Diyala Journal of Engineering Sciences

Journal homepage: https://djes.info/index.php/djes

ISSN: 1999-8716 (Print); 2616-6909 (Online)

Deploying Android-Based Smart RSUs with YOLOv8 and SAHI for Enhanced Traffic Management

Mohammed F. Rashad*, Qutaiba I. Ali

Department of Computer Engineering, University of Mosul, Mosul, Iraq

ARTICLE INFO

Article history:
Received November 3, 2024
Revised January 23, 2025
Accepted January 30, 2025
Available online March 1, 2025

Keywords:
YOLOv8
SAHI Algorithm
Traffic Monitoring
Real-Time Data Processing
Vehicle Detection

ABSTRACT

Traffic congestion remains a major challenge in urban areas due to the high cost, scalability issues, and inefficiencies of traditional monitoring systems. This study proposes an innovative, cost-effective traffic monitoring system utilizing Android-based Smart Roadside Units (RSUs) to detect vehicles and analyze real-time traffic data. The system leverages the You Only Look Once, version 8 (YOLOv8) model, enhanced with the Slicing Aided Hyper Inference (SAHI) algorithm to improve detection accuracy for small and distant objects. Field experiments were conducted using three Android device categories high, medium, and low-cost to assess detection accuracy across different distances. Results indicated that high-cost devices could accurately detect vehicles up to 500 meters away, whereas medium and low-cost devices exhibited reduced detection accuracy and range. The findings highlight the impact of hardware specifications and environmental conditions on system performance. The proposed approach addresses limitations of conventional traffic monitoring by providing an adaptable, open-source infrastructure that reduces hardware costs while ensuring real-time processing. Utilizing mobile devices enhances scalability and cost-effectiveness compared to traditional RSUs, which are expensive and hard to deploy at scale. Future research will integrate functionalities like pedestrian detection and vehicle tracking to further enhance smart transportation systems. This study demonstrates the feasibility of Android-based RSUs, offering a practical alternative to conventional methods and advancing intelligent traffic management solutions.

1. Introduction

With urban expansion and an increasing population, traffic congestion has become a significant challenge that requires effective solutions. Traffic control is essential to reduce the negative impacts on the economy, the environment, and the quality of life. Urban populations are projected to increase from 54% in 2014 to 66% by 2050 [1,2], leading to higher traffic volumes and exacerbating congestion. Traditional methods of expanding road networks are often unsustainable due to land constraints and high costs while improving traffic control systems provides more feasible solutions for managing the increased demand for transportation services [2,3]. Traffic congestion affects various aspects of urban life, including:

1. Economy: Longer travel times reduce productivity, increase costs for businesses and consumers, and affect the quality of life [4,5].
2. Environment: Emissions from vehicles contribute to air pollution and climate change [3,6,7].

* Corresponding author.
E-mail address: eng.mohammed.fadhil.r@gmail.com
DOI: 10.24237/djes.2025.18104

3. Quality of Life: It affects daily life by increasing stress levels among commuters, reducing the time available for personal activities, and contributing to a general decline in urban livability. and life satisfaction in urban areas [3,7]. Figure 1 illustrates the Traffic congestion effect.

To overcome the challenges of traffic congestion, Intelligent Traffic Management Systems (ITMS) have emerged as an advanced solution that relies on artificial intelligence, sensor networks, and data analysis to improve traffic flow and enhance safety. The advantages of these systems include real-time traffic monitoring [8], predictive analytics [9], adaptive signal control [10], vehicle-to-infrastructure communication [8], accident reduction [11], emergency response optimization [12] and sustainability [13]. Figure 2 illustrates the key features of ITMS.

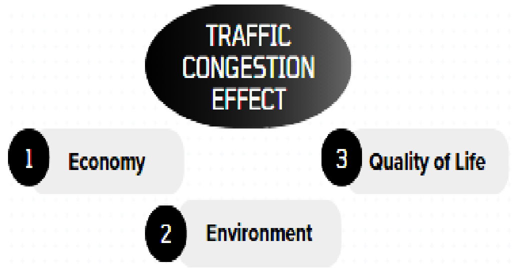


Figure 1. Illustrates traffic congestion effect.

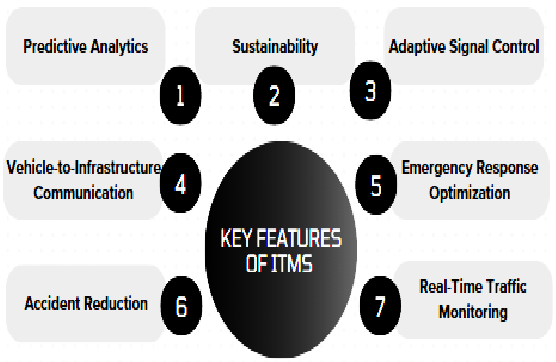


Figure 2. Illustrates key features of ITMS

Traditional traffic monitoring systems rely on RSU units but face challenges related to high costs, difficulty in scaling, and delays in data processing. These systems include technologies such as inductive loops, camera-based systems, Radio Frequency Identification (RFID) sensors, and microwave radar. Despite their effectiveness in specific applications, high costs and scalability issues pose significant barriers. The challenges include:

1. High deployment costs.
2. Difficulty in scaling in large urban areas.
3. Maintenance complexity.
4. Delays in data processing due to network congestion.

YOLOv8, an advanced object detection model developed by Ultralytics, represents a significant evolution in the YOLO series. Designed for real-time applications, YOLOv8 combines high accuracy and efficiency, making it suitable for diverse tasks such as autonomous vehicles, surveillance, and robotics [14-16].

Key features of YOLOv8 include real-time object detection, continuous improvement across versions, resource efficiency, and an open-source design under the AGPL-3.0 license [17-19].

YOLOv8 was chosen for its superior performance, combining high accuracy with fast processing speeds, making it ideal for tasks requiring real-time object detection. Its architectural improvements further enhance its efficiency and scalability [16].

The efficiency of YOLO-based models has been demonstrated in various domains. For instance, Yousif et al. [20] utilized YOLOv7 and Video Swin Transformer for assisting visually impaired individuals by providing real-time video descriptions. Our study builds upon this foundation, employing YOLOv8 and SAHI for enhanced traffic monitoring.

This work makes several key contributions to the field of traffic monitoring systems:

1. Innovative Traffic Monitoring System:

The system proposes to integrate advanced AI technologies such as YOLOv8 with Android devices at different cost levels (low, medium, high) to achieve an optimal balance between accuracy and economic efficiency.

The system enables road monitoring, vehicle detection, and congestion identification with high accuracy using affordable and available devices.

2. Reducing the cost of high-performance devices:

Using high-performance devices that provide detection accuracy of up to 85% for a distance of 500 meters, at an economical cost ($250 per unit), is a qualitative leap compared to traditional systems that cost between $1,500 and $17,680[21].

3. Open-source infrastructure:

The system provides an open-source architecture that allows future development and the addition of new functions, such as pedestrian monitoring, vehicle plate recognition, vehicle tracking, and speed measurement.

4. New framework for RSUs:

Development of an Android-based RSU framework to provide efficient traffic monitoring.

5. Experimental performance analysis:

Experimental performance analysis across different devices with various specifications to evaluate performance under different operating conditions.

6. Discuss challenges and adaptability:

Discuss the system's ability to adapt to different environments, highlighting environmental challenges such as lighting and natural interference.In this paper, we cover the following sections: Section 2: Related works, Section 3: Overview of the proposed system, Section 4: Methodology, Section 5: Results, Section 6: Discussion of the results, and Section 7: Conclusions.

2. Related works

To enhance RSU-based traffic monitoring, numerous strategies have been explored with a focus on cost-effectiveness and improved data collection. One approach is the use of inexpensive sensors, which are simpler and more affordable to implement in urban settings compared to traditional equipment. For example, a system leveraging Wi-Fi signal variations achieved a vehicle classification accuracy between 83% and 100% [22], while another employed embedded neural networks to classify vehicle types and speeds with 96% accuracy [23]. Mobile devices have also been proposed for low-cost traffic data collection, using error correction algorithms to enhance reliability [24]. Additionally, the Internet of Things (IoT) and fog computing have been integrated into low-cost monitoring systems, recording vehicle locations via Global Positioning System (GPS) to analyze traffic patterns [25]. Other innovative solutions, such as the Sense Magnetometer (SenseMag) [26], Wireless Traffic Monitoring System (WiTraffic) [27], and edge computing with Long Range Wide Area Network (LoRaWAN) [28] and Bluetooth Low Energy (BLE) [29], further improve vehicle detection and classification accuracy while remaining cost-efficient. However, low-cost systems may encounter limitations in accuracy and reliability due to environmental factors.

Mobile sensing offers another promising solution for enhancing RSU functionality by using vehicle-mounted sensors for broader coverage, thereby reducing the need for a dense RSU network. This approach leverages existing vehicles to reduce infrastructure costs. Examples include data collection via bus fleets [30], vehicle-to-vehicle (V2V) communication protocols for direct vehicle data exchange [31], and cooperative multi-agent systems with edge computing to estimate traffic density [32] and on-demand mobile sensing frameworks that utilize vehicle owners' devices for road condition monitoring [33]. Nonetheless, mobile sensing may face data reliability issues in low-density areas.

Cloud-based RSU systems centralize data processing, lowering the computational demands on individual RSUs and enabling scalability for expanding traffic requirements. Cloud-assisted mobile crowd-sensing has been applied to traffic data collection, which improves congestion estimates and supports proactive driver guidance [34]. However, such systems require stable network connectivity, and data security remains a concern.

Finally, fog computing and green technologies have been investigated to reduce latency and enhance real-time processing in traffic systems. Fog-based Vehicular Ad Hoc Network (VANET) infrastructures facilitate V2V and vehicle-to-RSU communication while decreasing energy usage [35]. Solar-powered smart camera-RSU platforms provide a low-cost, energy-efficient monitoring solution, supporting continuous operation even in changing weather conditions [36].

Although significant progress has been made in previous research, key gaps remain, such as:

• The lack of affordable solutions that match the performance of traditional RSUs.

• Limited scalability for large urban areas.

• Challenges in real-time data processing.

Furthermore, current road monitoring and traffic control systems heavily rely on expensive hardware and wireless networks such as Ad-hoc. These systems face significant challenges in terms of range and efficiency. They often require advanced infrastructure and fail to provide effective real-time monitoring, leading to increased traffic accidents and congestion in urban areas. Additionally, few studies have explored the use of commercially available mobile devices as RSUs, which could greatly reduce costs and enhance flexibility.

This paper aims to develop an innovative Android-based smart RSU system to provide a scalable, cost-effective solution for traffic monitoring. The system will be designed using mobile devices from various cost categories (high, medium, low) and integrate advanced object detection algorithms like YOLOv8 and SAHI to enhance detection accuracy, especially for small or distant objects.

The paper proposes an alternative to traditional RSUs, demonstrating how Android-based devices can process real-time data over 4G networks, offering continuous traffic updates to users. Existing traffic monitoring systems exhibit various strengths and weaknesses, as summarized in Table 1. The table highlights the unique advantages of the proposed Android-based RSU system, such as improved scalability, cost-efficiency, and adaptability, compared to traditional techniques like IoT with fog computing or inexpensive sensors.

Table 1: Comparison of existing traffic monitoring techniques with the proposed system

Technique	Advantages	Disadvantages	Comparison with Proposed System	References
Inexpensive Sensors	Low cost; suitable for urban settings; high classification accuracy (83%-100% using Wi-Fi variations).	Limited reliability in environmental fluctuations.	Less flexible and scalable than the Android-based system proposed, which allows dynamic adjustments for varied devices.	[22]
Embedded Neural Networks	High accuracy (96% for type and speed classification).	Computationally intensive; hardware-dependent.	Similar computational demands, but YOLOv8 with SAHI offers better scalability across hardware tiers.	[23]
IoT with Fog Computing	Low-cost monitoring; GPS-based traffic analysis.	Latency issues; limited real-time capabilities.	Proposed system enhances real-time data with 4G networks and Android-based versatility.	[25]
SenseMag (Magnetic Sensing)	Non-invasive; cost-efficient for vehicle detection.	Limited to specific environments and magnetic fluctuations.	Android-based approach integrates multiple detection methods for broader adaptability.	[26]
WiTraffic (WiFi-based)	Low-cost; non-intrusive.	Requires WiFi infrastructure; lower detection range.	Proposed system offers extended ranges with camera-based detection.	[27]
Mobile Sensing (Vehicle-Mounted)	Broader coverage using existing vehicles; reduces need for dense RSU deployment.	Data reliability issues in low-density areas.	Android-based RSUs can be strategically placed, ensuring consistent coverage even in low-density zones.	[30], [31], [32], [33]
Cloud-Based RSUs	Centralized processing reduces RSU computational demands; scalable for expanding urban needs.	Requires stable network connectivity; data security concerns.	Proposed system focuses on local processing to reduce dependency on continuous cloud connectivity.	[34]
Fog Computing with Green Tech	Low latency; energy-efficient with solar-powered smart RSUs.	Dependence on environmental conditions for energy generation.	Android-based system is power-flexible and adaptable across various energy sources.	[35], [36]

3. Proposed system overview

This section, presents the design and development of a smart RSU system that aims to improve traffic monitoring in urban areas and highways using advanced artificial intelligence techniques, this study to develop and implement smart RSUs as an essential part of intelligent transportation system (ITS) for smart city applications, focusing on providing low-cost solutions to improve transportation efficiency in urban areas and highways. We will experiment with different Android smart devices with varying performance to choose the most suitable one to be a smart roadside monitoring unit that can capture images and send them to a control monitoring center (CMC) for data processing, as well as receive and broadcast final reports on road conditions to end users. We aim is to find the optimal device that can be deployed in urban areas and highways efficiently and reliably. The CMC architecture consists of four specialized servers, each with its distinct functional role. This research mainly focuses on the first server, concerned with vehicle detection, recognition, and counting. The first server uses advanced deep learning models such as YOLOv8 and the SAHI algorithm process the images received from the smart RSUs, ensuring the accuracy of detection, traffic flow analysis, and congestion detection. The implementation of YOLOv8 on Android can be approached in three main ways: either the Android device itself captures and processes the image locally, edge computing is employed, or by using remote servers. In our case, YOLOv8 and SAHI are hosted on a server located in a central control and monitoring center, where the analysis is performed. This allows the Android device to act primarily as an interface for capturing and sending images to the server, with all computationally intensive tasks offloaded to the centralized infrastructure. At the same time, Android devices are used to broadcast reports of traffic conditions to end-users. Future research may explore the feasibility of implementing the first and second approaches to evaluate their effectiveness in various scenarios.

The second server analyses road conditions, detects accidents, and performs maintenance using AI techniques, while the third server generates statistical reports related to traffic density and road conditions.

In addition, a Hypertext Transfer Protocol (HTTP) server is used to receive images and forward them to servers 1 and 2 for further processing. This paper does not address the functions of other servers, as they will be addressed in future work, and the main goal here is to focus on the first server. Figure 3 illustrates the structure of the proposed system.

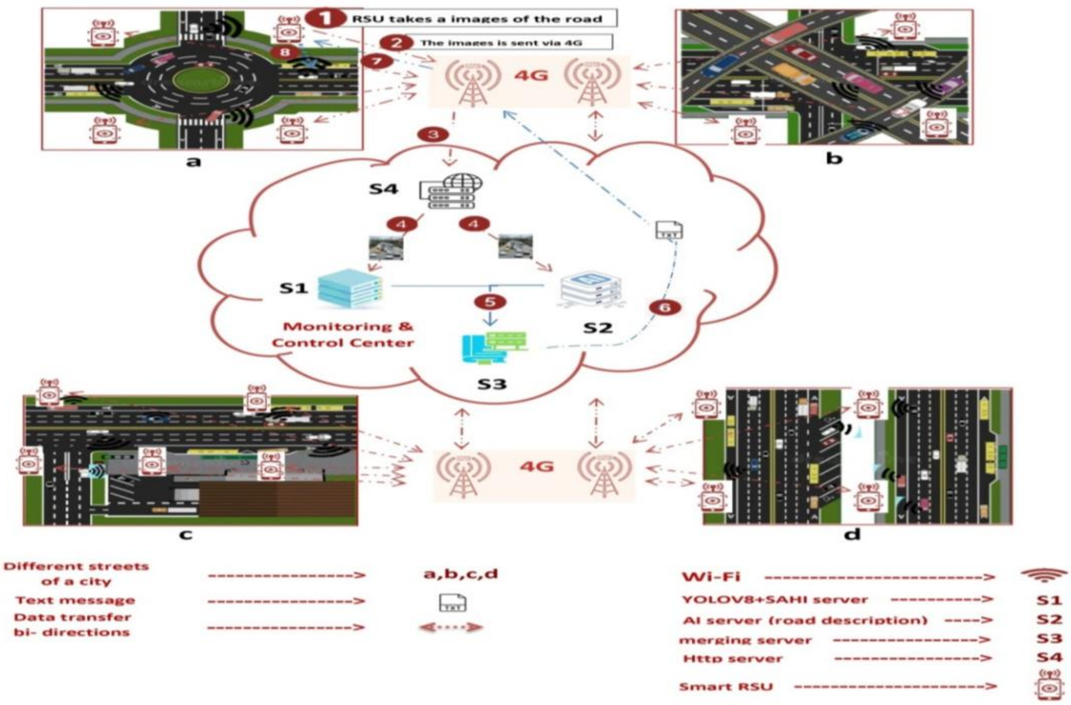


Figure 3. Proposed system architecture

Real-time processing is achieved, as evidenced by the following processing rates:

- Smart RSUs: Capable of handling 30,000 packets per second [37].
- HTTP Server: Processes up to 100,000 packets per second [37].
- YOLO+SAHI Server: Each server can process 55 images per second when utilizing optimized frameworks such as TensorRT [38].
- Chat GPT-4O Server: Similarly, each server can handle 55 images per second [39].

To manage these processing demands, the system relies on four servers, with each server assigned specific tasks based on its processing capacity. The division of labor ensures that high-throughput requirements are met, and bottlenecks are avoided. This approach leverages the efficiency of modern hardware accelerators like NVIDIA GPUs and AI-optimized infrastructure to support the computationally intensive operations of YOLOv8 and SAHI, while simultaneously meeting the real-time constraints of packet and image processing.

The system is based on an open-source infrastructure, which provides flexibility for future extensions, such as integrating pedestrian monitoring, license plate recognition, and vehicle speed measurement. This architecture contributes significantly to the enhancement of smart transportation infrastructure within urban environments, delivering comprehensive solutions to the challenges posed by traffic congestion and road management.

A. System process sequence

The smart RSUs initiate the process by capturing images of the road, which are transmitted via the 4G network to the HTTP server. This server then forwards the images to Servers 1 and 2 for processing. Once the processing is completed, the resulting data is transmitted to Server 3, which prepares detailed reports on traffic and road conditions. These reports are subsequently sent back through the 4G network to the smart RSUs, which broadcast real-time updates via Wi-Fi, allowing end users to access the information through a dedicated application that provides a city map with real-time road and traffic conditions.

B. System Architecture

1- Smart RSUs

The smart RSUs are a critical component of the ITS, providing efficient traffic monitoring capabilities. These units are based on open-source Android devices and utilize open detection models, such as YOLOv8, to enhance transportation efficiency. The proposed smart RSUs consist of mobile devices equipped with processors, memory, Global System for Mobile Communications (GSM), Wireless Fidelity (Wi-Fi), and battery power. The experimental evaluation was conducted using smartphones of varying quality to determine the optimal device for smart RSU deployment. The details of the selected phones are presented in Table 2 [40-42]. The devices were mounted on stable platforms to control image capture angles and enhance image stabilization. Performance comparisons were made based on image resolution, quality, and their impact on vehicle detection accuracy. The smart RSUs are a critical component of the ITS, providing efficient traffic monitoring capabilities. These units are based on open-source Android devices and utilize open detection models, such as YOLOv8, to enhance transportation efficiency. The proposed smart RSUs consist of mobile devices equipped with processors, memory, Global System for Mobile Communications (GSM), Wireless Fidelity (Wi-Fi), and battery power.

Table 2: Specifications of the Selected Proposal Smart RSUs

Mobile phone category	Mobile device name	Released	Mobile device camera resolution	Price of device
LOW	ZTE Blade A71	Oct-21	Triple, 16 MP, f/1.8 (wide), AF; 8 MP, f/2.2 (ultrawide); 2 MP, f/2.4 (depth)	78 $
Medium	Samsung Galaxy A32 5G	22-Jan-21	Quad, 48 MP, f/1.8 (wide), PDAF; 8 MP, f/2.2 (ultrawide); 5 MP, f/2.4 (macro); 2 MP, f/2.4 (depth)	85 $
High	Samsung Galaxy S21 Ultra	29-Jan-21	Quad, 108 MP, f/1.8 (wide), PDAF, OIS; 10 MP, f/2.4 (telephoto), OIS, 3x optical zoom; 10 MP, f/4.9 (periscope telephoto), OIS, 10x optical zoom; 12 MP, f/2.2 (ultrawide), Super Steady video	257 $

The experimental evaluation was conducted using smartphones of varying quality to determine the optimal device for smart RSU deployment. The details of the selected phones are presented in Table 1. The devices were mounted on stable platforms to control image capture angles and enhance image stabilization. Performance comparisons were made based on image resolution, quality, and their impact on vehicle detection accuracy. It is worth noting that the device classifications (low, mid-range, and high-end) are based on the classifications provided by the manufacturers themselves, which take into account a wide range of technical and commercial factors, including overall performance, camera quality, and hardware resources. Although the price difference between some devices, such as $78 and $85, seems small, the classification reflects tangible technical differences, including camera quality, overall performance, and memory capacity, which affects the efficiency of the device in performing smart surveillance tasks within the proposed system.

2- Control and Monitoring Center (CMC)

The CMC serves as the core of the system for collecting and analyzing traffic data. We assume it comprises four servers, each playing a specific role in data processing. In this work, we will focus on Server 1 only, which is responsible for vehicle detection, recognition, and counting. The other servers will be addressed in future works. Server1– Detection and Recognition: The server employs advanced techniques for real-time traffic monitoring, relying on the YOLOv8 deep learning model [43] and the SAHI algorithm [44] to analyze images received from smart RSUs. The SAHI algorithm was chosen because it efficiently slices large images, enhancing YOLOv8's performance on small or distant objects. Comparisons with standard YOLOv8 show significant accuracy improvements, particularly in high-resolution surveillance contexts. Parameter choices, like the 640x640 slice size, balance detail, and computational efficiency.

SAHI is designed to enhance object detection performance, particularly in large-scale and high-resolution images. It breaks down images into smaller ones for detection and then reassembles the results. This method is particularly useful for the detection of small objects in surveillance, aerial detection, and agriculture, where objects may be distant and represented by a few pixels, making their detection difficult [43].

YOLO-based models like YOLOv8, though effective, face challenges with small objects. SAHI improves the accuracy of detecting such objects by slicing the image into overlapping sections (usually set to 640x640 pixels) [44]. This method preserves detail and reduces computational strain, making it easier to extract key features of the model. SAHI can be used to create any object recognition model without the need for extensive customization, allowing flexibility across different applications [45].

In our road monitoring system, the SAHI algorithm significantly improved YOLOv8's detection capabilities, especially for small and distant vehicles. The slicing process enhances detection accuracy by ensuring that detailed features are captured, and it minimizes false detections by managing overlap ratios between sub-images. The size of slices and the overlap ratio are critical in balancing accuracy and computational load.

Key Features of SAHI [43]

1. Seamless Integration: Easily integrates with YOLO models without major code adjustments.
2. Resource Efficiency: Reduces memory use by slicing large images into smaller parts
3. High Accuracy: Combines overlapping detection boxes for more precise results.

Sliced inference entails segmenting high-resolution images into smaller sections for autonomous processing. This approach decreases computational burden, sustains detection accuracy, and enhances scalability. It enables object detection across multiple resolutions and proves effective in resource-constrained settings [45]. For example, our study employed various Android devices with different resolutions, yielding 63, 252, and 432 segmented images, respectively. A 20% overlap among segments was implemented to guarantee thorough object detection, thereby boosting the collective efficiency of the YOLOv8 and SAHI integration. The SAHI algorithm's versatility allows it to be widely adaptable across different applications, enhancing the detection of small objects while maintaining computational efficiency.

C. Experimental Setup

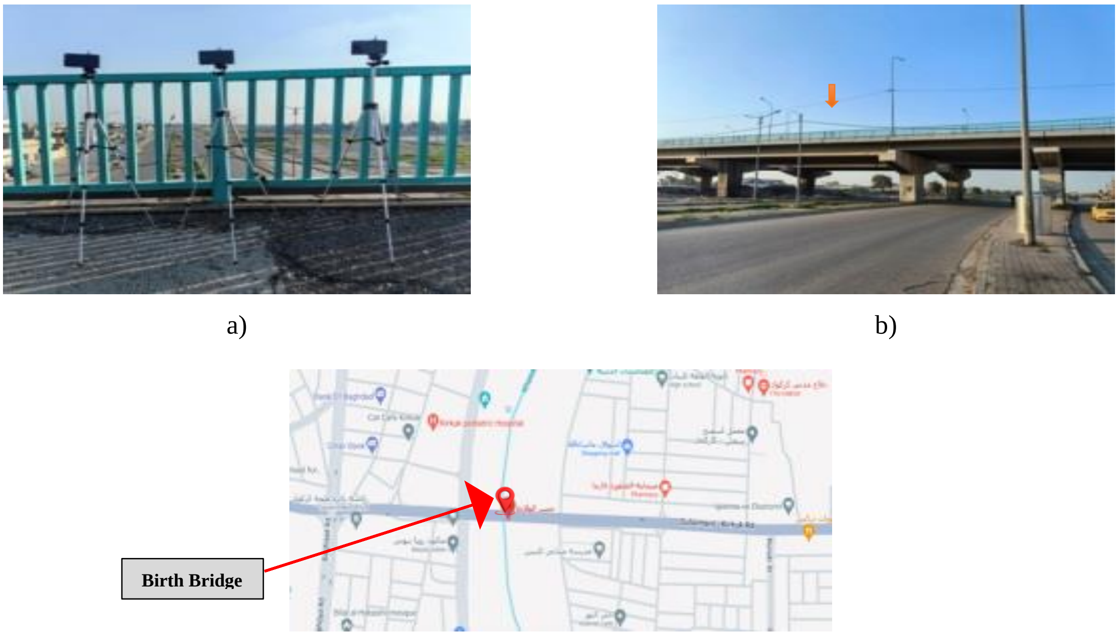
1. Hardware and software

This section describes the hardware and software components chosen for the experimental setup. Table 3 presents the hardware and software components used.

2. Experimental locations and data collection

This section describes the strategic locations chosen for data collection, offering diverse traffic and lighting conditions to enhance the comprehensiveness of the results.

2.1 Location 1: Birth Bridge
Cameras were installed at a height of 9.5 m, overlooking a four-lane street with two secondary streets. This location typically experiences low traffic, allowing for clear data collection and providing a wide field of view, as shown in Figure 4.



a) b) c)

Figure 4. First location (a) How to install Android- based devices (b) Real photography shooting location (c) Photography location on the map.

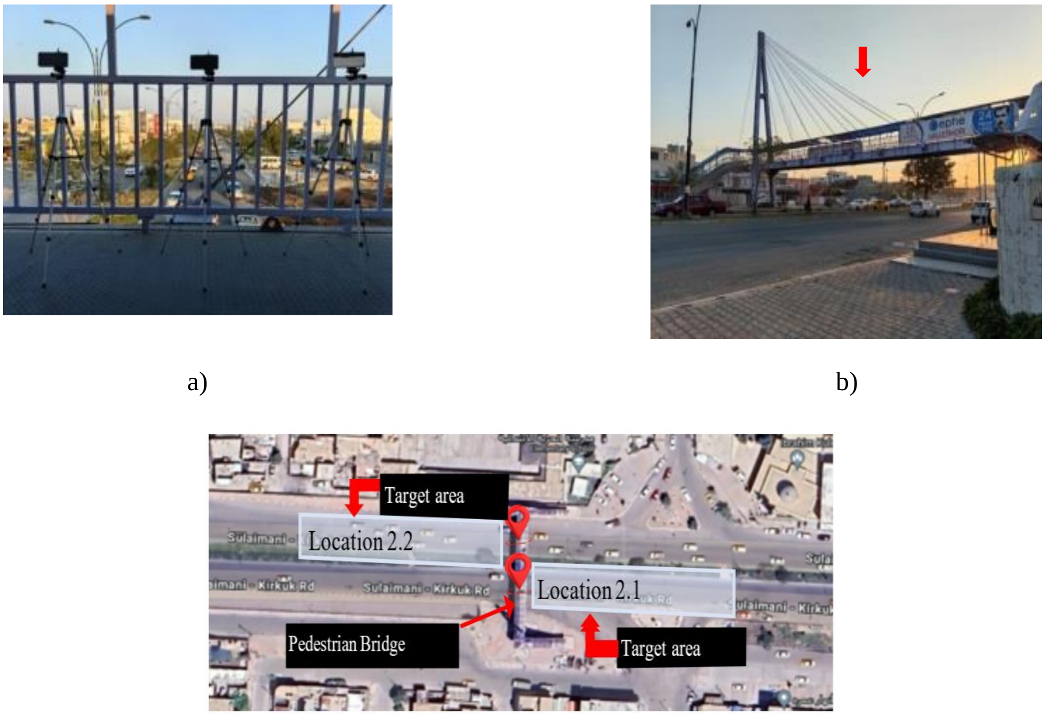
Table 3: Hardware and software components used

Category	Item	Details
Hardware	Laptop	Lenovo with an Intel Core i5 processor (11th generation) and 8GB of RAM.
	Cameras	Three cameras based on the Android system (high-cost, medium-cost, low-cost).
Software	Operating System	Windows 11 Pro.
	AI Framework	YOLOv8 and SAHI in Visual Studio Code

2.2 Location 2: Pedestrian Bridge
Cameras were mounted at a height of 6.95 m on this bridge, which spans a busy four-lane street connecting Birth Bridge and Shorja Bridge. This location includes two sub-sites:

- Site 2.1: Cameras directed towards Shorja Bridge for daytime photography during heavy traffic.
- Site 2.2: Cameras facing Birth Bridge for night-time photography, utilizing available lighting, as illustrated in Figure 5.

These locations were strategically selected to capture diverse data under varying traffic and lighting conditions. A total of 270 images were captured at both locations, managed by two individuals, with each device collecting 30 images per minute. Figure 6 illustrates the experimental setup. Table 4 summarizes the details of image capture across the three locations, including the number of images, object types, and specifications of the devices used.



a) b)

c)

Figure 5. Second location (a) How to install Android- based devices (b) Real photography shooting location (c) Photography location on the map

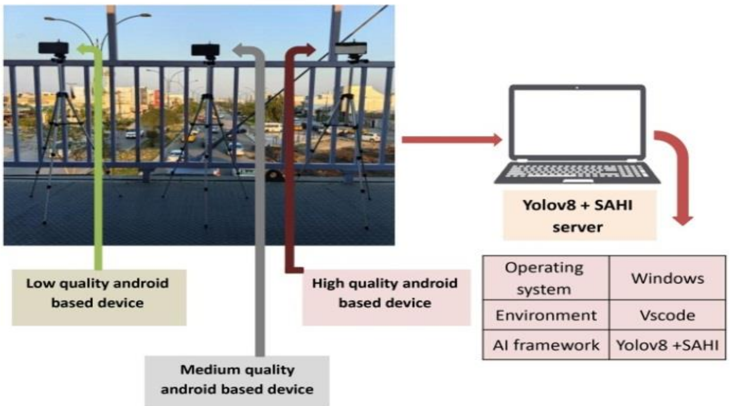


Figure 6. Experimental setup.

Table 4: Detailed overview of image acquisition across various locations

Location	Images per Device (images/min)	Total Images per Location	Object Types	Camera Resolution (MP)	Original Image Size (pixels)	Number of Slices (640x640)
Location 1	30	90	Buses, Trucks, Cars	16	4608 x 3456	63
Location 2.1	30	90	Buses, Trucks, Cars	64	9284 x 6936	252
Location 2.2	30	90	Buses, Trucks, Cars	108	12000 x 9000	432

The experiment in Kirkuk on 23/4/2024 tested the model under clear skies (21°C–34°C) across daylight, evening, and night to evaluate performance in varying lighting. Clear weather ensured visibility; while future experiments may explore performance under varying weather conditions such as rain or fog.

4. Methodology

This section outlines the methodology used in the experiment, focusing on distance measurement, detection rate calculations, and image processing techniques, while incorporating clearer explanations of formulas and technical terms.

Distance Measurement: Distance measurements were taken using basic tools and fixed reference points, such as the distance between lighting poles. For example, at Site 1, the distance from a specific reference point (Pillar No. 4) was approximately 150 meters. Similar distance measurements were applied at other locations to maintain consistency, as illustrated in Figure 7.

a) b) c)

Figure 7. Locations and distance limits. (a) first location (b) second location 2.1 (c) second location 2.2

Detection Rate (DR) Calculation: The Detection Rate (DR) was calculated to assess the accuracy of vehicle detection within specific distance ranges. DR is determined by dividing the number of detected vehicles by the total number of vehicles (both detected and undetected) within the range of interest. For example, within a 200-meter range, DR is calculated as:

$$\mathrm{DR} = \frac{\text{Number of detected vehicles}}{\text{Total number of vehicles (detected + undetected)}} \qquad (1)$$

where number of detected vehicles refers to the vehicles correctly identified by the system, Total number of vehicles includes all vehicles in the range, both detected and undetected.

The accuracy of the proposed system was further evaluated by comparing the actual number of vehicles present in an image (manually counted) with the number of vehicles detected by the YOLOv8 model combined with the SAHI algorithm. The following steps were undertaken for each device and distance:

1. The processed output images, generated by the YOLOv8 and SAHI model running on Visual Studio, were manually reviewed.
2. The total number of vehicles present in each image was manually counted and recorded as the ground truth.
3. The number of vehicles detected by the model was also recorded.
4. The detection accuracy for each image was calculated using equation (1).
5. These steps were repeated for all images captured by each device at different distances.
6. Results were tabulated in an Excel sheet, and the average accuracy was computed for each device and distance.

This approach ensured consistent and fair evaluation across all devices and scenarios. The results were used to determine the performance trends and capabilities of the system.

The experiment involved processing sample images from three Android-based devices (low-cost, medium-cost, high-cost) across different settings and models. The YOLOv8 model used in this study is a pre-trained standard model provided by Ultralytics, trained on the COCO data set. To adapt the model for this application, it was filtered to exclusively detect vehicles, excluding other object categories to focus on traffic-related analysis. Initially, images from each device were analyzed using the YOLOv8 large-sized model to establish baseline performance. The purpose of this step is to compare the performance of YOLOv8 when used alone versus its performance when combined with the SAHI algorithm. The same image was then analyzed using the YOLOv8 small-sized model combined with the SAHI algorithm. In this process, masks were used to overcome many challenges, including natural obstacles (e.g., trees), reducing the impact of irregular vehicles that could affect detection accuracy, mitigating issues caused by vehicle headlights, and ensuring detection focused on a single side of the road with multiple lanes. This reduced the target distance and improved the detection accuracy at the specified location.

Secondly, for location-based analysis, images from each device were taken at two distinct sites to evaluate the detection range of each device under varying conditions and locations. At the first location, 30 daytime images were captured and processed. At the second location, data was collected from two sub-sites: Site 2.1 with 30 daytime images and Site 2.2 with 30 night-time images. Each set of images was processed using the YOLOv8 small-sized model combined with the SAHI algorithm. It is worth noting that Table 4 provides detailed information about the images used in this analysis. The SAHI settings included a slice height of 640 pixels, a slice width of 640 pixels, an overlap height ratio of 0.2, and an overlap width ratio of 0.2, with masks applied to enhance detection accuracy by focusing on relevant areas of the images.

The purpose of the initial setup was to compare the performance between YOLOv8 alone and YOLOv8 combined with the SAHI algorithm, highlighting the effectiveness of SAHI in enhancing detection accuracy, especially for small or distant objects. The purpose of the second setup was to determine the detection range and performance of each device at different distances and under various environmental conditions, including different lighting and traffic scenarios. Figure 8 illustrates the application of these masks.

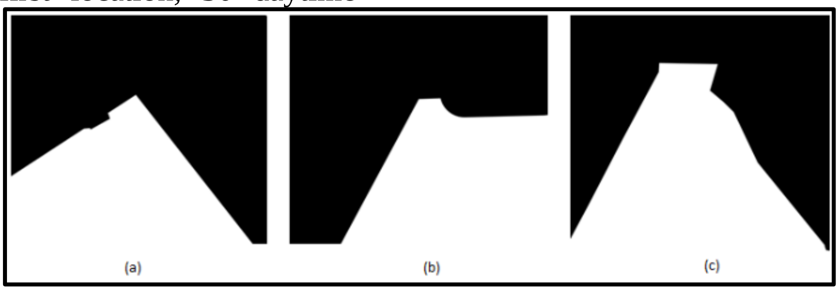


Figure 8. Masks used to remove unwanted image parts (a) Mask for first location, (b) Mask for second location 2.1, (c) Mask for second location 2.2

5. Results

This section presents the experimental results evaluating the system's performance in vehicle detection. The findings showcase how different models performed across various devices and conditions, providing a detailed analysis of detection rates and accuracy. Additionally, it discusses the challenges encountered and the advanced techniques employed to enhance results.

A. Comprehensive Statistical Analysis of Detection Performance

The performance data for all devices was collected and made in the form of tables containing 90 images for each device, with 30 images for each location. Table 5 shows the performance data using YOLOv8 only and YOLOv8 with the SAHI algorithm using the high-cost device (Samsung Galaxy S21 Ultra) at the first location across three distances: 150 meters, 300 meters, and 500 meters. The original data contains 30 images for each setting, and represents the number of vehicles, the number of detected vehicles, and the accuracy percentage for each image. Then, a statistical analysis was conducted to summarize the performance using criteria such as mean accuracy, median accuracy, and standard deviation. After analyzing the original data in the table x, the statistical indicators were extracted for each distance using YOLOv8 only and YOLOv8 with SAHI. Table 6 shows these results. The results showed that the performance of YOLOv8 alone was variable, especially at long distances (500 m), where the average accuracy was only 19%, with a high standard deviation of 14.8%. In contrast, the combination of SAHI with YOLOv8 significantly improved the accuracy, where the average accuracy at 500 m increased to 8.5%, and the standard deviation decreased to 9.8%. Based on these indicators, the same approach was used to analyze the performance of other devices at different locations, and the threshold of 85% was adopted as the criterion for determining the ideal performance at long distances.

Table 5: Performance analysis of the S21 at the first site at distances of 150 m, 300 m and 500 m using YOLOv8 only and YOLOv8 with SAHI

First location for device S21 using YOLOV8 only									
Images	150m			300m			500m		
	number of vehicles	number of detected vehicles	Accuracy	number of vehicles	number of detected vehicles	Accuracy	number of vehicles	number of detected vehicles	Accuracy
Image 1	5	5	100%	7	5	71%	10	5	50%
Image 2	5	5	100%	7	5	71%	10	5	50%
Image 3	3	3	100%	8	3	38%	11	3	27%
Image 4	2	2	100%	4	2	50%	10	2	20%
Image 5	3	2	67%	5	2	40%	10	2	20%
Image 6	4	3	75%	4	3	75%	9	3	33%
Image 7	4	3	75%	4	3	75%	9	3	33%
Image 8	4	2	50%	5	2	40%	10	2	20%
Image 9	4	2	50%	5	2	40%	10	2	20%
Image 10	2	1	50%	3	1	33%	8	1	13%
Image 11	4	1	25%	4	1	25%	9	1	11%
Image 12	5	2	40%	5	2	40%	9	2	22%
Image 13	4	1	25%	5	1	20%	10	1	10%
Image 14	1	1	100%	4	1	25%	9	1	11%
Image 15	1	1	100%	4	1	25%	8	1	13%
Image 16	1	1	100%	3	1	33%	10	1	10%
Image 17	2	1	50%	5	1	20%	12	1	8%
Image 18	2	1	50%	4	1	25%	12	1	8%
Image 19	2	1	50%	3	1	33%	8	1	13%
Image 20	3	2	67%	4	2	50%	9	2	22%
Image 21	4	3	75%	4	3	75%	10	3	30%
Image 22	4	1	25%	4	1	25%	7	1	14%
Image 23	4	3	75%	6	3	50%	11	3	27%
Image 24	4	2	50%	6	2	33%	11	2	18%
Image 25	2	2	100%	3	2	67%	7	2	29%
Image 26	2	0	0%	5	0	0%	10	0	0%
Image 27	2	1	50%	5	1	20%	10	1	10%
Image 28	1	1	100%	5	1	20%	10	1	10%
Image 29	1	0	0%	2	0	0%	8	0	0%
Image 30	1	1	100%	2	1	50%	8	1	13%

B. Detection accuracy analysis for all scenarios

In this section, the detection accuracy is analyzed across all scenarios, comparing the results between using the YOLOv8 and SAHI algorithms, as well as the role of masks in improving accuracy under different conditions.

1. **Scenario (1)**

In this scenario, several challenges are addressed regarding object detection with large image sizes. The images used are of high resolution (at least 3456 x 4608 pixels), providing detailed visuals that support accurate detection under typical circumstances. However, since YOLOv8 was trained on images of a smaller size (640 x 640 pixels), resizing these large images results in some loss of detail. This loss particularly affects the detection of small or distant objects, leading to decreased detection accuracy.

For detection range and accuracy, the results vary between daytime and night-time photography. During the day, the detection range reaches up to 150 meters with an accuracy rate of approximately 90%. Beyond this distance, accuracy noticeably declines. At night, the detection range is slightly reduced to 140 meters, and accuracy drops below 90% at this distance, as shown in Figure 9.

During the experiment, several challenges emerged progressively, each requiring tailored solutions to ensure accurate detection. The initial challenges included natural obstacles, such as trees separating road lanes, irregularly parked vehicles, and the glare from vehicle headlights at night. Figure 10 illustrates issues resolved using image masks. These factors interfered with the model's ability to distinguish objects accurately. To address these issues, the detection process was limited to a single side of the road rather than both sides, regardless of the number of lanes on that side. Masks were applied to constrain the detection area within the image, focusing only on the relevant sections of the road. This targeted approach reduced the impact of obstacles and minimized interference, leading to significant improvements in detection. several challenges were observed regarding object detection with large image sizes. The images used in the experiment were of high resolution (at least 3456 × 4608 pixels), providing detailed visuals that support accurate detection under typical circumstances. However, because YOLOv8 was trained on small images (640 × 640 pixels), resizing these large images resulted in a loss of detail. This reduction primarily affected the detection of distant objects, reducing detection accuracy.

To overcome this issue, the SAHI algorithm was applied in the second scenario. SAHI divides high-resolution images into smaller connected segments (slices) that best match the model's training dataset. This method preserved detail, and significantly improved the detection quality, especially for distant particles. The use of SAHI has proven helpful in extending the detection range and addressing the limitations imposed by image resolution. Table 7 shows that the detection results when using the YOLOv8 model alone compared to using the YOLOv8 model combined with the SAHI algorithm. The table highlights the improvement in detection accuracy, particularly for small or distant objects, when the SAHI algorithm is employed. This enhancement is attributed to SAHI's ability to slice high-resolution images into smaller sections, preserving critical details and overcoming challenges such as object overlap and image resizing effects. The results demonstrate the effectiveness of combining YOLOv8 with SAHI for more accurate and reliable detection under diverse conditions.

2. **Scenario** (2)

As previously mentioned, masks were applied to address challenges such as overlapping vehicles and natural obstacles. Additionally, the SAHI algorithm was employed to handle the large size of images. In this scenario, the SAHI algorithm was utilized to divide images into 640 x 640 slices.

The number of slices depends on the resolution of the image used. For low-cost devices, 63 slices were obtained for a resolution of 4608 x 3456. For medium-cost devices, the image was divided into 252 slices for a resolution of 9284 x 6936. Finally, for high-cost devices, 432 slices were created for a resolution of 12000 x 9000.

Regarding the results of detection using the sliced images, the daytime detection in Figure 11 indicates that at the first location, the detection accuracy reached a range of 500 meters, leading to an improved detection rate consistent with the initial research plan. However, at the second location, the detection range was limited to 300 meters due to geographical constraints. For night-time detection, in Figure 11, the detection range was again limited to 300 meters due to reduced visibility at night.

Lastly, the effectiveness of masks demonstrated their ability to effectively address issues related to overlapping vehicles and natural obstacles, resulting in improved detection accuracy.

Figure 9. YOLOv8 test results in locations (a) first location, (b) second location 2.1, (c) second location 2.2

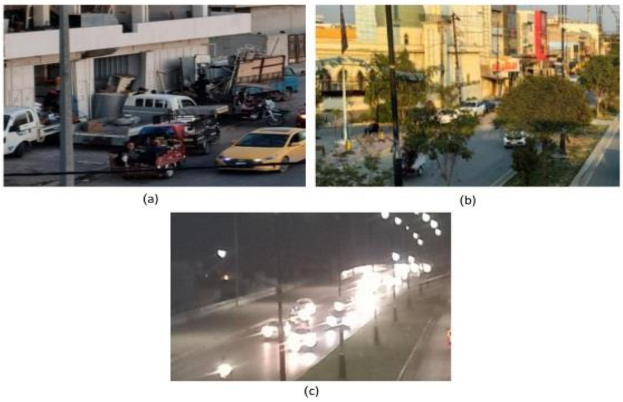


Figure 10. Issues resolved using image masks (a) overlapping vehicles parked on the side of the road, (b) natural obstacles, such as trees, (c) cars headlights

Table 7: Device performance rates under various traffic conditions and distances, comparing results using YOLOv8 alone and YOLOv8 combined with the SAHI algorithm

Detection rate using only YOLOV8							Detection rate using YOLOV8 and SAHI algorithm						
Device type	Condition Day/Night	100 m	150 m	200 m	300 m	500 m	Device type	Condition Day/Night	100 m	150 m	200 m	300 m	500 m
Low cost	Low traffic (Day)	N/A	97%	N/A	90%	60%	Low cost	Low traffic (Day)	N/A	97%	N/A	90%	60%
	Moderate traffic (Night)	72%	N/A	80%	56%	N/A		Moderate traffic (Night)	72%	N/A	80%	56%	N/A
	High traffic (Day)	95%	N/A	86%	77%	N/A		High traffic (Day)	95%	N/A	86%	77%	N/A
Medium cost	Low traffic (Day)	N/A	100%	N/A	92.8%	83%	Medium cost	Low traffic (Day)	N/A	100%	N/A	92.8%	83%
	Moderate traffic (Night)	98%	N/A	87%	45%	N/A		Moderate traffic (Night)	98%	N/A	87%	45%	N/A
	High traffic (Day)	95%	N/A	85%	56%	N/A		High traffic (Day)	95%	N/A	85%	56%	N/A

high cost	Low traffic (Day)	N/A	100%	N/A	95%	85%	high cost	Low traffic (Day)	N/A	100%	N/A	95%	85%
	Moderate traffic (Night)	98.42%	N/A	96%	90%	N/A		Moderate traffic (Night)	98.42%	N/A	96%	90%	N/A
	High traffic (Day)	97%	N/A	98%	95%	N/A		High traffic (Day)	97%	N/A	98%	95%	N/A

C. Detection accuracy analysis (by device type)

In this section, detection accuracy is analyzed by device type, with emphasis on the effect of device quality on detection accuracy and the extent to which environmental conditions and processing techniques used influence it.

1. Detection accuracy with low-cost devices

After applying the YOLOv8 model, the SAHI algorithm, and the masking technique to images from low-cost devices, the results demonstrated varying levels of detection accuracy across different scenarios.

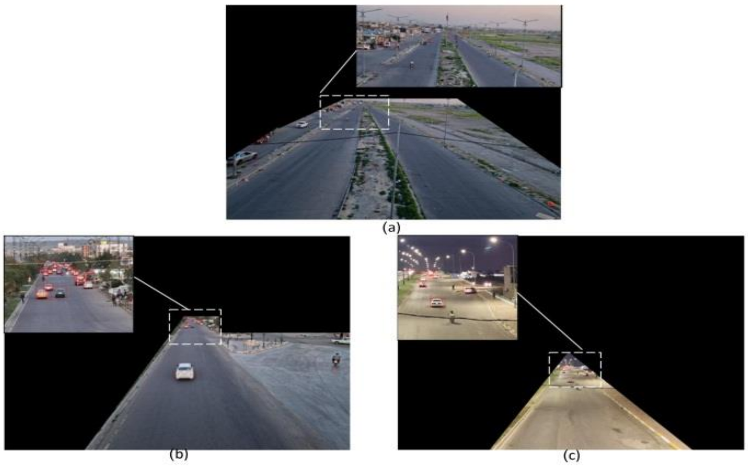


Figure 11. YOLOv8 and SAHI algorithm with mask results in locations (a) first location, (b) second location 2.1, (c) second location 2.2

- First location: During the day in non-congested traffic, detection accuracy was high, reaching 97% at 150 meters and 90% at 300 meters. However, accuracy dropped to 60% at 500 meters due to limitations in the camera sensor quality of low-cost devices.
- Second location 2.1: In a crowded traffic scenario during the day, detection accuracy was also high, reaching 95% at 100 meters and 86% at 200 meters. At 300 meters, accuracy dropped to 77%, primarily due to sensor quality issues and vehicle occlusion.
- Second location 2.2 (Night Photography) At night, in an average traffic scenario, detection accuracy was lower, with 72% at 100 meters and 56% at 300 meters. This reduction in accuracy was attributed to poor image quality from weak camera sensors and the effects of vehicle headlights. Figure 12 illustrates the undetected patterns for all locations and distances. Undetected patterns typically arise when vehicles are obscured by other vehicles or when the distance between the vehicles and the camera exceeds the effective detection range.

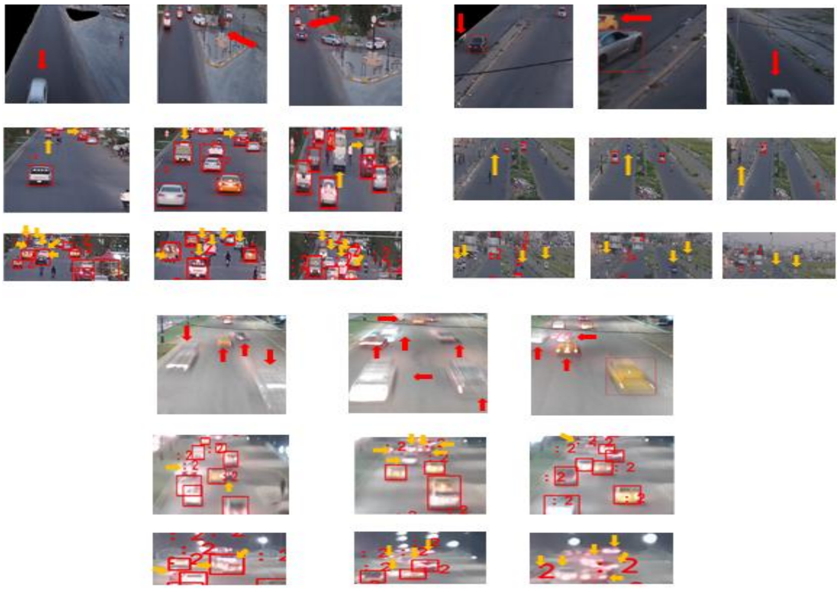


Figure 12. Examples of cases where accuracy was reduced for the low-cost Android-based device

2. Detection accuracy with medium-cost devices

After applying the YOLOv8 model, the SAHI algorithm, and the masking technique to images from medium-cost devices, the results demonstrated varying levels of detection accuracy across different scenarios.

- First location: Daytime detection accuracy in non-congested traffic was nearly perfect, reaching 100% at 150 meters and remaining high at 300 meters (92.8%). However, the accuracy dropped to 83% at 500 meters due to limitations of the medium-quality camera sensor.
- Second location 2.1: In crowded traffic conditions during the daytime, detection accuracy started at 95% at 100 meters and decreased progressively with distance, reaching 86% at 200 meters and 77% at 300 meters. The decrease in accuracy was due to vehicle occlusion and sensor limitations.
- Second location 2.2 (Night Photography): Under average traffic conditions at night, detection accuracy was high, achieving 98% at 100 meters. However, the accuracy dropped to 87% at 200 meters and 45% at 300 meters, mainly due to night-time conditions and sensor quality limitations. Figure. 13 illustrates the undetected models, Undetected patterns typically arise when vehicles are obscured by other vehicles or when the distance between the vehicles and the camera exceeds the effective detection range.

Figure 13. Examples of cases where accuracy was reduced for the medium-cost Android-based device

3. Detection Accuracy with High-Cost Devices

After applying the YOLOv8 model, the SAHI algorithm, and the masking technique to images from high-cost devices, the results demonstrated varying levels of detection accuracy across different scenarios.

- First location: During the daytime in non-congested traffic, detection accuracy was excellent, achieving 100% at 150 meters, 95% at 300 meters, and 85% at 500 meters. The slight drop in accuracy at 500 meters was attributed to the need for even higher resolution to distinguish distant objects.
- Second location 2.1: In crowded traffic, detection accuracy remained high, with 97% at 100 meters, 98% at 200 meters, and 95% at 300 meters. The primary challenge affecting detection accuracy was occlusion, with examples such as a vehicle pulled by a motorcycle, commonly referred to as a "Stota."
- Second Location 2.2 (Night Photography): Under average traffic conditions at night, high-cost devices performed well despite challenging conditions. Detection accuracy was 98.42% at 100 meters, 96% at 200 meters, and 90% at 300 meters. The slight drop in accuracy was attributed to night-time conditions, in addition to previously mentioned challenges such as reduced visibility and sensor limitations. Figure. 14 illustrates the undetected models, Undetected patterns typically arise when vehicles are obscured by other vehicles or when the distance between the vehicles and the camera exceeds the effective detection range.

Figure 14. Examples of cases where accuracy was reduced for the high-cost Android-based device

The experiments reveal performance variability across devices and environmental conditions. Key findings include:

- Detection range and accuracy: The YOLOv8 model showed improved performance with high-resolution images due to the SAHI algorithm, which enabled accurate object detection over greater distances. The use of the SAHI algorithm to split images into smaller segments effectively preserved image detail and improved detection accuracy across different devices. Additionally, the masking technique proved beneficial in mitigating issues caused by overlapping vehicles and natural obstacles, enhancing detection rates.
- Impact of device quality: Higher-cost devices generally provide better detection accuracy due to their superior image resolution and sensor quality. High-cost devices consistently outperformed medium- and low-cost devices across various scenarios, with significant improvements in detection accuracy during both daytime and night-time.

In addition, using a small dataset of only 270 images is limited in representing real-world scenarios. While device quality significantly impacts detection performance, another critical factor influencing generalization ability is the dataset size. This limited data size may lead to potential overfitting and reduced generalization ability of the model across diverse environments. To overcome this challenge, it is recommended to expand the dataset to include more diverse scenarios, including different lighting conditions, traffic congestion, and distances.

- Challenges: Detection accuracy was affected by factors such as image quality, lighting conditions, overlapping vehicles, natural obstacles, and vehicle headlights.

6. Discussion of results

The maximum detectable distance for each Android-based device varies notably depending on device quality and lighting conditions (day or night). The system performance was limited by the capabilities of the devices used, with the highest camera resolution used being 108 MP. The results showed that as the camera resolution increased, the detection range increased significantly, with devices with high-resolution cameras achieving detection distances of up to 500 meters with an accuracy of 85%. Therefore, to achieve longer detection distances in the future, it is recommended to use devices with higher-resolution cameras to extend the performance range and increase the detection accuracy.

For high-quality Android-based devices, the maximum detectable distance reaches 500 meters with a detection accuracy of 85%. Reducing the distance to 300 meters boosts the accuracy to 90%. This device maintains strong performance in both daytime and night-time photography.

Medium-cost devices were capped at a maximum detection range of 200 meters due to the significant drop in accuracy observed under challenging conditions. While detection accuracy was acceptable during the day with non-congested traffic (92.8% at 300 meters), it dropped to an unacceptable 55% in high-congestion and night-time scenarios. This limitation makes 200 meters a practical compromise for reliable performance, ensuring accuracy remains within an acceptable threshold across different conditions.

Low-cost devices were limited to a maximum detection range of 100 meters due to their sensitivity to low-light conditions. While daytime detection accuracy was high at 95%, night-time performance dropped to 72%, even at closer ranges. This trade-off reflects the decision to prioritize daytime usability over achieving high accuracy at night, given the lower capabilities of the device.

These findings suggest that higher-quality devices are more suitable for applications requiring extended detection ranges and greater accuracy, particularly in varied lighting scenarios. Selecting the appropriate device type in real-world implementations depends on the specific needs for detection distance and accuracy, with particular attention to differences in day and night performance. Figure 15 presents detailed results for these observations.

7. Limitations

While the proposed system demonstrated promising performance, several limitations were identified that warrant further attention:

1. Device Constraints: The system's performance was limited by the hardware capabilities, particularly the camera resolution and sensor quality, which directly influenced detection accuracy and range.
2. Dataset Size: The limited dataset of 270 images reduced the system's ability to generalize across diverse real-world conditions, potentially affecting its robustness in untested scenarios.
3. Environmental Factors: Issues such as overlapping vehicles, natural obstacles, and poor lighting conditions (e.g., night-time scenarios) posed challenges for consistent detection accuracy. While these challenges were addressed to a significant extent through the use of masking techniques, they remain influential in certain complex scenarios and highlight areas for further optimization.

These limitations underscore the need for future research and development efforts to address these challenges, particularly by improving hardware capabilities, expanding the dataset, and optimizing the system for diverse and challenging conditions.

7. Conclusion

This research developed a cost-effective, Android-based traffic monitoring system using smart RSUs that leverage YOLOv8 models and the SAHI algorithm to enhance detection accuracy under various conditions. Field experiments demonstrated that high-cost devices deliver superior performance in terms of detection range and accuracy, making them ideal for applications that demand high precision and extended coverage. Meanwhile, low- and medium-cost devices provide a budget-friendly alternative for less intensive monitoring needs, increasing the system's adaptability to diverse urban environments. This flexible design contributes significantly to improving traffic management in smart cities, reducing the environmental and economic impacts of traffic congestion, and providing offering real-time information to users. Future work could focus on optimizing performance in different weather conditions and expanding features, such as pedestrian monitoring, speed measurement, and plate recognition, to further enhance the system's utility. Ethical concerns and privacy protections are important when using a camera-based system for road surveillance. The proposed system has demonstrated the ability to improve traffic management without compromising individual privacy and focuses only on vehicle detection and traffic data analysis to ensure full privacy made anonymous. This approach is in line with international data protection standards and emphasizes the ethical responsibility of the system design, which is important when using camera-based systems for road surveillance. The proposed system enhances traffic management without compromising individual privacy by focusing solely on vehicle detection and traffic data analysis while ensuring that all personal information remains hidden.

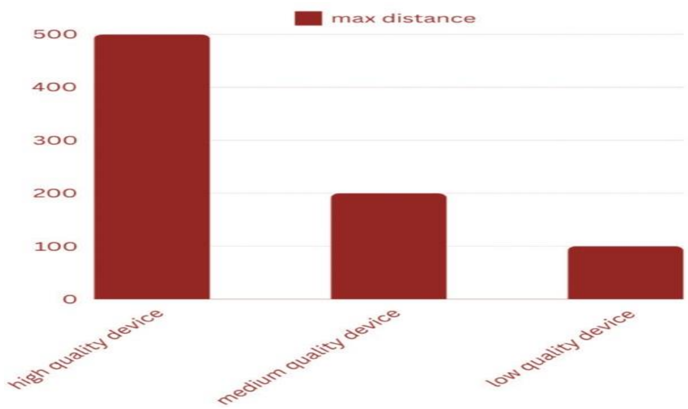


Figure 15. Maximum coverage for the three devices

Reference

[1] United Nations. “2014 Revision of the World Urbanization Prospects | Latest Major Publications - United Nations Department of Economic and Social Affairs.” Un.org, 2014, www.un.org/en/development/desa/publications/2014-revision-world-urbanization-prospects.html.

[2] M. A. Balwan, T. Varghese and S. Nadeera, "Urban Traffic Control System Review - A Sharjah City Case Study," 2021 9th International Conference on Traffic and Logistic Engineering (ICTLE), Macau, China, 2021, pp. 46-50, doi: 10.1109/ICTLE53360.2021.9525750.

[3] Agarwal, P.K. “Intelligent Transportation Systems and Its Tools as a Solution for Urban Traffic Congestion: A Review.” Journal of Emerging Technologies and Innovative Research, 2020, www.academia.edu/80802555/Intelligent_Transportation_Systems_and_Its_Tools_as_a_Solution_for_Urban_Traffic_Congestion_A_Review. Accessed 20 Jan. 2025.

[4] Bista, Raghu, and Surendra Paneru. “Does Road Traffic Congestion Increase Fuel Consumption of Households in Kathmandu City?” Journal of Economic Impact, vol. 3, no. 2, 29 July 2021, pp. 67–79, https://doi.org/10.52223/jei3022102. Accessed 11 Sept. 2021.

[5] A. K. M., N. K. T., A. J. K., A. P., and M. I. P., "Economic evaluation of traffic congestion & design of traffic signal with simulation at Ottapalam," International Journal of Research in Applied Science

and Engineering Technology, 2023. doi: 10.22214/ijraset.2023.53881.

[6] Cherednichenko, Oleksandra, and Asta Valackienė. “INTELLIGENT TRANSPORT SYSTEMS as TRAFFIC FLOW MANAGEMENT TOOL (the CASE of KYIV).” Urban Development and Spatial Planning, no. 80, 30 May 2022, pp. 416–450, https://doi.org/10.32347/2076-815x.2022.80.416-450. Accessed 17 Feb. 2023.

[7] SUBAIR, Sulaiman Olayinka, et al. “Evaluation of Traffic Congestion in an Urban Roads: A Review.” ABUAD Journal of Engineering and Applied Sciences, vol. 2, no. 2, 30 Aug. 2024, pp. 1–7, https://doi.org/10.53982/ajeas.2024.0202.01-j. Accessed 21 Oct. 2024.

[8] H. Abdelati, Mohamed. “Smart Traffic Management for Sustainable Development in Cities: Enhancing Safety and Efficiency.” International Journal of Advanced Engineering and Business Sciences, vol. 5, no. 1, 24 Feb. 2024, https://doi.org/10.21608/ijaebs.2024.242361.1088.

[9] Kadiyala Ramana, et al. “A Vision Transformer Approach for Traffic Congestion Prediction in Urban Areas.” “IEEE Transactions on Intelligent Transportation Systems,” vol. 24, no. 4, 1 Jan. 2023, pp. 3922–3934, https://doi.org/10.1109/tits.2022.3233801. Accessed 11 June 2023.

[10] None Prof. Vrushali Awle, et al. “Smart Traffic Management System.” International Journal of Advanced Research in Science, Communication and Technology, vol. 3, no. 1, 4 Dec. 2023, pp. 108–124, https://doi.org/10.48175/ijarsct-14013. Accessed 16 May 2024.

[11] Golhar, Yogesh, and Manali Kshirsagar. “Emerging Technologies for Driving Road Safety and Traffic Management for Urban Area.” Journal of Computer Science, vol. 17, no. 11, 1 Nov. 2021, pp. 1104–1115, https://doi.org/10.3844/jcssp.2021.1104.1115. Accessed 9 Feb. 2022.

[12] Drushya Kamuni, et al. “Density Based Traffic Congestion Control System and Emergency Vehicle Clearance.” 2022 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS), 8 Dec. 2022, pp. 1–6, www.researchgate.net/publication/368857431_Density_Based_Traffic_Congestion_Control_System_and_Emergency_Vehicle_Clearance,https://doi.org/ 10.1109/ICPECTS56089.2022.10047102.

[13] Li, Shanwen. “Real-Time Traffic Congestion Detection Technology in Intelligent Transportation Systems.” 2022 IEEE 11th International Conference on Communication Systems and Network Technologies (CSNT), 6 Apr. 2024, pp. 1029–1033, ieeexplore.ieee.org/document/10546078, https://doi.org/10.1109/csnt60213.2024.10546078. Accessed 20 Jan. 2025.

[14] Ultralytics, "Introducing Ultralytics YOLOv8," 2023. [Online]. Available: https://www.ultralytics.com/blog/introducing-ultralytics-yolov8. Accessed: Jan. 18, 2025.

[15] DataDrivenInvestor, "YOLOv8: The Evolution of Real-Time Object Detection," 2023. [Online]. Available: https://medium.datadriveninvestor.com/yolov8-the-evolution-of-real-time-object-detection-7c158948e0de. Accessed: Jan. 18, 2025.

[16] Lou, Haitong, et al. “DC-YOLOv8: Small-Size Object Detection Algorithm Based on Camera Sensor.” Electronics, vol. 12, no. 10, 21 May 2023, pp. 2323–2323, https://doi.org/10.3390/electronics12102323..

[17] Keylabs, "Getting Started with YOLOv8: A Beginner's Guide," 2024. [Online]. Available: https://keylabs.ai/blog/getting-started-with-yolov8-a-beginners-guide. Accessed: Jan. 18, 2025.

[18] Plugger.ai, "The Evolution of Object Detection: A Journey from YOLO to YOLOv8 and Beyond - An Exploration of Advancements in the Last Seven Years," 2023. [Online]. Available: https://www.plugger.ai/blog/the-evolution-of-object-detection-a-journey-from-yolo-to-yolov8-and-beyond-an-exploration-of-advancements-in-the-last-seven-years.

[19] Ultralytics, "Ultralytics Documentation," 2024. [Online]. Available: https://docs.ultralytics.com. Accessed: Jan. 18, 2025

[20] Yousif, Adel Jalal, and Mohammed H Al-Jammas. “A Lightweight Visual Understanding System for Enhanced Assistance to the Visually Impaired Using an Embedded Platform.” Diyala Journal of Engineering Sciences, vol. 17, no. 3, 1 Sept. 2024, pp. 146–162, djes.info/index.php/djes/article/view/1377, https://doi.org/10.24237/djes.2024.17310. Accessed 20 Jan. 2025.

[21] Evans, Michael, et al. “Vehicle-To-Everything (V2X) Communication: A Roadside Unit for Adaptive Intersection Control of Autonomous Electric Vehicles.” ArXiv.org, 2024, arxiv.org/abs/2409.00866. Accessed 20 Jan. 2025.

[22] Raoul Kanschat, et al. “Wireless-Signal-Based Vehicle Counting and Classification in Different Road Environments.” IEEE Open Journal of Intelligent Transportation Systems, vol. 3, no. 90, 1 Jan. 2022, pp. 236–250, www.researchgate.net/publication/359683275_Wireless-Signal-Based_Vehicle_Counting_and_Classification_in_Different_Road_Environments, https://doi.org/10.1109/OJITS.2022.3160934.

[23] Nguyen, Justin, et al. “TrafficNNode: Low Power Vehicle Sensing Platform for Smart Cities.” IEEE, 2021.

[24] Martín, Juan, et al. “Traffic Monitoring via Mobile Device Location.” Sensors, vol. 19, no. 20, 17 Oct. 2019, p. 4505, https://doi.org/10.3390/s19204505. Accessed 23 May 2020.

[25] Vergis, Spiridon, et al. “A Low-Cost Vehicular Traffic Monitoring System Using Fog Computing.” Smart Cities, vol. 3, no. 1, 19 Mar. 2020, pp. 138–156, https://doi.org/10.3390/smartcities3010008.

[26] Wang, Kafeng, et al. “SenseMag: Enabling Low-Cost Traffic Monitoring Using Non-Invasive Magnetic Sensing.” IEEE Internet of Things Journal, vol. 8, no. 22, 2021, pp. 1–1, https://doi.org/10.1109/jiot.2021.3074907. Accessed 25 Apr. 2021.

[27] M. Won, S. Zhang, and S. H. Son, "WiTraffic: Low-cost and non-intrusive traffic monitoring system using WiFi," in 2017 26th International Conference on Computer Communication and Networks (ICCCN), July 2017, pp. 1–9. , https://doi.org/10.1109/icccn.2017.8038380. Accessed 25 Oct. 2023.

[28] Seid, Salahadin, et al. “A Low Cost Edge Computing and LoRaWAN Real Time Video Analytics for Road Traffic Monitoring.” IEEE Xplore, 1 Dec. 2020, ieeexplore.ieee.org/document/9394326. Accessed 2 Oct. 2022.

[29] Maus, Gerrit, and Dieter Brückmann. “A Non-Intrusive, Single-Sided Car Traffic Monitoring System Based on Low-Cost BLE Devices.” 2020 IEEE International Symposium on Circuits and Systems (ISCAS), Oct. 2020, pp. 1–5, ieeexplore.ieee.org/abstract/document/9181215, https://doi.org/10.1109/iscas45731.2020.9181215. Accessed 20 Jan. 2025.

[30] Ji, Wen, et al. “Trip-Based Mobile Sensor Deployment for Drive-by Sensing with Bus Fleets.” Transportation Research Part C: Emerging Technologies, vol. 157, 3 Nov. 2023, p. 104404, www.sciencedirect.com/science/article/abs/pii/S0968090X23003947, https://doi.org/10.1016/j.trc.2023.104404.

[31] Martuscelli, Giuseppe, et al. “V2V Protocols for Traffic Congestion Discovery along Routes of Interest in VANETs: A Quantitative Study.” Wireless Communications and Mobile Computing, vol. 16, no. 17, 21 Sept. 2016, pp. 2907–2923, https://doi.org/10.1002/wcm.2729. Accessed 29 Aug. 2021.

[32] Guastella, Davide Andrea, and Evangelos Pournaras. “Cooperative Multi-Agent Traffic Monitoring Can Reduce Camera Surveillance.” IEEE Access, vol. 11, 2023, pp. 142125–142145, ieeexplore.ieee.org/abstract/document/10360829/, https://doi.org/10.1109/access.2023.3343620. Accessed 20 Jan. 2025.

[33] Rahman, Abdul. “On-Demand Mobile Sensing Framework for Traffic Monitoring. (C2017).” Lau.edu.lb, 11 May 2017, laur.lau.edu.lb:8443/xmlui/handle/10725/6539, http://hdl.handle.net/10725/6539. Accessed 20 Jan. 2025.

[34] Yilmaz, Ozgun, et al. “Cloud-Assisted Mobile Crowd Sensing for Route and Congestion Monitoring.” IEEE Access, vol. 9, 2021, pp. 157984–157996, https://doi.org/10.1109/access.2021.3129932. Accessed 17 Feb. 2022.

[35] Ali, Qutaiba I. “Realization of a Robust Fog-Based Green VANET Infrastructure.” IEEE Systems Journal, vol. 17, no. 2, June 2023, pp. 2465–2476, https://doi.org/10.1109/jsyst.2022.3215845. Accessed 10 Nov. 2024.

[36] Ibrahim, Qutaiba. “An Efficient Power Management Strategy of a Solar Powered Smart Camera-Road Side Unit Integrated Platform.” International Journal of Sensors, Wireless Communications and Control, vol. 13, 24 Oct. 2022, https://doi.org/10.2174/22103279136662210241600809. Accessed 3 Dec. 2022.

[37] Ali, Q. I. (2008). An efficient simulation methodology of networked industrial devices. Proceedings of the 2008 5th International Multi-Conference on Systems, Signals and Devices (SSD), Amman, Jordan, 1–6, https://doi.org/10.1109/ssd.2008.4632835. Accessed 20 Jan. 2025.

[38] Ultralytics. “TensorRT.” Ultralytics.com, 2024, docs.ultralytics.com/integrations/tensorrt/?h=yolov8#what-are-the-performance-improvements-observed-with-yolov8-models-exported-to-tensorrt. Accessed 20 Jan. 2025.

[39] NVIDIA Corporation, "NVIDIA DGX A100 system datasheet," 2020. [Online]. Available: https://images.nvidia.com/aem-dam/Solutions/Data-Center/nvidia-dgx-a100-datasheet.pdf. Accessed: Jan. 20, 2025.

[40] GSMArena, "ZTE Blade A71 - Full phone specifications," [Online]. Available: https://www.gsmarena.com/zte_blade_a71-11240.php. Accessed: Jan. 20, 2025.

[41] GSMArena, "Samsung Galaxy A32 5G - Full phone specifications," [Online]. Available: https://www.gsmarena.com/samsung_galaxy_a32_5g-10648.php. Accessed: Jan. 20, 2025.

[42] GSMArena, "Samsung Galaxy S21 Ultra 5G - Full phone specifications," [Online]. Available: https://www.gsmarena.com/samsung_galaxy_s21_ultra_5g-10596.php. Accessed: Jan. 20, 2025.

[43] Ultralytics, "Using YOLOv8 with SAHI for Sliced Inference," [Online]. Available: https://docs.ultralytics.com/guides/sahi-tiled-inference. Accessed: Jan. 20, 2025.

[44] M. Muzammul, et al. “Enhancing UAV Aerial Image Analysis: Integrating Advanced SAHI Techniques with Real-Time Detection Models on the VisDrone Dataset.” IEEE Access, 1 Jan. 2024, pp. 1–1, https://doi.org/10.1109/access.2024.3363413. Accessed 2 June 2024.

[45] Akyon, Fatih Cagatay, et al. “Slicing Aided Hyper Inference and Fine-Tuning for Small Object Detection.” ArXiv:2202.06934 [Cs], 15 Feb. 2022, arxiv.org/abs/2202.06934.