

# SISTEMA AVANZADO DE DETECCIÓN Y MEDICIÓN AUTOMÁTICA DE GRIETAS EN PAVIMENTOS MEDIANTE INTELIGENCIA ARTIFICIAL, VISIÓN POR COMPUTADOR Y APRENDIZAJE PROFUNDO

Sebastián Lara-Barría<sup>1</sup>, Carlos Real-Gutierrez<sup>2</sup>,  
Irene Indacoechea-Vega<sup>3</sup>, Pablo Pascual-Muñoz<sup>4</sup>, Daniel Castro-Fresno<sup>5</sup>

<sup>1</sup> Universidad de Cantabria, Santander, España, [sebastian.larabarría@unican.es](mailto:sebastian.larabarría@unican.es)

<sup>2</sup> Universidad de Cantabria, Santander, España, [carlos.real@unican.es](mailto:carlos.real@unican.es)

<sup>3</sup> Universidad de Cantabria, Santander, España, [irune.indacoechea@unican.es](mailto:irune.indacoechea@unican.es)

<sup>4</sup> Universidad de Cantabria, Santander, España, [pablo.pascualm@unican.es](mailto:pablo.pascualm@unican.es)

<sup>5</sup> Universidad de Cantabria, Santander, España, [daniel.castro@unican.es](mailto:daniel.castro@unican.es)

## Resumen

La gestión y conservación de infraestructuras viales requiere herramientas capaces de realizar inspecciones eficientes, objetivas y escalables del estado de los pavimentos. Los métodos tradicionales basados en inspección visual manual presentan limitaciones relacionadas con subjetividad, tiempos elevados de análisis, altos costos operacionales y riesgos asociados a trabajos en vías con tráfico activo. En este trabajo se presenta LIAISON, una plataforma inteligente orientada a la automatización de inspecciones de pavimentos mediante técnicas de aprendizaje profundo y una arquitectura basada en microservicios.

La plataforma integra módulos de procesamiento de vídeo geolocalizado, detección automática de defectos superficiales, cuantificación geométrica de fisuras, almacenamiento estructurado de resultados y visualización interactiva mediante una interfaz web. El sistema permite automatizar el flujo completo de inspección, desde la carga de secuencias de vídeo hasta la generación de información procesable para sistemas de gestión de pavimentos.

La arquitectura implementada combina servicios backend, procesamiento distribuido, computación en la nube y modelos de detección basados en YOLO dentro de un flujo automatizado de inspección orientado a entornos reales de operación. La plataforma fue validada utilizando secuencias de vídeo reales obtenidas durante recorridos de inspección vial sobre infraestructura real. Los resultados obtenidos demuestran la viabilidad de integrar modelos de inteligencia artificial dentro de una solución software escalable, capaz de gestionar el ciclo completo de inspección vial de forma automatizada.

La solución propuesta representa un enfoque de ingeniería de software aplicada a la ingeniería civil, proporcionando una plataforma escalable para inspección vial automatizada y apoyo a sistemas de gestión de pavimentos basados en datos.

**Palabras Clave:** Inteligencia artificial, ingeniería de software, pavimentos, inspección vial automatizada.

## 1 Introducción

La gestión y conservación de infraestructuras viales constituye uno de los principales desafíos operacionales para administraciones y organismos responsables del mantenimiento de la red viaria. La detección temprana de deterioros superficiales en pavimentos resulta determinante para optimizar recursos, planificar intervenciones y garantizar la seguridad de los usuarios. Sin embargo, los métodos tradicionales basados en inspección visual manual continúan siendo ampliamente utilizados[1], presentando limitaciones estructurales relacionadas con subjetividad en la evaluación, tiempos elevados de análisis, altos costos operacionales y riesgos asociados a trabajos en vías con tráfico activo.

El avance de las tecnologías de visión por computador, computación en la nube e inteligencia artificial ha abierto nuevas posibilidades para automatizar procesos de inspección vial. No obstante, gran parte de los desarrollos existentes se centran exclusivamente en el diseño y evaluación de modelos de detección, sin abordar aspectos fundamentales relacionados con la integración de servicios y arquitecturas escalables basadas en microservicios[2], el procesamiento distribuido, la gestión de datos geoespaciales, la escalabilidad del sistema y la visualización operacional de resultados. Esta brecha entre el desarrollo de modelos de inteligencia artificial y su integración en plataformas funcionales orientadas a entornos reales limita el impacto práctico de dichas soluciones en contextos de gestión de infraestructuras civiles.

En este contexto, el presente trabajo aborda el problema desde una perspectiva de ingeniería de software aplicada a la ingeniería civil, priorizando la integración, interoperabilidad y escalabilidad del sistema por encima del desarrollo aislado de modelos de detección. Se presenta LIAISON, una plataforma inteligente para la automatización de inspecciones de pavimentos que combina técnicas de aprendizaje profundo con una arquitectura modular basada en microservicios, permitiendo automatizar el flujo completo de inspección desde la adquisición de datos hasta la visualización georreferenciada de resultados.

## 2 Arquitectura general de la plataforma

Esta sección describe la arquitectura software desarrollada para dar soporte al flujo completo de inspección vial automatizada. El diseño de la plataforma se articuló en torno a tres principios fundamentales: escalabilidad, para permitir el procesamiento de grandes volúmenes de datos en entornos reales; interoperabilidad, para facilitar la integración con sistemas externos de gestión vial; y modularidad, para garantizar la mantenibilidad y extensibilidad del sistema a largo plazo. Las subsecciones siguientes describen cada uno de los componentes que materializan estos principios en la arquitectura implementada.

### 2.1 Arquitectura general del sistema

La arquitectura general de LIAISON se compone de cuatro capas principales —interfaz frontend, servicios backend, motor de procesamiento de inteligencia artificial y sistema de almacenamiento de datos— cada una con funciones específicas dentro del flujo automatizado de inspección vial. La plataforma soporta la carga de secuencias de vídeo geolocalizadas obtenidas mediante drones o cámaras instaladas en vehículos, que son procesadas automáticamente por los servicios backend encargados de coordinar tareas de extracción de fotogramas, sincronización de datos GPS, inferencia mediante modelos de inteligencia artificial y almacenamiento de resultados. El módulo de procesamiento incorpora modelos de detección de objetos basados en YOLO para identificar múltiples tipos de defectos superficiales, cuyos resultados son posteriormente almacenados y

visualizados mediante una interfaz web interactiva que permite consultar deterioros detectados, métricas geométricas y localización geoespacial.

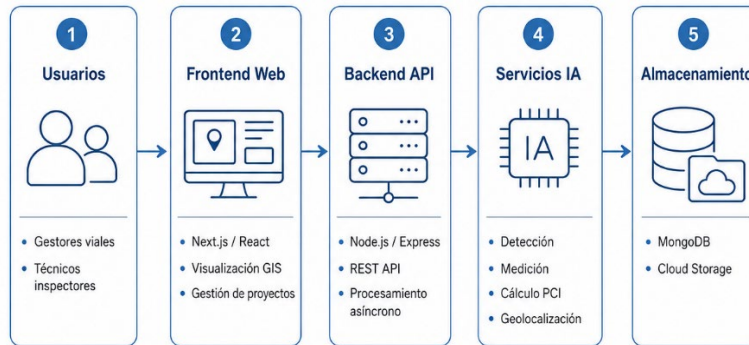


Figura 1. Arquitectura general de la plataforma LIAISON

## 2.2 Frontend web y visualización

La plataforma incorpora una interfaz web desarrollada con tecnologías frontend modernas orientadas a facilitar la interacción entre los usuarios y el sistema de inspección automatizada, permitiendo gestionar proyectos, cargar vídeos, monitorear el estado de procesamiento y visualizar los resultados obtenidos por los modelos de inteligencia artificial. La interfaz fue diseñada bajo criterios de accesibilidad y usabilidad, proporcionando herramientas para navegación geoespacial, consulta de defectos detectados y visualización de imágenes procesadas sobre mapas interactivos, lo que permite identificar con precisión la ubicación espacial de cada deterioro detectado en la infraestructura vial.

Antes de iniciar el procesamiento, el sistema permite configurar los hiperparámetros que controlan el comportamiento del flujo de análisis. Los parámetros configurables incluyen la distancia en metros entre fotogramas extraídos del vídeo, el intervalo de confianza mínimo para considerar una detección válida, los pesos relativos de cada tipo de defecto para el cálculo del índice de condición del pavimento (PCI), el umbral de detección, el área mínima de un defecto para ser incluido en el análisis, y el modelo de visión artificial a utilizar durante el procesamiento. Esta capacidad de configuración permite adaptar el comportamiento del sistema a distintos contextos de inspección sin necesidad de intervenir en el código, favoreciendo su uso en entornos operacionales con requisitos variables. Adicionalmente, la plataforma permite filtrar resultados según tipo de defecto, nivel de severidad y ubicación geográfica, facilitando procesos de análisis y priorización de mantenimiento.



Figura 2. Interfaz web de visualización y gestión de inspecciones

### 2.3 Backend y gestión de servicios

El backend de la plataforma fue desarrollado utilizando una arquitectura basada en servicios REST que actúa como núcleo de orquestación del sistema, gestionando la autenticación de usuarios, el procesamiento de solicitudes, el almacenamiento de resultados y la ejecución de las tareas asociadas a la inspección automatizada. Una función central de este módulo es la recepción y transmisión de los hiperparámetros de procesamiento configurados por el usuario en el frontend —distancia entre fotogramas, umbral de confianza, pesos de defectos, umbral de detección, tamaño mínimo y modelo de IA seleccionado— hacia el contenedor de inteligencia artificial, garantizando que cada proceso de inferencia se ejecute bajo las condiciones exactas definidas por el operador para esa inspección concreta. Esta mediación entre la configuración del usuario y el motor de procesamiento constituye uno de los flujos de integración más relevantes de la arquitectura, ya que desacopla la capa de interacción del usuario de la lógica de ejecución del modelo.

La comunicación entre servicios se realiza mediante intercambio de solicitudes HTTP y procesamiento asíncrono de tareas, lo que permite desacoplar las operaciones intensivas en cómputo del flujo principal de interacción con los usuarios y facilita la ejecución paralela de múltiples procesos de inspección sin degradar la capacidad de respuesta del sistema. El almacenamiento de información se realiza mediante una base de datos orientada a documentos, permitiendo gestionar resultados de detección, metadatos geoespaciales e información asociada a proyectos de inspección, con capacidad de integración con servicios cloud para almacenamiento y despliegue automatizado. Asimismo, el backend implementa mecanismos de gestión de estados asociados al procesamiento de videos que permiten monitorear las etapas de extracción de fotogramas, inferencia, almacenamiento y generación de resultados, mejorando la trazabilidad de los procesos ejecutados y facilitando el seguimiento operativo de cada inspección. La separación modular de responsabilidades que caracteriza esta capa facilita además futuras extensiones funcionales, la incorporación de nuevos modelos de inteligencia artificial y la interoperabilidad con plataformas externas de gestión vial, herramientas GIS y servicios de análisis de infraestructura basados en datos.

## 2.4 Infraestructura distribuida y despliegue basado en contenedores

La plataforma LIAISON fue diseñada utilizando una arquitectura distribuida basada en contenedores Docker[3], donde cada componente principal de la arquitectura —frontend, backend, servicios de inteligencia artificial y procesamiento de datos— se ejecuta de forma independiente dentro de entornos aislados. Esta estrategia permite encapsular las dependencias, librerías y configuraciones específicas de cada servicio, garantizando consistencia entre entornos de desarrollo, pruebas y producción, y facilitando al mismo tiempo la escalabilidad del sistema, la simplificación de los procesos de despliegue y la mejora de la mantenibilidad global de la plataforma. Los servicios de inferencia basados en inteligencia artificial pueden escalar horizontalmente mediante múltiples instancias ejecutándose en paralelo[4,5], lo que facilita el procesamiento simultáneo de grandes volúmenes de vídeo y reduce los tiempos de respuesta, mientras que la autonomía funcional de cada módulo permite actualizar, reiniciar o reemplazar componentes específicos sin afectar el funcionamiento global del sistema, mejorando la tolerancia a fallos y simplificando la integración futura de nuevos servicios. El despliegue de la plataforma se realiza sobre infraestructura cloud utilizando servicios de orquestación de contenedores [6], almacenamiento distribuido y monitoreo de recursos, lo que permite adaptar dinámicamente la capacidad computacional según la demanda de procesamiento y favorece la implementación de sistemas de inspección vial a gran escala con un uso optimizado de recursos.

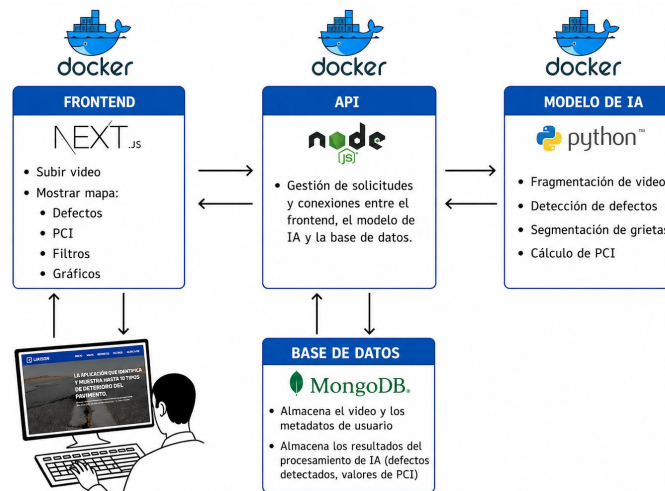


Figura 3. Arquitectura distribuida basada en contenedores Docker

## 2.5 Procesamiento de vídeo y geolocalización

El procesamiento de información se inicia con la carga de secuencias de vídeo geolocalizadas capturadas durante recorridos de inspección vial, a partir de las cuales el sistema extrae automáticamente fotogramas a los intervalos configurados por el usuario, reduciendo la redundancia temporal y optimizando el procesamiento posterior mediante inteligencia artificial. De forma paralela, la plataforma incorpora mecanismos de extracción y sincronización de la información GPS embebida en los vídeos, permitiendo asociar cada fotograma procesado con coordenadas geográficas específicas y posibilitando así la localización espacial de los defectos detectados y su visualización sobre mapas interactivos [7]. Antes de la inferencia, se aplican además procesos de filtrado y validación de imágenes orientados a descartar fotogramas borrosos, redundantes o con baja calidad visual, etapas que contribuyen a mejorar la estabilidad del sistema

y a optimizar el rendimiento de los modelos de detección en condiciones reales de operación. La sincronización temporal entre fotogramas y datos GPS permite generar recorridos georreferenciados precisos por cada inspección, facilitando tanto la identificación espacial de deterioros como la construcción de historiales de evolución del estado superficial de los pavimentos a lo largo del tiempo. En conjunto, este flujo automatizado transforma grandes volúmenes de información visual sin procesar en datos estructurados y georreferenciados directamente integrables con sistemas de gestión de pavimentos [8], eliminando la necesidad de intervención manual en las etapas intermedias del proceso y dejando abierta la posibilidad de incorporar en el futuro fuentes de información adicionales como sensores inerciales, datos LiDAR o sistemas de posicionamiento avanzados.

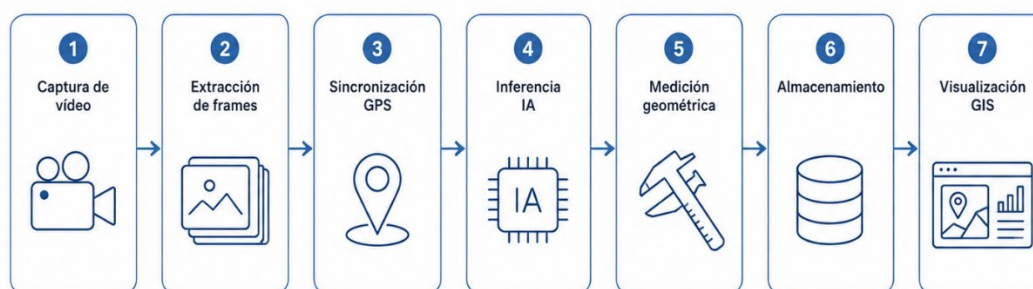


Figura 4. Flujo automatizado de procesamiento de vídeo y geolocalización

## 2.6 Stack tecnológico

La implementación de la plataforma LIAISON se apoya en un conjunto de tecnologías seleccionadas en función de su madurez, compatibilidad con arquitecturas basadas en microservicios y adecuación a los requisitos de cada capa del sistema.

El frontend fue desarrollado con **Next.js**, un framework basado en React que permite construir interfaces web modernas con renderizado híbrido, enrutamiento integrado y optimización de rendimiento en el cliente. Esta elección facilita tanto la gestión de proyectos y carga de vídeos como la visualización interactiva de resultados georreferenciados sobre mapas.

El backend y la capa de orquestación de servicios fueron implementados con **Node.js**, un entorno de ejecución JavaScript orientado a eventos que resulta especialmente adecuado para gestionar múltiples solicitudes concurrentes y coordinar la comunicación asíncrona entre el frontend, el motor de inteligencia artificial y la base de datos mediante una API REST.

El módulo de procesamiento de inteligencia artificial fue desarrollado en **Python**, aprovechando su ecosistema de librerías especializadas en visión por computador y aprendizaje profundo. Este contenedor es el responsable de ejecutar los modelos de detección basados en YOLO [9], realizar la segmentación de fisuras, calcular métricas geométricas y estimar el índice de condición del pavimento (PCI).

El almacenamiento de información se gestiona mediante **MongoDB**, una base de datos orientada a documentos que permite persistir de forma flexible los resultados de detección, los metadatos

geoespaciales asociados a cada fotograma y la información de configuración de cada inspección, sin imponer esquemas rígidos que dificulten la evolución del modelo de datos.

Finalmente, todos los servicios descritos se encapsulan en contenedores **Docker** independientes [3], garantizando la portabilidad entre entornos, la reproducibilidad de los despliegues y la capacidad de escalar cada componente de forma autónoma según la demanda operacional.

### 3 Procesamiento inteligente de defectos

La detección automática de defectos se realiza mediante modelos de detección de objetos basados en la arquitectura YOLO (You Only Look Once) [9,10], ampliamente utilizada en aplicaciones de visión por computador orientadas a procesamiento en tiempo real. El modelo fue entrenado para identificar múltiples tipos de deterioros superficiales presentes en pavimentos asfálticos, incluyendo fisuras, baches, roderas, desgaste superficial y parches, procesando automáticamente los fotogramas extraídos desde las secuencias de vídeo y generando predicciones asociadas a localización, categoría y nivel de confianza de cada defecto detectado. Las detecciones son posteriormente asociadas a las coordenadas geográficas obtenidas durante el proceso de geolocalización, permitiendo localizar espacialmente cada deterioro identificado dentro del recorrido de inspección.

Con el objetivo de mejorar la robustez del modelo frente a condiciones reales de operación, se incorporaron técnicas de aumento de datos [11,12] durante el entrenamiento, incluyendo transformaciones geométricas, variaciones de iluminación y modificaciones de escala. Estas estrategias permitieron mejorar la capacidad de generalización del sistema frente a la heterogeneidad de escenarios presentes en inspecciones viales reales, donde factores como cambios de iluminación, variaciones en la calidad del pavimento o diferencias en la altura de captura pueden afectar significativamente el rendimiento de los modelos [1]. La integración del módulo de detección dentro de la arquitectura distribuida de la plataforma permite además ejecutar inferencias de manera paralela sobre múltiples fotogramas, facilitando el procesamiento escalable de grandes volúmenes de información.



Figura 5. Ejemplo de detección automática de defectos mediante modelos YOLO

### **3.2 Clasificación y cuantificación geométrica de deterioros**

Además de identificar la presencia de defectos superficiales, la plataforma incorpora mecanismos de cuantificación geométrica orientados a caracterizar dimensionalmente el deterioro detectado, obteniendo métricas asociadas a longitud, ancho y área afectada en las fisuras [13] presentes en el pavimento. Este proceso se realiza mediante técnicas de procesamiento morfológico aplicadas sobre las regiones detectadas por los modelos de inteligencia artificial: a partir de la segmentación de las fisuras, el sistema ejecuta procesos de esqueletización y análisis geométrico [14] que permiten estimar dimensiones relevantes para la evaluación de severidad y la toma de decisiones en materia de mantenimiento vial.

La integración entre detección automática y cuantificación geométrica representa uno de los principales aportes funcionales de la plataforma, ya que permite transformar resultados visuales en información cuantificable directamente utilizable en procesos de evaluación técnica y priorización de intervenciones. Toda la información calculada es almacenada junto con los resultados de detección y las coordenadas geográficas asociadas, posibilitando consultas posteriores, visualización georreferenciada en mapas y generación de reportes estructurados vinculados a cada inspección realizada.

## **4 Resultados experimentales**

La plataforma LIAISON fue evaluada considerando tanto el desempeño de los modelos de inteligencia artificial integrados como el comportamiento general del sistema bajo escenarios reales de procesamiento de inspecciones viales. Las pruebas realizadas permitieron validar la integración entre los distintos módulos de la arquitectura —procesamiento distribuido, geolocalización, almacenamiento y visualización de resultados— así como la estabilidad del sistema bajo condiciones de carga continua, empleando para ello secuencias de vídeo reales obtenidas durante recorridos de inspección vial.

### **4.1 Entorno de implementación**

La plataforma fue desplegada sobre infraestructura cloud basada en contenedores Docker y servicios distribuidos, ejecutando cada módulo del sistema de forma independiente: frontend web, backend API, servicios de procesamiento de inteligencia artificial y almacenamiento de datos. El procesamiento de inferencias se realizó utilizando instancias con aceleración GPU, lo que permitió ejecutar los modelos de detección en paralelo sobre múltiples fotogramas extraídos desde las secuencias de vídeo geolocalizadas, mientras que la arquitectura basada en microservicios posibilitó distribuir dinámicamente las cargas de trabajo entre los distintos contenedores de procesamiento según la demanda operacional.

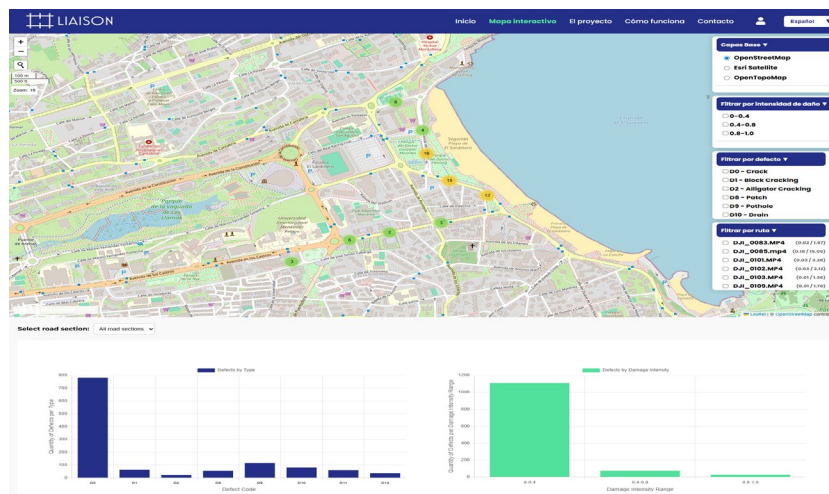


Figura 6. Plataforma LIAISON desplegada en entorno de procesamiento distribuido

## 4.2 Rendimiento del sistema

Las pruebas realizadas confirmaron que el diseño distribuido se comportó de forma estable bajo carga continua. El tiempo medio de procesamiento por fotograma se mantuvo consistente a lo largo de las sesiones de prueba, y el escalado horizontal de los servicios de inferencia respondió correctamente a los picos de demanda sin interrupciones en el resto de la plataforma. A lo largo de las evaluaciones, la plataforma mostró estabilidad durante el procesamiento continuo de múltiples secuencias de vídeo, manteniendo una comunicación consistente entre los servicios backend, los módulos de inteligencia artificial y los sistemas de almacenamiento, y respondiendo adecuadamente a los incrementos de carga mediante escalado horizontal de los servicios de inferencia. El enfoque basado en microservicios facilitó además la autonomía operativa de cada componente, permitiendo reiniciar o actualizar módulos específicos sin interrumpir el funcionamiento global de la plataforma, una característica especialmente relevante en escenarios de despliegue continuo y procesamiento masivo de información vial.

## 4.3 Visualización y explotación de resultados

La interfaz de visualización demostró ser funcional para la explotación operativa de los resultados de inspección. Durante las pruebas, se comprobó que los defectos detectados quedaban correctamente georreferenciados y eran accesibles mediante filtros por tipo, severidad y tramo, lo que facilitó la identificación de zonas prioritarias de intervención sin necesidad de procesamiento manual adicional. La integración entre el procesamiento automatizado y las herramientas de visualización facilita la generación de reportes, la realización de consultas geoespaciales y el apoyo a procesos de priorización de mantenimiento, centralizando en un único entorno software toda la información que en enfoques tradicionales se encuentra dispersa entre distintas herramientas y procedimientos manuales. Esta centralización representa una ventaja operacional significativa, ya que reduce la fricción entre las fases de adquisición, análisis y toma de decisiones en la gestión de infraestructuras viales.

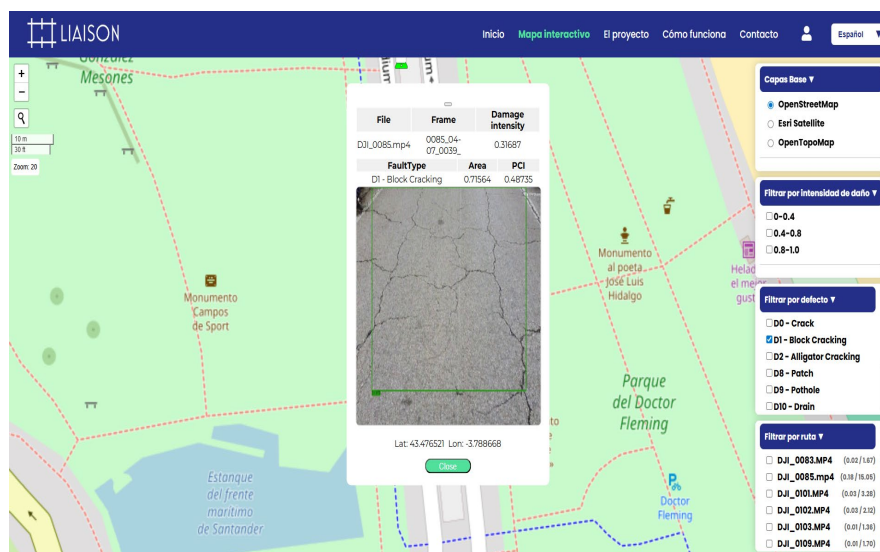


Figura 7. Visualización georreferenciada de defectos detectados en la plataforma

## 5 Discusión

Los resultados obtenidos demuestran que la adopción de principios de ingeniería de software — desacoplamiento, escalabilidad horizontal y autonomía de componentes— resulta tan determinante como la calidad del modelo de detección para que una plataforma de este tipo opere de forma fiable en entornos reales. Los resultados refuerzan la tesis de que la escalabilidad y la integración de servicios son condicionantes tan críticos como la precisión del modelo para el despliegue real de plataformas de inspección automatizada, un aspecto que la literatura existente aborda de forma incompleta [2,4,5].

La incorporación de métricas geométricas asociadas a fisuras representa un avance relevante respecto a sistemas que únicamente realizan identificación visual de deterioros [15], ya que permite transformar los resultados del modelo en información cuantificable directamente aplicable a procesos de evaluación técnica. En este sentido, los resultados experimentales muestran que la calidad y consistencia del conjunto de datos de entrenamiento influyen significativamente en el desempeño del sistema [9,10] el proceso de refinamiento y consolidación de las categorías de defectos durante el entrenamiento resultó determinante para mejorar la estabilidad del modelo y reducir las ambigüedades entre clases, lo que subraya la importancia de una gestión rigurosa de los datos dentro del ciclo de desarrollo de este tipo de plataformas.

Desde una perspectiva de ingeniería de software aplicada a la ingeniería civil, el trabajo demuestra que el valor de estas soluciones no reside únicamente en la precisión de los modelos de detección, sino en la capacidad de integrarlos dentro de sistemas funcionales completos, capaces de operar en entornos reales de infraestructura vial con criterios de escalabilidad, trazabilidad y usabilidad. La arquitectura desarrollada facilita además futuras extensiones funcionales, la incorporación de nuevos modelos de procesamiento y la adaptación de la plataforma a distintos escenarios de inspección automatizada, consolidándola como una base técnica sólida para el desarrollo de soluciones de gestión vial inteligente basadas en datos.

## 6 Conclusiones

En este trabajo se presentó LIAISON, una plataforma inteligente para la automatización de inspecciones viales que integra técnicas de aprendizaje profundo, procesamiento geoespacial y una arquitectura software basada en microservicios. La solución desarrollada permite automatizar el flujo completo de inspección de pavimentos, desde la carga y procesamiento de secuencias de vídeo geolocalizadas hasta la detección, cuantificación y visualización de defectos superficiales a través de una interfaz web interactiva con parámetros de procesamiento configurables por el usuario.

La arquitectura distribuida implementada mediante contenedores Docker permitió desacoplar los servicios de procesamiento, mejorar la escalabilidad del sistema y facilitar el despliegue de la plataforma sobre infraestructura cloud, mientras que la integración de modelos basados en YOLO permitió ejecutar tareas de detección multiclase bajo condiciones reales de operación con un comportamiento robusto y estable. En conjunto, los resultados obtenidos demuestran la viabilidad de aplicar principios de ingeniería de software al desarrollo de plataformas inteligentes orientadas a la gestión de infraestructuras civiles, y ponen de manifiesto que la combinación entre procesamiento distribuido, inteligencia artificial y visualización geoespacial permite generar soluciones escalables capaces de apoyar procesos de mantenimiento vial basados en datos.

Como trabajo futuro, se plantea incorporar técnicas de segmentación semántica, análisis predictivo del deterioro y generación automatizada de indicadores de condición de pavimento, además de fortalecer la integración de la plataforma con sistemas de gestión vial y herramientas de planificación de mantenimiento, con el objetivo de consolidar LIAISON como una solución de referencia en el ámbito de la inspección vial inteligente.

## 7 Agradecimientos

Los autores agradecen la financiación otorgada por el programa marco de investigación e innovación Horizonte Europa de la Unión Europea en virtud del acuerdo de subvención No. 101103698 (LIAISON).

## Referencias

- [1] Gong, H., Liu, L., Liang, H., Zhou, Y., & Cong, L. (2024). A State-of-the-Art Survey of Deep Learning Models for Automated Pavement Crack Segmentation. *International Journal of Transportation Science and Technology*, 13, 44–57.
- [2] Rodríguez-Sánchez, M. C., Torrijos-Casco, F., & Borromeo, S. (2021). Evaluating Service-Oriented and Microservice Architecture Patterns to Deploy eHealth Applications in Cloud Computing Environment. *Applied Sciences*, 11(10), 4350. <https://doi.org/10.3390/app11104350>
- [3] Merkel, D. (2014). Docker: Lightweight Linux Containers for Consistent Development and Deployment. *Linux Journal*, 239.
- [4] Wan, X., Guan, X., Wang, T., Bai, G., & Choi, B.-Y. (2018). Application Deployment Using Microservice and Docker Containers: Framework and Optimization. *Journal of Network and Computer Applications*, 119, 97–109. <https://doi.org/10.1016/j.jnca.2018.07.003>
- [5] Baresi, L., Quattrocchi, G., & Tamburri, D. A. (2022). Microservice Architecture Practices and Experience: A Focused Look on Docker Configuration Files. arXiv preprint arXiv:2212.03107.
- [6] Acharya, S., & Suthar, M. (2022). Docker Container Orchestration Management: A Review. *Proceedings of the International Conference on Intelligent Vision and Computing (ICIVC)*. Springer. [https://doi.org/10.1007/978-981-19-1592-7\\_55](https://doi.org/10.1007/978-981-19-1592-7_55)
- [7] Tsaimou, C., Kalfas, I., & Mitoulis, S. (2024). Mapping Cracks on Port Concrete Pavements by Analyzing Structural Health Monitoring Metadata with Computer Vision-Based Techniques. *Complex Engineering Systems*, 4, 28. <https://doi.org/10.20517/ces.2024.28>
- [8] Perda, F., Dell'Amico, A., & Ferrari, V. (2025). Automated Detection and Mapping of Pavement Cracks from Videos for Road Inspections. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, X-1/W2-2025, 99–106. <https://doi.org/10.5194/isprs-annals-X-1-W2-2025-99-2025>
- [9] Jocher, G., Chaurasia, A., & Qiu, J. (2023). Ultralytics YOLOv8 [Software]. GitHub. <https://github.com/ultralytics/ultralytics>
- [10] Wang, C.-Y., Bochkovskiy, A., & Liao, H.-Y. M. (2023). YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7464–7475.
- [11] Li, Y., et al. (2024). A Pavement Crack Detection and Evaluation Framework for a UAV Inspection System Based on Deep Learning. *Applied Sciences*, 14(3), 1157. <https://doi.org/10.3390/app14031157>
- [12] Maeda, H., Sekimoto, Y., Seto, T., Kashiya, T., & Omata, H. (2018). Road Damage Detection and Classification Using Deep Neural Networks with Smartphone Images. *Computer-Aided Civil and Infrastructure Engineering*, 33(12), 1127–1141. <https://doi.org/10.1111/mice.12387>
- [13] Qiu, S., Wang, W., Wang, S., & Wang, K. C. P. (2017). Methodology for Accurate AASHTO PP67-10-Based Cracking Quantification Using 1-mm 3D Pavement Images. *Journal of Computing in Civil Engineering*, 31(2), 04016056. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000624](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000624)
- [14] Ibragimov, E., Kim, Y., Lee, J. H., Cho, J., & Lee, J.-J. (2024). Automated Pavement Condition Index Assessment with Deep Learning and Image Analysis: An End-to-End Approach. *Sensors*, 24(7), 2333. <https://doi.org/10.3390/s24072333>
- [15] Guo, F., Qian, Y., Liu, J., & Yu, H. (2023). Pavement Crack Detection Based on Transformer Network. *Automation in Construction*, 145, 104646. <https://doi.org/10.1016/j.autcon.2022.104646>