Parking control system using computer vision and convolutional neural network models for object detection.

Carlos Andres Barredez Rios
IU International University of Applied Sciences
carlos-andres.barredezrios@iu-study.org

November 7, 2025

Abstract

The mismatch between the supply and demand for parking spots is a recurring problem for public street parking in places with high vehicle densities. Traditional parking meter systems fail to accurately align the paid time with the actual occupancy. To address this issue, we propose a computer vision–based management approach that estimates zone-level occupancy and enables charging based on actual usage time, without the need for intrusive hardware that could contribute to urban visual pollution. Commercial RTSP cameras provide visual data that are transmitted to edge processing devices responsible for real-time detection and tracking. This allows inference of individual space status, duration of presence, and anonymous usage records. The pilot design covers 2 to 5 zones and evaluates operational, financial, and urban impact outcomes.
Keywords— Yolo, parking, vision computing, camera, hough line.

1 Introduction

Over the years, there has been a quest to bring technology not only to factories, businesses, and medicine, but also to homes and streets. Industry 4.0 has brought with it an idea where technology lives among us and is also connected to transmit data, which ends up being processed in large computer complexes where, based on that process, necessary decisions are made and policies are implemented that can change the way we behave as a species living in large urban centers. The problems caused by excessive and unplanned urbanization and the growing economic growth of large metropolises, as well as their demographics since the Industrial Revolution in England, have been noted in recent years, with cities such as New York, Mexico City, and London, to name a few from the G20, ranking among the cities with the most traffic congestion. This is a problem that affects both the quality of life of citizens and the viability of the city itself. The existence of this problem has brought with it other problems, such as contributing to high levels of greenhouse gasses, as indicated by CO_2 ppm indicators.

During the early 2000s, the rapid acceleration of machine learning models, such as advances in the field of perceptrons and what we now know as deep learning, and in the field of embedded systems focused on robotics and public use, has made the proposals of Industry 4.0 increasingly possible, bringing us closer to the idea of introducing them into people's everyday lives. The need for computational models that allow us to process large volumes of data and subsequently take action based on these data has grown, either due to specific problems, such as the aforementioned problem of excessive vehicle concentration. The demand for parking spaces exceeds the available supply in many cases in densely populated cities. It can be inferred that the relationship between availability and waiting time for a vehicle can indirectly affect the increase in vehicle density on the streets instead of finding an available space.

The implementation of a system that allows drivers to know the availability of these spaces would reduce the number of cars circulating in search of a parking space and make this task more efficient. Integration of embedded systems that do not affect visual pollution in the streets is necessary because when looking for a solution that affects quality of life to some extent, aspects such as visual pollution must be taken into account.

Current object detection and computer vision models allow us to carry out several of the tasks necessary to implement a solution that seeks to be minimally invasive to the environment on a visual level, as it is a solution where many of the processes are carried out centrally and can use things already present on the streets themselves without the need to implement additional hardware in the field where it is to be deployed. In this work, we explore a computer vision-based solution that enables efficient and non-invasive urban parking management, using cameras already present in the environment to reduce congestion and improve the citizen experience.

2 Related Work

Various projects have been carried out that seek to address the aforementioned issues using multiple computer vision systems, sensors, and deep learning. [1] One example is the "A Distributed Markovian Parking Assist System" approach, which is based on vehicle and sensor networks, although it stands out for its distributed approach and relies on costly infrastructure that presents scalability challenges. [2] One of the most recent is the one developed in 2025 called On-Street "Parking Space Detection Using YOLO Models and Recommendations Based on KD-Tree Suitability Search," which uses Yolo to detect free parking spaces. [3] "Resource-Efficient Design and Implementation of Real-Time Parking Monitoring System with Edge Device" (2025) proposes a solution for integrating computer vision with vehicle networks to improve occupancy detection. Although its approach is more holistic, it emphasizes the need for joint integration between sensors and cameras. [4] "Parking Lot Occupancy Detection with Improved MobileNetV3" it proposes an improvement to the MobileNetV3 model for detecting parking space occupancy, where it is intended to be used in real time and with high accuracy in urban environments. The publication seeks to modify the architecture by incorporating a CBAM attention module and using separable convolutions of the blueprint type.
These studies agree on the need for a computer vision solution that can be implemented in urban areas. Although many models take different approaches, there is a clear way to implement the integration of computer vision models. It is important to note that several of these solutions seek to detect parking areas, which may be changed or affected by uncontrollable factors. Therefore, many of these models would lack the robustness to consider each of the scenarios or any type of changing scenario, such as a road closure or cut, and therefore do not adapt to the changing environment independent of them.

3 Technical Background

For the technical background, it is important to mention the main components of the solution. These are necessary for the efficient and effective implementation of the system.

3.1 CCN (Convolutional Neural Network)

CCN (Convolutional Neural Network) is a type of neural network designed for the analysis of data such as visual resources. They are useful for identifying patterns in images in order to recognize objects, classes, and categories. They are inspired by the visual cortex. Convolution applies a set of convolutional filters to the input images, and each filter activates different characteristics of the images. Unlike traditional neural networks, in CNN, in each convolutional layer, each filter has a unique set of weights and a bias per filter, shared spatially. Many object identification models, such as YOLO, are based on CNN in their structure.

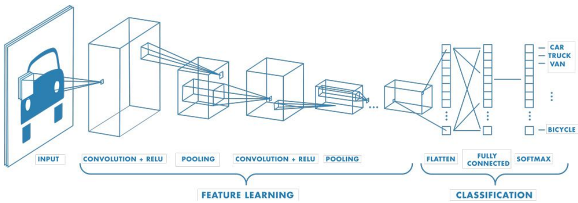

Figure 1: Architecture of a Convolutional Neural Network (CNN)

3.2 Hugh line transform

Hugh line transform is a technique for detecting parametric shapes that proceeds to find alignments in space in the same way as locating maxima in a parameter space. In the case of lines, each edge pixel proceeds to a “vote for all the lines that could pass through it,” and the aggregation of votes reveals dominant parameters.

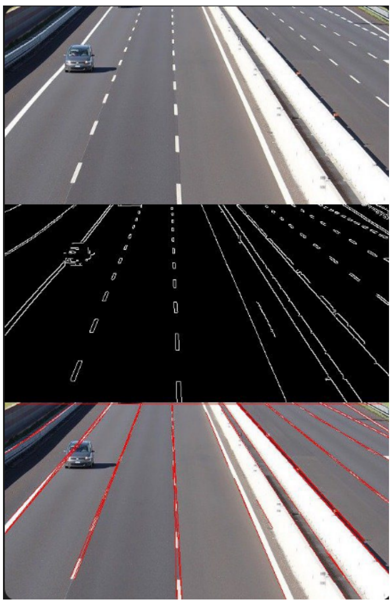

Figure 2: Visual representation of Hough Transform applied to a highway scene.

4 Method

The development of the system requires several elements, which are as follows:Methodology (algorithms, software design, or mathematical analysis).Gantt chartUML diagramData flow diagram

5 Implementation

Implementation details: language, libraries, architecture, datasets.

6 Testing

Validation strategy: metrics, experiments, results.

7 Conclusion

Conclusions and future work (limitations, improvements).

References

[1] Liu, M., Naoum-Sawaya, J., Gu, Y., Lecue, F., & Shorten, R. (2019). A distributed Markovian parking assist system. *IEEE Transactions on Intelligent Transportation Systems, 20*(6), 2230–2240. https://doi.org/10.1109/TITS.2018.2865648

[2] Kim, J., Jeong, I., Jung, J., & Cho, J. (2025). Resource-efficient design and implementation of real-time parking monitoring system with edge device. *Sensors, 25*(7), 2181. https://doi.org/10.3390/s25072181

[3] Garta, I. Y., Manongga, W. E., Huang, S.-W., & Chen, R.-C. (2025). On-street parking space detection using YOLO models and recommendations based on KD-tree suitability search. *Computers, Materials & Continua, 85*(3), 4458–4471. https://doi.org/10.32604/cmc.2025.067149

[4] Yuldashev, Y., Mukhiddinov, M., Abdusalomov, A. B., Nasimov, R., & Cho, J. (2023). Parking lot occupancy detection with improved MobileNetV3. *Sensors, 23*, 7642. https://doi.org/10.3390/s23177642

[5] Microsoft Azure. (n.d.). ¿Qué es Computer Vision? *Microsoft Azure — Cloud Computing Dictionary.* Retrieved November 6, 2025, from https://azure.microsoft.com/es-es/resources/cloud-computing-dictionary/what-is-computer-vision/

[6] MathWorks. (n.d.). ¿Qué son las redes neuronales convolucionales? *MATLAB & Simulink — Discovery.* Retrieved November 6, 2025, from https://es.mathworks.com/discovery/convolutional-neural-network.html

[7] OpenCV. (n.d.). Hough Line Transform. *OpenCV 4.x Documentation.* Retrieved November 6, 2025, from https://docs.opencv.org/4.x/d6/d10/tutorialpyhoughlines.html