











Sistema inteligente de recomendación y detección visual para el manejo de cultivos

Intelligent recommendation and visual detection system for crop management

Ian Thompson¹, Jason Arenales², Julio Fuentes³,
Edward Quintero⁴, Vladimir Villarreal⁵

Thompson, I; Arenales, J; Fuentes, J; Quintero, E; Villarreal, V. Sistema inteligente de recomendación y detección visual para el manejo de cultivos. *Tecnología en Marcha*. Vol. 39 N° especial. Junio, 2026. VIII Congreso Internacional en Inteligencia Ambiental, Ingeniería de Software, Salud Electrónica y Móvil (AmITIC). Pág. 99-107.

 <https://doi.org/10.18845/tm.v39i7.8747>

- 1 Universidad Tecnológica de Panamá, Panamá.
 ian.thompson@utp.ac.pa
 <https://orcid.org/0009-0009-6854-4390>
- 2 Universidad Tecnológica de Panamá, Panamá.
 jason.arenales@utp.ac.pa
 <https://orcid.org/0009-0008-5535-8161>
- 3 Universidad Tecnológica de Panamá, Panamá.
 julio.fuentes1@utp.ac.pa
 <https://orcid.org/0009-0007-7324-9957>
- 4 Universidad Tecnológica de Panamá, Panamá.
 edward.quintero@utp.ac.pa
 <https://orcid.org/0009-0008-9626-7258>
- 5 Grupo de Investigación en Tecnologías Computacionales Emergentes, Panamá.
 vladimir.villarreal@utp.ac.pa
 <https://orcid.org/0000-0003-4678-5977>



Palabras clave

Inteligencia artificial; modelos de lenguaje de gran escala (LLM); visión artificial; manejo de cultivos; agricultura de precisión.

Resumen

Este trabajo presenta el desarrollo de un sistema inteligente que combina visión artificial y un modelo de lenguaje (LLM) para asistir en el manejo de cultivos. Su objetivo es apoyar a los agricultores en la toma de decisiones preventivas y correctivas mediante el análisis de imágenes y texto. Frente a desafíos como el cambio climático, plagas y escasez de recursos, la herramienta busca identificar cultivos y enfermedades, ofreciendo recomendaciones precisas y accesibles. Con ello, se pretende reducir pérdidas agrícolas, mejorar la productividad y brindar asesoría en regiones con acceso limitado a expertos.

Keywords

Artificial intelligence; computer vision; crop management; Large Language Models (LLM); precision agriculture.

Abstract

This work presents the development of an intelligent system that combines computer vision and a large language model (LLM) to assist in crop management. Its goal is to support farmers in making preventive and corrective decisions through the analysis of images and text. In the face of challenges such as climate change, pests, and resource scarcity, the tool aims to identify crops and diseases, providing accurate and accessible recommendations. The system seeks to reduce agricultural losses, improve productivity, and offer guidance in areas with limited access to experts.

Introducción

El presente trabajo aborda el desarrollo de un sistema basado en inteligencia artificial que integra un modelo de lenguaje de gran escala (LLM) y visión artificial para asistir en el manejo avanzado de cultivos. La propuesta busca responder a la necesidad urgente de herramientas tecnológicas que apoyen a los productores agrícolas en la toma de decisiones, tanto a nivel preventivo como correctivo, mediante el análisis de datos visuales y textuales.

Actualmente, la agricultura enfrenta desafíos críticos como el cambio climático, la expansión urbana, la reducción de tierras cultivables y la escasez de recursos naturales, lo que, junto con plagas y enfermedades, genera pérdidas significativas en cantidad y calidad de las cosechas [1]. Estas pérdidas no solo afectan la economía de los productores, sino que también comprometen la seguridad alimentaria global. Esto resalta la importancia de contar con sistemas capaces de anticipar daños y brindar asistencia directa y personalizada a quienes trabajan la tierra [2].

En este contexto, el objetivo de este trabajo es diseñar e implementar un sistema que combine visión artificial y procesamiento de lenguaje natural para identificar cultivos y posibles afecciones, generando recomendaciones precisas y adaptadas a cada situación [3], [4], [6]. Esta herramienta busca mejorar la productividad agrícola, reducir pérdidas y ofrecer asesoría accesible, especialmente en zonas con acceso limitado a expertos.

Planteamiento del problema

Muchos productores enfrentan dificultades significativas en el manejo eficiente de sus cultivos debido a la falta de acceso a tecnologías modernas y sistemas de apoyo en la toma de decisiones [9]. Factores como el desconocimiento de enfermedades en las plantas [12], prácticas agrícolas inadecuadas y la limitada interpretación de datos agronómicos afectan directamente el rendimiento y la calidad de las cosechas. A pesar del crecimiento de las tecnologías inteligentes en otras áreas, el sector agrícola local aún depende, en gran parte, de métodos tradicionales, lo que limita la capacidad de anticiparse a problemas y de optimizar recursos [9], [11]. Esta situación se ve agravada por la variabilidad climática y la escasez de asistencia técnica continua en zonas rurales. “La adopción de herramientas tecnológicas aún es escasa en las zonas rurales, donde el desconocimiento y la falta de infraestructura impiden su implementación efectiva”.

Por tanto, se identifica la necesidad de desarrollar un sistema inteligente que integre técnicas de detección visual y recomendaciones automatizadas para apoyar a los agricultores en la identificación temprana de problemas y en la toma de decisiones informadas [7], [8]. Este tipo de solución podría contribuir significativamente a mejorar la productividad agrícola y a reducir pérdidas, beneficiando tanto a pequeños como a medianos productores de la región

Marco teórico

El desarrollo de soluciones tecnológicas en el sector agrícola ha tomado un rol protagónico dentro del paradigma de la agricultura de precisión, especialmente mediante la aplicación de inteligencia artificial (IA), procesamiento de lenguaje natural (PLN) y visión por computadora [9], [2]. Diversos estudios y proyectos recientes demuestran el creciente interés y eficacia de estas tecnologías para resolver problemas críticos como la detección temprana de enfermedades, la estimación de rendimiento y la emisión de recomendaciones agronómicas personalizadas [7], [8], [11].

Uno de los enfoques más innovadores en la clasificación automatizada de enfermedades en cultivos es el trabajo de Roumeliotis [3], quienes han integrado modelos de lenguaje de gran escala multimodales (LLMs) con redes neuronales convolucionales (CNNs). A partir de imágenes de hojas, esta combinación ha demostrado una notable eficacia en la identificación de patologías vegetales [3]. En su estudio, emplearon GPT-4o [15] junto con ResNet-50 y el conjunto de datos PlantVillage, logrando una precisión del 98.12% en la detección de enfermedades en hojas de manzano, superando a los modelos tradicionales. Estos resultados destacan el potencial de los LLMs multimodales en este ámbito [3]. Además, la comparación entre distintos modelos ha generado datos valiosos que aplicamos en nuestro proyecto, permitiéndonos aprovechar una alternativa ya probada con éxito [3].

Por su parte, Kamangir [5] introduce CMAViT, un modelo de transformadores de visión multimodal para estimar el rendimiento de cultivos a nivel de planta. Este sistema integra datos de teledetección, información climática y prácticas de manejo codificadas en texto, logrando un R^2 de 0.84 y un error absoluto medio porcentual (MAPE) de 8.22%. La incorporación de prácticas agronómicas textuales evidencia el valor del PLN en contextos agrícolas complejos y en escenarios donde la multimodalidad es esencial [5].

Todos estos trabajos demuestran un avance significativo en la integración de inteligencia artificial para la agricultura, y sientan las bases para sistemas más completos que, como el que se propone en este proyecto, combinan visión artificial y modelos de lenguaje para brindar recomendaciones contextualizadas, automáticas y de alta utilidad para productores en regiones como la provincia de Chiriquí, Panamá.

Materiales y métodos

Para este proyecto se utiliza los siguientes materiales y métodos, los cuales son esenciales para el desarrollo y funcionamiento del sistema de automatización de cultivos:

Materiales

Software

OpenCV-python para procesamiento de imágenes en tiempo real, *OpenCV-python-headless* para versión sin interfaz gráfica de OpenCV, *TensorFlow* para entrenamiento y despliegue de modelos de IA, *TensorFlow-hub* para uso de modelos preentrenados, *TensorFlow-datasets* para carga de datasets para entrenamiento, *TensorFlow-model-optimization* para optimización de modelos para dispositivos limitados, *keras-nightly* para versión en desarrollo de Keras para pruebas, *Keras-Preprocessing* para preprocesamiento de datos para redes neuronales, *object-detection* para detección de objetos en imágenes, *tf-models-official* para modelos oficiales de TensorFlow, *pillow* para manipulación de imágenes en Python, *matplotlib* para visualización de gráficos, *numpy* para cálculo numérico y matrices, *scipy* para ciencia e ingeniería computacional, *pySerial* para comunicación serial con dispositivos, *Ultralytics Yolo11* para detección de objetos con YOLOv11, *Roboflow* para anotación y preparación de datasets para visión artificial, *PostgreSQL* para base de datos relacional, *Google Colab* para entorno de ejecución para código Python en la nube, *CloudFlare* para despliegue seguro y escalable de aplicaciones web, *API ChatGPT* para uso de modelos de lenguaje natural de OpenAI, *Anaconda* para gestión de entornos y paquetes científicos en Python. HTML5: Lenguaje de marcado base de la estructura de la página, CSS / Tailwind CSS: Utiliza Tailwind CSS para el diseño visual, estilos rápidos y clases utilitarias, Node.js (Backend): backend/index.js: Servidor Node, package.json: gestor de dependencias, public/: carpeta de archivos estáticos servidos.

Hardware

En el cuadro 1 se presentan los materiales de Hardware utilizados en el proyecto de automatización de cultivos. Estos materiales son fundamentales para la implementación y el funcionamiento eficiente del sistema desarrollado:

Cuadro 1. Lista de materiales con base a sus costos.

Sensor/Componente	Precio (USD)	Sensor/Componente	Precio (USD)
DHT11 (Temp. y humedad)	2.49	Resistencias varias	1
Sensor de agua	0.55	Bombas de agua	7.5
Fotorresistencia (LDR)	0.45	Sensor ultrasónico	2.49
ESP32	7	Diodos LED	1
Fuente de voltaje	5	Relays	5
ESP32 con cámara	7.5	Display LCD 2x16	3.49

Métodos

La metodología empleada en el desarrollo del sistema propuesto se estructura en cinco fases, abarcando desde la recopilación de datos hasta el ajuste de modelos y su implementación en una plataforma interactiva.

Fase 1: Recolección y Preparación de Datos

Se utilizaron conjuntos de datos públicos provenientes de Kaggle, como *Tomato-Village* y otros relacionados con enfermedades agrícolas. Las imágenes incluyen hojas sanas, afectadas por plagas o con signos visibles de patologías. Asimismo, se incorporaron imágenes adicionales para cubrir cultivos de interés no representados en los datasets originales.

Fase 2: Entrenamiento del Modelo de Visión Artificial

Para la detección visual de cultivos y enfermedades se empleó el modelo YOLOv11 (Ultralytics). El proceso de entrenamiento se ejecutó en Google Colab, utilizando técnicas de aumento de datos para mejorar la robustez del modelo frente a variaciones visuales.

La evaluación se realizó aplicando métricas como *mean Average Precision (mAP)*, precisión y *recall*, a fin de garantizar una detección confiable y eficiente en distintos escenarios agrícolas.

Fase 3: Fine-Tuning del Modelo de Lenguaje (LLM)

Se seleccionó el modelo Phi-3.5-mini-instruct, por su eficiencia computacional y adecuación al despliegue en sistemas de recursos limitados. El modelo fue ajustado finamente mediante un proceso de *fine-tuning*, utilizando ejemplos representativos del dominio agrícola. Un par de entrenamiento típico incluye:

- *Prompt*: “¿Qué hago si mi planta de tomate tiene hojas con manchas negras?”
- *Response*: “Podría tratarse de tizón temprano. Recomendación: eliminar hojas afectadas y aplicar fungicida basado en cobre.”

Fase 4: Desarrollo de la Plataforma Interactiva

Se desarrolló una aplicación web accesible desde cualquier navegador, diseñada para que los usuarios puedan capturar o subir imágenes, las cuales se envían directamente al modelo de visión artificial encargado de la detección de enfermedades en cultivos. Además, la plataforma permite formular consultas en lenguaje natural mediante el modelo de lenguaje previamente entrenado (LLM), integrado en el sitio web, y recibir diagnósticos automáticos acompañados de recomendaciones específicas para cada caso.

Fase 5: Validación y Pruebas de Campo

La validación del sistema se llevó a cabo en zonas rurales de Chiriquí, Panamá, mediante pruebas controladas con agricultores locales. Se evaluó el desempeño del sistema en cuanto a precisión diagnóstica, relevancia de las recomendaciones y facilidad de uso.

Resultados

El sistema inteligente desarrollado demostró ser capaz de integrar con éxito visión artificial y modelos de lenguaje de gran escala para asistir a los productores agrícolas en el manejo de cultivos. A continuación, se presentan los resultados obtenidos durante las pruebas realizadas:

Precisión del Modelo de Visión Artificial (YOLOv11)

Se utilizaron dos variantes del modelo YOLOv11 (yolov11s y yolov11m) entrenadas con imágenes etiquetadas en la plataforma Roboflow. Los modelos lograron identificar correctamente enfermedades como mildiu, tizón temprano y mancha bacteriana en cultivos de tomate y lechuga. Precisión promedio (mAP@0.5) del modelo yolov11m: 92.4%

- Tiempo promedio de detección por imagen: 0.25 segundos en GPU (T4)
- Visualización en tiempo real de las detecciones con bounding boxes y etiquetas, facilitando la interpretación de los resultados.



Figura 1. Detección en Tiempo Real con Visión Artificial. Visualización del sistema en ejecución identificando una hoja de poroto mediante la cámara web del usuario. El modelo de visión artificial (YOLOv11) enmarca y etiqueta el objeto detectado, mostrando el nombre de la clase y su nivel de confianza [6], [7], [18].

Rendimiento del Modelo de Lenguaje (LLM Fine-Tuneado)

Para el desarrollo del asistente agrícola inteligente, se empleó como base el modelo Phi-3.5-mini-instruct, un modelo de lenguaje previamente entrenado para seguir instrucciones. Este modelo fue reentrenado (fine-tuneado) con un dataset agrícola personalizado, elaborado con ejemplos representativos que incluían preguntas frecuentes sobre el cuidado de cultivos, identificación de enfermedades, dosis recomendadas y condiciones óptimas de siembra.

El proceso de entrenamiento se llevó a cabo en la plataforma Google Colab, aprovechando el entorno con soporte para GPU. Se utilizó la técnica de fine-tuning completa (sin LoRA), y el dataset fue estructurado en formato de instrucciones (instruction, input, output) para que el modelo pudiera aprender a generar respuestas contextualizadas y precisas.

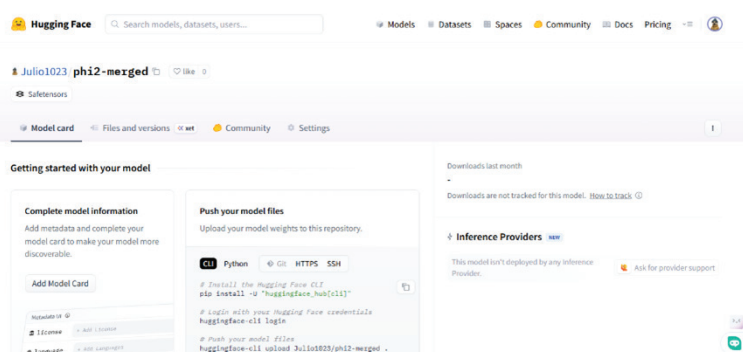


Figura 2. Modelo Ajustado Almacenado en Hugging Face. Interfaz del repositorio en Hugging Face donde se encuentra alojado el modelo fine-tuneado Julio1023/phi2-merged, permitiendo su descarga, integración y despliegue para tareas de inferencia relacionadas con agricultura [13], [14].

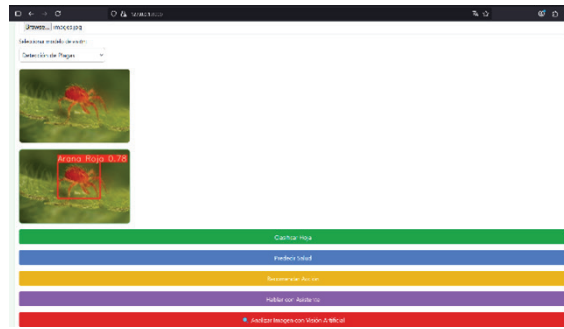


Figura 3. Subida de imágenes para análisis con un modelo basado en YOLOv11. Fragmento de la interfaz del Asistente Agrícola Inteligente donde se observa la subida de una imagen para el análisis con un modelo de detección personalizado, basado en la arquitectura YOLOv11, detectando una “Araña Roja” con un 78% de confianza[18].

Interfaz Local y Respuesta en Tiempo Real

El sistema fue ejecutado localmente desde una aplicación Flask en la computadora del usuario, permitiendo:

- Subida de imágenes para análisis con YOLOv11.

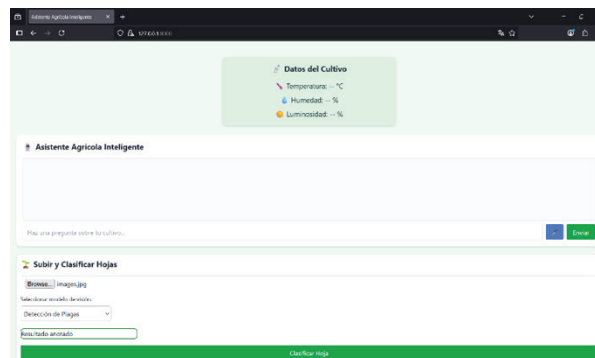


Figura 4. Envío de preguntas en texto para obtener respuestas desde el modelo de lenguaje. Fragmento de la interfaz del Asistente Agrícola Inteligente mostrando el campo de texto para el envío de preguntas y la obtención de respuestas desde el modelo de lenguaje, junto con la visualización de datos del cultivo como temperatura, humedad y luminosidad [13].

- Visualización de los resultados en tiempo real a través de una interfaz web básica servida desde localhost.

Conclusiones y/o recomendaciones (discusión)

El proyecto del “Sistema Inteligente de Recomendación y Detección Visual para el Manejo de Cultivos” se completó con éxito, ofreciendo una solución integral para huertos caseros. Integrando un microcontrolador ESP32 con sensores, visión artificial (YOLOv11) y un asistente inteligente (Phi-3.5-mini-instruct), el sistema automatiza el monitoreo, la detección de problemas y el riego. Su plataforma web permite el control remoto y la visualización de datos, empoderando a los usuarios para una gestión eficiente y productiva de sus cultivos en casa.

Agradecimientos

V.Villarreal es miembro del Sistema Nacional de Investigación de la SENACYT.

Referencias

- [1] Junaid, M. D., & Gokce, A. F. (2024). Global agricultural losses and their causes. *Bulletin of Biological and Allied Science Research*, 2024(1), 66. <https://doi.org/10.54112/bbasr.v2024i1.66>
- [2] Trejo, A., Ortiz, M., & Chávez, E. (2021). Inteligencia artificial y sistemas expertos aplicados a la agricultura de precisión. *Revista de Tecnología e Innovación Agropecuaria*, 5(2), 102–115. <https://www.redalyc.org/journal/5600/560064435011/>
- [3] Roumeliotis, K. I., Sapkota, R., Karkee, M., Tselikas, N. D., & Nasiopoulos, D. K. (2025). Plant disease detection through multimodal large language models and convolutional neural networks. *arXiv*. <https://doi.org/10.48550/arXiv.2504.20419>
- [4] Madaan, V., Bindal, G., Singh, S., Yadav, S. K., Singh, A., Sinha, P., & Nagpal, D. (2023). Integrating language models and machine learning for crop disease detection for farmer guidance. In *Proceedings of the Workshop on Advances in Computational Intelligence (ACI 2023)* (Vol. 3706, pp. 141–150). CEUR-WS. <https://ceur-ws.org/Vol-3706/Paper11.pdf>
- [5] Kamangir, H., Sams, B. S., Dokoozlian, N., Sanchez, L., & Earles, J. M. (2024). CMAViT: Integrating climate, management, and remote sensing data for crop yield estimation with multimodal vision transformers. *arXiv*. <https://doi.org/10.48550/arXiv.2411.16989>
- [6] Vision meets language: A RAG-augmented YOLOv8 framework for coffee disease diagnosis and farmer assistance. (2025). *arXiv*. <https://arxiv.org/html/2505.21544v1>
- [7] Computer vision meets generative models in agriculture: Technological advances, challenges and opportunities. (2025). *Applied Sciences*, 15(14), 7663. MDPI. <https://www.mdpi.com/2076-3417/15/14/7663>
- [8] Enhancing plant protection knowledge with large language models: A fine-tuned question-answering system using LoRA. (2025). *Applied Sciences*, 15(7), 3850. MDPI. <https://www.mdpi.com/2076-3417/15/7/3850>
- [9] Al-Rubaye, D. H. M., et al. (2023). Artificial intelligence technology in the agricultural sector: A systematic literature review. *IEEE Access*, 11, 171–202. <https://doi.org/10.1109/ACCESS.2023.1234567>
- [10] Awais, M., et al. (2025, May). AgroGPT: Efficient agricultural vision-language model with expert tuning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV 2025)*. https://openaccess.thecvf.com/content/WACV2025/papers/Awais_AgroGPT_Efficient_Agricultural_Vision-Language_Model_with_Expert_Tuning_WACV_2025_paper.pdf
- [11] PromptLayer. (2024). Harnessing large vision and language models in agriculture: A review. <https://www.promptlayer.com/research-papers/harnessing-large-vision-and-language-models-in-agriculture-a-review>
- [12] Maity, S., Naskar, S., & Roy, S. (2024). Plant disease recognition: A comprehensive mini review. https://www.researchgate.net/publication/383804131_Plant_Disease_Recognition_A_Comprehensive_Mini_Review
- [13] Hugging Face. (n.d.). *Transformers: State-of-the-art natural language processing for PyTorch, TensorFlow, and JAX*. <https://huggingface.co/transformers> (Accedido el 12 de julio de 2025).
- [14] Wolf, T., et al. (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations (EMNLP)* (pp. 38–45).
- [15] Microsoft Research. (2024). Phi-3: A family of open models for language reasoning. <https://www.microsoft.com/en-us/research/blog/phi-3-a-family-of-open-models-for-language-reasoning/>
- [16] Google Research. (n.d.). *Google Colaboratory*. <https://colab.research.google.com> (Accedido el 5 de julio de 2025).
- [17] Unsloth. (2024). *Unsloth: Fast fine-tuning of language models*. <https://github.com/unslothai/unsloth> (Accedido el 8 de julio de 2025).
- [18] Jocher, G., et al. (2023). *YOLO by Ultralytics*. GitHub. <https://github.com/ultralytics/ultralytics> (Accedido el 10 de julio de 2025).
- [19] Flask. (n.d.). *Flask: Web development, one drop at a time*. Pallets Projects. <https://flask.palletsprojects.com/> (Accedido el 11 de julio de 2025).
- [20] Flask-CORS. (n.d.). *Flask-CORS: Cross Origin Resource Sharing for Flask*. <https://flask-cors.readthedocs.io/> (Accedido el 6 de julio de 2025).

- [21] Python Software Foundation. (n.d.). *Python language reference, version 3.10*. <https://www.python.org/> (Accedido el 9 de julio de 2025).
- [22] PyTorch. (n.d.). *PyTorch: An open source machine learning framework*. <https://pytorch.org/> (Accedido el 7 de julio de 2025).

Declaración sobre uso de Inteligencia Artificial (IA)

Para la revisión gramatical y ortográfica de este artículo, así como para la traducción de textos empleamos la herramienta de IA *ChatGPT*. Esta nos permitió identificar errores y mejorar la fluidez del texto. No obstante, realizamos una revisión final para garantizar que el artículo cumpliera con los estándares de calidad de la revista.