Parking Space Occupancy Monitoring System Using Computer Vision

Svitlana Popereshnyak[1,*,†], Dmytro Chornobryvets[1,†] and Oleksandr Symonenko[2,†]

[1] *National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 37, Prospect Beresteiskyi, Kyiv, Ukraine, 03056*
[2] *Institute of Software Systems of the National Academy of Sciences of Ukraine, 40 Academician Glushkova Avenue, Building 5, Kyiv, Ukraine, 03187*

Abstract
This paper presents the development of an intelligent parking space monitoring system based on computer vision technologies. The study explores approaches for detecting and classifying the status of parking spaces using video streams from surveillance cameras. An algorithm is proposed that combines image processing techniques and machine learning to automatically identify occupied and available parking spots in real time. An experimental evaluation of the system's effectiveness was conducted under various lighting and weather conditions. The obtained results confirm the feasibility of implementing such solutions in urban infrastructure to improve traffic organization and reduce the load on parking facilities.

Keywords
computer vision, parking, image recognition, intelligent transportation systems, machine learning, video analytics, urban infrastructure, Internet of Things, smart city.

1. Introduction

The increase in the number of vehicles in cities leads to significant pressure on infrastructure and exacerbates the problem of parking space shortages. In the central areas of large cities, the inefficient allocation of parking resources becomes particularly noticeable, causing traffic congestion, higher levels of air pollution, and time losses for drivers searching for available parking spots.

Traditional methods of parking management often fail to account for the dynamic nature of demand and do not provide real-time updates, while existing sensor-based solutions tend to be expensive to install and maintain. In contrast, computer vision–based systems that utilize existing or newly deployed video infrastructure offer a scalable and more cost-effective approach, leveraging modern advancements in deep learning for real-time analysis.

Accordingly, there is a growing need for the implementation of information-analytical systems capable of real-time monitoring, tracking, and optimization of parking space usage.

The relevance of this study is driven by the need to enhance the efficiency of parking space management through the use of advanced digital technologies, aligning with the broader strategy of urban digitalization and the development of the "Smart City" concept.

The aim of this work is to improve the efficiency of parking resource management in urban environments, reduce the time spent searching for free spaces, and enhance user convenience

Workshop "Intelligent information technologies" UkrProg-IIT`2025 co-located with 15th International Scientific and Practical Programming Conference UkrPROG'2025 , May 13-14, 2025, Kyiv, Ukraine
* Corresponding author.
† These authors contributed equally.
spopereshnyak@gmail.com (S. Popereshnyak); dmitryfraire17@gmail.com (D. Chornobryvets); morden2k@gmail.com (O. Symonenko)
0000-0002-0531-9809 (S. Popereshnyak); 0009-0006-3647-93181 (D. Chornobryvets); 0009-0001-7443-9492 (O. Symonenko)

through the creation of an intelligent software system for real-time monitoring and management of parking space occupancy, based on computer vision methods. The proposed system is designed to perform automatic vehicle detection, visualize the status of parking spaces, and provide basic user interaction through a configurable interface.

2. Analysis of the subject area

The modern stage of development of society is characterized by relentless urbanization and an increase in the level of motorization of the population, especially in large cities of Ukraine. This gives rise to a complex of transport problems, among which one of the most acute is the shortage and inefficient use of parking space. The daily search for free parking spaces has become a routine problem for millions of drivers, which leads to significant loss of time, excessive fuel consumption, increased emissions of greenhouse gases and harmful substances, increased traffic congestion and increased social tension. Traditional methods of controlling the occupancy of parking lots, which rely on the human factor or simple mechanisms (barriers, manual accounting), do not meet the requirements of a dynamic urban environment and are unable to ensure effective resource management in real time.

The alternative is automated monitoring systems. Although there are solutions based on individual sensors (ultrasonic, magnetic), their deployment is often associated with significant capital costs for equipment and laying communications, as well as subsequent maintenance costs. Against this background, monitoring systems based on computer vision methods are becoming increasingly relevant. The use of video cameras, which are often already part of the security infrastructure, allows for flexible and potentially more cost-effective solutions for collecting and analyzing data on parking space occupancy.

The key tasks for such systems are reliable detection of vehicles in the camera's field of view and their correct correlation with physical parking spaces. However, the implementation of visually-oriented systems faces a number of challenges.

- Variability of lighting conditions: The system must operate stably both day and night, as well as under different weather conditions (sun, shade, rain, snow), which significantly affect image quality.
- Partial occlusions: Vehicles can be partially covered by other cars, columns, trees or pedestrians, which makes it difficult to detect them and determine their exact boundaries.
- Object diversity: The system must correctly identify vehicles of different types, sizes and colors, and ignore extraneous objects.
- Localization accuracy: To determine the occupancy of a specific space, it is necessary not only to detect the car, but also to accurately determine its spatial position relative to the parking space markings.
- Scene dynamics: The system must distinguish stationary parked cars from those that have temporarily stopped or are passing by.

Modern deep learning methods, in particular convolutional neural networks (CNN) and architectures such as YOLO (You Only Look Once), demonstrate high efficiency in real-time object detection tasks and can serve as the basis for solving the problem of vehicle detection. To localize and correlate detected objects with parking spaces, an approach based on the definition of Region of Interest (ROI) corresponding to the geometry of parking spaces is often used.

Despite significant progress, existing solutions often require careful tuning, adaptation to specific operating conditions and may demonstrate insufficient reliability when using simplified methods for determining occupancy (for example, analyzing only the central point of the object). The task of developing comprehensive software solutions that would combine effective detection algorithms, flexible ROI configuration mechanisms and a convenient interface for monitoring and interaction remains relevant.

3. Literature review and identification of research problems

The issue of efficient parking space monitoring has attracted significant attention from researchers in recent years, with various solutions leveraging computer vision and machine learning technologies.

In the work [1] was proposed an advanced system for parking space detection using next-generation computer vision algorithms. It solution emphasizes the integration of deep learning models to improve detection accuracy under different environmental conditions. However, while the system achieves high performance, it largely focuses on controlled environments and lacks extensive validation in dynamically changing real-world settings.

Similarly, in [2] was developed a video-based monitoring framework incorporating both traditional machine learning algorithms and computer vision techniques. It work highlights the benefits of continuous monitoring and predictive analysis for parking occupancy. Nonetheless, the system requires significant computational resources and assumes a stable and high-quality video feed, which can be challenging in urban deployments.

Earlier efforts, ex. [3], explored simpler image processing approaches for detecting available parking spaces. Although it methods were effective in low-complexity environments, they demonstrated limitations when applied to more congested and complex urban areas, where frequent occlusions and environmental variability are common.

The integration of IoT with computer vision, as examined in [4], presents an interesting direction. Its system combines visual data with IoT networks to enhance real-time monitoring capabilities. However, this approach demands a robust IoT infrastructure, which may not always be feasible or cost-effective, especially for retrofitting existing urban environments.

In the domain of specialized parking facilities, such as container drayage at seaports, ex. [5], was investigated the use of AI-driven vision systems to optimize large-scale parking logistics. Although this research demonstrates promising results in specific industrial settings, the techniques and system architectures are not directly transferable to general urban street parking scenarios.

In the work [6] was proposed a comprehensive approach to data processing within smart parking systems as an integral element of smart city infrastructure. This work focuses on optimizing the collection, transmission, and analysis of parking data to support intelligent decision-making processes. The authors highlight the importance of efficient data management for enhancing real-time monitoring, reducing traffic congestion, and improving urban mobility. Although the study presents a well-structured data processing pipeline, it primarily emphasizes the backend data handling aspect, with less attention to advanced real-time visual detection methods based on computer vision, which remain critical for the practical deployment of fully automated parking systems.

Despite notable advancements, several critical gaps remain unaddressed:

- Adaptability to Diverse Environments: Existing systems often assume optimal conditions and are insufficiently robust to varying lighting, weather, and real-world occlusion scenarios common in urban landscapes.
- Cost and Scalability Issues: Many proposed solutions require expensive hardware installations or intensive computational resources, limiting their practical scalability for widespread city-wide deployment.
- Real-Time Data Processing: Ensuring real-time detection, processing, and user-friendly feedback remains a challenge, particularly under high-traffic conditions where quick decision-making is essential.
- User Interaction and Integration: Current systems often lack intuitive interfaces for real-time user interaction and are not well integrated with broader smart city infrastructures.

Given these challenges, there is a clear need for the development of an intelligent, scalable, real-time parking monitoring and management system that leverages advances in deep learning, efficient video analytics, and adaptive user interfaces, while maintaining affordability and ease of deployment. This study aims to address these gaps by proposing a novel computer vision–based solution tailored for dynamic urban environments.

4. Description of the algorithms and neural network architectures used

The parking space monitoring system was implemented using a combination of computer vision algorithms for object detection, geometric analysis for determining the status of spaces, and standard methods for visualization and data management. A key element is a neural network model for vehicle detection.

4.1. Vehicle detection: YOLO architecture

The main task is to detect cars in video frames. For this purpose, a model from the YOLO (You Only Look Once) family was chosen, which belongs to the class of single-stage object detectors. The choice in favor of YOLO is due to the optimal ratio between the processing speed, which is critical for real-time video analysis, and the detection accuracy for medium and large objects, such as cars in a parking lot. This project uses a pre-trained YOLO model (the specific version, for example, YOLOv11, depends on the *.pt file specified in the configuration) through a convenient interface of the Ultralytics library, which works on the basis of the PyTorch framework.

How YOLO works:

Unlike two-stage detectors (such as Faster R-CNN), YOLO processes the image in one pass of the neural network. The main idea is as follows:

1. Image division: The input image is divided into a conditional grid of S x S cells (Grid Cells).
2. Prediction in each cell: Each grid cell is responsible for detecting objects whose centers fall into this cell. For each cell, the network predicts:

- **B** bounding boxes.
- **Confidence Score** for each box.
- **C** class probabilities, provided that there is an object in the cell.

To create a video stream and access individual frames, the capabilities of the OpenCV library (cv2) were used, which allows you to work efficiently with both video files (for example, in .mp4 format) and potentially with webcams or IP cameras. Since real-time video processing requires obtaining frames with minimal delay and should not block the main graphical interface thread, the multithreading mechanism provided by the Qt framework through the QThread class (implemented in VideoThread) was used. This allows the frame acquisition and analysis cycle to run in parallel with the GUI operation, ensuring a responsive interface.

4.2. Single-stage object detection method based on the YOLO architecture.

To find vehicles (cars) in a frame from a video stream, a neural network model of the YOLO architecture (You Only Look Once) was chosen. The YOLO approach belongs to single-stage detectors, which in one pass of the convolutional neural network generates a prediction grid containing:

Bounding Boxes (BBox):

$$B = [x_1, y_1, x_2, y_2]. \quad (1)$$

Confidence scores for each detected object:

$$S = P(object) \cdot \max_{c \epsilon C} P\,(c|object), \quad (2)$$

where C is the set of possible classes of objects.
Class probabilities:

$$P(c|object), \qquad c \epsilon C. \quad (3)$$

The detection process can be represented in the form of the following schemes – YOLO model in car detection (Figure 1, 2)

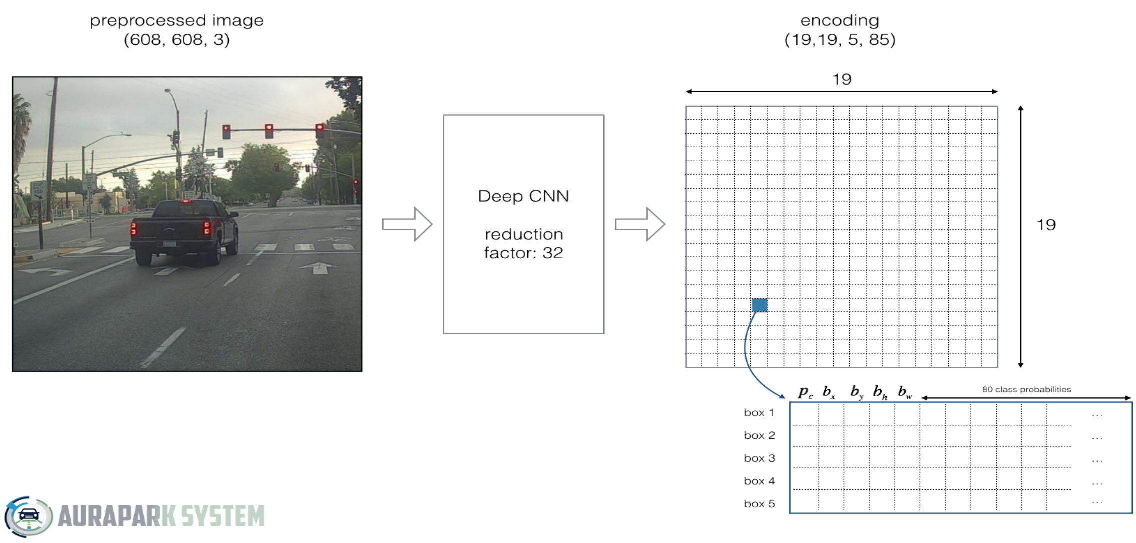


Figure 1: View of the car image processing process in Yolo

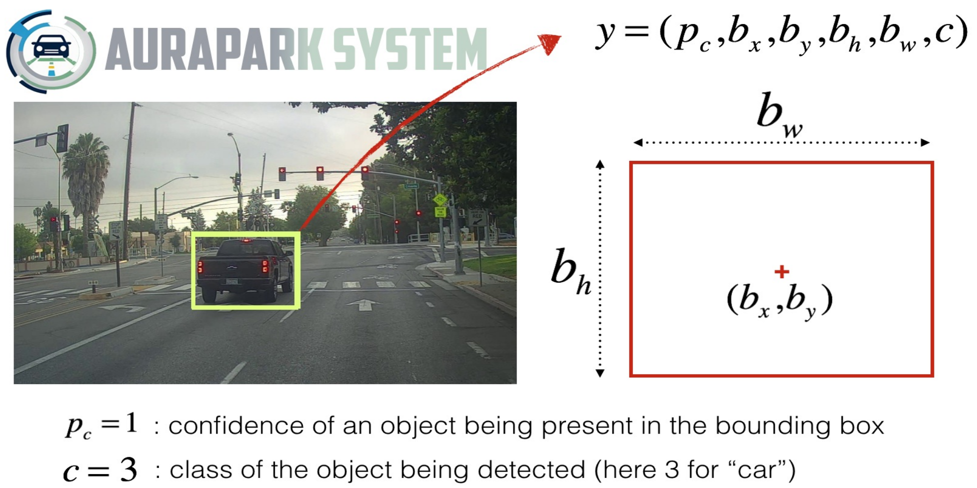
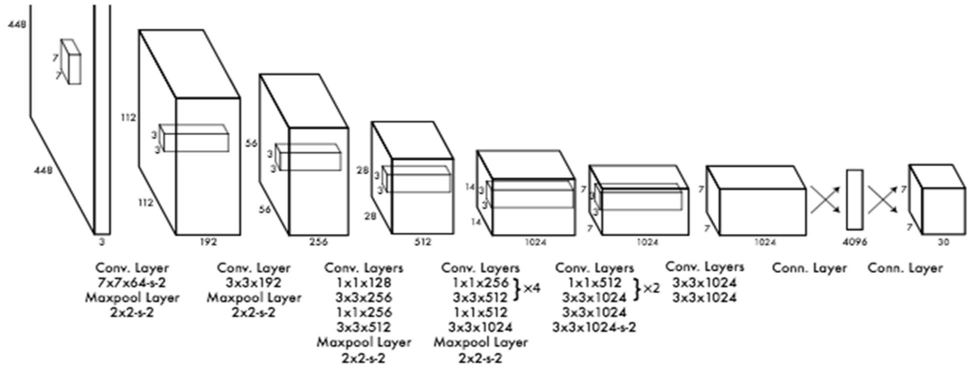


Figure 2: Final view of the processed car by tracking in Yolo

Modern implementations of YOLO (in particular, those used in the Ultralytics library) combine:

- Backbone – a deep network for feature extraction
- Neck – an aggregator of features from different levels;
- Head – a block for final prediction.

The pre-trained YOLO model (Figure 3) (.pt file) used in the prototype, trained on large datasets, is capable of detecting objects of the "car" class at high speed, which is critical for real-time systems.



AURAPARK SYSTEM

Figure 3: General view of the Yolo model

Car detection is performed by analyzing an input image of size 608×608, which is passed through a deep convolutional neural network with a reduction factor of 32×32, which leads to the output tensor of size 19×19×5×85

The interface of the Ultralytics library (based on PyTorch) allows you to conveniently load the model and obtain detection results:

$$\text{model. predict}(\ldots) \rightarrow \{B, C, S\}, \tag{4}$$

where the output contains a list of frames B, object classes C and corresponding confidence scores S.

Method of geometric analysis of parking space occupancy based on the spatial location of detected objects

To determine the occupancy status of a particular parking space, geometric analysis of the position of detected cars relative to predefined regions of interest (ROI) is used.

Each parking space is modeled by a polygon (in this implementation, a quadrilateral), the coordinates of the vertices of which are stored in the .pkl.pkl.pkl file via the Pickle module.

After receiving the frame B for the detected car, the center point of the car is calculated:

$$P_c = (c_x, c_y), \qquad c_x = \frac{x_1 + x_2}{2}, \qquad c_y = \frac{y_1 + y_2}{2}. \tag{5}$$

For each polygon R (which defines the ROI), the point P(c) is checked to be part of this polygon using OpenCV:

$$\text{cv2.pointPolygonTest}(R_{vertices}, P_c, \text{False}), \tag{6}$$

where if the result ≥ 0, then the place is considered occupied

An alternative more accurate approach is the IoU (Intersection over Union) analysis

$$IoU(B, R) = \frac{|B \cap R|}{|B \cup R|}. \tag{7}$$

If IoU > T (where T is a threshold value, e.g. 0.5), then the seat is defined as occupied. (Figure 4)

Figure 4: Parking space occupancy view

Visualization of the results (drawing ROI with colored status, seat numbers, optional BBox cars) and overlaying them on the video frame is done using the drawing functions of the OpenCV library (cv2.polylines, cv2.fillPoly, cv2.putText, cv2.rectangle, cv2.addWeighted).

So, a combination was chosen for the development: OpenCV for working with video and geometry, YOLO (via Ultralytics/PyTorch) for vehicle detection, point analysis in the ROI polygon for determining status, Pickle and YAML for saving/loading data/configuration, and PySide6 for the GUI. This creates hardware requirements (preferably GPU) to ensure real-time operation.

4.3. A method of multi-stage preprocessing of a dataset with image augmentation to improve recognition quality in conditions of visual noise and scene variability

To recognize cars, it is necessary to save the label assigned to the car and its image. The greater the number of different types of images, the better the recognition quality, but the longer the capture process.

As part of the implementation of the intelligent parking lot monitoring system, a specialized method of multi-stage dataset preprocessing was developed and applied. Its main goal is to ensure high accuracy of vehicle recognition in complex real-world conditions typical of Ukrainian urban environments: variable lighting, partial overlap of cars, the presence of shadows, rain, snow, changes in perspective and background obstacles.

The method includes the following stages:

1. Data collection and marking. A specialized dataset of images obtained from video surveillance cameras in parking lots was formed. Marking was carried out manually using appropriate tools, where each object of the "car" class was limited by a bounding box.
2. Basic normalization. All images were reduced to a single size, converted to RGB color space, and normalized pixel values to the range [0, 1].
3. Data augmentation. To increase the generalization ability of the model, an expanded set of augmentation techniques was applied.
4. Transfer learning. To increase the detection accuracy, a pre-trained YOLOv11 model was used, which underwent additional training on the generated dataset taking into account new features.

5. Model quality assessment. During additional training, accuracy and loss metrics (loss function) were monitored, accuracy–confidence graphs were constructed, as well as an error matrix. This allowed for timely detection of overtraining and correction of hyperparameters.

The use of this method allowed for significantly improving the accuracy of car detection even in complex visual conditions, which is critically important for real-time systems, where stability and reliability of the result are of key importance.

Although it is possible to use a pre-trained YOLO model for the basic functioning of the prototype, in order to achieve high accuracy and reliability in the conditions of specific Ukrainian parking lots, a specialized dataset was formed and the basic model was fine-tuned. The training process was performed using the PyTorch framework and the Ultralytics library. The pre-trained YOLOv11 model was taken as the base, as a new model. The transfer learning approach was used, where the model was trained on the dataset for 100 epochs using the NVIDIA GeForce RTX 4050 graphics accelerator(Figures 5, 6, 7).

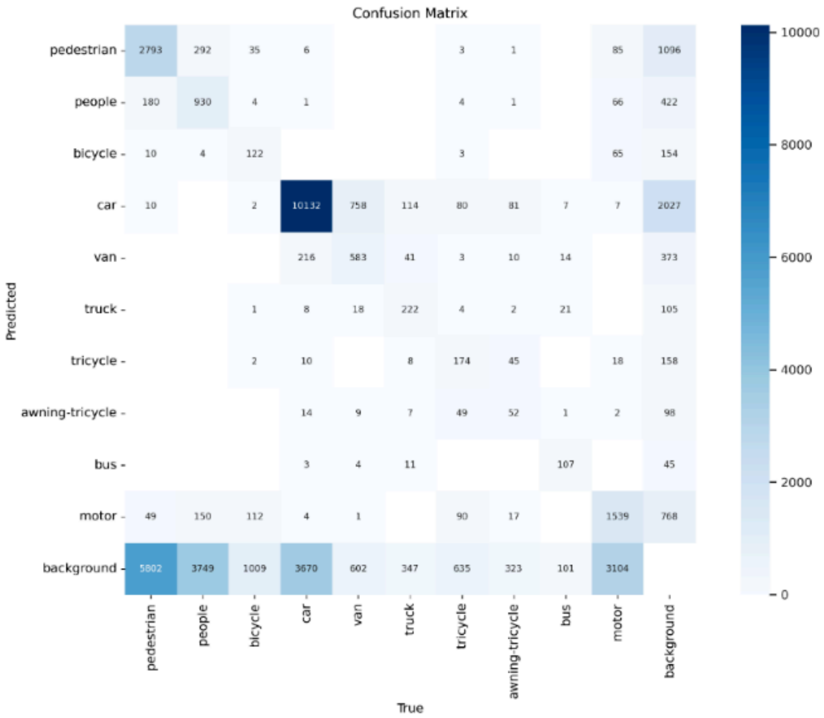


Figure 5: Error Matrix

During training, data augmentation techniques provided by Ultralytics, such as random brightness/contrast/saturation changes, horizontal reflections, scaling, shifts, and mosaic augmentation, were actively used to increase the model's resistance to input data variations.

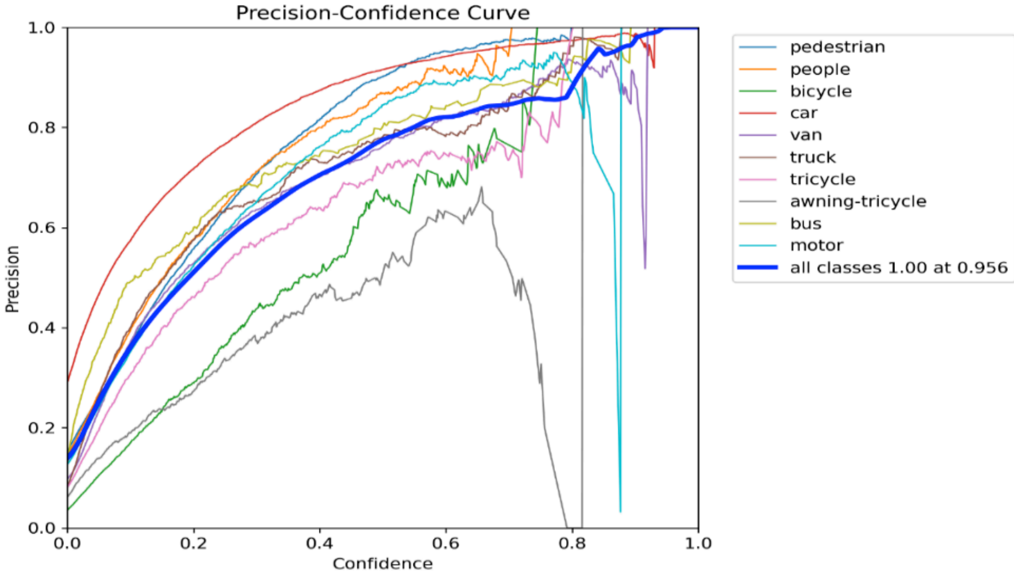


Figure 6: Accuracy-Confidence Graph

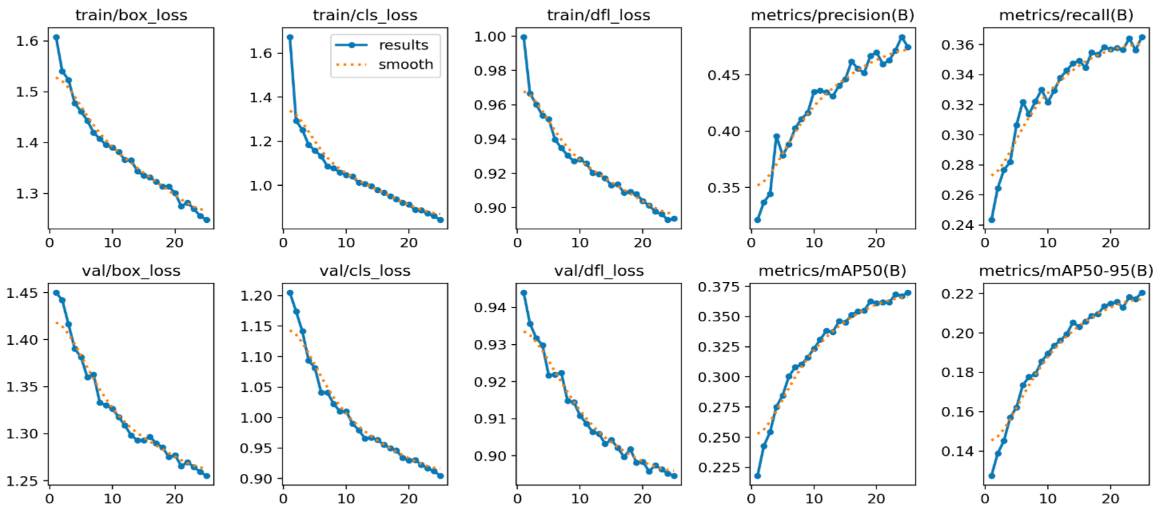


Figure 7: Graphs of successful dynamics of the model retraining process

5. Practical significance of the results obtained, ways of their improvement and further development

The results of the developed software can become a useful tool for implementation in commercial and private parking facilities, as well as serve as the basis for creating commercial parking management systems of various types (shopping centers, office centers, residential complexes). The chosen approach allows you to create a flexible and adaptive system that can function in different parking environments. Integration of tools for determining ROI and external configuration of parameters increases the practical value of the development and simplifies its implementation.

Practical significance:

1. The developed system provides a practical, visually-oriented solution for automating parking space occupancy monitoring, suitable for implementation in commercial and private parking facilities, as well as the basis for creating commercial parking management systems of various types (shopping centers, office centers, residential complexes, etc.).

2. Provides effective real-time information on parking availability, helping to reduce search time, fuel consumption and related emissions for drivers.
3. A flexible ROI definition tool allows for rapid deployment and adaptation of the system to various parking lot geometries.
4. Forms a modular basis for future extensions, including integration with reservation systems, payment platforms, license plate recognition (LPR) and advanced analytics.
5. An interactive graphical ROI editor integrated into the system has been developed, simplifying the process of configuring and adapting the system for various parking environments.

3 main areas of further development of the developed system can be distinguished:

- Increasing the accuracy and reliability of the system core: Transition from center point analysis to IoU-based occupancy determination, retraining the YOLO model on a specific target parking dataset to adapt to local conditions.
- Expanding functionality: Adding support for real IP cameras (RTSP), LPR integration, implementing parking usage analytics.
- Scaling and deployment: Moving to a client-server architecture to support multiple cameras/users, using a database to store state and configurations, considering deployment options on the server or using Edge devices.

6. Conclusions

As a result of the research, a universal prototype was created that combines fast and reliable detection with a user-friendly interface that has no direct analogues among existing open source software solutions. The scientific novelty of the work lies in the proposed integration of YOLO object detection methods with a flexible ROI management mechanism via a graphical interface and visualization of the status of parking spaces in real time within a single configurable application. The implemented system is operational, demonstrates a balance between accuracy, speed and flexibility of configuration, achieving the set goals. Architectural solutions for modularity, parallel processing (QThread) and configurability provide convenient deployment of the prototype in new environments. Prospects for further research have been identified, in particular in the direction of improving the accuracy of occupancy determination using deeper analytics, IoU intersections and multi-camera processing.

During the work, the goal was fully realized - improving the efficiency of parking lot management and providing users with up-to-date visual information about the status of spaces through a convenient graphical interface, by creating a functional prototype of an automated parking space monitoring system based on video analytics, which integrates modern deep learning approaches and computer vision tools with a flexible interface for real use.

Declaration on Generative AI

During the preparation of this work, the authors used AI program Chat GPT 4.0 for correction of text grammar. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the publication’s content.

References

[1] K Sriramdharnish, R Arun, R. Parvin Raj, H. Haseeb Batcha, A. Sanjay, Vision Park — Next Gen Computer Vision for Efficient Parking Space Monitoring, in: Proceedings of the International Conference on Emerging Research in Computational Science (ICERCS), Coimbatore, India, 2024, pp. 1-6. doi: 10.1109/ICERCS63125.2024.10895085.

[2] B. Sujitha, A. Ponraj, S. Parabrahmachari, T. V. Hyma Lakshmi, T. Annamani, Video Based Car Parking Management and Monitoring Using Computer Vision and Machine Learning, in: Proceedings of the International Conference on Multi-Agent Systems for Collaborative Intelligence (ICMSCI), Erode, India, 2025, pp. 1204-1208. doi: 10.1109/ICMSCI 62561.2025.10893979.

[3] M. M. Bachtiar, A. R. A. Besari, A. P. Lestari, Parking Management by Means of Computer Vision, in: Proceedings of the Third International Conference on Vocational Education and Electrical Engineering (ICVEE), Surabaya, Indonesia, 2020, pp. 1-6. doi: 10.1109/ICVEE 50212.2020.9243264.

[4] L. E. Giampaoli, F. Hessel, Parking Space Occupancy Monitoring System Using Computer Vision and IoT, in: Proceedings of the IEEE 7th World Forum on Internet of Things (WF-IoT), New Orleans, LA, USA, 2021, pp. 7-12. doi: 10.1109/WF-IoT51360.2021.9595935.

[5] H. Lee, I. Chatterjee, G. Cho, Enhancing Parking Facility of Container Drayage in Seaports: A Study on Integrating Computer Vision and AI, in: Proceedings of the IEEE 6th International Conference on Knowledge Innovation and Invention (ICKII), Sapporo, Japan, 2023, pp. 384-387. doi: 10.1109/ICKII58656.2023.10332699.

[6] S. Popereshnyak, I. Yurchuk, Car Parking Data Processing Technique for Smart Parking System as Part of Smart City. Lecture Notes in Computational Intelligence and Decision Making. ISDMCI. Advances in Intelligent Systems and Computing, Springer, Cham, vol 1246, 2021. doi:10.1007/978-3-030-54215-3