IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 9 May 2025, accepted 5 June 2025, date of publication 4 July 2025, date of current version 4 August 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3585986

RESEARCH ARTICLE

MRTMD: A Multi-Resolution Dataset for Evaluating Object Detection in Traffic Monitoring Systems

MARK BUGEJA, MATTHIAS BARTOLO, (Graduate Student Member, IEEE), MATTHEW MONTEBELLO, (Senior Member, IEEE), AND DYLAN SEYCHELL, (Senior Member, IEEE)
Department of Artificial Intelligence, University of Malta, 2080 Msida, Malta
Corresponding author: Mark Bugeja (mark.bugeja@um.edu.mt)

ABSTRACT Traffic monitoring reduces congestion, improves safety, and supports environmental sustainability. Real-time flow tracking, anomaly detection, and efficient management are key. Convolutional Neural Networks (CNNs) have become integral due to their compact size and easy deployment. However, their effectiveness depends heavily on the quality of the input data, especially image resolution. With high-resolution cameras, especially 4K, balancing image quality, detection accuracy, and system efficiency is critical. We propose the Multi-Resolution Traffic Monitoring Dataset (MRTMD), which captures transport scenes at resolutions ranging from 2160p to 360p. This dataset serves as a benchmark for standard object detection models, enabling the development of more efficient and cost-effective traffic monitoring solutions. MRTMD will be freely available on GitHub, offering a valuable resource for researchers and practitioners. We evaluate leading object detection models—YOLOv9, YOLOv8, YOLOv7, Faster R-CNN, FCOS, SSD, and RT-DETR—across varied resolutions. Our analysis focuses on mean Average Precision (mAP), recall, and processing time. We also assess the accuracy of Number Plate Recognition (NPR) for tasks that require fine-grained detail extraction. Our findings show that detection performance typically varies within ± 0.01 to ± 0.03 in mAP and recall across resolutions, suggesting higher resolutions are not always advantageous. However, they remain crucial for tasks like NPR. The multi-resolution dataset enables a comprehensive evaluation of the trade-off between image quality and task performance. Ultimately, our analysis highlights the importance of resolution selection in large-scale deployments, informing system designers and policymakers. This dataset is a vital tool for balancing performance, cost, and practical constraints in real-world traffic monitoring.

INDEX TERMS Dataset vehicle detection, high resolution image dataset, number plate recognition, computer vision.

I. INTRODUCTION

Over the past decade, extensive research has explored Computer Vision (CV) and Artificial Intelligence (AI) in urban traffic management systems, demonstrating the need for digital innovation in transport infrastructure. With increasing urban populations and vehicular traffic, the need for efficient and effective traffic management systems has never been more critical. High-resolution video technology, particularly 4K video, has emerged as a promising tool that might improve traffic monitoring and control by providing detailed and high-quality video feeds [1], [2], [3], [4]. Computer vision, particularly object detection, has witnessed remarkable advancements in recent years, primarily driven by the development of deep learning techniques [5]. CNNs have become the cornerstone of modern object detection systems, demonstrating superior performance in vehicle identification, pedestrian detection, and traffic sign recognition [6], [7]. The compact nature and ease of deployment of CNNs make them particularly well-suited for integration into Intelligent Transportation Systems (ITS), enabling real-time monitoring and decision-making capabilities [8]. The impact of 4K

The associate editor coordinating the review of this manuscript and approving it for publication was Jason Gu.

resolution processing is significant across various sectors, offering enhancements in detail and clarity. In healthcare, it allows for more precise imaging, which is crucial for accurate diagnoses and surgical planning [9]. It enhances surveillance capabilities in the security sector, providing sharper images that facilitate better identification and situational awareness [10]. Similarly, 4K enhances the viewer experience by providing superior picture quality in media and entertainment [11]. While the need for 4K and higher resolution is evident, its use in CV and AI is mainly limited due to computational resources. While it offers finer details in video feeds, its utility in everyday scenarios may not justify the high resource and energy consumption required. It suggests a potential re-evaluation of its application in fields requiring large-scale video and image processing [12].

Despite these advancements, the efficacy of object detection models in traffic monitoring scenarios heavily depends on the quality and diversity of training data [13]. While several datasets focus on vehicle detection and traffic monitoring, such as KITTI [14], there is a notable scarcity of high-resolution datasets for vehicle detection that are not based on aerial imagery. This gap in available data presents a challenge in developing and evaluating object detection models that can leverage the increased detail provided by high-resolution imagery, particularly in the context of ground-level traffic monitoring [15]. To address this critical limitation, we introduce the Multi-Resolution Traffic Monitoring Dataset (MRTMD), a novel and comprehensive collection of transport scenes captured at multiple resolutions, from 4K (2160p) down to 360p. The MRTMD is designed to serve as a benchmark for comparing the performance of common object detection models across various resolutions, thereby enabling the development of more efficient and cost-effective solutions for traffic monitoring. This dataset not only fills a crucial gap in the field but also provides a unique opportunity to investigate the impact of image resolution on detection accuracy, computational requirements, and overall system performance [16]. The primary objective of this study is to evaluate the necessity and efficacy of utilising 2160p resolution in traffic monitoring systems. We hypothesise that processing traffic data at resolutions lower than 2160p can achieve comparable accuracy and effectiveness while significantly reducing resource consumption and energy requirements. Specifically, we propose: Additionally, this study aims to create a dataset that can be used to determine the optimal balance between image quality and task performance, taking into account factors such as detection accuracy, computational costs, and practical constraints in large-scale deployments [17].

To assess the impact of resolution on object detection performance, we employ several state-of-the-art models, including YOLOv9 [18], YOLOv8 [19], YOLOv7 [20], Faster R-CNN [21], FCOS [22], SSD [23], and RT-DETR [24]. These models are evaluated using the MRTMD across all available resolutions, with performance measured through metrics such as mean Average Precision (mAP), recall, and processing time. Additionally, we conduct a focused assessment of NPR accuracy across different resolutions, providing valuable insights into tasks that require fine-grained detail extraction [25]. The implications of this study extend beyond academic interest, holding significant potential to influence the design and implementation of real-world traffic monitoring systems. By precisely determining the resolution at which performance plateaus or degrades, we aim to provide crucial guidance for decision-makers in selecting appropriate hardware and software configurations for large-scale traffic monitoring deployments. This research contributes to the ongoing discourse on optimising the use of technology in enhancing traffic management systems, striking a balance between performance requirements and practical considerations of cost, energy efficiency, and scalability [26], [27]. By investigating the potential for lower-resolution systems to match the performance of their high-resolution counterparts, we open avenues for more environmentally friendly and cost-effective solutions in urban traffic management [28]. The structure of this paper is as follows: We begin with a comprehensive background section that details the evolution of object detection techniques and surveys relevant datasets in the field of traffic monitoring. This is followed by a detailed methodology section, which outlines the construction and analysis of the MRTMD, as well as our experimental setup for evaluating various object detection models. We then present our results, analysing model performance across different resolutions. The discussion section interprets these findings in the context of real-world applications and explores their implications for the future of traffic monitoring systems. Finally, we summarise our key findings and suggest directions for future research in this rapidly evolving field. Through this investigation, we aim to bridge the gap between theoretical advancements in CV and practical implementations in traffic monitoring, paving the way for more efficient, effective, and sustainable urban transportation systems.

II. BACKGROUND

A. VEHICLE DETECTION TECHNIQUES

Object detection, a fundamental area in CV, has revolutionised various industries, including medical diagnostics [29], [30], [31], litter detection [32], [33], [34] and autonomous vehicles [35]. In particular, vehicle detection serves as a critical task in traffic monitoring and advanced driver-assistance systems, demanding robust methods to handle variable lighting, weather, and complex roadway scenarios [36], [37], [38].

Substantial research and development have enabled computing algorithms to locate and categorise objects within photos or video frames with precision [39]. Breakthroughs in machine learning and deep learning have further driven this transformation.

The evolution of object detection techniques can be broadly categorised into three phases: traditional methods, machine learning (ML) approaches, and deep learning-based detectors.

1) TRADITIONAL METHODS

The earliest object detection techniques relied on relatively inventive yet straightforward strategies and efficient computing processes. For instance, background subtraction [40] leveraged pixel-level changes to isolate moving objects from the static background. More advanced feature-based algorithms, such as Haar cascade classifiers [41], [42], employed Haar-like features to capture edge-like image patterns—initially popular for facial detection but later adapted for vehicle detection [43]. Similarly, the Histogram of Oriented Gradient (HOG) descriptor [44], [45] gained traction for pedestrian detection and was also extended to identify vehicles [46]. Other classical methods, including edge-detection filters [47] and background subtraction variants [48], further laid the foundation for advanced vehicle-detection approaches. Although these early systems showcased ingenuity, they typically required extensive fine-tuning and often struggled to perform reliably under shifts in lighting, weather conditions, vehicle types, and occlusions. Consequently, the advent of deep learning propelled vehicle-detection accuracy and robustness to new heights [49].

2) MACHINE LEARNING APPROACHES

Significant advancements in object detection resulted from the implementation of ML principles. Support Vector Machines (SVMs) [50] became an essential part of object detection, providing a more robust classification framework when trained on feature descriptors such as Histogram of Oriented Gradients (HOG) [51]. Other ML techniques also contributed to the field, including boosting algorithms like AdaBoost, used in the Viola-Jones face detection framework [52], and ensemble methods such as Random Forests [53] or RGB-D information [54]. For instance, hybrid approaches that combine Viola-Jones with HOG and SVM have been successfully applied to vehicle detection from UAV images [55]. At the same time, RGB-D-based methods further improved reliability in vehicle-following systems [56]. Nevertheless, these techniques still relied on hand-engineered features. Deformable Part Models (DPM) [57] further advanced the field by treating objects as collections of parts in a deformable configuration, effectively handling variable object appearances. These ML approaches significantly enhanced the accuracy and robustness of object detection systems, paving the way for the subsequent Deep Learning (DL) revolution in the field.

3) DEEP LEARNING-BASED DETECTORS

One of the most significant turning points in the history of object detection pertained to the introduction of deep learning and, more specifically, CNNs [58]. Laborious hand-crafted feature engineering became largely unnecessary, as CNNs could automatically learn features from large volumes of data [59]. This paradigm shift enabled the development of a new generation of more accurate and resilient object detectors, which can be broadly categorised into two-stage and one-stage detectors. More recently, transformer architectures—originally developed for natural language processing—have been adapted for vision tasks, such as object detection, further pushing performance boundaries [24], [60]. Other emerging approaches integrate reinforcement learning-based visual attention methods [61], [62] with saliency ranking techniques [63], allowing models to iteratively refine the detection and localisation of objects [64].

In the specific context of vehicle detection, deep learning has proven highly effective in handling varying conditions and viewpoints. Techniques that fuse 3D-LIDAR data with deep CNNs, for instance, have demonstrated improved performance in complex traffic scenarios [65]. Comparative studies of CNN-based detectors further highlight their robustness across diverse real-world conditions [66]. End-to-end pipelines can significantly mitigate issues such as occlusion or environmental noise [67]. Moreover, methods that combine saliency signals with CNN have shown promise in low-visibility situations, such as vehicle detection at night [68], reinforcing the role of attention-driven features in enhancing detection accuracy.

4) TWO-STAGE DETECTORS

Convolutional Neural Network (CNN) feature extractors have profoundly impacted object detection models such as R-CNN [69], Fast R-CNN [70], and Faster R-CNN [21]. These approaches follow a two-stage pipeline: first, they generate a set of region proposals that potentially contain objects, then a second stage uses CNN-based classifiers and bounding-box regressors to refine and categorise these proposals. This methodology often achieves higher accuracy but typically requires more computational resources, making it less suitable for real-time deployments on limited hardware. Nevertheless, targeted adaptations of these detectors have proven highly effective for vehicle detection, yielding robust results in tasks such as autonomous driving [71], [72], [73]. Recent comparisons of autonomous driving benchmarks also confirm that two-stage networks, such as Faster R-CNN, often outperform one-stage alternatives in terms of precision and handling minority classes, albeit with increased computational overhead [74]. In a similar vein, improved sparse R-CNN approaches have been proposed to tackle traffic sign detection, demonstrating that two-stage frameworks can excel even in complex tasks required by autonomous vehicles [75].

5) ONE-STAGE DETECTORS

In contrast, one-stage detectors merge proposal generation and bounding-box regression into a single step, directly predicting object locations and classification scores from a dense sampling of feature maps. Notable examples include SSD [23], YOLO [7], [20], [76], [77], [18], and FCOS [22]. By eliminating the need for a separate proposal step, one-stage architectures significantly reduce inference time, making them particularly attractive for real-time applications. However, they often display lower accuracy than their two-stage counterparts, particularly in scenes

with heavy clutter or when high precision is paramount. SSD utilises predefined anchor boxes (varying by scale and aspect ratio), whereas YOLO eliminates the need for region proposals by directly predicting class probabilities and bounding boxes. FCOS likewise eschews anchor boxes, framing detection as a point-based regression problem. Recently, TSD-YOLO introduced an enhanced YOLO v8 pipeline designed explicitly for small traffic sign detection, highlighting the ongoing evolution of one-stage detectors for traffic applications [78]. Although one-stage detectors can be computationally more efficient, scenarios requiring higher precision—such as specific autonomous driving tasks—may still benefit from two-stage frameworks or more advanced backbone networks [74].

a: CHOOSING BETWEEN ONE-STAGE AND TWO-STAGE METHODS

The decision often hinges on striking a balance between speed and accuracy. Two-stage detectors generally excel in precision, albeit at the expense of a higher computational load and slower throughput. One-stage frameworks, on the other hand, are optimised for real-time speed but may struggle in highly cluttered scenes or when minority classes demand special attention. Consequently, the best choice depends on the specific requirements for hardware constraints, throughput, and performance targets in a given application.

b: TRANSFORMERS

More recently, transformers have become increasingly popular techniques for object detection. The resilient architectures initially developed for natural language processing are currently being investigated for their potential applications in CV tasks such as object detection. DETR, which refers to ''DEtection TRansformer'' is a revolutionary approach to object detection that utilises the transformer encoder-decoder architecture coupled with bipartite matching to optimise the detection pipeline. By treating the object detection problem as a direct set prediction problem, Carlon et al. in [60] attempt to reduce the need for several hand-designed components, such as non-maximum suppression methods or anchor box generation. Leveraging the self-attention mechanisms in the encoder-decoder architecture, which explicitly describes all pairwise interactions between components in a sequence, makes the transformer architecture especially well-suited to address certain limitations for set prediction limitations, such as deleting duplicate predictions. The DETR architecture uses a CNN backbone to extract a 2D image representation, flattened and positionally encoded for the transformer encoder. The transformer decoder then processes learned object queries, predicting detections or a ''no object'' class through a shared feed-forward network. Building upon DETR, RT-DETR (Real-Time DEtection TRansformer) [24] introduces optimisations like query selection and dynamic feature aggregation, significantly reducing computational overhead while sustaining high accuracy. These refinements make RT-DETR feasible for low-latency scenarios, including real-time deployments. Further applications of transformer-based models to vehicle detection—such as using an improved RT-DETR [79] or fine-tuning DETR for specialised settings [80] underscore the versatility and growing importance of transformer architectures in practical traffic applications.

c: CHALLENGES IN HIGH-RESOLUTION REAL-TIME DETECTION

Contemporary object detection models are commonly evaluated using well-known benchmarks such as COCO and Pascal VOC datasets. These datasets typically include images of various resolutions. However, it is common practice to reduce the resolution of high-resolution images to match the network's input size. This down-sampling process can result in the model overlooking important details for detecting smaller objects. Ongoing research investigates several methods to tackle this issue, such as creating models that can naturally manage higher resolutions or utilising techniques like multi-scale feature extraction. Despite significant advancements in processing speed, achieving real-time performance for sophisticated object identification on various hardware platforms remains challenging. Low latency is essential for applications such as self-driving cars, where minimal delay is necessary and could potentially cost lives. Over the years, ongoing research has consistently enhanced deep learning architectures to improve efficiency and investigate hardware acceleration approaches. These involve employing specialised hardware, such as TPUs, to expedite computations and obtain faster detection speeds. It is, therefore, essential to determine the necessity of processing higher-quality images to address a problem or an opportunity to find new solutions.

III. SURVEY OF RELEVANT DATASETS

This section presents several relevant datasets for this study. It focuses on datasets that provide images used for classification, recognition, and other related CV tasks applied to the transport domain. The first section focuses on transport management datasets. This paper also investigates the impact of high-resolution videos and the performance of CV models on these videos. For this reason, this section also presents a survey of general datasets that include 4K resolution content. These were mainly covered since critical CV models were also trained on this data.

A. TRANSPORT MANAGEMENT DATASETS

1) BIT VEHICLE DATASET OVERVIEW

The BIT Vehicle Dataset is particularly notable for its application in ITS vehicle type classification (VTC) [81]. It encompasses six distinct vehicle classes, including buses, microbuses, minibuses, SUVs, sedans, and trucks, with a total of 900 vehicles featured. This dataset has been utilised in various appearance-based tasks such as speed estimation, illegal vehicle detection, and traffic flow analysis. Despite

its extensive use, one major challenge with the BIT Vehicle Dataset is the substantial time and effort required to prepare and pre-process the data, which can significantly slow down research progress. It achieves a classification accuracy of 0.938, demonstrating its robustness in capturing discriminant vehicle features under various illumination conditions, time, and scale.

2) COMPCARS DATASET OVERVIEW

The Comprehensive Cars (CompCars) [82] Dataset contains a mix of web-nature and surveillance-nature images and is known for its wide application in real-world tasks since its launch in 2015. The web-nature component comprises 136,727 images capturing complete cars and 27,618 parts of cars, each with annotated viewpoints and labels. The surveillance-nature component includes 44,481 images typically captured from the front and annotated with bounding boxes, model, and colour details. This dataset is distinct in offering insights into car hierarchy, viewpoints, attributes, and parts, making it a valuable resource for advanced vehicle recognition systems.

3) KITTI BENCHMARK DATASET OVERVIEW

The KITTI Benchmark Dataset [14] is extensively used in autonomous vehicle scenarios and includes modalities such as high-resolution RGB images, 3D laser scans, and greyscale stereo images. Although it lacks native segmentation ground truth, many researchers have manually annotated images to suit specific experimental needs. For instance, Alvarez et al. [83] provided ground truth for 323 images in road detection tasks. In contrast, Zhang et al. [84] annotated 252 images with detailed class information for more comprehensive object recognition.

4) STANFORD CAR DATASET OVERVIEW

Launched in 2013, the Stanford-Car Dataset [85] includes over 16,000 images split between training and unseen sets, featuring 196 different car types. This dataset has been pivotal in advancing 3D object recognition technologies, particularly in enhancing fine-grained categorisation tasks by allowing for detailed 3D geometry estimations from the images.

5) MotorBike7500 AND MotorBike10000 DATASETS OVERVIEW

The MotorBike7500 and MotorBike10000 datasets [86] serve as benchmarks for motorcycle detection, comprising 7,500 and 10,000 images, respectively. These datasets are characterised by their real-time traffic scene captures and high occlusion rates, making them challenging for object detection models. Both datasets have been instrumental in testing the efficacy of various detection schemes, consistently showing high performance.

6) EMERGING CHALLENGES AND DATASET UTILITY

As vehicle detection advances, the quality and diversity of datasets such as those mentioned play a pivotal role in developing robust models. However, challenges such as ensuring sufficient variability in dataset conditions and improving annotation accuracy remain critical. Future work in vehicle detection will likely focus on expanding dataset capabilities and exploring innovative model architectures that can better handle the complexities of real-world environments.

B. HIGH RESOLUTION DATASETS

1) COCO DATASET OVERVIEW

The COCO dataset (Common Objects in Context) [87] serves as an essential benchmark for researchers and developers pushing the boundaries of object detection and related CV tasks. This comprehensive dataset boasts a staggering collection of over 330,000 images, with more than 200,000 meticulously labelled to provide information for training and evaluating algorithms. Each image goes beyond a simple snapshot, offering a glimpse into everyday scenes brimming with objects from 80 distinct categories. Cars, people, and bicycles are just a few examples, with more specific entries like aeroplanes and kitchen utensils enriching the dataset's diversity. But COCO doesn't stop at bounding boxes and class labels for these 1.5 million object instances. It delves deeper, providing additional annotations that unlock the potential for tasks beyond essential detection. Each image is accompanied by five captions describing the scene and the relationships between objects. This makes it a valuable resource for tasks like image retrieval and captioning. For a subset of images featuring people, COCO includes keypoint annotations that pinpoint crucial body parts, such as joints and limbs. This data proves invaluable for tasks like human pose estimation, where the CV system needs to understand the posture and pose of a person in an image. The impact of the COCO dataset extends far beyond its role as a rich data source. By offering a standardised platform for evaluation, COCO has become a de facto standard for comparing the performance of object detection models. Researchers and developers can leverage this common ground to objectively assess their algorithms, fostering advancements and pushing the boundaries of what's possible in CV. Beyond object detection, the richness and diversity of COCO make it a valuable tool for exploring other areas of CV research. The additional annotations, like captions and key points, unlock its potential for tasks like image retrieval, image captioning, and human pose estimation. While the sheer size of COCO can present computational challenges, particularly for training resource-intensive deep learning models, and the class distribution might not be perfectly balanced, the overall value of this dataset remains undeniable. Self-supervised approaches were also explored for automating annotations in such datasets [88]. It stands as a cornerstone for researchers and developers, propelling the field of CV forward and paving the way for even more sophisticated and versatile applications in the future [89].

2) DLR 3K DATASET OVERVIEW

DLR 3K Dataset often referred to as DLR Munich [90] is a high resolution 5616 $\times$ 3744 pixel nadir images taken in

Munich, Germany. The dataset contains 20 images, although this dataset can be considered high-resolution. In reality, each image is taken at an altitude of 1000 meters above ground with an approximate ground sampling distance of 13 cm. This vehicle dataset is used in traffic monitoring. In their research, Liu and Mattyus [90] also subdivided the dataset into 180 images, converting each image into nine sections and lowering the resolution of the training and testing set to 1872 × 1248 pixels, which approximately fits more with 1080p than 2K resolution or higher.

3) GIGaPixel DATASET OVERVIEW

PANDA (gigaPixel-level humAN-centric viDeo dAtaset) [91] revolutionises object detection by offering high-resolution, large-scale video data with a human-centric focus. Unlike COCO's focus on individual images, PANDA delves into gigapixel resolution videos, capturing vast areas with meticulous detail – ideal for object detection at an unprecedented scale. Boasting a wide field of view encompassing up to 1 square kilometre in a single frame, PANDA allows researchers to analyse large-scale human activity within a scene, providing valuable insights into crowd behaviour and interactions. Despite its vast scope, PANDA doesn't sacrifice detail. The gigapixel resolution ensures that even small objects and subtle human actions remain discernible, making it ideal for tasks that require precise identification.

Further enriching PANDA's value are its comprehensive annotations. Thousands of bounding boxes pinpoint humans and objects of interest, while attribute labels, such as pose, clothing, or carried objects, provide a deeper understanding of the scene. PANDA even tracks human movement across frames (trajectories), offering insights into crowd flow and behaviour over time. In some cases, group and interaction annotations capture the dynamics between people in large public spaces. These rich annotations unlock numerous applications: PANDA can be used to develop intelligent systems for crowd management, anomaly detection, and public safety monitoring in smart cities. Urban planning and infrastructure design can benefit from analysing pedestrian and traffic flow patterns derived from PANDA. Additionally, studying human behaviour in large crowds using PANDA offers valuable insights into social dynamics and interactions. As advancements in data storage, processing, and automated annotation techniques are made, PANDA paves the way for advancements in large-scale, long-term object detection, particularly for human-centric activities, holding immense promise for revolutionising object detection tasks across various fields.

While the PANDA dataset boasts impressive gigapixel resolution and vast field-of-view, it wouldn't be the optimal choice for solely focusing on vehicle detection. PANDA's design prioritises human-centric activities, meaning rich annotations, such as bounding boxes, attribute labels, and even group interactions, are geared towards people, not vehicles. This inherent human focus creates a data imbalance – vehicles would be a much smaller fraction of objects compared to pedestrians. Training solely on PANDA could lead to object detection models that struggle with vehicle detection tasks, which require a wider variety and higher number of vehicle examples for robust performance. Furthermore, PANDA's high resolution comes at the cost of detail for individual objects, especially vehicles that might occupy a smaller portion of the frame than pedestrians. This can make it challenging for models to accurately detect and classify vehicles, particularly smaller ones or those partially obscured. Unfortunately, there is currently a lack of high-resolution datasets designed explicitly for vehicle detection that can rival PANDA's large-scale approach. This highlights a gap in the field—while datasets like COCO offer a good mix of object categories, including vehicles, their resolution might not be sufficient for tasks requiring intricate detail in vehicle detection. Ideally, a future dataset could combine both strengths, offering the high resolution of PANDA with a focus on vehicle annotations and a wider variety of vehicle types to address the limitations of current options.

4) TJU-DHD

The TJU-DHD (Tianjin University Diverse High-resolution Dataset) introduced by Pei et al. [92] presents a high-resolution object detection dataset comprising images of traffic and pedestrians. This dataset addresses the scarcity of large-scale, high-resolution datasets for object detection in real-world scenarios, including traffic monitoring. TJU-DHD comprises over 115,000 high-resolution images with over 700,000 annotated object instances across ten categories. The dataset's resolution range spans from 1624 × 1200 to 2560 × 1440 pixels, providing a set of images for studying how resolution affects detection performance. Though it does not extend to 4K resolution, this range still represents higher resolutions than many commonly used datasets, making it relevant for research into the effects of increased image resolution on object detection performance. The subset of images in the dataset that deals with traffic monitoring, referred to as TJU-DHD-Traffic, is pegged at 1624 × 1200 pixels resolution. TJU-DHD primarily comprises real-world scenes and challenging cases, including dense object distribution, scale variation, and partial occlusion. These characteristics make it relevant for traffic monitoring applications. The dataset has benchmarked several state-of-the-art object detection algorithms, providing valuable insights into their performance on high-resolution images. While TJU-DHD offers a comprehensive resource for high-resolution object detection, our MRTMD dataset differs in its specific focus on traffic monitoring scenarios, its systematic approach to resolution variation, and its inclusion of even higher resolutions up to 2160p. MRTMD provides the same scenes at multiple resolutions, allowing for a more controlled study of resolution impact on detection performance in traffic-specific contexts, especially at high resolutions.

TABLE 1. Comparison of datasets with traffic-related images and objects. MRTMD dataset contains 3733 images with 70,506 objects per resolution.

Name	#Images	Resolution	#Objects	Average	Categories
TJU-DHD-Traffic	45,266	1624x1200	239,980	5.3	pedestrian, rider, car, truck, van
DLR 3K	20	5616x3744	5,892	294.6	cars, truck
COCO (traffic images)	73,050	varies, median of 640x480	352,535	4.8	persons, motorcycles, trucks, cars, bicycles, buses
PANDA	600	25000x14000	111,000	185.0	persons
BIT Vehicle	9,850	1600x1200, 1920x1080	10,053	1.0	bus, microbus, minibus, suv, sedan, trucks
CompCars	136,727	640x480, 1024x768	27,618	0.2	1,716 car models
KITTI Benchmark	14,999	≈1240x374	51,865	3.5	car, van, pedestrian, cyclist, truck, misc, tram, person sitting
Stanford Car	16,185	276x182	16,185	1.0	196 car classes
Motorbike 7500	7,500	640x364	41,040	5.5	motorcycles
Motorbike 10000	10,000	640x365	56,975	5.7	motorcycles
MRTMD (Ours)	3,733	3840x2160 2560x1440 1920x1080 1280x720 854x480 640x360	70,506	18.9	persons, motorcycles, trucks, cars, bicycles, buses

IV. MATERIALS AND METHODS

A. OVERVIEW

This section outlines the comprehensive approach to evaluating the effectiveness of high-resolution video in traffic monitoring systems. The primary focus is developing and analysing the Multi-Resolution Traffic Monitoring Dataset (MRTMD), which is designed to support research in high-resolution video processing, particularly 2160p resolution, within traffic data collection and analysis. The dataset construction process is detailed, highlighting the collection of 2160p traffic videos from publicly available sources and their systematic downscaling to multiple lower resolutions. This enables a thorough examination of the trade-offs between different resolutions in terms of accuracy and efficiency in traffic monitoring applications. The dataset is meticulously annotated, emphasising frames featuring number plates, to assess the potential benefits of high-resolution imagery in NPR systems. Subsequent analysis of the dataset includes a variety of metrics that explore the distribution, size, and density of objects within the images, providing a deep understanding of the dataset's characteristics. These metrics are visualised through figures that illustrate the diversity and scale variability of objects, which are crucial for training and evaluating object detection models. The evaluation methodology involves processing the MRTMD dataset through multiple state-of-the-art object detection models at all available resolutions. Performance metrics, including mean Average Precision (mAP), precision, recall, and computational complexity, are used to assess each model's effectiveness. Additionally, the accuracy of NPR is evaluated using specialised metrics, such as Levenshtein distance, Character Error Rate (CER), and Word Error Rate (WER). Finally, the section compares the performance of various object detection models, examining their trade-offs between accuracy, computational cost, and resolution. This comprehensive analysis aims to determine whether using 2160p resolution offers substantial benefits in traffic monitoring applications, thereby providing valuable insights into the optimal use of technology in enhancing traffic management systems.

B. THE DATASET

This section presents the construction of the Multi-Resolution Traffic Monitoring Datasets (MRTMD).[1] This dataset is designed and built to support studies that handle high-resolution content. This paper aims to evaluate whether the high resolution of 2160p video offers tangible benefits over lower resolutions in traffic data gathering systems using CV technologies. We hypothesise that processing traffic data at resolutions lower than 2160p can achieve comparable accuracy and effectiveness in traffic monitoring and analysis while reducing resource and energy consumption. To our knowledge, no existing public dataset provides co-registered 2160p-to-360p image pairs of the same traffic scenes. MRTMD therefore enables the first empirical study of resolution trade-offs without confounding viewpoint, illumination, or annotation variance.

1) DATASET DESCRIPTION

The dataset used comprises a unique curated collection of 2160p traffic monitoring videos sourced from YouTube, covering various traffic scenarios and perspectives such as overhead views, IP camera angles, and cam recordings, as seen in Fig. 1. Each video in the dataset is available in multiple resolutions (2160p, 1440p, 1080p, 720p, 640p, and 360p). This enables systematic testing across a range of image qualities, allowing for direct assessment of the trade-offs between resolution and system performance. All videos in the dataset are covered by a Creative Commons license, permitting their use in this research. The open license of these resources ensures that the study adheres to legal and ethical standards regarding the use and dissemination of content.

[1] https://github.com/markbugeja/Multi-Resolution-Traffic-Monitoring-Dataset-MRTMD-

FIGURE 1. A selection of frames showing camera angles from different frames extracted from the videos used in the MRTMD dataset.

2) DATASET COMPARISON

Table 1 illustrates several different traffic datasets that can be compared with MRTMD datasets. Unlike larger datasets like TJU-DHD-Traffic and COCO, MRTMD comprises 3,733 images with 70,506 annotated objects. The dataset offers a range of resolutions, from 3840 × 2160 pixels to 640 × 360 pixels, containing the same number of images and annotations to facilitate benchmarking object detection model robustness across different resolutions. This range allows for a detailed evaluation of the impact of resolution on detection accuracy and computational efficiency. Other high-resolution datasets, such as PANDA and DLR 3K, are often processed into smaller images, thereby reducing the significance of their high resolution. MRTMD's average of 18.9 objects per image is higher than that of most other datasets, except for DLR 3K and PANDA, which are limited to cars, trucks, and persons, respectively. MRTMD also covers a broad spectrum of categories, including persons, motorcycles, trucks, cars, bicycles, and buses, offering greater versatility than datasets like DLR 3K and PANDA. While smaller in size compared to datasets like COCO or CompCars, MRTMD's detailed annotations and emphasis on high-resolution images make it a valuable resource for research that requires both resolution and object density.

3) KEY DISTINCTIONS FROM EXISTING DATASETS

Unlike many existing traffic-oriented datasets that either provide a single resolution or inconsistent viewpoints, MRTMD captures the *same* scenes at multiple resolutions (2160p, 1440p, 1080p, 720p, 480p, and 360p) while maintaining consistent annotations. This multi-resolution setup enables controlled experiments on how resolution impacts detection accuracy and computational cost, an aspect often overlooked in other collections where images vary in viewpoint, framing, or weather conditions. Additionally, the average object density in MRTMD (18.9 objects per image) is higher than in most other traffic datasets, providing more complex scenes for benchmarking. While some large-scale datasets (e.g., COCO, KITTI) contain diverse images, they generally do not offer a systematic way to assess resolution-based performance trade-offs. By contrast, MRTMD's focus on multi-resolution captures and dense annotations makes it a valuable resource for researchers investigating how hardware limitations, image quality, and overall efficiency intersect in real-world traffic-monitoring contexts.

a: OBJECTS PER IMAGE

The dataset comprises 70,506 annotated objects in bounding box format across six categories: cars (n = 31,573), persons (n = 21,361), motorcycles (n = 15,544), trucks (n = 1,627), buses (n = 337), and bicycles (n = 64). Object sizes exhibit considerable variation, with an average area of 26,984.93 pixels^2 (SD = 104,939.45 pixels^2) and a median of 9,505.68 pixels^2. This substantial difference between the mean and median, coupled with the significant standard deviation, suggests a right-skewed distribution of object sizes. Objects were classified into three size categories: large (n = 36,040), medium (n = 27,946), and small (n = 6,520). The prevalence of larger objects and the vast size distribution indicate a diverse dataset capturing objects at various scales, potentially suitable for training robust object detection models across different real-world scenarios. The annotation process involved three

TABLE 2. Count of objects per category, the dataset is well distributed across the cars, persons and motorcycles categories. Truck, bus and bicycle counts are lower by comparison, but this represents a typical bias in traffic scenarios.

Object	Count
Car	31,573
Person	21,361
Motorcycle	15,544
Truck	1,627
Bus	337
Bicycle	64

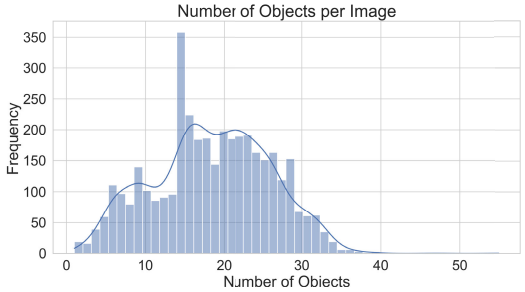

FIGURE 2. Number of objects per image.

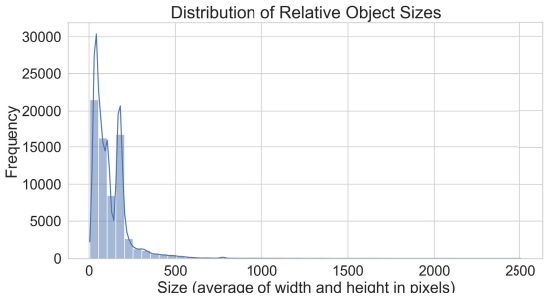

FIGURE 3. Distribution of relative object sizes, the size variation is taken from the ground truth of the 2160p images, nonetheless the same distribution is present in the lower variations of the same dataset.

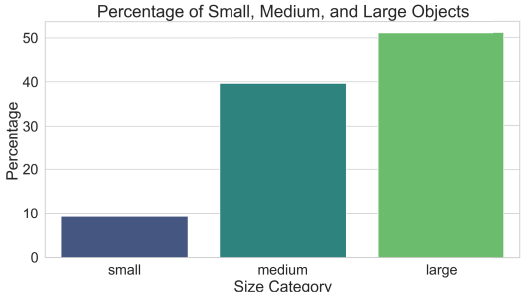

FIGURE 4. Size category distribution.

independent annotators to ensure accuracy and consistency. All three annotators individually labeled each image using the Roboflow annotation tool, adhering to the COCO annotation format, a widely adopted standard in the CV community. This format includes bounding box coordinates, object category labels, and segmentation masks where applicable. Discrepancies between annotators were resolved through a consensus review process, with a senior annotator making final decisions in cases of persistent disagreement. This rigorous approach to annotation, combined with the COCO standard, enhances the dataset's reliability and facilitates its integration with existing object detection and instance segmentation frameworks. The videos have been segmented into frames, and significant attention was paid to frames that prominently feature number plates. These frames underwent an extrapolation process to enhance the visibility of number plates, facilitating a focused evaluation of high-resolution benefits for NPR. This aspect of the dataset aims to directly assess the capabilities of high-resolution imaging in improving the accuracy and reliability of NPR systems. Extensive annotation efforts were undertaken to label the vehicles, pedestrians, and number plates within the frames. An extrapolation process was conducted for clearly visible number plates to isolate and enhance these details. This procedure enriches the dataset with high-value frames for NPR. It allows for a nuanced examination of the advantages of higher resolution in extracting and interpreting fine details critical for accurate automated recognition tasks.

4) DATASET ANALYSIS

a: OBJECTS PER CATEGORY

The distribution of objects across the eight categories is shown in Table 2. This table indicates the frequency of each object type, highlighting the predominance of cars and persons in the dataset.

b: OBJECTS PER IMAGE

Figure 2 illustrates the number of objects per image. This distribution shows the variation in object density, from images with few objects to those with many. For a typical image on the MRTMD dataset, the number of objects is around 14. Some images, especially in videos 1 and 3, can have as many as 30 objects. The average number of objects per image hovers around 18.9.

c: DISTRIBUTION OF RELATIVE OBJECT SIZE

Figure 3 presents the distribution of relative object sizes, defined as the ratio of the object's area to the image area. This measure provides insights into the scale variability of objects within the dataset.

d: SIZE CATEGORY DISTRIBUTION

Following the COCO dataset definitions, objects are categorised into three size categories: small, medium, and large. Figure 4 shows the distribution of objects across these size categories.

This enriched dataset, focusing on high-resolution video and detailed annotations, provides a robust foundation for analysing the efficacy of different resolutions in traffic monitoring applications. The findings derived from this dataset will significantly contribute to the ongoing discourse on the optimal use of technology in enhancing traffic management systems. The dataset includes:

- 3733 unique images.
- 24,372 images when accounting for six different resolutions.
- Annotations for six object categories.
- Detailed statistics on object distribution, size, and density.

This dataset provides a valuable resource for developing and evaluating object detection and traffic monitoring systems. Its comprehensive annotations, multiple resolutions, and detailed statistics ensure it is well-suited for various research and application scenarios in CV.

5) LIMITATIONS OF THE DATASET

MRTMD is a valuable resource for traffic analysis and object detection across various resolutions, yet it has notable limitations that restrict its broader applicability. With a relatively small size of 3,733 images and 70,506 annotations, the dataset is considerably smaller than benchmark datasets such as COCO or ImageNet. This limited scale reduces its utility for training modern deep-learning models from scratch. The dataset also exhibits category imbalances; for example, cars and persons are overrepresented in the annotations, while trucks, bicycles, and buses are underrepresented. This imbalance introduces bias during model training, which can lead to poor performance in specific categories. MRTMD primarily captures standard traffic conditions and lacks diversity in challenging scenarios such as adverse weather and low-light conditions. The absence of these factors limits the applicability of this approach for robust model training. However, despite these limitations, MRTMD provides a valuable basis for evaluating pre-trained models across different resolutions, allowing researchers to assess model performance under multi-resolution conditions effectively and forms the basis for any future research in the area.

C. METRICS USED

1) OBJECT DETECTION METRICS

Intersection over Union (IoU) is a metric that quantifies the overlap between the predicted bounding box and the ground truth, which measures the accuracy of object localisation. As depicted in Equation 1, b represents the bounding box and g denotes the ground truth.

$$\text{IoU}(b, g) = \frac{\text{area}(b \cap g)}{\text{area}(b \cup g)} \tag{1}$$

Precision, a metric that signifies the proportion of relevant items retrieved by the model, is formally defined in Equation 2. It is calculated as the ratio of true positives (objects correctly identified as relevant) to the sum of true positives and false positives (objects incorrectly identified as relevant).

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{\text{IoU}(b, g) > \text{threshold}}{\text{IoU}(b, g) > \text{threshold} + \text{FP}} \tag{2}$$

Recall, represented in Equation 3, quantifies the effectiveness of retrieving relevant items by the model. It is calculated as the ratio of true positives (objects correctly identified as relevant) to the sum of true positives and false negatives (objects that are relevant but not retrieved by the model).

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{\text{IoU}(b, g) > \text{threshold}}{\text{IoU}(b, g) > \text{threshold} + \text{FN}} \tag{3}$$

AP, as represented by Equation 4, calculates the cumulative precision for each recall value R_k in the range from 0 to n, where n signifies the total number of relevant items.

$$\text{Average Precision (AP)} = \sum_{k=0}^{n} (R_k - R_{k-1}) \cdot P_k \tag{4}$$

The mAP, expressed in Equation 5, computes the average precision across all classes N by summing up the individual average precision values AP_i and dividing by the total number of classes.

$$\text{Mean Average Precision (mAP)} = \frac{1}{N} \sum_{i=1}^{N} \text{AP}_i \tag{5}$$

2) NPR METRICS

The accuracy metrics were calculated using the Levenshtein distance, which measures the number of single-character edits required to change one string into another, as shown in Equation 6. This metric is particularly suited for OCR tasks as it quantifies the discrepancies between the recognised text and the ground truth.

$$\text{Lev}(x, y) = \begin{cases} \text{if } \min(|x|, |y|) = 0 \\ \quad \max(|x|, |y|) \\ \text{else} \\ \quad \min \begin{cases} \text{Lev}(x_{1..m-1}, y) + 1 \\ \text{Lev}(x, y_{1..n-1}) + 1 \\ \text{Lev}(x_{1..m-1}, y_{1..n-1}) + \text{cost}(x_m, y_n) \end{cases} \end{cases} \tag{6}$$

The Character Error Rate (CER), as shown in Equation 7, normalises the Levenshtein distance by the length of the ground truth string.

$$\text{CER} = \frac{\text{Lev}(x, y)}{|y|} \tag{7}$$

Similarly, the Word Error Rate (WER), as shown in Equation 8, normalises the sum of substitutions, deletions, and insertions by the number of words in the reference sequence.

$$\text{WER} = \frac{\text{Lev}(w_x, w_y)}{|w_y|} \tag{8}$$

D. TASK DESIGN

Task execution involves processing the MRTMD Dataset through object detection models at all available resolutions. The selection of object detection models in this study was driven by the need to evaluate both traditional and modern architectures across a range of computational and accuracy trade-offs. We include YOLOv7, YOLOv8, and YOLOv9,

representing the latest advancements in single-stage real-time object detection. These models are commonly used in ITS systems due to their speed and efficiency. Faster R-CNN is included as a two-stage detection approach, providing a high-accuracy baseline for comparison with real-time models. FCOS is selected as an anchor-free detector, offering an alternative to traditional anchor-based methods such as YOLO and Faster R-CNN. SSD (Single Shot MultiBox Detector) is included as an older baseline to demonstrate the progression of object detection techniques, particularly in handling small objects. Finally, we incorporate RT-DETR (Real-Time Detection Transformer) to assess the performance of transformer-based architectures, which have recently shown promise in object detection. Unlike DETR, which suffers from high computational cost and slow convergence, RT-DETR provides a real-time alternative, making it a practical choice for traffic monitoring applications. This selection of object detection models ensures a comprehensive evaluation of object detection techniques, covering both convolutional and transformer-based architectures, and allowing us to analyse performance trade-offs across resolution, model complexity, and inference speed.

Additionally, we evaluate NPR performance across multiple resolutions as a separate experiment. This analysis highlights a practical task in traffic monitoring where resolution plays a critical role. Unlike object detection, where lower resolutions may still yield reasonable accuracy, NPR relies on fine-grained details, and the ability to extract text accurately is highly dependent on the available pixel information. In this experiment, we extract number plates from the same dataset at different resolutions—2160p, 1440p, 1080p, 720p, 480p, and 360p—to assess how much resolution affects the ability of an OCR system to interpret license plates correctly. The extracted number plates from higher-resolution images retain significantly more detail, while those from lower resolutions experience loss of edge sharpness and character visibility, which can degrade recognition performance. Through this comparative analysis, we assess the efficiency of processing at each resolution, along with mAP and Recall scores, to establish whether the increase in resolution to 2160p provides a significant advantage over lower resolutions for NPR tasks. By including this task, we aim to illustrate how resolution trade-offs can impact specific real-world applications in ITS, where some tasks require higher resolution than others.

The results from these experiments will be analysed to determine if processing images at different resolutions offers significant advantages in practical traffic monitoring applications.

1) OBJECT DETECTION MODELS USED

The table (Table 3) presents the benchmark results for various object detection models trained on the COCO dataset and evaluated on COCOval17, comparing their performance in terms of mean Average Precision (mAP) across IoU thresholds from 50 to 95, the number of parameters (in millions), and the computational complexity measured in FLOPs (floating-point operations). All models were evaluated using an input size of 640 × 640 pixels. The results highlight the trade-offs between accuracy and computational cost, with YOLOv9 achieving the highest mAP of 55.6 while maintaining a relatively lower computational demand of 192.5B FLOPs. In contrast, YOLOv8, with a slightly lower mAP of 53.9, requires the highest computational cost at 257.8B FLOPs.To gauge the isolated impact of input resolution, we include an additional 1280 × 1280 pixels run only for YOLOv7. This detector achieved the highest mAP at an input size of 640 × 640 pixels, making it a strong representative baseline.

TABLE 3. Comparison of input size, mAP scores, number of Parameters in millions as well as FLOPs for the models used for evaluating the MRTMD dataset in this study. It is essential to note that YOLOv7* is simply a variant of the standard YOLOv7, with a larger input size.

Model	Size (pixels)	mAPval 50-95	Params (M)	FLOPs
YOLOv9 [18]	640	55.6	58.1	192.5B
YOLOv8 [19]	640	53.9	68.2	257.8B
YOLOv7 [20]	640	56.8	71.3	189.9G
YOLOv7* [20]	1280	53.1	151.7	843.2G
Faster R-CNN [21]	640	37	41.7	134.4G
SSD [23]	640	25.1	35.6	34.9G
FCOS [22]	640	39.2	128.2	128.2G
RT-DETR [24]	640	46.5	20	60.7G

2) NPR MODEL

The process adopted for this study starts with preparing an image for OCR, beginning with a series of pre-processing steps designed to enhance image quality and improve text recognition accuracy (illustrated in Fig. 5). These steps are primarily tailored for Tesseract, but the same cropped number plate images were used as input for EasyOCR and TrOCR to ensure consistency in evaluation.

The image is first resized by 200%, followed by conversion to greyscale to simplify the visual content by removing colour information. A Gaussian filter is applied to the greyscale image to reduce noise while preserving edge sharpness. Adaptive thresholding is then performed to convert the image into a binary format, enhancing the distinction between text and background. The resulting binary image is fed into the Tesseract OCR engine, which detects text regions and performs character recognition. For EasyOCR and TrOCR, no custom pre-processing was applied, as both models include internal pipelines that manage image normalisation and feature extraction as part of their deep learning-based architectures.

V. RESULTS AND INTERPRETATION

A. BENCHMARKING

This section we present results related to Object Detection and NPR evaluation based upon the MRTMD Dataset.

1) OBJECT DETECTION

This section evaluates the performance of various object detection models across different input resolutions. The

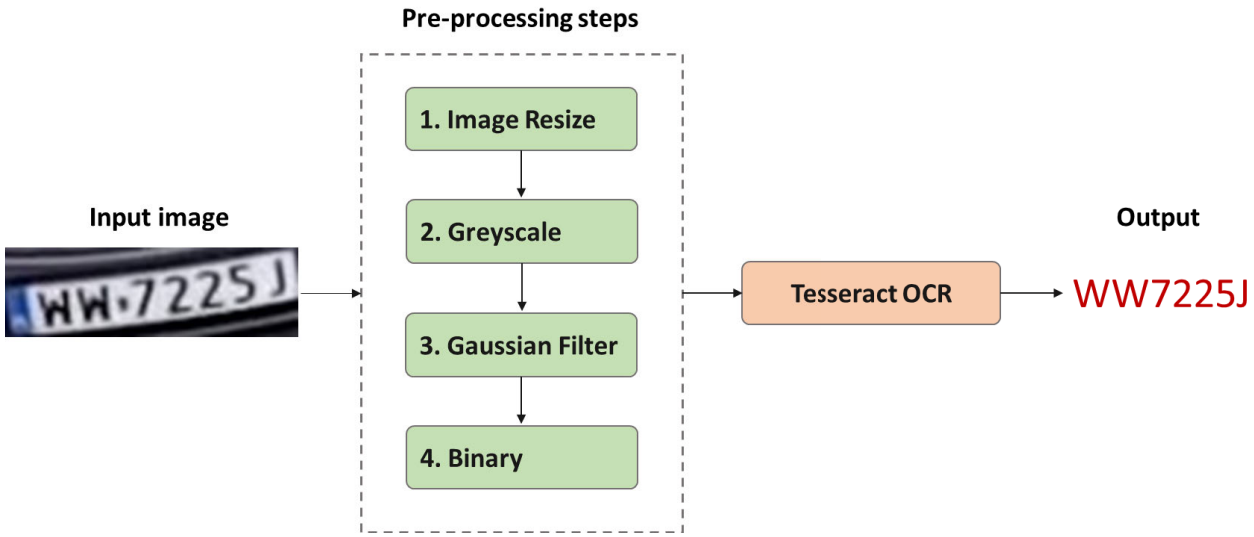

FIGURE 5. NPR starts with a pre-processing step that includes resizing the image, converting to greyscale, applying a Gaussian filter then finally converting to binary before processing through Tesseract OCR.

TABLE 4. Mean and standard deviation of mAP across resolutions for each object detection model.

Model	Mean mAP	Standard Deviation
YOLOv9 [18]	0.357	0.009
YOLOv8 [19]	0.359	0.027
YOLOv7 [20]	**0.365**	**0.035**
YOLOv7* [20]	0.362	0.014
SSD [23]	0.031	0.002
RT-DETR [24]	0.326	0.025
Faster R-CNN [21]	0.332	0.019
FCOS [22]	0.359	0.018

models under consideration include YOLOv9, YOLOv8, YOLOv7, YOLOv7*, Faster R-CNN, FCOS, SSD, and RT-DETR. The performance metric used is the Mean Average Precision (mAP), measured at six resolutions: 2160p, 1440p, 1080p, 720p, 480p, and 360p. The analysis focuses on several key metrics: the mean and standard deviation of mAP across resolutions for each model, the performance declines on, and the drop-off in performance as resolution decreases.

B. ACCURACY (mAP)

Figure 6 compares total mAP values across resolutions for each model. YOLOv7 exhibits the highest performance across most resolutions, achieving a peak mAP of 0.397 at 720p and maintaining strong performance at 480p (0.367). This indicates that YOLOv7 is particularly robust and well-suited to various input qualities, with a slight preference for mid-range resolutions. YOLOv9 shows relatively stable performance across resolutions, with a peak mAP of 0.370 at 1440p. However, its performance declines at lower resolutions, dropping to 0.344 at 360p. This suggests that YOLOv9 is generally reliable but somewhat dependent on higher resolution inputs to achieve optimal results. YOLOv8 performs well at higher resolutions, such as 1080p (0.381) and 720p (0.376), but encounters a significant drop in performance at 480p, where its mAP decreases to 0.300. This suggests that YOLOv8 is sensitive to changes in resolution and may struggle with lower-resolution inputs. YOLOv7*, while similar to YOLOv7, shows a consistent but slightly lower performance across all resolutions, with mAP values ranging from 0.343 at 2160p to 0.377 at 720p. This indicates that YOLOv7* is a stable performer, though it does not reach the levels of YOLOv7 (640 $\times$ 640 input). Faster R-CNN displays a variance in performance across different resolutions. It achieves its highest mAP of 0.341 at 720p, but its performance declines at lower resolutions, with a mAP of 0.299 at 360p. Faster R-CNN can perform well under certain conditions but tends to underachieve as the resolution decreases. FCOS exhibits reasonable performance at higher resolutions, with a mAP of 0.373 at 720p. However, it shows a significant decline at 360p, where its mAP drops to 0.334. This means that FCOS is more effective at higher resolutions and may require fine-tuning for lower resolution inputs.

RT-DETR shows relatively consistent performance across most resolutions. It achieves a peak mAP of 0.359 at 1440p and maintains strong results at 2160p (0.356) and 1080p (0.334). However, there is a gradual decline in performance at lower resolutions, with mAP values of 0.312 at 720p and 0.296 at 360p. Despite the decrease, RT-DETR demonstrates a balance of performance across different input qualities, making it suitable for scenarios with varying resolution requirements. SSD consistently underperforms across all resolutions, with mAP values around 0.030, highlighting its unsuitability for this object detection task.

1) MEAN AND STANDARD DEVIATION ACROSS RESOLUTIONS

The mean mAP for each model across all resolutions provides a summary of its overall performance. This metric is

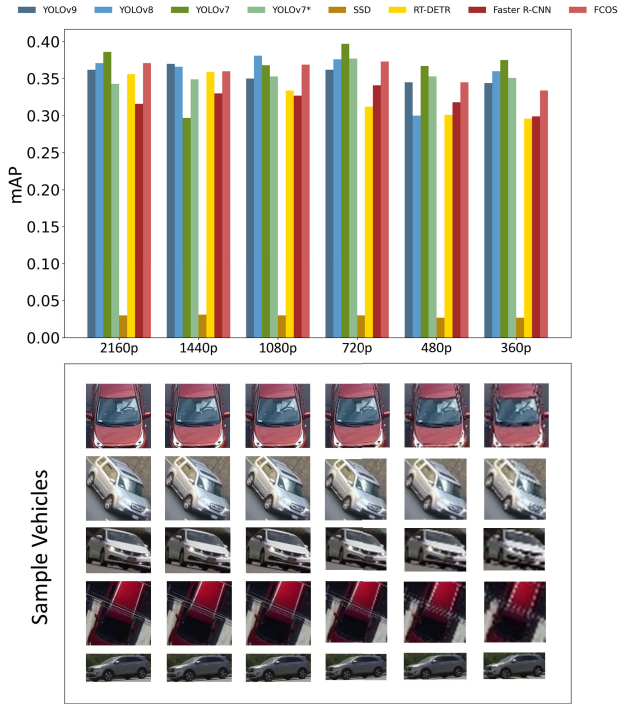

FIGURE 6. Comparison of mAP values across all resolutions.

calculated by averaging the mAP values at each model's six resolutions (2160p, 1440p, 1080p, 720p, 480p, and 360p).

From Table 4 YOLOv7 has a mean mAP of 0.365, indicating overall solid performance, though its standard deviation (0.032) suggests moderate variation across resolutions. This variability implies that while YOLOv7 generally performs well, its effectiveness may depend on the specific resolution used. YOLOv7* exhibits a slightly lower mean mAP of 0.354, accompanied by a lower standard deviation (0.011), indicating more consistent performance but slightly less robustness compared to YOLOv7. YOLOv9 and YOLOv8 exhibit similar mean mAP values (0.356 and 0.359, respectively), but YOLOv8 has a higher standard deviation (0.027), indicating more significant performance fluctuation across resolutions. In contrast, YOLOv9's lower standard deviation (0.010) suggests a more consistent performance, though it is slightly less robust overall.

Faster R-CNN shows a mean mAP of 0.322 with a standard deviation of 0.013, highlighting sensitivity to resolution changes. Its lower mean mAP compared to YOLO models suggests that Faster R-CNN may be less effective across all resolutions. FCOS has a mean mAP of 0.359 and a standard deviation of 0.014, indicating decent average performance with moderate variability. RT-DETR achieves a mean mAP of 0.326 with a standard deviation of 0.025, indicating a balance between performance and variability across different resolutions. While it does not match the peak performance of the YOLO models, it demonstrates reasonable consistency, making it suitable for a range of input resolutions. SSD exhibits a very low mean mAP of 0.029, with an extremely low standard deviation (0.002). This reflects consistently poor performance across all resolutions, making it unsuitable for this object detection task.

TABLE 5. Mean mAP across all models for each resolution.

Resolution	Mean mAP
2160p	0.317
1440p	0.308
1080p	0.314
720p	**0.321**
480p	0.295
360p	0.298

TABLE 6. Drop-Off in mAP from 2160p to other resolutions, this metric measures the stability of the different models across resolutions. Since the drop-off hovers in the 0 mark, the trend seems to indicate that models are mostly stable and are not that impacted by the initial image resolution.

Model	1440p	1080p	720p	480p	360p
YOLOv9 [18]	0.008	0.012	0.000	0.007	0.018
YOLOv8 [19]	0.005	-0.010	-0.005	0.071	0.011
YOLOv7 [20]	0.037	-0.003	-0.020	-0.007	-0.004
YOLOv7* [20]	**-0.015**	-0.007	**-0.024**	**-0.012**	**-0.015**
SSD [23]	-0.002	**-0.001**	-0.001	0.000	0.000
RT-DETR [24]	-0.003	0.022	0.044	0.055	0.060
Faster R-CNN [21]	0.007	0.003	-0.004	0.021	0.048
FCOS [22]	0.007	0.001	0.012	0.017	0.110

C. MEAN mAP ACROSS OBJECT DETECTION MODELS BY RESOLUTION

We also examine the mean mAP across all models for each resolution (as seen in Table 5). This analysis helps to identify which resolutions are generally more challenging or suitable for object detection tasks.

The highest mean mAP occurs at 720p (0.321), suggesting that mid-range resolutions may provide an optimal balance between detail and computational efficiency for object detection. This is followed by 1080p (0.314) and 2160p (0.317), indicating that higher resolutions, while offering more detail, do not always guarantee the best performance across all models. Specific models, such as YOLOv7, appear to be particularly well-optimised for 720p, which might explain this result.

The mean mAP decreases noticeably at 480p (0.295) and 360p (0.298), indicating that lower resolutions degrade performance due to the loss of crucial image details. The performance fluctuation across resolutions suggests that some models may handle mid-range resolutions more efficiently than higher ones, possibly due to resolution-based optimisations or scaling effects. This pattern warrants further investigation in the conclusion section to understand how different models are affected by changes in resolution.

D. DROP-OFF ANALYSIS BY OBJECT DETECTION MODEL

Drop-off analysis shows how much an object detection model's mAP performance decreases as the resolution lowers from 2160p to other resolutions (as seen in Table 6). This

analysis helps identify models that are more robust to changes in resolution.

YOLOv7 and YOLOv7* show negative drop-offs at most resolutions, demonstrating that these models improve as the resolution decreases. This trend is particularly evident at 720p, where YOLOv7 achieves its peak performance, suggesting that it might be optimally tuned for mid-range resolutions. Faster R-CNN maintains small drop-offs across all resolutions, with the largest being 0.017 at 360p. However, its overall mAP is lower than YOLOv7, indicating that while Faster R-CNN is relatively stable, it is not as robust in maintaining high performance across different resolutions. RT-DETR shows gradual but relatively stable drop-offs across resolutions, with a decrease of 0.003 at 1440p and larger drop-offs at lower resolutions, such as 0.044 at 720p and 0.060 at 360p. Despite these declines, the model maintains consistent performance across varying input qualities.

FCOS, while generally stable, shows a noticeable drop at 360p (0.037). This drop-off suggests that FCOS, similar to RT-DETR, may be more suitable for applications that consistently involve higher-quality inputs, although FCOS remains relatively stable up to 720p. SSD, with its negligible drop-offs, exhibits consistent performance across all resolutions; however, its consistently low mAP values underscore its inadequacy for this task, regardless of the input resolution. The almost flat drop-off values indicate that SSD's limitations are fundamental to its architecture and performance capabilities rather than being resolution-dependent. While YOLOv9 shows small positive drop-offs across most resolutions, it remains relatively stable, but its performance slightly declines as resolution decreases.

Overall, this analysis demonstrates that models such as YOLOv7 and Faster R-CNN are particularly robust across varying resolutions, with YOLOv7 even showing improvements at lower resolutions. These models are ideal for applications where resolution may vary or mid-range resolutions, such as 720p, are common. On the other hand, models such as RT-DETR and FCOS exhibit clear resolution sensitivities.

E. RECALL PERFORMANCE

Table 7 shows that YOLOv7 is the most effective model for the MRTMD dataset, capable of maintaining high detection rates across various resolutions. Faster R-CNN and YOLOv9 also demonstrate strong and consistent recall performance, making them reliable alternatives, depending on the specific task requirements. However, models like SSD demonstrate significant challenges in achieving satisfactory recall, particularly as resolution varies, indicating that this model may require further development or fine-tuning to be viable in real-world applications using the MRTMD dataset.

1) MEAN AND STANDARD RECALL DEVIATION ACROSS RESOLUTIONS

The mean recall for each model across all resolutions summarises the model's overall recall performance (as seen in Table 8). This metric is calculated by averaging the recall values at all six resolutions (2160p, 1440p, 1080p, 720p, 480p, and 360p) for each model.

TABLE 7. Total recall for each resolution and model.

Model	2160p	1440p	1080p	720p	480p	360p
YOLOv9 [18]	0.394	**0.404**	0.381	0.391	0.374	0.373
YOLOv8 [19]	0.397	0.390	**0.407**	0.398	0.320	0.385
YOLOv7 [20]	**0.405**	0.314	0.388	**0.416**	0.382	0.393
YOLOv7* [20]	0.365	0.370	0.373	0.397	0.373	0.371
SSD [23]	0.023	0.022	0.023	0.023	0.023	0.022
RT-DETR [24]	0.369	0.357	0.345	0.329	0.321	0.318
Faster R-CNN [21]	0.372	0.382	0.383	0.394	**0.389**	**0.394**
FCOS [22]	0.381	0.371	0.377	0.380	0.365	0.360

TABLE 8. Mean and standard deviation of recall across resolutions for each object detection model. Higher mean and lower standard deviation indicate a more stable model, able to handle a drop in image quality.

Model	Mean Recall	Standard Deviation
YOLOv9 [18]	0.386	0.011
YOLOv8 [19]	0.382	0.029
YOLOv7 [20]	**0.400**	0.013
YOLOv7* [20]	0.373	0.010
SSD [23]	0.023	0.001
RT-DETR [24]	0.340	**0.019**
Faster R-CNN [21]	0.386	0.008
FCOS [22]	0.372	0.010

YOLOv7 achieves the highest mean recall of 0.400, indicating strong overall recall performance. Its relatively low standard deviation (0.013) reflects consistent performance across different resolutions. YOLOv7*, while showing a slightly lower mean recall of 0.373, also exhibits low variability (standard deviation of 0.010), making it a reliable model for recall across various resolutions. YOLOv9 has a mean recall of 0.386 with a standard deviation of 0.011, indicating that it performs consistently across resolutions, albeit slightly less effectively than YOLOv7. YOLOv8 has a similar mean recall (0.382) but a higher standard deviation (0.029), indicating more variability in its recall performance depending on the resolution.

Faster R-CNN has a mean recall of 0.386 and the lowest standard deviation (0.008), highlighting its stable recall performance across resolutions. However, its overall recall is slightly lower than that of YOLOv7, indicating that while Faster R-CNN is stable, it may not capture as many objects as YOLOv7. FCOS exhibits a mean recall of 0.372 with a standard deviation of 0.010, indicating moderate performance with some consistency across different resolutions. RT-DETR demonstrates a mean recall of 0.340 with a standard deviation of 0.019, reflecting stable but somewhat lower recall performance compared to other models like YOLOv9 and Faster R-CNN. SSD consistently underperforms with a very low mean recall of 0.023 and an extremely low standard deviation (0.001), confirming its inadequacy for this object detection task.

TABLE 9. Mean recall across all models for each resolution.

Resolution	Mean Recall
2160p	0.338
1440p	0.326
1080p	0.335
720p	**0.341**
480p	0.318
360p	0.327

TABLE 10. Drop-Off in recall from 2160p to other resolutions, this metric measures the stability of the different models across resolutions. Since the drop-off hovers in the 0 mark, the trend seems to indicate that models are mostly stable and are not that impacted by the initial image.

Model	1440p	1080p	720p	480p	360p
YOLOv9 [18]	0.010	-0.013	-0.003	-0.020	-0.021
YOLOv8 [19]	-0.007	0.010	0.001	-0.077	-0.012
YOLOv7 [20]	**-0.091**	-0.017	0.011	-0.023	-0.012
YOLOv7* [20]	0.005	0.008	0.032	0.008	0.006
SSD [23]	-0.001	0.000	0.000	0.000	-0.001
RT-DETR [24]	-0.012	**-0.024**	**-0.040**	**-0.048**	**-0.051**
Faster R-CNN [21]	0.010	0.011	0.022	0.017	0.022
FCOS [22]	-0.010	-0.004	-0.001	-0.016	-0.021

F. MEAN RECALL ACROSS OBJECT DETECTION MODELS BY RESOLUTION

We also examine the mean recall across all models for each resolution (as seen in Table 9). This analysis helps to identify which resolutions are generally more suitable for object detection tasks across the board.

The highest mean recall is observed at 720p (0.341), followed by 1080p (0.335) and 360p (0.327). This suggests that mid-range resolutions such as 720p might still be suitable for general object detection tasks, possibly due to a balance between sufficient image detail and computational efficiency. Interestingly, 2160p, while offering the most image detail, has a lower mean recall (0.338), indicating that some models may not fully utilise the advantages of high-resolution input effectively. The mean recall decreases at 480p (0.318), showing that lower resolutions generally degrade recall due to loss of detail. However, the stabilisation of recall at 360p (0.327) suggests that certain models can maintain reasonable performance even at lower resolutions.

G. DROP-OFF ANALYSIS BY OBJECT DETECTION MODEL

Drop-off analysis shows how much an object detection model's recall performance decreases as the resolution lowers from 2160p to other resolutions. This analysis helps identify models that are more robust to changes in resolution.

YOLOv7 shows a significant negative drop-off from 2160p to 1440p (-0.091) and 1080p (-0.017), followed by positive drop-offs at lower resolutions, including 0.011 at 720p. This indicates that it tends to recover and slightly improve its recall performance at mid-range resolutions. This trend is particularly evident at 720p, where YOLOv7 achieves a strong recall, suggesting that it is optimised for mid-range resolutions. Faster R-CNN maintains small positive drop-offs across most resolutions, with the largest being 0.022 at 720p and 360p. This stability highlights that Faster R-CNN is robust to changes in resolution, making it a versatile model for various input qualities. RT-DETR exhibits consistent negative drop-offs, with a value of -0.012 at 1440p and progressively larger declines, reaching -0.051 at 360p. These results indicate that RT-DETR may struggle to maintain recall performance as resolution decreases, though the decline is gradual rather than drastic, showing reasonable stability despite the negative trend.

FCOS shows slight negative drop-offs across several resolutions, with a more noticeable decrease of -0.021 at 360p. This suggests that FCOS may be better suited for applications with higher-quality inputs, although it maintains reasonable performance up to 720p. SSD exhibits almost negligible drop-offs, reflecting its consistently poor recall performance across all resolutions. The minimal variation in recall underscores SSD's limitations, which are intrinsic to its design rather than resolution-dependent. YOLOv9 shows a slight positive drop-off at 1440p (0.010), followed by small negative drop-offs at lower resolutions, indicating a stable but slightly declining recall as resolution decreases.

Overall, this analysis reveals that models such as YOLOv7 and Faster R-CNN are particularly robust across varying resolutions, with YOLOv7 showing a slight improvement in recall at mid-range resolutions. These models are ideal for applications where resolution may vary or where mid-range resolutions like 720p are common. On the other hand, models like RT-DETR and FCOS show clear resolution sensitivities, particularly at lower resolutions such as 360p, indicating that they may require specific considerations or adjustments when deployed in environments with varied resolutions. SSD's poor recall performance across all resolutions further cements its position as a less suitable option for this object detection task as can be observed in Table 10.

1) PERFORMANCE

In our evaluation of various object detection models across different image resolutions shown in 7, YOLOv9 emerged as the fastest model, consistently demonstrating the lowest processing times (in seconds per image) across all tested resolutions. This makes YOLOv9 the most efficient model in terms of speed. YOLOv8 and RT-DETR also performed well, with slightly higher but competitive processing speeds. Conversely, SSD was the slowest among the models,

TABLE 11. Class-wise performance metrics.

Class	Precision	Recall	F1-Score
Person	**0.291**	**0.293**	**0.292**
Bicycle	0.058	0.057	0.058
Car	0.229	0.246	0.237
Motorcycle	0.257	0.257	0.257
Bus	0.221	0.258	0.238
Truck	0.155	0.219	0.181

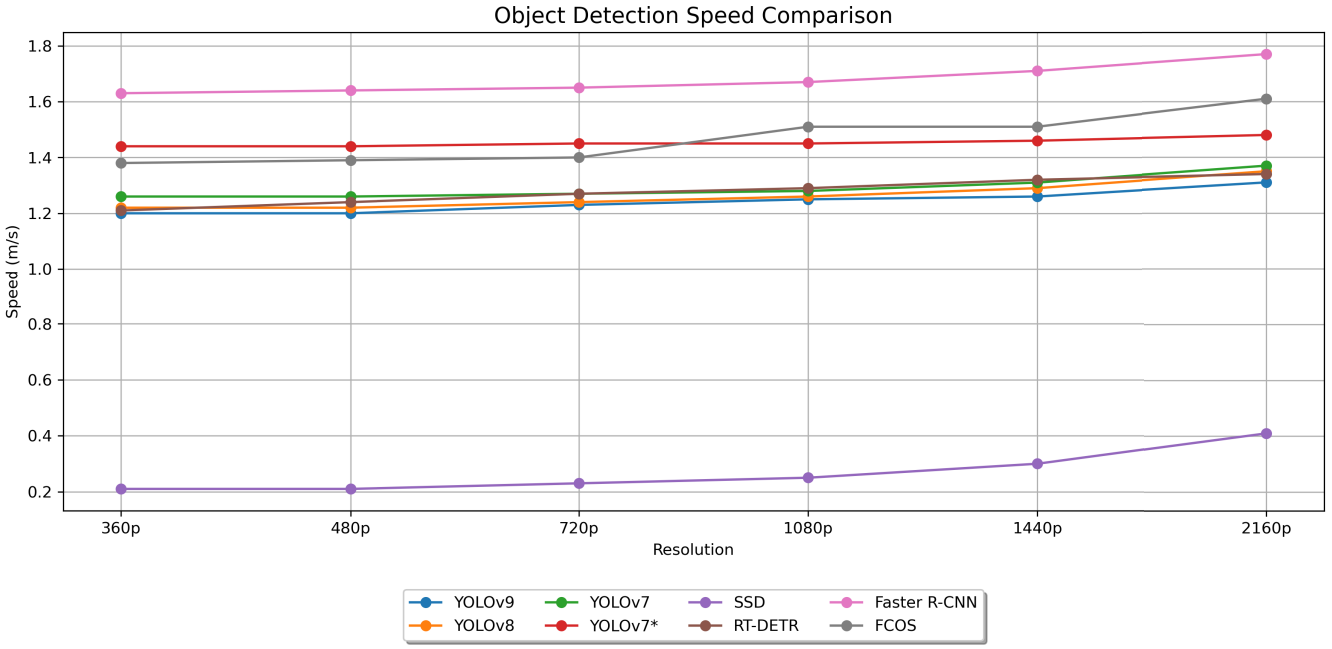

FIGURE 7. Comparison of object detection speed across different models and resolutions. The graph shows the performance of various object detection models (YOLOv7*, SSD, FCOS, Faster R-CNN, YOLOv7, YOLOv9, YOLOv8, and RT-DETR) across different video resolutions (360p to 2160p). The y-axis represents the processing speed in meters per second (m/s), while the x-axis shows the different resolutions.

exhibiting significantly longer processing times across all resolutions. Faster R-CNN and YOLOv7 displayed moderate performance, with Faster R-CNN being slower but stable, while YOLOv7 showed a slight decrease in speed as resolution decreased. Overall, YOLOv9 is optimal for speed-critical applications in object detection tasks. It is also important to note the distinction between YOLOv7 and YOLOv7-1280, which refers to the larger input size of the latter model. Despite the larger input size, the latter YOLO version performed almost as fast as the 640×640 input version of the same model.

TABLE 12. Average bounding-box areas at different resolutions versus COCO, rounded to the nearest integer. 720p stands out for its closer alignment with COCO on *cars*, the dominant category in our dataset.

Resolution	bus	motorcycle	truck	person	bicycle	car
COCO	52569	34963	30011	21205	16282	7091
360p	4744	199	3167	41	92	1329
480p	8441	354	5634	73	163	2365
720p	18977	797	12667	163	367	5318
1080p	42698	1793	28502	367	825	11965
1440p	75908	3187	50670	653	1466	21272
2160p	170793	7171	114007	1468	3299	47861

2) ANALYSING THE BEST OVERALL OBJECT DETECTION MODEL

Table 11 class-wise performance metrics reveal distinct differences in how well the model detects and classifies different object types. The Person class exhibits a balanced performance with a precision of 0.291, a recall of 0.293, and an F1-score of 0.292, indicating that the model is fairly consistent in detecting persons. However, the overall accuracy could be improved. The Car class exhibits slightly higher performance, with a precision of 0.229 and a recall of 0.246, resulting in an F1-score of 0.237. This suggests that the model is reasonably effective at identifying cars, though it still misses many. The Motorcycle and Bus classes exhibit similar performance, with F1-scores of 0.257 and 0.238, respectively, indicating moderate success in detecting these objects. However, both classes show room for improvement, particularly in precision. On the other hand, the Truck class has lower performance, with a precision of 0.155 and a recall of 0.219, resulting in an F1-score of 0.181, indicating that the model struggles more with accurately detecting trucks than other vehicles. The Bicycle class performs the worst, with a precision and recall of both around 0.058, reflecting significant challenges in detecting bicycles, which may be due to their smaller size or less distinct features in the dataset. To assess the spatial distribution of object detection across different videos, we generated heatmaps for each video segment, specifically for Video 1, Video 2, and Video 3 (fig. 8). The heatmaps were derived from the bounding box locations of detected objects in the images, with each pixel's intensity representing the frequency of bounding box overlaps in that region. This method highlights areas of the image where the YOLOv7 object detection model was most active. The results indicate that the best performance was achieved on video 1. In fact, at this angle, the mAP and recall values were 0.542 and 0.528, respectively.

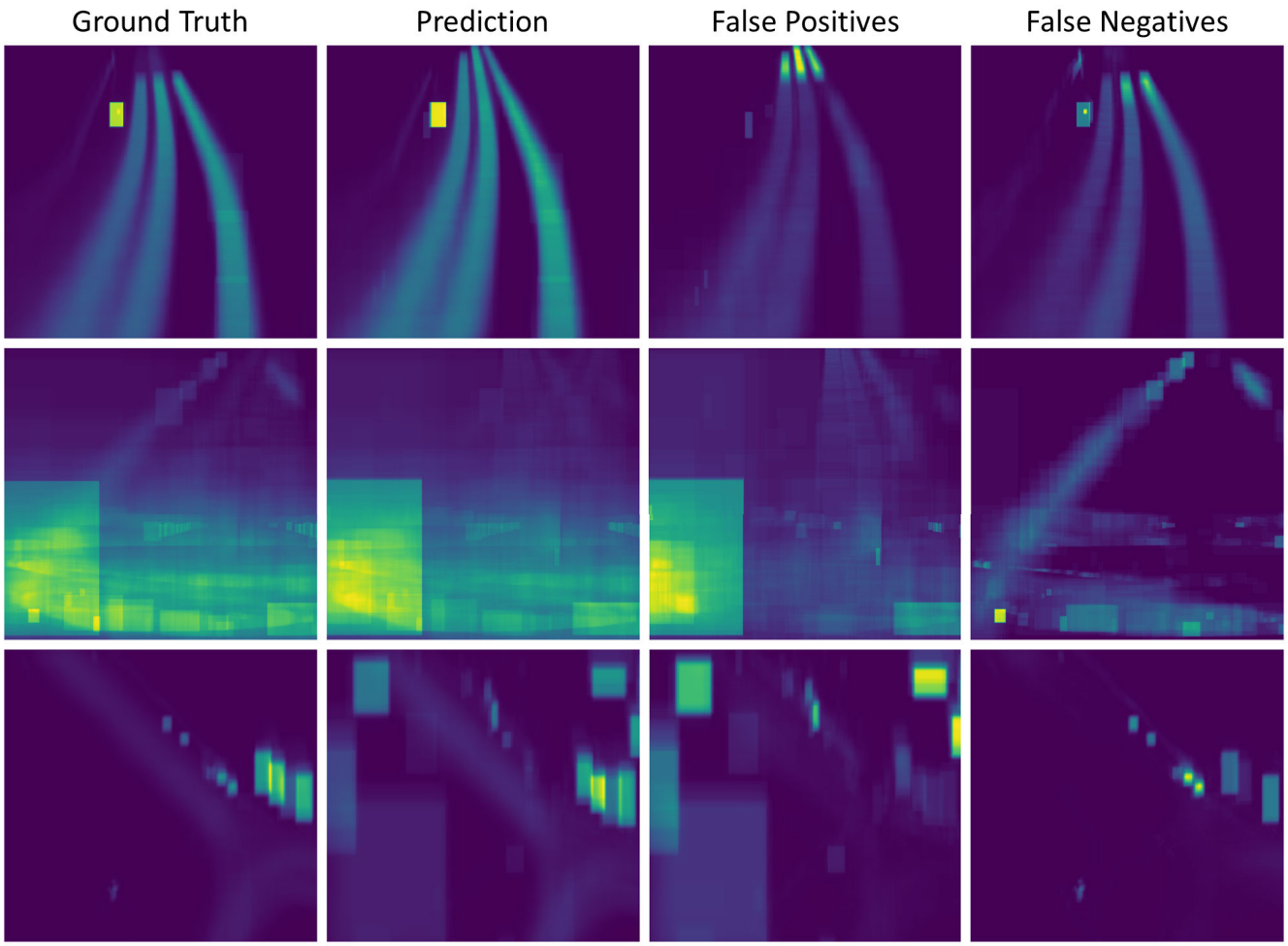

FIGURE 8. Comparison of heatmaps for ground truth, prediction, false positive and false negative results using YOLOv7 model for Video 1, Video 2, and Video 3, respectively.

3) 720p PERFORMANCE

In our experiments, the model trained on COCO performed best on our 720p dataset. Table 12 compares the average bounding-box areas of key object categories (*bus, motorcycle, truck, person, bicycle, car*) between the COCO reference and various resolutions, including 720p. From the table, we can observe that while 1080p and 1440p may be numerically closer to COCO on some individual categories, the large presence of *cars* (43.3% of total labels) in our dataset makes 720p more likely to achieve better accuracy as the average area for cars in 720p is far closer to COCO ($\approx$ 5318 vs. $\approx$ 7091) than in 1080p ($\approx$ 11965). Additionally, 720p often sits in a comfortable range for many detection backbones, which can boost performance in practical deployments.

As seen, the 720p dataset provides a good balance by producing car bounding boxes that are relatively close in area to those in COCO. Since *cars* dominate our data distribution, this closer alignment may translate and explain the improved overall performance in detection metrics at 720p. Additionally, 720p typically aligns well with standard model architecture strides, favoring more efficient and effective feature extraction compared to very high (e.g., 2160p) or very low (e.g., 360p) resolutions.

4) CONCLUSION ON OBJECT DETECTION TASK

When comparing recall, mAP, and computational performance, YOLOv7 emerges as the most well-rounded model, offering both high accuracy and fast processing times, particularly at 720p, where it excels in both recall (0.416) and mAP (0.397). This makes it a strong candidate for real-time ITS applications that require efficient, accurate vehicle detection. Faster R-CNN and FCOS also achieve strong accuracy, but their higher computational cost makes them more suitable for applications prioritising accuracy over speed, such as traffic law enforcement or forensic analysis. YOLOv9 and YOLOv8 provide a balanced approach, making them well-suited for tasks requiring moderate accuracy with efficient processing times.

RT-DETR shows consistent recall and mAP values across resolutions, but its performance degrades significantly at

TABLE 13. OCR accuracy across resolutions for different methods.

Method	2160p	1440p	1080p	720p	480p	360p
Tesseract	0.0885	0.0438	0.0133	0.0015	0.0015	0.0015
EasyOCR	0.1654	0.0524	0.0107	0.0001	0.0000	0.0000
TrOCR	0.6139	0.5060	0.3153	0.0000	0.0000	0.0000

lower resolutions. This is likely because transformer-based architectures rely on fine-grained feature representations, which become less effective when downsampled. In contrast, SSD, despite its speed advantage, exhibits consistently low recall and mAP scores, indicating poor generalisation to the MRTMD dataset. This is expected, as SSD lacks modern feature fusion strategies that improve small object detection. YOLOv8 and YOLOv9 show more variability in performance across resolutions, suggesting that their architecture is more resolution-sensitive than YOLOv7.

These findings underscore the importance of selecting a model tailored to the specific requirements of a particular ITS task. For real-time applications requiring high accuracy, YOLOv7 is the most balanced choice. If accuracy is the top priority and processing time is less critical, Faster R-CNN is a viable alternative. On the other hand, tasks requiring lightweight models may benefit from YOLOv9 or YOLOv8, while RT-DETR may be more suitable for applications that leverage transformer-based architectures at higher resolutions.

H. NPR PERFORMANCE ACROSS OCR METHODS

To evaluate the effectiveness of NPR under varying image resolutions, we benchmarked three OCR systems: Tesseract (baseline), EasyOCR, and TrOCR. Each system was tested using number plates cropped from traffic images rendered at six different base resolutions 2160p, 1440p, 1080p, 720p, 480p, and 360p. It is important to note that the resolution refers to the original traffic footage, not the cropped plate itself; thus, the quality of each number plate is inherently tied to the resolution of the source image. The aim was to measure the impact of changes in source image resolution on OCR accuracy and to identify the most resilient OCR method for real-world deployment.

1) COMPARATIVE RESULTS

Table 13 provides a visual comparison of OCR accuracy across all models and resolutions. At the highest resolution of 2160p, TrOCR achieved significantly better recognition accuracy (0.6139) compared to Tesseract (0.0885) and EasyOCR (0.1654). TrOCR maintained an accuracy above 0.3 at 1080p, while the other models dropped below 0.05. This result highlights TrOCR's robustness at medium-to-high resolutions, but also reveals a consistent failure across all models below 720p.

Despite its lower accuracy, Tesseract remains a common lightweight baseline in many open-source applications and is therefore included for reference. However, the inclusion of EasyOCR and TrOCR demonstrates that state-of-the-art OCR models can achieve substantially better results under high-resolution conditions, which may be essential for traffic enforcement and surveillance systems requiring reliable number plate recognition.

Lower resolutions produce images with poor quality, making it challenging even for human observers to distinguish number plates. The degradation in image quality at lower resolutions results in higher error rates during the OCR process, as finer details essential for accurate recognition are lost. Notably, the findings underscore the importance of high-resolution images in automatic NPR systems. For practical applications, especially in environments where high precision is required, such as law enforcement and toll collection, ensuring high image resolution is paramount.

I. DISCUSSION

A consistent pattern emerges when evaluating various object detection models utilising the MRTMD dataset. Most models exhibit performance within a narrow range, typically varying by ± 0.01 to ± 0.03 in terms of mean Average Precision (mAP) and recall across diverse resolutions. This stability is evident across different input qualities, from high-resolution (2160p) to lower-resolution (360p) images, indicating robust performance regardless of resolution. These models are thus well-suited for practical deployment where resolution may vary due to camera capabilities or storage limitations.

Despite the stability in mAP and recall values, higher resolutions incur significant costs in terms of bandwidth and storage. Using HEVC (H.265), a commonly used video compression standard, typical storage requirements for a single frame range from approximately 1 MB for 2160p to 30 KB for 360p. These differences become substantial when scaled to real-time video streams. For example, a 2160p video stream typically requires 7.5 Mbps, while a 360p stream needs only 0.5 Mbps under HEVC compression. This disparity has direct implications for large-scale traffic monitoring systems where hundreds of cameras might be deployed.

Furthermore, the mAP values indicate that most models are relatively insensitive to resolution changes. YOLOv9, for instance, shows only a 0.026 difference between its highest mAP at 1440p (0.370) and its lowest at 360p (0.344). YOLOv7* varies by just 0.034 across all resolutions. Similarly, recall performance shows only moderate sensitivity to resolution changes, with Faster R-CNN maintaining a range of 0.022 between 360p and 2160p. These findings suggest that mid-range resolutions, such as 1080p and 720p, may offer a good balance between accuracy and resource efficiency in many real-world applications.

From a storage perspective, HEVC-compressed video requires approximately 80 GB per day for continuous 2160p footage, while 720p footage needs only 13 GB per day. Bandwidth requirements also scale with resolution, with 100 cameras streaming at 2160p demanding 750 Mbps compared to just 130 Mbps for 100 cameras at 720p. These differences are crucial for planning deployments where

FIGURE 9. Sample of COCO vehicle data, the camera angle, scene, as well as the setting are very similar to the content found in video 1 and video 2 in the MRTMD dataset.

both storage and network infrastructure may be limited. Additionally, lower resolutions, such as 480p or 360p, further reduce bandwidth and storage demands but may introduce a slight decline in performance for specific applications. For instance, YOLOv8's recall drops from 0.407 at 1080p to 0.320 at 480p, although this change is manageable in scenarios where speed and storage are higher priorities than maximum accuracy.

In large-scale deployments, such as city-wide traffic monitoring, these resource considerations become even more significant. Ten cameras streaming at 2160p would consume 800 GB of storage daily and require 75 Mbps of bandwidth, while 100 cameras would require 7.5 TB and 750 Mbps, respectively. Reducing the resolution to 720p would decrease storage needs to 1.3 TB and bandwidth to 130 Mbps, offering substantial savings while still maintaining stable mAP and recall values across most object detection models. However, in specialised tasks such as OC for automatic NPR, the choice of resolution becomes more critical. OCR systems demonstrate significant performance degradation at lower resolutions. Accuracy at 2160p reaches 0.089, but declines to 0.013 at 1080p and approaches zero below 720p. This trend reflects the necessity of high-resolution images to capture fine details for character recognition, which lower resolutions fail to preserve even with HEVC compression.

These findings indicate that while most object detection tasks can operate effectively at mid-range resolutions with reduced bandwidth and storage requirements, applications that depend on fine visual details, such as OCR, still require higher resolutions despite the associated resource demands. Consequently, model deployment strategies must consider both performance stability and infrastructure costs, leveraging resolution selection to optimise overall efficiency in large-scale surveillance or monitoring networks. Moreover, this study used generalised models as a baseline for our evaluations. While this approach may have yielded lower performance metrics than those achievable with domain-specific fine-tuning, it serves a crucial purpose in our research. The use of generalised models reflects a common real-world scenario where resources for extensive fine-tuning may not always be available, particularly in smaller municipalities or organisations with limited budgets. It is also important to note that none of the models used for object detection or OCR in this study were fine-tuned explicitly for vehicle and NPR tasks. The accuracy observed across

different resolutions in both object detection and OCR could improve if the models were better trained on domain-specific datasets and if advanced image pre-processing techniques were employed. Fine-tuning the models with relevant data would likely enhance their ability to accurately detect and recognise vehicles and number plates, making them more reliable for practical applications such as law enforcement, toll collection, and automated traffic monitoring systems.

Figure 9 displays a sample of vehicle data used to train the models in this study. Notably, the images in this sample bear a closer resemblance to the camera angles found in video1 and some angles in video2. This similarity in perspective may explain the mAP and recall performance observed for these particular video feeds. For instance, examining the data across all resolutions, we find that video1 consistently outperforms the other videos regarding mAP and recall for most models. Taking YOLOv7 as an example, the average mAP for video1 across all resolutions is 0.519, compared to 0.396 for video4 and 0.331 for video5. Similarly, the average recall for YOLOv7 on video1 is 0.519, while video2 and video3 show average recalls of 0.394 and 0.330, respectively. This pattern is consistent across most models, highlighting the importance of training data that closely matches the intended application scenario. By clarifying the performance of these generalised models across various resolutions, we provide valuable insights to the community, especially for those needing to implement traffic monitoring systems with off-the-shelf solutions. Our findings offer a realistic baseline of what can be expected without specialised optimisation, particularly relevant for rapid deployments or resource-constrained environments.

It follows that organisations or researchers with the resources to fine-tune these models for specific traffic monitoring tasks would likely achieve better results than those presented in our study. This underscores an important avenue for future research: the potential improvements that could be realised through domain-specific training and optimisation. Our work, therefore, serves dual purposes. Firstly, it provides practical guidance for immediate implementation using generalised models. Secondly, it establishes a foundation for future research, potentially demonstrating the added value of fine-tuning in this specific domain. This approach aligns with our goal of bridging the gap between theoretical advancements and practical implementations in traffic monitoring systems.

In conclusion, while generalised models may be seen as a limitation resulting in lower performance metrics, this approach makes our findings widely applicable and immediately useful to the community. It provides a clear starting point for practitioners and researchers alike, whether they are working with constrained resources or have the capacity for more advanced optimisation. Future work in this area could focus on quantifying the improvements gained through domain-specific fine-tuning, further enhancing our understanding of the optimal balance between model sophistication, resolution requirements, and system performance in real-world traffic monitoring applications. This potential for improved performance suggests that the results presented here could represent a baseline, with room for enhancement through targeted model training and optimisation techniques. Future research directions could include developing specialised vehicle and number plate detection models, exploring advanced image enhancement techniques to improve low-resolution performance, and investigating the impact of environmental factors such as lighting conditions and weather on detection accuracy.

VI. CONCLUSION

This study highlights the complex interplay between resolution, model performance, storage requirements, and practical applications in object detection and NPR systems. While higher resolutions offer greater detail and accuracy, particularly for OCR tasks, they come with increased storage and bandwidth costs that may be prohibitive in large-scale deployments. Therefore, the choice of resolution and model must be carefully balanced against the application's specific requirements, considering factors such as accuracy needs, storage limitations, scalability concerns, and cost constraints.

A. FUTURE WORK

While this study provides valuable insights into the performance of generalised models across various resolutions in traffic monitoring scenarios, there are several promising avenues for future research:

1) MULTIMODAL DATASET DEVELOPMENT

Future iterations of the MRTMD could incorporate additional modalities such as LiDAR, radar, or thermal imaging. A multimodal dataset would allow researchers to explore the synergies between different sensing technologies and potentially improve the robustness and accuracy of traffic monitoring systems. For instance, combining high-resolution visual data with LiDAR point clouds could enhance object detection and classification in challenging weather conditions or low-light environments [93], [94], [95].

2) ADDITIONAL IMAGE VARIETY AND SEGMENTATION CAPABILITIES

The MRTMD dataset currently contains various scenes with different lighting conditions. However, object detection under diverse lighting and weather conditions remains a challenging task [3], [96]. To address this and allow for better benchmarking, future additions to the MRTMD dataset will include traffic images captured at night, scenes with adverse weather conditions (such as fog, snow, and rain), and various challenging lighting situations (like glare, shadows, and low-light environments). These enhancements will make the dataset more comprehensive, enabling researchers to develop and test more robust algorithms capable of performing well in real-world situations where lighting and weather conditions are often far from ideal. Moreover, alongside the increased image variety, there is potential to update and

broaden the MRTMD dataset for segmentation tasks. Further experiments on the updated dataset could help explore how segmentation capabilities impact traffic analysis across different resolutions.

3) FINE-TUNING AND DOMAIN-SPECIFIC OPTIMISATION

As previously discussed, fine-tuning the models used in this study for specific traffic monitoring tasks could potentially yield significant performance improvements. Future work could quantify these improvements and explore the most effective strategies for domain-specific optimisation.

4) CARBON FOOTPRINT

An important consideration for large-scale deployments of AI-powered traffic monitoring systems is their environmental impact. Future research could focus on analysing the carbon footprint of these systems at different resolutions and processing complexities. This analysis would provide valuable insights for policymakers and system designers aiming to balance performance with environmental sustainability. Work has already been done in this space [97], [98]; this particular factor could influence the choice of model used.

5) EDGE COMPUTING

As traffic monitoring systems increasingly rely on real-time data processing, future work could explore the performance of these models in edge computing scenarios [99]. This research could investigate the trade-offs between resolution, model complexity, and processing latency in resource-constrained edge devices [100]. This is to understand better the impact resolution has on a traffic monitoring ecosystem.

6) ADAPTIVE RESOLUTION SYSTEMS

Building on our findings regarding the relationship between resolution and detection performance, future research could explore adaptive systems that dynamically adjust image resolution based on scene complexity or specific detection tasks. Such systems could optimise performance while minimising computational resources and bandwidth usage.

7) EXTENDED TEMPORAL ANALYSIS

While our current dataset focuses on spatial resolution, future work could incorporate longer temporal sequences to study the impact of frame rate and temporal resolution on detection performance, particularly for tasks like traffic flow analysis or anomaly detection. The current dataset already contains the necessary building blocks for this, with a frame taken for every 3 seconds of video, further annotation is required to allow research on these tasks. These future directions would not only build upon the foundation laid by this study but also address emerging challenges in the field of intelligent transportation systems. By expanding the scope to include multimodal data, environmental considerations, and adaptive processing strategies, future research can continue to bridge the gap between theoretical advancements and practical, sustainable implementations in traffic monitoring.

8) ADDITIONAL RECENT METHODS IN VEHICLE DETECTION

Several recent methods have been proposed for vehicle detection in traffic environments, utilising both convolutional and transformer-based architectures. For example, YOLOv8-FDD [101] introduces a lightweight detection head and improved feature interaction modules to enhance the performance of YOLOv8, achieving high detection accuracy with reduced computational complexity. Similarly, Chughtai and Jalal [102] present a robust multi-class vehicle detection and classification framework tailored to traffic surveillance scenarios, demonstrating reliable performance across varying vehicle types and conditions. On the transformer side, DETR-SPP [80] offers a fine-tuned DETR variant enhanced with spatial pyramid pooling for improved detection accuracy in complex road scenes.

While these contributions are noteworthy, the models or implementation code were not publicly available at the time of writing, which makes direct benchmarking or replication challenging. Reproducing such systems without official releases would require substantial engineering effort and assumptions about training pipelines, hyper-parameters, and dataset specifics, potentially leading to incomparable results. As such, they were not included in our experimental evaluation. However, their proposed advancements align with the focus of this study, and we intend to consider them in future work once their implementations become more accessible.

B. CONCLUSION

The MRTMD dataset stands out due to several key strengths, making it a valuable resource for evaluating and optimising models in various practical scenarios. With 3,733 images captured from varying camera angles, the dataset offers a comprehensive view that closely mimics real-world conditions. One of its unique features is the inclusion of the same images at different resolutions, which is particularly useful for assessing model efficiency across various resolution settings. This capability allows researchers and developers to determine the optimal model based on specific IP camera configurations, ensuring the selected model performs effectively under the given resolution constraints.

Moreover, the dataset's diversity in resolution makes it an excellent tool for measuring model stability when the camera feed quality degrades, a common issue in situations where bandwidth for streaming is limited. This characteristic is crucial for applications where network reliability, such as remote traffic monitoring or mobile settings, cannot always be guaranteed. By testing models with the MRTMD dataset, developers can gain a deeper understanding of how models behave under less-than-ideal conditions and make necessary adjustments to maintain optimal performance.

Additionally, the variety in camera angles and resolutions within the MRTMD dataset makes it a versatile resource for object detection tasks and for evaluating OCR performance, particularly in NPR scenarios. These strengths collectively make the MRTMD dataset a crucial tool for advancing research and development in fields that require robust, adaptable, and efficient visual recognition systems for traffic monitoring.

REFERENCES

[1] V. Mandal, A. R. Mussah, P. Jin, and Y. Adu-Gyamfi, "Artificial intelligence-enabled traffic monitoring system," *Sustainability*, vol. 12, no. 21, p. 9177, Nov. 2020.

[2] M. Abbasi, A. Shahraki, and A. Taherkordi, "Deep learning for network traffic monitoring and analysis (NTMA): A survey," *Comput. Commun.*, vol. 170, pp. 19–41, Mar. 2021.

[3] M. Bugeja, A. Dingli, M. Attard, and D. Seychell, "Comparison of vehicle detection techniques applied to IP camera video feeds for use in intelligent transport systems," *Transp. Res. Proc.*, vol. 45, pp. 971–978, Jun. 2020.

[4] N. Nigam, D. P. Singh, and J. Choudhary, "A review of different components of the intelligent traffic management system (ITMS)," *Symmetry*, vol. 15, no. 3, p. 583, Feb. 2023.

[5] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," 2019, *arXiv:1905.05055*.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds., Red Hook, NY, USA: Curran Associates, May 2012, pp. 84–90.

[7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[8] F. Rasheed, K. A. Yau, R. Md. Noor, C. Wu, and Y.-C. Low, "Deep reinforcement learning for traffic signal control: A review," *IEEE Access*, vol. 8, pp. 208016–208044, 2020.

[9] F. Puccetti, L. Cinelli, M. Molteni, L. Gozzini, U. Casiraghi, L. A. Barbieri, E. Treppiedi, A. Cossu, R. Rosati, and U. Elmore, "Impact of imaging magnification on colorectal surgery: A matched analysis of a single tertiary center," *Techn. Coloproctology*, vol. 27, no. 11, pp. 1057–1063, Nov. 2023.

[10] A. Krishna, N. Pendkar, S. Kasar, U. Mahind, and S. Desai, "Advanced video surveillance system," in *Proc. 3rd Int. Conf. Signal Process. Commun. (ICPSC)*, May 2021, pp. 558–561.

[11] M. Schubin, "Why 4k: Vision & television," in *Spring Technical Forum of CableLabs–NCTA SCTE, National Cable Television Association*, 2012.

[12] R. Bohush, G. Ma, Y. Weichen, and S. Ablameyko, "Object detection in video surveillance based on multiscale frame representation and block processing by a convolutional neural network," *Pattern Recognit. Image Anal.*, vol. 32, no. 1, pp. 1–10, Mar. 2022.

[13] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 843–852.

[14] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, Sep. 2013.

[15] M. Z. Islam, M. M. Islam, and A. Asraf, "A survey on deep learning techniques for video anomaly detection," *ACM Comput. Surveys (CSUR)*, vol. 53, no. 5, pp. 1–32, 2020.

[16] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *Int. J. Comput. Vis.*, vol. 128, no. 2, pp. 261–318, Feb. 2020.

[17] Y. Zhang, X. Jiang, X. Gao, Z. Zhao, and X. Yan, "Real-time traffic light detection with high definition map in various illumination conditions," *IEEE Access*, vol. 8, pp. 82824–82834, 2020.

[18] C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao, "YOLOv9: Learning what you want to learn using programmable gradient information," 2024, *arXiv:2402.13616*.

[19] G. Jocher, A. Chaurasia, and A. Stoken, "YOLOv8: A high-performance single-stage object detector," 2023, *arXiv:2301.01740*.

[20] C.-Y. Wang, A. Bochkovskiy, and H.-Y. Mark Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 2022, *arXiv:2207.02696*.

[21] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[22] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9626–9635.

[23] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Computer Vision–ECCV* (Lecture Notes in Computer Science), vol. 9905, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., Springer, 2016, pp. 21–37. [Online]. Available: https://www.wikidata.org/entity/Q60638633

[24] Y. Zhao, W. Lv, S. Xu, J. Wei, G. Wang, Q. Dang, Y. Liu, and J. Chen, "DETRs beat YOLOs on real-time object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 16965–16974.

[25] S. Du, M. Ibrahim, M. Shehata, and W. Badawy, "An overview of using direction information in postal address recognition," *IEEE Access*, vol. 1, pp. 765–776, 2013.

[26] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2015.

[27] Y. Tian, K. Pei, S. Jana, and B. Ray, "Deep learning for image-based intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 6, pp. 2211–2225, Jul. 2018.

[28] Z. Chang, T. Lei, X. Zhou, T. Ristaniemi, and Z. Shi, "Energy-efficient optimization for wireless video transmission in intelligent transportation systems," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 13435–13449, Jun. 2020.

[29] A. Parvaiz, M. A. Khalid, R. Zafar, H. Ameer, M. Ali, and M. M. Fraz, "Vision transformers in medical computer vision—A contemplative retrospection," *Eng. Appl. Artif. Intell.*, vol. 122, Jun. 2023, Art. no. 106126.

[30] N. Micallef, C. J. Debono, D. Seychell, and C. Attard, "Automatic detection of COVID-19 pneumonia in chest computed tomography scans using convolutional neural networks," in *Proc. IEEE 21st Medit. Electrotechnical Conf. (MELECON)*, Jun. 2022, pp. 1118–1123.

[31] N. Micallef, D. Seychell, and C. J. Bajada, "Exploring the U-Net++ model for automatic brain tumor segmentation," *IEEE Access*, vol. 9, pp. 125523–125539, 2021.

[32] M. Schembri and D. Seychell, "Small object detection in highly variable backgrounds," in *Proc. 11th Int. Symp. Image Signal Process. Anal. (ISPA)*, Sep. 2019, pp. 32–37.

[33] D. Pisani and D. Seychell, "Detecting litter from aerial imagery using the SODA dataset," in *Proc. IEEE 22nd Medit. Electrotechnical Conf. (MELECON)*, Jun. 2024, pp. 897–902.

[34] D. Pisani, D. Seychell, C. J. Debono, and M. Schembri, "SODA: A dataset for small object detection in UAV captured imagery," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2024, pp. 151–157.

[35] S. Grigorescu, B. Trasnea, T. Cocias, and G. Macesanu, "A survey of deep learning techniques for autonomous driving," *J. Field Robot.*, vol. 37, no. 3, pp. 362–386, Apr. 2020.

[36] S. Sivaraman and M. M. Trivedi, "Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1773–1795, Dec. 2013.

[37] J. Azimjonov, A. Özmen, and T. Kim, "A nighttime highway traffic flow monitoring system using vision-based vehicle detection and tracking," *Soft Comput.*, vol. 27, no. 19, pp. 13843–13859, Oct. 2023.

[38] M. Fernández-Sanjurjo, B. Bosquet, M. Mucientes, and V. M. Brea, "Real-time visual detection and tracking system for traffic monitoring," *Eng. Appl. Artif. Intell.*, vol. 85, pp. 410–420, Oct. 2019.

[39] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *Proc. IEEE*, vol. 111, no. 3, pp. 257–276, 2023.

[40] B. Garcia-Garcia, T. Bouwmans, and A. J. R. Silva, "Background subtraction in real applications: Challenges, current models and future directions," *Comput. Sci. Rev.*, vol. 35, Feb. 2020, Art. no. 100204.

[41] T. Q. Vinh and N. T. N. Anh, "Real-time face mask detector using YOLOv3 algorithm and Haar cascade classifier," in *Proc. Int. Conf. Adv. Comput. Appl. (ACOMP)*, Nov. 2020, pp. 146–149.

[42] F. M. J. M. Shamrat, A. Majumder, P. R. Antu, S. K. Barmon, I. Nowrin, and R. Ranjan, "Human face recognition applying Haar cascade classifier," in *Proc. Pervasive Comput. Social Networking (ICPCSN)*. Springer, 2022, pp. 143–157.

[43] L. Zhang, J. Wang, and Z. An, "Vehicle recognition algorithm based on Haar-like features and improved AdaBoost classifier," *J. Ambient Intell. Humanized Comput.*, vol. 14, no. 2, pp. 807–815, Feb. 2023.

[44] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, May 2005, pp. 886–893.

[45] B. Bhattarai, R. Subedi, R. R. Gaire, E. Vazquez, and D. Stoyanov, "Histogram of oriented gradients meet deep learning: A novel multi-task deep network for 2D surgical image semantic segmentation," *Med. Image Anal.*, vol. 85, Apr. 2023, Art. no. 102747.

[46] Z. Chen, K. Chen, and J. Chen, "Vehicle and pedestrian detection using support vector machine and histogram of oriented gradients features," in *Proc. Int. Conf. Comput. Sci. Appl.*, Dec. 2013, pp. 365–368.

[47] M. Hanzla, S. Ali, and A. Jalal, "Smart traffic monitoring through drone images via Yolov5 and Kalman filter," in *Proc. 5th Int. Conf. Advancements Comput. Sci. (ICACS)*, Feb. 2024, pp. 1–8.

[48] S. Rakesh, N. P. Hegde, M. V. Gopalachari, D. Jayaram, B. Madhu, M. A. Hameed, R. Vankdothu, and L. K. S. Kumar, "Moving object detection using modified GMM based background subtraction," *Meas., Sensors*, vol. 30, Dec. 2023, Art. no. 100898.

[49] C. Meng, H. Bao, and Y. Ma, "Vehicle detection: A review," *J. Phys., Conf. Ser.*, vol. 1634, no. 1, Sep. 2020, Art. no. 012107.

[50] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intell. Syst. Appl.*, vol. 13, no. 4, pp. 18–28, Sep. 1998.

[51] H. Bhatt, V. Shah, K. Shah, R. Shah, and M. Shah, "State-of-the-art machine learning techniques for melanoma skin cancer detection and classification: A comprehensive review," *Intell. Med.*, vol. 3, no. 3, pp. 180–190, Aug. 2023.

[52] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Dec. 2001, pp. 5–11.

[53] A. Saffari, C. Leistner, J. Santner, M. Godec, and H. Bischof, "On-line random forests," in *Proc. IEEE 12th Int. Conf. Comput. Vis. Workshops, ICCV Workshops*, Sep. 2009, pp. 1393–1400.

[54] D. Seychell and C. J. Debono, "Efficient object selection using depth and texture information," in *Proc. Vis. Commun. Image Process. (VCIP)*, Nov. 2016, pp. 1–4.

[55] Y. Xu, G. Yu, Y. Wang, X. Wu, and Y. Ma, "A hybrid vehicle detection method based on viola-jones and HOG + SVM from UAV images," *Sensors*, vol. 16, no. 8, p. 1325, Aug. 2016.

[56] C. Zhao, W. Chen, Z. Zhao, and J. Liu, "An RGBD data based vehicle detection algorithm for vehicle following systems," in *Proc. IEEE 8th Conf. Ind. Electron. Appl. (ICIEA)*, Jun. 2013, pp. 1506–1511.

[57] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, May 2009.

[58] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A survey of convolutional neural networks: Analysis, applications, and prospects," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 12, pp. 6999–7019, Dec. 2022.

[59] M. Jogin, Mohana, M. S. Madhulika, G. D. Divya, R. K. Meghana, and S. Apoorva, "Feature extraction using convolution neural networks (CNN) and deep learning," in *Proc. 3rd IEEE Int. Conf. Recent Trends Electron., Inf. Commun. Technol. (RTEICT)*, May 2018, pp. 2319–2323.

[60] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.* Springer, Jan. 2020, pp. 213–229.

[61] M. Bartolo, D. Seychell, and J. Bajada, "Integrating saliency ranking and reinforcement learning for enhanced object detection," 2024, *arXiv:2408.06803*.

[62] M. Bartolo and D. Seychell, "Correlation of object detection performance with visual saliency and depth estimation," 2024, *arXiv:2411.02844*.

[63] D. Seychell and C. J. Debono, "Ranking regions of visual saliency in RGB-D content," in *Proc. Int. Conf. 3D Immersion (IC3D)*, Dec. 2018, pp. 1–8.

[64] J. C. Caicedo and S. Lazebnik, "Active object localization with deep reinforcement learning," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2488–2496.

[65] A. Asvadi, L. Garrote, C. Premebida, P. Peixoto, and U. J. Nunes, "DepthCN: Vehicle detection using 3D-LiDAR and ConvNet," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 1–6.

[66] H. Wang, Y. Yu, Y. Cai, X. Chen, L. Chen, and Q. Liu, "A comparative study of state-of-the-art deep learning algorithms for vehicle detection," *IEEE Intell. Transp. Syst. Mag.*, vol. 11, no. 2, pp. 82–95, May 2019.

[67] Y. Chen and Z. Li, "An effective approach of vehicle detection using deep learning," *Comput. Intell. Neurosci.*, vol. 2022, Jul. 2022, Art. no. 2019257.

[68] Y. Cai, X. Sun, H. Wang, L. Chen, and H. Jiang, "Night-time vehicle detection algorithm based on visual saliency and deep learning," *J. Sensors*, vol. 2016, pp. 1–7, May 2016.

[69] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 142–158, Jan. 2016.

[70] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[71] Q. Fan, L. Brown, and J. Smith, "A closer look at faster R-CNN for vehicle detection," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2016, pp. 124–129.

[72] H. Nguyen, "Improving faster R-CNN framework for fast vehicle detection," *Math. Problems Eng.*, vol. 2019, no. 1, Jan. 2019, Art. no. 3808064.

[73] N. Arora, Y. Kumar, R. Karkra, and M. Kumar, "Automatic vehicle detection system in different environment conditions using fast R-CNN," *Multimedia Tools Appl.*, vol. 81, no. 13, pp. 18715–18735, May 2022.

[74] M. Carranza-García, J. Torres-Mateo, P. Lara-Benítez, and J. García-Gutiérrez, "On the performance of one-stage and two-stage object detectors in autonomous vehicles using camera data," *Remote Sens.*, vol. 13, no. 1, p. 89, Dec. 2020.

[75] T. Liang, H. Bao, W. Pan, and F. Pan, "Traffic sign detection via improved sparse R-CNN for autonomous vehicles," *J. Adv. Transp.*, vol. 2022, Mar. 2022, Art. no. 3825532.

[76] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[77] M. Hussain, "YOLOv1 to v8: Unveiling each variant—A comprehensive review of YOLO," *IEEE Access*, vol. 12, pp. 42816–42833, 2024.

[78] S. Du, W. Pan, N. Li, S. Dai, B. Xu, H. Liu, C. Xu, and X. Li, "TSD-YOLO: Small traffic sign detection based on improved YOLO v8," *IET Image Process.*, vol. 18, no. 11, pp. 2884–2898, Sep. 2024.

[79] Y. Wang, S. Xu, P. Wang, L. Liu, Y. Li, and Z. Song, "Vehicle detection algorithm based on improved RT-DETR," *J. Supercomput.*, vol. 81, no. 1, pp. 1–23, Jan. 2025.

[80] P. Mohandas, "DETR-SPP: A fine-tuned vehicle detection with transformer," *Multimedia Tools Appl.*, vol. 83, no. 9, pp. 25573–25594, Aug. 2023.

[81] Z. Dong, Y. Wu, M. Pei, and Y. Jia, "Vehicle type classification using a semisupervised convolutional neural network," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 2247–2256, Aug. 2015.

[82] L. Yang, P. Luo, C. C. Loy, and X. Tang, "A large-scale car dataset for fine-grained categorization and verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3973–3981.

[83] J. M. Alvarez, M. Salzmann, and N. Barnes, "Data-driven road detection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Mar. 2014, pp. 1134–1141.

[84] X. Zhang, Z. Chen, Q. M. J. Wu, L. Cai, D. Lu, and X. Li, "Fast semantic segmentation for scene perception," *IEEE Trans. Ind. Informat.*, vol. 15, no. 2, pp. 1183–1192, Feb. 2019.

[85] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3D object representations for fine-grained categorization," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, Dec. 2013, pp. 554–561.

[86] J. E. Espinosa, S. A. Velastin, and J. W. Branch, "Motorcycle detection and classification in urban scenarios using a model based on faster R-CNN," 2018, *arXiv:1808.02299*.

[87] T.-Y. Lin, "Microsoft COCO: Common objects in context," in *Computer Vision ECCV 2014* (Lecture Notes in Computer Science), vol. 8693, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., Cham, Switzerland: Springer, 2014, pp. 740–755.

[88] A. Abela, D. Seychell, and M. Bugeja, "Exploring how weak supervision can assist the annotation of computer vision datasets," in *Proc. IEEE 21st Medit. Electrotechnical Conf. (MELECON)*, Jun. 2022, pp. 960–965.

[89] S. Singh, A. Yadav, J. Jain, H. Shi, J. Johnson, and K. Desai, "Benchmarking object detectors with COCO: A new path forward," in *Eur. Conf. Comput. Vis.* Springer, 2024, pp. 279–295.

[90] K. Liu and G. Mattyus, "DLR 3k Munich vehicle aerial image dataset," Tech. Rep., 2015.

[91] X. Wang, X. Zhang, Y. Zhu, Y. Guo, X. Yuan, L. Xiang, Z. Wang, G. Ding, D. Brady, Q. Dai, and L. Fang, "PANDA: A gigapixel-level human-centric video dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3265–3275.

[92] Y. Pang, J. Cao, Y. Li, J. Xie, H. Sun, and J. Gong, "TJU-DHD: A diverse high-resolution dataset for object detection," *IEEE Trans. Image Process.*, vol. 30, pp. 207–219, 2021.

[93] Y. Zhou, L. Liu, H. Zhao, M. López-Benítez, L. Yu, and Y. Yue, "Towards deep radar perception for autonomous driving: Datasets, methods, and challenges," *Sensors*, vol. 22, no. 11, p. 4208, May 2022.

[94] M. Hahner, C. Sakaridis, D. Dai, and L. Van Gool, "Fog simulation on real LiDAR point clouds for 3D object detection in adverse weather," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 15263–15272.

[95] M. Bijelic, T. Gruber, F. Mannan, F. Kraus, W. Ritter, K. Dietmayer, and F. Heide, "Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11679–11689.

[96] S. Galea, D. Seychell, and M. Bugeja, "A survey of intelligent transportation systems based modern object detectors under night-time conditions," in *Proc. 3rd Int. Conf. Intell. Sustain. Syst. (ICISS)*, Dec. 2020, pp. 265–270.

[97] U. Gupta, Y. G. Kim, S. Lee, J. Tse, H. S. Lee, G.-Y. Wei, D. Brooks, and C.-J. Wu, "Chasing carbon: The elusive environmental footprint of computing," in *Proc. IEEE Int. Symp. High-Performance Comput. Archit. (HPCA)*, Feb. 2021, pp. 854–867.

[98] G. Spillo, A. De Filippo, C. Musto, M. Milano, and G. Semeraro, "Towards sustainability-aware recommender systems: Analyzing the trade-off between algorithms performance and carbon footprint," in *Proc. 17th ACM Conf. Recommender Syst.*, Sep. 2023, pp. 856–862.

[99] H. Hua, Y. Li, T. Wang, N. Dong, W. Li, and J. Cao, "Edge computing with artificial intelligence: A machine learning perspective," *ACM Comput. Surveys*, vol. 55, no. 9, pp. 1–35, Sep. 2023.

[100] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, "Federated learning for Internet of Things: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1622–1658, 3rd Quart., 2021.

[101] X. Liu, Y. Wang, D. Yu, and Z. Yuan, "YOLOv8-FDD: A real-time vehicle detection method based on improved YOLOv8," *IEEE Access*, vol. 12, pp. 136280–136296, 2024.

[102] B. R. Chughtai and A. Jalal, "Traffic surveillance system: Robust multiclass vehicle detection and classification," in *Proc. 5th Int. Conf. Advancements Comput. Sci. (ICACS)*, Feb. 2024, pp. 1–8.

MARK BUGEJA received the B.S. degree in creative computing from the University of London, London, U.K., in 2012, and the M.S. degree in artificial intelligence from the University of Malta, Msida, Malta, in 2017, where he is currently pursuing the Ph.D. degree in artificial intelligence. From 2013 to 2017, he was a Resident Academic with the Saint Martin's Institute of Higher Education, Malta. From 2017 to 2020, he was a Research Assistant with the Department of Artificial Intelligence and the Institute of Climate Change and Sustainable Development. Since late 2020, he has been with the Institute of Tourism Studies, Malta, and a Visiting Lecturer with the Department of Artificial Intelligence, University of Malta. He has a number of publications on the subject. During this period, his research work focused on emerging technologies and artificial intelligence. His experience also includes various projects attributed to commercial and research interests in the area of emerging technology, such as virtual reality, augmented reality, and games, as well as being one of the co-leads of the Google Developer Group, Malta. His research interests include computer vision, reinforcement learning, and intelligent transport systems.

MATTHIAS BARTOLO (Graduate Student Member, IEEE) received the B.Sc.I.T. degree (Hons.) in artificial intelligence from the University of Malta, where he is currently pursuing the M.Sc. degree. He is a Research Support Officer with the Department of Artificial Intelligence. He is researching litter detection from aerial imagery. He was also selected as a member of the Google Developers Group, Malta, where he contributed to various projects and initiatives within the group and the university. He is actively involved in various projects related to these areas, contributing to advancements in computer vision techniques and applications. He has co-authored and peer-reviewed several publications. His primary research interests include computer vision, with a specific focus on object detection, saliency ranking, and image processing. In 2024, he was honored with the Dean's List Award from the Faculty of ICT, recognizing his outstanding academic achievements.

MATTHEW MONTEBELLO (Senior Member, IEEE) received the B.Ed. degree (Hons) from the University of Malta, in 1990, the master's and Ph.D. degrees from Cardiff University, Wales, in 1996 and 1998, respectively, the Ph.D. degree in computer science, in 1999, and the M.A. and Ed.D. (Higher Education) degrees with a specialization in the application of artificial intelligence to e-learning, in 2009 and 2016, respectively. He was already heavily involved in education in secondary schools. Having obtained extensive teaching experience and having been involved with the introduction of computer laboratories through the Ministry of Education, he proceeded to follow the computer science domain. He was offered a visiting academic status at the University of Illinois at Urbana–Champaign where he collaborated with the Computer Science Department and the College of Education on numerous projects and research initiatives. In May 2018, he was appointed as an Adjunct Professor with the University of Illinois at Urbana–Champaign. In 2019, he was reappointed as the Head of the Department. He is currently a Full Professor with the Department of Artificial Intelligence, Faculty of ICT, University of Malta. He also heads the Agent Technology Research Group at the departmental level, as well as coordinates a number of interest groups with the Faculty of ICT. In 2017, he published a Springer monograph entitled "AI-injected e-Learning." He published his second Springer monograph entitled "Ambient Intelligent Classrooms." He edited an IGI-Global Handbook of Research on Digital Learning while also in the process of authoring his third Springer monograph on Digital Learners.

DYLAN SEYCHELL (Senior Member, IEEE) received the B.Sc.I.T. degree (Hons.) in computer science and artificial intelligence, the M.Sc. degree in artificial intelligence, and the Ph.D. degree in computer engineering, specialized in computer vision. From 2011 to 2019, he also lectured at the St. Martin's Institute of Higher Education, an affiliate of the University of London. In 2015, he was selected to lead Malta's Google Developers Group. In March 2023, he was appointed as a Technical Expert with Malta Digital Innovation Authority. He is currently a Resident Academic with the Department of Artificial Intelligence, University of Malta, specializing in computer vision. He has been actively involved in research, since 2010, during which he has published several international peer-reviewed papers on the application of AI, computer vision, and UX design in various domains. He also supervised various B.Sc. and M.Sc. dissertations in different universities and programs. He served as a member for the Maltese Government's National AI Task Force. He was awarded international awards for his work, including the Gold Seal for e-Excellence at CeBit, the first prize in the European Space Agency Satellite Navigation Competition, in 2010, and the Runner-Up, in 2017.