

---

# Desarrollo de un Sistema de Identificación y Clasificación Automatizado de Residuos para Fomentar la Recolección y Reciclaje Eficiente para el Medio Ambiente

---

Jun Woo Lee Hong





UNIVERSIDAD DEL VALLE DE GUATEMALA  
Facultad de Ingeniería



**Desarrollo de un Sistema de Identificación y Clasificación  
Automatizado de Residuos para Fomentar la Recolección y  
Reciclaje Eficiente para el Medio Ambiente**

Trabajo de graduación presentado por Jun Woo Lee Hong para optar al  
grado académico de Licenciado en Ingeniería en Ciencias de la  
Computación y Tecnología de la Información

Guatemala,

2024







UNIVERSIDAD DEL VALLE DE GUATEMALA  
Facultad de Ingeniería



**Desarrollo de un Sistema de Identificación y Clasificación  
Automatizado de Residuos para Fomentar la Recolección y  
Reciclaje Eficiente para el Medio Ambiente**

Trabajo de graduación presentado por Jun Woo Lee Hong para optar al  
grado académico de Licenciado en Ingeniería en Ciencias de la  
Computación y Tecnología de la Información

Guatemala,

2024





Vo.Bo.:

(f) \_\_\_\_\_  
Cho Kyu Sang

Tribunal Examinador:

(f) \_\_\_\_\_  
Cho Kyu Sang

(f) \_\_\_\_\_  
Gabriel Antonio Barrientos Rodríguez

(f) \_\_\_\_\_  
Erick Francisco Marroquín Rodríguez

Fecha de aprobación: Guatemala, 4 de diciembre de 2024



|   |             |
|---|-------------|
| <b>Lista de figuras</b>   | <b>VII</b>  |
| <b>Lista de cuadros</b>   | <b>IX</b>   |
| <b>Resumen</b>  | <b>XI</b>   |
| <b>Abstract</b>   | <b>XIII</b> |
| <b>1. Introducción</b>  | <b>1</b>    |
| <b>2. Antecedentes</b>  | <b>3</b>    |
| <b>3. Justificación</b>   | <b>5</b>    |
| <b>4. Objetivos</b>   | <b>7</b>    |
| 4.1. Objetivo general . . . . .                                     | 7           |
| 4.2. Objetivos específicos . . . . .                                | 7           |
| <b>5. Marco teórico</b>   | <b>9</b>    |
| 5.1. Inteligencia Artificial . . . . .                              | 9           |
| 5.1.1. Aprendizaje Automatizado . . . . .                           | 9           |
| 5.1.2. Recolección y Preparación de Datos . . . . .                 | 12          |
| 5.1.3. Visión por Computadora . . . . .                             | 13          |
| 5.2. <i>YOLO: You Only Look Once</i> . . . . .                      | 14          |
| 5.2.1. Arquitectura del Modelo YOLO . . . . .                       | 14          |
| <b>6. Metodología</b>   | <b>17</b>   |
| 6.1. Recoleccion de Datos . . . . .                                 | 17          |
| 6.1.1. Obtencion del Conjunto de Datos TACO . . . . .               | 17          |
| 6.1.2. Etiquetado de Datos con <i>LabelMe</i> . . . . .             | 18          |
| 6.2. Preprocesamiento de Datos . . . . .                            | 18          |
| 6.2.1. Creación del Conjunto de Entrenamiento . . . . .             | 19          |
| 6.3. Ampliación del Conjunto de Datos . . . . .                     | 19          |
| 6.3.1. Obtención de Imágenes Adicionales de <i>Kaggle</i> . . . . . | 19          |

|   |           |
|---|-----------|
| 6.4. Evaluación del Modelo YOLOv8 . . . . .               | 20        |
| 6.4.1. Métricas de Evaluación . . . . .                   | 20        |
| 6.5. Implementación de Detección en Tiempo Real . . . . . | 21        |
| <b>7. Resultados</b>                                      | <b>23</b> |
| 7.1. Resultados del Primer Modelo . . . . .               | 23        |
| 7.1.1. Evaluación con la Matriz de Confusión . . . . .    | 23        |
| 7.1.2. Análisis de la Curva de Confianza F1 . . . . .     | 24        |
| 7.2. Resultados del Segundo Modelo . . . . .              | 25        |
| 7.2.1. Evaluación con la Matriz de Confusión . . . . .    | 25        |
| 7.2.2. Análisis de la Curva de Confianza F1 . . . . .     | 26        |
| <b>8. Conclusiones</b>                                    | <b>29</b> |
| <b>9. Recomendaciones</b>                                 | <b>31</b> |
| <b>10. Bibliografía</b>                                   | <b>33</b> |
| <b>11. Anexos</b>   | <b>35</b> |

---

## Lista de figuras

---

|     |   |    |
|-----|---|----|
| 1.  | Ejemplos de transformaciones posibles de imágenes . . . . . | 12 |
| 2.  | Modelo de como funciona el algoritmo YOLO . . . . .         | 15 |
| 3.  | Etiquetando en <i>LabelMe</i> . . . . .                     | 18 |
| 4.  | Matriz de Confusión del Primer Modelo . . . . .             | 24 |
| 5.  | Curva de Confianza F1 del Primer Modelo . . . . .           | 25 |
| 6.  | Matriz de Confusión del Segundo Modelo . . . . .            | 26 |
| 7.  | Curva de Confianza F1 del Segundo Modelo . . . . .          | 27 |
| 8.  | Detección de una botella de agua . . . . .                  | 35 |
| 9.  | Detección de una soda <i>Tiky</i> . . . . .                 | 36 |
| 10. | Detección de una botella de agua . . . . .                  | 36 |
| 11. | Detección de una lata de Coca . . . . .                     | 37 |
| 12. | Detección de una lata de Agua Mineral . . . . .             | 37 |
| 13. | Detección de caja de Capuchino . . . . .                    | 38 |
| 14. | Detección de un papel arrugado . . . . .                    | 38 |



---

## Lista de cuadros

---

|    |   |    |
|----|---|----|
| 1. | Cantidad total de imágenes por división . . . . . | 20 |
| 2. | Cantidad total de objetos por clase . . . . .     | 20 |
| 3. | Cantidad total de objetos por clase . . . . .     | 20 |





Este trabajo se centra en el desarrollo y entrenamiento de un modelo de aprendizaje automatizado basado en la arquitectura de detección de objetos *YOLO: You Only Look Once*, conocida por su alta velocidad y precisión en la identificación de elementos visuales, características que lo hacen especialmente adecuado para aplicaciones en tiempo real. En este caso, *YOLO* se ha implementado para la clasificación de residuos reciclables en tres categorías principales: Papel-Cartón, Metal y Plásticos. Este sistema tiene como objetivo facilitar el reciclaje mediante la detección automática de materiales reciclables en entornos cotidianos.

Para el entrenamiento de este modelo, se utilizó una combinación de conjuntos de datos, incluyendo el *TACO Database*, que ofrece una amplia colección de imágenes de residuos etiquetadas y categorizadas, así como otras bases de datos de *Kaggle* que complementaron el conjunto de entrenamiento con una mayor variedad de ejemplos y contextos visuales. Este enfoque integral permitió una preparación robusta del modelo, abarcando distintos tipos de residuos y condiciones ambientales para mejorar la precisión del sistema.

El resultado de este proyecto es un modelo de clasificación en tiempo real capaz de identificar residuos reciclables a través de una cámara de video, alcanzando una precisión promedio del 70%. Aunque el modelo aún no es completamente preciso, este nivel de exactitud es considerado satisfactorio y demuestra el potencial de la inteligencia artificial para fomentar prácticas sostenibles de reciclaje. Esta precisión permite que el sistema funcione de manera confiable en aplicaciones prácticas, contribuyendo a la separación de residuos y promoviendo la conciencia ambiental en el proceso de reciclaje. Además, el modelo presenta una escalabilidad que permite su implementación en una variedad de entornos, desde residencias hasta instalaciones industriales, donde puede contribuir a la gestión eficiente de residuos y al impulso de prácticas de reciclaje automatizadas.



Este proyecto presenta el diseño y desarrollo de un sistema automatizado de identificación y clasificación de residuos mediante visión por computadora, con el objetivo de optimizar las prácticas de reciclaje en espacios públicos. Ante la creciente necesidad global de una gestión de residuos eficiente, este proyecto aprovecha tecnologías avanzadas para agilizar la clasificación de materiales reciclables, reduciendo las ineficiencias y errores comunes en los procesos manuales de separación.

El sistema está basado en el modelo de aprendizaje profundo YOLOv8, seleccionado por su robustez en tareas de detección de objetos. Los datos de entrenamiento provienen principalmente del conjunto de la *database TACO* y se complementaron con imágenes de *Kaggle*, centradas en tres categorías de residuos: metal, plástico y papel-cartón. La preparación de datos incluyó una anotación detallada mediante la herramienta *LabelMe* y técnicas de preprocesamiento para convertir las imágenes en un formato compatible con YOLOv8.

Donde las métricas de evaluación, incluidas la matriz de confusión y las curvas de confianza F1, demostraron que el modelo es capaz de clasificar residuos con un equilibrio adecuado entre velocidad y precisión para aplicaciones prácticas. Ya que se obtuvo un resultado de *F1-score* promedio de 0.69. Lo cual para mejorar el rendimiento del modelo, se recomendaría ampliar más las imágenes que se usaron para entrenar el modelo de 3,755 hasta mucho más para incluir más variedades y diferentes ángulos de todos los objetos para que pueda entrenar mejor.



La gestión eficiente de los residuos es un desafío crucial en nuestros tiempos debido a la acumulación de desechos contaminantes en vertederos y la falta de reciclaje efectivo, lo que afecta negativamente el medio ambiente al igual que la salud pública a nivel global. Según el Global Waste Management Outlook 2024 de UNEP, la gestión inadecuada de residuos puede llevar a costos económicos significativos, contribuir al cambio climático y afectar la salud pública [1]. Por lo que surge la necesidad de desarrollar un Sistema de Identificación y Clasificación Automatizada de Residuos que contribuya a mejorar la recolección y el reciclaje de residuos de manera eficiente. Este proyecto se enfoca en abordar esta problemática a través de la aplicación de tecnologías avanzadas como la visión por computadora y el aprendizaje automático. El objetivo es diseñar, desarrollar e implementar un sistema que pueda identificar y clasificar automáticamente diferentes tipos de residuos, facilitando así su separación y reciclaje adecuado. Este enfoque no solo tiene un impacto positivo en la preservación de recursos naturales, sino que también contribuye a la reducción de la contaminación ambiental y al fomento de prácticas de reciclaje más sostenibles.

La importancia de este proyecto viene por su potencial para revolucionar la forma en que se gestionan los residuos, promoviendo una mayor eficiencia en la recolección y una mayor tasa de reciclaje. Además, busca sensibilizar a la comunidad sobre la importancia de la gestión adecuada de residuos y la adopción de prácticas más responsables desde el punto de vista ambiental.



Un estudio [2] destaca el uso de transformadores de visión en dispositivos portátiles para la clasificación de residuos, mostrando cómo el aprendizaje profundo puede adaptarse a diferentes contextos tecnológicos y necesidades específicas de clasificación. A su vez, investigaciones [3] enfatizan la importancia de la clasificación inteligente de residuos para promover prácticas de reciclaje sostenibles. Estas investigaciones coinciden en que los sistemas basados en aprendizaje profundo no solo superan a los métodos manuales en precisión, sino que también son herramientas fundamentales para fomentar el reciclaje mediante la identificación precisa de materiales. Además, investigaciones han señalado que un conjunto de datos más equilibrado y con múltiples ejemplos por clase permite mejorar la capacidad del modelo para generalizar y reducir errores en clasificaciones futuras [4].





La necesidad de desarrollar un Sistema de Identificación y Clasificación Automatizada de Residuos en espacios públicos viene debido a la creciente preocupación por optimizar las prácticas de reciclaje y la gestión de residuos. Dicha automatización se justifica por la eficiencia y precisión que ofrece en la identificación y clasificación de residuos reciclables superando los retos que plantea la clasificación manual, que es más lenta y propensa a errores. Además, sistemas automatizados pueden gestionar grandes volúmenes de residuos, asegurando una separación efectiva que facilita el reciclaje, contribuyendo significativamente a la sostenibilidad ambiental [2].

La tecnología de visión por computadora y el aprendizaje automático son herramientas claves para la clasificación eficaz de residuos. Ofreciendo una precisión y eficiencia que separa a los métodos manuales. Como un estudio detalla el uso de algoritmos avanzados para detectar, clasificar y localizar diversos tipos de residuos sólidos, mejorando significativamente la eficiencia y exactitud de los procesos de clasificación de residuos. Este enfoque ayuda a promover más el reciclaje y lo facilita al mismo tiempo [3].

Este proyecto surge como solución a los crecientes desafíos ambientales, como la contaminación que requieren soluciones innovadoras y efectivas. Por lo que con este proyecto se busca implementar una forma para que las personas puedan contribuir a mitigar estos problemas, aportando una solución en lugares públicos. Por lo que el desarrollo de un Sistema de Identificación y Clasificación Automatizada de Residuos es vital para mejorar la eficiencia del reciclaje en espacios públicos y contribuir a un futuro más sostenible.



#### 4.1. Objetivo general

Diseñar y desarrollar un sistema de identificación de residuos basado en visión por computadora que pueda analizar y clasificar objetos antes de arrojar a contenedores de reciclaje para ayudar a automatizar el proceso de clasificación de residuos en sus contenedores apropiados.

#### 4.2. Objetivos específicos

- Entrenar y ajustar el modelo de visión por computadora utilizando conjuntos de datos de referencia y técnicas de aprendizaje supervisado.
- Integrar una cámara y sensores en el sistema para capturar imágenes de los objetos arrojados a los contenedores de reciclaje.
- Optimizar el sistema para lograr una alta tasa de identificación y clasificación correcta de residuos en tiempo real.
- Evaluar y mejorar la eficiencia del sistema de identificación y clasificación en términos de precisión y velocidad.



## 5.1. Inteligencia Artificial

La inteligencia artificial(IA) es una disciplina de la informática que se dedica al desarrollo de sistemas y algoritmos capaces de realizar tareas que, tradicionalmente requieren inteligencia humana. Estas tareas incluyen el reconocimiento de voz, la toma de decisiones, la traducción de idiomas y la visión por computadora, entre otras. La IA se basa en el uso de algoritmos que pueden aprender de los datos, adaptarse a nueva información y mejorar su desempeño a través de la experiencia [5].

El concepto de IA se puede dividir en varias subáreas, incluyendo el aprendizaje automático, el procesamiento del lenguaje natural, la robótica y la visión por computadora. Cada una de estas subáreas se enfoca en diferentes aspectos de la inteligencia y ofrece diversas aplicaciones prácticas [6].

La IA se basa en la idea de crear máquinas que puedan replicar o simular funciones cognitivas humanas. Esto incluye la capacidad de razonar, resolver problemas, comprender el lenguaje y aprender de la experiencia. Los avances en la IA han sido posibles gracias a la disponibilidad de grandes volúmenes de datos, el aumento en la capacidad de procesamiento computacional y los avances en los algoritmos de aprendizaje [4].

### 5.1.1. Aprendizaje Automatizado

El aprendizaje automático es una subárea de la IA que se centra en el desarrollo de algoritmos y modelos que permiten a las computadoras aprender a partir de datos y mejorar su desempeño en tareas específicas sin ser programadas explícitamente para cada una de ellas. En lugar de seguir instrucciones estáticas, los sistemas de aprendizaje automático utilizan patrones y estadísticas derivados de los datos para tomar decisiones y hacer predicciones[7].

El aprendizaje automático se basa en la idea de que los sistemas pueden identificar patro-

nes y relaciones en los datos y utilizar este conocimiento para tomar decisiones informadas. Este enfoque permite que las máquinas se adapten a nuevas situaciones y mejoren continuamente a medida que se les proporciona más información. Los modelos de aprendizaje automático se desarrollan y perfeccionan mediante un proceso de entrenamiento, evaluación y ajuste, lo que les permite alcanzar niveles de precisión y eficiencia superiores [6]. El punto principal del aprendizaje automático es que los datos contienen patrones ocultos que pueden ser utilizados para predecir resultados futuros. Los algoritmos de aprendizaje automático analizan grandes conjuntos de datos para identificar estas relaciones y construir modelos que pueden generalizar a nuevos datos no vistos. Este proceso de descubrimiento de patrones es crucial para el éxito del aprendizaje automático y es lo que permite que los modelos aprendan y mejoren con el tiempo.

Los datos utilizados en el aprendizaje automático pueden ser de diversos tipos, incluyendo datos estructurados como tablas de bases de datos, datos no estructurados como texto, imágenes, o audios, y datos semiestructurados como documentos XML, JSON, etc. La representación adecuada de estos datos es esencial para el rendimiento del modelo. Por ejemplo, en el caso de imágenes, se utilizan matrices de píxeles, mientras que para el texto, se pueden utilizar técnicas de representación como bolsas de palabras o incrustaciones de palabras [8].

El proceso de aprendizaje en *machine learning* se puede dividir en varias etapas clave:

- **Recolección y Preparación de Datos:** La calidad de los datos es crucial para el éxito del modelo. Los datos deben ser recolectados, limpiados y procesados para asegurar que se puedan utilizar. Esto incluye la eliminación de valores atípicos, la eliminación de valores faltantes, la normalización y la transformación de características.
- **Selección del Modelo:** La elección del modelo adecuado depende de la naturaleza del problema y los datos disponibles. Existen varios algoritmos de aprendizaje automático, cada uno con sus ventajas y desventajas. La selección del modelo implica evaluar diferentes algoritmos y seleccionar el que mejor se adapte a los requisitos del proyecto.
- **Entrenamiento del Modelo:** El entrenamiento del modelo implica ajustar los parámetros del algoritmo para minimizar el error en el conjunto de entrenamiento. Este proceso se realiza mediante técnicas de optimización, como el descenso de gradiente, que ajustan los pesos del modelo para mejorar su precisión. Durante esta fase, el modelo aprende a mapear entradas a salidas utilizando los datos etiquetados.
- **Evaluación del Modelo:** Una vez entrenado, el modelo debe ser evaluado utilizando un conjunto de datos de prueba que no se ha utilizado durante el entrenamiento. Esto permite medir la capacidad del modelo para generalizar a nuevos datos y verificar su precisión y robustez. Las métricas comunes de evaluación incluyen la exactitud, la precisión, el valor f1, etc.
- **Optimización y Ajuste del Modelo:** Basado en los resultados de la evaluación, se pueden realizar ajustes adicionales para mejorar el rendimiento del modelo. Esto puede incluir el ajuste de parámetros, la implementación de técnicas de regularización para evitar el sobre ajuste y la expansión del conjunto de datos de entrenamiento mediante técnicas de aumento de datos.

- **Despliegue y Monitoreo del Modelo:** Finalmente, el modelo se implementa en un entorno de producción donde puede hacer predicciones en tiempo real o en lotes. Es crucial monitorear el rendimiento del modelo en producción para detectar posibles degradaciones en su precisión y realizar ajustes continuos según sea necesario.

El aprendizaje automático se clasifica en tres categorías principales:

### **Aprendizaje Supervisado**

En el aprendizaje supervisado, el modelo se entrena utilizando un conjunto de datos etiquetados, lo que significa que cada ejemplo de entrenamiento incluye tanto una entrada como una salida. El objetivo es aprender una función que mapee las entradas a las salidas correctas. Algunos algoritmos populares de aprendizaje supervisado incluyen regresión lineal, máquinas de soporte vectorial, y redes neuronales profundas.

- **Clasificación:** Asigna una etiqueta a cada conjunto de datos de entrada. Ejemplos incluyen la clasificación de correos electrónicos como spam o no spam y la identificación de tipos de residuos.
- **Regresión:** Predice un valor continuo basado en los datos de entrada. Ejemplos incluyen la predicción de si una entrada es plástico, metal, o papel.

### **Aprendizaje No Supervisado**

El aprendizaje no supervisado trabaja con datos no etiquetados. El objetivo es encontrar estructuras o patrones ocultos en los datos sin conocer las salidas correctas de antemano. Algunos algoritmos comunes de aprendizaje no supervisado incluyen el *clustering* y la reducción de dimensionalidad.

- **Clustering:** Agrupa los datos en conjuntos de elementos similares. Un ejemplo es el análisis de clientes para identificar segmentos de mercado.
- **Reducción de Dimensionalidad:** Simplifica los datos eliminando características redundantes o irrelevantes mientras se preserva la información importante. Un ejemplo es el análisis de componentes principales utilizado para visualizar datos de alta dimensión.

### **Aprendizaje por Refuerzo**

En el aprendizaje por refuerzo, un agente aprende a tomar decisiones mediante la interacción con un entorno y la recepción de recompensas o castigos basados en sus acciones. El objetivo es aprender una política que maximice la recompensa acumulada a lo largo del tiempo. Este tipo de aprendizaje es particularmente útil en aplicaciones como el control robótico y los juegos.

- **Agente y Entorno:** El agente realiza acciones en el entorno y recibe una recompensa basada en la calidad de sus acciones.
- **Política de Aprendizaje:** Define la estrategia que el agente sigue para elegir sus acciones en función de su experiencia pasada.

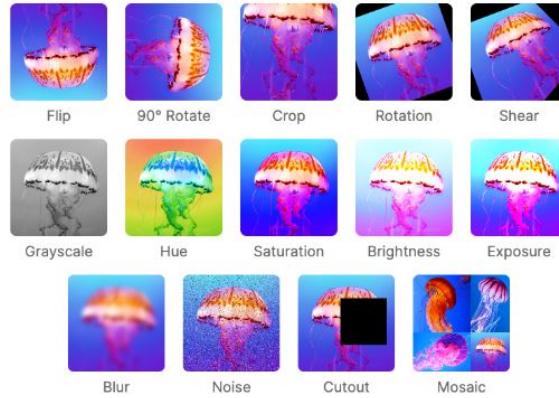


Figura 1: Ejemplos de transformaciones posibles de imágenes

### 5.1.2. Recolección y Preparación de Datos

La recolección y preparación de datos son pasos fundamentales en el desarrollo de modelos de aprendizaje automático y visión por computadora. Estos procesos aseguran que los datos utilizados para entrenar y evaluar los modelos sean de alta calidad, relevantes y representativos del problema que se desea resolver. Sin una recolección y preparación adecuadas, los modelos pueden sufrir de problemas como el sobreajuste o sesgos, lo que afecta negativamente su desempeño y precisión.

La recolección de datos implica la obtención de datos brutos de diversas fuentes, mientras que la preparación de datos incluye una serie de técnicas para limpiar, transformar y estructurar estos datos, de manera que sean aptos para el análisis y el entrenamiento de modelos. Estas etapas son cruciales para garantizar que los datos sean consistentes, completos y libres de errores, lo que a su vez mejora la eficacia de los modelos de aprendizaje automático [9].

**Limpieza y Transformación de Datos de Imágenes** Una vez procesadas las imágenes, el siguiente paso es la limpieza y transformación de los datos. Esto incluye la corrección de etiquetas incorrectas y la normalización de las imágenes. Además, se pueden aplicar técnicas de aumento de datos para generar nuevas muestras a partir de las imágenes existentes, lo que ayuda a aumentar la diversidad y robustez del conjunto de datos.

**División del Conjunto de Datos** Para asegurar una evaluación precisa del sistema, el conjunto de datos de imágenes debe dividirse en conjuntos de entrenamiento, validación y prueba. Esta división permite ajustar los parámetros del modelo y evaluar su rendimiento en datos no vistos, garantizando que el modelo pueda generalizar bien a nuevas imágenes de residuos.

La recolección y preparación de datos son pasos críticos en el desarrollo de un sistema de identificación y clasificación automatizada de residuos. Al asegurar que los datos utilizados sean de alta calidad y representativos, se mejora significativamente la precisión y eficiencia del modelo, contribuyendo al éxito del proyecto y a la sostenibilidad ambiental.



### 5.1.3. Visión por Computadora

La visión por computadora es una subdisciplina de la IA que se centra en el desarrollo de algoritmos y sistemas que permiten a las computadoras interpretar y comprender el contenido visual del mundo. Esto incluye imágenes y videos capturados por cámaras o sensores. La visión por computadora tiene como objetivo replicar la capacidad humana de ver y comprender el entorno visual, permitiendo que las máquinas realicen tareas que requieren reconocimiento y análisis de imágenes [10].

En esencia, la visión por computadora se basa en la idea de que las imágenes pueden ser procesadas y analizadas para extraer información útil. Los algoritmos de visión por computadora utilizan técnicas avanzadas de procesamiento de imágenes y aprendizaje automático para identificar patrones, objetos y características en las imágenes. Este proceso implica varias etapas, desde la adquisición y preprocesamiento de imágenes hasta la segmentación, detección y reconocimiento de objetos [11].

**Adquisición y Preprocesamiento de Imágenes** La primera etapa en la visión por computadora es la adquisición de imágenes, que pueden ser capturadas mediante cámaras digitales, sensores de profundidad, cámaras infrarrojas, entre otros dispositivos. Una vez adquiridas, las imágenes deben ser procesadas para mejorar su calidad y preparar los datos para el análisis posterior. El preprocesamiento puede incluir la reducción de ruido, el ajuste de contraste, la corrección de distorsiones y la normalización de las imágenes [12].

**Segmentación de Imágenes** La segmentación es el proceso de dividir una imagen en regiones o segmentos significativos que corresponden a diferentes objetos o partes de un objeto. Este paso es crucial para aislar los objetos de interés del fondo y facilitar su análisis. Los métodos de segmentación pueden basarse en el agrupamiento de píxeles con características similares, la detección de bordes, entre otros.[13].

**Detección y Reconocimiento de Objetos** Una vez segmentada la imagen, el siguiente paso es la detección y reconocimiento de objetos. La detección de objetos implica localizar instancias de objetos en una imagen, mientras que el reconocimiento de objetos implica identificar y clasificar estos objetos en categorías predefinidas. Los algoritmos de detección y reconocimiento utilizan características visuales como bordes, texturas, y formas para identificar objetos. Técnicas como las redes neuronales convolucionales han demostrado ser particularmente efectivas para estas tareas debido a su capacidad para aprender características jerárquicas y patrones complejos [14].

Hay una gran variedad de algoritmos de visión por computadora, pero unos de los más conocidos son los siguientes:

**Redes Neuronales Convolucionales (CNN)** Las CNN son una clase de redes neuronales profundas que son particularmente efectivas para la tarea de análisis de imágenes. Las CNN utilizan capas convolucionales para filtrar las imágenes y extraer características jerárquicas, lo que les permite detectar patrones complejos y reconocer objetos con alta precisión. Estos modelos han sido exitosos en aplicaciones como la clasificación de imágenes, la detección de objetos y la segmentación de imágenes [14].

**Detección de Bordes y Segmentación** Los algoritmos de detección de bordes iden-

tifican los contornos de los objetos en una imagen, lo que facilita la segmentación y el reconocimiento de objetos. La detección de bordes puede realizarse mediante operadores como el detector de *Canny*, que busca cambios abruptos en la intensidad de los píxeles para identificar los bordes. La segmentación puede realizarse mediante métodos basados en regiones o en bordes, que agrupan píxeles similares en segmentos significativos [13].

## 5.2. *YOLO: You Only Look Once*

El modelo YOLO fue desarrollado por Joseph Redmon, Santosh Divvala, Ross Girshick y Ali Farhadi, y fue presentado por primera vez en el un artículo [15]. Donde la motivación principal detrás de YOLO fue la necesidad de un modelo que pueda realizar detección de objetos en imágenes de manera rápida y eficiente sin comprometer la precisión.

El enfoque propuesto por YOLO era bien diferente a la de los modelos anteriores como R-CNN y sus variantes ya que los modelos basados en R-CNN dividían la tarea de detección en varias etapas: primero, identificaban las regiones candidatas en la imagen que probablemente contenían objetos y, posteriormente, pasaban esas regiones a una red neuronal para clasificar los objetos. Lo cual es un método bien efectivo en términos de precisión, pero era computacionalmente muy costoso y no apto para aplicaciones en tiempo real, como la conducción autónoma o la video vigilancia. Por lo que se necesitaba una alternativa mas rápida. Por lo que YOLO fue creado para tratar el problema de la detección de objetos como un solo problema de regresión desde la imagen completa hasta las clases y cuadros delimitadores de los objetos. Esto permitió que YOLO fuera mas rápido en comparación con otros modelos, ya que analizaba la imagen completa en una sola vista.

### 5.2.1. Arquitectura del Modelo YOLO

La arquitectura de YOLO se basa en una red neuronal convolucional que es capaz de dividir una imagen en una cuadrícula y realizar múltiples predicciones de objetos dentro de cada celda de la cuadrícula. Esta arquitectura innovadora está diseñada para lograr un equilibrio entre la precisión de detección y la velocidad de procesamiento, lo que permite la detección de objetos en tiempo real. A continuación, se describen los aspectos técnicos clave de la arquitectura YOLO.

#### Division de la Imagen en Celdas

YOLO comienza dividiendo la imagen de entrada en una cuadrícula de  $S \times S$  celdas. Cada celda de la cuadrícula es responsable de predecir un número fijo de cuadros delimitadores y una puntuación de confianza asociada para cada cuadro. Esta puntuación de confianza refleja tanto la probabilidad de que un objeto esté presente en el cuadro delimitador como la precisión con la que el cuadro delimita el objeto. Además, cada celda también predice las clases de los objetos que se encuentran en su región [16].

Por ejemplo, si la imagen se divide en una cuadrícula de  $7 \times 7$  celdas, cada celda de la cuadrícula puede predecir múltiples cuadros delimitadores y las clases correspondientes de

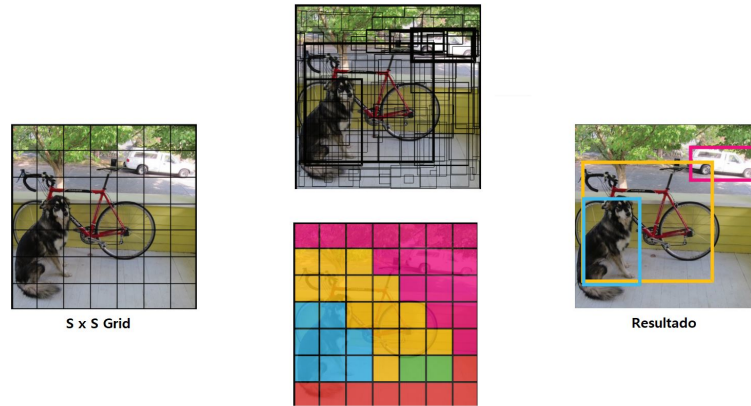


Figura 2: Modelo de como funciona el algoritmo YOLO

los objetos. A cada celda se le asigna la responsabilidad de predecir los objetos cuyo centro cae dentro de los límites de la celda. Esto reduce la complejidad del problema de detección al permitir que una región limitada de la imagen esté a cargo de hacer predicciones sobre un subconjunto de objetos.

## Cuadros Delimitadores y Ancas

En el modelo original de YOLO, cada celda de la cuadrícula predice un número fijo de cuadros delimitadores. Estos cuadros están definidos por cinco parámetros: las coordenadas  $x$  e  $y$ , que representan el centro del cuadro, la altura y el ancho del cuadro, y la puntuación de confianza. Sin embargo, las versiones posteriores de YOLO, como YOLOv2 y YOLOv3, introdujeron el concepto de *anchor boxes* para mejorar la precisión en la predicción de cuadros delimitadores.

Los *anchor boxes* son cuadros predefinidos de diferentes tamaños y proporciones, que permiten al modelo manejar de manera más eficaz la variabilidad en el tamaño y la forma de los objetos. Durante el entrenamiento, YOLO ajusta estos cuadros de ancla para que coincidan mejor con los objetos detectados en las imágenes de entrenamiento.

La introducción de *anchor boxes* redujo significativamente los errores de localización, especialmente en casos donde los objetos tenían diferentes tamaños o formas. Además, al utilizarlos, YOLO pudo manejar mejor la detección de múltiples objetos dentro de una misma celda de la cuadrícula.

## Clasificación de Objetos

Una vez que se han predicho los cuadros delimitadores, YOLO clasifica los objetos dentro de cada cuadro. Para hacer esto, el modelo asigna probabilidades a cada una de las clases predefinidas de objetos. Estas clases pueden variar ampliamente dependiendo del conjunto de datos utilizado. Por ejemplo, en el conjunto de datos COCO, YOLO es capaz de clasificar más de 80 clases de objetos, que incluyen personas, vehículos, animales y objetos cotidianos.

La clasificación se realiza utilizando una red neuronal completamente conectada que recibe como entrada las características aprendidas de las capas convolucionales anteriores. A partir de estas características, la red predice la clase más probable para cada objeto detectado en un cuadro delimitador.

La metodología utilizada en este trabajo de investigación para entrenar un modelo de detección de objetos basado en YOLOv8 sigue un enfoque estructurado que involucra varias etapas clave, desde la recolección y anotación de datos hasta la implementación de un sistema de detección en tiempo real mediante video. A continuación, se describen de manera detallada los pasos seguidos en cada fase del proceso.

### 6.1. Recoleccion de Datos

La primera fase consistió en la recolección de datos necesarios para entrenar un modelo de detección de objetos especializado en la clasificación de materiales reciclables. Se decidió utilizar un conjunto de datos de residuos y objetos comunes basados en el entorno doméstico para fomentar la clasificación y el reciclaje automatizado. En esta fase se realizaron las siguientes actividades:

#### 6.1.1. Obtencion del Conjunto de Datos TACO

El conjunto de datos inicial se obtuvo de la data set TACO, que es un data set público orientado a la clasificación de residuos. Este data set proporciona imágenes anotadas de residuos en diversos entornos. Las imágenes originales fueron recolectadas y etiquetadas para ayudar en la clasificación de materiales reciclables y no reciclables.

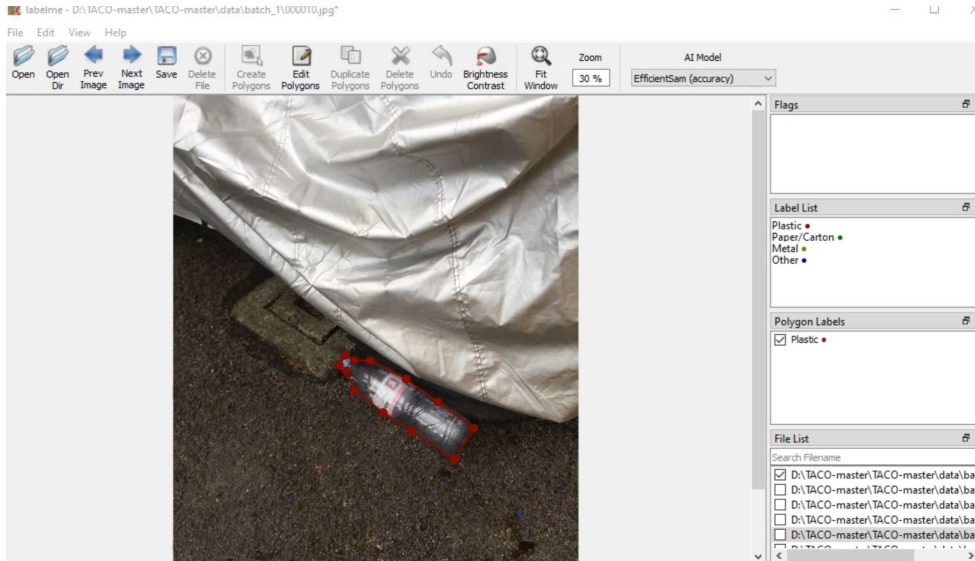


Figura 3: Etiquetando en *LabelMe*

### 6.1.2. Etiquetado de Datos con *LabelMe*

Para utilizar de manera eficaz los datos de TACO en nuestro modelo de YOLOv8, fue necesario realizar una anotación detallada de las imágenes. El *software LabelMe*, que es una herramienta de código abierto para la anotación de datos de imágenes, fue utilizado para etiquetar manualmente las categorías de los objetos presentes en las imágenes. Se establecieron inicialmente cuatro categorías para los objetos a clasificar:

- Metal
- Plástico
- Papel y cartón
- Otros (objetos que no encajaban claramente en las categorías anteriores)

Estas categorías fueron elegidas debido a la relevancia de estos materiales en el proceso de reciclaje y su frecuencia en los residuos domésticos.

## 6.2. Preprocesamiento de Datos

Una vez que se anotaron las imágenes utilizando *LabelMe*, se procedió a convertir las anotaciones en un formato compatible con el modelo de YOLOv8. Para lograr esta conversión, se utilizó la plataforma *Roboflow*, que permite gestionar de manera eficiente la transformación y manipulación de conjuntos de datos de imágenes para el entrenamiento de modelos de visión por computadora.

## Conversión del Formato

Las anotaciones generadas con *LabelMe* están en formato JSON, mientras que el modelo de YOLOv8 requiere las etiquetas en un formato específico, donde se define la clase del objeto y las coordenadas de los cuadros delimitadores. Por lo que *Roboflow* facilitó esta conversión de datos, transformando las anotaciones a un formato compatible con YOLOv8.

### 6.2.1. Creación del Conjunto de Entrenamiento

El conjunto de datos original, tras ser procesado en *Roboflow*, contenía 1,000 imágenes anotadas. Estas imágenes fueron distribuidas entre las cuatro categorías mencionadas. Sin embargo, tras una evaluación preliminar, se determinó que la cantidad de imágenes era insuficiente para entrenar un modelo robusto de YOLOv8. Esto llevó a resultados de baja precisión durante el entrenamiento del modelo inicial.

## 6.3. Ampliación del Conjunto de Datos

Para mejorar la precisión del modelo, fue necesario ampliar el conjunto de datos recolectando más imágenes. Se llevó a cabo una segunda fase de recolección y anotación, en la que se incrementó significativamente el número de imágenes por categoría.

### 6.3.1. Obtención de Imágenes Adicionales de *Kaggle*

Se recurrió a la plataforma *Kaggle*, que tiene una gran cantidad de datos públicos, para obtener más imágenes relacionadas con residuos y materiales reciclables. Se seleccionaron varios data sets relevantes que contenían imágenes de materiales similares a los del conjunto de TACO, enfocándose en las mismas categorías: metal, plástico, papel y cartón.

Al igual que en la fase inicial, las imágenes adicionales obtenidas de *Kaggle* también fueron anotadas manualmente utilizando *LabelMe*. Sin embargo, durante esta fase, se decidió eliminar la categoría Otros, ya que la falta de consistencia en esta categoría no aportaba valor al modelo y reducía su precisión. En lugar de clasificar objetos ambiguos como otros, se concentraron los esfuerzos en las tres categorías principales: metal, plástico, y papel-cartón.

Tras esta recolección adicional, el conjunto de datos final consistió en 3,000 imágenes, con aproximadamente 1,000 imágenes por categoría. Esta ampliación proporcionó una base de datos mucho más equilibrada y diversificada para el entrenamiento del modelo. Una vez recolectadas y anotadas las 3,000 imágenes, se siguió el mismo proceso inicial para preparar los datos para el entrenamiento del modelo YOLOv8. Luego se generaron en *Roboflow* una cantidad de imágenes transformadas con la transformación de *greyscale* para que el modelo no tome tanto en cuenta el color de los objetos y su iluminación. Como se puede ver en el cuadro 1 se utilizaron un total de 3,755 imágenes.

| División      | Cantidad |
|---------------|----------|
| Entrenamiento | 2817     |
| Prueba        | 317      |
| Validación    | 621      |

Cuadro 1: Cantidad total de imágenes por división

Además, es importante notar que dentro de estas 3,755 imágenes hay múltiples objetos presentes en cada una, como muestra la distribución de objetos por categoría en todas las divisiones. El cuadro 2 refleja que los objetos de papel y cartón están representados en una cantidad significativamente mayor (3,091) en comparación con los objetos de metal (1,899) y plástico (2,015).

| Clase          | Cantidad |
|----------------|----------|
| Papel y Carton | 3091     |
| Metal          | 1899     |
| Plastico       | 2015     |

Cuadro 2: Cantidad total de objetos por clase

Esta disparidad en la cantidad de objetos se explica al observar la densidad promedio y mediana de cada categoría en las imágenes. Según el cuadro 3, los objetos de papel-cartón presentan un promedio de 1.097 instancias por imagen. Esto es notablemente superior a los promedios de las otras clases ya que los objetos de metal tienen un promedio de 0.674 instancias y los de plástico 0.715.

## 6.4. Evaluación del Modelo YOLOv8

Una vez finalizado el proceso de entrenamiento, el modelo fue evaluado automáticamente utilizando el conjunto de prueba. Esta evaluación se realizó con la opción ‘plots=True’, que generó gráficas detalladas de las métricas de rendimiento directamente en el entorno de entrenamiento. Estas gráficas permiten un análisis visual e intuitivo del desempeño del modelo en imágenes no vistas previamente.

### 6.4.1. Métricas de Evaluación

YOLOv8 generó automáticamente las principales métricas de evaluación, facilitando la interpretación del rendimiento en cada categoría (metal, plástico y papel-cartón) y propor-

| Clase          | Promedio de Instancias |
|----------------|------------------------|
| Papel y Carton | 1.10                   |
| Metal          | 0.67                   |
| Plastico       | 0.72                   |

Cuadro 3: Cantidad total de objetos por clase



cionando una visión integral de la capacidad del modelo para detectar objetos de manera precisa. Las métricas clave incluyeron:

- **Matriz de Confusión:** La matriz de confusión detalla el rendimiento del modelo en cada clase, mostrando las predicciones correctas y los errores de clasificación entre categorías (falsos positivos y negativos). Este gráfico es fundamental para identificar patrones de error específicos, como confusiones frecuentes entre clases (por ejemplo, papel-cartón clasificado erróneamente como plástico).
- **Curva de *F1-Confianza*:** La curva de *F1-Confianza* muestra la relación entre la precisión y el *recall* del modelo en diferentes niveles de umbral de confianza. Esta curva permite observar cómo cambia el *F1-score*, una métrica que combina precisión y *recall* en un solo valor— a medida que se ajusta el umbral de confianza. Un *F1-score* alto indica que el modelo logra un buen equilibrio entre detectar correctamente los objetos y evitar falsos positivos. Esta curva es útil para seleccionar el umbral de confianza óptimo y evaluar el rendimiento general del modelo en la clasificación de objetos.

## 6.5. Implementación de Detección en Tiempo Real

Finalmente, se implementó el modelo en un entorno de video en tiempo real, utilizando una cámara para probar la detección en vivo de objetos reciclables. Para ello, se utilizó una cámara conectada directamente al sistema y la biblioteca *OpenCV* para capturar las imágenes en un flujo continuo. El modelo fue cargado y configurado para realizar predicciones en tiempo real sobre los objetos detectados en el video.

El sistema captura cada *frame* del video en tiempo real y realizaba una inferencia del modelo sobre cada uno. Para asegurar una precisión en las detecciones, se estableció un umbral de confianza de 0.60 en la función de predicción del modelo. Si el modelo detectaba objetos en el *frame*, éstos se marcaban automáticamente en el video con cuadros delimitadores que señalaban su ubicación y clasificación.



## 7.1. Resultados del Primer Modelo

El primer modelo fue entrenado utilizando el conjunto de datos inicial, que consistía en aproximadamente 1,000 imágenes totales distribuidas entre las categorías de metal, plástico, papel-cartón y otros. Sin embargo, la categoría Otros presentó problemas durante el entrenamiento, ya que era demasiado amplia y carecía de una definición clara, lo que llevó al modelo a confundirse con mayor frecuencia en la clasificación. Esta categoría fue posteriormente eliminada en versiones futuras del modelo, ya que no aportaba valor y afectaba negativamente la precisión general.

### 7.1.1. Evaluación con la Matriz de Confusión

La matriz de confusión para este primer modelo, como se muestra en la Figura 4, revela un desempeño subóptimo, especialmente en las clases de papel-cartón y metal. En particular, se observa que la clase Otros generó una cantidad considerable de errores, contribuyendo a una gran cantidad de falsos positivos y negativos debido a su falta de especificidad.

- **Papel-Cartón:** Esta clase fue clasificada correctamente solo en un 26 % de los casos, con un 35 % de confusión con el fondo y un 18 % de confusión con la clase de metal. Esto indica que el modelo tiene dificultades para distinguir objetos de papel-cartón de otras clases, especialmente en casos donde los objetos podrían compartir características visuales con el fondo.
- **Metal:** Esta clase obtuvo una precisión del 27 %, pero presentó un 20 % de confusión con la clase de plástico y un 9 % con el fondo. Esta tasa de error relativamente alta

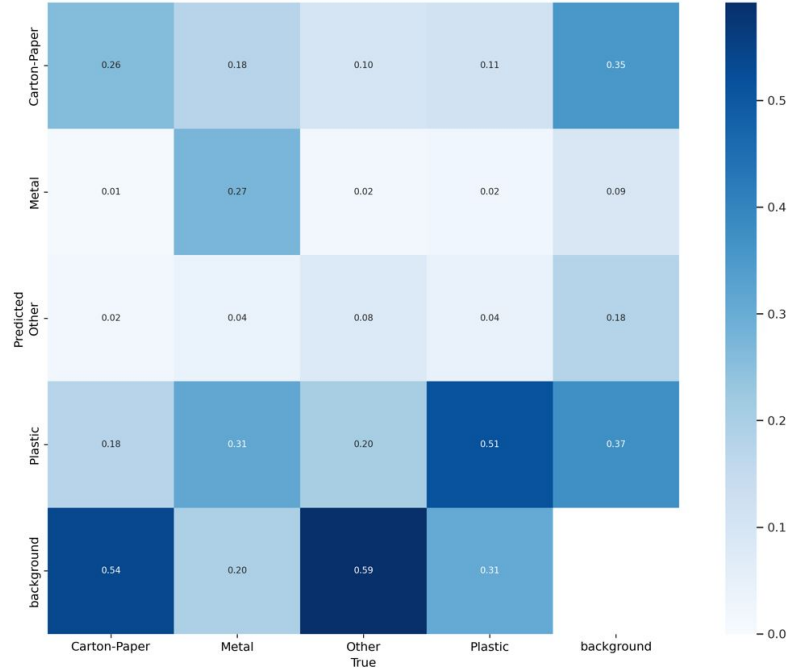


Figura 4: Matriz de Confusión del Primer Modelo

sugiere que el modelo no estaba optimizado para identificar correctamente objetos metálicos con el conjunto de datos inicial.

- **Plástico:** Esta clase fue la mejor clasificada, con una precisión del 51 %, en gran parte debido a su mayor representación en el conjunto de datos. Sin embargo, aún mostró un 37 % de confusión con el fondo y un 20 % con la clase de metal, lo que indica que el modelo podría estar sobre ajustado a las características visuales más comunes de esta clase, limitando su precisión en situaciones de variabilidad.
- **Otros:** Esta clase mostró un desempeño particularmente bajo, con una precisión del 4 % y confusiones significativas con todas las demás clases, especialmente con el fondo (18 %) y el plástico (8 %). Esta confusión resalta la falta de consistencia en esta categoría y justifica su eliminación en versiones futuras del modelo.

### 7.1.2. Análisis de la Curva de Confianza F1

En la Figura 5 se muestra la curva de confianza F1 del primer modelo, que evidencia el bajo rendimiento en términos de precisión y *recall*. La línea gruesa azul representa el *F1-score* promedio para todas las clases, alcanzando un máximo de aproximadamente 0.35 cuando la confianza es de 0.391. Este valor bajo refleja la necesidad de un conjunto de datos más equilibrado y bien definido para mejorar el rendimiento general del modelo.

- **Papel-Cartón:** Esta clase tiene un *F1-score* máximo cercano a 0.25, lo que confirma las dificultades del modelo para clasificar correctamente esta clase. La variabilidad visual de los objetos de papel y cartón contribuye a esta baja precisión.

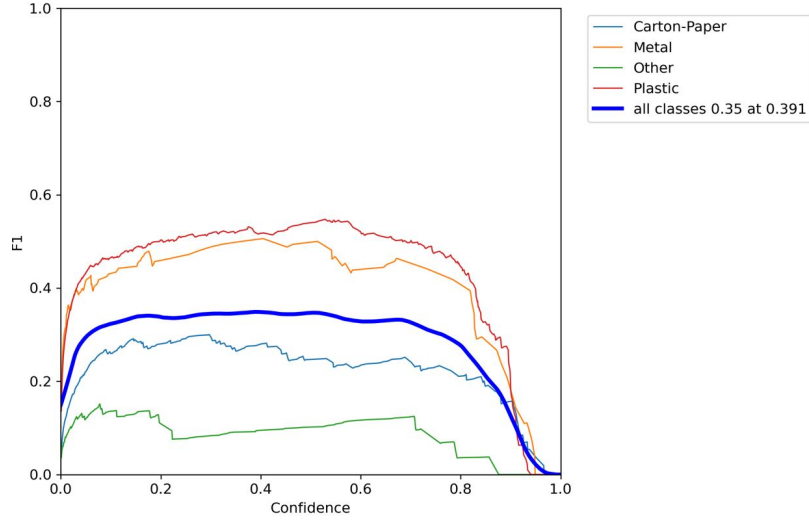


Figura 5: Curva de Confianza F1 del Primer Modelo

- **Metal:** Esta clase alcanza un  $F1$ -score máximo de aproximadamente 0.4, lo que indica un rendimiento inferior en comparación con el plástico. La baja disponibilidad de ejemplos y la variabilidad de los objetos metálicos probablemente influyen en este rendimiento deficiente.
- **Plástico:** Esta clase muestra el mejor  $F1$ -score con valores cercanos a 0.5, lo cual se debe en gran medida a su mayor cantidad de ejemplos en el conjunto de datos. Sin embargo, este rendimiento no es óptimo, ya que la precisión disminuye notablemente a medida que se eleva el umbral de confianza.
- **Otros:** Esta clase tiene el rendimiento más bajo, con un  $F1$ -score máximo de aproximadamente 0.15, evidenciando problemas de precisión y *recall* significativos. La inconsistencia de esta categoría confunde al modelo, lo cual motivó su eliminación en posteriores experimentos.

## 7.2. Resultados del Segundo Modelo

Luego de conseguir las nuevas imágenes y realizar los cambios necesarios, los resultados del segundo modelo mostraron mejoras significativas en todas las métricas de evaluación, especialmente en la detección de las clases de metal y papel-cartón. Con el conjunto de datos ampliado y la eliminación de la categoría Otros, el modelo pudo aprender patrones más consistentes en cada categoría, reduciendo los errores de clasificación y mejorando tanto la precisión como el *recall* en todas las clases.

### 7.2.1. Evaluación con la Matriz de Confusión

La matriz de confusión del segundo modelo, mostrada en la Figura 6, revela una mejora notable en la precisión de clasificación para cada clase en comparación con el primer modelo.

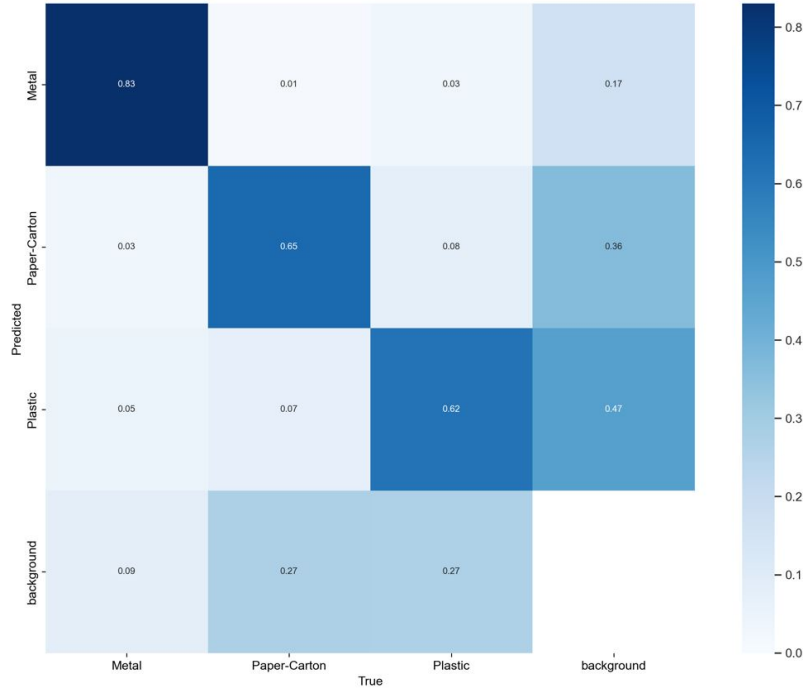


Figura 6: Matriz de Confusión del Segundo Modelo

- **Papel-Cardón:** La precisión en esta clase aumentó al 65 %, aunque todavía existe un 36 % de confusión con el fondo y un 8 % con la clase de plástico. Esto indica que, aunque el modelo ha mejorado en la detección de papel-cartón, todavía existen desafíos para diferenciar esta clase del fondo en algunas condiciones.
- **Metal:** Esta clase alcanzó una precisión del 83 %, lo cual representa una mejora significativa en comparación con el primer modelo. Solo un 17 % de los objetos de metal fueron clasificados incorrectamente como fondo, indicando que el modelo ha aprendido a diferenciar mejor las características de los objetos metálicos.
- **Plástico:** La clase de plástico también mejoró, logrando una precisión del 62 %. A pesar de esto, todavía se observa un 47 % de confusión con el fondo, lo cual sugiere que algunas características de los objetos de plástico aún resultan difíciles de diferenciar del fondo en ciertas imágenes.

La eliminación de la categoría .ºtrosz el balance de ejemplos en las demás clases contribuyeron a mejorar la precisión general del modelo, especialmente en las clases de metal y plástico.

### 7.2.2. Análisis de la Curva de Confianza F1

La curva de confianza F1 del segundo modelo, mostrada en la Figura 7, refleja una mejora general en el rendimiento en comparación con el primer modelo. La eliminación de la categoría .ºtrosz el aumento de ejemplos por clase contribuyeron a mejorar tanto la precisión como el *recall*.

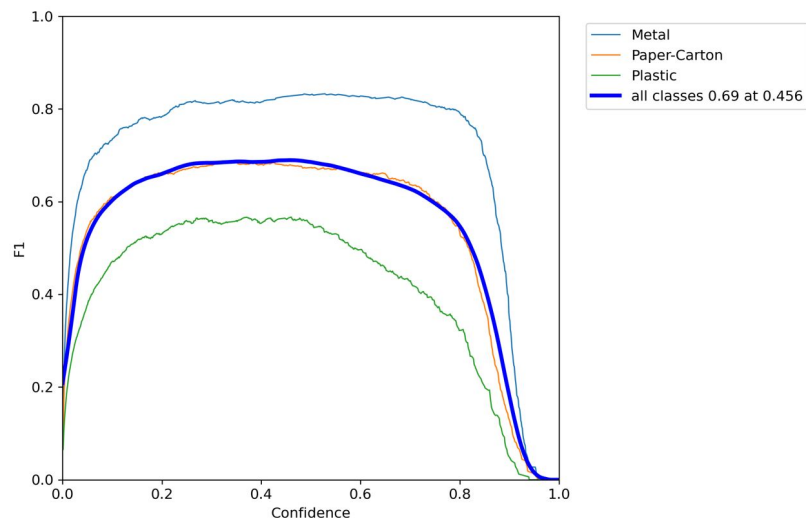


Figura 7: Curva de Confianza F1 del Segundo Modelo

- **Papel-Cardón:** La clase de papel-cartón mostró un *F1-score* máximo de aproximadamente 0.6, lo cual representa una mejora en comparación con el primer modelo. Sin embargo, la variabilidad en los objetos de papel-cartón sigue siendo un desafío, lo que se refleja en una menor precisión en niveles de confianza más altos.
- **Metal:** La clase de metal alcanzó el *F1-score* más alto, llegando a valores cercanos a 0.8, lo que indica un alto nivel de precisión y *recall* en esta categoría. Esta mejora se debe a la mayor consistencia de los ejemplos y a una mejor representación de esta clase en el conjunto de datos.
- **Plástico:** La clase de plástico mostró un *F1-score* máximo cercano a 0.5. Aunque ha mejorado en relación al primer modelo, todavía presenta un rendimiento inferior al de las otras clases, especialmente debido a la confusión con el fondo.

La línea gruesa azul representa el *F1-score* promedio de todas las clases, alcanzando un valor máximo de 0.69 cuando la confianza es de 0.456. Este valor indica un mejor balance entre precisión y *recall* en comparación con el primer modelo, lo que refleja las mejoras realizadas en el conjunto de datos y en el proceso de entrenamiento.





El desarrollo de un sistema de detección de residuos reciclables mediante el modelo YOLOv8 ha sido un proceso con varias fases críticas, desde la recolección de datos hasta su implementación en tiempo real. Este proyecto demuestra la importancia de contar con conjuntos de datos adecuados, precisión en la anotación de imágenes y herramientas eficaces para el preprocesamiento de datos.

Se logró cumplir los objetivos de entrenamiento, ajuste y evaluación del modelo, optimizándolo para alcanzar una alta precisión en la clasificación de residuos en tiempo real, integrando una cámara para capturar imágenes de los residuos.

En conclusión, se diseñó y desarrolló un sistema de identificación de residuos basado en visión por computadora, capaz de analizar y clasificar objetos en tiempo real, contribuyendo así a la automatización del proceso de clasificación de residuos en sus contenedores apropiados. El sistema alcanzó una precisión promedio de aproximadamente el 70 %, cumpliendo con los objetivos planteados.



---

### Recomendaciones

---

Con base en los resultados obtenidos y los desafíos encontrados durante este proyecto, se sugieren varias recomendaciones para mejorar el rendimiento del sistema y su aplicabilidad en escenarios reales.

En primer lugar, se recomienda ampliar el conjunto de datos, recolectando un mayor número de imágenes en una variedad más amplia de entornos y condiciones, como diferentes tipos de iluminación, ángulos y contextos. Esto permitiría al modelo mejorar su capacidad de generalización y su precisión en situaciones más diversas. Además, no solo debe considerarse la cantidad de imágenes, sino también la cantidad total de objetos dentro de cada imagen. Un conjunto de datos con múltiples objetos en cada imagen ayudaría a entrenar al modelo en detecciones más complejas y mejoraría su estabilidad en la clasificación.

También sería beneficioso explorar el uso de técnicas avanzadas de aumento de datos, como la generación de imágenes sintéticas mediante redes generativas antagónicas *GANs*. Estas técnicas permiten crear ejemplos adicionales que simulan situaciones complejas o inusuales que podrían no estar presentes en el conjunto de datos original, como la superposición de objetos, distintos niveles de deterioro o condiciones de poca visibilidad. El uso de *GANs* no solo incrementaría la variedad de los datos, sino que también fortalecería la robustez del modelo frente a escenarios impredecibles.

Por último, se recomendaría no solo tomar en cuenta cantidad de imágenes, pero cantidad de objetos totales en las imágenes para mantener un entrenamiento mas estable



- 
- [1] United Nations Environment Programme, *Global Waste Management Outlook 2024: Beyond an Age of Waste – Turning Rubbish into a Resource*, 2024. dirección: <https://www.unep.org/resources/global-waste-management-outlook-2024>.
  - [2] K. Huang, H. Lei, Z. Jiao y Z. Zhong, «Recycling waste classification using vision transformer on portable device,» *Sustainability*, vol. 13, n.º 21, pág. 11 572, 2021. DOI: 10.3390/su132111572.
  - [3] M. Mudemfu, *Intelligent Solid Waste Classification System Using Deep Learning*, 2023.
  - [4] Y. LeCun, Y. Bengio y G. Hinton, «Deep learning,» *Nature*, vol. 521, n.º 7553, 2015.
  - [5] S. Russell y P. Norvig, *Artificial Intelligence: A Modern Approach, Global Edition*. Pearson, 2021.
  - [6] J. Heaton, «Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning: The MIT Press, 2016, 800 pp, ISBN: 0262035618,» *Genetic Programming and Evolvable Machines*, vol. 19, n.º 1–2, 2018.
  - [7] T. Mitchell, *Machine Learning*. New York: McGraw Hill, 1997.
  - [8] T. Mikolov, K. Chen, G. Corrado y J. Dean, «Efficient estimation of word representations in vector space,» en *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*, 2013.
  - [9] S. García, J. Luengo y F. Herrera, *Data Preprocessing in Data Mining* (Intelligent Systems Reference Library). Springer, 2015, vol. 72.
  - [10] R. Szeliski, *Computer Vision: Algorithms and Applications*. Springer Science & Business Media, 2010.
  - [11] J. Ponce y D. Forsyth, *Computer Vision: A Modern Approach*. Pearson, 2012.
  - [12] R. Gonzalez y R. Woods, *Digital Image Processing*, 3rd. Pearson International Edition, 2008.
  - [13] J. Shi y J. Malik, «Normalized cuts and image segmentation,» *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, n.º 8, 2000. DOI: 10.1109/34.868688.

- [14] A. Krizhevsky, I. Sutskever y G. Hinton, «ImageNet classification with deep convolutional neural networks,» *Communications of the ACM*, vol. 60, n.º 6, 2017. DOI: 10.1145/3065386.
- [15] J. Redmon, S. Divvala, R. Girshick y A. Farhadi, «You Only Look Once: Unified, Real-Time Object Detection,» *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [16] J. Redmon, S. Divvala, R. Girshick y A. Farhadi, «You Only Look Once: Unified, Real-Time Object Detection,» *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pág. 2, 2016.

Github: <https://github.com/jwlh00/TrashClassification>



Figura 8: Detección de una botella de agua



Figura 9: Detección de una soda *Tiky*



Figura 10: Detección de una botella de agua





Figura 11: Detección de una lata de Coca



Figura 12: Detección de una lata de Agua Mineral



Figura 13: Detección de caja de Capuchino

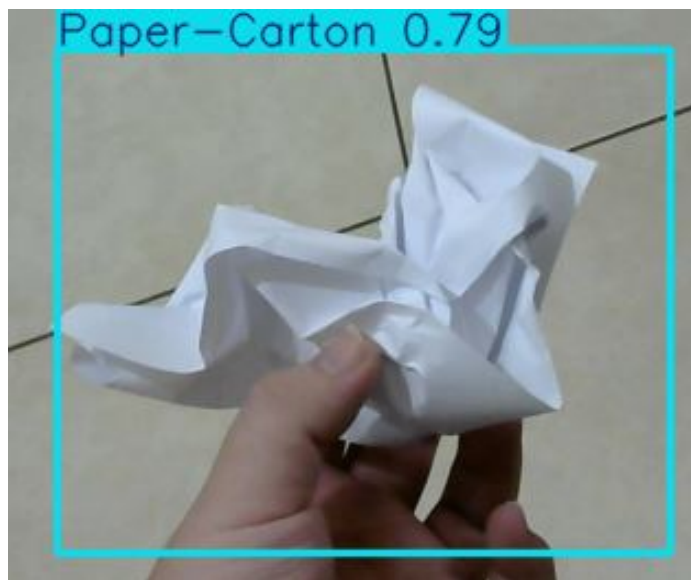


Figura 14: Detección de un papel arrugado