International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

IJISAE

ISSN:2147-6799 www.ijisae.org Original Research Paper

Real-Time Traffic Study Using Smart Technology

Kannadasan B. [1], Yogeswari K. [2]

Submitted: 04/02/2024 **Revised**: 12/03/2024 **Accepted**: 18/03/2024

Abstract: The concomitant spike in traffic creates substantial obstacles to effective traffic management as metropolitan areas experience rapid development. This article details an innovative project in the Australian city of Liverpool that uses smart visual sensors to conduct a real-time traffic study. To monitor various forms of transportation in real-time while protecting the privacy of individuals, these sensors were created as a pilot project and use computer vision and deep neural networks. The study used a complete town center dataset to evaluate the edge-computing device's performance. We present the Agnosticity Framework, an open-standards system that can read and write data from various sensors. Two experimental results show that the framework improves our general understanding and control of urban traffic dynamics. This research provides important insights for future smart city projects and helps create intelligent and privacy-conscious solutions for urban traffic studies.

Keywords: *real time, traffic study, smart technology, Agnosticity Framework, Australian city*

1. Introduction

In the fast-paced and ever-expanding urban landscapes of the 21st century, the challenges associated with traffic congestion have become increasingly pronounced [1]. As cities grow and populations surge, traditional traffic management strategies struggle to keep pace with the escalating demands on transportation infrastructure [2]. In response to this pressing issue, the integration of smart technology is emerging as a transforming force, offering innovative solutions to monitor and manage traffic patterns in real-time [3]. This paradigm shift enhances our understanding of urban mobility and lays the foundation for informed decision-making, ultimately paving the way for more innovative, efficient cities [4].

Smart technology has ushered in a new era in transportation management, allowing for the collection and analysis of real-time data on traffic conditions [5]. This data, drawn from a network of sensors, cameras, and other intelligent devices strategically positioned throughout urban environments, provides a comprehensive and up-to-the-minute perspective on the flow of vehicles [6]. By advancing data analytics & harnessing IoT (Internet of Things) power, traffic authorities can gain valuable insights into congestion patterns, peak travel times, and potential bottlenecks, formulating proactive strategies for alleviating congestion and improving overall traffic efficiency [7].

One of the advantages of 'real-time traffic study' through smart technologies lies in their ability to adapt dynamically to changing circumstances [8]. Unlike traditional traffic studies that rely on periodic surveys or historical data, smart technology enables continuous monitoring and adjustment [9]. This adaptability is crucial in dynamic urban environments where traffic conditions can evolve rapidly. As a result, city planners and traffic management authorities can respond in real-time to emerging challenges, optimizing traffic signals, rerouting vehicles, and implementing other responsive measures to enhance the overall flow of traffic [10].

Furthermore, integrating smart technology in traffic studies fosters a more inclusive and participatory approach to urban planning. By leveraging data from mobile applications, GPS devices, and social media platforms, authorities can engage with citizens to gather real-time feedback on traffic conditions [11]. This collaborative approach empowers individuals to make informed travel decisions and enables authorities to crowdsource valuable information about traffic incidents, road closures, and alternative routes [12]. The result is a more interconnected and responsive urban transportation ecosystem that benefits city planners and residents [13].

Therefore, the real-time traffic studies facilitated by smart technology represent a groundbreaking paradigm shift in urban mobility management [14]. As cities grapple with the complexities of rapid urbanization, these innovative solutions provide a pathway to more intelligent, adaptive, and citizen-centric transportation systems [15]. By

[1]*Research Scholar, Department of Civil Engineering, B. S. Abdur Rahman Crescent Institute of Science and Technology, Chennai, India.*
mail id: kdasan9@gmail.com
Orcid id: 0000-0003-1563-4068
[2]*Professor, Department of Civil Engineering, B. S. Abdur Rahman Crescent Institute of Science and Technology, Chennai, India.*
Corresponding author mail id: yogeswari@crescent.education
Orcid id: 0000-0002-7533-5193

harnessing the power of real-time data, cities can mitigate the challenges posed by traffic congestion and work towards building more sustainable, resilient, and livable urban environments for future generations.

1.1 Contributions

The novel contributions of this study are:

1. We introduce a 'smart visual sensor' ensuring 'real-time multi-modal tracking' while prioritizing citizens' privacy through advanced anonymization for Privacy-First Tracking.
2. We propose an interoperable system, the '*Agnosticity framework*', to collect, store seamlessly, and access data from various sensors, enhancing collaboration and scalability.
3. We develop an intelligent traffic signal algorithm that minimizes energy consumption by dynamically adjusting timings based on real-time traffic conditions.
4. We implement a priority system within traffic management for emergency vehicles, ensuring swift response times without disrupting regular traffic flow.

2. Related works

Mohammad et al. [16] (2020) introduced a three-layer VFC model for ‘dispersed traffic organization’, minimizing ‘Average Response Time (ART)’ significantly. The offloading scheme optimizes running & vehicles as ‘fog nodes’, addressing challenges in VFC-enabled traffic management.

Atta et al. [17] (2020) utilized RFID technology to dynamically control traffic signals based on real-time congestion, providing a unique approach for timing signals proportional to road congestion. The IoT-enabled sensors enable dynamic signal timings, minimizing congestion for enhanced communication technologies.

Ma et al [18] (2021) conducted study on ‘Mobility-Based Real-Time Air Pollution Exposure Assessment’ in Beijing. They demonstrated the impact of smart technologies on assessing individual-level pollution exposure, comparing ‘residence-based monitoring’ with mobility-based real-time assessment. They highlighted exposure level difference using different approaches, emphasizing the need for fine-resolution assessments.

Shengdong et al [19] (2019) conducted study on ‘intelligent Traffic Control System with Cloud Computing and Big Data Mining’. They addressed challenges in modern intelligent traffic systems, proposing an integrated cloud-based control system. They utilized deep learning for traffic flow prediction and intelligent optimization algorithms for real-time traffic flow control, proving the effectiveness through simulation results.

Babar and Arif [20] (2019) developed ‘three-phase architecture for real-time Big Data processing in a smart transportation system’. They utilized Spark and Hadoop for processing and highlights the scheme's effectiveness in terms of throughput in ‘IoT-based smart transportation’ environments.

Yang et al [21] (2020) developed a high-performance computing model using DBN and K-means for dynamic traffic planning based on real-time IoT and GIS data. They demonstrated model's precision in ‘optimal traffic network planning under real-time mass data situations and low cost’.

Yu and Gu [22] (2019) introduced a novel deep-learning model, the ‘graph convolutional generative autoencoder, for real-time traffic speed estimation’. They outperformed existing techniques in comprehensive case studies, emphasizing the model's superiority in traffic speed estimation.

Yu et al [23] (2020) developed a ‘deep learning-based traffic safety solution for a mixture of autonomous and manual vehicles in a 5G-enabled ITS’. They achieved high intention recognition rates, effectively improving accuracy and real-time intention recognition in a mixed traffic environment.

Chen et al. [24] (2021) introduced ‘gradient boosting partitioned regression tree model, for forecasting travel time based on big data from industrial IoT infrastructure’. They demonstrated enhanced predictive accuracy in real traffic IoT data compared to other computational methods.

Sarrab et al. [25] (2019) developed an ‘intelligent traffic monitoring system for a smart city, broadcasting congestion updates through roadside message units’. They provided ‘real-time traffic updates’ to improve mobility, with potential enhancements for ‘optimal re-route suggestions to drivers’.

Table 1: Summary of Research gaps

Ref No.	Author/Year	Method	Finding	Research Gap
[16]	Mohammad et al. (2020)	Three-layer VFC model for dispersed traffic organization	Reduced Average Response Time (ART) significantly.	Need study extension
[17]	Atta et al. (2020)	RFID technology for dynamic traffic signal control	Dynamic signal timings proportional to real-time congestion, minimizing congestion.	Probing optimal ways to adapt to varying traffic conditions in real-time.
[18]	Ma et al. (2021)	Mobility-Based Real-Time Air Pollution Exposure Assessment	Highlighted substantial differences in exposure levels between residence-based and mobility-based assessment.	Need for fine-resolution assessments in understanding individual-level pollution exposure.
[19]	Shengdong et al. (2019)	Cloud Computing and Big Data Mining for Traffic Control	Integrated cloud-based control system with deep learning for traffic flow prediction.	Addressing real-time challenges in traffic flow control and optimization within cloud-based systems.
[20]	Babar and Arif (2019)	Real-Time Data Processing in IoT-based Smart Transportation	Developed a 3-phase architecture for real-time Big Data processing.	Investigating scalability and adaptability of the proposed architecture in diverse smart transportation.
[21]	Yang et al. (2020)	Optimization of Real-Time Traffic Network Assignment	Utilized DBN and K-means for dynamic traffic planning. Demonstrated precision in optimal traffic network planning.	Examining the scalability and generalizability of the proposed high-performance computing model.
[22]	Yu and Gu (2019)	Graph Convolutional Generative Autoencoder for Speed Estimation	Outperformed existing techniques in traffic speed estimation.	Investigating the applicability of the proposed model to different transportation network structures.
[23]	Yu et al. (2020)	Deep Learning-Based Traffic Safety Solution for Autonomous and Manual Vehicles	Achieved high intention recognition rates in a mixed traffic environment.	Addressing the adaptability of the proposed solution to diverse traffic scenarios in a 5G-enabled ITS.
[24]	Chen et al. (2021)	'Pragmatic Real-Time Logistics Management with Traffic IoT Infrastructure'	Introduced gradient boosting partitioned regression tree model for travel time forecasting.	Investigating the applicability of the proposed model to various logistics scenarios and data conditions.

Table 1: Summary of Research gaps				
Ref No.	**Author/Year**	**Method**	**Finding**	**Research Gap**
[25]	Sarrab et al. (2019)	‘Real-Time Traffic Monitoring Systems Based on Magnetic Sensor Integration’	Developed an ‘intelligent traffic monitoring system for a smart city’. Aimed to improve mobility.	Assessing the scalability & practical implementation challenges of proposed traffic monitoring system.

2.1 Research gap

The collective findings from the reviewed studies reveal several common research gaps within the domain of smart traffic management. Firstly, there is a consistent need for investigations into the adaptability and scalability of proposed models and systems across diverse urban environments. Many studies have shown promising results within specific contexts, but understanding how these solutions perform under varying conditions remains essential for further research. Additionally, integrating these technologies often introduces new complexities and potential challenges, emphasizing the necessity for comprehensive studies on the practical implementation and real-world performance of smart traffic solutions. Furthermore, the identified research gaps include the need for standardized approaches to address privacy concerns in deploying smart traffic systems and exploring ways to enhance the resilience of these systems in the face of unforeseen disruptions or cyber threats. Bridging these gaps contributes to robust and universally applicable smart traffic management solutions.

3. Problem statement

The escalating urbanization of cities poses a critical challenge in the form of burgeoning traffic congestion, compromising the effectiveness of transportation systems and negatively impacting the quality of life for residents. Traditional traffic management strategies often need to improve dress multi-modal transportation's complexities and fail to adapt in urban scenarios. As a result, there is a pressing need for innovative solutions that leverage smart technology to alleviate congestion and enhance overall traffic management, with a particular emphasis on privacy preservation and adaptability to diverse urban environments.

4. Objectives

The novel objectives of this study are:

1. To dynamically reroute vehicles and optimize traffic signal timings in real-time, reducing urban congestion during peak hours.
2. To employ advanced deep neural networks & computer vision for real-time, privacy-preserving tracking of multiple transportation modes in urban environments.
3. To minimize energy consumption by implementing an intelligent traffic signal control algorithm that adjusts timing based on real-time traffic conditions.
4. To develop a priority system within the urban traffic management framework, identifying and prioritizing routes for emergency vehicles to ensure rapid response times while minimizing disruptions to regular traffic flow.

5. Methodology

For real-time monitoring of multi-modal transportation while protecting privacy of citizens, we develop & test an ‘edge-computing device’ which employs deep neural networks & computer vision.

5.1 Pilot Project

Using CCTV live feeds, the project intends to create and assess movement trackers. The heart of Liverpool was chosen as the experimental site. In order to track traffic patterns, twenty sensors will be placed over the downtown area. While fifteen make use of fixed CCTVs, five will be able to move around with the help of mobile CCTVs. Town center where the sensors are located on the map is seen in Fig 1. Twenty noise sensors & air quality are placed beside the ‘mobility trackers’ in this pilot project to assess the effect of traffic on air pollution and noise levels. Fig 1 illustrates one such case.

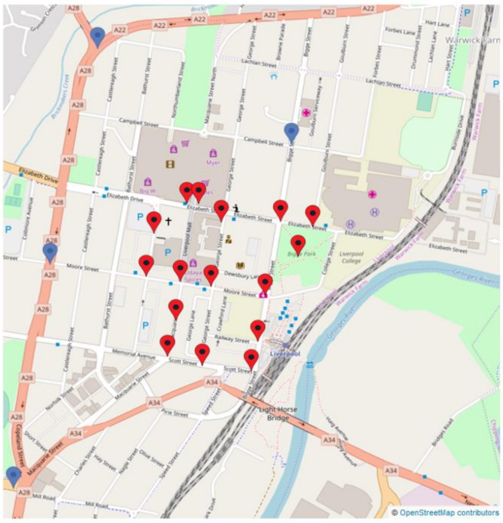

Fig 1: Town center map: (left): the twenty visual sensors' locations. Urban planners anticipate that the real-time data generated from such locations will assist them in updating the mobility plan of city. (Right) 2 CCTV cams placed side by side on a pole, together with a purple air quality indicator

5.2 Traffic Monitoring using Edge-Computing Device

We present the layout of the sensor that will track and identify moving objects including cars, pedestrians, and bicycles.

This study suggests and assesses a sensor that satisfies the criteria. The plan is to set up a network of these sensors so that traffic may be tracked in real-time throughout the entire city. First, we will go over the hardware and how the sensor works. After that, we'll go over the software components that combine a tracking technique with a detection method and explain why we chose them.

5.2.1. Hardware and Functionality

Using video analytics, we developed a sensor that can identify and follow moving objects in a live video feed; this will allow us to keep tabs on network mobility. The sensor's most salient characteristic is its adherence to the edge-computing paradigm; that is, it does video analytics locally and only transmits the processed results. There are two primary benefits to this:

1. Because only indicators and meta-data are communicated, rather than raw photos, the network bandwidth demand is reduced.
2. The device is privacy compliant and transmits only a limited quantity of information.

Before deploying the device in smart cities or using it in the real world, it must be privacy compliant. It is true that the device can work in tandem with preexisting CCTV systems without actually sending the video feed. Since the sensor can make use of the preexisting CCTV infrastructure without requiring any extra cameras, the deployment cost is reduced.

The device has the option of transmitting data over Ethernet or the LoRaWAN network, a low-power network, wireless long-range that is part of the IoT [26]. Duty cycles & Low bandwidth of LoRaWAN device are another argument in favor of edge computing. Two main parts make up the prototype shown in Fig. 2 that are:

1. A powerful and energy-efficient ARM-based embedded computer called an NVIDIA Jetson TX2 that runs Ubuntu 16.04 LTS and accelerates neural network computations for image processing;

2. A Python module called LoPy 4 that manages LoRaWAN communications on the AS923 frequency plan used in Australia. Observe that the module can broadcast on any frequency plan that the LoRaWAN protocol supports.

Fig 2: Inside and outside of smart visual sensor

Table 2 details NVIDIA Jetson TX2 modules' primary technical specs, whereas Table 3 details the Pycom LoPy 4 modules' specs. To show the sensor's status, it is coupled with a DuinoTECH XC-4384 small monochrome OLED module. The gadget gets its juice from a 35 W battery bank. We see the connections of the components of sensors in Fig. 3, which is a simple diagram. Lastly, sensor is housed in an aluminum heat sink shell that is waterproof to an IP67 standard. This shell can disperse the heat that the Jetson TX2 produces, making it ideal for use outside.

Table 2: NVIDIA Jetson TX2 Specification

Features	Details
Power	15 W, 12 V
Thermals	−25 °C to 80 °C
Socket	50 × 87 mm, 400-pin Samtec board-to-board connector
Miscellaneous I/O	CAN, GPIOs, I2S, I2C, SPI, UART
PCie	Gen 2 | 2 × 1 + 1 × 2 or 1 × 4 + 1 × 1
USB	USB 2.0 + USB 3.0
Ethernet	1000/100/10 BASE-T Ethernet
Wireless	Bluetooth 4.1, 802.11a/b/g/n/ac 2×2 867 Mbps
Supported video codecs	VP9, VP8, H.265, H.264
Decoder	(12-Bit Support) 2 K x 4 K 60 Hz Decode
Storage	32 GB eMMC 5.1, SDIO, SATA
Memory	@ 1866 Mhz | 59.7 GB/s with 8 GB 128-bit LPDDR4
GPU	@ 1300 MHz with 256-core Pascal
CPU	NVIDIA Denver2 (dual-core) @ 2 GHz + ARM Cortex-A57 (quad-core) @ 2 GHz

Table 3: Pycom LoPy 4 Specifications

Features	Details
Power	0.35 W, 3.3 V
Thermals	−40 °C to 85 °C
Miscellaneous I/O	PWM, UART, SPI, DAC, ADC, GPIO
Lora and Sigfox connectivity	Semtech SX1276

Features	Details
Wireless	868/915 MHz LoRa and Sigfox, Bluetooth BLE, Wifi 802.11b/g/n 16 Mbps
Memory	External flash: 8 MB, RAM: 4 MB + 520 KB
CPU	Up to 600 DMIPS Xtensa® 32-bit (dual-core) LX6 microprocessor

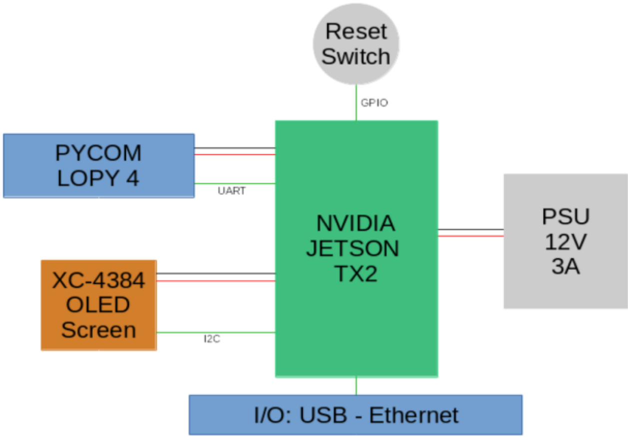

Fig 3: Sensor

Jetson TX2 sensor powers the LoPy 4 and the OLED screen in addition to managing USB, video analytics, Ethernet communication. In turn, power supply unit (PSU) powers the Jetson TX2. The LoPy transmits data via LoRaWAN, and the screen shows the sensor's status in its most simple form.

On average, twenty times per second, the sensor iteratively does the following steps:

1. Capturing images using a webcam or IP camera.

2. Identifying the target objects within the frame.

3. Following the objects by comparing this frame's detections to those from the previous one.

4. Incorporating newly discovered items into the device database or revising the trajectories of things already there.

While these processes are ongoing, the sensor will occasionally send the video processing findings to the IoT Core using LoRaWAN or Ethernet. While the user has some leeway in determining the minimum interval between transmissions, the protocol limitations of LoRaWAN prevent it from being shorter than 5 minutes. So that the 32 GB of local storage doesn't get full, the database is cleared out after every transmission. Fig 4 shows the sensor's activity flow diagram.

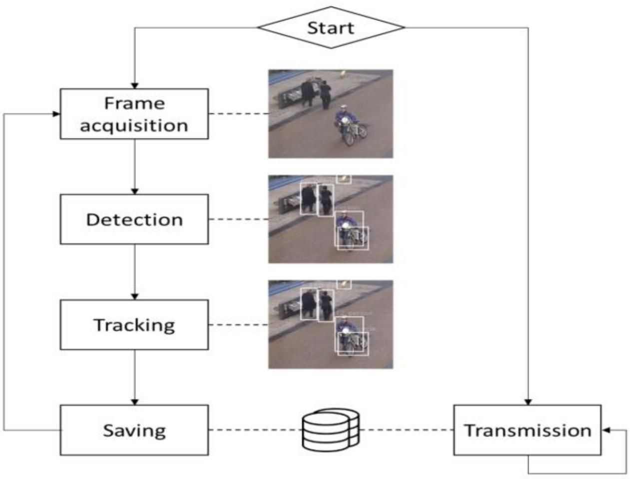

Fig 4: Sensor activity flowchart

Fig. 5 shows the 3U rack version that was developed for use in scenarios where LoRaWAN is unnecessary & device accessing IP camera's video feeds, which is common in CCTV installations. Fifteen separate Jetson TX2 modules, each with the capability to handle a single camera feed, make up this rack unit. The configuration is resilient, meaning that even if one unit in the rack fails, the others can continue to work.

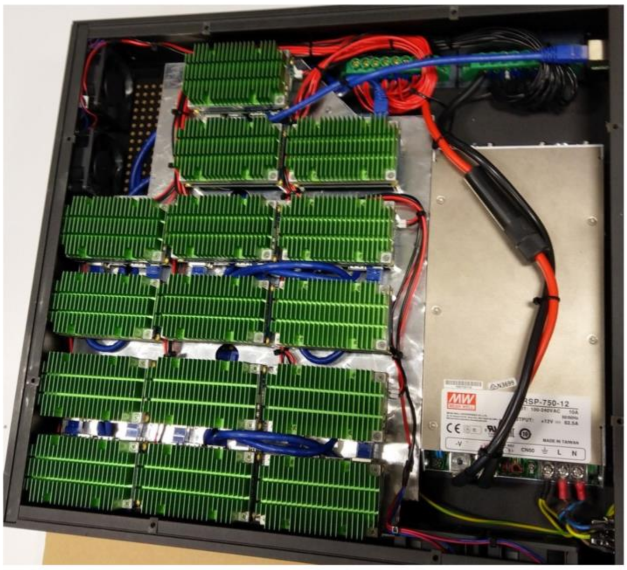

Fig 5: The 3U server vision sensor housing 15 NVIDIA Jetson TX2 modules, each which can handle one live video stream from security camera in real-time

5.2.2. YOLO V3: Detecting Objects

This section describes the parts that follow offer more information regarding the procedures for detection and tracking. These days, there are numerous deep learning-based computer vision algorithms accessible for object detection in images. Choosing an algorithm that can detect in real-time in an embedded system with a high degree of accuracy is crucial when it comes to traffic flow monitoring. Because of these two factors, YOLO V3 [27], a well-liked and cutting-edge object detector built on a fully convolutional deep neural network, is a strong contender. When compared to previous algorithms, YOLO V3 can detect objects at three distinct scales and provides a fair balance between speed and accuracy. Since the size of a moving item is determined by its distance from the camera, this final feature is equally essential in our situation.

In contrast to earlier approaches, the YOLO design uses only the Darknet deep neural network to process an input image once, after scaling it to a certain input size. With 106 hidden layers collected in residual blocks, this network is completely convolutional. It can now identify six distinct object types thanks to extensive training and adaptation:

• pedestrian	• bus
• bicycle	• truck
• car	• motorbike

Each detection scale causes the network to split the image into three grids, with each grid cell predicting K bounding boxes. These are the characteristics of each bounding box B:

- its shape defined by the its centroid coordinates (x,y), its width w and height h
- an object confidence score O; and
- six class probabilities P_i (one for each object type)

'if $O \geq \theta$, then B is associated with the object type o such that $o = \arg\max_i P_i$

If $O < \theta$, then B is discarded in order to remove the bounding boxes with the least confidence score. where θ is a given confidence threshold'

'Finally, to remove duplicate detection of the same object, the Non-Maximal Suppression (NMS) technique is applied. If we have an intersection-over-union (*IoU*) between two bounding boxes B_m and B_m greater than a predefined NMS threshold γ, i.e., if $IoU(B_m, B_n) \geq \gamma$ then the bounding box with the least objectness score is removed'. Fig 6 summarize the workflow detection.

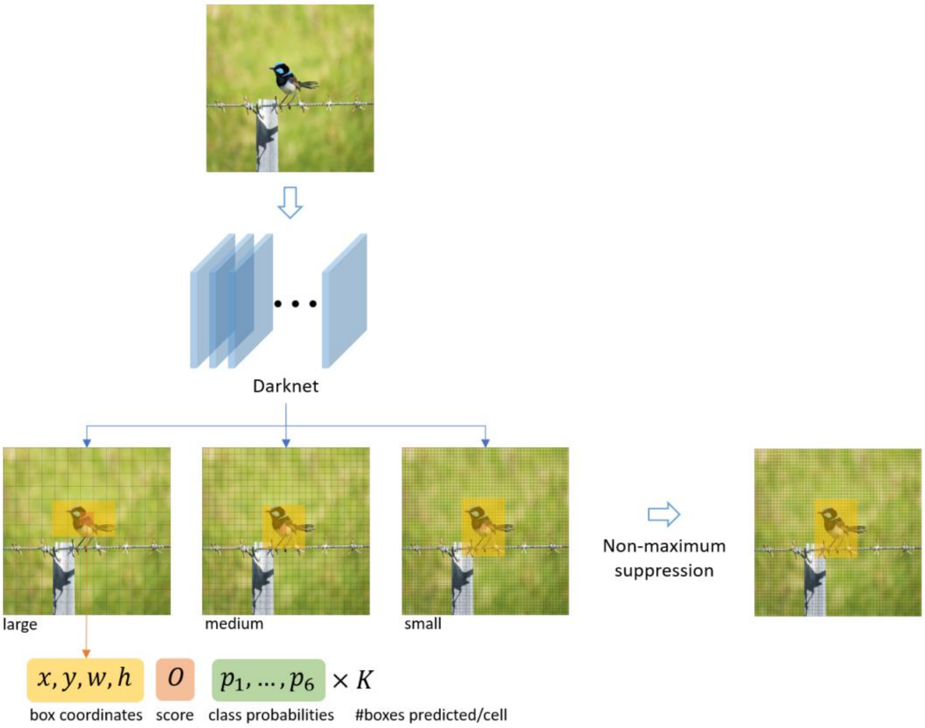

Fig 6: YOLO V3's architecture for object detection. 'A fully convolutional neural network including 406 hidden layers is trained to predict class probabilities and box coordinates at three distinct sizes (big, medium, small). We then apply a non-maximum suppression procedure to keep only the category and coordinates that have the highest score'.

We have taken advantage of the NVIDIA Jetson TX2 GPU's CUDA cores by integrating YOLO V3 into PyTorch 1.1 for our application. Table 3 displays the values of the parameters. Anyone interested in learning more about YOLO V3 and how it works can do so here.

Table 3: Detection task by YOLO V3 parameters

Parameters	Values
Input sizes	416 × 416 pixel
Large scale detection grids	13 × 13 cell
Medium scale detection grids	26 × 26 cell
Small scale detection grids	52 × 52 cell
NMS γ	0.5
Confidence θ	0.9
No. of bounding box per cell K	3

5.2.3. SORT: Tracking Objects

Matching the objects detected in current frames with those from previous frame is following step once the detection process is completed. The 'Simple Online and Real-time Tracking (SORT)' technique, which has been thoroughly documented and benchmarked, is utilized and implemented in Python 3 to carry out this multiple object tracking (MOT) task [28].

One of our application's most important requirements is the ability to use SORT in real-time, hence the development of the algorithm has been heavily focused on efficiency without sacrificing the exceptional tracking performances it is known for. To do this, we combine two approaches that have a reputation for being both accurate and efficient in computing:

• The Hungarian algorithm for an optimal solution to the prediction-bounding-box problem;

• A Kalman filter for estimating bounding box positions in the current frame from past placements.

In Kalman filter, the following is the model of each tracked item state, or tracklet, t:

$$t = [x, y, , r, x, y, a]^T$$

Define the bounding box of an object by 'centroid coordinates', which are x and y; its area and aspect ratio are a and s; and its velocity, which is k˙ with $k \in \{x, y, a\}$ feature

'It is mentioned that a constant aspect ratio is expected. The velocity components are derived by the Kalman filter, and only the geometric components of the bounding box computed by YOLO V3 are used to update the state of t. The initial velocity components in the error covariance matrix that the Kalman filter uses are set to large values to reflect the uncertainty surrounding the "t" initial speed'

In order to associate detections with tracklets, the Kalman filter predicts the bounding box geometry and location for each tracklet in the current frame. The Hungarian algorithm then finds the best match between the predictions and the detections, with the cost matrix determined by the IoU between each pair of detections and predictions. One advantage of employing the IoU-based distance, according to SORT, is that it can handle temporary occlusions [30]. 'Every association for which the IoU is less than a threshold $IoU_{\min}$ is discarded following the assignment stage'.

'In order to alleviate the issue of false positive detection, keep the device's memory from filling up fully, and enhance the tracking of objects that can be obscured for a maximum of $age_{\max}$ frame, two other parameters, $hit_{\min}$ and $age_{\max}$ are also employed'. If a tracklet is not identified for a consecutive $age_{\max}$ frame, it will be destroyed. Tracklets are only saved if they have been viewed for at least in $hit_{\min}$ frames. Version 4-UUID is the unique identifier assigned to each new tracklet. Table 4 lists the tracking parameter values that the visual sensor employed.

Table 4: Tracking task by SORT parameters

Parameters	Values
Threshold $IoU_{\min}$	0.3
Maximum age $age_{\max}$	40
Minimum hits $hit_{\min}$	3

Object of interest will be assigned a new tracklet and id if it later re-enters the camera's range of vision. The architecture for gathering data from the sensor network is presented in the next section.

5.3 Agnosticity

The software framework called "Agnosticity" was created for 'Liverpool project'. The basic principle is leveraging open-source technology & software to their fullest extent possible, without assuming anything about the sensors or communication protocols that are being used. The purpose of Agnosticity is to facilitate the gathering, storing, and retrieval of data from IoT and to make it possible for various technologies to work together through the OneM2M standard [31, 32]. The rapid expansion of the Internet of Things has made interoperability an essential need for any Smart City application [33]. In order to gather data from many sources and offer a unified entry point, the infrastructure level is seeing an increase in the number of sensors, protocols for networks, and use delegates. The open-source oneM2M and SmartM2M standard implementation is the Eclipse OM2M project. Through offering a horizontal M2M service platform for service development independent of the underlying network, it hopes to facilitate the deployment of vertical applications and heterogeneous devices. OM2M provides a RESTful API that includes basic methods for re-targeting, group organization, synchronized and asynchronous communications, resource discovery, container management, application registration, and machine authentication.

The overall design of the Agnosticity framework used in this research is shown in Fig 7. The OM2M platform receives data from 15 fixed optical sensors that are linked to CCTV network by HTTP Post. Twenty air quality sensors and five mobile visual sensors use IoT and LoRaWAN Network to transmit data. By utilizing the MQTT broker, a dedicated plugin retrieves data from The Things Network and re-publishes it to the OM2M platform via HTTP Post. This means that the OM2M platform has immediate access to all data in a designated container. The subscription system automatically saves each one to a special database for the future.

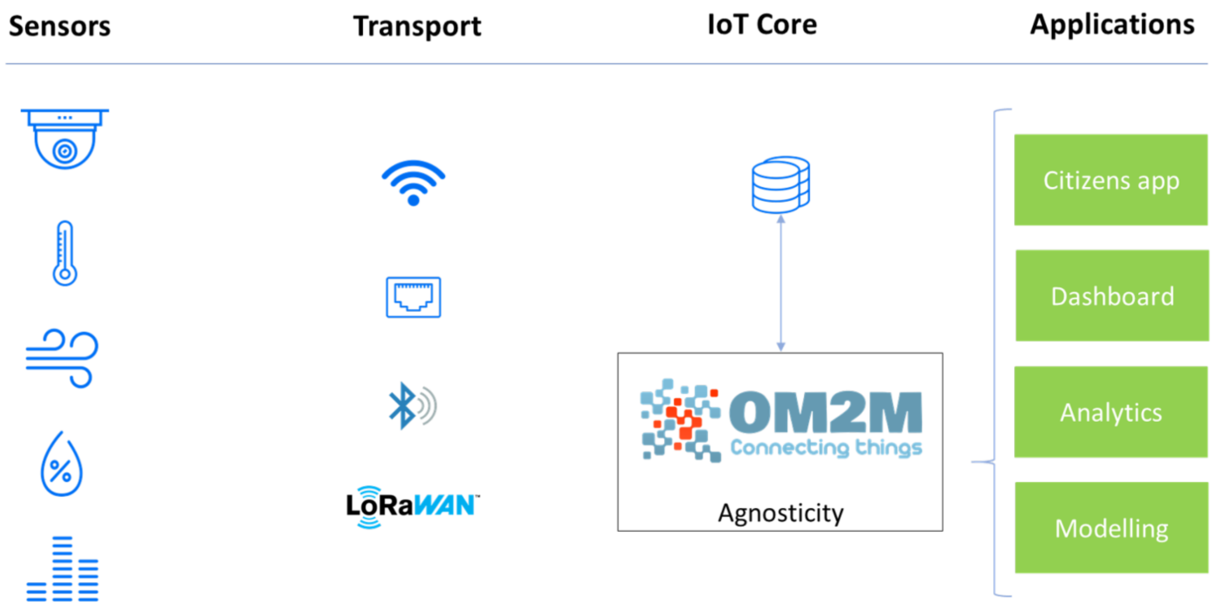

Fig 7: The project's overall design. The open-source software that the Agnosticity software stack is built upon is quite well-established.

'The open-source implementation of the OM2M standard, data collection and access are guaranteed'. We can build several apps on Agnosticity framework top. We can access data directly using the 'OM2M RESTful API' or we can request data from database. As an example, the web-based dashboard depicted in Fig 8 allows for a visual examination of the data gathered from the various sensors.

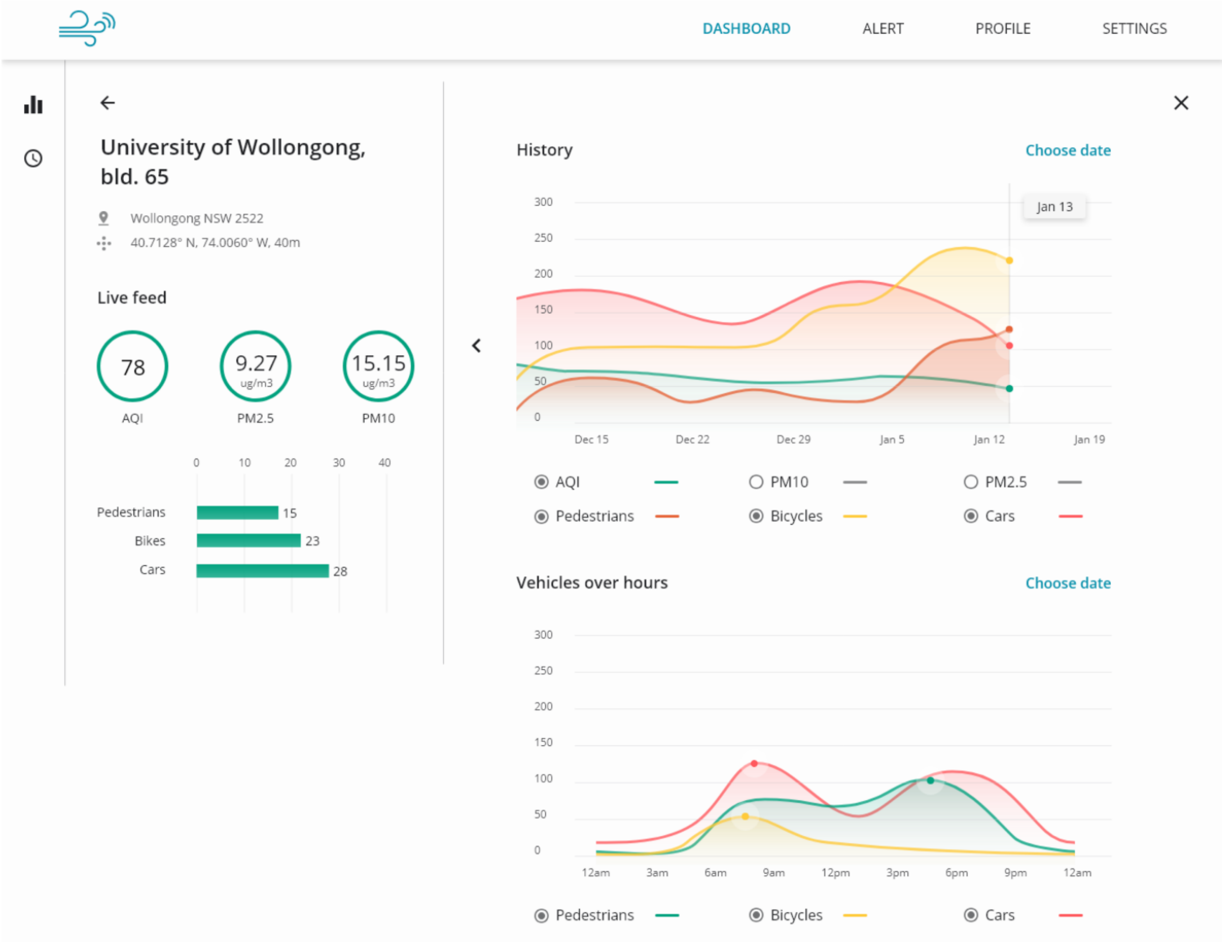

Fig 8: The data obtained from the many sensors used in the Liverpool project is shown in an interactive dashboard that is accessible through the web. You can use it on desktop browsers as well as mobile ones; the interface is responsive.

6. Validation Experiment

We look at the sensor's efficiency, precision, and system utilization to see how well it worked. We began by using a validation dataset culled from the literature to assess the sensor's accuracy and effectiveness in detecting pedestrians. After that, we analyzed the system's and network's usage based on data collected from an actual experiment in which the sensor was linked to a CCTV.

6.1. Performance and Accuracy

Using the Oxford Town Center Dataset, we assessed the sensor's precision and functionality [34]. The movie shows a bustling street in the town center from a CCTV angle, and it's in high quality (1920 × 1080 at 25 fps). Three minutes' worth of footage, or 4,500 frames, served as the basis for the validation experiment. In this particular sequence, the creators of the dataset labeled the video with the positions of 230 pedestrians. So, we compared our sensor's readings to the ground truth to see how well it worked.

Summarized in Table 5 are the performance results for the following variables, together with statistic calculated over the 4500 frames:

The performance outcomes with basic statistics are calculated across the 4500 frames is presented in Table 5. The findings show that the algorithm under-estimates the number of detections while yet maintaining an acceptable frame rate.

Table 5: Performance summary with basic data calculated over 4500 frames. These data show that the algorithm under-estimates the number of detections while yet maintaining an acceptable frame rate

	Detections	True	Error	Relative Error	Accuracy	fps
Mean	10.53	15.88	−5.35	0.32	0.70	19.58
Standard Deviations	2.81	4.70	3.36	0.16	0.16	3.50

	Detections	True	Error	Relative Error	Accuracy	fps
Minimum	2.01	6.01	17.01	0.01	0.23	4.64
25th-percentiles	8.01	13.01	−8.01	0.22	0.58	17.29
Medians	11.01	16.01	−5.01	0.34	0.67	19.78
75th-percentiles	13.01	19.01	−3.01	0.43	0.79	22.23
Maximum	20.01	28.01	2.01	0.78	1.34	22.98

- detections: sensor detecting the number of objects
- true: the ground truth; the amount of object annotated in the dataset
- error: difference between true and detections
- $$relativeerror = \frac{|error|}{true}$$
- $$accuracy = \frac{\det ection}{true}$$
- fps: the number of frames per second (FPS), which is the inverse of the time it takes to process one frame of the video.

We now examine the validation analysis in greater detail. The sensor's early results show that it had a median relative inaccuracy of 35% and an average accuracy of 70%. The relative error and accuracy were both affected by the sensor's tendency to underestimate the number of detections, as indicated by the error. Figure 9 shows the distribution of the relative error and accuracy across all frames.

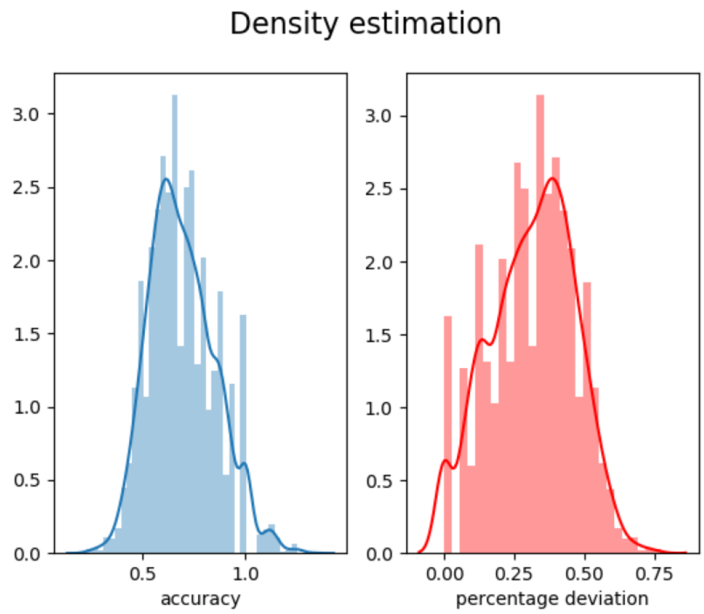

Fig 9: Accuracy (left) and percentage deviation (right) of the kernel density estimation calculated over the Oxford dataset's 4500 frames

The number of FPS and the number of pedestrians detected with time are shown in Fig. 10. The anti-correlation between the two curves is plain to notice. True, FPS performance did seem to drop as the number of detections increased. That was because the SORT tracking method was a bottleneck because it wasn't optimized to use the Jetson TX2's CUDA cores. Applying this tracker on the GPU will solve this limitation in future algorithm iterations.

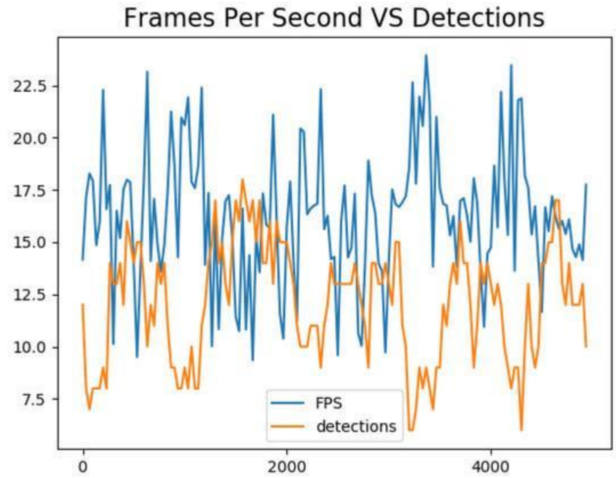

Fig 10: The change over time in FPS processed by the sensor (blue) and no. of objects recognized (red). Its evident that a lesser number of detections results in a higher frame rate. Since there are more items to track, the SORT algorithm is mostly to blame for the FPS decline. This task does not make use of the Jetson TX2's CUDA cores.

We show in Fig. 11 how the sensor's accuracy and the ground truth have changed over a period of more than 4500 frames. The curves show that they were anti-correlated, with accuracy rising with low ground truth detection numbers and falling with big crowd sizes. The occurrence of occlusions in densely populated areas explains this. In fact, the algorithm may only identify one of the two individuals if the other is hiding the other. This is partially caused by the way YOLO V3 behaves, that uses a ‘non-maximum suppression’ mechanism in scenarios when there are several overlapping bounding boxes.

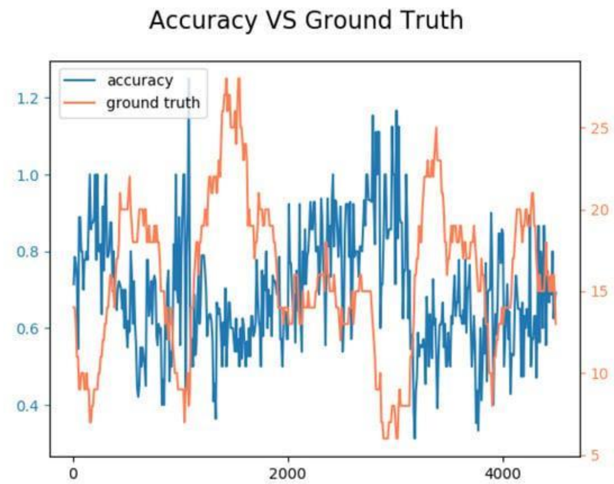

Fig 11: Accuracy's evolution over time with the ground truth (orange line) and the ideal accuracy (blue line). With small groups, accuracy is higher and falls with huge crowds.

‘A scatter plot showing the ratio of ground truth detections to sensor detections is shown in Fig 12. The correlation was clearly linear; that is, the higher the ground truth, the higher the number of objects picked up by the sensor. Since the sensor was underestimating the amount of detections, this result further demonstrates that it was more prone to false negative errors than false positives’. While both under- and overestimation can lead to inaccurate traffic monitoring results, under-estimation is more common and less troublesome. Our algorithm's precision would unquestionably be enhanced by a more comprehensive examination of the error rate. Still, the trends in object detection rates are spot on, leading to this sensor's generally good accuracy and performance.

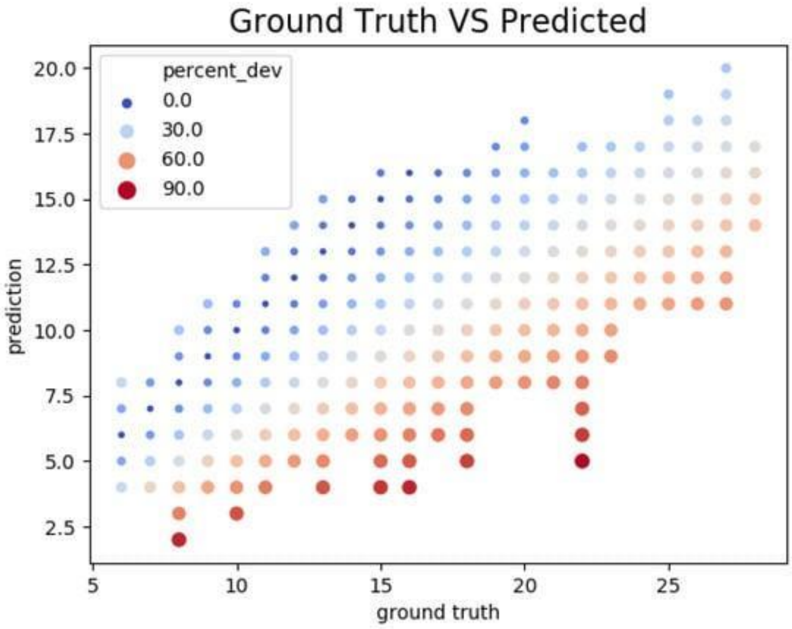

Fig 12: Ratio of ground truth detections to detections. We see that the relationship is linear, indicating that trends are captured by the algorithm.

6.2. Network & System Utilization

As the sensor is deployed in the actual world, Fig. 13 depicts the evolution of average temperatures, disk, memory, GPU, CPU, & network use every 2 seconds. Throughout this 10-minute trial, the sensor was linked to a closed-circuit television system that was watching a building's entry and the adjacent street.

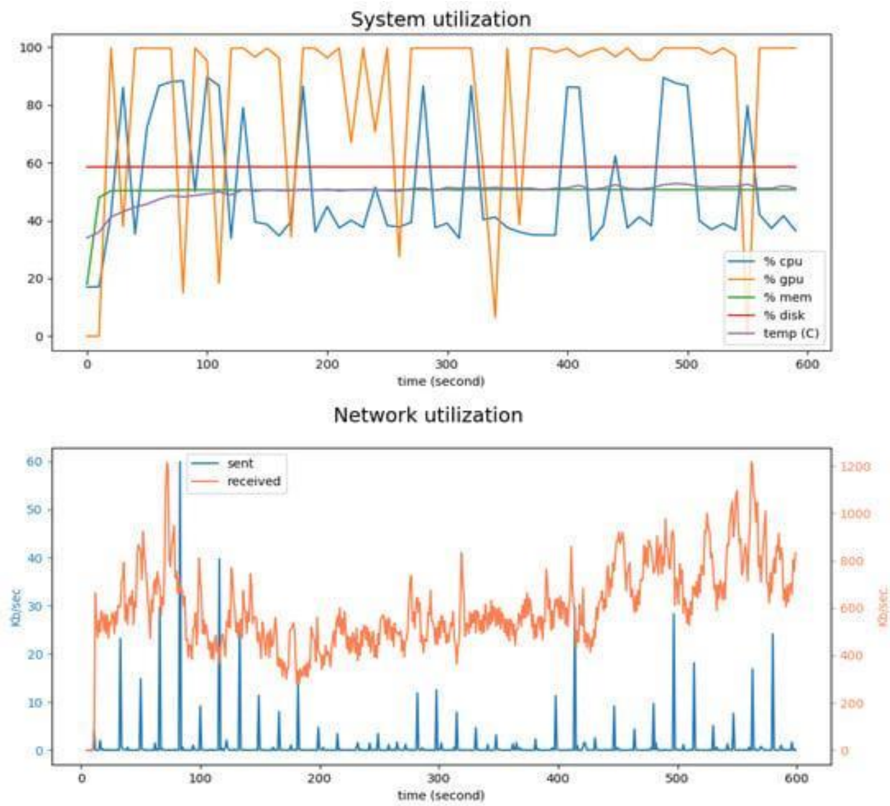

Fig. 13: Real-time installation of sensor during a monitoring in 15-minute of the GPU, CPU, RAM, average temperature, and disk usages (top) & network use (bottom). A total of 280 distinct artifacts have been found over this time.

The ‘top panel’ shows that the system temperature, memory & disk use, and utilization were all stable. Throughout the course of the trial, the GPU's utilization fluctuated between 100% and 40% on occasion. The ‘tracking algorithm's implementation on ‘CPU’, which served as a bottleneck, explains this, as indicated earlier. There was a correlation between spikes in CPU usage and dips in GPU usage.

The outbound & inbound network bandwidth used in each second is shown in the bottom panel. A frame was acquired from the CCTV system, and the incoming data matched that. There was a correspondence between the data transmission and the data posting to the Agnosticity platform. Every minute, there is a noticeable outgoing peak that corresponds to the sensor's transmitting rate. The fact that there was less data sent out than data received is readily apparent. This was due to the fact that the transmission just included meta-data retrieved from the frame like trajectories & counts. That being said, the sensor doesn't necessitate a significantly high bandwidth.

7. Real -time Applications

Two practical uses of the visual sensor are described in this section. The first one was keeping track of how many people were entering and leaving a facility during emergency evacuations for one hour. Outcomes from a week-long traffic monitor exercises in Liverpool were presented in second. The method’s validity in real-life settings was established by those two applications. The closed-circuit television cameras were transmitting a 25-fps full HD video stream in both cases.

7.1. Indoor Installation

The vision sensor was first tested by tracking the movement of people on the ground level of the University of Wollongong's SMART Infrastructure Facility building. Pedestrian movement can be easily monitored from this building, which includes several labs and instructional activities. A stairwell serves as the primary means of entrance to the first level, and the camera was positioned in front of it.

From 14:30 to 15:30 on a typical workday, the experiment tracked the number of persons detected by the sensor (Fig. 14). Inside a structure, it is hardly surprising that no vehicle or bicycle was identified. It was also possible to see two peaks with no detection in the middle. The fact that fire alarm go off while the experiment is underway provides an explanation. The writers of this research did not intentionally set off this false alarm in order to collect more data, so that's a relief. Curiously, the facility was essentially evacuated because none is found during fire alarm events. Also, 2nd peak is low than 1st, which means that few persons went back when the firefighters said it was okay to do so. It appears from these preliminary results that this sensor has the potential to identify suspicious crowd movements.

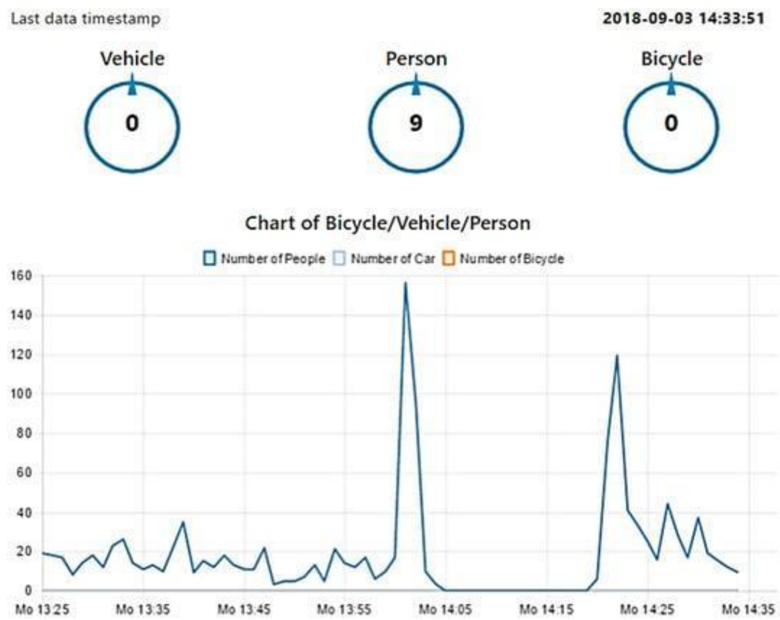

Fig 14: The number of people found within a structure over the course of an hour.

In Fig 14, the beginning and ending of the fire alarm occurrence are represented by the two peaks. In this one-hour experiment, 631 individuals were found and followed. The generated trajectories are displayed in Fig. 15, superimposed on the sensor's actual field of view. The trajectories are evidently in line with expectations; there are, for example, no trajectories between floors. Apart from the trajectories, the gathered data could also yield other intriguing insights, such a ‘heat map like the one’ shown in Fig 16 that illustrates highest number of detections within the sensor's field of view throughout the experiment. The longest or shortest amount of times spent at a particular spot, the speeds at which the boxes are identified, or the automatic recognition of unusual activities like crossing lines without permission are examples of additional metrics.

Fig 15: The paths taken by the people identified and tracked by the sensor. The 631 lines each correspond to a single individual

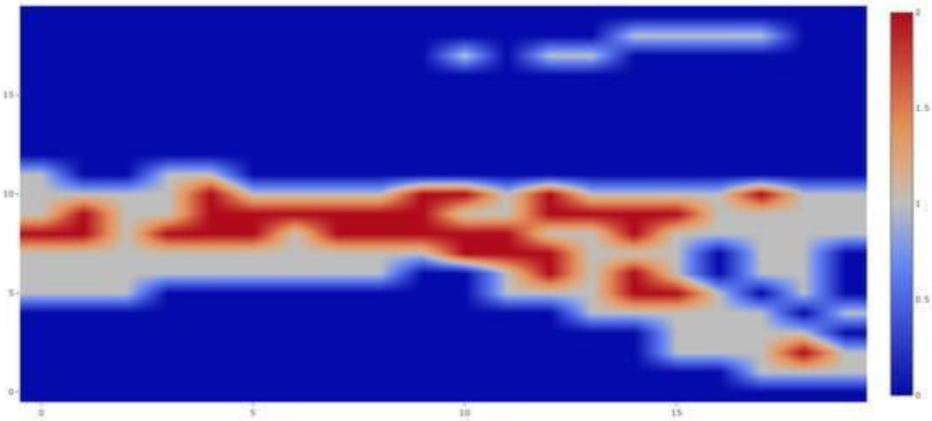

Fig 16: A heat map displaying the highest number of individuals identified inside the sensor's area of vision

7.2. Liverpool: Outdoor Installation

After the first round of testing in a controlled setting, twenty vision sensors were placed across the heart of Liverpool (Fig 1). The camera, whose position is shown in Fig 17, is the primary focus of this application. The abundance of nearby restaurants and businesses means that this spot is right close to a street that sees a lot of foot traffic. There are three crosswalks visible from where the camera is positioned.

Fig 17: The sensor is situated near a pedestrian street in the city's center (highlighted in red). The blue line represents the camera's field of vision

From 20 to 27 February 2023, a week's worth of counting results are displayed in Fig. 18. The chart shows the amount of people, cars, and bikes spotted every minute over the course of eight days. It is clear from this graph that the city's circadian rhythm is at its most active between the hours of 08:00 and 16:00. Figure 19, which shows the number of hourly detections, makes this point clearer.

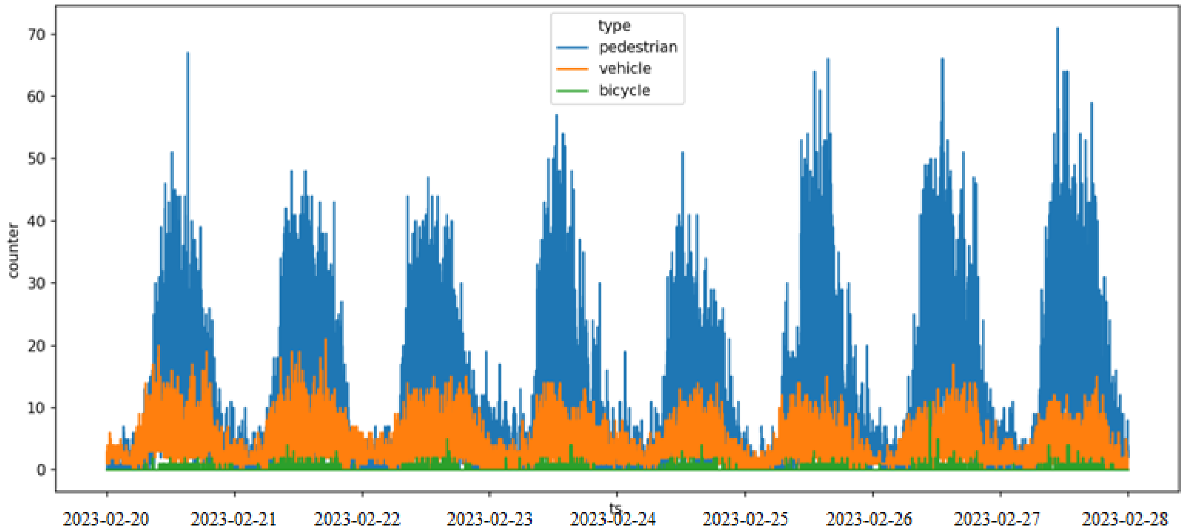

Fig. 18: The number of bicycles, cars, and pedestrians the sensor saw between February 20, 2023, and February 27, 2023. Every data point shows how many objects of a particular type were found during the previous minute

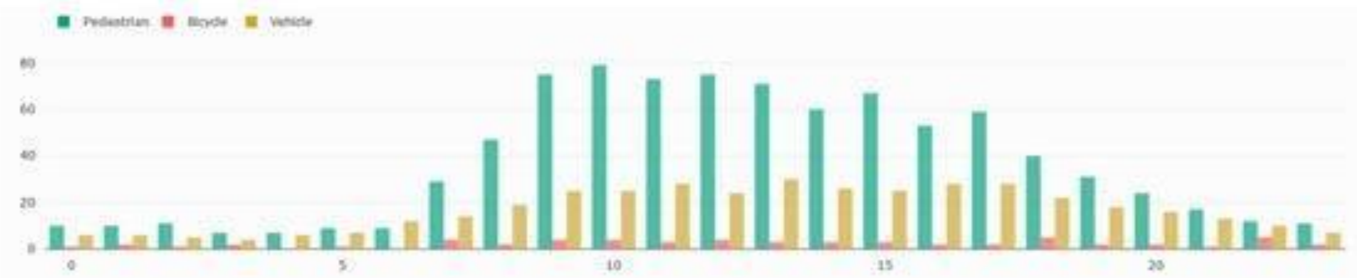

Fig 19: The number of walkers (green), bicycles (red), and cars (yellow) that were observed on February 23, 2023, hourly

Compared to the previous three days, there appeared to be more pedestrian activity on these days. Day 5, or February 24, 2023, was a Sunday and had the least amount of pedestrian activity. The busiest time of day was usually about noon. ‘The vehicle graph (shown in orange) shows two daily activity peaks, the first occurring around 9:00 and the second around 16:00. Although the outcomes appear to be in line with what one could anticipate from an intersection of that kind, urban planners must do a thorough analysis. However, it does demonstrate the sensor's ability to identify changes in the daily circadian cycle of traffic flows’.

The coordinates of all bicycle and pedestrian detections made on February 23, 2023, are shown in Fig. 20. In these 20,399 distinct things were found that day. Every blue dot indicates the presence of a pedestrians, whereas every orange dot indicates presence of a bicycles. Although a single dot represents a single detection, in most cases, a single person was picked up more than once while they were within the camera's field of vision. ‘Examining the spatial distribution of the detection over the frame is made possible by this display. It permits taking into account the pedestrian flows that cross the road at a crosswalk in the current context (corresponding to the top and bottom of the graph)’. It was seen that several of the pedestrians were not inside the crosswalk. Bicycles were detected, indicating a combination of bicycle and pedestrian movement. On the other hand, very few bicycles were seen, which is consistent with the opinions that residents voiced at the community workshop.

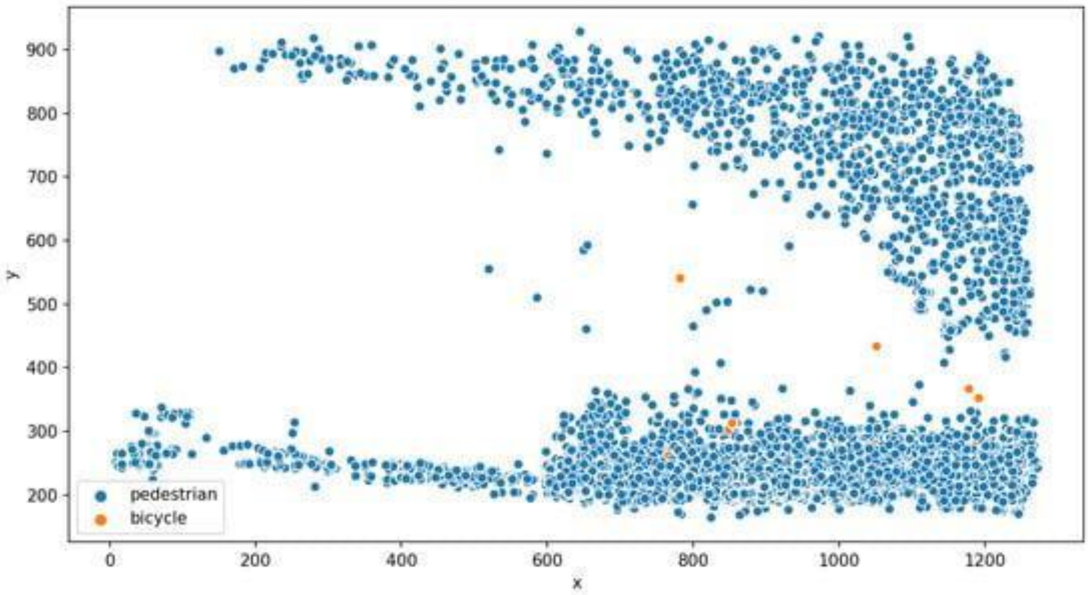

Fig 20: The pixel coordinates (X,Y) for the identified bicycles (orange) and walkers (blue) on February 23, 2023, within the frame. Bounding box centroid associated with an item identified at those coordinates is represented by each dot

Fig. 21 displays the resulting paths that the cyclists and pedestrians used. It appears that there are two distinct flows: 'one from the bottom left to the bottom right, and the other from the top left to the middle right'. The mobility patterns at this crossing are shown by these two flows.

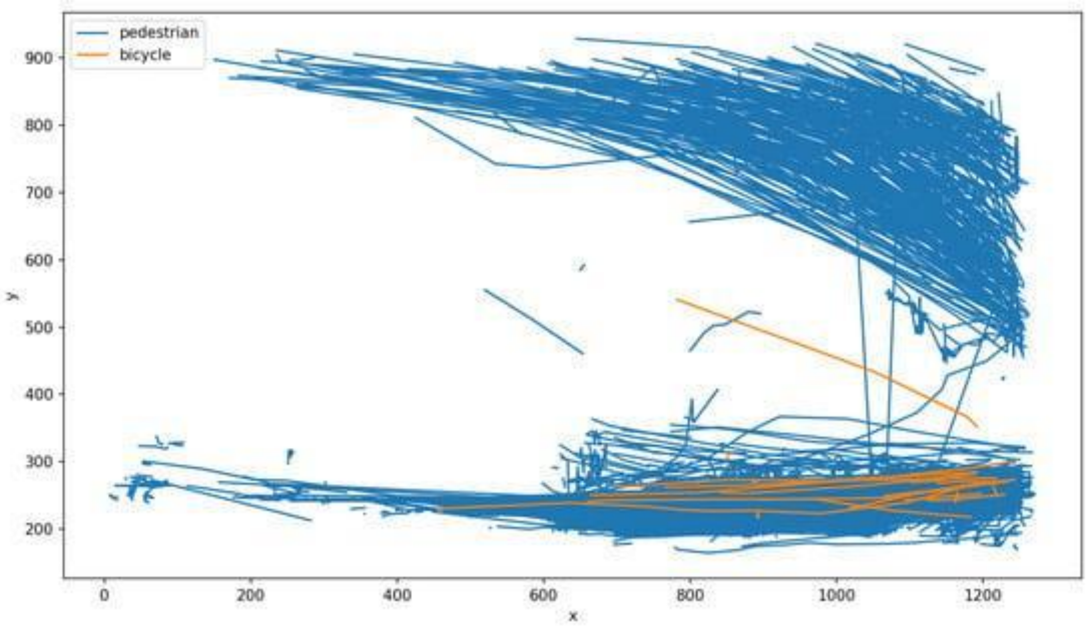

Fig 21: Wheelchair and pedestrian trajectories inside the frame on February 23, 2023. There are two distinct pedestrian flows

The various outcomes displayed in this section are meant to draw attention to the information provided to urban planners. Urban planners are presently analyzing those data in greater detail.

8. Conclusion

This study introduces a novel edge computing visual sensor for an Internet of Things system, employing the Agnosticity framework to track the movement of cars, bikes, and pedestrians. The sensor, built on the NVIDIA Jetson TX2 platform, combines the SORT real-time tracking technique with YOLO V3 for object detection, transmitting privacy-compliant metadata over Ethernet or LoRaWAN. Deployed in Liverpool, Australia, the system, comprising 20 sensors, enables real-time traffic monitoring. Future work will focus on enhancing tracking and detection algorithms, exploring the NVIDIA Xavier platform for improved performance, and adapting YOLO V3 to frameworks like Caffe and Tensorflow for increased efficiency. The study will investigate optimizing the SORT method for GPU processing in the future.

References

[1] Tuvikene, Tauri. "Between traffic and walking discourse: Pedestrians in the traffic machine, hints from the Estonian case." *The Journal of Transport History* (2023): 00225266231179557.

[2] Yannis, George, and Antonis Chaziris. "Transport system and infrastructure." *Transportation research procedia* 60 (2022): 6-11.

[3] Ouallane, Asma Ait, Assia Bakali, Ayoub Bahnasse, Said Broumi, and Mohamed Talea. "Fusion of engineering insights and emerging trends: Intelligent urban traffic management system." *Information Fusion* (2022).

[4] Napolitano, Enea Vincenzo. "Intelligent technologies for urban progress: exploring the role of ai and advanced telecommunications in smart city evolution." In *European Conference on Advances in Databases and Information Systems*, pp. 676-683. Cham: Springer Nature Switzerland, 2023.

[5] Pompigna, Andrea, and Raffaele Mauro. "Smart roads: A state of the art of highways innovations in the Smart Age." *Engineering Science and Technology, an International Journal* 25 (2022): 100986.

[6] Hanif, Arissa Nalia Binti, Safaa Najah Saud Al-Humairi, and R. Junaidi Daud. "IoT-based: Design an autonomous bus with QR code communication system." In *2021 IEEE International Conference on Automatic Control & Intelligent Systems (I2CACIS)*, pp. 225-230. IEEE, 2021.

[7] Yue, Wenwei, Changle Li, Guoqiang Mao, Nan Cheng, and Di Zhou. "Evolution of road traffic congestion control: A survey from perspective of sensing, communication, and computation." *China Communications* 18, no. 12 (2021): 151-177.

[8] Singh, Gurjeet. "Leveraging ChatGPT for Real-Time Decision-Making in Autonomous Systems." *Eduzone: International Peer Reviewed/Refereed Multidisciplinary Journal* 12, no. 2 (2023): 101-106.

[9] Ali, Akbar, Nasir Ayub, Muhammad Shiraz, Niamat Ullah, Abdullah Gani, and Muhammad Ahsan Qureshi. "Traffic efficiency models for urban traffic management using mobile crowd sensing: A survey." *Sustainability* 13, no. 23 (2021): 13068.

[10] Tomar, Ishu, Indu Sreedevi, and Neeta Pandey. "State-of-Art review of traffic light synchronization for intelligent vehicles: current status, challenges, and emerging trends." *Electronics* 11, no. 3 (2022): 465.

[11] Wan, Xiangpeng, Michael C. Lucic, Hakim Ghazzai, and Yehia Massoud. "Empowering real-time traffic reporting systems with nlp-processed social media data." *IEEE Open Journal of Intelligent Transportation Systems* 1 (2020): 159-175.

[12] WOLNIAK, Radosław, and Wies GREBSKI. "THE USAGE OF SMARTPHONE APPLICATIONS IN SMART CITY DEVELOPMENT–URBAN MOBILITY AND TRAFFIC MANAGEMENT." *Scientific Papers of Silesian University of Technology. Organization & Management/Zeszyty Naukowe Politechniki Slaskiej. Seria Organizacji i Zarzadzanie* 179 (2023).

[13] Porru, Simone, Francesco Edoardo Misso, Filippo Eros Pani, and Cino Repetto. "Smart mobility and public transport: Opportunities and challenges in rural and urban areas." *Journal of traffic and transportation engineering (English edition)* 7, no. 1 (2020): 88-97.

[14] Paiva, Sara, Mohd Abdul Ahad, Gautami Tripathi, Noushaba Feroz, and Gabriella Casalino. "Enabling technologies for urban smart mobility: Recent trends, opportunities and challenges." *Sensors* 21, no. 6 (2021): 2143.

[15] Martins, Sofia, António Costa, Zafeiris Kokkinogenis, and Rosaldo JF Rossetti. "Enabling citizen-centric ITS services through blockchain and human incentives." In *International Conference on Intelligent Transport Systems*, pp. 85-94. Cham: Springer International Publishing, 2021.

[16] Mohmmad, Sallauddin, Mohammed Ali Shaik, K. Mahender, Ranganath Kanakam, and Bonthala Prabhanjan Yadav. "Average Response Time (ART): Real-Time Traffic Management in VFC Enabled Smart Cities." In *IOP Conference Series: Materials Science and Engineering*, vol. 981, no. 2, p. 022054. IOP Publishing, 2020.

[17] Atta, Ayesha, Sagheer Abbas, M. Adnan Khan, Gulzar Ahmed, and Umer Farooq. "An adaptive approach: Smart traffic congestion control system." *Journal of King Saud University-Computer and Information Sciences* 32, no. 9 (2020): 1012-1019.

[18] Ma, Jing, Yinhua Tao, Mei-Po Kwan, and Yanwei Chai. "Assessing mobility-based real-time air pollution exposure in space and time using smart sensors and GPS trajectories in Beijing." In *Smart Spaces and Places*, pp. 102-116. Routledge, 2021.

[19] Shengdong, Mu, Xiong Zhengxian, and Tian Yixiang. "Intelligent traffic control system based on cloud computing and big data mining." *IEEE Transactions on Industrial Informatics* 15, no. 12 (2019): 6583-6592.

[20] Babar, Muhammad, and Fahim Arif. "Real-time data processing scheme using big data analytics in internet of things based smart transportation environment." *Journal of Ambient Intelligence and Humanized Computing* 10 (2019): 4167-4177.

[21] Yang, Jiachen, Yurong Han, Yafang Wang, Bin Jiang, Zhihan Lv, and Houbing Song. "Optimization of real-time traffic network assignment based on IoT data using DBN and clustering model in smart city." *Future Generation Computer Systems* 108 (2020): 976-986.

[22] Yu, James Jian Qiao, and Jiatao Gu. "Real-time traffic speed estimation with graph convolutional generative autoencoder." *IEEE Transactions on Intelligent Transportation Systems* 20, no. 10 (2019): 3940-3951.

[23] Yu, Keping, Long Lin, Mamoun Alazab, Liang Tan, and Bo Gu. "Deep learning-based traffic safety solution for a mixture of autonomous and manual vehicles in a 5G-enabled intelligent transportation system." *IEEE transactions on intelligent transportation systems* 22, no. 7 (2020): 4337-4347.

[24] Chen, Yi-Ting, Edward W. Sun, Ming-Feng Chang, and Yi-Bing Lin. "Pragmatic real-time logistics management with traffic IoT infrastructure: Big data predictive analytics of freight travel time for Logistics 4.0." *International Journal of Production Economics* 238 (2021): 108157.

[25] Sarrab, Mohammed, Supriya Pulparambil, Naoufel Kraiem, and Mohammed Al-Badawi. "Real-time traffic monitoring systems based on magnetic sensor integration." In *Smart City and Informatization: 7th International Conference, iSCI 2019, Guangzhou, China, November 12–15, 2019, Proceedings 7*, pp. 447-460. Springer Singapore, 2019.

[26] Ruz-Nieto, Andres, Esteban Egea-Lopez, Jose-Marıa Molina-Garcıa-Pardo, and Jose Santa. "A 3D simulation framework with ray-tracing propagation for LoRaWAN communication." *Internet of Things* 24 (2023): 100964.

[27] Kim, J. L. "Smart Mountain: A Solution Based on a Low-Cost Embedded System to Detect Urban Traffic in Natural Parks." (2023): 132.

[28] Wang, Li, Xinyu Zhang, Wenyuan Qin, Xiaoyu Li, Jinghan Gao, Lei Yang, Zhiwei Li et al. "Camo-mot: Combined appearance-motion optimization for 3d multi-object tracking with camera-lidar fusion." *IEEE Transactions on Intelligent Transportation Systems* (2023).

[29] Meimetis, Dimitrios, Ioannis Daramouskas, Isidoros Perikos, and Ioannis Hatzilygeroudis. "Real-time multiple object tracking using deep learning methods." *Neural Computing and Applications* 35, no. 1 (2023): 89-118.

[30] Qin, Zheng, Sanping Zhou, Le Wang, Jinghai Duan, Gang Hua, and Wei Tang. "MotionTrack: Learning Robust Short-term and Long-term Motions for Multi-Object Tracking." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17939-17948. 2023.

[31] Tsampoulatidis, Ioannis, Nicos Komninos, Evangelos Syrmos, and Dimitrios Bechtsis. "Universality and Interoperability across Smart City Ecosystems." In *International Conference on Human-Computer Interaction*, pp. 218-230. Cham: Springer International Publishing, 2022.

[32] Mishra, Ayaskanta, Anita Swain, Arun Kumar Ray, and Raed M. Shubair. "HetNet/M2M/D2D communication in 5G technologies." In *5G IoT and Edge Computing for Smart Healthcare*, pp. 45-87. Academic Press, 2022.

[33] Antonios, Pliatsios, Kotis Konstantinos, and Goumopoulos Christos. "A systematic review on semantic interoperability in the IoE-enabled smart cities." *Internet of Things* (2023): 100754.

[34] https://www.kaggle.com/datasets/almightyj/oxford-town-centre