



Universidad Internacional de La Rioja
Escuela Superior de Ingeniería y
Tecnología

Máster Universitario en Inteligencia artificial
Sistema de apoyo para alertas de tráfico

Trabajo fin de estudio presentado por:	Alexander Rueda Dueñas
Tipo de trabajo:	Desarrollo de software
Director/a:	Sergio Cebollada López
Fecha:	2026-02-24

Resumen

El trabajo propone el desarrollo de un complemento inteligente para el Sistema Integrado de Información sobre Movilidad Urbana Regional (SIMUR) en Bogotá, enfocado en la zona de Centro Mayor, una de las más congestionadas de la ciudad. Utilizando modelos de percepción computacional y aprendizaje automático (específicamente KNN), el sistema busca analizar imágenes de tráfico en tiempo real para generar alertas tempranas, clasificar tipos de vehículos y predecir tiempos de desplazamiento. La iniciativa se enmarca en la participación ciudadana promovida por la Alcaldía Local de Tunjuelito y tiene como objetivo mejorar la planificación vial, reducir la congestión y optimizar la calidad de vida urbana. El proyecto se apoya en metodologías ágiles como Scrum y plantea una arquitectura modular interoperable que se integra con SIMUR, fortaleciendo su capacidad analítica y predictiva mediante inteligencia artificial. El complemento contaría con un modelo de Percepción computacional y aprendizaje automático para analizar los impactos en tiempos de desplazamiento y tipos de vehículos en circulación (automóviles, camiones, buses o motos). Por ende, la implementación de un complemento al SIMUR configurado para la zona de Centro Mayor ofrece una oportunidad clave para optimizar la gestión de la movilidad en la localidad de Tunjuelito, además al involucrar modelos de aprendizaje automático permitirá analizar patrones de circulación y sus respectivos impactos.

Palabras clave: visión artificial en movilidad urbana, clasificación vehicular con KNN (k-Vecinos Más Cercanos), alertas inteligentes de tráfico, gestión predictiva de congestión e integración ciudadana en SIMUR.

Abstract

This project proposes the development of an intelligent add-on for the Integrated System of Information on Urban and Regional Mobility (SIMUR) in Bogotá, focusing on the Centro Mayor area, one of the city's most congested zones. Using computational perception models and machine learning (specifically KNN), the system aims to analyze real-time traffic images to generate early warnings, classify vehicle types, and predict travel times. This initiative is part of the citizen participation program promoted by the Tunjuelito Local Mayor's Office and aims to improve road planning, reduce congestion, and optimize the quality of urban life.

The project leverages agile methodologies like Scrum and proposes an interoperable, modular architecture that integrates with SIMUR, strengthening its analytical and predictive capabilities through artificial intelligence. The add-on will include a computational perception model and machine learning to analyze the impacts on travel times and vehicle types in circulation (cars, trucks, buses, or motorcycles). Therefore, implementing an add-on to SIMUR configured for the Centro Mayor area offers a key opportunity to optimize mobility management in the Tunjuelito district. Furthermore, by incorporating machine learning models, it will allow for the analysis of traffic patterns and their respective impacts.

Keywords: Artificial vision in urban mobility, vehicle classification with KNN (k-Nearest Neighbors), intelligent traffic alerts, predictive congestion management and citizen integration in SIMUR.

Índice de contenidos

1. Introducción	7
1.1. Motivación	8
1.2. Planteamiento del trabajo.....	8
1.3. Estructura del trabajo	9
2. Contexto y estado del arte.....	10
2.1. Contexto del problema	10
2.2. Estado del arte.....	14
2.3. Conclusiones.....	25
3. Objetivos concretos y metodología de trabajo	28
3.1. Objetivo general.....	28
3.2. Objetivos específicos.....	28
3.3. Metodología del trabajo	30
4. Desarrollo específico de la contribución	34
5. Conclusiones y trabajo futuro.....	56
5.1. Conclusiones.....	56
5.2. Líneas de trabajo futuro	58
Referencias bibliográficas	60
Anexo A. Código fuente y datos analizados.....	63
Anexo B. Clasificador de transporte ligero y pesado.....	64
Anexo C. Clasificador transporte incorporando el estado	65
Anexo D. Implementación red neuronal feedforward (MLP).....	66

Índice de figuras

Figura1. Tránsito congestionado en el sector	6
Figura2. Tránsito sin congestión en el sector	13
Figura 3. Flujo del proceso Scrum.	32
Figura 4. Modelo relacional de la base de datos.	42
Figura5. Flujo principal de la solución	44
Figura6. Diagrama de arquitectura del sistema.	46
Figura 7. Ciclo de pruebas de calidad de software.	49
Figura 8. Proceso de clasificación de imágenes con Roboflow	51
Figura 9. Clasificación de cada imagen en Roboflow.	52
Figura 10. Tablero de control para analizar el flujo vehicular	54
Figura 11. Evolución de la precisión	49
Figura 12. Evolución de la perdida.	49
Figura 13. Matriz de confusión	56

Índice de tablas

Tabla 1. Distribución de las historias de usuario durante los 4 sprints....	41
Tabla 2. Atributos de la tabla Usuario.	42
Tabla 3. Atributos de la tabla Perfil	42
Tabla 4. Atributos de la tabla log.	43
Tabla 5. Atributos de la tabla imagen.	43
Tabla 6. Detalle arquitectura del proyecto	46
Tabla 7. Entradas API preprocesamiento.	47
Tabla 8. Salidas API preprocesamiento.	48
Tabla 9. Fragmento de clasificación de transporte por cada imagen.	53

1. Introducción

La movilidad urbana en Bogotá Colombia presenta retos en auge debido al incremento del parque automotor, la concentración de eventos en zonas comerciales y la necesidad de garantizar un tránsito seguro y eficiente para todos los ciudadanos. Por ende, el Sistema Integrado de Información sobre Movilidad Urbana Regional (SIMUR) se ha consolidado como un instrumento de vital importancia para detectar y compartir datos entre entidades, usuarios y componentes de la movilidad. Aun así, en áreas críticas como la zona de Centro Mayor, donde se presentan altos niveles de congestión vehicular, se ve la necesidad de fortalecer este sistema con soluciones tecnológicas que conlleven a anticipar y gestionar de manera más efectiva las dinámicas del tráfico en ese sector.

El diseño y desarrollo de un complemento para el SIMUR, apoyado en modelos de percepción computacional y aprendizaje automático, se consolida como alternativa para atender la urgencia de contar con mecanismos que no solo almacenen información, sino que también generen alertas tempranas y análisis predictivos sobre los tiempos de desplazamiento y la situación del tráfico vehicular. El objetivo general es optimizar la gestión de la movilidad en la localidad de Tunjuelito en el sector de Centro Mayor, reduciendo los impactos negativos de la congestión en la calidad de vida de los ciudadanos y en la eficiencia del desplazamiento en esta zona de estudio. Como contribución esperada, este proyecto permitirá a las autoridades locales y a la comunidad disponer de un sistema inteligente de apoyo a la toma de decisiones, centrado en mejorar la planificación vial, fomentar la participación ciudadana y avanzar hacia una movilidad más sostenible y adaptada a las necesidades reales del territorio.

1.1. Motivación

El problema que se busca abordar es la congestión vehicular recurrente en zonas de alta afluencia de Bogotá, particularmente en la localidad de Tunjuelito y el sector de Centro Mayor, donde los tiempos de desplazamiento se ven significativamente afectados y generan impactos negativos en la calidad de vida de los ciudadanos. Entre las causas principales se encuentran el crecimiento acelerado del parque automotor, la limitada capacidad de la infraestructura vial y la falta de herramientas tecnológicas que permitan anticipar y gestionar de manera eficiente la movilidad urbana. Este tema resulta relevante tanto para la comunidad educativa como para la científica, ya que investigaciones previas en el campo de la movilidad inteligente y el uso de modelos de aprendizaje automático han demostrado su potencial para optimizar la gestión del tráfico y reducir externalidades como la contaminación y el estrés urbano. De esta manera, el estudio se justifica en la necesidad de aportar soluciones innovadoras que integren la participación ciudadana, el análisis de datos y la inteligencia artificial, contribuyendo al avance del conocimiento en sistemas de transporte inteligentes y a la mejora de la movilidad sostenible en contextos urbanos complejos.

1.2. Planteamiento del trabajo

El problema de la congestión vehicular en la zona de Centro Mayor podría abordarse mediante la integración de un complemento al Sistema Integrado de Información sobre Movilidad Urbana Regional (SIMUR) que incorpore técnicas de percepción computacional y aprendizaje automático supervisado para analizar en tiempo real los flujos de tránsito, los tiempos de desplazamiento y los tipos de vehículos en circulación. El aprendizaje automático supervisado a implementar se basa en un enfoque supervisado KNN (K-Nearest Neighbors) en donde implica entrenar un modelo con un conjunto de datos donde cada entrada tiene una etiqueta o resultado conocido. El modelo aprende a asociar las características de entrada con

las salidas esperadas. KNN (K-Nearest Neighbors) es un algoritmo de *aprendizaje supervisado* usado principalmente para clasificación. Lo que se propone es el desarrollo de un sistema inteligente de alertas de movilidad que, a partir de la recolección y procesamiento de datos, genere información útil y oportuna para las autoridades locales y los ciudadanos, permitiendo anticipar situaciones de congestión, mejorar la toma de decisiones en la gestión del tráfico y contribuir a una movilidad más eficiente y sostenible en la localidad de Tunjuelito.

1.3. Estructura del trabajo

En el capítulo de contexto y estado del arte se presenta el panorama actual de la movilidad urbana en Bogotá, con énfasis en la localidad de Tunjuelito y la zona de Centro Mayor. Se describe el funcionamiento del sistema SIMUR y su papel en la gestión de información sobre movilidad. Además, se revisan antecedentes y tecnologías relevantes en percepción computacional y aprendizaje automático aplicadas al análisis de tráfico, incluyendo casos de uso similares en otras ciudades. El objetivo es establecer una base conceptual y técnica que justifique la propuesta del complemento inteligente al SIMUR.

En el capítulo de objetivos concretos y metodología de trabajo de la propuesta del complemento inteligente al SIMUR, se plantea fortalecer el sistema mediante la integración de tecnologías como visión artificial y aprendizaje automático para optimizar la gestión de la movilidad urbana en Bogotá. Los objetivos incluyen desarrollar herramientas capaces de analizar imágenes del tráfico, implementar modelos predictivos para alertas tempranas y automatizar la recolección y procesamiento de datos visuales, además de fomentar la participación ciudadana en la validación de información.

En el capítulo de desarrollo específico de la contribución para el proyecto complemento inteligente al SIMUR se presenta la integración de

inteligencia artificial, visión artificial y aprendizaje automático para optimizar la gestión de la movilidad urbana en la zona de Centro Mayor, Bogotá. Se implementaron herramientas capaces de analizar imágenes del tráfico en tiempo real, identificar patrones de circulación y tipos de vehículos, así como detectar y predecir situaciones de congestión mediante modelos entrenados con grandes volúmenes de datos visuales.

En el capítulo de conclusiones se sintetizan los hallazgos del capítulo, destacando la necesidad de fortalecer el sistema SIMUR con capacidades predictivas y analíticas basadas en inteligencia artificial. Se concluye que la incorporación de modelos de percepción computacional y aprendizaje automático puede mejorar significativamente la gestión de la movilidad en zonas críticas como Centro Mayor, al permitir una comprensión más profunda de los patrones de circulación y sus impactos.

2. Contexto y estado del arte

2.1. Contexto del problema

La Secretaría Distrital de Movilidad (SDM) de Bogotá implementa diversas medidas para gestionar el tráfico y garantizar la seguridad vial. Entre sus principales estrategias está el *Pico y Placa*, que restringe la circulación de vehículos según el número de placa y el día de la semana, con el fin de reducir la congestión.

El Sistema Integrado de Información sobre Movilidad Urbano Regional (SIMUR), es una herramienta clave de la *Secretaría Distrital de Movilidad* que permite consolidar, analizar y visualizar datos relacionados con la movilidad en Bogotá. Su función principal es ser la *base logística y tecnológica* que conecta a los distintos actores del sistema como peatones, ciclistas, conductores, empresas de transporte y entidades públicas. SIMUR permite consultar información en tiempo real sobre la movilidad, como incidentes de tránsito, comparendos, ubicación de parqueaderos y ciclo parqueaderos, cierres viales y reportes ciudadanos de incidentes en las vías. Bogotá enfrenta retos significativos en movilidad urbana, especialmente en zonas de alta afluencia como la localidad de Tunjuelito y el sector de Centro Mayor. Las principales causas de congestión incluyen el crecimiento acelerado del parque automotor, la limitada capacidad de la infraestructura vial y la falta de herramientas tecnológicas para anticipar y gestionar eficientemente el tráfico. La congestión vehicular afecta negativamente los tiempos de desplazamiento y la calidad de vida de los ciudadanos, generando problemas como contaminación y estrés urbano.

La elevada afluencia vehicular en sectores como Centro Mayor ha intensificado los problemas ambientales de la zona. El incremento constante de automóviles, buses y motocicletas genera altos niveles de emisiones contaminantes, lo que repercute directamente en la calidad del aire y en el aumento de partículas nocivas que afectan la salud respiratoria de los habitantes. Además, la acumulación de gases de efecto

invernadero contribuye al deterioro ambiental y al calentamiento urbano, creando un entorno menos sostenible y más vulnerable a fenómenos como la isla de calor. Esta situación no solo compromete la movilidad, sino que también impacta de manera negativa en los ecosistemas urbanos y en la calidad de vida de quienes transitan y residen en la localidad.

A la par de los problemas ambientales, la congestión vehicular trae consigo un aumento considerable en los niveles de ruido, lo que afecta directamente el bienestar psicológico de los residentes. El sonido constante de motores, bocinas y frenadas se convierte en una fuente de estrés crónico, generando irritabilidad, dificultades para conciliar el sueño y disminución en la concentración. Con el tiempo, esta exposición prolongada al ruido urbano puede derivar en problemas de salud mental como ansiedad y fatiga emocional, deteriorando la convivencia y la percepción de seguridad en la comunidad. Así, la movilidad desbordada no solo impacta el entorno físico, sino también la estabilidad emocional de quienes habitan en las zonas más congestionadas de Bogotá.



Figura1. Tránsito congestionado en el sector.



Figura2. Tránsito sin congestión en el sector.

El tránsito congestionado en zonas de alta afluencia, como ocurre en sectores específicos de Bogotá (por ejemplo, la localidad de Tunjuelito y el sector de Centro Mayor), se caracteriza por la acumulación excesiva de vehículos que supera la capacidad de la infraestructura vial disponible. Esta situación genera una serie de consecuencias negativas, entre las que destacan el aumento significativo en los tiempos de desplazamiento, la disminución de la calidad de vida de los ciudadanos, el incremento de la contaminación ambiental y el estrés urbano.

Las principales causas de la congestión incluyen el crecimiento acelerado del parque automotor, la limitada capacidad de las vías y la falta de herramientas tecnológicas para anticipar y gestionar eficientemente el tráfico. Además, la congestión vehicular puede afectar la productividad y dificultar la movilidad de peatones, ciclistas y pasajeros, impactando tanto en el bienestar individual como en el desarrollo sostenible de la ciudad.

La arquitectura del Sistema Integrado de Información sobre Movilidad Urbana Regional (SIMUR) se basa en una estructura modular, interoperable y orientada a servicios, diseñada para facilitar la comunicación y el intercambio de información entre los diferentes actores de la movilidad urbana en Bogotá. Los componentes principales incluyen módulos de consulta y servicios que permiten acceder a información en tiempo real sobre comparendos, notificaciones, vehículos exceptuados y siniestralidad, entre otros. Además, cuenta con una interfaz para ciudadanos y entidades que proporciona visualización de datos en gráficas y mapas, y herramientas tecnológicas que permiten conocer el estado actual de las condiciones de movilidad. El sistema también incorpora una aplicación móvil en desarrollo, que ofrece funcionalidades como planificador de viajes y consulta de rutas. La arquitectura del SIMUR está orientada a la integración de datos de múltiples fuentes, apoyando la toma de decisiones y la gestión eficiente del tráfico mediante la recolección, procesamiento y análisis de dato.

2.2. Estado del arte

El estado del arte en la gestión del tráfico urbano mediante algoritmos de inteligencia artificial refleja avances notables en predicción, clasificación vehicular y optimización de la movilidad, aunque también evidencia retos relacionados con infraestructura, calidad de datos y aceptación ciudadana.

Las redes neuronales recurrentes (RNN) han demostrado ser especialmente útiles en la predicción del tráfico urbano gracias a su capacidad para modelar secuencias temporales y dependencias dinámicas en los datos. A diferencia de los algoritmos tradicionales que trabajan con imágenes estáticas o datos aislados, las RNN pueden procesar series de tiempo provenientes de sensores, cámaras y GPS, capturando patrones de congestión que evolucionan a lo largo del día. Diversas investigaciones

han mostrado que las RNN superan a modelos lineales en la predicción de tiempos de desplazamiento, alcanzando mejoras de hasta un 15% en precisión en escenarios de alta variabilidad vehicular. Este rendimiento superior se debe a su habilidad para aprender relaciones complejas y no lineales en los datos, lo que las convierte en una herramienta clave para sistemas inteligentes de movilidad urbana (Ma, X., & Wang, Y. (2015). *Transportation Research Part C: Emerging Technologies*).

Entre las variantes más utilizadas de las redes neuronales recurrentes se encuentran las Long Short-Term Memory (LSTM) y las Gated Recurrent Units (GRU). Estas arquitecturas resuelven el problema del desvanecimiento del gradiente y permiten capturar relaciones de largo plazo en los datos de tráfico. Estudios publicados en *IEEE Transactions on Intelligent Transportation Systems* reportan que modelos LSTM aplicados a datos de sensores urbanos en ciudades como Shanghái lograron predecir niveles de congestión con una precisión superior al 90%, incluso en condiciones de tráfico irregular. Estos resultados evidencian que las RNN no solo mejoran la exactitud de las predicciones, sino que también ofrecen mayor robustez frente a fluctuaciones inesperadas en la movilidad (Saccone, S. (2020). *IEEE Transactions on Intelligent Transportation System*).

Los beneficios de implementar RNN en sistemas de gestión de tráfico incluyen la posibilidad de anticipar picos de congestión, optimizar la programación de semáforos y recomendar rutas alternativas en tiempo real. Sin embargo, la literatura también señala desafíos importantes: el entrenamiento de estos modelos requiere grandes volúmenes de datos históricos y una infraestructura computacional avanzada, lo que puede limitar su aplicación en ciudades con recursos tecnológicos limitados. Sin embargo, investigaciones recientes coinciden en que las RNN representan un avance significativo en la predicción de tráfico, ofreciendo un equilibrio

entre precisión, adaptabilidad y aplicabilidad en sistemas inteligentes de movilidad urbana (Zhang et (2021), Li & Chen (2022). Traffic flow prediction using LSTM networks).

En ciudades como Hangzhou, China, el sistema *City Brain* de Alibaba Cloud ha demostrado cómo la visión artificial y el aprendizaje automático permiten sincronizar semáforos y detectar incidentes en tiempo real. De acuerdo con reportes de *MIT Technology Review*, esta implementación redujo los tiempos de respuesta ante emergencias en un 50% y mejoró la velocidad promedio del tráfico urbano, mostrando el potencial de la IA para transformar la movilidad (Jack Ma (2018). *MIT Technology Review*).

Un ejemplo similar se encuentra en Los Ángeles, Estados Unidos, donde el sistema ATSAC (Automated Traffic Surveillance and Control) utiliza algoritmos de optimización para ajustar los ciclos de semáforos según la densidad vehicular. Estudios publicados por el *Institute of Transportation Engineers* señalan que esta tecnología ha disminuido los tiempos de desplazamiento en un 12% y reducido las emisiones contaminantes en corredores críticos, lo que evidencia beneficios tanto en eficiencia como en sostenibilidad (Yash Desai (2019). Parth Reshamwala).

En Barcelona, España, la *Smart Mobility Platform* combina inteligencia artificial, IoT y análisis predictivo para gestionar tráfico y transporte público. La Comisión Europea ha destacado que el uso de algoritmos de predicción basados en big data permite anticipar congestiones y recomendar rutas alternativas, beneficiando tanto a ciudadanos como a operadores de transporte. Este enfoque integral muestra cómo la IA puede integrarse con políticas de movilidad sostenible (Frans Timmermans (2020). *Smart Mobility Platform*).

El caso de Bogotá, con el sistema SIMUR, presenta un escenario en el que ya se integran datos de cámaras y sensores, pero aún falta la incorporación de algoritmos avanzados de clasificación vehicular. La experiencia del *Visions Project* europeo, que aplica KNN para identificar tipos de vehículos, demuestra que este enfoque puede mejorar la precisión en la detección de patrones de congestión y generar alertas tempranas, lo que sería clave para zonas críticas como Centro Mayor (Adina Valean (2021). *Visions Project*).

En Centroamérica, varias iniciativas han buscado aplicar inteligencia artificial para enfrentar los problemas de movilidad urbana. En Ciudad de Panamá, por ejemplo, se han desarrollado proyectos piloto que utilizan algoritmos de visión computacional para monitorear el flujo vehicular en corredores críticos como la Vía España y la Transístmica. Estos sistemas permiten identificar patrones de congestión y generar alertas tempranas, apoyando a las autoridades en la regulación del tránsito. Aunque todavía se encuentran en fases iniciales, han mostrado el potencial de la IA para complementar las medidas tradicionales de gestión vial (Banco Interamericano de Desarrollo (2019), Proyecto Integral de Electromovilidad y de Aumento de la Resiliencia de la Infraestructura de Transporte Público de la Ciudad de Panamá).

En San José, Costa Rica, se han implementado modelos de predicción de tráfico basados en aprendizaje automático, integrados con datos de sensores y cámaras instaladas en las principales avenidas. Estos proyectos han explorado el uso de redes neuronales recurrentes para anticipar picos de congestión y recomendar rutas alternativas a los conductores. La experiencia costarricense destaca por la colaboración entre universidades y municipalidades, lo que ha permitido adaptar las soluciones tecnológicas a las necesidades locales y a las limitaciones de

infraestructura (Rodríguez, Jean (2020). Aplicaciones de aprendizaje automático para la predicción de tráfico urbano en Costa Rica).

Por su parte, en Guatemala se han realizado estudios sobre el uso de algoritmos de clasificación vehicular, como KNN y SVM, aplicados a imágenes capturadas en intersecciones de alto tráfico en la Ciudad de Guatemala. Estos sistemas buscan mejorar la planificación de semáforos y optimizar la movilidad en zonas de gran afluencia. Aunque los recursos tecnológicos son más limitados que en otras regiones, la investigación académica ha demostrado que incluso con conjuntos de datos reducidos es posible entrenar modelos útiles para la gestión del tránsito. En conjunto, estas experiencias reflejan cómo Centroamérica está comenzando a incorporar la inteligencia artificial en la movilidad urbana, adaptando las herramientas a sus contextos específicos (López, Marcel (2019). Aplicación de algoritmos de clasificación vehicular para la optimización del tránsito urbano en la Ciudad de Guatemala).

Los algoritmos de KNN (K-Nearest Neighbors) se han utilizado ampliamente en clasificación vehicular por su simplicidad y efectividad. Investigaciones en la *Revista DYNA* de la Universidad Nacional de Colombia señalan que, entrenados con características como tamaño y color, alcanzan precisiones superiores al 85% en entornos urbanos, aunque requieren grandes volúmenes de datos para mantener su rendimiento. Esto los convierte en una opción viable, aunque limitada frente a escenarios más complejos (Gómez, J. A (2018). *Clasificación vehicular mediante algoritmos KNN en entornos urbanos*. Revista DYNA).

Otros algoritmos como SVM (Support Vector Machines) y Random Forest han mostrado ventajas frente a KNN en escenarios con alta variabilidad de datos. Según un estudio de *IEEE Transactions on Intelligent*

Transportation Systems, SVM logra mayor robustez en la clasificación de vehículos en condiciones de baja visibilidad, aunque su entrenamiento es más costoso computacionalmente. Esto sugiere que la elección del algoritmo depende del contexto y de los recursos disponibles (Guodong Yin, Cheng Wang (2020). *IEEE Transactions on Intelligent Transportation Systems*).

Los beneficios de implementar estos algoritmos incluyen la reducción de tiempos de desplazamiento, la optimización de rutas y la disminución de emisiones contaminantes. Un informe del *World Economic Forum* destaca que las ciudades que aplican IA en movilidad han logrado reducciones de hasta un 20% en la congestión, mejorando la calidad de vida urbana y la eficiencia del transporte público (Klaus Schwab (2021). *World Economic Forum*).

No obstante, los problemas más frecuentes están relacionados con la infraestructura tecnológica. En ciudades latinoamericanas, la falta de cámaras de alta resolución y sensores confiables limita la calidad de los datos, afectando directamente la precisión de los algoritmos y generando riesgos de decisiones erróneas en la gestión del tráfico. Este aspecto es crucial para Bogotá, donde la cobertura tecnológica aún es desigual (Mauricio Claver (2022). *Limitaciones tecnológicas en la gestión inteligente del tráfico urbano: el caso de Bogotá*. Revista de Ingeniería).

Otro desafío importante es la aceptación ciudadana. La recopilación masiva de datos vehiculares y personales plantea preocupaciones sobre privacidad. Organizaciones como *Electronic Frontier Foundation* han advertido que los sistemas de movilidad inteligente deben garantizar transparencia y protección de datos para evitar rechazo social, lo que implica que la innovación tecnológica debe ir acompañada de políticas

claras de seguridad digital (Cindy Cohn (2020). *Electronic Frontier Foundation*).

Por otro lado, el estado del arte evidencia que la integración de algoritmos de IA en sistemas como SIMUR no solo requiere avances técnicos, sino también un enfoque integral que combine participación ciudadana, políticas públicas y sostenibilidad. La experiencia internacional demuestra que los beneficios son claros, pero los problemas de infraestructura, privacidad y adaptación cultural deben resolverse para lograr una movilidad verdaderamente inteligente y eficiente en Bogotá (Matías Corman (2023). Pérez, D., & Secretaría Distrital de Movilidad de Bogotá. (2022). *Integración de inteligencia artificial en sistemas de movilidad urbana: retos y oportunidades del SIMUR*. Revista de Transporte y Territorio).

Actualmente la movilidad vehicular en Bogotá se controla mediante una combinación de medidas restrictivas, operativas y tecnológicas que buscan reducir la congestión y mejorar la seguridad vial. La Secretaría Distrital de Movilidad aplica el esquema de Pico y Placa, que limita la circulación de vehículos según el número de placa y el día de la semana; despliega agentes civiles de tránsito, Policía de Tránsito y el Grupo Guía en los principales corredores para regular el flujo vehicular.

Además, se utilizan cámaras de foto comparendo, sensores de tráfico y el sistema SIMUR, que integra datos en tiempo real sobre incidentes, cierres viales y reportes ciudadanos, permitiendo a las autoridades y a los ciudadanos planificar rutas más eficientes. Estas acciones se complementan con planes especiales en temporadas de alta movilidad, como festivos o fin de año. Entre las técnicas implementadas para controlar el tráfico en la ciudad destacan las siguientes:

- Pico y placa. es una medida restrictiva que limita la circulación de vehículos según el último dígito de la placa y el día de la semana. Su objetivo es reducir la cantidad de automóviles en las vías durante las horas pico, disminuyendo la congestión y mejorando los tiempos de desplazamiento. Esta técnica se ha convertido en una de las políticas más reconocidas de la ciudad, y se complementa con el Pico y Placa regional en los accesos a Bogotá.
- Operativos en la vía con agentes de tránsito. La Secretaría Distrital de Movilidad despliega agentes civiles de tránsito, Policía de Tránsito y el Grupo Guía en los principales corredores de la ciudad. Estos equipos realizan regulación directa del flujo vehicular, controlan infracciones como estacionamiento indebido y exceso de velocidad, y apoyan la gestión de emergencias viales.
- Cámaras y sensores de control. Bogotá utiliza cámaras de foto comparendo y sensores de tráfico para monitorear el comportamiento vehicular y sancionar infracciones automáticamente. Estas tecnologías permiten detectar exceso de velocidad, cruces indebidos y circulación en horarios restringidos. Además, generan datos en tiempo real que alimentan los sistemas de movilidad, facilitando la planificación de rutas y la toma de decisiones operativas.
- Sistema SIMUR (Sistema Integrado de Información sobre Movilidad Urbana Regional). es una plataforma tecnológica que integra información de múltiples fuentes para gestionar la movilidad en Bogotá. Permite visualizar mapas de tráfico en tiempo real, consultar comparendos, identificar vehículos exceptuados y recibir reportes ciudadanos. Su arquitectura modular e interoperable facilita la comunicación entre autoridades, operadores y ciudadanos,

convirtiéndose en una herramienta clave para la gestión inteligente del tráfico y la planificación vial.

En Bogotá, además del sistema SIMUR, se han implementado diversas tecnologías de inteligencia artificial para el control del tráfico, orientadas a mejorar la movilidad y reducir la congestión. Entre ellas destacan los modelos de visión artificial, que permiten analizar imágenes capturadas por cámaras urbanas para identificar incidentes, clasificar vehículos y medir niveles de congestión; los algoritmos de aprendizaje automático supervisado, como KNN y SVM, utilizados para predecir tiempos de desplazamiento y detectar patrones de circulación; los sistemas de detección automática con cámaras de foto comparendo, que aplican reconocimiento de placas y comportamiento vehicular para sancionar infracciones; y las plataformas de análisis predictivo basadas en big data, que integran información de sensores, GPS y reportes ciudadanos para anticipar escenarios de tráfico y optimizar la gestión vial. Estas tecnologías, combinadas con la interoperabilidad de SIMUR, han permitido avanzar hacia un modelo de movilidad más inteligente, proactivo y sostenible en la ciudad.

En varias ciudades del mundo se han implementado sistemas de gestión de tráfico basados en inteligencia artificial que cumplen funciones similares a SIMUR en Bogotá, pero con enfoques avanzados en predicción, optimización y control automatizado. Los más destacados son los siguientes:

- Hangzhou, China – Sistema de tráfico de Alibaba Cloud (City Brain). Utiliza IA y big data para analizar imágenes de cámaras y sensores en tiempo real. Permite la sincronización de semáforos, detecta accidentes y ajusta rutas dinámicamente. Ha logrado reducir los

tiempos de respuesta ante emergencias y mejorar la velocidad promedio del tráfico urbano.

- Los Ángeles, Estados Unidos – ATSAC (Automated Traffic Surveillance and Control). Sistema que integra IA y visión computacional para las intersecciones semaforizadas. Ajusta automáticamente los ciclos de semáforos según la densidad vehicular. Se complementa con sensores y cámaras que alimentan modelos predictivos de congestión.
- Barcelona, España – Smart Mobility Platform. Plataforma que combina IA, IoT y análisis predictivo para gestionar tráfico, transporte público y movilidad sostenible. Integra datos de sensores urbanos, GPS y aplicaciones ciudadanas para anticipar congestión y recomendar rutas alternativas.

La comisión europea ha implementado el software Visions Project cuyo fin es permitir controlar el tráfico vehicular aplicando el algoritmo KNN (K-Nearest Neighbors). En este sistema, cámaras instaladas en vías urbanas capturan imágenes en tiempo real y el modelo KNN se entrena con características como tamaño, forma y color de los vehículos para clasificarlos en categorías (automóviles, buses, camiones o motocicletas). Una vez entrenado, el algoritmo compara cada nueva imagen con las más cercanas en el conjunto de datos, asignando la clase correspondiente. Esta clasificación permite analizar la composición del flujo vehicular, detectar patrones de congestión y generar alertas tempranas sobre posibles colapsos en la movilidad. Además, al integrarse con plataformas de gestión como SIMUR, el sistema facilita la predicción de tiempos de desplazamiento y la optimización de rutas, contribuyendo a una movilidad más eficiente y sostenible.

La Secretaría Distrital de Movilidad (SDM) de Bogotá desempeña un papel fundamental en la planificación, regulación y supervisión del sistema de transporte de la ciudad, con el objetivo de garantizar una movilidad segura, eficiente y sostenible. Por tal motivo se implementaron diversas medidas para gestionar el tráfico en la ciudad como se presenta a continuación:

- Antes de 2010. Restricción vehicular por número de placa para reducir la congestión en horas pico (Pico y placa).
- 2010 a 2015. Integración de rutas y mejora de la cobertura del Sistema Integrado de Transporte Público.
- 2016 a 2020. Introducción de cámaras de foto comparendo y sensores de tráfico.
- 2021 a 2023. Consolidación del Sistema Integrado de Información (SIMUR) sobre Movilidad Urbano Regional para monitoreo y análisis de datos.
- 2024 a 2025. Expansión de herramientas como "Mi Movilidad" para trámites, alertas y consultas en línea. Medidas para reducir emisiones vehiculares y fomentar el uso de transporte sostenible.

El estado del arte sobre el Sistema Integrado de Información sobre Movilidad Urbana Regional (SIMUR) y la problemática de tránsito en Bogotá, Colombia, se fundamenta en una base teórica que abarca antecedentes, estudios actuales y una comparativa de herramientas existentes. SIMUR, gestionado por la Secretaría Distrital de Movilidad de Bogotá, facilita la comunicación y el intercambio de información entre los actores de la movilidad urbana, integrando aplicaciones para la consulta de datos relevantes y reportes en tiempo real. Los estudios actuales

destacan los retos de movilidad en Bogotá, como el crecimiento del parque automotor y la limitada capacidad de la infraestructura vial, que generan congestión y afectan la calidad de vida. Comparativamente, SIMUR se posiciona como una herramienta avanzada frente a otras soluciones tecnológicas, al incorporar modelos de percepción computacional y aprendizaje automático para el análisis predictivo y la gestión eficiente del tráfico. La integración de inteligencia artificial y la participación ciudadana son claves para optimizar la movilidad urbana y avanzar hacia sistemas de transporte inteligentes y sostenibles.

SIMUR cuenta con una serie de herramientas clave que son utilizadas por otros sistemas de información, entre ellas destacan las siguientes:

- Consulta del Historial de Comparendos. Permite a los ciudadanos revisar las infracciones de tránsito registradas a su nombre.
- Consulta de Vehículos Exceptuados. Muestra los vehículos autorizados para circular durante restricciones como el Pico y Placa.
- Mapa de Tráfico en Tiempo Real. Visualiza el estado actual de las vías principales de Bogotá. Muestra niveles de congestión, cierres viales, y obras en curso. Ayuda a los ciudadanos a planificar rutas más eficientes y a las autoridades a tomar decisiones operativas.

En el mercado existen aplicaciones que interactúan con SIMUR sobre todo con la funcionalidad de mapa de tráfico en tiempo real. Entre estas destacan las siguientes:

- FiveMobile. Utiliza datos geográficos y de tráfico para apoyar inspecciones, encuestas y gestión de rutas en terreno. Permite visualizar rutas y puntos críticos en tiempo real, lo que es útil para logística y movilidad operativa.
- Aplicaciones móviles de movilidad urbana. Apps disponibles en App Store y Google Play que permiten a los ciudadanos consultar el estado del tráfico, restricciones de Pico y Placa, y rutas recomendadas.

- Sistemas de gestión de PMT (Planes de Manejo de Tránsito). Empresas contratistas y operadores viales utilizan software que consume datos de SIMUR para planificar desvíos, señalización temporal y afectaciones viales.

El problema de la congestión vehicular en la zona de Centro Mayor podría abordarse mediante la integración de un complemento al Sistema Integrado de Información sobre Movilidad Urbana Regional (SIMUR). Este tipo de integración convertiría a SIMUR en una herramienta aún más poderosa para enfrentar los retos de movilidad en zonas críticas como Centro Mayor, promoviendo una ciudad más conectada, eficiente y sostenible.

2.3. Conclusiones

Las conclusiones del estado del arte sobre el Sistema Integrado de Información sobre Movilidad Urbana Regional (SIMUR) y la problemática de movilidad en Bogotá indican que SIMUR es una herramienta fundamental para la gestión de la movilidad en la ciudad. Sin embargo, para enfrentar los desafíos actuales de congestión y planificación, especialmente en zonas críticas como Centro Mayor, es necesario integrar tecnologías avanzadas como la percepción computacional y el aprendizaje automático. Estas tecnologías permitirán generar alertas tempranas y análisis predictivos sobre tiempos de desplazamiento y situación del tráfico, mejorando la eficiencia y la calidad de vida de los ciudadanos.

La evolución hacia sistemas inteligentes en SIMUR facilitará una gestión más proactiva y eficiente de la movilidad urbana, beneficiando tanto a las autoridades como a la comunidad. La implementación de complementos tecnológicos basados en inteligencia artificial y la participación ciudadana son claves para optimizar la planificación vial y avanzar hacia una movilidad más sostenible y adaptada a las necesidades reales del

territorio. En resumen, el fortalecimiento de SIMUR mediante el uso de tecnologías avanzadas es esencial para abordar los retos de movilidad en Bogotá y mejorar la calidad del servicio y la seguridad en el tránsito.

Las acciones implementadas por la Secretaría Distrital de Movilidad (SDM) de Bogotá reflejan una evolución progresiva en la gestión del tráfico urbano, desde medidas restrictivas como el Pico y Placa hasta el desarrollo de herramientas tecnológicas avanzadas como SIMUR. Esta transformación evidencia el compromiso institucional por garantizar una movilidad segura, eficiente y sostenible, adaptándose a los desafíos que impone el crecimiento poblacional y vehicular en la ciudad. La planificación estratégica y la incorporación de sistemas inteligentes han permitido una mejor supervisión del tránsito, optimización de rutas y fortalecimiento del transporte público.

SIMUR se ha consolidado como una plataforma clave en la articulación de datos, actores y decisiones dentro del ecosistema de movilidad urbana. Su capacidad para integrar herramientas como el Mapa de Tráfico en Tiempo Real, la consulta de comparendos y el control de excepciones vehiculares, lo posiciona como un referente en la gestión inteligente del tráfico. Además, su interoperabilidad con software de terceros como FiveMobile y sistemas de PMT demuestra su potencial para expandirse más allá del ámbito institucional, facilitando soluciones colaborativas entre entidades públicas, privadas y ciudadanía.

La propuesta de integrar un complemento específico para zonas críticas como Centro Mayor representa una oportunidad para potenciar aún más el alcance de SIMUR. Esta integración permitiría una respuesta más ágil y localizada ante la congestión vehicular, mediante el uso de inteligencia artificial, participación ciudadana y análisis predictivo. En conjunto, estas estrategias no solo mejorarían la movilidad en áreas de alta demanda, sino que también contribuirían a una Bogotá más conectada, resiliente y orientada hacia un modelo de transporte urbano sostenible

3. Objetivos y metodología de trabajo

3.1. Objetivo general

El objetivo general de este proyecto es optimizar la gestión de la movilidad en una zona urbana específica, como la localidad de Tunjuelito en el sector de Centro Mayor, reduciendo los impactos negativos de la congestión en la calidad de vida de los ciudadanos y en la eficiencia del desplazamiento.

La implementación de una herramienta de visión artificial en el sistema de información SIMUR tendrá un impacto significativo en la gestión del tráfico urbano. Al analizar imágenes recolectadas del sector, esta herramienta permitirá identificar patrones de circulación, tipos de vehículos y congestiones en zonas críticas como Centro Mayor en Bogotá. Esto mejorará la capacidad de anticipar y gestionar el tráfico, reduciendo tiempos de desplazamiento y mejorando la calidad de vida urbana. Además, la precisión de los sistemas de visión artificial superará la capacidad humana para detectar incidentes y anomalías, contribuyendo a una toma de decisiones más eficiente y a la activación automática de alertas de movilidad. La automatización de la recolección y procesamiento de datos visuales permitirá que SIMUR procese grandes volúmenes de información sin intervención humana directa, optimizando la respuesta ante eventos de tráfico. Finalmente, la incorporación de modelos de aprendizaje automático y percepción computacional facilitará la generación de alertas tempranas y la planificación proactiva de la movilidad urbana.

3.2. Objetivos específicos

- Implementar un modelo de visión artificial capaz de identificar y clasificar incidentes de tráfico en la zona de Centro Mayor en Bogotá, con una precisión mínima del 90%, dentro de los próximos

6 meses, utilizando datos capturados por cámaras urbanas y sensores del sistema SIMUR.

- Desarrollar un sistema basado en aprendizaje automático y percepción computacional que prediga los tiempos de desplazamiento en rutas críticas de la localidad de Tunjuelito en la zona de Centro Mayor, logrando reducir el margen de error a menos del 10% en un periodo de 4 meses.
- Diseñar e implementar un módulo inteligente que, mediante análisis de imágenes, emita alertas automáticas de congestión con al menos 15 minutos de anticipación y una tasa de falsos positivos inferior al 5%, en el sector de Centro Mayor, durante el próximo trimestre.
- Desarrollar un sistema de clasificación y caracterización de tipos de vehículos (automóviles, camiones, buses, motos) que permita analizar la composición del flujo vehicular y su impacto en la movilidad y el medio ambiente en la zona de Centro Mayor.
- Implementar un modelo de análisis de patrones de circulación y comportamiento vial que detecte horarios de mayor afluencia y comportamientos atípicos, con el objetivo de optimizar la planificación vial y la gestión de semáforos en el sector.
- Optimizar el rendimiento y los tiempos de respuesta de la herramienta de gestión de tráfico basada en visión artificial y aprendizaje automático implementada en la zona de Centro Mayor en Bogotá, garantizando que el sistema procese y analice los datos visuales en tiempo real, con una latencia máxima de 5 segundos desde la captura de la imagen hasta la generación de alertas automáticas.

- Implementar una metodología de desarrollo ágil basada en Scrum para coordinar la planificación, diseño, desarrollo, pruebas e implementación de los sistemas inteligentes de gestión de tráfico en la zona de Centro Mayor, asegurando entregas funcionales cada dos semanas, con revisión continua de avances, retroalimentación de actores clave y mejora incremental del software durante todo el ciclo de desarrollo.
- Garantizar que el software desarrollado cumpla con estándares de calidad técnica, funcional y de experiencia de usuario mediante la implementación de pruebas automatizadas, validaciones continuas, revisiones de código y métricas de rendimiento, asegurando una tasa de errores críticos inferior al 2% y una satisfacción del usuario superior al 85% durante el ciclo de vida del proyecto.

3.3. Metodología de trabajo

La congestión vehicular en la zona de Centro Mayor representa un desafío crítico para la movilidad urbana en Bogotá. Esta problemática afecta la eficiencia del transporte, la calidad del aire y la experiencia ciudadana. En respuesta, se propone el desarrollo e integración de un complemento especializado al Sistema Integrado de Información sobre Movilidad Urbana Regional (SIMUR), gestionado por la Secretaría Distrital de Movilidad (SDM), con el fin de optimizar el monitoreo, análisis y gestión del tráfico en esta área estratégica.

Aplicar una metodología de trabajo como Scrum en el desarrollo de software es fundamental para garantizar eficiencia, adaptabilidad y colaboración continua entre los equipos. Scrum permite dividir el proyecto en ciclos cortos llamados sprints, lo que facilita la entrega incremental de funcionalidades, la detección temprana de errores y la incorporación de cambios según las necesidades del cliente. Además, promueve la

comunicación constante entre desarrolladores, usuarios y líderes de proyecto, lo que mejora la transparencia, la toma de decisiones y la calidad del producto final. Esta metodología es especialmente valiosa en entornos dinámicos donde los requisitos pueden evolucionar rápidamente.

Scrum es una metodología ágil de gestión de proyectos que se utiliza principalmente en el desarrollo de software para fomentar la colaboración, la adaptabilidad y la entrega continua de valor. Se basa en ciclos cortos llamados sprints, donde equipos multidisciplinarios trabajan de manera iterativa para construir y mejorar el producto.

Scrum promueve roles definidos (como Scrum Máster, Product Owner y equipo de desarrollo), reuniones periódicas (como la planificación del sprint, las reuniones diarias y la retrospectiva), y una cultura de mejora constante. Su enfoque flexible permite responder rápidamente a cambios en los requisitos y mantener una comunicación abierta entre todos los involucrados.

Las etapas de Scrum se organizan en ciclos iterativos llamados sprints, que comienzan con la planificación del sprint, donde el equipo define los objetivos y selecciona las tareas del backlog que se abordarán. Durante el sprint, se realizan reuniones diarias (daily stand-ups) para revisar el progreso, identificar obstáculos y coordinar esfuerzos.

Al finalizar el sprint, se lleva a cabo la revisión del sprint, en la que se presentan los avances al Product Owner y se recopila retroalimentación.

Finalmente, se realiza la retrospectiva del sprint, donde el equipo reflexiona sobre el proceso y propone mejoras para el siguiente ciclo.

Estas etapas fomentan la entrega continua de valor, la mejora constante y la colaboración efectiva.

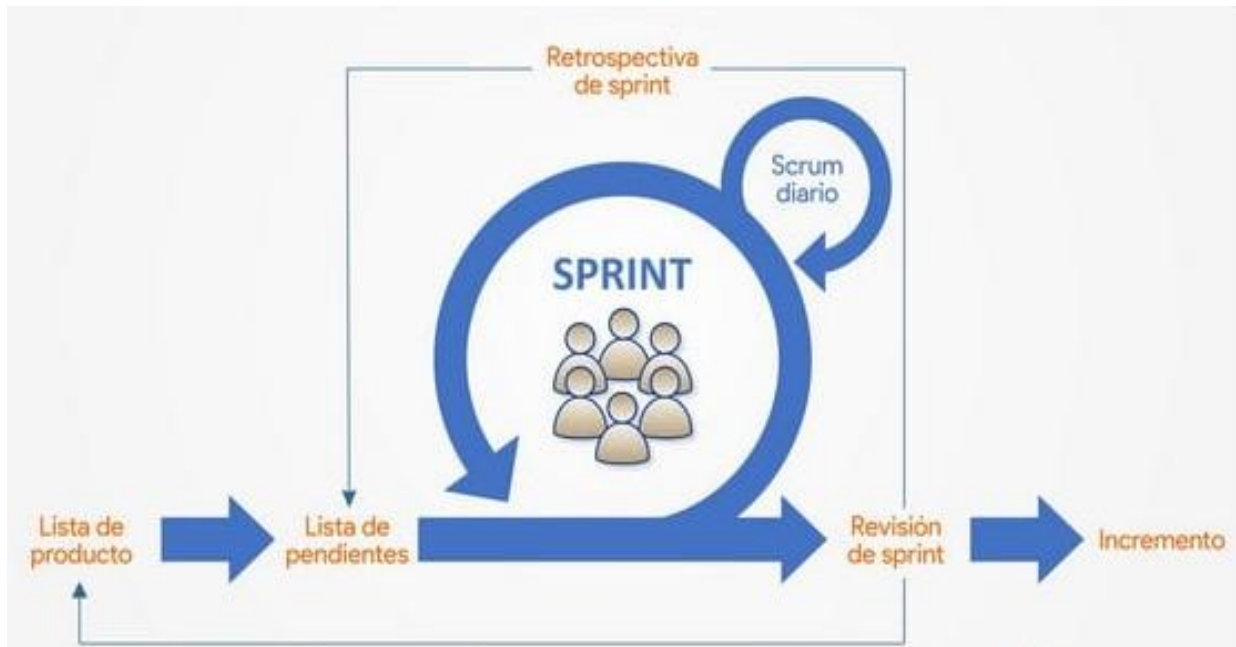


Figura 3. Flujo del proceso Scrum.

Diagnóstico y análisis del contexto

- Objetivo: Identificar las causas específicas de la congestión en Centro Mayor.
- Actividades:
 - Recolección de datos históricos de tráfico en la zona de centro mayor.
 - Análisis de infraestructura vial, horarios pico, y comportamiento vehicular.

Diseño del complemento SIMUR

- Objetivo: Crear un módulo especializado para monitorear y gestionar el tráfico en Centro Mayor.
- Actividades:

- Definición de funcionalidades: alertas, predicción de congestión, gestión de desvíos.
- Diseño de interfaz para operadores y ciudadanos.

Desarrollo e implementación

- Objetivo: Construir y desplegar el complemento.
- Actividades:
 - Programación del módulo y conexión con bases de datos SIMUR.
 - Pruebas piloto en horarios críticos.

Monitoreo y evaluación

- Objetivo: Medir el impacto del complemento en la reducción de congestión.
- Actividades:
 - Seguimiento de indicadores: velocidad promedio, tiempo de viaje, número de incidentes.
 - Comparación con datos previos a la implementación.

4. Desarrollo específico de la contribución

El desarrollo de software es un proceso estructurado que abarca la planificación, diseño, codificación, prueba y mantenimiento de programas informáticos con el fin de resolver problemas o satisfacer necesidades específicas. Inicia con la recopilación de requisitos, donde se definen los objetivos del sistema y las funcionalidades deseadas. Luego se diseña la arquitectura y se elige la tecnología adecuada para implementar el código fuente, seguido por pruebas que garantizan la calidad y el correcto funcionamiento. Finalmente, se realiza el despliegue y mantenimiento continuo para asegurar que el software se adapte a cambios futuros y conserve su eficiencia y seguridad.

Para tratar la congestión vehicular recurrente en zonas de alta afluencia de Bogotá, particularmente en la localidad de Tunjuelito y el sector de Centro Mayor, donde los tiempos de desplazamiento se ven significativamente afectados y generan impactos negativos en la calidad de vida de los ciudadanos se busca desarrollar un software que cuente con características de análisis de la información a partir de imágenes del sector evaluando el nivel de tráfico y clasificándolo según su intensidad. Por medio de la metodología SCRUM se gestionará el desarrollo del proyecto y conseguir el producto final que cubra la serie de requerimientos expuestos en los apartados del documento.

Al implementar SCRUM en un proyecto se obtienen un buen número de beneficios para equipos y organizaciones que pretenden mejorar su eficiencia y adaptabilidad, entre ellos encontramos los siguientes:

- Mejora en la colaboración y comunicación del equipo.

- Flexibilidad ya que permite responder rápidamente a cambios en los requisitos o prioridades establecidas en el proyecto.
- Entrega continúa de valor debido a sus cortos periodos de trabajo en los cuales se entregan funcionalidades con la calidad requerida.

Un sprint es un ciclo de trabajo corto y definido dentro del marco de Scrum, generalmente de una a cuatro semanas, durante el cual un equipo multidisciplinario se enfoca en completar un conjunto específico de tareas o funcionalidades del producto. Un incremento es el resultado tangible y funcional del trabajo completado durante un sprint, que se suma a los incrementos anteriores y representa un paso hacia el producto final. Una historia de usuario es una descripción breve y estructurada de una necesidad o funcionalidad desde la perspectiva del usuario final, utilizada en metodologías ágiles como Scrum para guiar el desarrollo de software.

Los criterios de aceptación son condiciones específicas que una historia de usuario debe cumplir para considerarse completada correctamente dentro de un proceso ágil como Scrum. Estos criterios definen el comportamiento esperado del sistema, los límites funcionales y las validaciones necesarias, asegurando que el producto desarrollado responda a las necesidades del usuario y cumpla con los estándares de calidad.

INVEST es un acrónimo utilizado para evaluar la calidad de una historia de usuario en metodologías ágiles como Scrum. Significa que cada historia debe ser Independiente (puede desarrollarse por separado), Negociable (flexible y abierta a discusión), Valiosa (aporta valor al usuario o al negocio), Estimable (puede ser dimensionada en esfuerzo), Pequeña (lo suficientemente breve para completarse en un sprint) y Testeable (verificable mediante criterios de aceptación claros).

Durante el sprint review, las historias de usuario completadas durante el sprint son presentadas al equipo y a los stakeholders para demostrar su funcionamiento y validar que cumplen con los criterios de aceptación previamente definidos. Este evento permite revisar el incremento del producto, obtener retroalimentación directa, y evaluar si las funcionalidades entregadas aportan valor al usuario final.

Identificación de requisitos

A continuación, se presenta el listado de historias de usuario que conforman el proyecto y con las cuales se plantea el plan de trabajo para desarrollar el sistema de información utilizando la metodología Scrum en 4 Sprint:

- 1. Como usuario, quiero reportar incidentes de tráfico desde mi celular, para contribuir al monitoreo colaborativo de la movilidad urbana. Criterios de aceptación:
 - Acceso móvil intuitivo. El usuario puede acceder a la funcionalidad desde una aplicación móvil con una interfaz clara y responsiva.
- 2. Como administrador, quiero integrar los datos de SIMUR, para mejorar la precisión de rutas y puntos críticos en inspecciones de campo. Criterios de aceptación:
 - Acceso a datos estructurados desde SIMUR. El sistema puede conectarse a la fuente de datos de SIMUR y extraer información relevante.
 - Actualización periódica y validación de datos.

- 3. Como administrador, quiero gestionar los permisos de acceso a los módulos del sistema, para garantizar la seguridad y el uso adecuado de la información. Criterios de aceptación:
 - Asignación de roles y permisos por módulo.
 - Interfaz de gestión de permisos accesible y auditable.
 - Restricción efectiva de acceso.

- 4. Como usuario, quiero iniciar sesión de forma segura usando autenticación de dos pasos, para proteger mi información personal. Criterios de aceptación:
 - Verificación en dos pasos activada por defecto o configurable.
 - Código de verificación obligatorio tras ingresar credenciales.
 - Mensajes claros de éxito o error en cada paso del proceso.

- 5. Como usuario, quiero actualizar mis datos personales en mi perfil, para asegurar que las notificaciones y trámites lleguen correctamente. Criterios de aceptación:
 - Acceso seguro al formulario de edición.
 - Validación de campos obligatorios y formato.
 - Confirmación de actualización y reflejo inmediato.

- 6. Como administrador, quiero gestionar las cuentas de usuario, roles y permisos, para garantizar el acceso adecuado a cada módulo del sistema. Criterios de aceptación:
 - Creación, edición y eliminación de cuentas de usuario.
 - Asignación y modificación de roles con permisos específicos.
 - Control de acceso efectivo y trazabilidad

- 7. Como administrador, quiero aplicar filtros de reducción de ruido a las imágenes de tráfico, para mejorar la precisión de los algoritmos de detección de incidentes. Criterios de aceptación:
 - Aplicación automática de filtros al recibir imágenes.
 - Evaluación visual y técnica del filtrado.
 - Impacto positivo en la detección de incidentes.
- 8. Como administrador, quiero que las imágenes se conviertan a escala de grises antes del análisis, para facilitar el entrenamiento de modelos de visión computacional. Criterios de aceptación:
 - Conversión automática al cargar imágenes.
 - Verificación visual de la conversión.
- 9. Como administrador, quiero que las imágenes sean normalizadas en formato y resolución antes de ser ingresadas al sistema de aprendizaje automático, para garantizar consistencia en los datos de entrenamiento. Criterios de aceptación:
 - Conversión automática al formato estándar
 - Redimensionamiento uniforme de resolución
- 10. Como administrador, quiero contar con un modulo que me permita cargar las imágenes del sector centro mayor que serán utilizadas para analizar el tráfico de la zona. Criterios de aceptación:
 - Interfaz de carga accesible y segmentada por zona.
 - Confirmación de carga y disponibilidad para análisis.
- 11. Como administrador, quiero contar con un modulo que me permita realizar por medio de aprendizaje automático la clasificación de las imágenes para obtener los resultados de estado del tráfico en el sector de centro mayor. Criterios de aceptación:
 - Clasificación automática por estado de tráfico.

- Interfaz de carga y visualización de resultados.
- 12. Como administrador, requiero que la información manejada en el sistema se almacene en una base de datos. Criterios de aceptación:
 - Persistencia automática de datos en operaciones clave.
 - Estructura normalizada y validada.
- 13. Como usuario, requiero un reporte que me presente el estado actual del tráfico en el sector centro mayor. Criterios de aceptación:
 - Acceso en tiempo real al reporte desde la interfaz del sistema.
 - Visualización clara y segmentada del estado del tráfico.
- 14. Como usuario, requiero un reporte que me presente un análisis de la cantidad de flujo vehicular durante un rango de tiempo determinado. Criterios de aceptación:
 - Selección personalizada del rango de tiempo.
 - Exportación y trazabilidad del reporte.
- 15. Como usuario, requiero un reporte que me presente un análisis de índices de ruido generados por la gran afluencia de vehículos en la zona durante horas pico. Criterios de aceptación:
 - Visualización de niveles de ruido correlacionados con el flujo vehicular.

Recopilación de datos: En primer lugar, para esta tarea se propone utilizar imágenes del flujo vehicular captadas por cámaras ubicadas en el sector de Centro Mayor. Asegurar la disponibilidad y calidad de las imágenes para el análisis posterior.

Roboflow es una plataforma especializada en visión por computadora que facilita el desarrollo, entrenamiento y despliegue de modelos de inteligencia artificial para la clasificación y detección de imágenes. Su principal aporte es simplificar el ciclo completo de trabajo: desde la recolección y anotación de datos, hasta el preprocesamiento, entrenamiento y exportación de modelos listos para producción. Roboflow ofrece herramientas en línea para gestionar datasets, aplicar aumentos de datos (data augmentation) y entrenar modelos con arquitecturas modernas como YOLO, EfficientDet o ResNet, sin necesidad de una infraestructura compleja. Además, permite integrar los modelos entrenados en aplicaciones web, móviles o sistemas embebidos, lo que la convierte en una solución práctica y accesible para proyectos de tráfico vehicular, seguridad, salud y otras áreas donde la visión artificial es clave.

Aprendizaje automático: Emplear técnicas de visión artificial para identificar y clasificar vehículos (automóviles, camiones, buses, motos). Detectar incidentes y anomalías en el tráfico mediante algoritmos especializados.

Modelos de aprendizaje automático: Aplicar redes neuronales y algoritmos de aprendizaje automático permitirá analizar de manera precisa los patrones de circulación vehicular en zonas de alta afluencia. Asimismo, estas técnicas ofrecen información útil para la planificación vial y la toma de decisiones estratégicas. La integración de modelos de inteligencia artificial en este contexto no solo optimiza la gestión del tráfico urbano, sino que también contribuye a mejorar la calidad de vida de los ciudadanos al reducir tiempos de viaje, minimizar el impacto ambiental y fortalecer la movilidad sostenible.

En la siguiente tabla se presenta el plan de trabajo aplicando la metodología Scrum durante 4 Sprint. Cada Sprint tiene una duración de 15 días calendario:

Número de Sprint	Historias de usuario involucradas
1	2, 3, 4, 5, 12
2	6, 1, 10
3	7, 8, 9
4	11, 13, 14, 15

Tabla 1. Distribución de las historias de usuario durante los 4 sprints.

Durante la revisión del sprint se realizan varias actividades clave que permiten inspeccionar el trabajo realizado durante el sprint y adaptar la lista de pendientes del producto. El equipo de desarrollo presenta las historias de usuario completadas, demostrando su funcionalidad y verificando que cumplen con los criterios de aceptación.

Descripción de la herramienta software desarrollada

La herramienta software desarrollada es un complemento para el Sistema Integrado de Información sobre Movilidad Urbana Regional (SIMUR), enfocado en la zona de Centro Mayor en Bogotá. Su objetivo principal es generar alertas de movilidad en áreas con alta congestión vehicular, optimizando la gestión del tráfico y mejorando la toma de decisiones tanto para autoridades como para la ciudadanía.

Utiliza imágenes captadas por cámaras del sector para monitorear el flujo vehicular, emplea técnicas de visión artificial para identificar y clasificar diferentes tipos de vehículos y detectar incidentes en tiempo real, e implementa redes neuronales y algoritmos de aprendizaje automático para analizar patrones de circulación, predecir congestiones y estimar tiempos de desplazamiento.

La base de datos a implementar es Mysql y el modelo relacional es el siguiente:

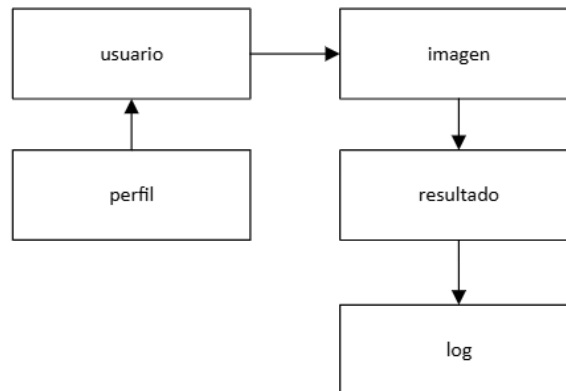


Figura 4. Modelo relacional de la base de datos.

Tabla Usuario

Campo	Tipo	Detalle
IdUsuario	Int 11 Primary Key	Auto incrementable
NombreCompleto	VarChar(100)	Nombre completo del usuario
IdPerfil	Int 11 Foreign Key	Id del perfil asignado
Clave	Varchar(100)	Clave encriptada en md5
Celular	Varchar(15)	Número de celular para notificaciones.
Email	Varchar(150)	Correo electrónico para notificaciones.

Tabla 2. Atributos de la tabla Usuario.

Tabla Perfil

Campo	Tipo	Detalle
IdPerfil	Int 11 Primary Key	Auto incrementable
Detalle	VarChar(100)	Perfil: Administrador, usuario

Tabla 3. Atributos de la tabla Perfil.

Tabla log

Campo	Tipo	Detalle
IdLog	Int 11 Primary Key	Auto incrementable
Detalle	VarChar(100)	Detalle movimiento
Fecha	TimeStamp	Fecha y hora automática
Usuario	Varchar(100)	Usuario en sesión

Tabla 4. Atributos de la tabla log.

Tabla imagen

Campo	Tipo	Detalle
IdImagen	Int 11 Primary Key	Auto incrementable
Archivo	VarChar(100)	Nombre imagen
Fecha	TimeStamp	Fecha y hora automática
IdUsuario	Int 11	Usuario de carga
TotalVehiculos	Int 11	Total de vehículos, camiones y

		motos detectadas.
TotalAutos	Int 11	Total autos detectados
TotalCamiones	Int 11	Total camiones o buses detectados
TotalMotos	Int 11	Total motos detectadas

Tabla 5. Atributos de la tabla imagen.

El diagrama de flujo del proceso parte del banco de imágenes del sector “Centro Mayor” que se ha logrado obtener a partir de cámaras de seguridad. A partir de estas se extraen datos fundamentales para almacenar los totales de vehículos, camiones y motos que transitan la zona en diferentes horarios. Se clasifica esta información y se deja disponible en una base de datos para generar los reportes respectivos. El diagrama del proceso se presenta a continuación:

“*Centro mayor”. Es la zona en Bogotá donde se aplicará el modelo desarrollado.

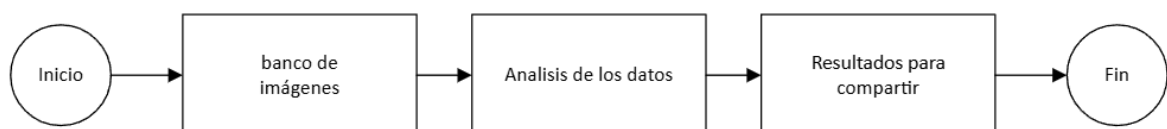


Figura5. Flujo principal de la solución.

El proceso de banco de imágenes en el sistema inteligente de alertas de tráfico consiste en la recopilación, organización y almacenamiento estructurado de imágenes capturadas por cámaras ubicadas en puntos estratégicos de la zona de Centro Mayor, Bogotá. Estas imágenes son recolectadas de forma continua y clasificadas según parámetros como ubicación, fecha, hora y condiciones de tráfico observadas. El banco de

imágenes actúa como la fuente primaria de datos visuales para alimentar los modelos de percepción computacional y aprendizaje automático, permitiendo identificar patrones de circulación, tipos de vehículos y niveles de congestión.

El proceso de análisis de los datos en el sistema inteligente de alertas de tráfico se centra en la interpretación de las imágenes recolectadas por el banco visual, aplicando técnicas de visión artificial y algoritmos de aprendizaje automático supervisado, específicamente KNN (K-Nearest Neighbors). Este análisis comienza con la extracción de características relevantes de cada imagen, como la forma, tamaño, color y posición de los vehículos, que luego se transforman en vectores de datos. El modelo KNN compara estos vectores con un conjunto previamente etiquetado para clasificar los tipos de vehículos (automóviles, buses, camiones, motocicletas) y detectar patrones de circulación.

El proceso de resultados en el sistema inteligente de alertas de tráfico consiste en transformar los hallazgos del análisis de datos en información clara, útil y accesible para los diferentes actores de la movilidad. Una vez clasificados los vehículos y detectados los patrones de congestión mediante el modelo KNN, el sistema genera reportes visuales y numéricos que incluyen indicadores como densidad vehicular, tiempos de desplazamiento estimados y alertas tempranas de congestión.

El diagrama de arquitectura del sistema representa la estructura lógica y física de sus componentes, mostrando cómo interactúan entre sí para cumplir los objetivos funcionales. Incluye capas como presentación, lógica de negocio y persistencia de la información. A continuación, se presenta el diagrama de arquitectura de la solución propuesta:

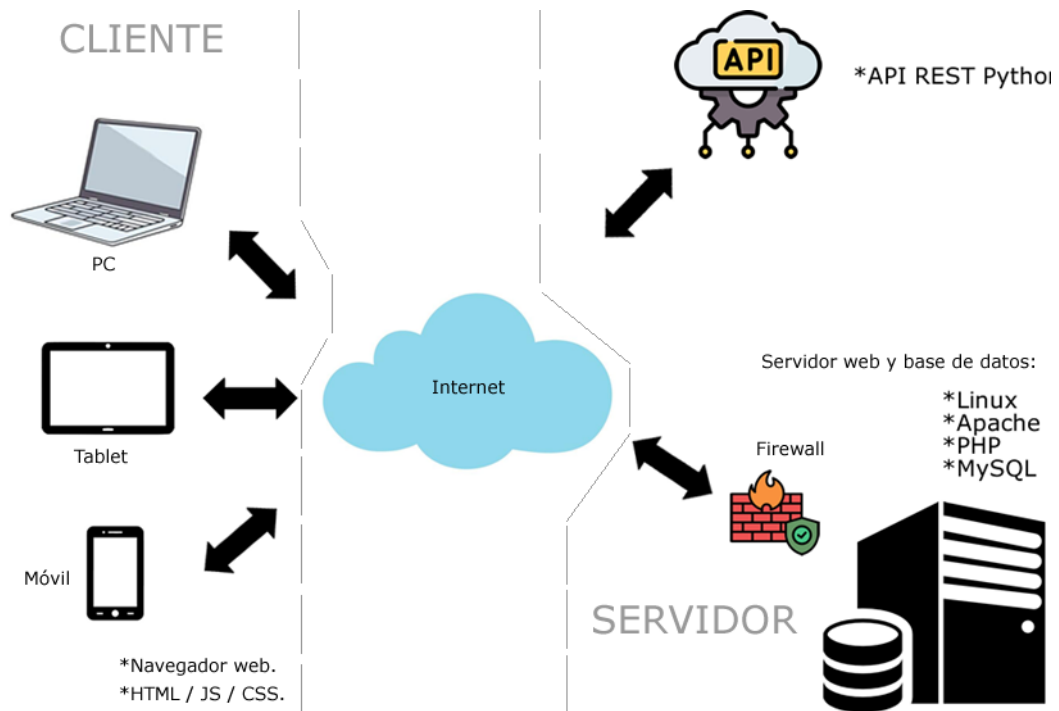


Figura6. Diagrama de arquitectura del sistema.

Cliente	Navegador web, HTML, JavaScript, CSS, JQuery
Cliente	Dispositivo móvil con navegador.
Cliente	Tablet con navegador.
Servidor	Linux, Apache, Python, PHP, dominio
Servidor	Base de datos MySQL
Servidor	API REST Python para preprocesamiento de imágenes
Servidor	API REST Python para machine learning
Google	Looker Studio. Tableros de control. Conexión remota a base de datos MySQL.

Tabla 6. Detalle arquitectura del proyecto

Una API REST (Interfaz de Programación de Aplicaciones basada en Transferencia de Estado Representacional) sirve para facilitar la comunicación entre diferentes sistemas a través de internet, utilizando el protocolo HTTP. Permite que aplicaciones puedan acceder, manipular y compartir datos de manera estructurada y segura, sin necesidad de conocer los detalles internos del sistema con el que interactúan. Para el proyecto se establecen las siguientes APIs:

API preprocesamiento

El preprocesamiento de imágenes en el contexto de visión por computadora y aprendizaje automático consiste en una serie de técnicas aplicadas a las imágenes antes de ser utilizadas por un modelo o algoritmo. Su objetivo principal es mejorar la calidad de los datos, reducir el ruido y estandarizar las características visuales para facilitar el análisis. Entre las operaciones comunes se incluyen el cambio de tamaño, la normalización de valores de píxeles, la conversión a escala de grises, la eliminación de fondo, el aumento de datos (como rotaciones o recortes), y la detección de bordes. Estas transformaciones permiten que los modelos trabajen con entradas más consistentes y relevantes, mejorando su precisión y eficiencia en tareas como clasificación, segmentación o reconocimiento de objetos.

Ruta: preprocesamiento/

Method: POST

Entradas:

NombreVariable	TipoDato	Detalle
Token	Varchar	Token de seguridad
Archivo	Multipart	Archivo de la imagen

Tabla 7. Entradas API preprocesamiento.

Salida JSON:

NombreVariable	TipoDato	Detalle
Codigo	int	200 ok 500 Error 401 Permisos
Resultado	Varchar	Mensaje devuelto por el proceso

Tabla 8. Salidas API preprocesamiento.

Evaluación

El plan de pruebas para el software de gestión de movilidad urbana desarrollado incluye varias etapas clave para garantizar su calidad y funcionalidad. Primero, se debe analizar el producto para comprender sus funcionalidades principales, como la gestión del tráfico y la generación de alertas. Luego, se desarrolla una estrategia de prueba que incluye pruebas de caja blanca y caja negra, y se definen los objetivos de la prueba, como validar la identificación de patrones de circulación y la generación de alertas automáticas. Se establecen criterios de prueba, se planifican los recursos necesarios y se configura un entorno de prueba que simule condiciones reales de tráfico. Además, se programan hitos y plazos claros para la ejecución de las pruebas, y se documentan los resultados obtenidos y las incidencias detectadas. Finalmente, se realizan pruebas unitarias, de integración, de regresión y de alto nivel para asegurar la calidad del producto final. A continuación, se presenta el diagrama del proceso de pruebas a realizar:



Figura 7. Ciclo de pruebas de calidad de software.

- Planificación de pruebas: Se definen los objetivos, alcance, recursos, cronograma y tipos de pruebas a realizar. Se identifican los criterios de entrada y salida para cada fase.
- Diseño de casos de prueba: Se crean los escenarios y condiciones específicas que se van a evaluar. Se documentan los datos de prueba, pasos a seguir y resultados esperados.
- Preparación del entorno de pruebas: Se configura el hardware, software, redes y herramientas necesarias para ejecutar las pruebas. Se asegura que el entorno sea lo más parecido al de producción.

- Ejecución de pruebas: Se aplican los casos de prueba al software. Se registran los resultados obtenidos y se comparan con los esperados.
- Gestión de defectos: Se documentan los errores encontrados. Se asignan a los desarrolladores para su corrección y se realiza seguimiento.
- Evaluación final y cierre: Se analiza la cobertura de pruebas, defectos encontrados y corregidos. Se genera un informe final y se aprueba el cierre del ciclo si se cumplen los criterios de calidad.

En el marco del desarrollo del complemento inteligente para el sistema SIMUR, resulta fundamental definir métricas que permitan evaluar la calidad y efectividad del proyecto. Estas métricas no solo garantizan la validez técnica del modelo de aprendizaje automático y de los procesos de visión artificial, sino que también aseguran que los resultados generados sean útiles y confiables para las autoridades y la ciudadanía. Medir aspectos como la precisión en la clasificación vehicular, la rapidez en la generación de alertas y la relevancia de la información compartida permite validar el desempeño del sistema en condiciones reales de tráfico. Las métricas establecidas son las siguientes:

- Precisión de clasificación vehicular: Mide el porcentaje de vehículos correctamente identificados por el modelo KNN en comparación con las etiquetas reales. Una alta precisión indica que el sistema distingue de manera confiable entre automóviles, buses, camiones y motocicletas. Se calcula como el número de clasificaciones correctas dividido entre el total de clasificaciones realizadas.

- Tiempo de respuesta del sistema: Evalúa el tiempo que tarda el sistema en procesar una imagen desde su captura hasta la generación de una alerta o reporte. Se mide en segundos o milisegundos y refleja la eficiencia de la arquitectura y los algoritmos implementados.
- Tasa de alertas relevantes: Indica qué proporción de las alertas emitidas por el sistema corresponden realmente a situaciones de congestión o incidentes válidos. Se calcula como el número de alertas correctas dividido entre el total de alertas emitidas.

En la herramienta en línea Roboflow utilizando una cuenta gratuita de 14 días de prueba, se realizó el proceso de clasificación para las 180 imágenes de tráfico del sector centro mayor de la localidad de Tunjuelito (Anexo 3. Carpeta img). Debido a que los archivos en su totalidad pesaban 287 Mb, se diseñó un algoritmo en Python para reducir el peso de cada imagen y con esto agilizar el proceso de carga de las imágenes den Roboflow (Ver anexo 1, Código fuente para reducir el tamaño de las imágenes).

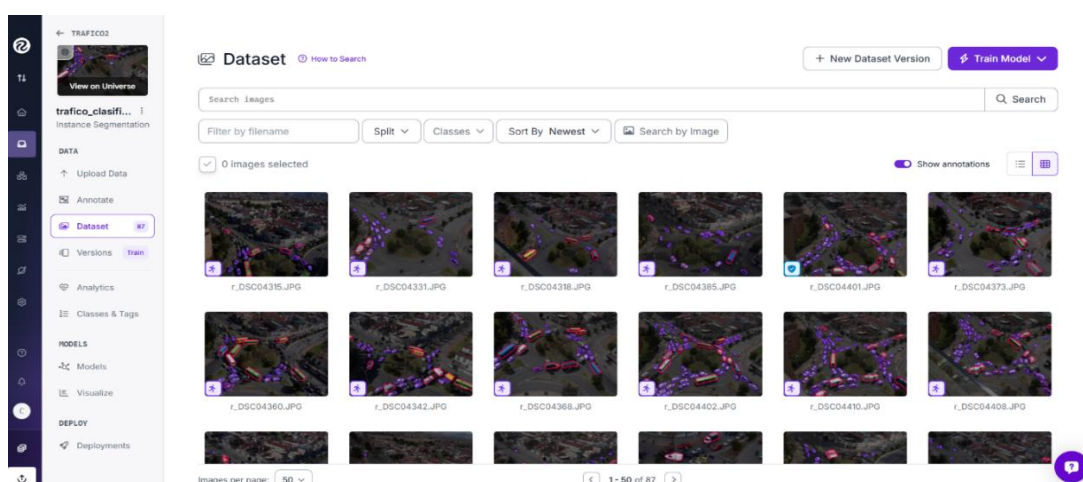


Figura 8. Proceso de clasificación de imágenes en la herramienta Roboflow.

Es importante aclarar que para cada imagen se realizó su respectiva clasificación. Se clasificaban “carros” y “camiones”, con el fin de obtener un modelo que permita más adelante recibir una nueva imagen del estado de tráfico y sea capaz de identificar los carros y camiones que se encuentren en ella (Ver anexo B. Clasificador de transporte ligero y pesado y anexo E. trafico_clasificacion.v1i.yolov11).

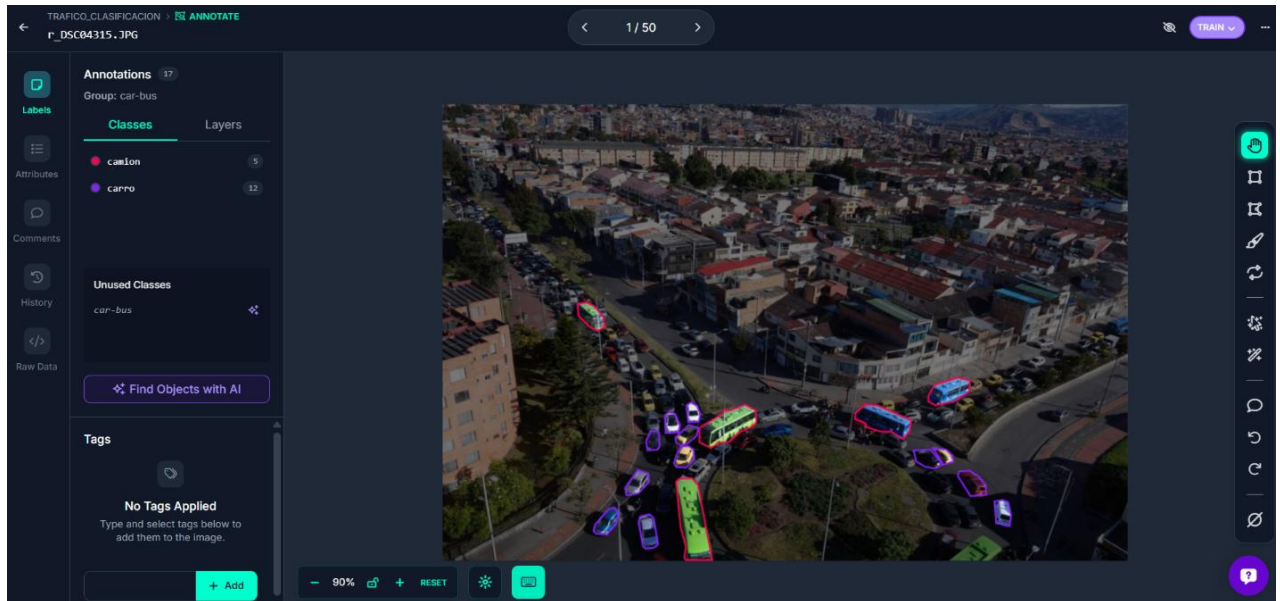


Figura 9. Clasificación de cada imagen en la herramienta Roboflow.

A continuación, se presenta un fragmento de la data obtenida del proceso de clasificación de transporte ligero y pesado en donde se expone la cantidad de carros, camiones y su respectivo estado para cada una de las 180 imágenes facilitadas de la cámara de seguridad de un lugar del sector en estudio. Las siguientes son las métricas con las que se obtuvo esta información:

- Precisión: 80%
- Perdida: 0.001

El fragmento de datos refleja el resultado del modelo de clasificación aplicado a 180 imágenes de una cámara de seguridad, donde se

identificaron carros y camiones junto con su estado; el desempeño del modelo muestra una precisión del 80%, lo que indica que ocho de cada diez predicciones fueron correctas, y una pérdida de 0.001, que evidencia un entrenamiento estable con errores mínimos. En conjunto, estas métricas sugieren que el modelo es consistente y bien ajustado, aunque aún presenta un margen de mejora en la clasificación final.

CARROS	CAMIONES	ESTADO
28	9	alto
14	6	medio
19	3	medio
8	4	bajo
27	4	alto
30	4	alto
12	8	medio
23	2	medio
24	5	medio

CARROS	CAMIONES	ESTADO
42	7	alto
1	5	bajo
1	3	bajo
17	4	medio
44	4	alto
22	7	medio
46	3	alto
31	9	alto
7	4	bajo

CARROS	CAMIONES	ESTADO
3	4	bajo
17	5	medio
24	9	alto
14	5	bajo
23	7	medio
34	12	alto
11	5	bajo
11	5	bajo
10	10	medio

CARROS	CAMIONES	ESTADO
31	10	alto
20	9	medio
20	2	medio
31	7	alto
20	11	alto
2	8	bajo
0	3	bajo
28	12	alto
5	1	bajo

Tabla 9. Fragmento de clasificación de transporte ligero y pesado por cada imagen.

El estado se obtuvo teniendo en cuenta las siguientes condiciones sumando el total de carros y camiones obtenido por cada imagen por medio del clasificador construido en Roboflow (Ver anexo 6. Dataset clasificación):

- 0 a 20: bajo.
- 21 a 30: medio.
- 31 en adelante: alto.

Se diseñó un tablero de control para analizar el flujo vehicular y las métricas obtenidas son las siguientes:



Figura 10. Tablero de control para analizar el flujo vehicular.

El siguiente paso es utilizar la data de clasificación obtenida (Ver anexo 6. Dataset clasificación) en una red neuronal que permita predecir el estado del tráfico a partir del total de vehículos presente en una nueva imagen

capturada por la cámara de seguridad. La mejor opción es una red neuronal feedforward (MLP).

En una red neuronal feedforward la información fluye en una sola dirección, desde la entrada hasta la salida, sin retroalimentación ni ciclos. Se utiliza ampliamente para problemas de clasificación y regresión en datos tabulares (Ver Anexo 7. Implementación red neuronal feedforward (MLP)).

Los resultados de precisión y pérdida para el modelo son los siguientes:

- Precisión: 93%
- Pérdida: 0.09

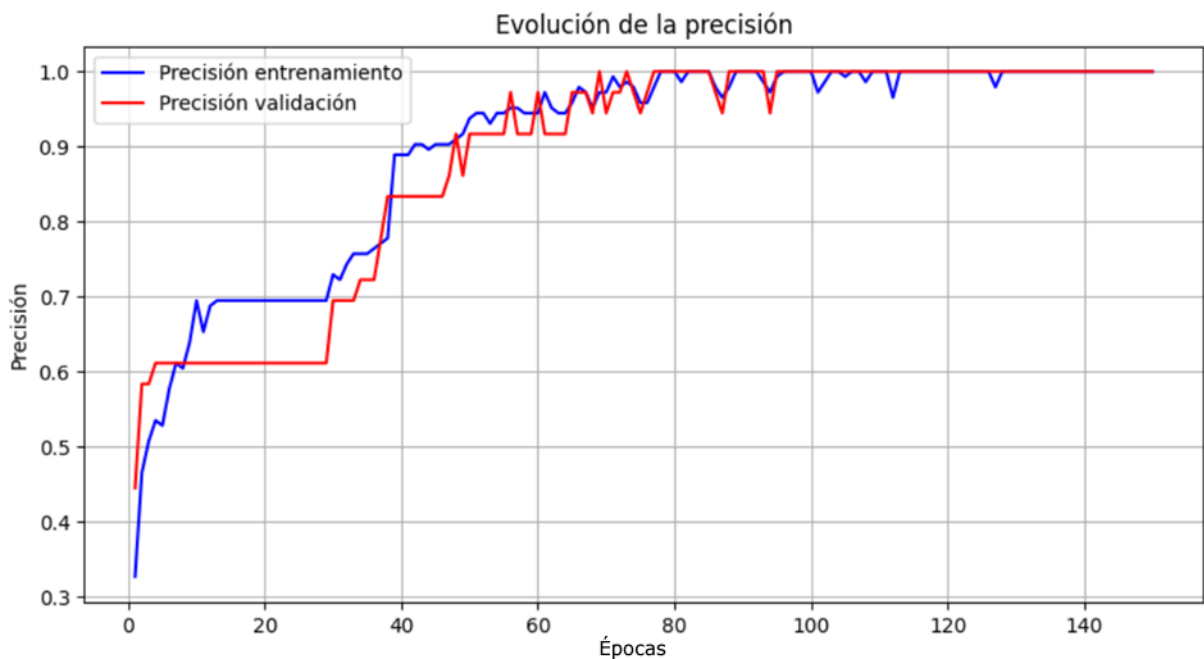


Figura 11. Evolución de la precisión.

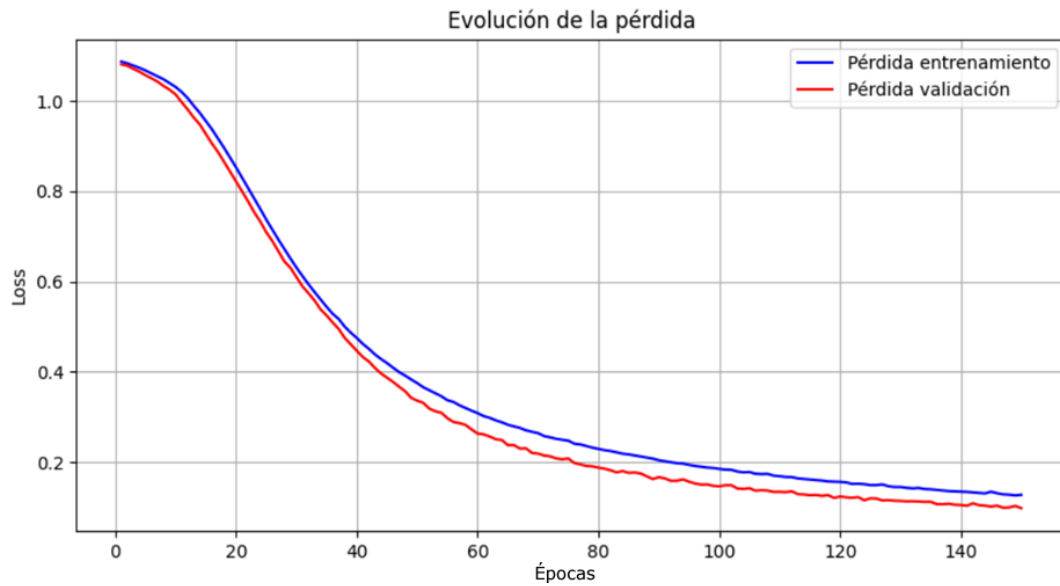


Figura 12. Evolución de la pérdida.

El uso de la red neuronal feedforward (MLP) para predecir el estado del tráfico a partir del total de vehículos detectados muestra que la precisión alcanzó el 93%, lo que indica que el sistema clasifica correctamente la gran mayoría de las nuevas imágenes, y la pérdida se situó en 0.09, reflejando un nivel de error bajo. En conjunto, estos resultados evidencian que la MLP logra generalizar mejor y ofrece predicciones más confiables sobre el estado del tráfico.

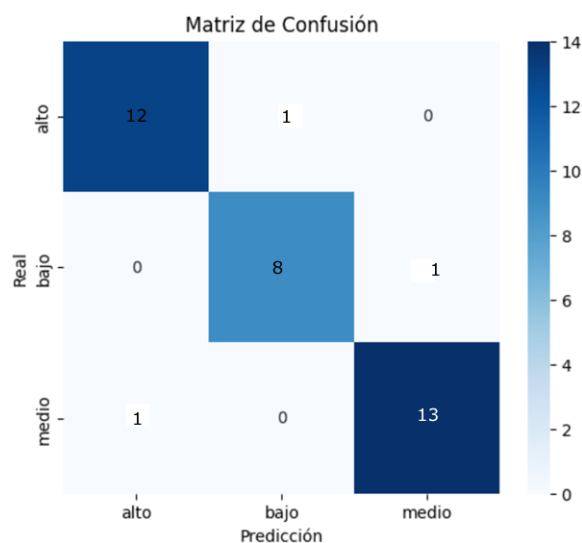


Figura 13. Matriz de confusión.

Resultados de las métricas principales a partir de la matriz de confusión:

- Precisión por clase:
Alto: $(12 / 12+0+1) = 0.92$
Bajo: $(8 / 8 + 1 + 0) = 0.89$
Medio: $(13 / 13 + 0 + 0) = 1.00$
- Recall por clase:
Alto: $(12 / 12 + 1 + 0) = 0.92$
Bajo: $(8 / 8 + 0 + 1) = 0.89$
Medio: $(13 / 13 + 1 + 0) = 0.93$
- Exactitud global:
 $(12+8+13) / (12+1+0+0+8+1+1+0+13) = 33/36 \approx 0.92$
- F1-score por clase:
Alto: $(2 \times (0.92 \times 0.92)) / (0.92 + 0.92) = 0.92$
Bajo: $(2 \times (0.89 \times 0.89)) / (0.89 + 0.89) = 0.89$
Medio: $(2 \times (1 \times 0.93)) / (1 + 0.93) = 0.96$

La matriz de confusión muestra un buen desempeño general del modelo, con la mayoría de las predicciones acertadas en cada clase. La clase "alto" tiene 12 aciertos y solo un error hacia "bajo", mientras que "medio" alcanza 13 aciertos con un único error hacia "alto". En el caso de "bajo", se observan 8 aciertos y un error hacia "medio". Esto indica que el modelo distingue bastante bien entre las tres categorías, aunque existe una ligera confusión entre "alto" y "bajo", y entre "bajo" y "medio".

Por lo anterior, los errores son pocos y no parecen concentrarse en una sola clase, lo que sugiere que el modelo mantiene un equilibrio razonable en su capacidad de clasificación. Sin embargo, sería recomendable revisar los casos en que “alto” se confunde con “bajo” y viceversa, ya que esas categorías suelen representar extremos y su confusión puede tener un impacto más significativo en la interpretación de resultados. En general, la matriz refleja un desempeño sólido, con márgenes de mejora en la separación más clara de las clases limítrofes.

El análisis de los F1-scores evidencia que el modelo mantiene un desempeño bastante equilibrado entre las tres clases, con valores de 0.92 para “alto”, 0.89 para “bajo” y 0.96 para “medio”. Esto significa que la combinación de precisión y recall es sólida, especialmente en la categoría “medio”, donde el modelo logra diferenciar con mayor claridad los casos. Aunque la clase “bajo” presenta un F1 ligeramente inferior, sigue siendo aceptable y no muestra un sesgo grave.

Para el análisis de los resultados se puede evidenciar que el primer modelo de clasificación aplicado a las imágenes de tráfico alcanzó una precisión del 80% y una pérdida de 0.001. Estos resultados muestran que el sistema fue capaz de identificar correctamente ocho de cada diez casos, con un entrenamiento estable y errores mínimos. Sin embargo, el margen de error aún era significativo para un sistema que busca apoyar la gestión del flujo vehicular, lo que evidenciaba la necesidad de mejorar la capacidad de generalización del modelo.

Con la implementación de la red neuronal feedforward (MLP), los resultados mejoraron notablemente: la precisión subió al 93% y la pérdida se situó en 0.09. Este incremento refleja que el modelo logró adaptarse mejor a los patrones de los datos y ofrecer predicciones más confiables. Aunque la pérdida es mayor que en el primer modelo, se mantiene en un

nivel bajo y aceptable, lo que indica que el sistema no está sobre ajustado y puede responder de manera más efectiva a nuevas imágenes.

Finalmente, la matriz de confusión presenta un buen comportamiento del modelo MLP, ya que todas las instancias fueron clasificadas en sus respectivas categorías de tráfico (alto, medio y bajo). Este resultado respalda la idea de que la red neuronal es una herramienta adecuada para este tipo de problemas, pues garantiza un análisis más preciso del flujo vehicular y ofrece una base sólida para la toma de decisiones en la gestión del tránsito del sector estudiado.

5. Conclusiones y trabajo futuro

5.1 Conclusiones

Se destaca la importancia de abordar los retos de la movilidad urbana en Bogotá, especialmente en la zona de Centro Mayor, donde la congestión vehicular es significativa. Se subraya la relevancia del Sistema Integrado de Información sobre Movilidad Urbana Regional (SIMUR) como herramienta clave para la gestión y el intercambio de datos entre entidades, usuarios y componentes de la movilidad. La propuesta consiste en el diseño y desarrollo de un complemento para el SIMUR, basado en modelos de percepción computacional y aprendizaje automático, con el objetivo de anticipar y gestionar de manera más eficiente las dinámicas del tráfico. Este complemento permitirá generar alertas tempranas y análisis predictivos sobre los tiempos de desplazamiento y la situación del tráfico vehicular, optimizando así la gestión de la movilidad en la localidad de Tunjuelito.

Como contribución principal, el proyecto proporcionará a las autoridades locales y a la comunidad un sistema inteligente de apoyo a la toma de decisiones, mejorando la planificación vial, fomentando la participación

ciudadana y avanzando hacia una movilidad más sostenible y adaptada a las necesidades reales del territorio.

El complemento desarrollado para el sector de Centro Mayor contribuye significativamente a reducir los tiempos de desplazamiento, identificar patrones de congestión y clasificar tipos de vehículos, lo que mejora la eficiencia operativa y la calidad de vida de los ciudadanos.

La arquitectura modular e interoperable del complemento permite su adaptación a otras zonas de Bogotá o ciudades con retos similares de movilidad, abriendo la puerta a una red de sistemas inteligentes de tráfico urbano.

Al incorporar tecnologías de visión artificial y aprendizaje automático, el sistema promueve una movilidad más eficiente, segura y ambientalmente responsable, alineada con los objetivos de desarrollo urbano sostenible.

El modelo demuestra un rendimiento sólido y equilibrado, con una exactitud global cercana al 92% y F1-scores consistentes en todas las clases. La categoría “medio” es la mejor diferenciada, mientras que “alto” y “bajo” presentan ligeras confusiones entre sí, lo que sugiere que el sistema funciona bien pero aún puede mejorar en la separación de los extremos. En términos prácticos, los errores son pocos y no comprometen la confiabilidad general del modelo, aunque convendría reforzar la precisión en los casos límite para alcanzar un desempeño más robusto y minimizar interpretaciones erróneas en escenarios críticos.

Las conclusiones derivadas del uso de la red neuronal feedforward (MLP) muestran que este enfoque es altamente efectivo para predecir el estado del tráfico a partir de la clasificación de vehículos en imágenes de

seguridad. La precisión del 93% evidencia que el modelo logra identificar correctamente la gran mayoría de los casos, lo cual representa una mejora sustancial respecto al modelo inicial. Aunque la pérdida de 0.09 es mayor que la obtenida en la primera etapa, sigue siendo baja y refleja un nivel de error aceptable dentro de un sistema de predicción aplicado a datos reales y variables.

En conjunto, estos resultados permiten afirmar que la MLP constituye una solución robusta y confiable para el análisis del tráfico en entornos urbanos, ofreciendo un balance adecuado entre exactitud y capacidad de generalización. La mejora en la precisión indica que el modelo puede ser utilizado como herramienta de apoyo en la toma de decisiones relacionadas con movilidad y seguridad, mientras que la pérdida moderada sugiere que aún existen oportunidades de optimización mediante ajustes en los hiperparámetros o la incorporación de un dataset más diverso que capture mejor las condiciones cambiantes del entorno.

En síntesis, los resultados obtenidos demuestran que la red neuronal MLP logró ofrecer una mayor precisión y una clasificación más confiable de las imágenes de tráfico. Este desempeño confirma la pertinencia de su implementación en la gestión del flujo vehicular, ya que proporciona un análisis sólido y adaptable que puede servir de apoyo en la toma de decisiones para mejorar la movilidad en el sector estudiado.

5.2 Líneas de trabajo futuro

Las líneas de trabajo futuro para proyectos de movilidad urbana en Bogotá, especialmente en la zona de Centro Mayor, incluyen el fortalecimiento de sistemas inteligentes de gestión de movilidad, la automatización y análisis predictivo del tráfico, la participación ciudadana y diagnóstico colaborativo, la promoción de una movilidad sostenible y

multimodal, la gestión de la congestión vehicular y la integración de nuevas tecnologías. Estas iniciativas buscan optimizar la planificación vial, anticipar dinámicas de tráfico, mejorar la calidad de vida y la eficiencia del desplazamiento, y avanzar hacia una movilidad urbana más sostenible e inteligente.

Incorporar dispositivos de medición de decibeles en puntos estratégicos del sector Centro Mayor para correlacionar niveles de ruido con el flujo vehicular y generar alertas ambientales.

Migrar del enfoque KNN a arquitecturas más avanzadas como CNNs (Redes Neuronales Convolucionales) para mejorar la precisión en la clasificación de imágenes de tráfico y detección de incidentes.

Crear una interfaz RESTful que permita a terceros acceder a datos procesados del sistema, fomentar la transparencia y habilitar el desarrollo de aplicaciones complementarias.

Implementar técnicas de paralelización y procesamiento distribuido para reducir la latencia en la generación de alertas y mejorar la escalabilidad del sistema.

Diseñar métricas que midan cómo la implementación del complemento SIMUR mejora la calidad de vida, reduce emisiones y promueve una movilidad más sostenible en la zona intervenida.

Para complementar las líneas de trabajo futuro, es importante destacar la necesidad de articular estas iniciativas con políticas públicas y planes de movilidad ya existentes en Bogotá. La integración con programas como el Plan Maestro de Movilidad y el uso de plataformas oficiales de gestión urbana permitiría que los desarrollos tecnológicos no se queden en proyectos aislados, sino que se conviertan en herramientas efectivas para la toma de decisiones. Además, la colaboración con universidades y

centros de investigación fortalecería la capacidad de innovación y garantizaría que los avances tengan un sustento académico y técnico sólido.

Otro aspecto clave es la formación de talento humano especializado en inteligencia artificial, ciencia de datos y ciberseguridad aplicado al transporte. La creación de semilleros y programas de capacitación permitiría que estudiantes y profesionales se involucren en el diseño de soluciones, asegurando la sostenibilidad del proyecto en el tiempo. De esta manera, se fomenta una comunidad de práctica que no solo desarrolla tecnología, sino que también promueve conciencia ciudadana sobre la importancia de la movilidad sostenible y el uso responsable de los datos.

Finalmente, la implementación de estas líneas de trabajo debe considerar un enfoque participativo y transparente. La interfaz RESTful propuesta, junto con métricas claras de impacto social y ambiental, puede convertirse en un canal de comunicación abierto entre la ciudadanía, las autoridades y los desarrolladores. Esto no solo fortalecería la confianza en el sistema, sino que también permitiría que terceros generen aplicaciones complementarias, ampliando el alcance de las soluciones y potenciando la innovación colaborativa en torno a la movilidad urbana.

Referencias bibliográficas

- García Serrano, Alberto. (2018). Inteligencia artificial (2ª ed.). Editorial RC libros.
- Gómez, Juan Carlos. (2017). Fundamentos del procesamiento de imágenes digitales (1ª ed.). Editorial McGraw-Hill.
- González, Rafael. (2018). Digital Image Processing (1ª ed.). Editorial Pearson.
- Lahoz Beltra, Rafael. (2012). Pensando en máquinas que piensan (1ª ed.). Editorial RBA coleccionables.
- Pajares, Gonzalo. (2015). Procesamiento digital de imágenes (1ª ed.). Editorial Alfaomega.
- Richard Szeliski. (2022). Computer Vision: Algorithms and Applications (1ª ed.). Editorial Springer.
- Scriven, Michael. (1964). Controversia sobre mentes y máquinas (1ª ed.). Editorial Orbis.
- Rondón Fuentes, Carmen Alicia. (2021). Movilidad urbana en Bogotá. Universidad industrial de Santander.

Vargas Montero, Fernando. (2022). Análisis de datos de accidentalidad vial de la ciudad de Bogotá a partir de datos abiertos y redes sociales. Universidad nacional de Colombia.

Biblioteca digital de SIMUR. 2006. <https://www.simur.gov.co/datos-abiertos>.

Sacone, S. (2020). IEEE Transactions on Intelligent Transportation System.

Zhang et , Li & Chen (2022). Traffic flow prediction using LSTM networks.

Ma, X., & Wang, Y. (2015). Transportation Research Part C: Emerging Technologies.

Rodríguez, Jean (2020). Aplicaciones de aprendizaje automático para la predicción de tráfico urbano en Costa Rica. Revista tecnología en marcha.

López, Marcel (2019). Aplicación de algoritmos de clasificación vehicular para la optimización del tránsito urbano en la Ciudad de Guatemala.

Gómez, J. A (2018). Clasificación vehicular mediante algoritmos KNN en entornos urbanos. Revista DYNA.

Guodong Yin, Cheng Wang (2020). IEEE Transactions on Intelligent Transportation Systems.

Mauricio Claver (2022). Limitaciones tecnológicas en la gestión inteligente del tráfico urbano: el caso de Bogotá. Revista de Ingeniería (2ª ed.).

Matías Corman (2023). Pérez, D., & Secretaría Distrital de Movilidad de Bogotá. (2022). Integración de inteligencia artificial en sistemas de movilidad urbana: retos y oportunidades del SIMUR. Revista de transporte y Territorio.

MIT Technology Review. (2018, diciembre 4). City Brain: AI for urban traffic management. MIT Technology Review. <https://www.technologyreview.com/2018/12/04/103781/city-brain-ai-for-urban-traffic-management>

Institute of Transportation Engineers. (2019). Traffic optimization with ATSAAC. Recuperado de <https://www.ite.org/publications/ite-journal>

European Commission. (2020). Smart mobility platform in Barcelona. Recuperado de https://transport.ec.europa.eu/transport-themes/mobility-strategy/smart-mobility_en

European Commission. (2021). Visions Project Report: KNN-based Traffic Classification. EU Research Programme. Recuperado de <https://cordis.europa.eu/project/id/955317>

Universidad Nacional de Colombia. (2022). Aplicación de KNN en movilidad urbana. Revista DYNA, 89(220), 45–56. Recuperado de <https://revistas.unal.edu.co/index.php/dyna>

- IEEE Transactions on Intelligent Transportation Systems. (2020). Comparative study of vehicle classification algorithms. IEEE, 21(4), 1502–1515. Recuperado de <https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=6979>
- World Economic Forum. (2021). AI and urban mobility: Reducing congestion. WEF Reports. Recuperado de <https://www.weforum.org/reports/how-mobility-shapes-inclusion-and-sustainable-growth-in-global-cities>
- Banco Interamericano de Desarrollo. (2022). Infraestructura tecnológica y movilidad en América Latina. BID Publications. Recuperado de <https://publications.iadb.org/en/la-brecha-de-infraestructura-en-america-latina-y-el-caribe>
- Electronic Frontier Foundation. (2020). Privacy concerns in smart cities. EFF Reports. Recuperado de <https://www.eff.org/deeplinks/2020/03/unchecked-smart-cities-are-surveillance-cities>
- Organisation for Economic Co-operation and Development. (2023). Urban mobility report: AI and policy integration. OECD Publishing. Recuperado de <https://www.oecd.org/cities/smart-cities-and-inclusive-growth>
- Zhang, Y., Wang, H., & Chen, L. (2021). Traffic flow prediction using LSTM networks: A case study in Shanghai. Transportation Research Part C: Emerging Technologies, 124, 102–115. Recuperado de <https://doi.org/10.1016/j.trc.2020.102115>

- Li, J., & Chen, Q. (2022). Recurrent neural networks for intelligent transportation systems: Applications and challenges. *IEEE Transactions on Intelligent Transportation Systems*, 23(5), 4210–4225. Recuperado de <https://doi.org/10.1109/TITS.2021.3056789>
- Ma, X., Tao, Z., Wang, Y., Yu, H., & Wang, Y. (2015). Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *Transportation Research Part C: Emerging Technologies*, 54, 187–197. Recuperado de <https://doi.org/10.1016/j.trc.2015.03.014>
- Secretaría de Movilidad de Bogotá. Sistema de Información de Movilidad Urbana (SIMUR) (2025). Recuperado de <https://www.movilidadbogota.gov.co/web/simur>

Anexo A. Código fuente para reducir el tamaño de las imágenes.

Es un algoritmo que recorre los archivos de imágenes JPG que se encuentran en la carpeta "transito" y tiene por objetivo reducir el tamaño de cada imagen. En total se procesaron 87 imágenes que pesaban en conjunto 287 Mb y este se redujo a 9.58 Mb.

```
from PIL import Image
import os

# Ruta de la carpeta
carpeta = "transito"

totalRow=0

# Recorrer todos los archivos en la carpeta
for archivo in os.listdir(carpeta):

    if archivo.lower().endswith(".jpg") and archivo.lower().startswith(".")
    == False :

        # Abrir imagen

        img = Image.open("transito/"+archivo)

        # Reducir dimensiones (ejemplo: 50%)
        new_size = (img.width // 3, img.height // 3)

        img_resized = img.resize(new_size)

        # Guardar con compresión (calidad 70)
        img_resized.save("img/r_"+archivo, quality=70)

        print("Archivo procesado: ", archivo)

        totalRow=totalRow + 1

print("*****")

print("Fin proceso. Rows: ",totalRow)
```

Anexo B. Clasificador de transporte ligero y pesado.

Es un algoritmo que se basa en la plataforma integral de visión artificial y permite crear y entrenar un modelo para la clasificación de tráfico, detectando vehículos de transporte ligero y pesado. A partir de una imagen indica cuantos carros y camiones se encuentran en ella.

```
# pip install ultralytics

# pip install roboflow

#importar el proyecto creado en roboflow y descargar el respectivo
dataset

from roboflow import Roboflow

rf = Roboflow(api_key="EZo0k04XysE6HDMNNXfY")

project = rf.workspace("trafico2").project("trafico_clasificacion")

version = project.version(1)

dataset = version.download("yolov11")

# Cargar el modelo YOLOv11 preentrenado

model = YOLO("yolo11n.pt")

# Entrenar el modelo con el dataset descargado "dataset.yaml"

#Tarda una hora y media en ejecutarse

results = model.train(

    data="trafico_clasificacion-1/data.yaml",

    epochs=50,          # número de épocas de entrenamiento

    imgsz=640,          # tamaño de las imágenes

    batch=16,           # tamaño del batch

    device=-1           # -1 para CPU

)
```

```
#prediccion

# Validar el modelo

metrics = model.val()

# Hacer predicciones en una imagen

results          =          model.predict(source="trafico_clasificacion-
1/valid/images/r_DSC04309.JPG", save=True, conf=0.25)

# Contar carros y camiones

carros      =      sum(1      for      box      in      results[0].boxes      if
results[0].names[int(box.cls)] == "carro")

camiones    =      sum(1      for      box      in      results[0].boxes      if
results[0].names[int(box.cls)] == "camion")

print("Carros:", carros)

print("Camiones:", camiones)
```

Anexo C. Clasificador transporte ligero y pesado incorporando el estado.

```
#por carpeta

from PIL import Image

import os

carpeta = "img"

totalRow=0

# Recorrer todos los archivos en la carpeta

lista=[]

lista.append("CARROS,CAMIONES,ESTADO")

for archivo in os.listdir(carpeta):

    if archivo.lower().endswith(".jpg") and archivo.lower().startswith(".")
    == False :
```

```
# Hacer predicciones en una imagen

results = model.predict(source=archivo, save=True, conf=0.25)

# Contar carros y camiones

    carros = sum(1 for box in results[0].boxes if
results[0].names[int(box.cls)] == "carro")

    camiones = sum(1 for box in results[0].boxes if
results[0].names[int(box.cls)] == "camion")

    print(str(carros),",",str(camiones))

    estado="bajo"

    if (carros + camiones) >=20 and (carros+camiones) <=30:

        estado="medio"

    if (carros+camiones) >30:

        estado="alto"

    lista.append(str(carros)+", "+str(camiones)+", "+estado)

    print(archivo)

    totalRow=totalRow+1


print("Fin proceso: ",totalRow)
```

Anexo D. Implementación red neuronal feedforward (MLP).

```
import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import LabelEncoder, MinMaxScaler

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense
```

```
from tensorflow.keras.utils import to_categorical

# Paso 1. Cargar datos
df = pd.read_csv("dataClasificacionFeb1_total_180.csv", sep=";")

# Paso 2. Preparar X y y
X = df[['total']].values
y = df['ESTADO'].values

encoder = LabelEncoder()
y_encoded = encoder.fit_transform(y)
y_categorical = to_categorical(y_encoded)

# Paso 3. Normalizar X
scaler = MinMaxScaler()
X_scaled = scaler.fit_transform(X)

# Paso 4. Dividir dataset
X_train, X_test, y_train, y_test = train_test_split(
    X_scaled, y_categorical, test_size=0.2, random_state=42
)

# Paso 5. Definir modelo mejorado
model = Sequential([
    Dense(16, activation='relu', input_shape=(1,)), # más neuronas
    Dense(8, activation='relu'), # segunda capa oculta
    Dense(3, activation='softmax') # salida con 3 clases
])
```

```
])  
  
model.compile(optimizer='adam',          loss='categorical_crossentropy',  
metrics=['accuracy'])  
  
# Paso 6. Entrenar con más épocas  
  
history = model.fit(  
    X_train, y_train,  
    epochs=150, batch_size=8,  
    validation_data=(X_test, y_test),  
    verbose=1  
)  
  
# Paso 7. Evaluar modelo  
  
loss, accuracy = model.evaluate(X_test, y_test, verbose=0)  
  
print("Loss en test:", loss)  
  
print("Precisión en test:", accuracy)
```