

EcoVision: Sistema de inteligencia artificial basado en visión por computadora para la detección y segmentación de latas y botellas

Autores

José Luis Martínez Díaz — jose_luis.martinez@uao.edu.co

Juan David Arroyave Ramírez — juan_dav.arroyave@uao.edu.co

Neiberth Aponte Aristizábal — neiberth.aponte@uao.edu.co

Stevens Ricardo Bohórquez Ruiz — stevens.bohorquez@uao.edu.co

UNIVERSIDAD AUTÓNOMA DE OCCIDENTE



Facultad de Ingenierías

Especialización en Inteligencia Artificial

Curso: Desarrollo de Proyectos de Inteligencia Artificial

Cali, Colombia

Septiembre 2025

Resumen

La gestión de residuos sólidos representa un desafío crítico en Colombia, donde la acumulación de latas y botellas plásticas contamina ecosistemas acuáticos y terrestres y eleva costos operativos de recolección. Este trabajo propone un sistema de inteligencia artificial basado en visión por computadora y Deep Learning para la detección y segmentación de latas y botellas, entrenado con YOLOv8 y desplegado mediante una interfaz Streamlit. Los objetivos incluyen la recolección y anotación de un dataset diverso en Roboflow, el entrenamiento y optimización del modelo con técnicas de Data Augmentation y ajuste de hiperparámetros, la validación en entornos controlados usando métricas de precisión y graficas de pérdida. El alcance comprende el desarrollo del modelo, la aplicación web, la documentación técnica y pruebas de campo con videos de Playas, orillas de ríos, zonas urbanas, o CCTV libres de derechos de autor que estén disponibles para uso académico sin fines comerciales o en datasets públicos o los generados por EcoVision, excluyendo la recolección manual de datos en tiempo real, la provisión de infraestructura de hardware, mantenimiento o soporte continuo post-entrega y desarrollo de modelos adicionales distintos a YOLOv8. Este enfoque aporta datos objetivos para optimizar rutas de recolección, apoyar iniciativas de economía circular y facilitar la toma de decisiones en autoridades ambientales y redes de recicladores.

EcoVision: Sistema de inteligencia artificial basado en visión por computadora para la detección y segmentación de latas y botellas

Septiembre 2025

Índice	ca	6
1 Introducción	3	7
2 Objetivo General	3	
3 Objetivos Específicos	4	
4 Alcance del Proyecto en Colombia	4	
4.1 Incluye	4	
4.2 Excluye	5	
5 Contexto del Proyecto	5	
5.1 Justificación	5	
5.2 Selección del Modelo .	5	
5.3 Problema Real	6	
5.4 Integración en Producción	6	
6 Cronograma y Ruta Crítica	11	
7 Marco de Investigación	7	
7.1 Modelos a utilizar . . .	8	
7.2 Datasets seleccionados	9	
7.3 Enlaces a repositorios .	9	
7.4 Técnicas	9	
7.5 Frameworks y Librerías	10	
7.6 Metodologías de trabajo adoptadas	11	
7.7 Tablero Kanban: Github Projects (Milestones e Issues) . . .	12	
8 Anexos	12	
9 Modelcard	12	
10 Restricciones técnicas	13	
11 Bibliografía	14	

1 Introducción

En la actualidad, la crisis ambiental causada por residuos sólidos representa uno de los desafíos más apremiantes para el planeta. Según datos recientes del Banco Mundial, se prevé que la generación mundial de residuos aumente un 70 % de aquí a 2050, mientras que cada año se vierten al medio ambiente más de 52 millones de toneladas de plásticos. En Colombia, se producen aproximadamente 25 000 toneladas de residuos sólidos diarios, de los cuales únicamente se recupera el 13 % y sólo el 7 % es comercializado por recicladores. Dentro de estos residuos, las botellas plásticas y las latas de aluminio constituyen una fracción considerable de los desechos que terminan en ríos, mares y rellenos sanitarios, generando graves impactos en los ecosistemas y la salud humana.

El aluminio presenta una tasa de reciclaje del 75 % a nivel mundial, superando significativamente al plástico que apenas alcanza el 9 %. Esta disparidad en las tasas de reciclaje subraya la necesidad urgente de implementar tecnologías innovadoras que faciliten la identificación y clasificación automatizada de estos materiales. En este contexto, los sistemas de visión por computadora y el aprendizaje profundo (Deep Learning) emergen como herramientas revolucionarias para la automatización de procesos de recolección y clasificación de residuos reciclables.

En el marco del proyecto colaborativo para la asignatura “Desarrollo

de Proyectos de Inteligencia Artificial”, este trabajo surge con el objetivo de abordar la siguiente cuestión:

¿Qué estrategia puede implementarse para la detección y segmentación de latas y botellas como residuos contaminantes del medio ambiente?

Surge así EcoVision como una solución innovadora que, mediante técnicas de visión por computadora y YOLOv8 para segmentación por instancias, identifica y genera máscaras de segmentación precisas a nivel de píxel para una cuantificación exacta.

La implementación de esta solución mediante Streamlit facilitará el acceso y uso de la tecnología al mismo tiempo que fortalece competencias en MLOps/DevOps y en desarrollo de aplicaciones interactivas, contribuyendo directamente a los objetivos de desarrollo sostenible relacionados con la gestión responsable de residuos y la protección del medio ambiente. El sistema propuesto representa una contribución significativa hacia la automatización del reciclaje y la promoción de una economía circular más eficiente, donde la inteligencia artificial se convierte en un aliado fundamental para enfrentar los desafíos ambientales del siglo XXI.

2 Objetivo General

Desarrollar un sistema de inteligencia artificial basado en visión por compu-

tadora para la detección y segmentación de latas y botellas, con el fin de apoyar la identificación automatizada de contaminantes y fortalecer estrategias de gestión ambiental.

3 Objetivos Específicos

- Recolectar y preparar un dataset de imágenes de latas y botellas en distintos contextos ambientales que incluya imágenes con diferentes condiciones de iluminación, fondos variados, múltiples orientaciones de los objetos, y diversos tamaños para garantizar la robustez del modelo y su generalización en escenarios reales.
- Etiquetar adecuadamente las imágenes que correspondan a botellas y latas mediante la plataforma de Roboflow, implementando técnicas de anotación precisas con cajas delimitadoras y segmentación poligonal.
- Organizar el dataset con división estratégica de 70-20-10 para entrenamiento, validación y prueba.
- Entrenar y evaluar un modelo de visión por computadora para la detección y segmentación de objetos utilizando YoloV8.
- Optimizar el rendimiento del modelo mediante técnicas de Data Augmentation y ajuste de hiperparámetros.

- Desplegar el sistema de inteligencia artificial basado en visión por computadora mediante la plataforma Streamlit, desarrollando una interfaz web interactiva que permita la carga de imágenes, videos, activación de la cámara para la utilización del modelo en tiempo real.

4 Alcance del Proyecto en Colombia

El sistema de inteligencia artificial EcoVision puede tener un alcance significativo y multifacético en Colombia, tanto en términos ambientales como sociales y económicos.

4.1 Incluye

• Modelo de Visión por Computadora

Selección, entrenamiento e implementación de la arquitectura YOLOv8 para la detección y segmentación de latas y botellas.

Interfaz de Usuario con Streamlit

- Desarrollo de una aplicación web interactiva que permita cargar imágenes o recibir flujo de video.
- Visualización en tiempo real de las detecciones y máscaras de segmentación en latas y botellas.

- Ajuste de parámetros de confianza y opciones de exportación de resultados con el fin de guardar pesos del modelo optimizado para futuros entrenamientos.

Documentación Técnica

- Descripción detallada de la arquitectura del sistema, flujo de datos y diagramas de componentes.
- Guía de uso de la aplicación Streamlit y manual de despliegue en servidores o en la nube.
- Informe de pruebas de rendimiento, métricas de precisión, como el mAP@50 y IoU y la matriz de confusión.
- La documentación técnica servirá tanto para reproducibilidad y replicabilidad en otros contextos.

Validación y Evaluación

- Pruebas en entornos controlados, videos de campo (Playas, orillas de ríos, zonas urbanas) libres de derechos de autor que estén disponibles para uso académico sin fines comerciales o en datasets públicos o los generados por EcoVision.
- Análisis estadístico de resultados y comparación con umbrales aceptables de detección.
- Ajustes de métricas avanzadas para técnicas de visión computacional como mAP@50 y IoU.

4.2 Excluye

- Recopilación manual de datos en tiempo real.
- Provisión de hardware y cámaras o servidores físicos.
- Mantenimiento o soporte post-entrega.
- Modelos adicionales distintos a YOLOv8.

5 Contexto del Proyecto

5.1 Justificación

La gestión de residuos sólidos es uno de los desafíos ambientales y sociales más urgentes en Colombia y en el mundo. La acumulación de latas y botellas en playas, riberas y zonas urbanas genera contaminación del suelo y del agua, afectando la biodiversidad, la salud pública y el turismo. Además, los procesos de recolección y clasificación manuales resultan costosos, lentos y propensos a errores humanos. En este escenario, aplicar visión por computadora y aprendizaje profundo permite automatizar la identificación de materiales reciclables, optimizar rutas de recolección y generar datos objetivos para políticas públicas.

5.2 Selección del Modelo

Se seleccionó YOLOv8 por su equilibrio entre velocidad y precisión en tareas de detección y segmentación

de instancias. Además, de su arquitectura optimizada para inferencia en tiempo casi real, fundamental para desplegar en Streamlit y probar en entornos controlados.

YOLOv8 es reconocido por su soporte nativo de segmentación semántica y poligonal, lo que facilita la cuantificación precisa de cada objeto.

5.3 Problema Real

EcoVision se propone como una herramienta de apoyo en los procesos de reciclaje y gestión de residuos sólidos, enfocándose en la detección de latas y botellas, los cuales representan una parte significativa del problema global de contaminación ambiental.

Este sistema de inteligencia artificial busca ser implementado por las autoridades locales, con el fin de identificar de manera temprana los puntos críticos de acumulación de estos residuos. De esta forma, se podrán programar intervenciones de limpieza más focalizadas y eficientes, contribuyendo a una mejor gestión ambiental en el país.

5.4 Integración en Producción

- **Implementación académica:** inicialmente se desplegará la aplicación en Streamlit para validar prototipos con el equipo desarrollador o compañeros estudiantes, donde se pueda utilizar datasets de campo, con sus debidos permisos o licencias de uso, incluyendo videos libres de

derechos de autor que estén disponibles para uso académico tipo CCTV o cámara en tiempo real activada por el mismo sistema de inteligencia artificial EcoVision.

Escalabilidad práctica:

- Tras la validación, el modelo podrá integrarse en sistemas de monitoreo continuo usando cámaras IP o circuitos cerrados en playas, riberas o zona urbana, enviando alertas automáticas a plataformas de gestión ambiental.

Colaboración institucional:

- Los datos generados pueden compartirse con alcaldías, corporaciones autónomas regionales (CAR) y redes de recicladores, alimentando dashboards de gestión y fortaleciendo la investigación aplicada en economía circular.

6 Cronograma y Ruta Crítica

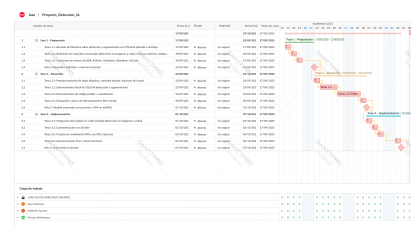


Figura 1: Diagrama de Gantt

El cronograma del proyecto contempla una duración total de quince días,

distribuidos en tres fases consecutivas: Preparación, Desarrollo e Implementación. La primera fase, del 17 al 19 de septiembre, se centra en la revisión de literatura sobre YOLOv8 aplicado al reciclaje, la definición de requisitos funcionales y la instalación del entorno de trabajo (CUDA, PyTorch, Ultralytics, Roboflow y VSCode), alcanzando como hito la disponibilidad de un entorno de desarrollo estable y alineado con los objetivos del proyecto. La segunda fase, que abarca del 20 al 27 de septiembre, concentra el desarrollo del modelo de IA, comenzando con el preprocesamiento del dataset (limpieza, aumento y balanceo de clases), seguido por el entrenamiento inicial de YOLOv8 en detección y segmentación, la refactorización del código de predicción y visualización, y finalmente la evaluación y ajuste de hiperparámetros. El hito de esta fase es contar con un modelo que supere el 90 % de precisión en mAP50. La tercera y última fase, del 28 de septiembre al 1 de octubre, está dedicada a la implementación práctica del modelo entrenado, integrándolo en un script modular para imágenes y video, contenedorizando la solución con Docker, ejecutando pruebas de rendimiento (FPS, uso de GPU y latencia), y cerrando con la documentación y la demo funcional final.

En cuanto a la ruta crítica, esta se compone de las tareas que no admiten retraso: la instalación del entorno, el preprocesamiento del dataset, el entrenamiento de YOLOv8, la evalua-

ción y ajuste de hiperparámetros y las pruebas de rendimiento. Estas actividades encadenadas determinan la duración total del proyecto, de modo que cualquier retraso en una de ellas impactará directamente en la entrega final de la demo.

7 Marco de Investigación

El presente proyecto se enmarca en el área de la Inteligencia Artificial aplicada a la Visión por Computador (CV), específicamente en el uso de modelos de segmentación y detección de objetos para la identificación de residuos reciclables. El problema central radica en la acumulación de botellas plásticas y latas metálicas en los ecosistemas, lo cual genera impactos negativos en el medio ambiente.

Esta investigación busca evaluar la capacidad que tiene esta arquitectura dentro del entorno Deep Learning para resolver problemas de clasificación automática, contribuyendo a iniciativas de reciclaje inteligente, tales como máquinas de retorno y sistemas de gestión ambiental. Este marco se justifica en la creciente necesidad de soluciones sostenibles que integren tecnología y preservación ambiental.

El proyecto emplea el modelo YOLO (You Only Look Once) en sus variantes más recientes de segmentación (como YOLOv8), ampliamente utilizado en tareas de detección de objetos en tiempo real debido a su balance entre precisión y velocidad. Este modelo permite no solo detectar, sino también segmentar las instancias de cada

objeto (botella o lata) dentro de imágenes y videos.

7.1 Modelos a utilizar

El proyecto emplea el modelo YOLO (You Only Look Once) en sus variantes más recientes de segmentación (como YOLOv8), ampliamente utilizado en tareas de detección de objetos en tiempo real debido a su balance entre precisión y velocidad. Este modelo permite no solo detectar, sino también segmentar las instancias de cada objeto (botella o lata) dentro de imágenes y videos.

YOLO (You Only Look Once) ha emergido como una de las arquitecturas más influyentes en la detección de objetos en tiempo real. Su filosofía radica en procesar una imagen en una sola pasada de red neuronal, lo que la convierte en un modelo altamente eficiente y aplicable en entornos productivos. Pertenece a la rama de Visión por Computador (CV), se sustenta en Redes Neuronales Convolucionales (CNNs) que forman parte del Deep Learning (DL) y se aplica en el subcampo de Detección y Segmentación de Objetos, esencial para aplicaciones como vehículos autónomos, seguridad, salud digital, robótica, reciclaje inteligente, entre otros. Su fortaleza radica en unir eficiencia computacional con alta precisión, lo que lo posiciona como un estándar de facto en proyectos académicos e industriales.

desarrollado por Ultralytics en 2023, es la evolución más reciente de esta

familia de modelos. Su arquitectura se compone de tres partes fundamentales:

Backbone:

Basado en una variante de CSP-Darknet (Cross Stage Partial Network), optimizada para YOLOv8, donde su función principal es extraer características jerárquicas de la imagen (bordes, texturas, formas y patrones más complejos).

Incluye bloques de convoluciones profundas y técnicas como Focus layers y CSP connections que mejoran la eficiencia computacional y la retención de información.

Neck:

Se emplea una estructura PAnet (Path Aggregation Network) combinada con FPN (Feature Pyramid Network), que le permite fusionar características a diferentes escalas, lo cual es esencial para detectar objetos pequeños como latas en el fondo y grandes como botellas en primer plano.

Head:

En YOLOv8, el head de detección ha sido rediseñado hacia una estructura anchor-free, es decir que en lugar de predefinir cajas de anclaje (anchor boxes), el modelo predice directamente la ubicación, tamaño y clase de los objetos, simplificando la arquitectura y mejorando la precisión en datasets variados.

Además, incorpora salidas para clasificación, regresión de cajas delimitadoras y segmentación de máscaras, lo que lo convierte en un modelo versátil para tareas de detección y segmentación por instancias.

7.2 Datasets seleccionados

El dataset fue gestionado a través de la plataforma Roboflow, que permitió la curación, anotación y preprocesamiento de imágenes. Las imágenes empleadas corresponden a fotografías de botellas plásticas y latas metálicas, capturadas en distintos entornos (fondos claros, oscuros y escenarios naturales) para garantizar la robustez del modelo frente a variaciones de contexto.

En Roboflow se aplicaron transformaciones de data augmentation (rotaciones, escalado, recorte, cambios de iluminación y contraste), lo cual incrementó la diversidad del dataset y redujo el riesgo de sobreajuste durante el entrenamiento. El conjunto final se dividió en:

Entrenamiento (70 %): imágenes utilizadas para ajustar los parámetros de la red.

Validación (20 %): conjunto empleado para optimizar hiperparámetros y monitorear el rendimiento.

Prueba (10 %): imágenes nunca vistas por el modelo, destinadas a la evaluación final.

Adicionalmente, el proyecto utilizó la técnica de Transfer Learning, partiendo de un modelo YOLOv8 pre-entrenado en un dataset de gran escala como COCO (Common Objects in Context) que es uno de los conjuntos de datos más usados en Visión por Computador; contiene más de 330.000 imágenes con anotaciones de más de 80 clases de objetos en contextos reales, siendo la base de en-

trenamiento para muchos modelos de detección como YOLO.

Esto permitió aprovechar las características previamente aprendidas (bordes, texturas, formas básicas) y adaptarlas a la tarea específica de detección de botellas y latas, logrando mayor precisión con un número relativamente reducido de imágenes propias.

En síntesis, la combinación de Roboflow para la gestión y aumento de datos, junto con transfer learning, fortaleció el proceso de entrenamiento y redujo significativamente los tiempos de cómputo requeridos.

7.3 Enlaces a repositorios

- Repositorio oficial de YOLOv8 (Ultralytics): www.github.com/ultralytics/ultralytics
- Roboflow: Dataset de botellas y latas: www.app.roboflow.com/stevensjuan/botellas-y-latas-4cxj5/1
- Repositorio del proyecto en GitHub: www.github.com/davidarroyave/EcoVision
- Repositorio del seguimiento y tickets del proyecto en GitHub Projects: www.github.com/users/davidarroyave/projects/2/views/1
- Repositorio del cronograma general del proyecto en GanttPro: www.app.ganttpro.com/#/project/1758949443744/gantt

7.4 Técnicas

Deep Learning (DL):

El proyecto emplea técnicas de aprendizaje profundo a través de redes neuronales convolucionales (CNN), que permiten extraer de manera jerárquica características visuales desde patrones simples (bordes, texturas) hasta representaciones más complejas (formas y objetos completos). Mediante el uso de transfer learning, se parte de un modelo previamente entrenado en un gran dataset (COCO) y se ajusta específicamente para la detección de botellas y latas. Esto reduce tiempos de entrenamiento y mejora la precisión del modelo con un dataset relativamente reducido.

Transfer Learning:

Partiendo de un modelo YOLOv8 pre entrenado en el dataset COCO, este enfoque de Transfer Learning nos permite reutilizar características generales ya aprendidas (bordes, texturas, formas básicas) y adaptarlas al problema específico de detección de botellas y latas. Gracias a esto se puede optimizar el rendimiento y se reduce significativamente el tiempo y costo computacional del entrenamiento.

Visión Computacional (CV):

Se aplica la visión por computador para la detección y segmentación de instancias de objetos en imágenes y videos. En particular, se utilizan técnicas de detección en tiempo real mediante el modelo YOLOv8, que no solo localiza los objetos mediante bounding boxes, sino que además permite segmentarlos para diferenciar cada instancia. Este enfoque es esencial en contextos como reciclaje automatizado, donde es necesario identificar

con exactitud los objetos en distintos escenarios y condiciones de iluminación.

7.5 Frameworks y Librerías

El desarrollo de este proyecto requiere el uso de diversos frameworks y librerías de programación en Python, cada uno aportando un rol específico en la implementación, entrenamiento y validación del modelo:

PyTorch: Es el framework principal de aprendizaje profundo, encargado de manejar las operaciones con tensores y el entrenamiento de las redes neuronales. Su flexibilidad permite implementar de manera eficiente el proceso de transfer learning sobre YOLOv8, ajustando los pesos preentrenados del modelo a las particularidades del dataset de botellas y latas.

Ultralytics YOLO: esta librería proporciona la implementación más reciente y optimizada de la familia YOLO. Simplifica las fases de entrenamiento, validación e inferencia, ofreciendo comandos de alto nivel que integran métricas, visualizaciones y exportación de modelos listos para producción.

OpenCV: se emplea para el procesamiento y manipulación de imágenes y videos, tanto en la etapa de preprocesamiento (ajustes de tamaño, formato y color) como en la fase de inferencia, donde permite visualizar de manera práctica los resultados de la detección y segmentación en tiempo real.

NumPy y Pandas: facilitan la ges-

tión de los datos asociados al proyecto, incluyendo la manipulación de etiquetas, estadísticas de distribución de clases y análisis de métricas de desempeño. Estas herramientas son esenciales para mantener la trazabilidad y organización del dataset.

Matplotlib y Seaborn: se utilizan para la representación gráfica de resultados. Con ellas se generan visualizaciones de curvas de aprendizaje, matrices de confusión, métricas de precisión y ejemplos de segmentación, lo cual permite una interpretación clara y didáctica de la evolución del modelo.

En conjunto, estas herramientas conforman un ecosistema integrado, donde cada una cumple un papel específico y complementario, asegurando que el proyecto pueda fluir desde la gestión del dataset hasta la obtención de un modelo funcional de visión por computador.

7.6 Metodologías de trabajo adoptadas

El desarrollo del proyecto EcoVision integra tres enfoques metodológicos que permiten estructurar el proceso tanto desde la gestión del equipo como desde el ciclo de vida del desarrollo científico-técnico: SCRUM, Ágil y CRISP-DM.

SCRUM

Se adopta como marco ágil de gestión para la parte técnica del proyecto, organizando las tareas en sprints y gestionando los avances mediante tableros Kanban en GitHub Projects.

El rol del Scrum Master garantiza la coordinación entre las actividades de investigación y las de desarrollo MLOps y DevOps, permite mantener visibilidad sobre tareas críticas como entrenamiento del modelo, implementación de pipelines, despliegue de la aplicación y pruebas de validación.

Metodologías Ágiles

El agilismo se incorpora como filosofía general de trabajo colaborativo e incremental, este proyecto se divide en entregables iterativos como el prototipo inicial del modelo, primera versión de la aplicación en Streamlit, versión final para validación, que son refinados con base en retroalimentación continua. Se busca maximizar adaptabilidad frente a cambios, ya sea en el dataset, en los requisitos de validación o en los recursos de despliegue.

CRISP-DM (Cross-Industry Standard Process for Data Mining)

Aunque surgió en el ámbito de minería de datos, sigue siendo un estándar robusto para proyectos científicos con componentes de Machine Learning, sus fases estructuran el desarrollo de EcoVision.

Comprensión del negocio: Impacto de residuos en Colombia y necesidad de automatización.

Comprensión de los datos: Selección y análisis de datasets (Roboflow, COCO).

Preparación de los datos: Limpieza, anotación, balanceo y preprocesamiento.

Modelado: Entrenamiento de YOLOv8 con transfer learning.

Evaluación: Validación con métricas (mAP, Precision, Recall) y pruebas en entornos reales.

Despliegue: Aplicación con Streamlit, backend en Python y contenedorización con Docker.

Esta metodología asegura rigor científico, reproducibilidad y documenta el proceso de forma estructurada.

7.7 Tablero Kanban: Github Projects (Milestones e Issues)

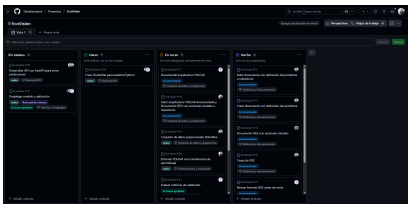


Figura 2: Tablero Kanban: Github Projects (Milestones e Issues)

8 Anexos

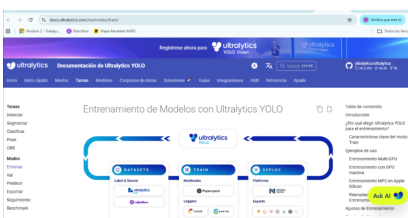


Figura 3: Anexo 1: Captura de Ultralytics Yolo Documentacion.

9 Modelcard

En el presente proyecto se empleó el modelo YOLOv8 (You Only Look On-

Sección	Descripción
Nombre del modelo	YOLOv8 (You Only Look Once, versión 8)
Tipo de modelo	Red neuronal convolucional para detección de objetos
Implementación	Framework Ultralytics en Google Colab
Tarea principal	Localización y clasificación de objetos en imágenes mediante bounding boxes
Datos utilizados	Dataset con imágenes etiquetadas en clases de interés, dividido en train/val/test
Preprocesamiento	Normalización de imágenes, ajuste de tamaño, partición de conjuntos
Métricas de evaluación	mAP@50, mAP@50-95, Recall, Pérdida (loss)
Uso previsto	Aplicaciones de visión computacional: detección automática de objetos, seguridad, monitoreo
Limitaciones	Dependencia de la calidad del dataset, dificultad con objetos pequeños o condiciones de baja iluminación, necesidad de GPU para entrenamiento
Consideraciones éticas	Posible sesgo en los datos, riesgos en vigilancia o usos que vulneren la privacidad

Figura 4: Anexo 2: Model Card.

ce, versión 8), una arquitectura de redes neuronales convolucionales diseñada para la detección de objetos en imágenes. Con el fin de documentar de manera transparente su funcionamiento, se elaboró una Model Card, la cual resume los principales aspectos relacionados con el entrenamiento, la evaluación y las condiciones de uso del modelo.

El modelo YOLOv8 fue implementado en Google Colab mediante el framework Ultralytics, configurado para identificar y clasificar objetos dentro de imágenes mediante la predicción de bounding boxes. El entrenamiento se llevó a cabo con un dataset previamente preparado, dividido en conjuntos de entrenamiento, validación y prueba, garantizando una adecuada evaluación del desempeño. Dicho dataset incluyó imágenes etiquetadas según las clases de interés definidas para el proyecto.

En cuanto al rendimiento, se evaluaron métricas como precisión promedio (mAP@50 y mAP@50-95), recall y la función de pérdida (loss) durante las épocas de entrenamiento. Estas métricas permitieron analizar la capacidad del modelo para detectar correctamente los objetos y la cobertura en la predicción de todas las ins-

tancias presentes en las imágenes.

El uso previsto de este modelo se centra en tareas de visión computacional, especialmente en la detección automática de objetos para aplicaciones de monitoreo, seguridad y análisis de imágenes. Sin embargo, es importante reconocer ciertas limitaciones: el rendimiento del modelo depende en gran medida de la calidad y representatividad del dataset utilizado, puede presentar dificultades con objetos muy pequeños o en condiciones de iluminación desfavorables, y requiere de recursos computacionales adecuados, especialmente GPU, para un entrenamiento eficiente.

Finalmente, desde la perspectiva ética, se deben considerar los posibles sesgos del dataset y los riesgos derivados de un uso indebido del modelo, en particular en contextos sensibles como la vigilancia masiva o el tratamiento de datos personales. En consecuencia, la Model Card permite ofrecer una visión integral del modelo YOLOv8, sirviendo como una hoja de vida técnica que detalla su propósito, desempeño, limitaciones y recomendaciones para un uso responsable.

10 Restricciones técnicas

Restricciones de Datos

Disponibilidad limitada de dataset: la precisión del modelo depende de la calidad y volumen de imágenes etiquetadas (botellas y latas, etc.). Un dataset insuficiente o desbalanceado puede afectar la generalización.

Privacidad y licenciamiento: solo se pueden usar datasets con licencias abiertas o datos recolectados por el equipo, evitando restricciones legales.

Formato de entrada: el sistema únicamente aceptará imágenes en formatos estándar (.jpg, .png); otros tipos (RAW, HEIC) requieren conversión previa.

Resolución mínima requerida: imágenes por debajo de 224x224 px pueden ser descartadas, ya que limitan la detección precisa.

Restricciones de Procesamiento y Modelado

Capacidad de hardware:

Entrenamiento inicial requiere GPU (NVIDIA con soporte CUDA/cuDNN).

Inferencia en tiempo real puede limitarse a CPU, pero con mayores tiempos de respuesta.

Tiempo de entrenamiento: entrenamientos largos (horas/días) no son viables en entornos sin aceleración de hardware.

Uso de librerías específicas: se restringe a frameworks compatibles con el entorno Python (TensorFlow, PyTorch, OpenCV).

Optimización de modelos: modelos demasiado pesados (>200 MB) no podrán desplegarse en dispositivos de bajo recurso (ej. Raspberry Pi, móviles).

Restricciones de Infraestructura

Entorno virtual obligatorio: todo el desarrollo debe ejecutarse dentro de entornos virtuales (venv/uv) para evitar conflictos de dependencias.

Compatibilidad con contenedores:

la solución debe ser empaquetada en Docker; librerías o dependencias que no soporten Docker limitan la portabilidad.

Requerimientos de red: si el sistema consulta APIs externas (ej. para validación de datos o despliegue en Azure/AKS), requiere conexión estable.

Restricciones de almacenamiento: los modelos entrenados y datasets pueden superar varios GB, lo que exige espacio de almacenamiento adicional en disco/nube.

Restricciones de Integración y Despliegue

Compatibilidad de SO: se garantiza soporte solo en sistemas Linux y Windows 10/11 (no probado en MacOS ni versiones anteriores).

Uso de contenedores orquestados: despliegues productivos deben realizarse en Kubernetes (AKS), limitando el uso en entornos no compatibles.

Control de versiones: todo el código debe residir en Git; integraciones fuera de Git (copias locales aisladas) no están permitidas.

Disponibilidad de GPU en la nube: el despliegue en Azure puede estar condicionado a la cuota de GPU habilitada por la cuenta (restricción de costos y disponibilidad).

Restricciones de Rendimiento

Tiempo de inferencia: cada predicción de imagen no puede superar los 2 segundos en entornos productivos.

Uso de memoria: el sistema no debe consumir más del 70

Restricciones de Seguridad

Gestión de dependencias: solo se permite instalar librerías verificadas desde PyPI o repositorios oficiales (evitando vulnerabilidades de paquetes maliciosos).

Exposición de API: si se expone un servicio REST/Flask, debe restringirse con autenticación básica o tokens.

Cumplimiento de estándares: la aplicación debe ajustarse a normativas básicas de protección de datos y uso ético de IA.

11 Bibliografía

Referencias

- [1] Banco Mundial, *What a Waste 2.0: A Global Snapshot of Solid Waste Management to 2050*, Accessed: Sept. 27, 2025, sep. de 2018. dirección: <https://www.bancomundial.org/es/news/immersive-story/2018/09/20/what-a-waste-an-updated-look-into-the-future-of-solid-waste-management>.
- [2] J. Planelles, “Cada año se vierten más de 52 millones de toneladas de plásticos al medioambiente según un estudio de Nature”, *El País*, sep. de 2024, Accessed: Sept. 27, 2025. dirección: <https://elpais.com/clima-y-medio-ambiente/2024-09-04/cada-ano-se-vierten-mas-de-52-millones-de-toneladas-de-plasticos-al>

- medioambiente-segun-un-estudio-de-nature.html.
- [3] Greenpeace Colombia, *El problema de los residuos*, Accessed: Sept. 27, 2025, 2025. dirección: <https://www.greenpeace.org/colombia/el-problema-de-los-residuos/>.
- [4] “Contaminación de residuos sólidos y sus efectos en la salud y medio ambiente”, *Revista INVECOM*, vol. 4, n.º 2, dic. de 2024, Accessed: Sept. 27, 2025. dirección: <http://revistainvecom.org/index.php/invecom/article/view/3557>.
- [5] Condensa, *Aluminio vs. Plástico: La sostenibilidad en envases*, Accessed: Sept. 27, 2025, abr. de 2024. dirección: <https://condensa.com/2024/04/17/aluminio-vs-plastico-la-sostenibilidad-en-envases/>.
- [6] ARPAL, *Memoria de Actividades 2024*, Accessed: Sept. 27, 2025, jun. de 2025. dirección: <https://aluminio.org/arpal-presenta-su-memoria-de-actividades-2024/>.
- [7] “Sistema de procesamiento de imágenes para la detección de residuos sólidos”, *AIBI Revista de Investigación*, vol. 13, n.º 1, abr. de 2025, Accessed: Sept. 27, 2025. dirección: <https://revistas.udes.edu.co/aibi/article/view/4426>.
- [8] Reciclaje y Gestión, *Tendencias y tecnologías en gestión de residuos para 2024*, Accessed: Sept. 27, 2025, mayo de 2024. dirección: <https://reciclajeygestion.es/2024/05/27/tendencias-y-tecnologias-en-gestion-de-residuos-para-2024-innovaciones-y-desarrollos-recientes-en-el-sector/>.
- [9] Recycleye, *La IA y el reconocimiento de residuos: por qué funciona*, Accessed: Sept. 27, 2025, jun. de 2024. dirección: <https://recycleye.com/es/la-ia-y-el-reconocimiento-de-residuos/>.
- [10] “Uso de la Visión Artificial para la Clasificación de Residuos Orgánicos e Inorgánicos”, *Nexos Científicos*, Accessed: Sept. 27, 2025. dirección: <https://nexoscientificos.vidanueva.edu.ec/index.php/ojs/article/download/61/268/501>.
- [11] RCB Trace, *Perspectivas futuras: avances tecnológicos en la gestión de residuos*, Accessed: Sept. 27, 2025, abr. de 2024. dirección: <https://www.rcbtrace.com/perspectivas-futuras-avances-tecnologicos-en-la-gestion-de-residuos/>.
- [12] Recyclever, *Reverse Vending: Beneficios y Futuro Sostenible*, Accessed: Sept. 27, 2025, feb. de 2025. dirección: <https://www.recyclever.com/es/blog/article->

10/reverse-vending-beneficios-y-futuro-sostenible-310.

- [13] Tiempo, *Reverse vending: el fenómeno de las máquinas que te pagan por reciclar*, Accessed: Sept. 27, 2025, dic. de 2024. dirección: <https://www.tiempo.com/noticias/actualidad/reverse-vending-el-fenomeno-de-las-maquinas-que-te-pagan-por-reciclar.html>.
- [14] Ultralytics, *Segmentación de instancias con seguimiento de objetos*, Accessed: Sept. 27, 2025, mayo de 2025. dirección: <https://docs.ultralytics.com/es/guides/instance-segmentation-and-tracking/>.
- [15] Ultralytics, *Segmentación de Instancias*, Accessed: Sept. 27, 2025, mar. de 2025. dirección: <https://docs.ultralytics.com/es/tasks/segment/>.
- [16] *Detección y Cuantificación de Erosión Fluvial con Visión por Computadora*, Accessed: Sept. 27, 2025, dic. de 2021. arXiv: 2507.11301. dirección: <https://arxiv.org/html/2507.11301v1>.
- [17] Ultralytics, *Un futuro más verde mediante Vision AI y Ultralytics YOLO*, Accessed: Sept. 27, 2025, sep. de 2025. dirección: <https://www.ultralytics.com/es/blog/greener-future-through-vision-ai-and-ultralytics-yolo>.