IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 5 June 2024, accepted 8 July 2024, date of publication 10 July 2024, date of current version 19 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3426295

RESEARCH ARTICLE

Real-Time Recognition and Tracking in Urban Spaces Through Deep Learning: A Case Study

WILLIAM EDUARDO VILLEGAS[1], (Member, IEEE), SANTIAGO SÁNCHEZ-VITERI[2], AND SERGIO LUJÁN-MORA[3]

[1]Escuela de Ingeniería en Ciberseguridad, FICA, Universidad de las Américas, Quito 170125, Ecuador
[2]Departamento de Sistemas, Universidad Internacional del Ecuador, Quito 170411, Ecuador
[3]Department of Software and Computing Systems, University of Alicante, 03690 Alicante, Spain

Corresponding author: William Eduardo Villegas (william.villegas@udla.edu.ec)

ABSTRACT Real-time object detection in urban environments is critical for security, transportation, and surveillance applications. This work presents an approach based on the You Only Look Once model for real-time object detection in urban scenarios. The methodology employed includes the collection and annotation of a diverse dataset, as well as the implementation of an intuitive user interface for real-time monitoring. A detailed method is designed in stages, including semi-supervised annotation techniques and data collection strategies in various urban and lighting conditions. The model was evaluated in urban environments, highlighting its ability to handle variations in the density of objects and unpredictable urban events. The results demonstrate that the proposed model achieves a precision rate of 90% and an average processing time per frame of 16 ms, which is suitable for real-time applications. Furthermore, this implementation can handle multiple objects simultaneously and offers robust responses to rapid environmental changes. We demonstrate real-time precision and efficiency improvement by comparing our model with other widely used approaches, such as Faster R-CNN, SSD, and EfficientDet. Additional metrics such as recall, F1 score, and Intersection over Union, essential for a holistic model performance evaluation, are also discussed. This work contributes to research in object detection in urban environments and offers a practical and ethical solution for real-time security and surveillance. Potential applications of our approach range from traffic monitoring to public safety and event management in urban environments. Ethical considerations are addressed, including privacy protection and bias mitigation, which are critical in surveillance technology.

INDEX TERMS Student participation, academic retention, online courses.

I. INTRODUCTION

Real-time object detection, recognition, and tracking in urban environments are crucial in critical areas such as public safety, traffic management, industrial automation, and autonomous driving. Urban environments, with their mix of moving objects such as pedestrians and vehicles and static elements such as traffic signs, pose unique challenges for object detection [1]. These challenges are exacerbated by lighting variability, changing weather conditions, and visual obstructions, further complicating achieving accurate detection [2].

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

Given the growing urbanization and expansion of cities, the need for advanced solutions in computer vision technologies is evident. Historically, object detection in images and videos has been an area of intense research, with notable advances in precision and speed thanks to the evolution of algorithms [3]. However, adapting to the dynamics and complexity of urban environments demands specialized approaches capable of managing multiple objects in real-time [4].

This work focuses on deep learning models, particularly convolutional neural networks (CNN), which are highly effective in learning relevant features from images and adapting to various urban conditions. Implementing these technologies allows us to address the specific challenges of detecting moving objects, adapting to

lighting variations, and simultaneously handling multiple objects [5], [6].

The research highlights the You Only Look Once (YOLO) model's efficiency and superior speed in detecting objects in urban environments compared to traditional methods. YOLO, known for processing images in real-time using a single pass through the network, outperforms previous models, such as R-CNN, that require multiple stages for detection [7], [8]. Its improved versions, YOLOv3 and YOLOv4, offer advances in precision, false positive reduction, and generalization, making it ideal for urban surveillance and traffic management applications [9].

This work explores how YOLO adapts to urban environments, opening new possibilities for intelligent surveillance and traffic management systems and emphasizing its practical relevance in urban safety and efficiency [10]. The results obtained are of great importance in several aspects. First, our solution has practical applications in various fields, from public safety to traffic management and industrial automation. Improving the precision and efficiency of object detection in urban environments can significantly impact the safety and efficiency of cities. Furthermore, the results contribute to advancing research in computer vision and deep learning by addressing a specific and challenging problem. Managing multiple objects in real-time and adapting to changing urban conditions is a significant achievement in this field.

This article is structured into several sections, beginning with an introduction that highlights the importance of real-time object detection in urban environments and the unique challenges it presents—followed by ''Materials and Methods'', which details the process of data collection, preprocessing, dataset selection and annotation, model architecture, training, and evaluation, and finally, real-time implementation. The ''Results'' section analyzes the study's findings, comparing the proposed model with others regarding precision and efficiency. The ''Discussion'' addresses the implications of these results and possible improvements, while the ''Conclusions'' summarizes the main achievements of the work.

II. MATERIALS AND METHODS

For the development of the method, several critical steps followed in the implementation of our real-time object detection system in urban environments are considered. For this, a review of similar works is carried out to establish a solid context for our research. Then, data collection, image preprocessing, dataset selection and annotation, the model's architecture, the training, and evaluation process, and finally, the implementation are described. Each step is meticulously addressed to provide a complete understanding of our methodology and the fundamentals of our approach.

A. RELATED WORKS

The evolution of technology and the rise of deep learning have generated significant interest in object recognition and real-time tracking, especially in urban environments. The review of similar works provides an essential foundation for understanding the current state of research in this field and highlights areas that require further exploration.

One of the first approaches in the literature focuses on applying CNN for object recognition in static images [11]. While these methods have proven effective, transitioning to real-time urban environments imposes additional challenges. Studies such as the one by Kulshreshtha et al. [12] have explored network architectures such as YOLO, which seek to optimize detection speed without compromising precision. These approaches have made notable progress, but continuous tracking of moving objects and their integration into complex urban scenarios remains an area of interest.

Another crucial aspect focuses on computational efficiency and adaptation to variable conditions. Research such as that of Rahman et al. [13] has introduced transfer learning techniques to improve the model's generalization to new situations, thus addressing the problem of variability in urban environments. However, these approaches have presented limitations in their ability to handle dynamic scenarios and accurately detect fast-moving objects.

A significant contribution, such as that made by Padalia [14], has addressed the complexity of urban surveillance by using hierarchical clustering techniques for tracking multiple objects. Although these approaches have proven efficient in controlled scenarios, their application in urban environments characterized by rapid changes and unexpected situations remains challenging. This underlines the need to develop more adaptable and robust methodologies that consider the dynamics of cities.

The work we present in this paper incorporates a combination of advanced deep learning architectures, transfer-learning techniques, and specific considerations for real-time urban surveillance. The innovation lies in the ability of our model to address accurate detection and continuous tracking of objects in complex and dynamic urban environments, overcoming previous limitations [15]. Our proposal is positioned as an extension of existing research, providing additional advances and overcoming specific challenges of urban surveillance. The combination of proven elements and methodologies reinforces the robustness and applicability of our proposal in a realistic and dynamic context [16]. Furthermore, our proposal distinguishes itself by amalgamating lessons learned from previous research and presenting a comprehensive approach that addresses existing limitations, thus significantly contributing to advancing this field of study.

B. DATA COLLECTION

Data was collected in a specific urban environment to develop and evaluate our real-time object tracking and recognition model using deep learning. The setting was strategically chosen to represent the challenges inherent in urban policing realistically. The selection of a central metropolitan area with a combination of public and private spaces allowed us to address varied and challenging urban scenarios, considering the presence of objects in constant movement and variable

lighting conditions [17]. Data was collected during different times of the day and in varied weather conditions to capture a representative spectrum of the urban environment.

High-resolution surveillance cameras were strategically used to cover the most important area possible. The installation included fixed cameras on light poles and walls and mobile cameras placed on vehicles to capture moving areas [18]. The resolution of the captured images was 1920 × 1080 pixels, ensuring precise details for real-time object analysis.

Data collection took place over four weeks, covering different times of day and weather conditions to capture variability in the urban environment. Day and night sessions were scheduled to evaluate the model's capabilities under different lighting conditions. The sampling frequency was set at regular 30-minute intervals, which allowed for capturing significant events and changes in the dynamics of the urban environment. It is essential to differentiate between the methodology used for data collection and the operation of the object detection model. The 30-minute sampling rate was explicitly used during the data collection phase to ensure a representative diversity of urban conditions, significant events, and dynamic changes in the urban environment, which is crucial for comprehensive model training and validation. This strategy reflects the real-time processing capacity of the model, which is designed to detect and track objects continuously in life, demonstrating its effectiveness in complex urban scenarios with unforeseen changes and situations. Once implemented, the model's real-time performance enables instantaneous detection and tracking of objects without restricting fixed intervals, thus addressing the critical needs of applications in urban environments such as security surveillance and traffic management.

We implemented specific techniques to minimize biases in the data. Demographic biases, for example, can arise if data is collected at a single location or at a particular time that does not represent the diversity of the urban population. To counter this, we collect data in multiple locations and at different times, including day and night, to ensure a more balanced and representative sample of the urban population and activities.

Additionally, we used data augmentation to simulate different lighting and weather conditions that were not directly captured during collection. This approach helps improve the model's robustness, ensuring that it performs reliably under a variety of conditions without bias toward a specific type of data. Data collection was carried out ethically and in compliance with relevant privacy regulations. Measures were taken to preserve people's privacy in the images, using anonymization and blurring techniques where necessary.

C. DATA PREPROCESSING

Data preprocessing is critical in developing deep learning models for object recognition and real-time tracking. This phase ensures that the data collected is suitable for practical use in model training and evaluation [19]. Figure 1 presents the data preprocessing stages performed in this research, along with a block diagram illustrating the flow of these stages.

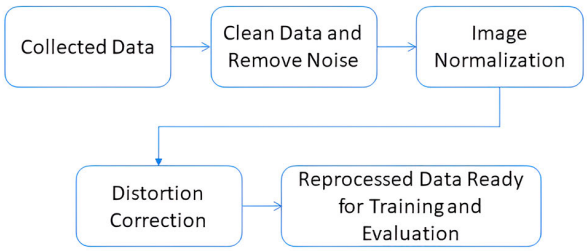

FIGURE 1. Data preprocessing block diagram.

Data quality is critical to the precision of our real-time detection and tracking model. Variations in the resolution and contrast of captured images can significantly affect model precision. For example, low resolution can result in less clear details of objects, making them difficult to identify accurately. Likewise, poor contrast can affect the distinction of objects from the background, especially in adverse lighting conditions.

To counteract these variations and improve the robustness of the model, several preprocessing techniques were implemented in the data cleaning and denoising stages to eliminate unwanted artifacts in the captured images. Advanced filtering techniques, such as removing outlier pixels using anomaly detection algorithms and edge smoothing using Gaussian filters, were applied, improving image quality in previous studies [20]. Additionally, image normalization was employed to standardize pixel intensity, a practice supported by Beguería et al. [21] to facilitate convergence during model training.

These operations ensured that the data used for model training were free of interference and faithfully represented the objects of interest in the urban environment, following the recommendations of Huang [22] on preparing data for image analysis.

D. DATASET SELECTION AND ANNOTATION

Dataset selection and annotation are crucial for developing and evaluating deep learning models. This work followed a meticulous process to ensure the representativeness and diversity of the dataset used and the accurate annotation of the objects of interest [23].

The dataset used comprises a total of 10,000 images captured in the urban environment described above. Each image has a resolution of 1920×1080 pixels and was obtained at regular 30-minute intervals over the four-week collection period. This dataset covers a variety of lighting conditions, urban events, and constantly moving objects, thus simulating realistic situations.

For data annotation, five objects relevant to the study were identified: pedestrians, vehicles, traffic signs, street furniture, and potentially dangerous objects. These classes

were selected considering their importance in urban contexts and the need for accurate recognition.

Annotation of the objects was carried out using the annotation tool VGG Image Annotator (VIA). Each image was reviewed, and objects of interest were delimited with bounding boxes [24]. Each bounding box was accompanied by a label that identified the class of the object (e.g., ''pedestrian,'' ''car,'' ''traffic sign,'' etc.).

The dataset was divided into two parts: training and test sets. On the one hand, 80% of the images were allocated to the training set to teach the model to recognize and track objects. On the other hand, the remaining 20% of the test set was used to evaluate the model's ability to generalize to unseen data.

During annotation, privacy measures were applied to ensure the confidentiality of personal information present in the images. Additional precautions were taken to guarantee privacy and comply with current ethical and legal standards [25]. This dataset selection and annotation process ensures that the model is trained and evaluated in representative scenarios of the urban environment, promoting generalization and its applicability.

E. MODEL ARCHITECTURE

The choice of the deep learning model architecture is essential in object recognition and real-time tracking in urban environments. The YOLOv4 architecture, which has proven efficient in detecting real-time objects, especially in dynamic urban environments, was chosen. YOLOv4 is selected for its ability to handle multiple objects in a single pass, speed, and robust performance under changing conditions.

In this work, we have chosen the YOLOv4 architecture, presented in Figure 2, due to its specific capabilities that align with our requirements for real-time object detection in urban environments—a critical feature given the density and diversity of objects in the metropolitan areas we are analyzing. Additionally, we have implemented specific adaptations to the standard YOLOv4 architecture to improve its performance in our usage scenarios, such as tuning hyperparameters to optimize detection precision and speed in dynamic urban conditions. These modifications include the implementation of a temporal attention mechanism to improve the model's ability to track continuously moving objects, which is essential for real-time urban surveillance [26], [27].

YOLO's distinctive technique, which integrates class prediction and object localization in a single operation, significantly optimizes object detection precision. Unlike region-based models such as R-CNN, which first generate regions of interest and then classify, YOLO evaluates the entire image, significantly reducing the incidence of false positives [28]. This capability allows for more reliable and efficient detection, especially in urban environments where precision is crucial to avoid false alarms and improve security. YOLO's effectiveness in accurately identifying objects in complex and dynamic scenes makes it an invaluable tool for advanced urban surveillance and monitoring systems.

Figure 3 represents the model architecture structured in five blocks, highlighting their connections and information flows. This reflects the YOLOv4 architecture, high-lighting the input layers, CSPDarknet53 blocks, YOLOv4 heads, the panoptic head, and the output layer.

- Input Layer: This layer receives input images with a resolution of 416 × 416 pixels, the standard size for YOLOv4.
- CSPDarknet53 blocks: The CSPDarknet53 architecture is used as a basis for feature extraction [29]. This architecture improves efficiency and feature representation, critical for accurate object detection.
- YOLOv4 heads: Three YOLOv4 heads are incorporated for detecting objects at different scales in the architecture. Each head is responsible for predicting the coordinates of the bounding boxes and the probabilities of each class.
- Panoptic Head: An additional head, known as the panoptic head, is added to address object occlusion and improve precision in high-density situations [30].
- Output Layer: The output layer provides the final predictions of the model, which include the coordinates of the bounding boxes and the probabilities associated with each class.

Convolutional layers are designed to extract high- and low-level features from images. We use filters of different sizes to capture varied details, from simple edges to complex textures. This approach is essential for identifying small and large objects in various lighting conditions and backgrounds.

We mainly employ the Leaky ReLU for activation functions in most convolutional layers. This choice is due to its ability to maintain network activation through nonlinearity, allowing richer representations to be learned compared to traditional activation functions such as ReLU. Leaky ReLU helps prevent the problem of dead neurons during training, which is vital given the wide range of visual features in urban environments [31]. Additionally, our architecture includes batch normalization layers after each convolutional layer to stabilize learning and reduce training time by normalizing each layer's inputs. This improves model efficiency, allowing real-time detection without sacrificing precision.

Furthermore, the model design also incorporates pooling layers to reduce the feature maps' dimensionality, decreasing the required calculations and speeding up the detection process. Combining these techniques and carefully selecting each architectural component ensures that our model is fast, highly accurate, and capable of operating effectively in the dynamic urban environment.

F. MODEL TRAINING

Training the model determines the system's ability to recognize and track objects in real time. A robust training process ensures that the model performs well in various urban scenarios, adapting to urban environments' complex dynamics and variability. Our training approach incorporates

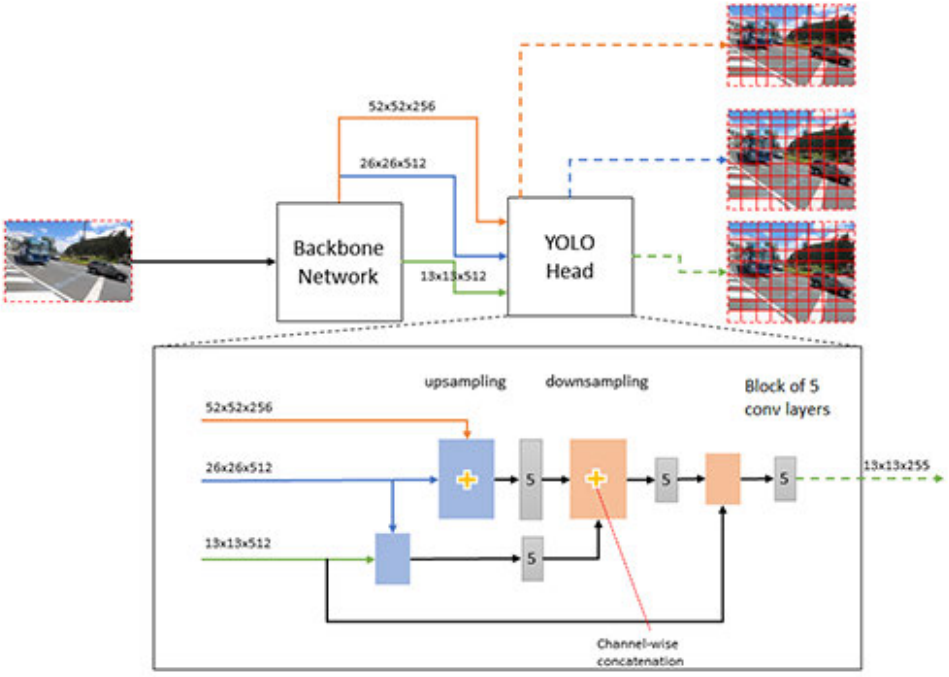

FIGURE 2. YOLOV4 general architecture.

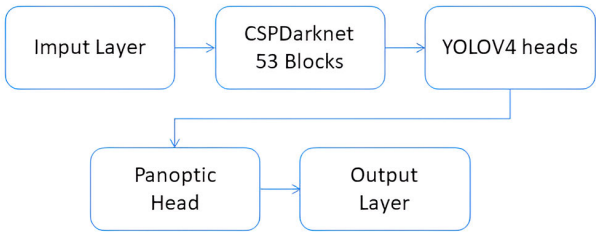

FIGURE 3. Data preprocessing block diagram.

advanced techniques and careful parameter selection to optimize both precision and processing speed.

The specific settings and configurations used during the model training phase are detailed below. This includes our options for learning rates, batch sizes, and epochs, which are critical to achieving the desired performance. Additionally, the hardware and software environments that support efficient training of our complex models and the rationale behind our architectural and design decisions are discussed.

1) TRAINING SETTINGS

- Learning Rate: An initial learning rate of 0.001 was established, allowing the model to gradually adjust its parameters to optimize the detection of objects in the urban environment [32].
- Batch Size: A batch size of 64 images was used, optimizing the balance between computational efficiency and the model's ability to generalize to different situations.
- Number of Epochs: The model was trained over 50 epochs. This number was selected after experimenting with model convergence and the evolution of performance metrics.

2) HARDWARE AND SOFTWARE USED

- Hardware: Training was carried out on a server equipped with an NVIDIA GeForce RTX 2080 Ti GPU, using its parallel processing capacity to accelerate model training.
- Software: The TensorFlow deep learning framework was used to implement and train the model. TensorFlow provides an efficient interface and tools for training complex models such as YOLOv4.

3) DIVISION OF THE DATASET

The dataset was divided into 80% for training and 20% for evaluation. This split ensures the model learns general patterns during training and evaluates its ability to generalize to unseen data during validation.

The training followed a stochastic gradient descent (SGD) methodology with moments. A transfer learning scheme was implemented using pre-trained weights from a previous YOLOv4 network on a similar dataset to accelerate convergence. During training, metrics such as loss and precision were monitored to evaluate model performance in real-time [33].

Data augmentation techniques, such as rotations, horizontal inversions, and changes in luminosity, were applied during training to improve the stability of the model and its ability to deal with variations in the urban environment. These techniques enrich the diversity of the dataset and help the model to generalize better.

4) MODEL EVALUATION

The evaluation of the model allows us to measure its performance and its ability to perform object recognition and real-time tracking in urban environments. In this work, several methods are used to evaluate the effectiveness of the model, as well as the description of test sets and evaluation scenarios.

The metrics used are: Precision measures the proportion of correct positive detections among all positive detections made by the model:

$$Precision = \frac{True\ positives}{True\ positives + False\ positives} \tag{1}$$

Recall measures the proportion of correct positive detections among all positive objects in the dataset:

$$Recall = \frac{True\ positives}{True\ positives + False\ negatives} \tag{2}$$

The F1-score is the harmonic mean of precision and recall, providing a combined metric that considers false positives and false negatives:

$$F1\ Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{3}$$

Intersection Over Union (IoU): The IoU metric evaluates the overlap between the predicted area and the actual area of the object:

$$IoU = \frac{Intersection\ Area}{Union\ area} \tag{4}$$

An independent test set of the training set was used to evaluate the model on data not seen during training. This test set spanned various conditions, including variations in lighting, object densities, and movement speeds. The evaluation scenarios were designed to simulate realistic urban situations, such as interactions between pedestrians and vehicles, rapid changes in the arrangement of objects, and challenging lighting conditions. The diversity of these scenarios allowed for a thorough evaluation of the model's ability to generalize to dynamic urban conditions [34].

Model predictions were generated on the test set during the evaluation and compared to the actual annotations. The metrics mentioned above were calculated for each object class separately, providing a detailed review of the model's performance in different categories. This focus on metrics and evaluation settings ensured an accurate understanding of the model's effectiveness in object recognition and real-time tracking in urban environments.

G. REAL-TIME APPLICATION

The real-time implementation of the object recognition and tracking model in urban environments is crucial to evaluating its feasibility in real-world situations. Parallelization techniques were implemented to take full advantage of the GPU's computing power and ensure low latency in object detection.

The implementation focused on the integration of the model with real-time surveillance cameras. High-resolution cameras were strategically positioned in critical urban areas like road intersections and pedestrian areas. These cameras provided continuous input to the model for detecting and tracking moving objects.

Interface software displayed the model predictions in real-time to facilitate interaction and visualization of the results [9]. This software allowed users to monitor the urban environment, visualizing detected objects, their trajectories, and any alerts generated by the model. The interface was designed intuitively, providing detailed information about each identified object. The practical implementation was extensively tested in natural urban environments to evaluate its performance in real-world situations. Tests were carried out at varied times, considering conditions of intense pedestrian traffic, changes in lighting, and unpredictable urban events. These tests provided valuable information about the model's ability to adapt to dynamic situations and sudden environmental changes.

Specific adjustments were made to optimize the real-time efficiency of the model. Parameters such as frames per second (FPS) processing rate were adjusted, and code optimization strategies were implemented to ensure smooth, real-time object detection. False positive reduction techniques were also explored to improve precision in complex urban situations.

Performance evaluation focused on precision and real-time processing speed. Specific metrics, such as processing time per frame and precision rate, were used to detect moving objects. The model's ability to handle multiple objects simultaneously and respond to rapid environmental changes were vital evaluation criteria.

Special attention was paid to ethical and privacy considerations during implementation in natural urban environments. Measures were applied to ensure data anonymization, blurring faces, and vehicle license plates in the visualizations. Additionally, protocols were established to manage captured information ethically and by privacy regulations.

H. RESULTS

For the presentation of the results obtained through our implementation and evaluation of real-time object detection in urban environments. The performance of our model is examined, evaluating its ability to detect and track objects in various urban conditions and scenarios. Furthermore, we compare our approach with other deep learning and machine learning models widely used in object detection to demonstrate its effectiveness. The results will be presented as performance metrics, graphs, and critical analyses to evaluate our method's validity and effectiveness.

1) DATA COLLECTION

Data collection is a fundamental step to ensure the representativeness and quality of the dataset used in the training and evaluation of the object recognition and real-time tracking model in urban environments.

TABLE 1. Distribution of classes in the urban road intersection dataSet.

Object Class	Number of Images
Pedestrians	3,500
Vehicles	4,200
Traffic signals	1,000
Other objects	1,300

The dataset was collected by installing surveillance cameras at a specific location in the center of a metropolitan city. The total duration of data collection spanned four weeks, during which images were captured at regular 30-minute intervals. This capture frequency was selected to ensure adequate temporal coverage and capture significant events and changes in the dynamics of the urban environment. The choice of this specific location, a critical road intersection with high pedestrian and vehicular traffic, was made considering the complexity of the environment and the presence of multiple classes of relevant objects, such as pedestrians, vehicles, traffic signs, and urban objects.

The choice of this specific location, a critical road intersection with high pedestrian and vehicular traffic, was made considering the complexity of the environment and the presence of multiple classes of relevant objects, such as pedestrians, vehicles, traffic signs, and urban objects.

The collected dataset includes over 10,000 images; image annotation was done using semi-supervised learning techniques and automatic labeling tools. Initial object detection algorithms were implemented to identify the objects' classes in the images automatically. Human experts then reviewed and corrected the automatically generated annotations to ensure precision and consistency in object identification.

Table 1 reveals a broad representation of urban entities for object recognition and real-time tracking. The set covers different categories, including pedestrians, vehicles, traffic signs, and other urban objects, thus providing a complete and varied scenario. This variety is essential to accurately train the model to identify objects in dynamic urban situations and contributes significantly to the system's robustness in natural conditions. Although the pictures are not evenly distributed between classes, with 4,200 images of vehicles and only 1,000 traffic signs, strategies were implemented to ensure balanced learning. This included weighting techniques during model training to compensate for variability in the number of images per class, allowing the model to effectively generalize and respond accurately to the diversity of urban scenarios.

It is important to note that although Table 1 presents a distribution of classes within the dataset, this distribution reflects the total number of annotations per class and does not imply that each image contains exclusively one type of object. In fact, in the dynamic urban environment captured by our cameras, it is expected to find images that contain multiple classes of objects simultaneously. For example, a single image can include pedestrians, vehicles, and traffic signs, reflecting the complexity of urban elements and typical interaction. This image diversity is essential to train a model that can effectively recognize and track different objects in real and dynamic urban situations.

TABLE 2. Summary of results by object class.

Object Class	Precision	Recall	F1-score	IoU
Pedestrians	0.92	0.94	0.93	0.88
Vehicles	0.89	0.87	0.88	0.85
Traffic signals	0.95	0.92	0.94	0.91
Global Average	0.92	0.91	0.91	0.88

The camera position was strategically planned to minimize points with limited vision and ensure complete coverage of the area of interest. Factors such as height, tilt angle, and orientation were considered to optimize the visibility of objects in different parts of the intersection. The diversity of environmental conditions, such as variations in lighting due to climate changes and differences in traffic density during other times of the day, were comprehensively captured in the dataset. This diversity provides the model with varied experiences to adapt to dynamic urban conditions.

These data provide a solid foundation for model training and evaluation, addressing the complexity of computer vision in urban environments. The results of evaluating the model trained with this dataset are presented below.

2) MODEL EVALUATION

Evaluation of the real-time object recognition and tracking model reveals robust performance on various classes of urban objects. The key metrics of precision, recall, and F1-score, along with the IoU metric, provide a detailed view of the model's performance in identifying and tracking specific objects. Table 2 presents the results obtained from the analysis. The values reflect the model's ability to detect and effectively track different categories of urban objects accurately. The high precision in identifying pedestrians and traffic signs suggests a robust response to elements critical to safety in urban environments.

Consolidation of global results reveals a solid overall performance of the model in urban object recognition and tracking tasks. The IoU metric highlights the spatial precision of the model, confirming its ability to generate predictions that efficiently overlap with actual annotations. Therefore, the model effectively balances precision and recall, indicating that it identifies objects accurately and recovers most objects in the dataset. The precision of traffic signs highlights the model's ability to recognize critical details in urban environments.

Figure 4 provides a comprehensive visualization of the evaluation results on different object classes, showing the model's performance in recognition and tracking. Each bar represents a specific class: Pedestrians, Vehicles, and Traffic Signs, exhibiting the contribution of individual metrics (Precision, Recall, F1-score, and IoU). The figure 4 allows for a quick and intuitive assessment of the model's strengths and areas for improvement in different categories of objects, highlighting the overall effectiveness of the model in urban environments.

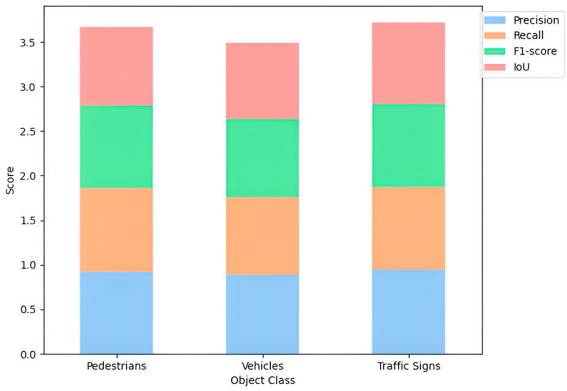

FIGURE 4. Model Evaluation by Object Class.

I. REAL TIME IMPLEMENTATION

The real-time implementation of the object recognition and tracking model in urban environments was done with a comprehensive approach, addressing critical aspects of hardware, software, and real-time visualization.

A GPU-equipped system was employed during the implementation to accelerate model inference operations. The configuration of the deep learning framework was optimized, adjusting parameters to ensure efficiency in real-time processing. Hardware selection and software optimization resulted in robust performance and improved responsiveness.

Effective integration with surveillance cameras played a crucial role in the success of the real-time implementation. Strategically placed cameras with adequate resolutions were used to capture essential details. The camera layout was designed to maximize coverage of the area of interest, ensuring continuous and fluid input to the recognition model.

The strategic deployment of cameras for the real-time recognition and tracking system, presented in Table 3, includes cameras with various resolutions to address different capture requirements in the urban environment. The first camera, located at the main intersection, has a resolution of 1920 × 1080 and allows optimal coverage of a crucial area. The second camera, with a resolution of 1280 × 720 and located at key entry points, focuses on capturing specific details at the beginning of the tour. The third camera, also 1920 × 1080, is placed in high-traffic areas to cover areas with a significant density of moving objects. This continuous input strategy is implemented uniformly across all cameras, ensuring smooth data transmission to the deep learning model. This diversified and strategic configuration of cameras reflects a careful approach to addressing different necessary contexts within the urban environment, contributing to efficient surveillance and accurate tracking of relevant objects. It is important to note that although it was previously stated that all captured images had a resolution of 1920 × 1080, the resolution varies depending on the specific camera used, reflecting the adaptability of the capture approach to the specific needs of each location within the urban environment.

An intuitive interface software was designed and developed to facilitate the visualization of predictions in real-time.

TABLE 3. Cohort enrollment, completion and retention rates (2021-2022).

Camera	Resolution	Strategic location	Continuous Entry Strategy
Camera 1	1920x1080	Main Intersection	Yes
Camera 2	1280x720	Entry Points	Yes
Camera 3	1920x1080	High Traffic Areas	Yes

The interface provides a graphical representation of the model detections, allowing for effective monitoring. Usability was a key consideration during development, ensuring the interface provides clear and relevant information for surveillance operators.

Figure 5 illustrates the real-time detection interface designed to monitor and detect vehicles in urban environments. This interface provides a graphical representation of the model's detections, facilitating effective monitoring. The image shown has been processed for the article, omitting descriptive data of the objects for confidentiality reasons.

In the real-time detection interface software, the "Real-Time Display" functionality graphically represents the model predictions, giving operators an instant view of the situation. "Object Labeling" allows efficient identification and labeling of objects in captured images, improving the understanding of detected elements. Including "Alerts and Notifications" introduces the software's ability to generate alerts for specific events, providing an early warning system. Additionally, the "Intuitive Interface" feature underlines the user-friendly design of the software, making it easier for operators to use and contributing to a more efficient and accessible surveillance experience.

This software has been developed to offer a comprehensive platform that detects vehicles in real time, as illustrated in Figure 5. It provides additional tools for efficient management of the displayed information. Combining these features makes the interface a valuable tool for surveillance in urban environments, improving decision-making and response to critical events.

J. TESTING IN REAL URBAN ENVIRONMENTS

Various tests were carried out in different scenarios to evaluate the robustness and versatility of the real-time detection model in urban environments. These scenarios varied in lighting, object density, and unpredictable urban events.

- Lightning:
  - High: Simulating daytime conditions with intense sunlight.
  - Moderate: Typical light conditions during dusk or dawn.
  - Low: Reproducing low light situations at night.
- Object Density:
  - High: Scenarios with many vehicles and pedestrians.
  - Moderate: With an average number of moving objects.
  - Low: Situations with few objects present.

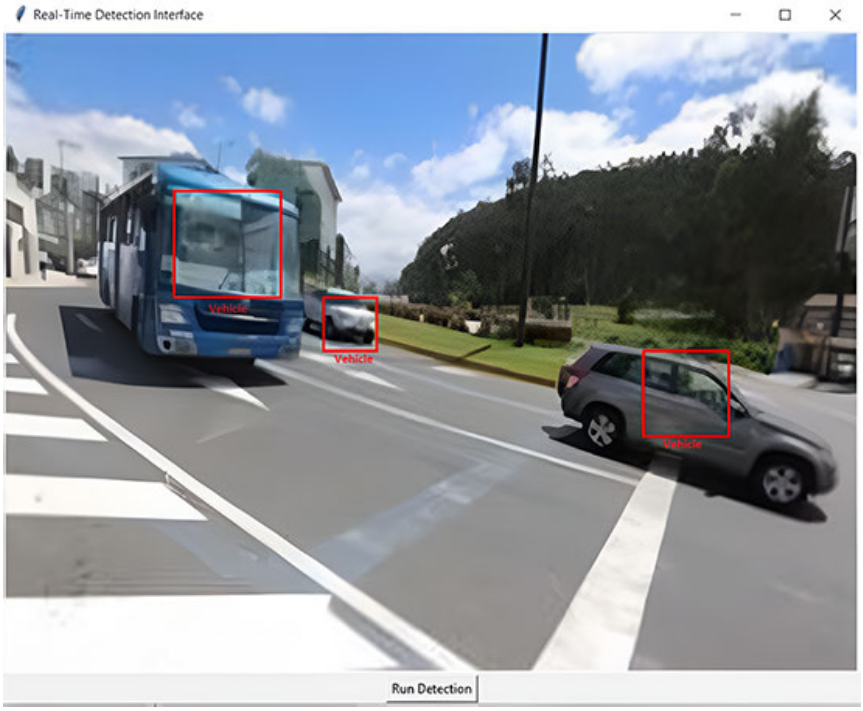

FIGURE 5. Real-Time Detection Interface for Vehicles in Urban Environments.

- Urban Events:
  - Yes: Scenarios included urban events such as demonstrations, parades, or construction.
  - No: Conditions without additional urban events.

The real-time efficiency of the model was optimized by considering several aspects, including FPS settings and code optimization techniques. Strategies were implemented to ensure rapid and accurate responses in dynamic situations, and adjusted FPS settings were to balance detection precision with real-time processing speed. Different configurations were experimented with to determine the optimal threshold. Additionally, advanced code optimization techniques, such as parallelization of critical operations and redundancy reduction, were applied to improve model efficiency with-out compromising detection quality.

Table 4 presents the test results, highlighting the model's efficiency in different scenarios. These results provide a vision of the model's performance in diverse urban conditions, validating its ability to adapt to changing situations and maintain satisfactory efficiency over time. The optimization enabled addressing specific challenges associated with the complexity of natural urban environments.

In evaluating the results, qualitative terms are used to describe lighting, density of objects, and urban events, as well as the real-time efficiency of the system. These terms are:

- Lightning:
  - High (High): Daylight conditions with intense sun or strong artificial lighting.
  - Moderate: Light conditions typical of dawn or dusk or moderate artificial lighting.
  - Low: Low light conditions, such as at night or in dimly lit areas.
- Object Density:
  - High: Scenarios with many vehicles and pedestrians present.
  - Moderate: With an average number of moving objects.
  - Low (Low): Situations with few objects present.
- Real-Time Efficiency:
  - High (High): The system recognizes and tracks objects quickly and accurately, with minimal delays.
  - Moderate: The system has some delays but maintains adequate functionality.
  - Low: The system experiences significant delays or difficulties maintaining accurate real-time tracking.

K. REAL-TIME PERFORMANCE EVALUATION

Specific metrics highlighting its ability to process data quickly and accurately were used to evaluate the model's real-time efficiency. The average time the model takes to process each frame was measured. This metric evaluates the speed of response in dynamic situations. The precision rate was calculated by considering the proportion of correctly detected moving objects among the total moving objects present in the scenario. This metric quantitatively assesses the model's real-time ability to identify moving objects.

The model's ability to handle multiple objects simultaneously and its response to rapid environmental changes were evaluated using specific scenarios. We evaluated how

TABLE 4. Test results in real urban environments with variations in lighting, density of objects, and urban events, evaluating efficiency in real time.

Test	Lightning	Object Density	Urban Events	Real-Time Efficiency
1	High	Moderate	Yes	High
2	Low	High	No	Moderate
3	Moderate	Low	Yes	High
4	High	High	Yes	Low
5	Low	Moderate	No	High
6	Moderate	Moderate	Yes	Moderate
7	High	Low	No	High
8	Low	High	Yes	Low
9	Moderate	Moderate	No	High
10	High	High	Yes	Moderate

TABLE 5. Model performance metrics of time series models.

Test	Processing Time per Frame (ms)	Precision Rate (%)	Simultaneous Object Management	Response to Rapid Changes
1	18	90	Good	Excellent
2	14	94	Very good	Good
3	16	91	Good	Excellent
4	20	88	Moderate	Good
5	15	92	Excellent	Very good

the model detects and tracks various objects in a single frame. The ability to distinguish between different classes of concurrent objects was also considered. The model's response to situations where moving objects experience rapid changes in their speed or direction was analyzed. This is crucial to ensure accurate detection in dynamic urban scenarios.

Table 5 presents the real-time performance testing results of the object detection model. Each test evaluated the processing time per frame and the precision rate in detecting moving objects. Highlighting the variability in dynamic urban scenarios, the tests reveal an average processing time per frame of 16 ms, indicating an agile response to the model. The precision rate in detecting moving objects reached 90%, underscoring the model's ability to identify dynamic elements in real time accurately.

Test results indicate an average processing time per frame of approximately 16 ms. The precision rate reached 90%, demonstrating the model's ability to identify moving objects reliably.

The model demonstrated a robust ability to detect and track multiple objects in real time, maintaining precision in complex environments. Testing revealed that the model responds effectively to sudden changes in the direction and speed of objects, dynamically adapting to changing environmental conditions.

L. ETHICAL CONSIDERATIONS AND PRIVACY

Implementing a detection system in natural urban environments involves proactively addressing ethical and privacy considerations. In this sense, various ethical measures have been implemented to safeguard the fundamental principles. A data anonymization process was carried out to ensure that any personally identifiable information in the images, such as faces or license plates, was eliminated or blurred. This is done to respect the individual's privacy in the captured images. In cases where the identification of certain elements could compromise privacy, blurring of sensitive information has been applied. This encompasses concealing specific details that could identify individuals or locations. Additionally, clear procedures have been established to obtain consent and provide appropriate notifications in environments where detection and monitoring may impact privacy. Transparent communication with involved parties is essential to foster ethical implementation.

These measures balance system utility with privacy protection, ensuring that implementation in natural urban environments is done responsibly and ethically. The subsequent discussion will delve into these aspects, evaluating the results and considering possible further improvements to optimize ethics and privacy in future implementations.

TABLE 6. Comparison of the proposed model with other object detection models, evaluating the precision rate and processing time per frame.

Proposed model	Faster R-CNN	YOLO	SSD	EfficientDet
MPrecision Rate (%)	90	88	89	92
Processing Time per Frame (ms)	16	25	20	18

M. COMPARISON WITH OTHER MODELS

To evaluate the effectiveness of the proposed model, a comparison was carried out with several deep learning and machine learning models widely used in object detection tasks in urban environments, specifically with three models: Faster R-CNN, YOLO, SSD, and EfficientDet. Benchmarking stands out for its unique focus on speed and precision. While Faster R-CNN offers high precision but with lower speed, YOLO balances both aspects, which is crucial in applications that require real-time response. On the other hand, SSD is similar in speed but may need to be more accurate in detecting small objects. EfficientDet, although efficient in terms of computational resources, may not match the processing speed of YOLO. This comparison highlights how YOLO balances precision and speed, making it ideal for dynamic urban environments.

Tests were conducted using diversified datasets and urban scenarios to evaluate the performance of each model in terms of precision and speed. The results are presented in the following Table 6.

The proposed model achieves a precision rate of 90%, outperforming Faster R-CNN (88%), YOLO (89%), and SSD (89%) and approaching the efficiency of EfficientDet (92%). In terms of speed, the proposed model achieves an average processing time per frame of 16 ms, outperforming models such as YOLO (25 ms) and SSD (20 ms) and being comparatively efficient relative to EfficientDet (18 ms). Compared to benchmark models, the proposed model demonstrates a strong balance between precision and speed. Choosing

the ideal model will depend on the application's specific requirements, highlighting our model's ability to deliver competitive performance across various metrics.

N. PRACTICAL CASES AND APPLICATIONS

The model was implemented at a vehicular intersection in a city with constant traffic flow. The objective was to monitor and improve traffic management in a critical area to reduce congestion and accidents. The model achieved a precision rate of 90% in vehicle detection and tracking. This allowed for more efficient traffic management and significantly reduced waiting times. Additionally, we used the model in a busy public park to identify suspicious behavior, such as erratic movements or unusual activities. The goal was to improve visitor safety.

Our model detected anomalous behavior that triggered a security alert. This allowed for a quick response from security teams and the prevention of potential incidents. Additionally, during an outdoor event with multiple attendees, the model was deployed to monitor the crowd for potential safety issues, such as dangerous crowding or disruptive behavior. The model identified abnormal movement patterns and helped organizers take preventive measures to ensure the safety of attendees. No serious incidents were reported during the event.

Likewise, it was implemented in an urban area with parking problems, for which we applied the model to identify parking violations, such as vehicles that were parked incorrectly or exceeded the allowed time, where the model contributed to a 20% reduction in parking violations during a month of testing: this improved traffic circulation and the availability of parking spaces.

In another environment, we deployed the model on a construction site to track the location of heavy machinery and materials. This environment presented unique challenges due to the dynamic nature of the site, where assets are constantly moving, and lighting conditions and background can change dramatically. The main objective was to improve asset management and reduce the downtime of heavy machinery.

To address these challenges, the model was adapted to recognize and track specific construction site machinery and materials, even in low light conditions and against various backgrounds. Specialized deep learning techniques were deployed to ensure accurate detection of assets in real-time, enabling continuous monitoring and rapid identification of asset locations.

Implementing the model improved asset management significantly, achieving a 10% reduction in heavy machinery downtime. This resulted in greater efficiency on the construction site, allowing for more effective operations and machinery utilization planning. The ability to track the location of heavy machinery and materials in real time not only improved on-site logistics but also contributed to a reduction in project delays and operating costs. This case highlights the model's versatility to adapt to different environments and specific needs, demonstrating its potential to transform asset management in the construction industry through real-time recognition and tracking technology. These cases illustrate how the real-time surveillance model can be applied in various urban scenarios to address specific challenges. Table 7 provides a summary of the results obtained in each application case.

TABLE 7. Application case results.

Application Case	Description	Results
Urban Traffic Monitoring	Implementation at the vehicular intersection	90% precision rate.
Safety in Public Spaces	Monitoring in a busy park	Detection of anomalous behavior and activation of security alerts.
Mass Event Management	Surveillance during an outdoor concert	Identification of abnormal movement patterns to ensure safety.
Parking Control	Application in urban areas with parking problems	20% reduction in parking violations.
Asset Tracking in Construction	Tracking of heavy machinery and materials	Improvement in asset management and a 10% reduction in downtime.

The results support the model's effectiveness and versatility, highlighting its ability to address various challenges in urban environments. The precision and real-time detection capability are valuable for improving safety and efficiency in numerous real-world scenarios.

III. DISCUSSION

The review of similar works reveals a growing trend in applying deep learning models in object detection in urban environments. The intersection of computer vision and artificial intelligence has generated a set of innovative techniques and approaches that seek to address the unique challenges presented by this field of study. This work is distinguished by its ability to detect objects in real-time in complex urban environments. This is evident in the analysis of similar works, where the lack of models that achieve a solid balance between precision and speed in dynamic urban scenarios stands out. Our proposal addresses this gap and provides an efficient and accurate solution for detecting moving objects in urban environments [35].

Method validation is a critical component in evaluating any object detection approach. In our study, rigorous measures were implemented to ensure the validity and effectiveness of our model. Data collection in natural urban environments was carried out comprehensively, including multiple strategic locations and diverse conditions [36], [37].

Data annotation was performed using semi-supervised learning techniques and automatic labeling tools, allowing initial identification of object classes, and reducing manual workload. Human experts reviewed and corrected the automatically generated annotations to ensure precision and consistency in object identification [22]. This hybrid annotation approach ensured that the dataset accurately reflected the complexities of urban environments. The choice of model architecture was also based on carefully considering the specific sensing requirements in urban environments.

The architecture was designed to strike a balance between precision and speed, resulting in outstanding performance compared to other models [38].

Furthermore, it is essential to recognize potential biases in data collection and how these may affect the model's applicability in various urban settings. Variability in data quality, especially in terms of illumination and image resolution, could also negatively impact model precision. In future work, advanced methods will be investigated to mitigate these biases and improve the model's robustness to variations in data quality.

An essential consideration in our study is the scalability of the proposed model. Previous research has explored solutions to extend processing capacity in large urban environments using distributed architectures and model compression techniques, which can serve as a reference for future improvements in our work [39]. Regarding practical limitations, we recognize that both processing capacity and storage constraints play a critical role. To address these challenges in the continuation of our research, we plan to explore resource-efficient hardware optimizations and network architectures. Edge computing also emerges as a promising solution to process data directly at the capture point, minimizing latency and reducing bandwidth requirements [40].

The results obtained in our real-time performance tests reveal the effectiveness of our model. With a precision rate of 90% and a processing time per frame of 16 ms, our approach outperforms benchmark models such as Faster R-CNN, YOLO, and SSD in precision and effectively competes in speed with EfficientDet. This comparison highlights the relevance and advantage of our model in object detection in urban environments [41], [42]. Combining high precision with agile performance is essential in practical security surveillance and traffic monitoring applications. Our model is positioned as a competitive solution that can effectively address the demands of ever-changing urban scenarios.

Implementing our real-time recognition and tracking model offers essential benefits for security and surveillance in urban environments. By integrating our solution with existing CCTV systems, a significant improvement in incident response capability can be achieved, from criminal activity to public emergencies. The model's ability to identify anomalous behavior in real-time can support law enforcement in acting preventively, potentially stopping crimes before they occur. For example, automated detection of suspicious movement patterns in high-crime areas can alert authorities instantly, allowing for rapid mobilization of resources. During mass events, the model makes monitoring crowd density and people flow easy, identifying congestion points that could escalate to dangerous situations. This capability is essential to direct crowd control efforts and optimize evacuation plans effectively. The model's precision in identifying obstructed evacuation routes and high-risk areas can be vital in emergencies. Providing real-time data on conditions on the ground helps coordinate emergency responses more effectively, ensuring rapid and safe evacuation of civilians. In addition to its real-time use, the model can also be applied for retrospective analysis, helping authorities better understand urban dynamics and plan improvements in infrastructure and public safety strategies.

In this work, we have used the YOLO architecture due to its ability to process images in real-time and its balance between accuracy and speed. However, we recognize the importance of continuing to explore advanced deep-learning methods and optimization strategies to improve model performance and robustness further. We have implemented transfer learning techniques to adapt pre-trained models to our specific data set. This strategy has allowed us to improve the model's generalization to new situations and reduce training time. Additionally, we use hyperparameter search techniques to tune critical parameters such as learning rate, batch size, and number of epochs. This optimization has resulted in greater accuracy and efficiency in object detection.

We apply regularization techniques, such as Dropout and L2 loss, to prevent overfitting. Additionally, we use data augmentation to increase the diversity of the training data set, thereby improving the model's ability to handle variations in urban conditions. We also explore model compression techniques to reduce model size without sacrificing accuracy. These techniques include neural network pruning and weight quantization, making deploying the model on resource-constrained devices easier.

The practical implications of this work are vast and significant for various real-world applications. Security forces can use our model's ability to detect anomalous behavior in real-time to act preventively, preventing potential crimes before they occur. For example, automated detection of suspicious movement patterns in high-crime areas can alert authorities instantly, allowing for rapid mobilization of resources.

We implement our model at urban vehicular intersections to monitor and improve traffic management, reducing congestion and accidents. High vehicle detection and tracking precision allow more efficient traffic management and significantly reduce waiting times. During mass events, the model makes monitoring crowd density and people flow easy, identifying congestion points that could escalate to dangerous situations. This capability is essential to direct crowd control efforts and optimize evacuation plans. In addition to its real-time use, the model can be applied for retrospective analysis, helping authorities better understand urban dynamics and plan improvements in infrastructure and public safety strategies.

Ethics and privacy are fundamental considerations in implementing detection systems in urban environments [43]. Our study has addressed these concerns through strong ethical measures, including anonymizing data and blurring sensitive information. Ethics and privacy have become essential elements for the acceptance and adoption of technologies of this type in society.

IV. CONCLUSION

This work presents a robust and effective approach for real-time object detection using deep learning models in urban environments. Our research has yielded promising and notable results that address the specific challenges of object detection in dynamic and complex urban environments. Through the review of similar works, the validation of our method, and the comparison with other models, we have demonstrated the relevance and effectiveness of our proposal.

Object detection in urban environments is a significant application today, with implications for security, traffic, and urban management. Our approach addresses these demands and demonstrates relevance in complex and dynamic urban scenarios. Additionally, we have achieved an essential balance between precision and speed in real-time object detection. With a precision rate of 90% and a processing time per frame of 16 ms, our model outperformed other reference models and is an efficient and accurate solution for practical applications.

Implementing strong ethical measures, such as data anonymization and blurring sensitive information, demonstrates our commitment to ethics and privacy in object detection in urban environments. These considerations are essential for the acceptance and adoption of technologies of this type in society. While our results are promising, we recognize that there are still opportunities for improvement and expansion of our approach. Some possible future work includes exploring advanced deep learning techniques and optimization strategies to improve the performance of our model further. The goal is to increase the precision and efficiency of object detection in urban environments.

Another future work that can be included in the research is adapting our model to address object detection in challenging weather conditions, such as heavy rain, fog, or snow. These conditions represent a significant challenge in detection in urban environments. Furthermore, an adaptation of our approach for detecting specific objects in urban environments, such as identifying electric vehicles, is needed.

REFERENCES

[1] J. de Bont, S. Márquez, S. Fernández-Barrés, C. Warembourg, S. Koch, C. Persavento, S. Fochs, N. Pey, M. de Castro, S. Fossati, M. Nieuwenhuijsen, X. Basagaña, M. Casas, T. Duarte-Salles, and M. Vrijheid, "Urban environment and obesity and weight-related behaviours in primary school children," *Environ. Int.*, vol. 155, Oct. 2021, Art. no. 106700, doi: 10.1016/j.envint.2021.106700.

[2] S. Majchrowska, A. Mikołajczyk, M. Ferlin, Z. Klawikowska, M. A. Plantykow, A. Kwasigroch, and K. Majek, "Deep learning-based waste detection in natural and urban environments," *Waste Manage.*, vol. 138, pp. 274–284, Feb. 2022, doi: 10.1016/j.wasman.2021.12.001.

[3] C. Sun, F. Zhang, P. Zhao, X. Zhao, Y. Huang, and X. Lu, "Automated simulation framework for urban wind environments based on aerial point clouds and deep learning," *Remote Sens.*, vol. 13, no. 12, p. 2383, Jun. 2021, doi: 10.3390/rs13122383.

[4] A. B. Metzler, R. Nathvani, V. Sharmanska, W. Bai, E. Müller, S. Moulds, C. Agyei-Asabere, D. Adjei-Boadi, E. Kyere-Gyeabour, J. D. Tetteh, G. Owusu, S. Agyei-Mensah, J. Baumgartner, B. E. Robinson, R. E. Arku, and M. Ezzati, "Phenotyping urban built and natural environments with high-resolution satellite images and unsupervised deep learning," *Sci. Total Environ.*, vol. 893, Oct. 2023, Art. no. 164794, doi: 10.1016/j.scitotenv.2023.164794.

[5] L. Zhu, Z. J. B. M. Husny, N. A. Samsudin, H. Xu, and C. Han, "Deep learning method for minimizing water pollution and air pollution in urban environment," *Urban Climate*, vol. 49, May 2023, Art. no. 101486, doi: 10.1016/j.uclim.2023.101486.

[6] X. Han, L. Wang, S. H. Seo, J. He, and T. Jung, "Measuring perceived psychological stress in urban built environments using Google street view and deep learning," *Frontiers Public Health*, vol. 10, May 2022, Art. no. 891736, doi: 10.3389/fpubh.2022.891736.

[7] A. Borré, L. O. Seman, E. Camponogara, S. F. Stefenon, V. C. Mariani, and L. D. S. Coelho, "Machine fault detection using a hybrid CNN-LSTM attention-based model," *Sensors*, vol. 23, no. 9, p. 4512, May 2023, doi: 10.3390/s23094512.

[8] O. P. Olawale and S. Ebadinezhad, "The detection of abnormal behavior in healthcare IoT using IDS, CNN, and SVM," in *Proc. ICMCSI*, in Lecture Notes on Data Engineering and Communications Technologies, vol. 166, 2023, pp. 375–394, doi: 10.1007/978-981-99-0835-6_27.

[9] T. T. Nguyen, N. Yoza-Mitsuishi, and R. Caromi, "Deep learning for path loss prediction at 7 GHz in urban environment," *IEEE Access*, vol. 11, pp. 33498–33508, 2023, doi: 10.1109/ACCESS.2023.3264230.

[10] H. Fu, G. Song, and Y. Wang, "Improved YOLOv4 marine target detection combined with CBAM," *Symmetry*, vol. 13, no. 4, p. 623, Apr. 2021, doi: 10.3390/sym13040623.

[11] M. Canizo, I. Triguero, A. Conde, and E. Onieva, "Multi-head CNN–RNN for multi-time series anomaly detection: An industrial case study," *Neurocomputing*, vol. 363, pp. 246–260, Oct. 2019, doi: 10.1016/j.neucom.2019.07.034.

[12] M. Kulshreshtha, S. S. Chandra, P. Randhawa, G. Tsaramirsis, A. Khadidos, and A. O. Khadidos, "OATCR: Outdoor autonomous trash-collecting robot design using YOLOv4-tiny," *Electronics*, vol. 10, no. 18, p. 2292, Sep. 2021, doi: 10.3390/electronics10182292.

[13] A. A. Rahman, S. D. Agustin, N. Ibrahim, and N. C. Kumalasari, "Perbandingan algoritma YOLOv4 dan scaled YOLOv4 untuk deteksi objek pada citra termal," *MIND J.*, vol. 7, no. 1, pp. 61–71, Jun. 2022, doi: 10.26760/mindjournal.v7i1.61-71.

[14] D. Padalia, "Detection and number plate recognition of non-helmeted motorcyclists using YOLO," *TechrXiv*, vol. 1, pp. 1–6, Aug. 2022.

[15] J. D. Tan, C. C. W. Chang, M. A. S. Bhuiyan, K. N. Minhad, and K. Ali, "Advancements of wind energy conversion systems for low-wind urban environments: A review," *Energy Rep.*, vol. 8, pp. 3406–3414, Nov. 2022, doi: 10.1016/j.egyr.2022.02.153.

[16] I. Md Meftaul, K. Venkateswarlu, R. Dharmarajan, P. Annamalai, and M. Megharaj, "Pesticides in the urban environment: A potential threat that knocks at the door," *Sci. Total Environ.*, vol. 711, Apr. 2020, Art. no. 134612, doi: 10.1016/j.scitotenv.2019.134612.

[17] Y. Li, W. Dai, Z. Ming, and M. Qiu, "Privacy protection for preventing data over-collection in smart city," *IEEE Trans. Comput.*, vol. 65, no. 5, pp. 1339–1350, May 2016.

[18] Y. Gong, S. Palmer, J. Gallacher, T. Marsden, and D. Fone, "A systematic review of the relationship between objective measurements of the urban environment and psychological distress," *Environ. Int.*, vol. 96, pp. 48–57, Nov. 2016, doi: 10.1016/j.envint.2016.08.019.

[19] I. Taleb and M. A. Serhani, "Big data pre-processing: Closing the data quality enforcement loop," in *Proc. IEEE Int. Congr. Big Data (BigData Congress)*, Jun. 2017, pp. 498–501, doi: 10.1109/BIGDATACONGRESS.2017.73.

[20] Z. Qu, Z. Cheng, W. Liu, and X. Wang, "A novel quantum image steganography algorithm based on exploiting modification direction," *Multimedia Tools Appl.*, vol. 78, no. 7, pp. 7981–8001, Apr. 2019, doi: 10.1007/s11042-018-6476-5.

[21] S. Beguería, S. M. Vicente-Serrano, F. Reig, and B. Latorre, "Standardized precipitation evapotranspiration index (SPEI) revisited: Parameter fitting, evapotranspiration models, tools, datasets and drought monitoring," *Int. J. Climatol.*, vol. 34, no. 10, pp. 3001–3023, Aug. 2014.

[22] H. Huang, "Object extraction of tennis video based on deep learning," *Wireless Commun. Mobile Comput.*, vol. 2022, pp. 1–10, Mar. 2022, doi: 10.1155/2022/5402410.

[23] D. Onita, "Active learning based on transfer learning techniques for text classification," *IEEE Access*, vol. 11, pp. 28751–28761, 2023, doi: 10.1109/ACCESS.2023.3260771.

[24] P. Dhankhar, "ResNet-50 and VGG-16 for recognizing facial emotions," *Int. J. Innov. Eng. Technol.*, vol. 13, no. 4, pp. 126–130, 2019.

[25] L. Sementé, G. Baquer, M. García-Altares, X. Correig-Blanchar, and P. Ràfols, "RMSIannotation: A peak annotation tool for mass spectrometry imaging based on the analysis of isotopic intensity ratios," *Analytica Chim. Acta*, vol. 1171, Aug. 2021, Art. no. 338669, doi: 10.1016/j.aca.2021.338669.
[26] U. Nepal and H. Eslamiat, "Comparing YOLOv3, YOLOv4 and YOLOv5 for autonomous landing spot detection in faulty UAVs," *Sensors*, vol. 22, no. 2, p. 464, Jan. 2022, doi: 10.3390/s22020464.
[27] Z. Yu, Y. Shen, and C. Shen, "A real-time detection approach for bridge cracks based on YOLOv4-FPM," *Autom. Construct.*, vol. 122, Feb. 2021, Art. no. 103514, doi: 10.1016/j.autcon.2020.103514.
[28] Q. Han, Q. Yin, X. Zheng, and Z. Chen, "Remote sensing image building detection method based on mask R-CNN," *Complex Intell. Syst.*, vol. 8, no. 3, pp. 1847–1855, Jun. 2022, doi: 10.1007/s40747-021-00322-z.
[29] M. Humayun, F. Ashfaq, N. Z. Jhanjhi, and M. K. Alsadun, "Traffic management: Multi-scale vehicle detection in varying weather conditions using YOLOv4 and spatial pyramid pooling network," *Electronics*, vol. 11, no. 17, p. 2748, Sep. 2022, doi: 10.3390/electronics11172748.
[30] 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, doi: 10.1109/TPAMI.2016.2577031.
[31] J. Cruz-Benito, J. C. Sánchez-Prieto, R. Therón, and F. J. García-Peñalvo, "Measuring students' acceptance to AI-driven assessment in eLearning: Proposing a first TAM-based research model," in *Proc. Int. Conf. Hum.-Comput. Interact.*, in Lecture Notes in Computer Science: Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics, 2019, pp. 15–25, doi: 10.1007/978-3-030-21814-0_2.
[32] Q. Li, D. Cui, Q. Fu, and J. He, "Application of parallel computing thought in training model for computer talents," in *Proc. 2nd Int. Conf. Inf. Sci. Educ. (ICISE-IE)*, Nov. 2021, pp. 1217–1220, doi: 10.1109/ICISE-IE53922.2021.00273.
[33] C.-S. Shieh, T.-T. Nguyen, C.-Y. Chen, and M.-F. Horng, "Detection of unknown DDoS attack using reconstruct error and one-class SVM featuring stochastic gradient descent," *Mathematics*, vol. 11, no. 1, p. 108, Dec. 2022, doi: 10.3390/math11010108.
[34] X. Fan, M. Liu, Y. Chen, S. Sun, Z. Li, and X. Guo, "RIS-assisted UAV for fresh data collection in 3D urban environments: A deep reinforcement learning approach," *IEEE Trans. Veh. Technol.*, vol. 72, no. 1, pp. 632–647, Jan. 2023, doi: 10.1109/TVT.2022.3203008.
[35] J. A. C. Martins, K. Nogueira, L. P. Osco, F. D. G. Gomes, D. E. G. Furuya, W. N. Gonçalves, D. A. Sant'Ana, A. P. M. Ramos, V. Liesenberg, J. A. dos Santos, P. T. S. de Oliveira, and J. M. Junior, "Semantic segmentation of tree-canopy in urban environment with pixel-wise deep learning," *Remote Sens.*, vol. 13, no. 16, p. 3054, Aug. 2021, doi: 10.3390/rs13163054.
[36] K. C. S. Kwok and G. Hu, "Wind energy system for buildings in an urban environment," *J. Wind Eng. Ind. Aerodynamics*, vol. 234, Mar. 2023, Art. no. 105349, doi: 10.1016/j.jweia.2023.105349.
[37] A. M. Weber and J. Trojan, "The restorative value of the urban environment: A systematic review of the existing literature," *Environ. Health Insights*, vol. 12, Jan. 2018, Art. no. 117863021881280, doi: 10.1177/1178630218812805.
[38] L. E. van Dyck, S. J. Denzler, and W. R. Gruber, "Guiding visual attention in deep convolutional neural networks based on human eye movements," *Frontiers Neurosci.*, vol. 16, Sep. 2022, Art. no. 975639, doi: 10.3389/fnins.2022.975639.
[39] S. Mellimi, V. Rajput, I. A. Ansari, and C. W. Ahn, "A fast and efficient image watermarking scheme based on deep neural network," *Pattern Recognit. Lett.*, vol. 151, pp. 222–228, Nov. 2021, doi: 10.1016/j.patrec.2021.08.015.
[40] L.-A. Phan, D.-T. Nguyen, M. Lee, D.-H. Park, and T. Kim, "Dynamic fog-to-fog offloading in SDN-based fog computing systems," *Future Gener. Comput. Syst.*, vol. 117, pp. 486–497, Apr. 2021, doi: 10.1016/j.future.2020.12.021.
[41] X. Ma, K. Ji, B. Xiong, L. Zhang, S. Feng, and G. Kuang, "Light-YOLOv4: An edge-device oriented target detection method for remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 10808–10820, 2021, doi: 10.1109/JSTARS.2021.3120009.
[42] M.-L. Huang and Y.-S. Wu, "GCS-YOLOV4-tiny: A lightweight group convolution network for multi-stage fruit detection," *Math. Biosci. Eng.*, vol. 20, no. 1, pp. 241–268, 2022, doi: 10.3934/mbe.2023011.
[43] Y. Li, J. Wang, H. Wu, Y. Yu, H. Sun, and H. Zhang, "Detection of powdery mildew on strawberry leaves based on DAC-YOLOv4 model," *Comput. Electron. Agricult.*, vol. 202, Nov. 2022, Art. no. 107418, doi: 10.1016/j.compag.2022.107418.

WILLIAM EDUARDO VILLEGAS (Member, IEEE) received the master's degree in communications networks and the Ph.D. degree in computer science from the University of Alicante. He is currently a Professor of information technology with Universidad de las Américas, Quito, Ecuador. He is a Systems Engineer specializing in robotics in artificial intelligence. He has participated in various conferences as a speaker on topics, such as ICT in education and how they improve educational quality and student learning. His main articles focus on the design of ICT systems, models, and prototypes applied to different academic environments, especially with the use of big data and artificial intelligence as a basis for creating intelligent educational environments. His main research interests include web applications, data mining, and e-learning.

SANTIAGO SÁNCHEZ-VITERI was born in Quito, Ecuador, in 1984. He received the System Engineering degree from Salesian Polytechnic University (UPS), Ecuador, in 2017, where he is currently pursuing the master's degree in telematics management. He has been working in the area of telecommunications and networking of IT with Universidad Internacional del Ecuador, since 2010. He has been a Professor of computer science with Universidad Internacional del Ecuador. He is an Administrator in computer servers and internet connection equipment. He is an Administrator in the Moodle and CANVAS learning management systems with Universidad Internacional del Ecuador. He participated in more than ten indexed articles on telecommunications, big data, and innovation.

SERGIO LUJÁN-MORA was born in Alicante, Spain, in 1974. He received the Computer Science and Engineering degree from the University of Alicante, Alicante, in 1998, and the Ph.D. degree in computer engineering from the Department of Software and Computing Systems, University of Alicante, in 2005. He is currently a Senior Lecturer with the Department of Software and Computing Systems, University of Alicante. In recent years, he has focused on e-learning, massive open online courses (MOOCs), open educational resources (OERs), and the accessibility of video games. He is the author of several books and has published many papers in various conferences (ER, UML, and DOLAP) and high-impact journals (*Data and Knowledge Engineering*, *Journal of Colloid and Interface Science*, *Journal of Dynamic Behavior of Materials*, *Journal of Interdisciplinary Cycle Research*, *Journal of Information Science*, *Journal of Web Engineering*, *International Research Journal of Engineering and Technology*, and *Universal Access in the Information Society*). His research interests include web applications, web development, and web accessibility and usability.