. Moreover, integrating digital twins, computer vision, and edge computing, DT-STOP offers a sustainable and efficient approach to traffic management, improving urban mobility and reducing emissions in unstructured environments.

Acknowledgment

This research was conducted as part of the UKIERI-SPARC project "DigIT—Digital Twins for Integrated Transportation Platform", grant number UKIERI-SPARC/01/23..

References

[1] A. Sohail, B. Shen, M. A. Cheema, M. E. Ali, A. Ulhaq, M. A. Babar, and A. Qureshi, "Beyond data, towards sustainability: A sydney case study on urban digital twins," *arXiv preprint arXiv:2406.04902*, 2024.

[2] M. Grieves, "Digital twin: manufacturing excellence through virtual factory replication," *White paper*, vol. 1, no. 2014, pp. 1–7, 2014.

[3] M. Grieves and J. Vickers, "Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems," *Transdisciplinary perspectives on complex systems: New findings and approaches*, 2017.

[4] A. Abouelrous, L. Bliek, and Y. Zhang, "Digital twin applications in urban logistics: An overview," *Urban, Planning and Transport Research*, vol. 11, no. 1, p. 2216768, 2023.

[5] K. Kušić, R. Schumann, and E. Ivanjko, "Building a motorway digital twin in sumo: Real-time simulation of continuous data stream from traffic counters," in *2022 international symposium ELMAR*. IEEE, 2022.

[6] T. Tettamanti, M. Szalai, S. Vass, and V. Tihanyi, "Vehicle-in-the-loop test environment for autonomous driving with microscopic traffic simulation," in *2018 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*. IEEE, 2018, pp. 1–6.

[7] M. Szalai, B. Varga, T. Tettamanti, and V. Tihanyi, "Mixed reality test environment for autonomous cars using unity 3d and sumo," in *2020 IEEE 18th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*. IEEE, 2020, pp. 73–78.

[8] A. Majee, K. Agrawal, and A. Subramanian, "Few-shot learning for road object detection," in *AAAI workshop on meta-learning and metadl challenge*. PMLR, 2021, pp. 115–126.

[9] C. Parikh, R. Saluja, C. Jawahar, and R. K. Sarvadevabhatla, "Idd-x: A multi-view dataset for ego-relative important object localization and explanation in dense and unstructured traffic," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 14 815–14 821.

[10] G. Varma, A. Subramanian, A. Namboodiri, M. Chandraker, and C. Jawahar, "Idd: A dataset for exploring problems of autonomous navigation in unconstrained environments," in *2019 IEEE winter conference on applications of computer vision*. IEEE, 2019.

[11] B. A. Paranjape and A. A. Naik, "Dats_2022: a versatile indian dataset for object detection in unstructured traffic conditions," *Data in brief*, vol. 43, p. 108470, 2022.

[12] B. Baheti, S. Innani, S. Gajre, and S. Talbar, "Eff-unet: A novel architecture for semantic segmentation in unstructured environment," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, pp. 358–359.

[13] S. Kolekar, S. Gite, B. Pradhan, and A. Alamri, "Explainable ai in scene understanding for autonomous vehicles in unstructured traffic environments on indian roads using the inception u-net model with grad-cam visualization," *sensors*, vol. 22, no. 24, p. 9677, 2022.

[14] F. Kamal, E. Lou, and V. Zhao, "Design and validation of a small-scale 5.9 ghz dsrc system for vehicular communication," 04 2012, pp. 1–4.

[15] A. Rayamajhi, A. Yoseph, A. Balse, Z. Huang, E. Leslie, and V. Fessmann, "Preliminary performance baseline testing for dedicated short-range communication (dsrc) and cellular vehicle-to-everything (c-v2x)," *IEEE 92^{nd} Vehicular Technology Conference (VTC2020-Fall)*, 2020.

[16] R. Ali, D. N. Hakro, M. R. Tanweer, and A. A. Kamboh, "Simulation based vehicle to vehicle and base station communication," in *2019 International Conference on Information Science and Communication Technology (ICISCT)*, 2019, pp. 1–6.

[17] M. F. Norazmi, Y. H. Lee, and S. Manandhar, "Investigating the impact of spatial location of rsu in trilateration network," in *2020 IEEE International Symposium on Antennas and Propagation and North American Radio Science Meeting*, 2020, pp. 136–1370.

[18] S. Mihai, P. Shah, G. Mapp, H. Nguyen, and R. Trestian, "Towards autonomous driving: A machine learning-based pedestrian detection system using 16-layer lidar," in *2020 13th International Conference on Communications (COMM)*, 2020, pp. 271–276.

[19] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics yolov8," 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[20] E. Mohamed, A. Shaker, A. El-Sallab, and M. Hadhoud, "Insta-yolo: Real-time instance segmentation," *preprint arXiv:2102.06777*, 2021.