License plate recognition system for data collection on the Zacatenco campus.

M. A. Acevedo-Mosqueda [1]; F. Felipe-Duran. D[2]; A.J. Argüelles-Crz [3]
[1,2]Instituto Politécnico Nacional, ESIME Zacatenco, México, *macevedo@ipn.mx, ffelipe@ipn.m,*
[3]Instituto Politécnico Nacional, CIC, México, aarguelles@ipn.mx

Abstract– In Mexico, vehicular traffic represents a significant challenge. The Mexican government highlights how the lack of accurate data hinders the implementation of efficient solutions for its management. In contexts such as those of educational institutions, having reliable information is essential for making informed decisions that improve safety and traffic flow. This project proposes the development of a vehicle license plate recognition system to obtain data that would allow authorities to make informed decisions to improve safety and traffic management at the Zacatenco Campus of the National Polytechnic Institute. The solution integrates advanced computer vision technology using the YOLOv8 model for license plate detection and EasyOCR for text extraction, storing the information in real time in an AWS DynamoDB database. The collected data is visualized through a web interface implemented in AWS EC2, designed to facilitate access to statistics and reports by campus authorities.

Keywords-- License plates, YOLOv8, Internet of Things, cars, recognition, Smart cities

Sistema de reconocimiento de placas para la obtención de datos en el campus Zacatenco

M. A. Acevedo-Mosqueda [1]; F. Felipe-Duran. D[2]; A.J. Argüelles-Crz [3]
[1,2]Instituto Politécnico Nacional, ESIME Zacatenco, México, *macevedo@ipn.mx, ffelipe@ipn.m,*
[3]Instituto Politécnico Nacional, CIC, México, aarguelles@ipn.mx

Resumen– ***En México, el tráfico vehicular representa un desafío significativo, el gobierno de México destaca cómo la falta de datos precisos dificulta la implementación de soluciones eficientes para su gestión. En contextos como el de instituciones educativas, contar con información confiable resulta esencial para tomar decisiones informadas que mejoren la seguridad y el flujo vehicular. Este proyecto propone el desarrollo de un sistema de reconocimiento de matrículas vehiculares para la obtención de datos que permitirían a las autoridades tomar decisiones informadas para la mejora de la seguridad y la gestión del tráfico en el Campus Zacatenco del Instituto Politécnico Nacional. La solución integra tecnología avanzada de visión por computadora utilizando el modelo YOLOv8 para la detección de matrículas y EasyOCR para la extracción de texto, almacenando la información en tiempo real en una base de datos AWS DynamoDB. Los datos recolectados se visualizan mediante una interfaz web implementada en AWS EC2, diseñada para facilitar el acceso a estadísticas e informes por parte de las autoridades del campus.***

Palabras clave—Placas, YOLOv5, Internet de las cosas, automóviles, reconocimiento, Ciudades Inteligentes.

I. Introducción

Las ciudades del mundo abarcan el 1.75 de la superficie de la tierra y en ellas vive el 57% de la población de la tierra, esta situación implica retos y tareas cotidianas para los habitantes de las ciudades y los gobiernos a nivel municipal, estatal o a nivel país para mejorar la calidad de vida en ellas.

Los retos son varios y existen varias organizaciones a nivel mundial que han creado grupos de trabajo para abordarlos, uno de ellos es el grupo de Ciudades Inteligentes y Sostenibles de la Unión Internacional de Telecomunicaciones. Ellos han establecido los siguientes retos [1]:

- **Infraestructuras y servicios limitados**.
- **Demandas de conectividad**.
- **Estrés ambiental**
- **Movilidad y congestión**
- **Desigualdades digitales**
- **Seguridad e inclusividad**

Otras organizaciones como la IEEE, la ISO también tienen sus grupos de trabajo y han establecido tareas y retos desde su perspectiva, es importante comentar que coinciden en varios aspectos y divergen en pocos [2] [3].

Las herramientas tecnológicas para lograr ciudades sostenibles e inteligentes son el Internet de las Cosas (IoT por sus siglas en inglés), la Inteligencia Artificial y las Tecnologías de la Información y Comunicaciones (TIC´), de hecho, estas tecnologías se entremezclan y son necesarias todas para lograr las mejoras mencionadas.

Una de las tareas en que coinciden todos los organismos que abordan el tema de las Ciudades Inteligentes, es la Movilidad Urbana en las ciudades modernas. Este en un problema que se presenta en todas y su solución es compleja, va desde construir medios masivos de transporte como metros, teleféricos, trenes suburbanos, avenidas, optimizar el flujo vehicular y priorizar los desplazamientos en medios no contaminantes como bicicletas y en la medida de lo posible que la gente se desplace a pie por las ciudades [4].



Fig. 1 Tráfico en la CDMX

Los beneficios de mejorar la Movilidad Urbana tienen su principal beneficio en disminuir los tiempos muertos, los tiempos utilizados en los desplazamientos por la Ciudad. Entre menos sea el tiempo utilizado para esta actividad, se tendrá más tiempo para estudiar, trabajar, descansar o bien para la

diversión, de ahí nace la importancia del mejoramiento de la Movilidad Urbana.

La Movilidad Urbana tiene a su vez varios temas para tratar, desde el consumo óptimo de energía para mover los vehículos, pasando por la seguridad de los vehículos, desplazamientos óptimos por la Ciudad hasta el tema de los estacionamientos. Las grandes ciudades de hoy requieren espacios donde estacionar vehículos, espacios que deben ser seguros, en lugares estratégicos para hacer eficiente la movilidad urbana. En este trabajo se aborda el problema de manejar la información de vehículos que ingresan a los estacionamientos de un Campus Académico y obtener datos relevantes de su estancia en el Campus, para ello se utilizaran tecnología IoT, Inteligencia Artificial y TIC´s.

La Inteligencia Artificial se usa mejorar la movilidad urbana, por ejemplo, para predecir flujos de tráfico de acuerdo a los patrones observados en datos estadísticos, también por medio de vehículos autónomos o bien detectado en videos accidentes o embotellamientos II. TRABAJOS RELACIONADOS

En la India, en el año 2021 se desarrolló el trabajo "Optical Character Recognition (OCR) basedVehicle's License Plate Recognition System Using Python and OpenCV", El propósito de este trabajo es desarrollar y evaluar un sistema de Reconocimiento Automático de Placas de Vehículos (AVLPR) utilizando Python y la biblioteca OpenCV. Este sistema implementa técnicas de visión por computadora para detectar y reconocer placas de vehículos en imágenes digitales. Se emplea un algo- ritmo de reconocimiento óptico de caracteres (OCR) para extraer y procesar los caracteres de las placas de los vehículos. La arquitectura del sistema está diseñada para capturar y procesar imágenes de alta resolución, con el objetivo de mejorar la precisión del reconocimiento. Las imágenes de las placas se capturan a una distancia de 2-3 pies, se convierten a escala de grises y luego se binarizan para facilitar el procesamiento posterior. El sistema ha sido evaluado en diferentes condiciones y ha demostrado ser eficaz en la detección y reconocimiento de placas, alcanzando una precisión del 93.34% en la localización de la placa, un 85.71% en la segmentación de caracteres y un 91.67% en el reconocimiento de caracteres [5].

Otro proyecto implementó un sistema de Reconocimiento Automático de Placas de Vehículos (ALPR) utilizando Raspberry Pi. Se utilizaron técnicas de visión por computadora para detectar y reconocer placas de vehículos en imágenes digitales. Para obtener los caracteres de las placas se usó el software Tesseract OCR.

El sistema tiene varios módulos funcionales para trabajar: adquisición de imágenes, preprocesamiento, segmentación de bordes (Se implementó software incluyendo Canny Edge y Otsu Thresholding) y reconocimiento de caracteres. Las imágenes se capturan una distancia ya establecida, así se optimiza el porcentaje de reconocimiento. El sistema ha sido evaluado bajo diferentes condiciones de iluminación y ha mostrado una efectividad notable, alcanzando una precisión del 72% en el reconocimiento de placas, destacándose el método Canny Edge por su capacidad para detectar el 100% de los bordes de las placas frente al 70% detectado por Otsu Thresholding.

Este proyecto proporciona una solución práctica para la detección automática de placas de vehículos, mejorando la eficiencia y precisión del reconocimiento en aplicaciones de monitoreo vehicular y control de acceso [6].

Un sistema de nueva generación de reconocimiento de placas con Deep Learning en la nube para ciudades inteligentes Este estudio presenta un sistema de Reconocimiento de Placas de Matrícula (LPRS) basado en redes neuronales convolucionales profundas implementadas en plataformas en la nube. El objetivo principal del sistema es mejorar la eficiencia y precisión del reconocimiento de placas en entornos desafiantes como el tráfico congestionado y condiciones climáticas adversas.

El método propuesto utiliza técnicas avanzadas de extracción de características mediante redes CNN para la localización, detección y segmentación de caracteres en las placas de matrícula. Esto permite al sistema reconocer placas correctamente incluso bajo condiciones subóptimas, como placas con información adicional o distorsionadas.

Los resultados experimentales indican que el sistema LPRS alcanza una alta precisión en la detección y reconocimiento de placas en comparación con métodos tradicionales como se muestra en la figura 2.2. Además, la implementación en plataformas de nube con servidores bare-metal equipados con GPU NVIDIA permite acelerar significativamente el proceso de entrenamiento del modelo, mejorando tanto la velocidad como la precisión del sistema.

En resumen, el sistema propuesto demuestra ser efectivo y eficiente para aplicaciones de reconocimiento de placas de matrícula en entornos urbanos y de tráfico intenso, superando las limitaciones de los métodos convencionales y ofreciendo resultados prometedores para su implementación en sistemas de ciudades inteligentes y seguridad vial [7]

También existen sistemas comerciales como "LPR Vehicle License Plate Recognition System", que tiene un eficiente y sólido Sistema de control de acceso y es fundamental parte de la seguridad y protección de estacionamientos. El Sistema LPR de reconocimiento de

placas proporciona una solución de fácil utilización, es adaptable a diferentes estacionamientos físicos y proporciona herramientas para controlar estacionamientos. El sistema se ha instalado en varios lugares, por ejemplo en Barranquilla Colombia con resultados muy buenos en centros comerciales de aquella ciudad.

III. Marco teórico

Visión por Computadora es un área de estudio de la Inteligencia Artificial, .su objeto de estudio son imágenes y vídeos de computadora para simular o imitar la visión de las personas, siendo la principal tarea el reconocimiento de patrones presentes en esos medios digitales mencionados. La Visión por Computadora se puede separar en dos grandes etapas de trabajo, de Bajo Nivel y de Alta Nivel [8].

La etapa de Bajo Nivel se encarga del preprocesamiento de las imágenes donde se debe reducir el ruido o las imperfecciones que tiene la imagen digital comparada con la imagen real, por lo general cambiar la imagen a tono de grises, segmentar la imagen para simplificar la detección de objetos entre otros posibles tratamientos.

El procesamiento de alto nivel se encarga de las tareas más complejas en las que se requieren interpretar y comprender el contenido de la imagen. Se utilizan representaciones abstractas de los datos para identificar, clasificar y extraer la información más relevante, además de emplear técnicas avanzadas como aprendizaje profundo y algoritmos de detección de objetos que finalmente realizan en las imágenes un proceso [9] [10].

Las tecnologías OCR (Optical Character Recognition) están enfocadas en reconocer los textos escritos desde una imagen y transformarlos a su representación digital como caracteres. Su principal aplicación es con escáneres que digitalizan documentos escritos. OCR utiliza una tecnología llamada ICR (Intelligent Character Recognition) que proporciona herramientas para reconocer escritura en varios idiomas e incluso reconoce textos manuscritos. ICR tiene su base en Redes neuronales profundas como la CNN y LSTM que son del tipo de red convolucional [11].

Modelos de aprendizaje profundo. YOLO (You Only Lokk Once) también es una red neuronal especializada en la detección de objetos en tiempo real. La técnica que utiliza YOLO está basada en una Red Neuronal Convolucional que analiza las imágenes en una sola pasada y realiza una detección unificada y trabaja con probabilidades para predecir coordenadas de cuadros delimitadores (bounding boxes), esos cuadros delimitadores son los objetos que se han propuesto para ser detectados. Esto ha permitido a YOLO identificar múltiples objetos en una imagen con resultados excelentes.

La Figura 2 muestra de forma gráfica la aplicación de YOLO para detectar automóviles en una imagen [12].

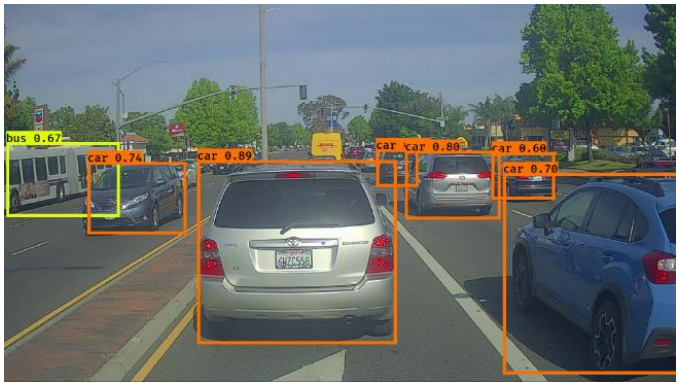


Fig. 2 YOLO reconociendo el objeto automóvil

Internet de las Cosas (IoT) es una tecnología que ha acelerado su desarrollo en los últimos diez años, que conecta cosas o dispositivos al exterior del medio donde se encuentra el objeto vía Internet. El dispositivo tiene asociados sensores, cámaras o bien pequeñas computadoras que se encargan de operar in situ la operación del objeto o cosa. Ejemplos de cosas pueden ser el aire acondicionado de una empresa o casa, un semáforo de una avenida, la bomba de un pozo de agua entre otros.

Uno de los aspectos importantes del IoT es la información que se puede recabar y almacenar en un servidor al otro lado del sistema completo, esa información puede ser almacenada para su posterior análisis, pero en otras aplicaciones se suele enviar la información del nodo IoT a dispositivos móviles como un teléfono celular desde el cual se puede monitorear el estatus del nodo y del mismo celular tomar decisiones sobre el control del nodo IoT [13].

La Raspberry Pi 5 es una microcomputadora de propósito general diseñada para aplicaciones educativas, industriales y de investigación, por su tamaño reducido y su capacidad de cómputo es muy utilizada en sistemas IoT. Cuenta son CPU´s ARM multicore, interfaces para cámaras y sensores, y conectividad inalámbrica integrada, la Raspberry Pi 5 es, hoy en día, utilizada en aplicaciones de monitoreo, análisis de imágenes y control remoto. También puede trabajar con diferentes herramientas de software como Python sus bibliotecas de procesamiento de imágenes como OpenCV y varias bibliotecas de funciones más [14].

El Módulo SIM808 es un dispositivo integrado que combina las funcionalidades de comunicación GSM/GPRS con un sistema de posicionamiento global (GPS). Este módulo es ampliamente utilizado en aplicaciones de IoT que requieren

conectividad móvil y capacidades de localización en tiempo real. La integración de estas funcionalidades permite que el módulo SIM808 envíe datos a través de redes móviles y registre coordenadas geográficas precisas, lo que lo hace especialmente adecuado para escenarios donde la conectividad Wi-Fi no está disponible o es inestable. En el contexto de sistemas IoT, el SIM808 permite la transmisión de datos relevantes, como mediciones, alertas o ubicaciones, a servidores remotos o plataformas de análisis en la nube. Este módulo ha sido implementado en soluciones que incluyen monitoreo vehicular, rastreo de activos y sistemas de seguridad, destacándose por su bajo consumo de energía y su capacidad para operar en entornos remotos. La versatilidad del SIM808 y su compatibilidad con microcontroladores como Arduino y plataformas como Raspberry Pi, lo convierten en una herramienta esencial en el desarrollo de sistemas IoT confiables y eficientes [15].

Almacenamiento y Procesamiento de Datos. Los Sistemas IoT utilizan diferentes medios para almacenas la información que recaban de cada nodo que se tiene instalado, si el sistema IoT estuviera instalado en una red de semáforos, la información que se genera en cada uno de ellos puede ser de miles de datos por día en cada uno de ellos. Por eso es importante desarrollar almacenamientos donde se puedan procesar todos esos datos [16] [17].

Una base de datos es un conjunto de datos que se almacena de forma digital en un servidor o algún servicio en la nube. Los datos almacenar diferentes tipos de información, como imágenes, vídeos y archivos. Las bases de datos cuentan con un sistema de administración de bases de datos (DBMS) que se encarga del control de la información de su integridad y de su consistencia.

Hoy en día existen servicios de almacenamiento en los que la información se guarda en centros de datos que no pertenecen a la organización dueña de la información, se le conoce de forma genérica como la nube. Los usuarios de la información pueden acceder a ella vía Internet y dejan el trabajo de almacenamiento y procesamiento de información a ese tercero. Un servicio de Cómputo en la nube es AWS Amazon Web Services, infraestructura en la nube que facilita el almacenamiento, manejo de bases de datos, configuración de redes, análisis y seguridad a través de la colección de servicios de cómputo en la nube publica ofrecidas por la empresa Amazon [18]

IV. Metodología y Desarrollo del Prototipo

La metodología usada para construir el prototipo constó de cuatro grandes etapas o módulos; en el primero se estudiaron las alternativas para la construcción del mismo, primero el nodo IoT, computadora a utilizar, cámara, GPS y medio de comunicación, después se determinaron las herramientas de la IA para reconocer vehículos y placas a continuación, a continuación se determinó el servidor y base de datos y por último la interfaz de usuario para obtener la información del prototipo. La Figura 3 muestra los bloques o etapas de la metodología de construcción del prototipo.

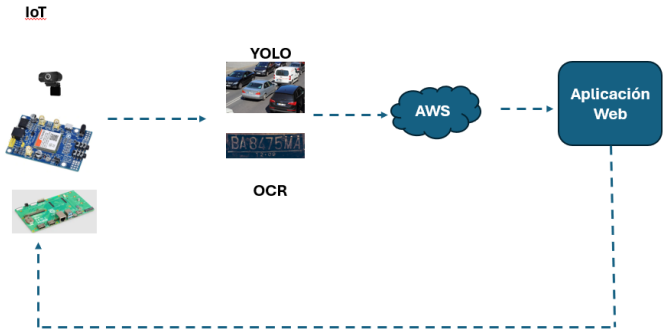


Fig. 3 Metodología del Prototipo

La primera etapa es la adecuación del nodo IoT, este consta de una computadora Raspberry Pi 5 una minicámara Web, el modulo SIM808 con el GPS y la tarjeta SIM para mensajes SMS. La Raspberry Pi 5 es de suma importancia porque en ella se realiza el reconocimiento de las placas, para ello se deben realizar las siguientes tareas:

Detección del área de interés, esto es la región donde se encuentra la placa, esto se hace con YOLOv8, la imagen se captura, se redimensiona a 640 X 640 pixeles, se aplican las capas convolucionales y se optimizan, la salida son las coordenadas de los cuadros delimitadores de la imagen analizada, una etiqueta de clase y un nivel de confianza.

En el Preprocesamiento consiste en el recorte de la imagen, se redimensiona a un tamaño fijo de 200 X 100 pixeles, se aplica un filtro Gaussiano para reducir el ruido de la imagen, después se convierte a escala de grises y se mejora el contraste con el algoritmo CLAHE, a continuación, se binariza la imagen con el método OZU. Para la base de datos de referencia se utilizó un universo de ciento veinte automóviles para la detección del área de las placas.

Reconocimiento óptico de caracteres con EasyOCR. La Figura 4, muestra el flujo del ´proceso OCR

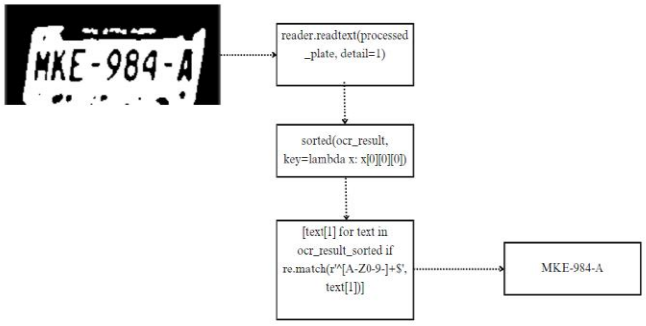


Fig. 4 Reconocimiento de texto con OCR

El proceso inicia con la función ***readtext*** de EasyOCR, con ella se obtienen las coordenadas del texto, el texto detectado y la confianza asociada a la solución; a continuación, se ordena el texto en forma horizontal para reconstruir el texto en el formato correcto, después

La segunda etapa fue la implementación de servicios en AWS, para ello se creó una base de datos con DynamoDB escalable y de alta disponibilidad, los archivos o tablas implementadas fueron para Placas, Usuarios, de Lista Negra, esta última tabla tiene importancia estratégica porque en ella se registran placas a las que se debe dar un seguimiento especial por algún motivo, esto no significa que les sea negado el acceso a los estacionamientos, es más bien una alerta, por ejemplo, en ella se registran los vehículos propiedad del IPN, a los que se debe registrar con más detalle en sus movimientos por control administrativo.

La tercera y última etapa es el desarrollo de la aplicación que servirá como interface para los usuarios del sistema, ésta página se alojó en AWS EC2, lo que permite consultar la base de datos en AWS DynamoDB, la estructura del servidor incluye los archivos y carpetas siguientes:

- **app.py**: Archivo principal que gestiona las rutas y la lógica de la página.
- **requirements.txt**: Archivo que contiene las dependencias necesarias para la ejecución de la aplicación.
- **templates/**: Carpeta que contiene los archivos HTML para la interfaz de usuario.
- **static/**: Carpeta que contiene los archivos CSS y JavaScript para la personalización de la página web.

El servidor EC2 se conecta con la base de datos de DynamoDB mediante una IAM Role que le otorga los permisos necesarios para acceder y consultar las tablas ’placas’ y ’usuarios’. De esta forma, se permite la visualización de la información de entrada y salida de vehículos en la interfaz web, la cual muestra el historial de entradas y salidas por matrícula, así como estadísticas de acceso.

IV. Pruebas y Conclusiones

Las pruebas se realizaron con diferentes videos e imágenes de vehículos tomadas a la entrada de los estacionamientos del Smart Campus Zacatenco en diferentes horarios, con diferentes velocidades.

La parte crítica del sistema es el reconocimiento de las placas, globalmente, el modelo YOLOv8 detectó correctamente el área de las matrículas hasta en un 96% de los casos (por ejemplo, en condiciones de mediodía), mientras que en escenarios de baja iluminación la precisión disminuyó hasta un 82%. En condiciones óptimas de luz (por ejemplo, mañana y mediodía), YOLOv8 alcanzó entre 96% y 97% de precisión, y EasyOCR logró un 92%-93%. Esto se ve reflejado también en los Dice Score más altos (0.95 y 0.96). Por el contrario, en noche o con distancias largas (>10 m), la precisión del OCR bajó hasta el 60%-65%, y YOLOv8 llegó a un 78%-82%. El tiempo de procesamiento promedio se mantiene entre 6.5 y 7.5 segundos, salvo en condiciones de señal baja (<50%) donde puede llegar hasta 9.0 segundos. La tabla 1 muestra un resumen de resultados.

Los flujos vehiculares son diferentes en cada etapa del día, pero en cada etapa se probó el prototipo con al menos quince automóviles.

TABLA I
RESULTADOS DE LAS PRUEBAS

Condición de prueba	Dice Score	Precisión YOLOv8	Precisión OCR	Tiempo	Comentarios
Mañana 6:00 10:59	0.95	96%	92%	6.5	Iluminación óptima, excelente detección
Mediodía 11:00 14:59	0.96	97%	95%	6.5	Excelente rendimiento, condiciones ideales
Tarde 15:00 18:59	0.88	90%	80%	6.7	Luz disminuye, OCR se ve afectado
Noche 19:00 5:59					Baja iluminación impacta YOLOv8 y OCR
Ángulos pronunciados	0.84	85%	72%	6.8	YOLOv8 mantiene precisión , OCR afectado
Distancias largas	0.72	78%	60%	7.2	Resolución limitada, afecta la detección
Velocidad baja < 20 km/h	0.92	95%	90%	6.5	Precisión alta, buena respuesta
Velocidad alta > 50 km/h	0.70	75%	55%	7.5	Movimiento borroso, OCR afectado
Buena señal de red > 75%	0.93	94%	88%	6.8	Envío de datos eficiente, tiempo esperado
Señal media 50% - 75%	0.87	89%	78%	7.5	Aumenta procesamiento, señal media
Señal baja < 50%	0.75	80%	65%	9.0	Retrasos significativos, señal débil

Si el prototipo se compara con otros desarrollados, se puede observar que los otros sistemas tienen poco éxito a velocidades superiores a los diez kilómetros por hora, de hecho alguno requieren que los vehículos casi se detenga.

La aplicación Web, la interfaz con el usuario tiene varias páginas, pero la que muestra el resultado final de todo el

sistema se puede ver en la Figura 5, se muestra de forma global los datos almacenados y las estadísticas generadas.



Fig. 5 Interfaz web mostrando los datos simulados almacenados y las estadísticas generadas.

Ya con las pruebas realizadas se concluye que el prototipo tiene un alto grado de identificación de las placas con condiciones óptimas y que la información de los usuarios de los estacionamientos es almacenada correctamente, con el prototipo se pueden obtener varios tipos de estadísticas muy importantes para los administradores del sistema, como horas de estancia, rutas de entrada / salida del campus, patrones de ocupación de los estacionamientos o bien horas en que el tráfico se incrementa.

Ente los trabajos futuros se sugiere que las autoridades del Smart Campus Zacatenco de implementarlo en algún de los estacionamientos, esto permitiría hacer adecuaciones, por ejemplo, de iluminación, para que el prototipo mejorará en el reconocimiento de las placas, si se instalan más prototipos, en las entradas / salidas del Smart Campus se podría estudiar patrones de desplazamiento, ya que es frecuente que un usuario entre por una de entrada por un lugar y se retire por otra de las salidas

AGRADECIMIENTOS

Los autores expresan en esta sección su agradecimiento a los Sistemas de Becas por Exclusividad de la Comisión Organizadora de Fomento a las Actividades Académicas, al Sistema de Estímulo al Desempeño Docente de la Secretaría Académica y al Sistema de Estímulos a la Investigación de la Secretaría de Investigación y Postgrado, ambos Sistemas soportados por el Instituto Politécnico Nacional de México. También a la Secretaría de Investigación y Postgrado porque este trabajo se deriva de las actividades del proyecto de investigación "Micro movilidad urbana en el Smart Campus Zacatenco", clave 20240757. Los autores agradecen la dedicación de la alumna Ana Laura Arenas Estrada quien desarrolló el prototipo como parte de su proyecto de titulación curricular en el Instituto Politécnico Nacional.

REFERENCIAS

[1] International Telecommunications Union, "Ciudades Inteligentes y Sostenibles", https://www.itu.int/en/mediacentre/backgrounders/Pages /smart- sustainable-cities.aspx#/es, consultado 30 /06 / 2025
[2] Institute of Electrical and Electronics Enginnerings, "What is a IEEE Smart Cities", https://smartcities.ieee.org/images/files /pdf/IEEE_Smart_Cities-_Flyer_Nov_2017.pdf. consultado 30 /06 / 2025
[3] International Organization for Standardization, "Movilidad en ciudades inteligentes: Transformar el futuro del transporte urbano", https://www.iso.org/es/transporte/movilidad-ciudades-inteligentes, consultado 30 /06 / 2025.
[4] International Organization for Standardization, "¿Qué es la movilidad en ciudades inteligentes?", https://www.iso.org/es/transporte/movilidad-ciudades-inteligentes#toc2 consultado 30 /06 / 2025
[5] Singh, D.; Sud, S.K.; Singh, S.K. Optical Character Recognition (OCR) based Vehicle's License Plate Recognition System Using Python and OpenCV. 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC), 2021, pp. 1–6. doi:black10.1109/ISPCC53510.2021.9676015.
[6] J.-G. Sari, R.; Nugroho, A.S.; Saputra, A.R.S. License Plate Detection Using OCR Method with Raspberry Pi. 2022 4th International Sari, R.; Nugroho, A.S.; Saputra, A.R.S. License Plate Detection Using OCR Method with Raspberry Pi. 2022 4th International
[7] Alotaibi, B.; Alghamdi, M.A.; Alzahrani, A.; Alzahrani, A.; Alzahrani, A. A Next-Generation Secure Cloud- Based Deep Learning License Plate Recognition for Smart Cities. IEEE Access 2021, 9, 123456–123467. doi:black10.1109/ACCESS.2021.1234567.
[8] IBM. ¿Qué es la inteligencia artificial (IA)? https://www.ibm.com/mx-es/topics/artificial-intelligence, 2024. Accedido el 3 de diciembre de 2024.
[9] Corporation, I. Computer Vision: The Next Frontier in AI. Online, 2024. Accedido el 20 de junio de 2024.
[10] Mateu Villa, M.; Sánchez Yáñez, R.E. Fundamentos de la reducción de ruido en imágenes. Universidad de Guanajuato 2023.
[11] Amazon Web Services (AWS). ¿Qué es OCR? Reconocimiento óptico de caracteres, 2024. Último acceso: 12 de diciembre de 2024.
[12] Ultralytics. YOLO: Tecnología de detección de objetos en tiempo real, 2024. Último acceso: 12 de diciembre de 2024.
[13] A. Rayes, S. Salam, "Internet of Things From Hype to Reality", Second Edition, 2019, Springer Natures, Switzerland, pp 2-22
[14] S. Monk, "Programming the Raspbery Pi: Getting Started with Python", 1st Edition, Union Kingdom, 2013.
[15] SimCom. SIM808 Hardware Design. Online, 2017. Accedido el 5 de octubre de 2023.
[16] Amazon Web Services (AWS). ¿Qué es una base de datos?, 2024. Último acceso: 12 de diciembre de 2024.
[17] Oracle. ¿Qué es una base de datos?, 2024. Último acceso: 12 de diciembre de 2024.
[18] Amazon Web Services (AWS). AWS Documentation, 2024. Consultado el 14 de diciembre de 2024.