SUMO User Conference 2025

Conference paper

https://doi.org/10.52825/scp.v6i.2619

Published: 15 Jul. 2025

Digital Twin-Aided Municipal Traffic Control

Reeti Pradhananga[1], Shelby Williams[1], Sercan Aygun[1,*], Li Chen[1], Yazhou Tu[2], Whitney Crow[3], Sathyanarayanan Aakur[2], and Nian-Feng Tzeng[1]

[1]University of Louisiana at Lafayette, Lafayette, LA, USA

[2]Auburn University, Auburn, AL, USA

[3]Mississippi State University, Mississippi State, MS, USA

*Correspondence: Sercan Aygun, sercan.aygun@louisiana.edu

Abstract. Swift advances in computing and artificial intelligence (AI) technologies of late have prompted the increasing applications of digital twins (DiTs) to various sectors for boosting effectiveness and productivity. DiTs have been envisioned to possess immense potential for transforming numerous domains and sectors in the recent report of National Academics due to their powerful real-time decision-making based on modeling and simulating physical systems. This paper deals with a novel design of digital twin-aided municipal traffic control (DiTAT) for best traffic management over a targeted municipal region, based on real-world traffic video imagery gathered by available roadside surveillance cameras. DiTAT analyzes sequences of video frames to extract traffic volume details, including the *start time*, *speed*, *incoming zone*, and *outgoing zone* of every vehicle in existence. Being DiT-based, DiTAT employs the automated, open-source traffic simulator (SUMO) as the digital twin of the physical roadway configuration over the target region to try various traffic light control settings under the extracted traffic volume details for identifying the most favorable setting. The identified setting is then sent to the physical roadway traffic lights for realization to manage traffic during the next time window, when its resulting traffic is simulated by SUMO again to get the best setting for the subsequent time window reactively. This process repeats continuously window by window, with a bidirectional interplay between SUMO simulation and physical traffic for the target region. DiTAT is demonstrated to lift transport performance under real-world traffic scenarios.

Keywords: Digital Twins, SUMO Traffic Simulation, Traffic Control, Video Image Processing, YOLO (You Only Look Once) AI Tool

1. Introduction

Evolving over time, DiTs [1] enable effective optimization persistently by monitoring system behaviors continually. Uninterrupted system monitoring for traffic control is realized by roadside surveillance cameras, whose deployment has grown rapidly in many municipalities. While the concept of digital twins (DiT) has gained attention in recent technological advancements, its application to computationally efficient traffic control remains to be further explored. This paper deals with a novel design on digital twin-aided municipal traffic control (DiTAT) for best traffic management over a targeted municipal region of interest. DiTAT leverages live traffic video streams obtained from physical roadways over a target municipal region to identify the most desirable traffic light setting on the roadways in real-time. It employs an efficient AI-based object detection and tracking technique called YOLO (You Only Look Once) [2] to extract traffic volume details from video streams, including the start time, speed, incoming zone, and outgoing zone of every vehicle in existence therein. Extracted traffic volume details are then inputted to an automated, open-source traffic simulator (SUMO) [3] [4], which serves as the DiT of the physical roadways at the targeted region to simulate transportation performance under various traffic light settings for identifying the most desirable setting with the best performance in real-time. The transport performance metrics include throughput over the physical roadways and the mean travel time of vehicles between when they come to exist in video streams and when they exit.

1.1 Motivation and Challenges

Existing efforts have focused on developing transportation systems with intelligence [5] [6] to provide users with more information content and safety and to increase the level of interactions. From the end user and municipal transportation infrastructure standpoints, optimizing traffic performance naturally stands out as the key objective. Such optimization requires external monitoring continuously as traffic volumes fluctuate over time, usually achieved by aerial cameras positioned at critical intersections. These stationary cameras help to capture road conditions and traffic flows, producing real-time streams of videos for analyses. Video streams contain moving entities such as vehicles and motorcyclists/cyclists, which are analyzed over consecutive video frames to get their times to exist plus travelling speeds and directions needed for simulating traffic flows for transportation performance optimization. This real-time surveillance is critical for agile decision-making on-site and for long-term traffic optimizations, based on changing traffic flow patterns dictated by date/time, weather conditions, and special events taking place in the area. The challenges of traffic performance optimization include (1) obtaining traffic volume details from sequences of video frames rapidly and automatically, (2) deciding the most favorable traffic light settings at roadway intersections based on observed traffic to optimize transport performance metrics reactively, and (3) predicting the upcoming traffic volumes for identifying the best traffic light setting proactively, in particular in response to accidents, emergency situations, special events, and extreme weather conditions. The first challenge can be met with online video processing for object detection and tracking, e.g., YOLO [2]. SUMO (Simulation of Urban MObility) simulator inputted by analyzed or predicted traffic volumes to imitate roadway traffic serves as the digital twin of physical roadway traffic situations, able to try various traffic light settings (referring to the green-yellow-red cycles) for choosing the best setting in terms of performance metrics. This way enables traffic control for best transport performance reactively, based on real-time traffic volumes to address the second challenge. The third challenge is not dealt with in this article, left for future investigation.

1.2 Background and Pertinent Work

This section provides background and existing work pertinent to our paper, including DiTs, traffic control, YOLO, and SUMO, as follows.

1.2.1 Digital Twins (DiTs)

Digital Twins (DiTs) are in a wide range of usages these days, especially for the many-scenario optimization cases like traffic control [5] [6] [7] [8], manufacturing [9], and electric power systems [10]. DiTs typically involve the physical-to-virtual (PTV) twinning and virtual-to-physical (VTP) twinning processes [11] [12]. The physical side of a DiT refers to a real entity or system, whereas its virtual side incurs a high-fidelity software implementation of the physical entity/system, able to faithfully realize changes in the physical side at the same rate. Hence, changes in the physical or virtual side should be mirrored to the other side [11], with both sides exhibiting identical states at all times to obtain a fully twinned system.

DiTs have been applied for traffic control to explore various aspects, ranging from the development of intelligent transport systems [5] to traffic simulation [6] and traffic flow generation for highways [14]. Specifically, intelligent transport systems treated in [5] resorted to innovative developments in modeling transport systems, regulating traffic flows, and providing end users with greater information content and safety, able to qualitatively increase the level of interactions between road users in comparison with conventional transport systems. Later work on motorway traffic simulation with real-time data integration during system run-time was pursued [6], built on a developed novel paradigm in microscopic motorway traffic modeling. It was demonstrated by a continuously synchronized digital model to simulate on-the-fly synchronized digital replicas of real traffic utilizing SUMO. Meanwhile, an end-to-end traffic flow generation for implementing digital twin highway systems was treated in [13], employing a paired radar-camera sensing system and a developed fusion framework. It followed a unified coordinate system to combine the traffic flow data among multiple sites at a targeted highway to construct a dynamic real-time traffic flow digital twin model for the highway. All aforementioned DiT pursuits on traffic control focused on intelligent transportation, traffic simulation on motorways, or highway traffic flow generation for the digital twin model, instead of the aim of this work to apply DiTs for traffic light setting optimization to get the highest throughput and/or the shortest travel time.

1.2.2 Traffic Control

Municipal traffic management is often acute, previously solved by control centers. Such management by a control center can fall short in timely response to actual traffic scenarios on roadways, as it typically follows preset traffic light patterns at roadway intersections. In addition, it becomes more unlikely to yield efficient transport flows on roadways when the vehicle-to-population ratio grows, thus calling for digitizing transport via intelligent systems [5]. An intelligent transport system makes it possible to deal with several traffic-related tasks, providing critical services for city traffic police, ambulances, fire departments, traffic alerts that inform drivers of real-time traffic changes or hazards, etc. [14].

1.2.3 YOLO (You Only Look Once)

As a popular AI tool with high efficiency, YOLO performs real-time object detection and tracking speedily in a single pass, making it ideal for dynamic environments where real-time processing is critical [2]. Integrated with a Deep Neural Network (DNN), YOLO can be used effectively for tracking both moving and stable objects in various environments. For live dynamic environment analysis, especially in the context of tracking moving/stable objects, an agile and responsive framework could incorporate the following components: static objects (e.g., buildings) and dynamic objects (e.g., moving cars and pedestrians). YOLO is effective, able to accurately classify objects and locate them in real time. Once an object is detected, its trajectory can be tracked across successive frames. For moving objects, algorithms like SORT (Simple Online and Realtime Tracking) [15] or Deep SORT (Deep Learning-based SORT) [16] can be used to link detections between frames and track the movement. These tracking algorithms are typically faster and more accurate when paired with YOLO.

YOLO operates through a series of steps [2]. First, the input image is divided into a grid, where each grid cell is responsible for making object predictions. If an object is detected within a grid cell, YOLO makes successive predictions for its attributes, such as the class (e.g., a car) and bounding box coordinates. YOLO applies the *non-maximum suppression algorithm* [17] to address overlapping bounding boxes and ensure precise object localization, selecting the best bounding box for the detected object. The YOLO network architecture mainly consists of cascaded convolutional and fully connected layers, enabling efficient feature extraction and classification.

1.2.4 SUMO

SUMO is a traffic modelling and simulation platform that leverages a digital environment to replicate real-world traffic dynamics, particularly well known for its microscopic traffic simulation capabilities [3] [4] [18]. In general, traffic control simulators support various levels of analysis: macroscopic, microscopic, mesoscopic, and submicroscopic ones. In a macroscopic scenario, traffic density is managed based on average vehicle dynamics. The microscopic level focuses on individual vehicles, allowing for more detailed behavior analysis. The mesoscopic approach combines both macroscopic and microscopic elements, offering a balanced perspective. Submicroscopic simulation extends the analysis further by considering individual vehicles along with their surrounding sensors and on-board devices, providing a highly detailed traffic representation [19].

SUMO, as a microscopic simulation platform, supports multi-modal simulation, including vehicles, motorcycles, bicycles, pedestrians, and public transport. It also integrates traffic signal control and allows the use of *OpenStreetMap* [20], for realistic road network representations. These features make SUMO simulation a versatile tool for applications, ranging from intelligent transportation systems to smart city planning [3] [4] [18].

While some simulation platforms accept video inputs, SUMO simulation primarily relies on user-defined control for simulation scenarios. In this work, we enhance SUMO's capabilities by incorporating real traffic videos as an input source. Through a preprocessing step for object detection and tracking, we generate Extensible Markup Language (XML)-based simulation files that automate scenario creation and control within SUMO simulation seamlessly.

2. DiTAT Design Overview

As the first step for validating our developed DiTAT, real-time data obtained from real-world video clips is first mirrored by SUMO simulation to create a dynamic and accurate DiT of the current traffic scenario. A critical innovation in this way is using short video clips of current traffic to recreate and parameterize the simulation for multiple traffic scenarios. This DiT so created enables the exploration of various scenarios, helping to analyze the most suitable time interval for regulating the traffic light settings to better traffic management, as the result of improved traffic performance metrics (e.g., the throughput and mean travel time of each vehicle). These decisions include updating traffic light timings, alerting traffic police, and broadcasting updates via GPS systems to inform drivers of real-time traffic changes or hazards.

Figure 1 illustrates the proposed design of DiTAT, where the SUMO simulation is employed as the digital twin of the physical roadway configuration over the target region. There are mainly two sources of data in support of SUMO simulation, with static data and dynamic data referring respectively to road network layout information and to information about the traffic volumes including vehicle movement details. These static and dynamic data are extracted from video analysis on the CCTV-captured real-time video clips via YOLO. They are further converted into XML files, which can run in simulation software (like SUMO) to simulate resulting traffic flows.

2.1 Static Data

Traffic control systems and road network layout constitute static data which supply fixed road information for simulation. The source of our static data is from CCTV-captured traffic videos [21]. Figure 2 shows the screenshot of the real-world CCTV captured traffic data.

The surveillance camera videos provide insights into the number of lanes, intersection type, and the presence of traffic lights. Attributes, like the type in the node file, are used to define the role of each node, categorizing it as either a traffic light or a priority intersection. Additionally, other key attributes (such as the number of lanes, priority, etc.) are gathered to ensure an accurate representation of the road network. Using static data, we generated the network file, which is validated using *OpenStreetMap* (OSM). Figure 3 shows the manually generated network layout. OSM gives the privilege to select the desired location and provides the network layout, including detailed road layout, lane with its configurations, and traffic lights. The downloaded network file was employed as a reference to validate our manually defined road network layout. As the static data does not change, it serves as the initial input for our simulation.

2.2 Dynamic Data

Vehicle movements, speed changes, and direction changes are essential dynamic data necessary for simulating real-world traffic videos in a simulation environment. Traffic data is extracted from real-world traffic video streams and successfully converted into XML files to accurately replicate real traffic flows in the SUMO simulation environment.

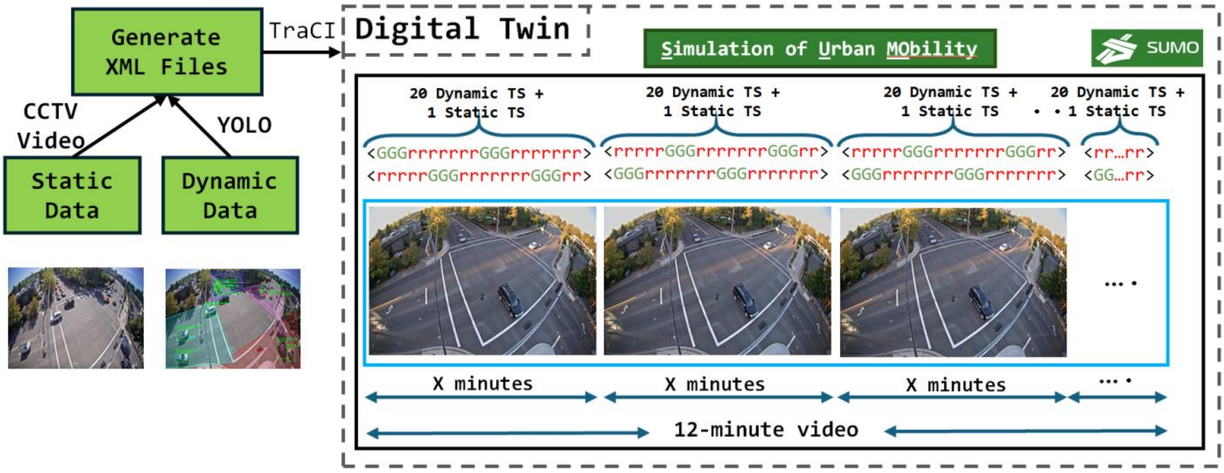

Figure 1. *DiTAT System Overview (TS: Traffic Scenario).*

Figure 2. *Real-World Road Layout.*

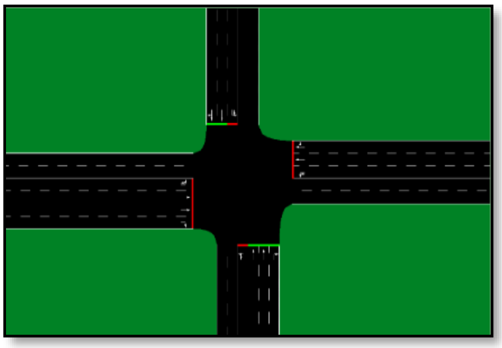

Figure 3. *Manually Generated Road Layout.*

2.2.1 Vehicle Detection and Tracking Using YOLO

YOLO [2] is a deep learning-based model that is used effectively for detecting and tracking objects. We are using a pre-trained YOLO model that identifies vehicles in each frame and assigns a unique identifier (Vehicle ID) to each vehicle. As illustrated in Figure 4, YOLO assigns a green bounding box around each detected vehicle, which helps to visually represent the detected vehicle in each frame. Along with that, the positional information (X, Y coordinates) is also provided by YOLO. This allows continuous tracking of each individual vehicle across each frame and gives information as they move across different frames. YOLO also has the ability to define the class of vehicles. The type of vehicles can be cars, buses, motorcycles, cycles, trucks, etc. For our case, we are only considering 3 vehicle types: cars, buses, and trucks.

2.2.2 Zone-Based Analysis on Object Movements

The real-world video is divided into four zones: east, west, north, and south, as visible in Figure 4. This way allows to better analyze object movements, where an object can be a vehicle or motorcyclist/cyclist. It helps to determine the entry and exit zones of each object, essential for proper tracking of moving objects. Each moving object is assigned an entry zone, from which the object enters the intersection. Additionally, the exit zone for each object is also captured. Each object's entry and exit zones provide valuable insights into the traffic patterns and flows. They also make it possible to get the route information for each moving object. This zone-based analysis is also essential for understanding traffic congestion, as it can get information on how long objects are stationary in one zone, mostly because of waiting at a red light. For simplicity, a moving object is referred to as a vehicle hereafter.

2.2.3 Travel Time and Departure Time

To measure the travel time of a vehicle, two reference lines are defined within the traffic frames: a start line and an end line, as illustrated in Figure 4. These lines serve as fixed markers to determine how long it takes for a vehicle to traverse a particular section of the road.

A counter that represents the frame number is assigned to each video frame, and it increments as the video sequence progresses. Since, we have defined two different reference lines, representing the start and end time points. Whenever a vehicle crosses the start line, the frame number at that instance is recorded, and whenever the same vehicles cross the end line, the corresponding frame is noted as the end frame count. The difference between these two start and end frame times gives the number of frames it requires for the vehicle to travel between the two reference lines. The total time taken by the vehicle to travel (in seconds) is calculated by:

$$Travel\,Time = \frac{Frame\,Difference}{FPS} \tag{1}$$

Additional parameters, such as vehicle departure times, are also recorded, as they are required for simulation. The departure time of vehicles is determined based on when they enter the frame. The frame at which YOLO detects the vehicle is noted and divided by Frames Per Second (FPS) to obtain the time (in seconds) when the vehicle appears on the screen. That frame number is used for determining the departure time of the vehicle.

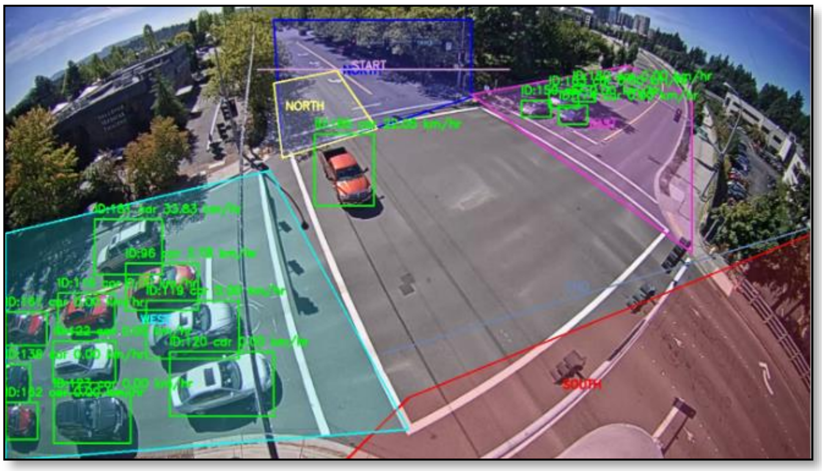

Figure 4. *Vehicle Detection and Tracking Using YOLO.*

2.3 Generation of SUMO-Compatible XML Files

The static road network layout file and the dynamic vehicle route file are necessary to run a SUMO simulation.

The generation of static road network layout involves 3 important files:

- Node File (.nod.xml): This file defines the intersection points and the junctions in the road. It also includes the signal locations and priority intersections.
- Edge File (.edg.xml): It defines the road segments that connect different nodes, which finally represent the real-world road links.
- Type File (.typ.xml): The file contains the characteristics of the road, such as the number of lanes and speed limits.

Once the node, edge, and type XML files are generated, they are converted into a network file (.net.xml) using the standard SUMO tool of 'netconvert' for converting plain XML files into SUMO-compatible network files (.net.xml). Any obtained SUMO network file (.net.xml) represents a 4-way intersection, as depicted in Figure 3. It is validated against the OSM-derived network layout to ensure that it accurately represents the real-world traffic infrastructure as derived from OSM data.

Apart from the static road network file generation, a dynamic route file (.rou.xml) is required to simulate the vehicle movement in the network. The information extracted from YOLO along with other parameters during real-time video analysis is used to make the route file. A simple conversion using python is carried out where every tracked vehicle information is converted into XML files as required by the simulator. The details extracted in the dynamic video analysis, such as vehicle id, departure time, speed, route, and vehicle type for each vehicle are used to create the route (.rou.xml) file. This route file serves as a dynamic input to the SUMO simulator to replicate the real video traffic flow in the given network manually created. Such an approach ensures the accurate and real representation of traffic volumes as in the real world.

Lastly, in order to be able to run the simulation in SUMO, a SUMO configuration (.sumocfg) file is necessary, obtained by combining static and dynamic data and by referring to the location of each file. The configuration file is the main input to initiate SUMO simulation. It indicates the road network to be used, and vehicle traffic in the simulation. Additionally, it also holds settings like simulation duration and step length. Input with this configuration file, simulation can be performed.

2.4 SUMO Traffic Simulation

The open-source traffic simulator (SUMO) serves as the DiT of the physical roadways to simulate various traffic scenarios to figure out the most favorable setting for the subsequent time window. DiTAT simulates 20 different traffic scenarios, plus one static traffic scenario in each 12-minute real-world video. Different decision intervals are evaluated, represented by X, as shown in Figure 1, where X means the time at which the decision is made to adjust the traffic light setting (i.e., the green-yellow-red cycle). For this study, the decision intervals of 2 minutes, 3 minutes, 4 minutes, and 6 minutes are explored. A thorough analysis is done at each time interval using the key transport performance metrics: throughput and mean travel time. The throughput represents the number of vehicles passing through the intersection per minute under the current traffic light setting, whereas the mean travel time indicates how long on average it takes for a vehicle to travel from the entry line and the exit line.

At every X interval, different traffic light settings are tested to obtain their throughput and average time via SUMO simulation, as digital twin-aided control. The setting that yields the best throughput is desirable, chosen for use in the next time window. This process repeats window by window, which ensures robust and dynamic traffic control.

The time period of each traffic light (green, yellow, and red) is dynamically adjusted through TraCI (Traffic Control Interface) [22], which is a Python-based API that gives access to control SUMO traffic simulation in real-time. TraCI provides the ability to extract real-time vehicle information during SUMO simulation, such as the current vehicle position, lane number, and speed. Additionally, it provides flexibility for the dynamic modifications of the traffic signals.

For instance, consider the traffic intersection shown in Figures 5, 6, and 7, where the intersection layout has the lane number defined for each direction. Green lines in Figure 5 represent the traffic flow within each lane during the course of simulation. The phase for Figure 5 is defined as "GGGrrrrrrrGGGrrrrrrr", denoting a 15-second green light duration for the North-South direction and a 15-second red light duration for the remaining directions. This phase is followed by a yellow light duration for 5 seconds in the North-South direction and then a red light in the remaining direction, as depicted in Figure 6, followed by green lights for turning to the east and west direction respectively for 5 seconds, as illustrated in Figure 7. Afterward, the traffic light cycle is for East-West traffic in the next phase.

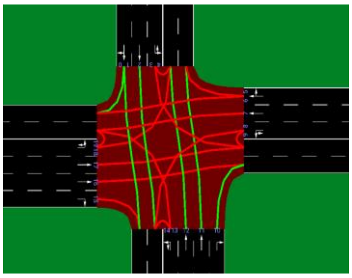

Figure 5. *Green Phase for NS Direction.*

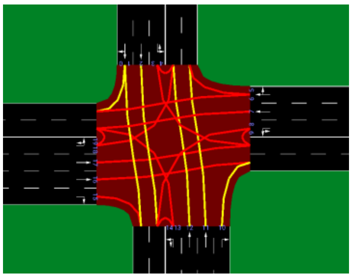

Figure 6. *Yellow Phase for NS Direction.*

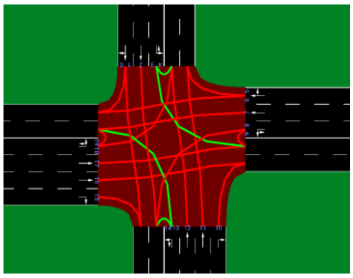

Figure 7. *Red Phase for NS Direction.*

A total of 21 scenarios are tested in a similar manner but having varying cases. Some of the key scenarios include:

- Balanced traffic movement, with an equal amount of green light for the NS and EW directions each.
- Longer green lights are provisioned for direction with greater vehicle movements.
- Shorter green lights for both directions upon more frequent phase changes.
- Priority for turning traffic to enable vehicles with uninterrupted flows.

They ensure that DiTAT employs possible scenarios before choosing the optimal setting for the next time window.

3. Evaluation and Result Discussion

This section details our DiTAT evaluation and obtained results in comparison with those under the static traffic light setting for the data extracted from real-world 12-minute traffic videos available to the public [21]. The transport performance metrics of our evaluation are the throughput (the number of vehicles exiting from a roadway intersection per minute) and the mean travel time (for a vehicle to travel from its entering zone of the intersection to its exiting zone, in second). Bar graphs below show the comparative performance metric results of DiTAT and static traffic light settings under different decision intervals (2, 3, 4, and 6 minutes). DiTAT adjusts the traffic light settings at decision time points adaptively, whereas the static setting employed a fixed traffic light setting. Our evaluation was conducted on various real-world traffic videos, with the results of two representative 12-minute videos taken at one traffic intersection (i.e., 116th & NE12th St.) in the City of Bellevue, Washington [21].

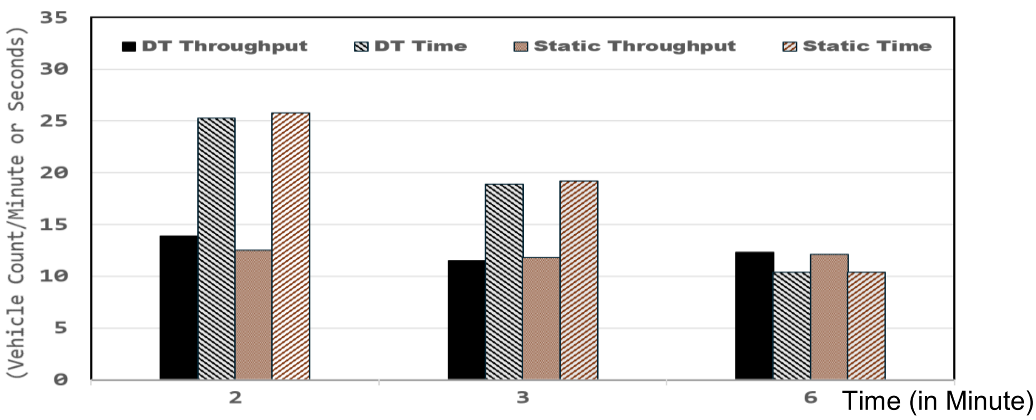

Figure 8. *Comparative Results of DiTAT and Static Traffic Light Settings.*
(for the first video under the traffic light setting time of 6 minutes, with Y-axis referring to vehicle count per minute for throughput and to the mean travel time of a vehicle across the pair of reference lines).

3.1 Evaluation Results

Figure 8 shows the evaluation results for the first video when the traffic light setting is changed once in 6 minutes (i.e., the time window for light setting changes), for the traffic light cycle time to range from 2 minutes to 6 minutes. The unit of Y-axis is the vehicle count per minute for throughput results, and it is in seconds for mean travel time results. As can be seen, DiTAT gives rise to the best performance at the cycle time = 6 minutes, if both metrics are considered. For the cycle time of 2 minutes, DiTAT has a slightly larger throughput value (13.9 versus 12.3) but a substantially longer mean travel time (25.3 seconds versus 10.4 seconds), in comparison to the cycle time of 6 minutes. When compared with the static setting under the cycle time of 6 minutes, DiTAT is marginally superior in both metrics.

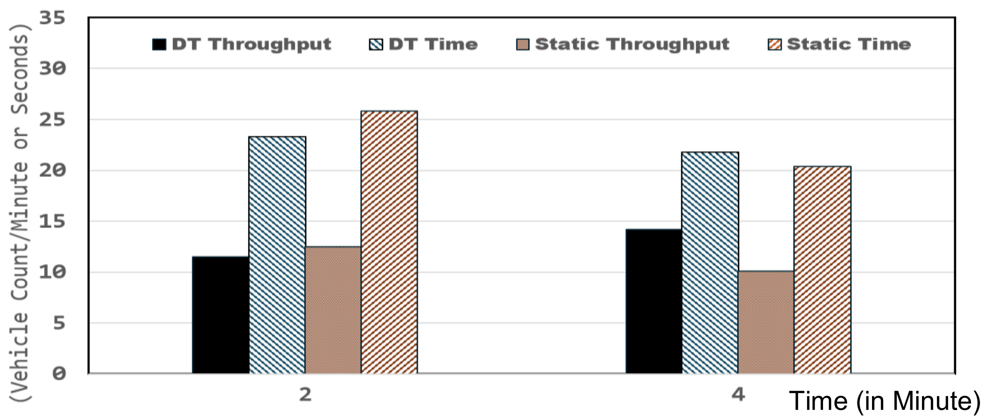

Figure 9. *Comparative Results of DiTAT and Static Traffic Light Settings.*
(for the first video under the traffic light setting time of 4 minutes, with Y-axis referring to vehicle count per minute for throughput and to the mean travel time of a vehicle across the pair of reference lines).

The evaluation results for the first video when the traffic light setting is changed once in 4 minutes are demonstrated in Figure 9, where the traffic light cycle time equals 2 minutes and 4 minutes. It is found that DiTAT enjoys better performance for the cycle time of 4 minutes, exhibiting a higher throughput and yet a smaller travel time, when compared with those for the cycle time of 2 minutes. Furthermore, DiTAT under the cycle time of 4 minutes has a markedly higher throughput (of 14.2 versus 10.1) in comparison to its static setting counterpart, albeit a slightly longer travel time (21.8 seconds versus 20.4 seconds).

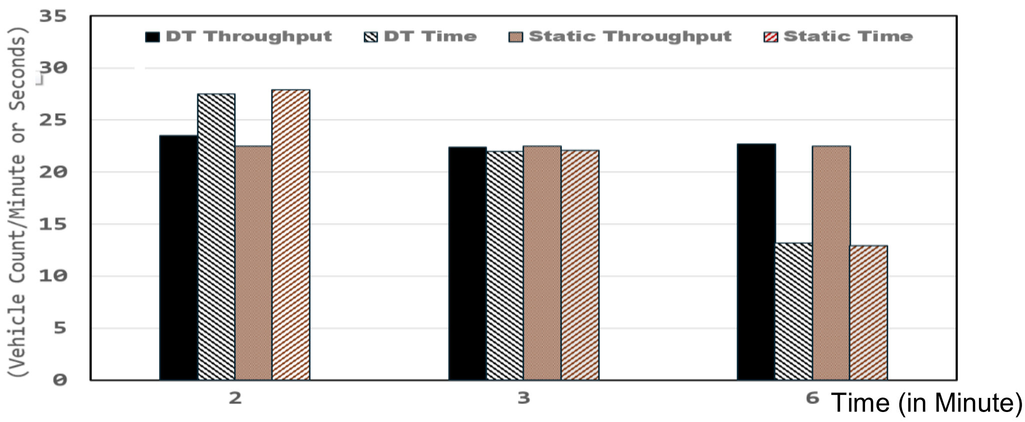

Figure 10. *Comparative Results of DiTAT and Static Traffic Light Settings.*
(for the second video under the traffic light setting time of 6 minutes, with Y-axis referring to the vehicle count per minute for throughput and to the mean travel time of a vehicle across the pair of reference lines).

Figure 10 depicts the evaluation results for the second video when the traffic light setting is changed once in 6 minutes (i.e., the time window for light setting changes), for the traffic light cycle time to range from 2 minutes to 6 minutes. The unit of Y-axis is the vehicle count per minute for throughput results, and it is in seconds for mean travel time results. As can be seen, DiTAT again has the best performance at the cycle time = 6 minutes, when both performance metrics are taken into consideration, since its throughput is slightly smaller (22.7 versus 23.5) but its mean travel time is lower substantially (13.2 seconds versus 27.5 seconds) when compared with the cycle time of 2 minutes. DiTAT has compatible performance when compared with its static traffic light setting counterpart.

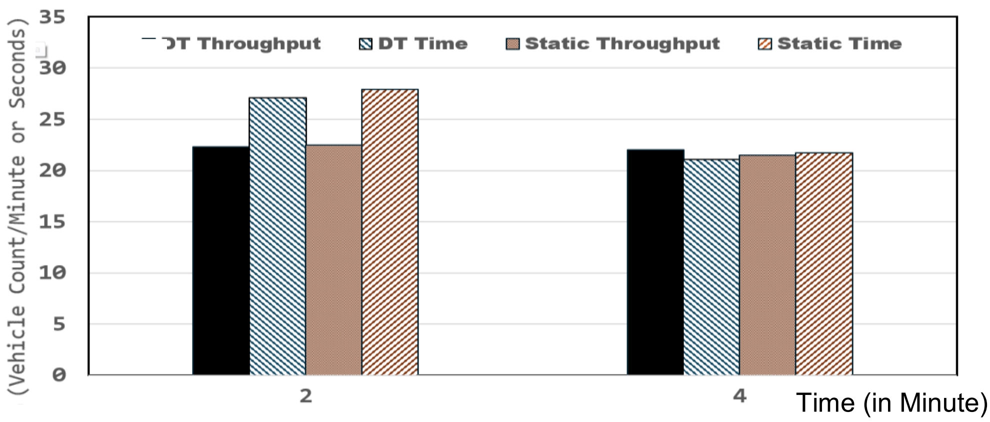

Figure 11. *Comparative Results of DiTAT and Static Traffic Light Settings.*
(for the second video under the traffic light setting time of 4 minute, with Y-axis referring to the vehicle count per minute for throughput and to the mean travel time of a vehicle across the pair of reference lines).

The evaluation results for the second video when the traffic light setting is changed once in 4 minutes are shown in Figure 11, where the traffic light cycle time equals 2 minutes and 4 minutes. It is found that DiTAT enjoys better performance for the cycle time of 4 minutes, exhibiting a slightly lower throughput but a noticeably smaller travel time, when compared with those for the cycle time of 2 minutes. Additionally, DiTAT under the cycle time of 4 minutes enjoys a marginally higher throughput (of 22 versus 21.5) and a measurably smaller travel time, in comparison to its static setting counterpart.

3.2 Discussion

Overall, the traffic light setting time of 6 minutes for DiTAT appears advantageous when both transport performance matrices are taken into consideration, always outperforming its static light setting counterpart. DiTAT may exhibit a markedly higher throughput than its static counterpart, for the cycle time of 4 minutes.

DiTAT is suited for deployment in the field since it makes near real-time control decisions on the fly. Its involved YOLO starts to process video frames as soon as observing several of them, with its processing times on the arrival frame stream no more than the inter-frame time interval (~ 35 *ms*), The SUMO simulator takes some 50 seconds to start and can be launched soon after YOLO starts its process, waiting for the output file produced by YOLO to complete before its simulation scenarios actually proceed. Multiple threads carry out simulation scenarios concurrently, as they are all independent. Under the MacBook Pro we employed, each simulation thread takes some 50 *ms* to consume the YOLO output data corresponding to 1-minute video frames. Hence, the traffic light setting decision will be available in less than 1 second after the video frames of a traffic light cycle (say, 3 minutes) end, achieving near real-time decision-making. In addition, DiTAT can work (with proper result adjustments) when the surveillance camera moves to the sides of an intersection. The adjustment depends on the position after a move.

4. Conclusion

This proposed design on digital twin-aided municipal traffic control (DiTAT) seeks to boost traffic management in municipalities, where abundant road-side surveillance cameras exist for monitoring traffic continuously. Data observed by those cameras are pre-processed in real time by YOLO as the input of a digital twin (DiT) realized by the SUMO simulator, which simulates the physical roadway configuration of a target municipal region with inputted traffic volumes to arrive at the best control settings of all involved traffic lights at an intersection in the next time window reactively. Our performance evaluation based on two real-world traffic videos indicates that DiTAT leads to improved transport performance, possibly with noticeable throughput improvement when the traffic light cycle time equals 4 minutes. Further investigation into proactive traffic light settings for DiTAT is underway, relying on traffic predictions for more pronounced performance improvement.

Data Availability Statement

The dataset used in this article, Bellevue Traffic Video Dataset, is available at [21]. Our implementation codes and obtained sub-datasets for the simulation are also shared at the GitHub repository, available to the public for research purposes at https://github.com/shebywiliam-sjr/TrafficDT and https://github.com/reetipd/SumoFiles.

Author Contributions

Reeti Pradhananga: Conceptualization, Methodology, Software, Investigation, Writing – Original Draft. **Shelby Williams**: Data Curation, Formal Analysis, Visualization, Writing – Review & Editing. **Sercan Aygun**: Supervision, Methodology, Writing – Original Draft, Review & Editing. **Li Chen**: Investigation, Resources, Validation, Writing – Review & Editing. **Yazhou Tu**: Methodology, Software, Formal Analysis, Validation. **Whitney Crow**: Supervision, Validation. **Sathyanarayanan Aakur**: Supervision, Conceptualization, Writing – Review & Editing. **Nian-Feng Tzeng:** Project Administration, Supervision, Investigation, Funding Acquisition, Writing – Review & Editing.

Competing interests

The authors declare that they have no competing interests.

Funding

This work was supported in part by National Science Foundation (NSF) under Grant OIA-2019511.

References

[1] National Academies of Science, Engineering, and Medicine, "Foundational Research Gaps and Future Directions for Digital Twins (2024)," *The National Academies Press*, 2024. URL – https://doi.org/10.17226/26894.
[2] Explore Ultralytics YOLOv8, January 2023. URL – https://docs.ultralytics.com/models/yolov8/.
[3] M. Behrisch, L. Bieker-Walz, J. Erdmann, and D. Krajzewicz, "SUMO – Simulation of Urban Mobility: An Overview," in *Proceedings of 3rd International Conference on Advances in System Simulation* (*SIMUL*), October 2011.
[4] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent Development and Applications of SUMO – Simulation of Urban Mobility," *International Journal on Advances in Systems and Measurements,* vol. 5, pp. 128 – 138, 2012.
[5] A. Rudskoy, I. Ilin, and A. Prokhorov, "Digital Twins in the Intelligent Transport Systems," *Transportation Research Procedia*, vol. 54, no. 7, pp. 927 – 935, 2021. Doi: https://doi.org/10.1016/j.aei.2022.101858
[6] K. Kušić, R. Schumann, and E. Ivanjko, "A Digital Twin in Transportation: Real-Time Synergy of Traffic Data Streams and Simulation for Virtualizing Motorway Dynamics," *Advanced Engineering Informatics*, vol. 55, January 2023. https://doi.org/10.1016/j.aei.2022.101858
[7] S. Dasgupta, M. Rahman, A. Lidbe, W. Lu, and S. Jones, "A Transportation Digital-Twin Approach for Adaptive Traffic Control Systems," *Transportation Research Record*, 2021. https://doi.org/10.48550/arXiv.2109.10863
[8] V. Lehtola, M. Koeva, S. Elberink, P. Raposo, J. Virtanen, F. Vahdatikhaki, and S. Borsci, "Digital Twin of a City: Review of Technology Serving City Needs," *International Journal of Applied Earth Observation and Geoinformation*, vol. 114, 2022. Doi: https://doi.org/10.1016/j.jag.2022.102915
[9] M. Ebni, S. M. Bamakan, and Q. Qu, "Digital Twin-Based Smart Manufacturing: From Design to Simulation and Optimization Schema," *Procedia Computer Science*, vol. 221, August 2023, pp. 1216 – 1225. https://doi.org/10.1016/j.procs.2023.08.109
[10] M. A.M. Yassin, A. Shrestha, and S. Rabie, "Digital Twin in Power System Research and Development: Principle, Scope, and Challenges," *Energy Reviews*, vol. 2, September 2023. https://doi.org/10.1016/j.enrev.2023.100039
[11] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, "Characterising the Digital Twin: A Systematic Literature Review," *CIRP Journal of Manufacturing Science and Technology*, vol. 29, May 2020, pp. 36-52. https://doi.org/10.1016/j.cirpj.2020.02.002
[12] D. Menon, B. Anand, and C. Chowdhary, "Digital Twin: Exploring the Intersection of Virtual and Physical Worlds," *IEEE Access*, vol. 11, July 2023, pp. 75152 – 75172. https://doi.org/10.1109/ACCESS.2023.3294985
[13] Y. Li and W. Zhang, "Traffic Flow Digital Twin Generation for Highway Scenario Based on Radar-Camera Paired Fusion," *Scientific Reports*, Nature, 2023. https://doi.org/10.1038/s41598-023-27696-z
[14] Y. Shamlitsky, O. Aleksey, E. Morozov, and T. Strekaleva, "Using Digital Twins to Manage Traffic Flows," in *Proceedings of E3S Web of Conferences*, EDP Sciences, 2024. https://doi.org/10.1051/e3sconf/202447104027
[15] N. Wojke, A. Bewley, and D. Paulus, "Simple Online and Realtime Tracking with a Deep Association Metric," In *Proceedings of 2017 IEEE International Conference on Image Processing* (*ICIP*), September 2017. https://doi.org/10.48550/arXiv.1703.07402
[16] A. Kouidri, "Mastering Deep Sort: The Future of Object Tracking Explained," November 2023. URL – https://www.ikomia.ai/blog/deep-sort-object-tracking-guide.
[17] N. Zaghari, "The Improvement in Obstacle Detection in Autonomous Vehicles Using YOLO Non-Maximum Suppression Fuzzy Algorithm," *Journal of Supercomputing*, vol. 77, November 2021, pp. 13421–13446. DOI: https://doi.org/10.1007/s11227-021-03813-5.

[18] S. Haddouch, H. Hachimi, and N. Hmina, "Modeling the Flow of Road Traffic with the SUMO Simulation," in *Proceedings of 4th International Conference on Optimization and Applications* (*ICOA*), April 2018.
[19] P. A. Lopez *et al*., "Microscopic Traffic Simulation Using SUMO," in *Proceedings of 2018 21st IEEE International Conference on Intelligent Transportation Systems* (*ITSC*), November 2018, pp. 2575-2582. DOI: 10.1109/ITSC.2018.8569938.
[20] OpenStreetMap, URL – https://www.openstreetmap.org/. Access Date: 01/19/2025.
[21] Bellevue Traffic Video Dataset, URL – https://github.com/City-of-Bellevue/TrafficVideoDataset. Access Date: 10/8/2024.
[22] Traci, URL – https://sumo.dlr.de/docs/TraCI/Interfacing_TraCI_from_Python.html. Access Date: 12/10/2024.