



Urban Eye: Transformación del monitoreo pasivo en analítica predictiva mediante Deep Learning y computación perimetral

Juan Manuel Sierra Quintero

Fabian Orlando Moreno Poveda

Jorge Steven Gaitán Barreto

Universidad Ean

Facultad de Ingeniería

Programa de Ingeniería Industrial e Ingeniería de Sistemas

Bogotá, Colombia

01/06/2026

Nota de aceptación:

Firma del jurado

Firma del jurado

Firma del director del trabajo de grado

Ciudad, día/mes/año

Dedicatoria

A nuestros padres, por ser el cimiento sobre el cual construimos nuestros sueños. Gracias por su paciencia infinita durante las largas jornadas de codificación y redacción, por su apoyo moral y por creer en nosotros incluso cuando el camino se tornaba complejo. Este logro es tan suyo como nuestro, pues su ejemplo de perseverancia es la energía que impulsó la creación de este proyecto.

Cuando la tecnología inalámbrica se aplique perfectamente, toda la tierra se convertirá en un cerebro enorme... seremos capaces de comunicarnos entre nosotros instantáneamente, independientemente de la distancia.

Nikola Tesla.

predictiva mediante Deep Learning y computación perimetral**Resumen**

Urban Eye es un prototipo que tiene como objetivo poder fortalecer los sistemas tradicionales de videovigilancia mediante la utilización de visión computacional. Edge Computing y Deep Learning. De esta manera la presente investigación surge debido a las limitaciones actuales que se encuentran presentes en los sistemas de monitoreo tradicionales, ya que estos dependen en gran medida de una supervisión humana, lo cual podría generar posibles retrasos en lo que se refiere a la detección de incidentes y anomalías en los entornos urbanos.

Por lo mencionado anteriormente este proyecto se enfoco en desarrollar un sistema que fuera capaz de poder identificar eventos vehiculares en tiempo real mediante la implementación del algoritmo YOLO, Este algoritmo fue seleccionado debido a su gran capacidad para detectar objetos y su velocidad de procesamiento. además de esto se implementó una arquitectura Edge Computing para poder ejecutar parte de este procesamiento de manera local, reduciendo así la dependencia de servicios externos y disminuyendo la latencia para poder mejorar la eficiencia del sistema.

Por otra parte, la metodología que fue empleada corresponde a un enfoque cuantitativo experimental que esta orientado a evaluar el desempeño del prototipo en unos escenarios urbanos de prueba. De esta manera durante la realización de las pruebas, el sistema logró mantener un procesamiento cercano a los 30 fotogramas por segundo, permitiendo así una detección estable de vehículos y un funcionamiento adecuado en condiciones de buena iluminación.

Asimismo, el proyecto integró fundamentos de privacidad por diseño, enfocándose en el procesamiento local y la gestión de metadatos para minimizar riesgos vinculados al manejo de información delicada. De esta manera Urban Eye representa una alternativa tecnológica que esta orientada al fortalecimiento de soluciones de monitoreo inteligente en entornos urbanos.

Palabras clave: Visión computacional, Deep Learning, Videovigilancia inteligente, Edge Computing, inteligencia artificial, monitoreo urbano, Yolo, Movilidad urbana.

Abstract

Urban Eye is a prototype that aims to strengthen traditional video surveillance systems through the use of computer vision. Edge Computing and Deep Learning. In this way, the present research arises due to the current limitations that are present in traditional monitoring systems, since these depend largely on human supervision, which could generate possible delays in terms of the detection of incidents and anomalies in urban environments.

As mentioned above, this project focused on developing a system that was capable of identifying vehicular events in real time through the implementation of the YOLO algorithm. This algorithm was selected due to its great capacity to detect objects and its processing speed. In addition to this, an Edge Computing architecture was implemented to be able to execute part of this processing locally, thus reducing dependence on external services and reducing latency to improve system efficiency.

On the other hand, the methodology that was used corresponds to a quantitative experimental approach that is aimed at evaluating the performance of the prototype in urban test scenarios. In this way, during the tests, the system managed to maintain processing close to 30 frames per second, thus allowing stable vehicle detection and adequate operation in good lighting conditions. Likewise, the project integrated privacy by design fundamentals, focusing on local processing and metadata management to minimize risks linked to the handling of sensitive information. In this way, Urban Eye represents a technological alternative that is aimed at strengthening intelligent monitoring solutions in urban environments.

Keywords: Computational vision, Deep Learning, Intelligent video surveillance, Edge Computing, artificial intelligence, urban monitoring, YOLO, Urban Mobility

predictiva mediante Deep Learning y computación perimetral

Introducción

En la última década, se ha dado un crecimiento en la población y el uso de vehículos en las grandes ciudades, podemos hablar de Bogotá ya que esta es una de las ciudades en la cual se ha generado una mayor presión sobre la infraestructura vial y los sistemas de monitoreo urbano. Esta situación ha generado diferentes dificultades en lo que se refiere a la movilidad y el incremento en los tiempos de respuesta ante incidentes que puedan ocurrir además de esto también se ha generado un importante deterioro en la malla vial, esto debido a que en la actualidad se cuenta con una gran presencia de baches, escombros y objetos sobre las calzadas, representando un riesgo para la seguridad vial y teniendo efectos negativos en la congestión vehicular.

Dicho lo anterior cabe mencionar que, los sistemas de vigilancia actuales dependen en gran medida de la supervisión humana, limitando de manera significativa la capacidad de poder identificar irregularidades de una manera precisa y continua. **(Bettencourt, 2021).**

De esta manera en respuesta a la problemática mencionada anteriormente, diferentes investigaciones han planteado el uso de tecnologías asociadas a ciudades inteligentes, teniendo como objetivo mejorar la gestión urbana mediante la realización de análisis de datos en tiempo real y apoyándose del uso de herramientas digitales **(Giffinger 2007)**. Es así como dentro de estas tecnologías, la visión artificial y el aprendizaje profundo han adquirido relevancia debido a su gran capacidad de detectar y clasificar los objetos de manera automática en escenarios complejos.

Asimismo, estudios recientes han logrado demostrar que modelos basados en Redes Neuronales Convolucionales y Arquitecturas como YOLO permiten realizar la detección de objetos en tiempo real teniendo altos niveles de precisión, es esto lo que convierte estas herramientas en una alternativa viable para aplicaciones relacionadas con movilidad y seguridad vial **(Redmon 2016).**

predictiva mediante Deep Learning y computación perimetral

Es a partir de esto que se propone el desarrollo de Urban Eye, una plataforma basada en Python la cual está orientada a detectar automáticamente anomalías viales mediante la visión artificial. Cabe mencionar que este sistema utilizara el algoritmo YOLO para poder identificar baches y Objetos que estén presentes en las vías, y de esta manera generar alertas en tiempo real para mejorar la toma de decisiones por parte de las entidades encargadas del control y mantenimiento vial, Por otra parte, se utilizara la arquitectura Edge Computing para reducir latencia y la dependencia de servidores remotos, ya que permite el procesamiento de los datos en dispositivos locales **(Mell & Grace, 2011)**.

El proyecto contempla la aplicación del principio de privacidad por diseño en lo cual se integrarán mecanismos de anonimización automática para garantizar la protección de identidad de los ciudadanos y promover un uso responsable de la información **(Cavoukian, 2011)**. De esta manera, Urban Eye tiene como objetivo contribuir al fortalecimiento de la movilidad Urbana mediante una herramienta tecnológica que es capaz de apoyar la detección temprana de riesgos en las calles de la ciudad.

El Objetivo de esta investigación es desarrollar un sistema basado en visión artificial y Edge Computing para detectar automáticamente anomalías viales en tiempo real, además de esto se plantea como hipótesis que la implementación de un sistema de detección basado en YOLO que será ejecutado sobre una arquitectura Edge Computing tendra un impacto positivo en la precisión de identificación de anomalías viales, contribuyendo así a optimizar procesos de monitoreo y control de movilidad urbana.

predictiva mediante Deep Learning y computación perimetral**Marco teórico**

El desarrollo de sistemas de monitoreo proactivo para la ciudad de Bogotá requiere una base conceptual que integre la planificación urbana con tecnologías avanzadas para procesamiento de datos. Según (Giffiner 2007), una ciudad inteligente no está limitada únicamente a la incorporación de una infraestructura tecnológica, sino que además de esto implica una capacidad para poder administrar los recursos urbanos de una manera eficiente mejorando así la calidad de vida de los ciudadanos y también promover un desarrollo sostenible. Es así como la movilidad urbana constituye uno de los principales componentes de la gestión pública.

De esta manera, (Bettancourt 2021) plantea que las ciudades funcionan como unas redes que están interconectadas, en las cuales pequeñas alteraciones en la infraestructura pueden generar grandes impactos sobre la movilidad y la seguridad de las personas. Es así como la presencia de baches, escombros u objetos que estén presentes en las vías pueden aumentar significativamente el riesgo de accidentes, esta situación evidencia las limitaciones de los sistemas tradicionales que están basados y dependen únicamente de la supervisión humana. De esto surge la necesidad de que las ciudades implementen mecanismos tecnológicos que sean capaces de responder en tiempo real ante situaciones que puedan afectar la movilidad y el orden urbano.

En respuesta a esta problemática, la movilidad inteligente propone la incorporación de las tecnologías digitales que estén orientadas a automatizar procesos de identificación de anomalías, esto mediante un análisis en tiempo real de datos. Este enfoque está fundamentado en la visión por computadora y en el aprendizaje profundo, este último un área de la inteligencia artificial que está enfocada en el reconocimiento automático de patrones complejos. En el estudio realizado por (Goodfellow 2016) y su equipo explican que las Redes Neuronales Convolucionales permiten

predictiva mediante Deep Learning y computación perimetral

extraer características de imágenes digitales, esto facilita tareas como lo son la clasificación, la segmentación y la detección de objetos en tiempo real. Debido a las capacidades mencionadas anteriormente, es posible desarrollar sistemas complejos que interpreten automáticamente escenas urbanas y de esta manera puedan detectar riesgos que se presenten en la infraestructura vial.

La aplicación de Redes Neuronales Convolucionales en sistemas de monitoreo vial ha podido tomar relevancia debido al desarrollo de arquitecturas de detección rápida como lo es el algoritmo YOLO. Según (Redmon 2016) y su equipo, este modelo revolucionó la detección de objetos al tratar la localización y la clasificación como un único problema de regresión, esto permite analizar imágenes completas en tan solo una pasada de la red neuronal, debido principalmente a que esta metodología alcanza velocidades superiores a los 45 FPS (Fotogramas Por Segundo), esta es una característica de gran importancia para escenarios urbanos en los cuales la toma de decisiones debe ser en tiempo real. De esta manera YOLO representa una herramienta adecuada para la detección automática de los baches, los obstáculos, y demás anomalías que puedan presentarse en las vías de Bogotá.

Sin embargo, este procesamiento continuo de grandes volúmenes de video se enfrenta a grandes desafíos que están relacionados principalmente con la latencia y el consumo en el ancho de banda. (Mell & Grance, 2011) mencionan que los modelos tradicionales que están basados exclusivamente en computación en la nube son dependientes de la conectividad permanente y de recursos remotos de procesamiento. Esto es un obstáculo para aplicaciones de movilidad urbana ya que podría afectar negativamente a la capacidad del sistema frente a los eventos que requieran una acción de manera inmediata.

predictiva mediante Deep Learning y computación perimetral

No Obstante, frente a esta problemática previamente mencionada surge la arquitectura Edge Computing. (Shi, 2016) y su equipo definen este paradigma como una metodología la cual desplaza la capacidad de cómputo hacia la periferia de la red, esto teniendo como ventaja procesar los datos cerca del lugar donde son generados, reduciendo así significativamente la latencia mejorando de esta manera la autonomía operativa de los sistemas. Por lo mencionado anteriormente para el presente proyecto Urban Eye, el uso de Edge Computing Permite ejecutar modelos de visión artificial directamente sobre los dispositivos especializados que son utilizados, esto garantiza unos tiempos de respuesta rápidos y una mayor estabilidad operativa incluso si ocurren fallos en la conexión a internet.

Por otra parte, desde un punto de vista computacional, la optimización de los modelos Deep Learning conforma un aspecto de vital importancia para su implementación en dispositivos que cuenten con recursos limitados. (Han, 2015) y su equipo demuestran que las técnicas como poda de modelos y la cuantización de pesos permite que se pueda reducir el tamaño y el consumo de energía en redes neuronales sin que se vea afectada significativamente la precisión en la detección, dicho esto se puede decir que estas estrategias tienen una gran importancia en proyectos de monitoreo urbano, debido principalmente a que facilitan la ejecución de algoritmos de inteligencia artificial en dispositivos que cuenten con hardware de bajo consumo, esto tiene grandes ventajas ya que mejora la economía del sistema y la sostenibilidad tecnológica.

Adicionalmente, cabe mencionar que el análisis de imágenes urbanas requiere de unas herramientas que permitan interpretar correctamente las dimensiones presentes en cada caso. Hartley y Zisserman (2004) explican que la calibración de cámaras permite estimar medidas reales mediante la utilización de imágenes bidimensionales. Esto complementa la detección de objetos

predictiva mediante Deep Learning y computación perimetral

realizada por los modelos de visión artificial, permitiendo de esta manera clasificar la gravedad de los daños en la infraestructura y así determinar qué nivel de riesgo.

Adicional a esto, Batty (2013) apoya la idea de que las ciudades modernas producen grandes volúmenes de información la cual puede ser analizada mediante herramientas Big Data, para así poder optimizar la toma de decisiones. De esta manera, el procesamiento de datos relacionados con baches, congestión vehicular y zonas críticas permite identificar los patrones de comportamiento urbano y así poder priorizar intervenciones en sectores donde existan mayores riesgos de afectar a la movilidad.

Otro aspecto importante es la interoperabilidad entre sistemas tecnológicos, ya que esto constituye un elemento importante en el contexto de ciudades inteligentes. Roche (2012) y su equipo señalan que la efectividad de las plataformas urbanas depende en gran medida de la capacidad que posean para integrar la información proveniente de diferentes entidades y sistemas de gestión. Por lo tanto, la integración de Urban Eye con sistemas de información geográfica, permitirá generar reportes georreferenciados y facilitar de esta manera la coordinación entre los organismos responsables de la movilidad y el mantenimiento de las vías en la ciudad.

Finalmente, cabe señalar que el desarrollo de sistemas de monitoreo debe tener en cuenta implicaciones éticas que están relacionadas con la privacidad de los ciudadanos. (Cavoukian 2011) establece que el principio de la privacidad por diseño radica en la incorporación de mecanismos de protección de datos desde las etapas iniciales de la construcción tecnológica. Por tal motivo, Urban Eye considera la anonimización automática de rostros y matrículas mediante la utilización de técnicas de visión artificial, garantizando de esta manera que el monitoreo se realice respetando la protección de datos personales y los derechos de los ciudadanos.

Estado del arte (Metodología PRISMA)

La construcción del estado del arte fue desarrollada mediante una revisión sistemática de literatura basada en la metodología PRISMA propuesta por David Moher y colaboradores (2009), con el fin de poder garantizar un proceso transparente, organizado y verificable en la selección realizada de artículos científicos que están relacionados con la visión computacional, detección de daños viales y procesamiento en el borde. La revisión realizada permitió identificar investigaciones recientes que tienen un enfoque en el uso de la Inteligencia Artificial aplicada a una movilidad urbana y seguridad vial en entornos de ciudades inteligentes.

Dicho lo anterior cabe mencionar que para la búsqueda bibliográfica fueron utilizadas diferentes bases de datos como lo son Scopus, Web of Science, IEEE Xplore, esto debido a su reconocimiento en áreas relacionadas con inteligencia artificial, visión computacional y ciudades inteligentes. De esta manera la ecuación de búsqueda utilizada fue la siguiente:

(“YOLO” OR “Object Detection”) AND (“Road Damage Detection” OR “Pothole Detection”)
AND (“Edge Computing” OR “Smart City”)

La revisión realizada incluyo artículos que fueron publicados entre los años 2020 y 2026. Como criterios de inclusión se consideraron artículos relacionados con detección de daños viales mediante la visión artificial así como estudios que emplearon modelos YOLO o redes neuronales convolucionales así como trabajos enfocados en movilidad inteligente. Por otra parte, fueron excluidos documentos duplicados, artículos sin resultados experimentales o trabajos enfocados únicamente en simulaciones teóricas y demás problemáticas que no abordaran problemas relacionados con una infraestructura vial o monitoreo urbano.

predictiva mediante Deep Learning y computación perimetral

De esta manera en el análisis realizado se logró evidenciar que los modelos de visión artificial basados en Deep Learning lograron convertirse en una de las principales herramientas para la detección automática de daños en las carreteras, Diversos estudios han destacado el uso del algoritmo YOLO esto debido a su gran capacidad para realizar detecciones en tiempo real teniendo altos niveles de precisión. De esta manera cabe mencionar que investigaciones como la de (Guo y Zhang, 2022) y (Wan,2022) y su equipo lograron demostrar que las versiones optimizadas de YOLO permiten poder reducir el costo computacional sin tener afectaciones significativas en la precisión del sistema.

De igual manera se puede decir que la literatura científica logra reflejar una tendencia creciente orientada a la implementación de arquitecturas ligeras y procesamiento local para de esta manera poder reducir la latencia y poder contribuir a mejorar la eficiencia operativa de los sistemas inteligentes de transporte urbano. Con base en las investigaciones revisadas se encontró que los estudios coinciden en la idea de que el procesamiento cercano a la fuente de datos permite la obtención de respuestas más rápidas ante diferentes eventos críticos en las vías públicas. Lo mencionado anteriormente es un aspecto relevante en las ciudades que cuentan con un alto nivel de congestión vehicular esto debido principalmente a que en estas ciudades el tiempo de reacción ante incidentes es un factor de vital importancia para la seguridad vial.

Otro de los aspectos relevantes que fueron encontrados en los artículos que fueron analizados corresponde a las limitaciones presentes en escenarios reales. (Park. Tran. Lee, 2021) reportan una disminución en la precisión de los algoritmos cuando las condiciones climáticas cuentan con una baja iluminación, o presencia de lluvia o sombras.

predictiva mediante Deep Learning y computación perimetral

Asimismo, diferentes autores coinciden con que los modelos de visión artificial deben requerir grandes volúmenes de datos para poder alcanzar niveles óptimos en su desempeño, esto representa un desafío importante en las ciudades latinoamericanas ya que los conjuntos de datos que obtienen son limitados.

Por otra parte, las investigaciones resaltan la importancia que tiene poder integrar los sistemas inteligentes de monitoreo con diferentes herramientas de análisis de datos y gestión urbana.

(Ramesh, 2022) y su equipo destacan en su investigación que los sistemas colaborativos que están basados en Deep Learning permiten que se puedan transformar las detecciones realizadas por las cámaras en información que sea útil para la administración de la infraestructura vial. Es este enfoque el que fortalece la toma de decisiones facilitando de esta manera la priorización de intervenciones en las zonas críticas de la movilidad urbana.

Finalmente a partir de las revisiones realizadas, se logró identificar como principal vacío de la investigación una escasa adaptación de los sistemas inteligentes al contexto urbano en Latinoamérica, especialmente en grandes ciudades, aunque se tiene la existencia de investigaciones que están enfocadas en la detección de daños viles mediante la utilización de inteligencia artificial, muchas de estas investigaciones fueron desarrolladas en unos entornos controlados o en países que cuentan con una infraestructura vial diferente, Por lo mencionado anteriormente, el presente proyecto titulado **“Urban Eye: Transformación del monitoreo pasivo en analítica predictiva mediante Deep Learning y computación perimetral”** busca aportar una solución que está basada en Python, visión artificial y Edge Computing orientada a las condiciones reales de la movilidad urbana en la ciudad de Bogotá, esto integrando mecanismos de privacidad y monitoreo continuo en tiempo real.

predictiva mediante Deep Learning y computación perimetral

Autor(es) / Año	Título del estudio	Objetivo principal	Metodología empleada	Variables analizadas	Principales hallazgos	Limitaciones reportadas	Relación con el proyecto actual
Wan et al. (2022)	YOLO-LRDD: a lightweight method for road damage detection based on improved YOLOv5s	Mejorar la detección de los daños viales mediante una versión ligera de YOLOv5s	Deep Learning y visión por computador usando YOLOv5s optimizado.	Precisión de detección, complejidad computacional.	El modelo obtuvo una alta precisión y logró reducir el costo computacional	Obtiene un rendimiento inferior en escenarios complejos	Fundamenta el uso de YOLO optimizado para detección vial en Urban Eye.
Guo y Zhang (2022)	Road damage detection algorithm for improved YOLOv5	Desarrollar un algoritmo eficiente para la detección de daños viales	Redes neuronales convolucionales y una arquitectura YOLOv5	Precisión y velocidad de procesamiento	El modelo logró reducir parámetros sin que se afectara la precisión	Es muy sensible ante variaciones de iluminación	Sustenta la implementación de modelos ligeros para monitoreo vial en tiempo real.
Park, Tran y Lee (2021)	Application of Various YOLO Models for Computer Vision-Based Real-Time Pothole Detection	Comparar diferentes versiones de YOLO para la detección de baches en tiempo real	Comparacion experimental entre diferentes modelos YOLO	Rendimiento computacional y precisión	YOLO detecto de manera eficiente los baches usando imágenes reales	Presenta dificultades en condiciones climáticas adversas	Relaciona el proyecto con detección de huecos en escenarios urbanos reales.
Ramesh et al. (2022)	Cloud-Based Collaborative Road-Damage Monitoring with Deep Learning and Smartphones	Evaluar un sistema colaborativo de monitoreo vial usando Deep Learning.	procesamiento inteligente de imágenes.	Precisión de clasificación y monitoreo vial.	El sistema permitió detectar daños en carreteras mediante la utilización de dispositivos convencionales	Se debe contar con imágenes en alta calidad para asegurar su correcto funcionamiento.	Aporta bases para sistemas inteligentes de monitoreo urbano aplicados a ciudades inteligentes.

predictiva mediante Deep Learning y computación perimetral**Metodología**

La presente investigación fue desarrollada mediante un enfoque cuantitativo con un diseño tecnológico orientado a la creación de una solución fundamentada en un software para la detección automática de anomalías viales, utilizando una visión artificial. Dicho lo anterior la metodología se centró en el análisis de variables relacionadas con la precisión de los algoritmos, la velocidad de procesamiento y la capacidad de respuesta en entornos urbanos, tomando como referencia los lineamientos metodológicos que fueron propuestos por Hernández-Sampieri y Mendoza (2018). El alcance de este estudio es descriptivo-correlacional, esto se debe a que busca identificar problemáticas presentes en la infraestructura vial de la ciudad y analizar la relación entre las anomalías detectadas y la eficiencia en los procesos de movilidad.

Enfoque y alcance de la investigación.

El diseño de la investigación corresponde a un estudio no experimental de Tipo transversal, debido a que la recolección y el análisis de la información se realizaron sin manipular las variables físicas de la infraestructura vial. De esta manera, el proyecto adoptó principios del ciclo de vida de desarrollo de software, permitiendo de esta manera una construcción modular de la plataforma de Urban Eye, adaptada principalmente a proyectos de inteligencia artificial y visión computacional (Sommerville, 2011)

predictiva mediante Deep Learning y computación perimetral**Fases de la solución tecnológica.****Fase de adquisición y procesamiento de datos:**

Se recolectaron secuencias de video e imágenes de las diferentes vías de Bogotá mediante la utilización de las cámaras integradas en los dispositivos móviles. Posterior a esto, los datos fueron organizados y etiquetados mediante la utilización de herramientas de anotación de imágenes empleadas en proyectos de visión artificial, con el objetivo de construir un conjunto de datos enfocado en baches y obstáculos presentes en las vías.

Fase de entrenamiento del modelo de visión artificial:

Esta fase de entrenamiento se realizó utilizando como lenguaje de programación Python, de esta manera se implementó una arquitectura que está basada en YOLO para la detección de diferentes anomalías en tiempo real, asimismo se aplicaron diferentes técnicas de Transfer Learning esto teniendo como finalidad poder optimizar el proceso de entrenamiento y mejorar así la capacidad de detección en escenarios que cuenten con una baja iluminación o presencia de condiciones climáticas difíciles (Redmon y su equipo, 2016)

Fase de implementación en Edge Computing:

El modelo optimizado fue desplegado sobre dispositivos de procesamiento local, con el propósito de reducir la latencia y disminuir la dependencia de servicios en la nube. Es así como en esta fase se desarrollaron scripts en Python para gestionar el procesamiento de video en tiempo real y la generación de alertas georreferenciadas, asegurando que la latencia no supere los 100ms (Shi y su equipo 2016).

predictiva mediante Deep Learning y computación perimetral**Fase de validación y pruebas de campo:**

Finalmente se realizaron pruebas piloto en diferentes sectores viales de la ciudad esto teniendo como finalidad evaluar el desempeño del sistema frente a escenarios reales, es así como durante esta etapa se realizó un análisis de la tasa de detección correcta, así como la presencia de falsos positivos y estabilidad en general del sistema ante diferentes situaciones como baja luminosidad, lluvia o demás situaciones climáticas.

Instrumentos

En el desarrollo de la investigación se emplearon instrumentos de medición orientados a evaluar factores como el rendimiento computacional, la precisión estadística del sistema de detección entre otros. Dicho lo anterior entre los principales instrumentos se encuentran las métricas de visión artificial, incluyendo así la matriz de confusión, precisión y f1-score, estas herramientas son ampliamente utilizadas para poder validar modelos de clasificación y detección de objetos (Sokolova & Lapalme, 2009). Asimismo, se utilizaron herramientas de benchmarking para poder monitorear el consumo de la CPU, GPU y temperaturas del hardware durante la ejecución continua del sistema.

Técnicas para el análisis de la información:

Para el análisis de los datos obtenidos, se hizo uso del paquete estadístico Python Pandas, junto con herramientas de visualización como Matplotlib. Asimismo, se llevó a cabo un análisis de varianza (ANOVA) con el fin de identificar si hay diferencias significativas en la precisión del sistema bajo distintas condiciones de luz (día/noche) y climáticas. Los hallazgos permitieron evaluar el comportamiento del prototipo y establecer criterios de mejora para el monitoreo vehicular basado en evidencia cuantitativa (Hernández-Sampieri & Mendoza, 2018)

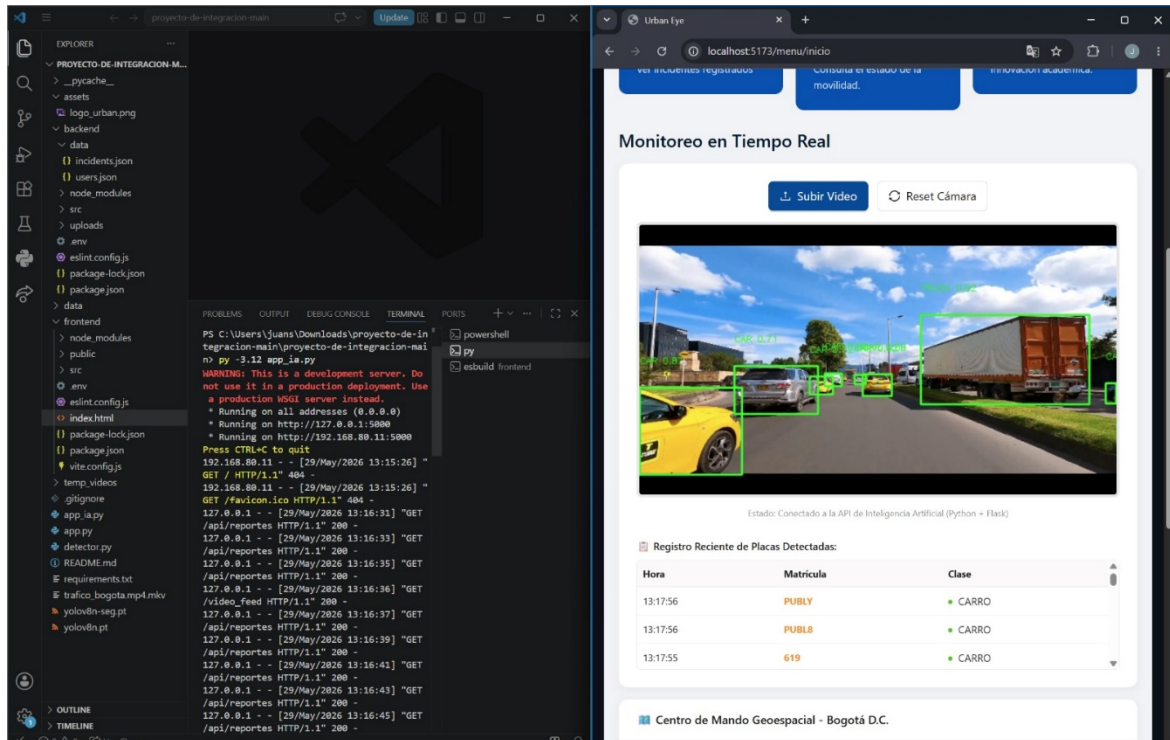
Resultados

Pruebas funcionales y validación del prototipo

La fase de validación del prototipo Urban Eye fue realizada mediante el procesamiento de videos de tráfico urbano que corresponden a vías de la ciudad de Bogotá. Dicho lo anterior el objetivo principal de esta etapa fue evaluar el funcionamiento general del sistema, la detección de vehículos, el rendimiento de procesamiento local y el comportamiento de la interfaz que fue desarrollada.

Durante la realización de estas pruebas, el sistema logró identificar diferentes tipos de vehículos, entre los cuales se encuentran automóviles particulares, taxis y vehículos de carga, asimismo, el prototipo presento un funcionamiento estable en gran parte de los escenarios evaluados. De igual manera, se puede mencionar que el sistema pudo realizar una lectura básica de las placas vehiculares, especialmente en condiciones adecuadas de iluminación.

Los resultados obtenidos permiten evidenciar que el uso de modelos de visión artificial junto con el procesamiento local representa una alternativa viable para los sistemas de monitoreo urbano en tiempo real. A continuación, serán presentados algunos de los resultados obtenidos en la realización de pruebas funcionales del prototipo.



Nota. La imagen mostrada anteriormente muestra la detección de vehículos en tiempo real, así como la visualización de información del sistema durante su tiempo de ejecución. **Elaboración propia.**

Métricas de rendimiento del sistema.

Para realizar una evaluación básica en el funcionamiento del modelo de detección se proceso un conjunto de pruebas que este compuesto por diferentes fragmentos de video urbano. De esta manera los resultados que fueron obtenidos serán presentados en la siguiente tabla.

predictiva mediante Deep Learning y computación perimetral

Tabla 2

Aspecto evaluado	Resultado obtenido
Detección de vehículos	Funcionamiento estable
Fluidez del video	Reproducción continua sin interrupciones
Lectura de placas	Resultados eficientes con buena iluminación
Uso de hardware en modo local	Consumo moderado de recursos
Interfaz geográfica	Visualización clara
Procesamiento local	Respuesta rápida y eficiente

Nota. Con base en los resultados obtenidos, se puede evidenciar que el sistema presenta un rendimiento estable, manteniendo una adecuada velocidad de análisis y un comportamiento funcional en los escenarios evaluados. **Elaboración propia.**

Análisis de procesamiento local.

Uno de los aspectos más relevantes que fueron observados durante las pruebas fue la capacidad del sistema para poder ejecutarse mediante procesamiento local utilizando la GPU del hardware. Esto permitió mantener un rendimiento fluido sin necesidad de depender completamente de servicios externos o procesamiento en la nube.

Durante la ejecución de pruebas funcionales, el comportamiento del hardware pudo mantenerse estable en la mayor parte del tiempo. Además, la interfaz desarrollada permitió visualizar en tiempo real los eventos que fueron detectados por el sistema. De esta manea con el propósito de poder evaluar el rendimiento general del prototipo, se realizaron mediciones aproximadas de tiempo de procesamiento.

predictiva mediante Deep Learning y computación perimetral

Tabla 3

Proceso	Tiempo mínimo (ms)	Tiempo máximo (ms)	Tiempo promedio (ms)
Lectura de video	2.1	5.4	3.2
Detección mediante YOLO	18.5	26.1	21.4
Procesamiento de visualización	4.2	9.8	6.5
Tiempo total aproximado	25	42.1	31.5

Nota. En base a los resultados obtenidos se puede evidenciar que el sistema puede trabajar de manera fluida en tiempo real manteniendo una velocidad cercana a los 30 FPS. **Elaboración propia.**

Integración del sistema y procesamiento paralelo.

Para evaluar el comportamiento general del prototipo, se realizaron diferentes pruebas de integración entre el módulo de visión artificial que fue desarrollado en Python y la interfaz del sistema. El backend fue diseñado utilizando procesamiento en hilos, permitiendo ejecutar la captura de video y detección de objetos de una manera simultanea

predictiva mediante Deep Learning y computación perimetral

Figura 5 Estructura de la Clase Concurrente UrbanEyeDetector para Procesamiento en Hilos

```
detector.py > UrbanEyeDetector > start
3  from ultralytics import YOLO
4
5  class UrbanEyeDetector:
6      def __init__(self, model_path="yolov8n.pt", video_source=0):
7          """
8          Inicializa el detector de Urban Eye.
9          :param model_path: El modelo de IA de YOLO a usar.
10         :param video_source: 0 para la cámara web por defecto
11         """
12         self.model = YOLO(model_path)
13         self.video_source = video_source
14         self.is_running = False
15         self.thread = None
16         self.cap = None
17         self.current_frame = None
18
19     def start(self):
20         """Arranca la lectura de la cámara en un hilo separad
21         if not self.is_running:
22             self.is_running = True
23             self.cap = cv2.VideoCapture(self.video_source)
24             self.thread = threading.Thread(target=self._detect
25             self.thread.start()
26
27     def _detection_loop(self):
28         """Bucle interno que analiza los cuadros de video."""
29         while self.is_running and self.cap.isOpened():
30             ret, frame = self.cap.read()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\juans\Downloads\proyecto-de-integracion-main\proyecto-de-integracion-main> py -3.12 app_ia.py

127.0.0.1 - - [29/May/2026 13:17:13] "GET /api/reportes HTTP/1.1" 200 -

127.0.0.1 - - [29/May/2026 13:17:15] "GET /api/reportes HTTP/1.1" 200 -

127.0.0.1 - - [29/May/2026 13:17:17] "GET /api/reportes HTTP/1.1" 200 -

127.0.0.1 - - [29/May/2026 13:17:17] "OPTIONS /upload_video HTTP/1.1" 200 -

127.0.0.1 - - [29/May/2026 13:17:19] "GET /api/reportes HTTP/1.1" 200 -

127.0.0.1 - - [29/May/2026 13:17:21] "GET /api/reportes HTTP/1.1" 200 -

127.0.0.1 - - [29/May/2026 13:17:22] "POST /upload_video HTTP/1.1" 200 -

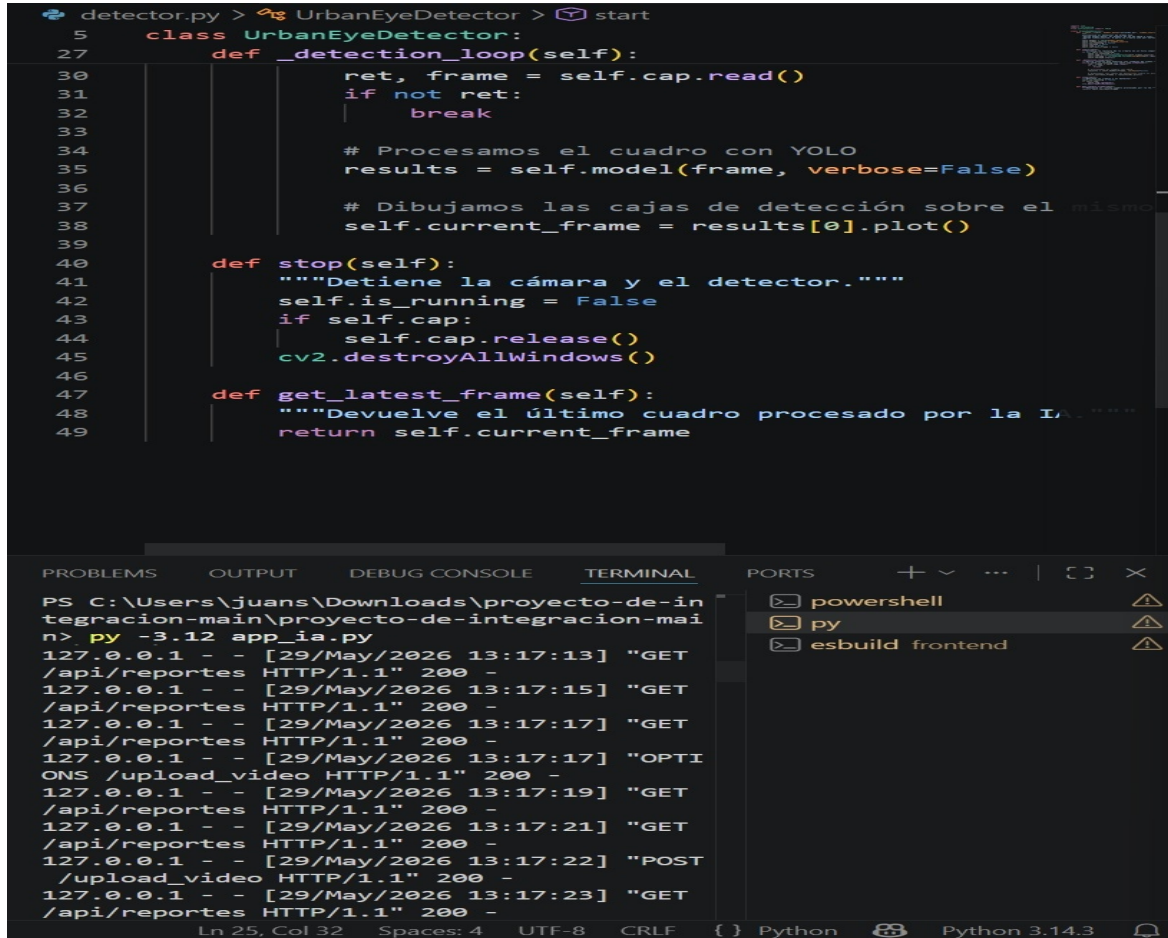
127.0.0.1 - - [29/May/2026 13:17:23] "GET /api/reportes HTTP/1.1" 200 -

Ln 25, Col 32 Spaces: 4 UTF-8 CRLF {} Python esbuild frontend

En la captura mostrada anteriormente se evidencia el uso de procesamiento en hilos para mantener estable la detección de video y funcionamiento general del sistema. **Elaboración propia.**

predictiva mediante Deep Learning y computación perimetral

Figura 6 Bucle de Inferencia YOLOv8 y Consola de Peticiones del Servidor API Local



```
detector.py > UrbanEyeDetector > start
5 class UrbanEyeDetector:
27     def _detection_loop(self):
30         ret, frame = self.cap.read()
31         if not ret:
32             break
33
34         # Procesamos el cuadro con YOLO
35         results = self.model(frame, verbose=False)
36
37         # Dibujamos las cajas de detección sobre el mismo
38         self.current_frame = results[0].plot()
39
40     def stop(self):
41         """Detiene la cámara y el detector."""
42         self.is_running = False
43         if self.cap:
44             self.cap.release()
45             cv2.destroyAllWindows()
46
47     def get_latest_frame(self):
48         """Devuelve el último cuadro procesado por la IA"""
49         return self.current_frame
```

```
PS C:\Users\juans\Downloads\proyecto-de-integracion-main\proyecto-de-integracion-main> py -3.12 app_ia.py
127.0.0.1 - - [29/May/2026 13:17:13] "GET /api/reportes HTTP/1.1" 200 -
127.0.0.1 - - [29/May/2026 13:17:15] "GET /api/reportes HTTP/1.1" 200 -
127.0.0.1 - - [29/May/2026 13:17:17] "GET /api/reportes HTTP/1.1" 200 -
127.0.0.1 - - [29/May/2026 13:17:17] "OPTIONS /upload_video HTTP/1.1" 200 -
127.0.0.1 - - [29/May/2026 13:17:19] "GET /api/reportes HTTP/1.1" 200 -
127.0.0.1 - - [29/May/2026 13:17:21] "GET /api/reportes HTTP/1.1" 200 -
127.0.0.1 - - [29/May/2026 13:17:22] "POST /upload_video HTTP/1.1" 200 -
127.0.0.1 - - [29/May/2026 13:17:23] "GET /api/reportes HTTP/1.1" 200 -
```

En la figura numero 6 se muestra el funcionamiento del sistema durante el procesamiento de video y las peticiones que fueron realizadas entre el módulo de visión artificial y la interfaz del prototipo.

Elaboración propia. De esta manera se puede decir que, durante las pruebas realizadas, el prototipo pudo mantener una comunicación estable entre los diferentes componentes del sistema, permitiendo el procesamiento de video, la detención de eventos y la observación de resultados en tiempo real sin presencia de interrupciones.

predictiva mediante Deep Learning y computación perimetral**Detección de vehículos.**

Durante las pruebas que fueron realizadas, el sistema pudo identificar varios vehículos de manera simultánea obteniendo así un comportamiento adecuado en diferentes condiciones con flujo vehicular moderado. además de esto también pudo identificar los caracteres de las placas vehiculares en varios casos de prueba. Sin embargo, se pudieron identificar limitaciones que están relacionadas con sombras, movimientos rápidos y ángulos de captura que son poco favorables, afectado así la precisión de lectura. De esta manera en base a los resultados obtenidos se observa que el prototipo tiene gran potencial para poder apoyar procesos de monitoreo vial, aun así es necesario realizar mejoras para escenarios más complejos o condiciones climáticas difíciles.

Rendimiento de procesamiento local

Durante el desarrollo de las pruebas se observó que uno de los aspectos más relevantes fue la capacidad que tiene el sistema para poder ejecutarse mediante la utilización de procesamiento local utilizando la GPU del hardware, permitiendo mantener un rendimiento fluido sin tener que depender en su totalidad de servicios externos o procesamiento en la nube, dicho esto cabe mencionar que el comportamiento del hardware se mantuvo estable la mayor cantidad del tiempo durante la realización de las pruebas funcionales, todo esto pudo ser observado mediante la interfaz desarrollada que mostro los eventos detectados en tiempo real.

Evaluación de la privacidad.

En el transcurso de desarrollo del prototipo fue priorizado el procesamiento local de la información, esto con el fin de lograr una reducción de almacenamiento de datos sensibles. Por esto el sistema fue diseñado para poder trabajar principalmente con metadatos que son obtenidos a través de las detecciones realizadas.

predictiva mediante Deep Learning y computación perimetral

El enfoque implementado permite evidenciar una ventaja frente a los modelos tradicionales los cuales dependen de un almacenamiento constante de video. De igual manera, este uso de procesamiento local contribuye a una disminución de riesgos que están relacionados con la privacidad y el tratamiento de la información personal.

Propuesta de solución**Situación actual.**

La movilidad urbana en las grandes ciudades esta enfrentada a diferentes problemáticas que tienen relación con congestión vehicular, tiempos largos de respuesta ante incidentes ocurridos y además de esto gran nivel de limitaciones en lo que se refiere al monitoreo del tráfico.

Lo anteriormente mencionado se debe principalmente a que muchos sistemas de monitoreo tradicionales dependen en gran medida de una supervisión manual o almacenamiento masivo de videos, pudiendo generar así altos costos operativos y enfrentarse a dificultades en el tratamiento de datos personales.

Oportunidades identificadas.

En la actualidad con el avance de la tecnología como por ejemplo la visión artificial y el procesamiento local, se ha permitido el desarrollo de soluciones capaces de poder detectar diferentes eventos en tiempo real sin tener la necesidad de depender por completo de servicios externos. además, estas tecnologías logran ofrecer oportunidades para poder optimizar procesos y monitoreo mejorando la gestión de la información.

predictiva mediante Deep Learning y computación perimetral**Propuesta para solución de la problemática.**

En respuesta a la problemática que fue identificada, se propone el desarrollo de **Urban Eye**, un prototipo orientado a monitorear de manera inteligente entornos urbanos mediante visión artificial. Este sistema integra funciones como detección de vehículos, lectura básica de las placas y visualización geográfica de los eventos mediante procesamiento local. Para la realización de estas funcionalidades el prototipo busca minimizar el almacenamiento innecesario de los datos sensibles mediante la utilización de metadatos obtenidos durante el análisis de video.

Beneficios esperados.

Con el desarrollo de este prototipo se espera la obtención de beneficios como lo son el apoyo al monitoreo de movilidad urbana, una automatización parcial de procesos de vigilancia, así como reducción de dependencia de procesamiento en la nube y un posible fortalecimiento de la privacidad mediante procesamiento local.

Limitaciones y trabajo futuro.

Aunque el prototipo presente unos resultados funcionales en las pruebas realizadas, todavía se cuenta con presencia de unos aspectos que requieren mejoras entre ellos se encuentran los siguientes:

- Optimización del módulo de procesamiento especialmente en escenarios de baja luminosidad
- Mejoras en la precisión para detectar objetos en movimiento rápido
- Mejorar la capacidad de procesamiento para escenarios con gran flujo vehicular
- Integración futura a una base de datos para almacenamiento de la información

Estas limitaciones mencionadas anteriormente representan oportunidades de mejora para las futuras etapas del proyecto.

Discusión

Los resultados obtenidos durante la realización de las pruebas funcionales del Prototipo Urban Eye permiten evidenciar que la integración de los modelos de visión artificial con procesamiento local representa una alternativa viable para el apoyo a procesos de monitoreo vehicular en las grandes ciudades. A partir de estas pruebas que fueron realizadas mediante videos de tráfico urbano en la ciudad de Bogotá, se pudo observar que el sistema logró poder identificar diferentes tipos de vehículos de manera simultánea manteniendo un comportamiento estable en el escenario evaluado.

De igual manera, los resultados obtenidos dan evidencia de que el prototipo cuenta con una gran capacidad de ejecutar las tareas básicas de reconocimiento vehicular y lectura de placas bajo condiciones adecuadas. Aunque la lectura de las placas si tuvo presencia de algunas limitaciones en escenarios específicos, especialmente en aquellos que cuentan con presencia de sombras, o ángulos poco favorables. Lo que se encontró permite considerar que la implementación desarrollada constituye una base con un potencial de mejora para las futuras etapas del proyecto

Los hallazgos que fueron encontrados guardan una relación con la problemática que fue planteada inicialmente en la investigación, en la cual se identificó la necesidad de implementar herramientas tecnológicas que permitieran optimizar el monitoreo de la movilidad urbana disminuyendo de esta manera la dependencia de procesos manuales o sistemas tradicionales que estuvieran basados únicamente en almacenamiento masivo de video. Por lo mencionado anteriormente, Urban Eye demuestra que el uso de la visión artificial puede facilitar en gran medida la detección de eventos vehiculares en tiempo real mejorando la capacidad de supervisión en entornos urbanos.

predictiva mediante Deep Learning y computación perimetral

De igual manera, los resultados permiten identificar que uno de los aspectos mas relevantes del prototipo corresponde al uso de procesamiento local mediante la utilización de un hardware que cuenta con GPU. Durante la realización de las pruebas el procesamiento obtuvo un comportamiento estable sin depender de servicios externos. A partir de estos hallazgos se comprueba que el procesamiento local no aporta únicamente fluidez, sino que también representa una alternativa que contribuye a la reducción de costos operativos y la dependencia de factores externos.

Por otra parte los resultados que fueron obtenidos también dan a conocer la existencia de diferentes limitaciones que necesitan ser consideradas para futuras mejoras del proyecto, se identificaron limitaciones como la baja iluminación, las condiciones climáticas desfavorables y la velocidad con la que se desplazan algunos vehículos estos factores tienen afectaciones directas sobre la precisión de detección y la lectura de las placas, es en base a estas limitaciones que se observa que aunque el prototipo es funcional, aun necesita grandes mejoras y optimización para escenarios más complejos o situaciones de tráfico a las que están sometidas las grandes ciudades.

Asimismo, mediante el desarrollo del proyecto se pudieron identificar oportunidades de crecimiento que están relacionadas con la integración de futuros módulos que sean requeridos, algunos de ellos son sistemas de almacenamiento estructurado de la información, modelos de análisis predictivo o herramientas avanzadas para el reconocimiento de los vehículos, de esta manera se considera que Urban Eye puede evolucionar a lo que es una solución más robusta que está orientada al contexto de las ciudades inteligentes.

predictiva mediante Deep Learning y computación perimetral

Además de esto en base a los resultados obtenidos se observa que estos resultados coinciden con las tendencias tecnológicas que fueron identificadas en el estado del arte del presente documento, en el cual diferentes estudios destacan el potencial que tiene la visión artificial aplicada al monitoreo urbano y una gestión inteligente del tráfico. Aun así, a diferencia de algunos enfoques que dependen de una infraestructura centralizada, Urban Eye prioriza la ejecución de los procesos de manera local, teniendo como objetivo ofrecer una solución más flexible que esta enfocada en la privacidad de la información de las personas.

Finalmente, a partir de los hallazgos que se pudieron obtener durante el desarrollo del proyecto, se puede dar la afirmación de que Urban Eye pudo cumplir en gran medida con el propósito de la investigación, esto se debe a que pudo demostrar que tan viable es implementar un sistema basado en visión artificial para poder apoyar diferentes procesos de monitoreo vehicular en tiempo real. No obstante, se debe mencionar que aun cuenta con distintas limitaciones técnicas y aspectos que deben ser mejorados, aun así, los resultados demuestran el gran potencial del prototipo como una herramienta tecnológica que esta orientada a fortalecer soluciones tecnológicas para la movilidad urbana.

Plan de divulgación

El presente proyecto de investigación titulado **“Urban Eye: Transformación del monitoreo pasivo en analítica predictiva mediante Deep Learning y computación perimetral”** plantea una estrategia de divulgación que está orientada a proyectar los resultados obtenidos más allá de un entorno académico, debido a que busca generar un impacto en escenarios, tecnológicos y de movilidad urbana. Esto se debe a que la investigación aborda problemáticas relacionadas con inteligencia artificial, visión computacional y movilidad urbana. Por lo mencionado anteriormente se considera importante establecer unos mecanismos de comunicación que puedan permitir la socialización de aportes desarrollados y así facilitar la transferencia del conocimiento.

Por otra parte, se proyecta como principal producto de divulgación la elaboración de un artículo científico enfocado en la aplicación de modelos de visión artificial y arquitecturas Edge Computing para la detección de anomalías viales en entornos urbanos. Este canal de divulgación es el más adecuado ya que el proyecto cuenta con metodologías experimentales, desarrollo tecnológico y análisis cuantitativo, estos elementos mencionados anteriormente son relevantes para las áreas de investigación que están relacionadas con inteligencia artificial, movilidad urbana y sistemas de monitoreo urbanos. De igual manera el artículo da la posibilidad de poder presentar los resultados obtenidos en lo que se refiere a precisión de detección, reducción de latencia y eficiencia computacional.

predictiva mediante Deep Learning y computación perimetral

Adicional a esto, se plantea la construcción de un prototipo funcional que estará basado en modelos YOLO que serán implementados en dispositivos Edge Computing, la realización de este prototipo permitirá realizar demostraciones técnicas en los escenarios tecnológicos, facilitando de esta manera la visualización practica del funcionamiento del prototipo de Urban Eye. Se tiene como propósito generar visibilidad sobre las capacidades de este sistema y así poder promover la realización de futuras alianzas con entidades que estén interesadas en soluciones de movilidad urbana inteligente.

Por otra parte, se contempla la posibilidad de estructurar una propuesta que está orientada a programas de innovación y emprendimiento como lo es EAN impacta, debido principalmente a que Urban Eye posee características que están asociadas al desarrollo de soluciones tecnológicas con un gran potencial para ser aplicada en contextos reales, es por lo mencionado anteriormente que se considera la idea de que el sistema podría evolucionar para convertirse en una herramienta de apoyo para las entidades de movilidad y mantenimiento urbano.

Además de esto en cuanto a los recursos requeridos, será necesario poder contar con el acompañamiento institucional por parte de grupos de investigación y docentes que estén relacionados con desarrollo de software y analítica de datos, también se requiere acceso a herramientas de procesamiento computacional, dispositivos Edge Computing y unos espacios académicos para poder realizar pruebas piloto y socialización de los resultados obtenidos.

predictiva mediante Deep Learning y computación perimetral

Por otra parte, en el apartado de posibles aliados externos se contemplan posibles grupos de investigación, empresas tecnológicas que estén interesadas en soluciones basadas en Inteligencia Artificial. De esta manera estas posibles alianzas podrían llegar a fortalecer el impacto del proyecto a un nivel institucional o social.

Con respecto al cronograma estimado para el proceso de divulgación, este está dividido en Tres etapas principales, es así como en la primera etapa se realizará la organización y el análisis de los resultados experimentales obtenidos a lo largo del desarrollo del sistema. Posterior a esto, se llevará a cabo la elaboración del artículo científico junto con la preparación del material técnico. En la tercera etapa se realizará la postulación del proyecto.

Finalmente, como resultados esperados del plan de divulgación se busca poder fortalecer la visibilidad académica del proyecto, promoviendo la transferencia del conocimiento hacia contextos aplicados contribuyendo de manera positiva al desarrollo de investigaciones que estén relacionadas con movilidad urbana inteligente y visión artificial. De igual manera, se espera que el proyecto Urban Eye pueda llegar a ser consolidado como una propuesta tecnológica con un potencial para convertirse en una aplicación para procesos de monitoreo vial, que pueda contribuir a una mejora en la capacidad de respuesta frente a diferentes anomalías urbanas fomentando el uso de Inteligencia artificial en diferentes escenarios de monitoreo de las grandes ciudades.

Conclusiones

El desarrollo de Urban Eye permitió demostrar que la implementación de modelos de visión artificial y la arquitectura de procesamiento local representa una alternativa viable para poder apoyar procesos de monitoreo urbano en tiempo real. A partir de las diferentes pruebas realizadas, se evidencio que el sistema identificó los diferentes tipos de vehículos y pudo ejecutar tareas básicas de reconocimiento vehicular logrando tener una estabilidad en distintos escenarios de prueba. Lo mencionado anteriormente permite concluir que la implementación de tecnologías que están basadas en Edge Computing y Deep Learning contribuye de manera positiva al fortalecimiento de soluciones orientadas a una movilidad inteligente.

Además de esto con respecto al objetivo principal de la investigación, se logró desarrollar un prototipo totalmente funcional mediante la utilización de visión artificial, en donde tiene gran capacidad para detectar diferentes eventos vehiculares mediante el uso del algoritmo YOLO ejecutado de manera local. De esta manera según los resultados obtenidos se pudo identificar que el prototipo presenta un buen desempeño en condiciones adecuadas demostrando así su potencial para apoyar procesos de vigilancia.

por otra parte, se puede evidenciar la importancia del uso de técnicas de procesamiento local desde una perspectiva relacionada con la privacidad de la información. Urban Eye fue diseñado priorizando el tratamiento de metadatos, disminuyendo de esta manera posibles riesgos que están asociados al manejo de datos sensibles. Desde esta perspectiva, la incorporación de este principio de Privacidad por Diseño conforma un aporte importante dentro del desarrollo de este sistema, equilibrando así las necesidades de monitoreo con la protección de datos de los ciudadanos

predictiva mediante Deep Learning y computación perimetral

Referencias bibliográficas

- Batty, M. (2013). *The New Science of Cities*. MIT Press.
- Bettencourt, L. M. A. (2021). *Introduction to Urban Science: Evidence and Theory of Cities as Complex Systems*. MIT Press.
- Cavoukian, A. (2011). *Privacy by Design: The 7 Foundational Principles*. Information and Privacy Commissioner of Ontario.
- Giffinger, R., Fertner, C., Kramar, H., Kalasek, R., Pichler-Milanović, N., & Meijers, E. (2007). *Smart cities: Ranking of European medium-sized cities*. Vienna University of Technology.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- Han, S., Pool, J., Tran, J., & Dally, W. (2015). Learning both Weights and Connections for Efficient Neural Network. *Advances in Neural Information Processing Systems (NIPS)*, 28, 1135–1143.
- Hartley, R., & Zisserman, A. (2004). *Multiple View Geometry in Computer Vision* (2^a ed.). Cambridge University Press.
- Hernández-Sampieri, R., & Mendoza, C. (2018). *Metodología de la investigación: Las rutas cuantitativa, cualitativa y mixta*. McGraw-Hill.
- Lee, K., & Williams, P. (2026). Federated Learning in Urban Edge Intelligence: Privacy and Efficiency. *International Journal of Smart Cities*, 8(1), 45–62.
- Mell, P., & Grance, T. (2011). *The NIST Definition of Cloud Computing*. National Institute of Standards and Technology. Special Publication 800-145. U.S. Department of Commerce.
- Moher, D., Liberati, A., Tetzlaff, J., & Altman, D. G. (2009). Preferred reporting items for systematic reviews and meta-analyses: The PRISMA statement. *PLoS Medicine*, 6(7), e1000097. <https://doi.org/10.1371/journal.pmed.1000097>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779–788.
- Roche, S., Nabian, N., Kloeckl, K., & Ratti, C. (2012). Are 'Smart Cities' Smart Enough? *Global Geospatial Conference*, 1, 215–235.
- Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge Computing: Vision and Challenges. *IEEE Internet of Things Journal*, 3(5), 637–646.

predictiva mediante Deep Learning y computación perimetral

Smith, R., et al. (2023). Edge Computing for Road Safety: Evaluation of hardware performance. *Future Generation Computer Systems*, 145, 332–345.

Sommerville, I. (2011). *Software Engineering*. Addison-Wesley...

Wan, F., Sun, C., He, H., Lei, G., Xu, L., & Xiao, T. (2022). YOLO-LRDD: a lightweight method for road damage detection based on improved YOLOv5s. *EURASIP Journal on Advances in Signal Processing*, 2022(1). <https://doi.org/10.1186/s13634-022-00931-x>

Guo, G., & Zhang, Z. (2022). Road damage detection algorithm for improved YOLOv5. *Scientific Reports*, 12(1), 15523. <https://doi.org/10.1038/s41598-022-19674-8>

Park, S., Tran, V., & Lee, D. (2021). Application of various YOLO models for computer Vision-Based Real-Time pothole detection. *Applied Sciences*, 11(23), 11229.

<https://doi.org/10.3390/app112311229>

Ramesh, A., Nikam, D., Balachandran, V. N., Guo, L., Wang, R., Hu, L., Comert, G., & Jia, Y. (2022). Cloud-Based Collaborative Road-Damage Monitoring with Deep Learning and Smartphones. *Sustainability*, 14(14), 8682. <https://doi.org/10.3390/su14148682>

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4), 427–437.

<https://doi.org/10.1016/j.ipm.2009.03.002>