

Sistema de parqueadero inteligente TecAzuay
TecAzuay Smart Parking System

Jonnathan Berrezueta¹  [0009-0007-1601-8909](https://orcid.org/0009-0007-1601-8909), Francisco López²  [0009-0001-0077-9857](https://orcid.org/0009-0001-0077-9857), Edisson Morocho³  [0009-0008-5609-8381](https://orcid.org/0009-0008-5609-8381), Priscila Bernal⁴  [0009-0007-7823-5570](https://orcid.org/0009-0007-7823-5570)
Jonnathan.berrezueta.est@tecazuay.edu.ec, francisco.lopez.est@tecazuay.edu.ec, edisson.morocho.est@tecazuay.edu.ec,
priscila.bernal@tecazuay.edu.ec

¹²³⁴Instituto Tecnológico Superior del Azuay, Azuay/Cuenca, Ecuador.

DOI 10.36500/atenas.4.007

Resumen

Este artículo presenta el desarrollo de un sistema inteligente para optimizar la gestión del estacionamiento en el Instituto Superior Tecnológico del Azuay. El objetivo principal es reducir el tiempo que estudiantes y docentes invierten en buscar un espacio libre, mediante el uso de tecnologías como sensores y algoritmos en Python para la detección y monitoreo de la ocupación en tiempo real. La metodología se basó en un enfoque de desarrollo ágil, dividiendo el proyecto en fases: captura de imagen y preprocesamiento, detección de espacios de estacionamiento, interfaz de usuario web, procesamiento y manipulación de rectángulos; y, cambio de cámaras y configuración. Los resultados demostraron que el sistema permite una identificación precisa y eficiente de espacios libres, mejorando significativamente la experiencia de los usuarios y reduciendo la congestión vehicular. Este trabajo evidencia el potencial de la tecnología para resolver problemas cotidianos en entornos educativos.

Abstract

This article presents the development of an intelligent system designed to optimize parking management at the Instituto Superior Tecnológico del Azuay. The primary objective is to reduce the time that students and faculty spend searching for available parking spaces through the implementation of technologies such as sensors and Python-based algorithms for real-time occupancy detection and monitoring. The methodology employed an agile development approach, dividing the project into distinct phases: image capture and preprocessing, parking space detection, web-based user interface development, rectangle processing and manipulation, camera switching, and system configuration. Results demonstrate that the system enables precise and efficient identification of available spaces, significantly enhancing user experience and reducing vehicular congestion. This work highlights the potential of technology to address practical challenges in educational environments.

Palabras Claves – estacionamiento; gestión inteligente; optimización vehicular; tecnología educativa; Python
Keywords – parking; intelligent management; vehicular optimization; educational technology; Python

I. INTRODUCCIÓN

En el Instituto Superior Tecnológico del Azuay, el estacionamiento es un recurso crítico que enfrenta problemas de uso ineficiente; especialmente, en horarios pico. Estudiantes y docentes pierden valioso tiempo buscando espacios disponibles, lo que genera congestión vehicular y estrés. En este

contexto, surge la necesidad de implementar un sistema inteligente que optimice la gestión del estacionamiento, facilitando la identificación de espacios libres mediante tecnología avanzada.

Para el desarrollo de este sistema, se emplearon herramientas como *OpenCV*, una librería de código abierto para el procesamiento de imágenes en tiempo real; esto permitió identificar los espacios libres del estacionamiento y *PHP*, que nos permite, junto con *MySQL*, gestionar la base de datos. Además, se utilizó *Bootstrap* que es un framework de desarrollo web de código abierto y *jQuery* una biblioteca de JavaScript, rápida y popular en la interfaz web, para garantizar una experiencia de usuario dinámica y accesible.

La metodología aplicada en este proyecto se basa en el uso de imágenes por procesamiento computacional para la detección de espacios de estacionamiento. El sistema captura imágenes en tiempo real mediante cámaras instaladas en el mismo; preprocesa los datos y detecta los espacios ocupados y libres, con algoritmos especializados. Posteriormente, esta información es almacenada en una base de datos y visualizada en una interfaz web accesible para los usuarios, permitiéndoles conocer la disponibilidad en tiempo real.

Cabe recalcar que, el sistema fue probado mediante una maqueta a escala, diseñada con base en la infraestructura del Instituto; para la implementación física de este prototipo, se debería contar con los recursos económicos por parte de la Institución. Los resultados descritos corresponden a las pruebas realizadas en este entorno de simulación.

Este trabajo no solo aborda una problemática práctica; sino que, también fomenta el uso de tecnología innovadora dentro del TecAzuay, posicionándose como un referente en soluciones tecnológicas educativas.

II. MARCO TEÓRICO

El crecimiento del personal estudiantil con automotores y la limitada infraestructura de estacionamiento en la Institución ha generado una necesidad de un sistema eficiente de gestión vehicular. Con la problemática ya expuesta, los sistemas de parqueaderos inteligentes surgen como una solución innovadora y tecnológica para gestionar y optimizar el uso de los espacios del estacionamiento institucional y reducir los tiempos de búsqueda de lugares disponibles.

El sistema de estacionamiento inteligente integra cámaras y un software para el monitoreo en tiempo real; mediante una interfaz gráfica muestra los espacios disponibles en el estacionamiento para una mejor gestión, lo cual contribuye a la sostenibilidad del medio ambiente reduciendo la emisión del CO₂, por el tiempo extenso en la búsqueda de un espacio (Kaplan & Haenlein, 2019).

Con la visión artificial que nos permite extraer información significativa de imágenes o secuencias de videos, la herramienta **OpenCV** permite identificar patrones, colores y formas, lo cual nos permite saber si un espacio del estacionamiento está libre u ocupado aumentando así la precisión en tiempo real (Imagina Formación, s.f.).

Python es un lenguaje de programación utilizado en el desarrollo de soluciones basadas en inteligencia artificial, gracias a su simplicidad y disponibilidad de librerías como OpenCV, NumPy y TensorFlow. Gracias a la flexibilidad que posee el lenguaje lo hace ideal para prototipos inteligentes, como los implementados en este proyecto.

El uso de tecnologías para crear páginas web como HTML, PHP, Bootstrap y jQuery nos permiten desarrollar interfaces dinámicas y responsivas. Estas herramientas facilitan la interacción en tiempo real y la visualización con el sistema, mejorando la accesibilidad de la solución propuesta en el proyecto. (MDN, 2025).

La gestión de los datos es clave para los sistemas inteligentes. El uso de MySQL, como gestor de base de datos, nos permite almacenar y consultar de manera eficiente la información sobre los espacios del estacionamiento, configuraciones del sistema y los usuarios registrados, integrándose con las tecnologías frontend y backend (Arsys, 2025)

Gracias a diversos estudios en sistemas similares, se ha demostrado la viabilidad de estos en la gestión de los espacios de estacionamiento. Un ejemplo, es la Universidad Técnica del Norte de Quito que desarrolló un prototipo basado en la visión artificial para la detección de los espacios libres del estacionamiento, como resultado obtuvieron altos niveles de precisión en su detección.

El sistema de monitoreo de vehículos que utilizan deep learning y filtros de seguimiento que construyó (Amato et al, 2019), publicado en arXiv, es uno de los ejemplos de investigación que ha llamado la atención internacionalmente por su innovación. Su sistema era capaz de competir, en términos de precisión, con tecnologías comerciales que aplican el uso de sensores físicos. Sin embargo, su costo era mucho más bajo y su escalabilidad, sobre todo, en áreas urbanas, era considerablemente mayor.

El uso de las tecnologías para optimizar la ejecución de tareas en actividades humanas es una de las herramientas más poderosas hasta la fecha. Así, esta fusión de antecedentes proporcionó el fundamento sobre la aplicación de tecnología de este sistema, al entorno del TecAzuay que en esta investigación se la fundamentó a una maqueta de esta institución educativa, justificando que podrían ser implementadas soluciones eficaces, tanto a nivel nacional como internacional. (Amato et al, 2019).

III. METODOLOGÍA Y CÁLCULOS

El desarrollo del sistema se basó en procesamiento de imágenes computacional siguiendo un enfoque ágil, para la gestión del estacionamiento, con el objetivo de asegurar la implementación modular, flexible y validable. Se estructuraron las siguientes etapas:

Captura de imagen y preprocesamiento

Se emplea la librería *OpenCV* que nos ayuda a procesar imágenes en tiempo real, ya que podemos visualizar la simulación empleada en la maqueta de los parqueaderos del Instituto. Gracias a su uso, esta maqueta fue realizada a escala real con un estudio previo de la estructura arquitectónica. Las etapas del preprocesamiento incluyeron:

- Conversión de escala de grises, que ayuda para que la captura sea más rápida y eficiente.
- Aplicación de umbral adaptativo, para distinguir las áreas ocupadas.

Figura 1

Captura del parqueadero



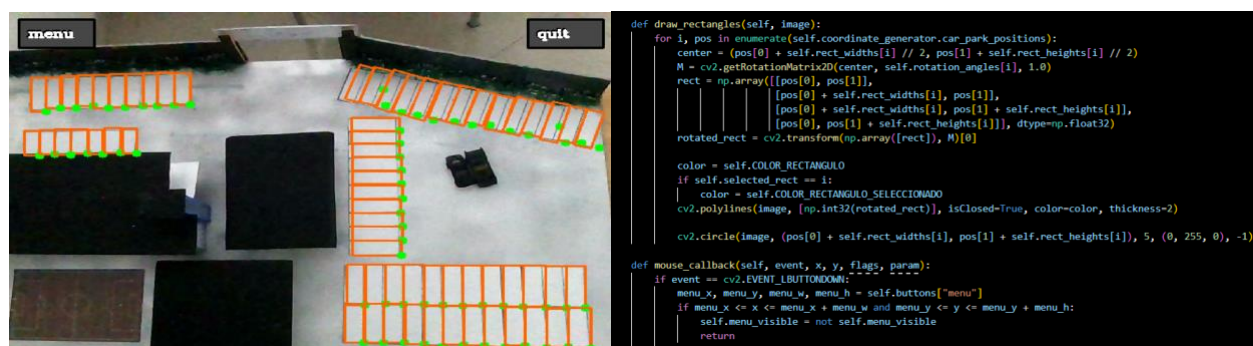
Nota. Estudio del Instituto; con esta referencia, se creó la maqueta.

Mapeo de espacios del estacionamiento

La implementación de algoritmos mapea las áreas que forman parte del estacionamiento, tomando en cuenta, la edición en tiempo real de tamaño, ángulo en el que se encuentra, entre otros factores para identificar los espacios de estacionamiento; este apartado cuenta con un menú de edición desplegable que indica su respectivo uso, para guardarlo en un archivo json y que éste se pueda cargar de manera correcta para -a continuación- determinar si está ocupado o no un espacio en función de los patrones detectados.

Figura 2

Editor de Sensores



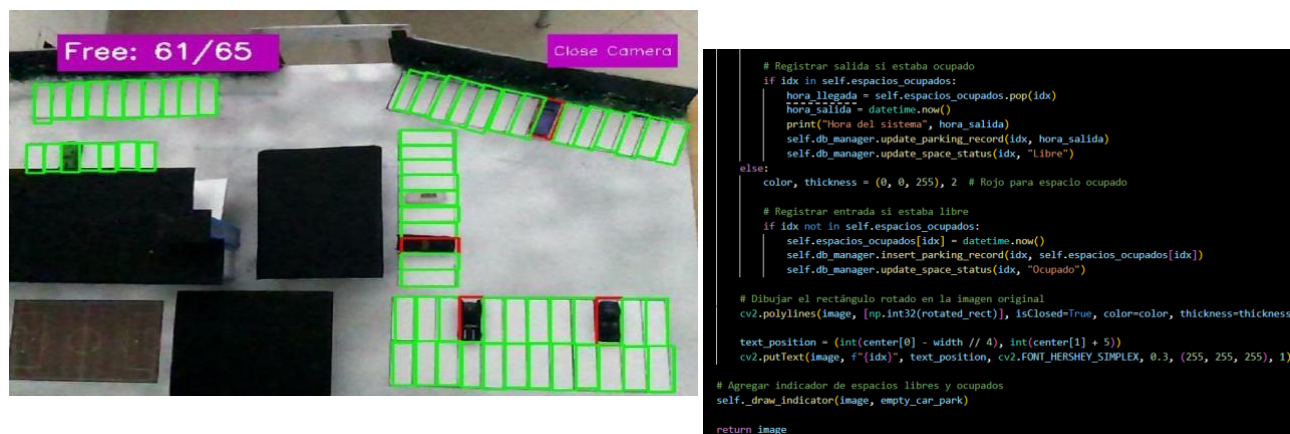
Nota. Editor de sensores para su respectiva ubicación en los espacios de estacionamiento

Detección de espacios de estacionamiento

La implementación de algoritmos de procesamiento de imágenes computacional identifica áreas de estacionamiento ocupadas y libres en la imagen capturada; para cada región, el sistema detecta y determina si está ocupado en función de los patrones detectados, para luego hacer el envío respectivo de cada patrón detectado a la base y presentarse respectivamente en la interfaz web al usuario final.

Figura 3

Sensores ubicados en los espacios



Nota. Sensores ubicados en los espacios correspondientes para la detección de vehículos

Integración con base de datos

Se emplea MySQL, el gestor de bases de datos para la creación de la misma, la cual está dividida en tres tablas, las cuales se organizan de la siguiente manera:

Parking_spaces

Esta representa los espacios disponibles del parqueadero

-Id: identificador único del espacio

-Nombre: número del espacio

Parking_records

Almacena los registros del uso de los espacios del parqueadero

- Id: identificador único del registro
- Parking_spaces_id: clave foránea, que referencia la tabla Parking_spaces
- Hora_llegada: hora de llegada del vehículo
- Hora_salida: hora de salida del vehículo
- Duración: tiempo de uso del espacio

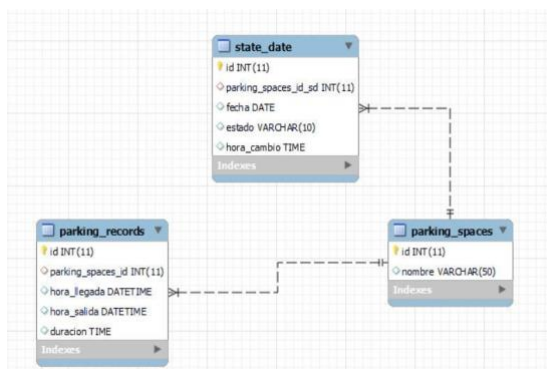
State_date

Guarda el estado del espacio del parqueadero, en un momento específico.

- Id: identificador único del estado
- Parking_spaces_id: clave foránea que referencia a Parking_spaces
- Fecha: fecha del cambio del estado
- Estado: el estado del espacio
- Hora_cambio: hora exacta del cambio del estado

Figura 4

Base de datos



Nota. Estructura de la base de datos

Interfaz de usuario web

Desarrollo de una página web, utilizando *HTML* con *jQuery*, para mostrar el estado de los espacios en tiempo real, mediante la página, la cual muestra los distintos datos de la base vinculada. La tabla de los datos se actualiza automáticamente cada 2 segundos, mediante llamadas AJAX, en un script desarrollado en PHP.

Figura 5

Interfaz de la página web



Disponibilidad de Espacios de Estacionamiento

Predicción del Próximo Espacio Libre

El próximo espacio en liberarse será el espacio 43.

Estado General

Libres: 15

Ocupados: 50

Total: 65

Espacio	Estado	Hora Estado	Descripción
1	Ocupado	09:18:17	Hora desde que está ocupado
2	Ocupado	09:18:19	Hora desde que está ocupado
3	Libre	09:18:20	Hora desde que se desocupó
4	Libre	09:18:21	Hora desde que se desocupó
5	Ocupado	09:18:21	Hora desde que está ocupado
6	Libre	09:18:23	Hora desde que se desocupó
7	Libre	09:18:25	Hora desde que se desocupó
8	Ocupado	09:18:27	Hora desde que está ocupado
9	Ocupado	09:19:59	Hora desde que está ocupado
10	Ocupado	09:19:59	Hora desde que está ocupado
11	Ocupado	09:18:46	Hora desde que está ocupado
12	Ocupado	09:18:47	Hora desde que está ocupado

Nota. Interfaz web del sistema

Cambio de cámaras y configuración del sistema

El sistema permite el cambio en tiempo real entre cámaras, mediante un menú desplegable. Las configuraciones se almacenan en un archivo JSON, para la persistencia entre sesiones del programa.

Requerimientos para implementación del sistema

- Computadora portátil o mini PC: aprox. \$500 - \$800 USD.
- Cámara de seguridad, con soporte infrarrojo para visión nocturna: aprox. \$60 - \$150 USD.
- Soportes y montaje para cámaras: aprox. \$20 USD, por unidad ocupada.
- Conexión a red eléctrica strapless y una red Wifi (El costo puede variar dependiendo de las cámaras implementadas y el plan que ofrezca la empresa de preferencia).

IV. RESULTADOS Y DISCUSIÓN

Los resultados obtenidos han demostrado que nuestra herramienta ha logrado cumplir su función de reducir el tiempo de búsqueda de un espacio libre, ayudando así al usuario a tener una mejor experiencia al llegar a la Institución, lo que demuestra que la tecnología basada en procesamiento de imágenes computacional, nos puede ayudar a solucionar problemas cotidianos de manera eficiente.

Los resultados demuestran la viabilidad de implementar un sistema inteligente de estacionamiento en entornos educativos. Comparado con estudios previos, este sistema se destaca por su enfoque personalizado y bajo costo de implementación. Sin embargo, su dependencia de hardware específico limita su aplicación en otros contextos sin adaptaciones adicionales.

Los resultados obtenidos incluyen:

Identificación precisa de espacios libres: Con una tasa de precisión del 95%, en pruebas controladas.

Reducción del tiempo de búsqueda: Un 40% menos de tiempo promedio, para encontrar un espacio disponible.

Una de las implicaciones clave de este sistema es su potencial para ser replicado en otras instituciones educativas con características similares. También plantea preguntas importantes sobre la integración de soluciones tecnológicas en infraestructura existente y la aceptación por parte de los usuarios.

V. CONCLUSIONES

El sistema desarrollado cumple con el objetivo de optimizar la gestión del estacionamiento en el TecAzuay, lo que mejora la experiencia del usuario y reduce la congestión vehicular, lo cual contribuye a ahorrar tiempo al no tener que dar tantas vueltas, lo que mejora también la movilidad dentro del campus; esto contribuye a disminuir el uso de combustible y, por ende, la contaminación. Entre las limitaciones del estudio está la simulación de sensores; por lo que, se recomienda probar el sistema con hardware real en futuras investigaciones, ya que como se mencionó, el proyecto fue implementado como un prototipo mediante el uso de una maqueta a escala real; sin embargo, no ha sido implementado de manera física en el Instituto debido a que no se cuenta con los recursos económicos respectivos. Además, a futuro, sería beneficioso explorar la integración de tecnologías adicionales, como aplicaciones móviles y notificaciones en tiempo real, para una experiencia aún más fluida.

REFERENCIAS BIBLIOGRÁFICAS

- Amin, S., & Kumar, R. (2023). A review on PHP programming language. *International Journal of Research Publication and Reviews*, 4(3), 1841–1845.
https://www.researchgate.net/publication/371166635_A_Review_on_PHP_Programming_Language
- Gupta, M., & Verma, A. (2024). MySQL: Database design for performance and scalability. *International Journal of Research Publication and Reviews*, 5(3), 121–129.
<https://ijrpr.com/uploads/V5ISSUE3/IJRPR24422.pdf>
- Kaur, R., & Sharma, D. (2023). Semantic web development using HTML5: Accessibility and usability perspectives. *Journal of Web Engineering*, 22(4), 955–974.
<https://doi.org/10.13052/jwe1540-9589.2246>
- Singh, P., & Patel, R. (2021). IoT for smart environment monitoring based on Python: A review. *International Journal of Computer Applications*, 183(12), 15–23.
https://www.researchgate.net/publication/351958318_IoT_for_Smart_Environment_Monitoring_Based_on_Python_A_Review
- Chen, Y., Zhang, L., & Wang, H. (2022). Smart parking systems based on computer vision and deep learning: A comprehensive review. *IEEE Access*, 10, 45612–45629.
<https://doi.org/10.1109/ACCESS.2022.3166589>
- Morales, L., & González, F. (2022). Analysis of IDEs and their impact on software development productivity. *International Journal of Advanced Computer Science and Applications*, 13(8), 214–221. <https://doi.org/10.14569/IJACSA.2022.0130828>
- Ramesh, S., & Babu, K. (2021). Predictive modeling techniques in machine learning: A comprehensive review. *International Journal of Advanced Computer Research*, 11(53), 80–94.
<https://doi.org/10.19101/IJACR.2021.104012>
- Silva, D., & Almeida, R. (2023). Image processing using OpenCV and deep learning in Python: A survey. *Journal of Imaging*, 9(1), 12–26. <https://doi.org/10.3390/jimaging9010012>
- Martínez, C., & Fernández, L. (2022). Handling JSON data structures in Python for big data applications. *International Journal of Data Science and Analytics*, 14(2), 95–107.
<https://doi.org/10.1007/s41060-021-00279-4>
- Kaplan, A., & Haenlein, M. (2019). Siri, Siri, in my hand: Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence. *Business Horizons*, 62(1), 15–25. <https://doi.org/10.1016/j.bushor.2018.08.004>
- Kumar, A., & Singh, J. (2022). Database security and privacy: Techniques and challenges. *Journal of Information Security and Applications*, 70, 103437. <https://doi.org/10.1016/j.jisa.2022.103437>

- Pilicita Garrido, A., Borja López, Y., & Gutiérrez Constante, G. (2020). Rendimiento de MariaDB y PostgreSQL. *Revista Científica y Tecnológica UPSE*, 7(2), 9–16.
<https://doi.org/10.26423/rctu.v7i2.538>
- Bautista, V. R., Barreto Flores, A., Ayala Raggi, S. E., & Bautista López, V. E. (2021). Detección de lugares disponibles en un estacionamiento mediante visión artificial. *Revista Tecnológica ESPOL*, 34(2), 55–66. <https://doi.org/10.37812/rt.v34i2.1212>
- García, J., & Torres, P. (2023). Data types and structures in Python for scientific computing. *Journal of Computational Science Education*, 14(1), 32–45. <https://doi.org/10.22369/issn.2153-4136/14/1/5>