

Desarrollo de un Algoritmo mediante Redes Neuronales Convolucionales para la clasificación de residuos sólidos en universidades

Ryan A. LEÓN LEÓN

La Libertad – Universidad Privada del Norte
Trujillo, 13006, Perú

Karen D. BARRETO ESPEJO

La Libertad – Universidad Privada del Norte
Trujillo, 13006, Perú

Katherine J. CABANILLAS VIGO

La Libertad – Universidad Privada del Norte
Trujillo, 13006, Perú

Joffry S. MUÑOZ CHACÓN

La Libertad – Universidad Privada del Norte
Trujillo, 13006, Perú

Saúl A. PEÑA MARTINEZ

La Libertad – Universidad Privada del Norte
Trujillo, 13006, Perú

Astrid E. TAFUR SEPÚLVEDA

La Libertad – Universidad Privada del Norte
Trujillo, 13006, Perú

RESUMEN: El presente trabajo de investigación puso en práctica el uso de un algoritmo mediante redes neuronales convolucionales, para llevar a cabo una correcta clasificación de residuos sólidos en universidades. Para ello se hizo uso del lenguaje de programación de Python, el cual permitió manejar con facilidad el lenguaje de programación de la red neuronal YOLOv8, que es el modelo más avanzado de la serie YOLO. Por otro lado, en su desarrollo el algoritmo trabaja con la información de una Dataset de 1203 imágenes recopilados en Roboflow, la cual es utilizada en la etapa de entrenamiento, donde la red se dedica a aprender y luego lograr detectar el tipo de residuo, cabe resaltar que el modelo trabaja a un nivel de confianza de 0.94. Por último, los resultados indican que el valor de precisión del modelo para clasificar el tipo de residuo es de 84.6% en cartón, 97.1% en madera, 94.2% en papel y 96.4% en plástico; logrando una precisión global de 90.6%. Es así, se puede afirmar que, la codificación y entrenamiento de la red neuronal YOLOv8 fue exitosa, permitiendo al algoritmo detectar y clasificar con eficacia el tipo de residuo (plástico, papel, cartón y madera) presentes en las universidades.

Palabras Claves: YOLOv8, residuos, universidades, red neuronal, Python.

Development of an Algorithm using Convolutional Neural Networks for the classification of solid waste in universities.

ABSTRACT: This research work put into practice the use of an algorithm using convolutional neural networks to carry out a correct classification of solid waste in universities. For this purpose, the Python programming language was used, which made it possible to easily handle the programming language of the YOLOv8 neural network, which is the most advanced model of the YOLO series. On the other hand, in its development the algorithm works with the information of a Dataset of 1203 images collected in Roboflow, which is used in the training stage,

where the network is dedicated to learn and then manage to detect the type of residue, it should be noted that the model works at a confidence level of 0.94. Finally, the results indicate that the accuracy value of the model to classify the type of waste is 84.6% for cardboard, 97.1% for wood, 94.2% for paper and 96.4% for plastic; achieving an overall accuracy of 90.6%. Thus, it can be affirmed that the coding and training of the YOLOv8 neural network was successful, allowing the algorithm to effectively detect and classify the type of waste (plastic, paper, cardboard and wood) present in the universities.

Keywords: YOLOv8, residuals, universities, neural network, Python.

1. INTRODUCCIÓN

Es sabido que el manejo y clasificación de residuos mejora la calidad y condiciones ambientales de nuestro entorno, como también la salud humana. Por ello es necesario que las personas tengan conocimiento sobre la clasificación de residuos, pero es algo que no suele verse en este contexto ya que es un problema que afecta a nivel mundial. Al igual que el mundo, el Perú también se ve afectado por este problema, que involucra al sector económico, la vida cotidiana, el crecimiento de las ciudades, etc. Por otro lado, también se estima que las instituciones, tanto públicas como privadas son responsables de la contaminación que producen estos residuos, por lo que estas también deberían aportar alguna solución a esta problemática. Entre las entidades con mayor impacto se encuentran las universidades que, debido a la gran afluencia de personas, generan grandes cantidades de residuos que son desechados diariamente.

En el ámbito de la clasificación de residuos sólidos mediante redes neuronales en entornos universitarios, una variedad de estudios ha demostrado avances significativos en precisión y

aplicabilidad. Por ejemplo, el estudio [1] reveló una precisión del 80% con YOLOv3, destacando la efectividad de los modelos avanzados en la clasificación de residuos para contenedores inteligentes. Asimismo, investigaciones como las [2] y [3] mostraron respectivamente precisiones del 54% y hasta el 83%, ilustrando tanto los desafíos como las posibles mejoras mediante el uso de ensamblajes de redes neuronales en la clasificación de imágenes. Por otro lado, estudios con altas precisiones como los [4] y [5], que alcanzaron un 95.2% en la clasificación de plásticos residuales, junto con [6] con un 95% en la selección de empaques, enfatizan el potencial de los modelos YOLOX y sistemas de visión artificial en entornos controlados. Igualmente, trabajos como los [7] y [8], con precisiones cercanas al 93%, refuerzan cómo la integración de tecnología avanzada puede mejorar la infraestructura de reciclaje en las universidades. Finalmente, investigaciones como la [9] con un 85.2%, y [10] con un 89.3%, demuestran la utilidad de las redes neuronales en categorías específicas de residuos, como orgánicos y sólidos, respectivamente. Además, el estudio [11] con un 93% de precisión, muestra cómo las soluciones automatizadas pueden implementarse para una gestión más eficiente de residuos sólidos, subrayando la relevancia de la tecnología en la sostenibilidad ambiental en las universidades. Estos estudios colectivamente destacan la importancia de continuar el desarrollo y optimización de tecnologías de clasificación de residuos, adaptando soluciones que puedan responder efectivamente a los desafíos ambientales contemporáneos. Actualmente, los residuos sólidos son desechos que no aportan valor ya que representa un desafío significativo a nivel mundial, su clasificación antes de su reutilización o reciclaje constituye un paso clave hacia la administración sostenible de los residuos [12]. Para abordar este problema, se propone desarrollar un algoritmo en Python para clasificar residuos sólidos. Python es un lenguaje de programación flexible y de código abierto, reconocido por su extensa variedad de bibliotecas y herramientas especializadas en el procesamiento y gestión de datos [13]. Un componente para la programación en Python es la red neuronal convolucional (CNN) que es un sistema de aprendizaje supervisado que vincula datos de entrada como imágenes, de salida como clases o categorías y puede ajustar su ponderación utilizando el proceso de retroprogramación con el objetivo de mejorar la exactitud en la clasificación del modelo que está siendo entrenado [14]. Para completar todos los elementos para realizar el algoritmo, se planteó utilizar la biblioteca de aprendizaje automático PyTorch ya que demuestra que dos objetivos pueden ser compatibles ofreciendo un estilo de programación imperativo y que su interoperabilidad es fácil y eficiente ya que permite a los usuarios aprovechar el amplio ecosistema de bibliotecas de Python en sus programas.[15]. En la investigación [16], se tuvo como objetivo proponer un sistema eficiente de detección y clasificación de residuos en una cinta móvil en instalaciones de recolección de desechos por medio de una red neuronal convolucional. Para ello fue utilizado un “modelo adaptativo” y una metodología de diez etapas: requisitos del sistema, planificación del proceso, clasificación de uso, desarrollo de soluciones propuestas, elección de solución, diseño de solución, aplicación piloto, evaluación del sistema, implementación y mantenimiento. Asimismo, como base de datos se empleó TACO con un total de 1500 imágenes para entrenar al detector y como modelo de detección se seleccionó a la red YOLOv4. Posterior a su implementación se obtuvo una precisión de 92.43% a partir de vídeos de detección en tiempo real y un IoU promedio del 63,58%. Se concluyó al utilizar pequeños dispositivos integrados previamente capacitados, es posible lograr un bajo costo y complejidad de implementación. Por su parte, en la investigación

[17], se tuvo como objetivo desarrollar un modelo personalizado de detección y clasificación de objetos de desecho para un contenedor de basura inteligente. Para este modelo se empleó y combinó el algoritmo YOLO de última generación que utiliza un método basado en redes neuronales convolucionales con el algoritmo inspirado en la naturaleza para la detección y clasificación de objetos de desecho. El modelo YOLO fue entrenado con diferentes pesos, para Yolov3, Yolov4 y Yolov4-tiny. Se utilizan ocho clases de residuos, 3171 imágenes de residuos y se consideraron un total de 3171 imágenes de desperdicio. Adicionalmente, la estructura algorítmica subyacente para el funcionamiento del contenedor de basura inteligente se implementó en Python. Los resultados experimentales muestran que la combinación del parámetro de tasa de aprendizaje KSA de 0,0007 y Yolov3 se identifica como el modelo más preciso para la detección y clasificación de objetos de desecho, con una precisión del 100% en detectar imágenes de lata, 97% de precisión en detectar imágenes de periódicos y un 99% de precisión en detectar bolsas de plástico.

Actualmente generar conciencia colectiva es un proceso complejo e iniciar por las universidades resulta importante para impulsar el cambio. De igual forma, conocer la correcta clasificación de los residuos y realizar una buena segregación de estos, resulta ser uno de los primeros pasos para una correcta gestión de los residuos. Este proceso puede considerarse tedioso y difícil de aprender en un inicio, es por ello por lo que el apoyo de un programa para identificar los diversos residuos y la clasificación de estos será crucial para un adecuado manejo. Por lo tanto, el presente estudio tiene como principal objetivo desarrollar un algoritmo mediante redes neuronales convolucionales para la clasificación de residuos sólidos en universidades. Así mismo, se cuenta con objetivos específicos como la recopilación y preparación de una dataset representativa de los diferentes tipos de residuos sólidos, en este caso (cartón, madera, plástico, papel). También tenemos la codificación y entrenamiento de la red neuronal YOLOv8 para que logre detectar e identificar el tipo de residuo presente en la imagen y finalmente tenemos la posterior evaluación de su funcionamiento mediante métricas de precisión y sensibilidad para lograr un reconocimiento exitoso.

2. MATERIALES Y MÉTODOS

A continuación, se da a conocer los materiales necesarios para el desarrollo del algoritmo, mediante redes neuronales convolucionales. Con respecto a la recolección de las imágenes, se necesitan cientos de imágenes para el entrenamiento y validación de éstas, para ello, se debe tener fotografías en donde se pueda observar la conserva los distintos tipos de residuos sólidos (papel, plástico, madera y cartón) que son desechados en las universidades, las fotos deben ser claras para que le permitirá al algoritmo diferenciarlas, cabe aclarar, que las imágenes han sido tomadas dentro de la universidad y también se ha hecho uso data set exterior. Por otro lado, es necesario el manejo de una cámara que cuente con una buena resolución para poder utilizarla en el sistema, de modo que se encargará de capturar y posteriormente le dará cabida a la codificación programada mediante el algoritmo para la identificación de los residuos sólidos. Así mismo el algoritmo deberá ejecutarse en un ordenador portátil que cuente con una capacidad de procesamiento aceptable y un sistema operativo compatible para el funcionamiento del programa Python, el cual se apoyará en

Anaconda. En la (Tabla 1), se describe con mayor detalle lo necesario para el desarrollo de la codificación.

Tabla 1. Componentes para la codificación del algoritmo

Componente	Especificación
Procesador	hp
RAM	16 gb
Sistema	Windows 11 Pro
Python	Versión 3.8
Anaconda	Software
Roboflow	Plataforma web
YOLO	Versión 8
Librería	Ultralytics
	Open Cv (Cv2)
	Numpy
	OS

Las redes neuronales convolucionales se están convirtiendo en herramientas esenciales para detectar patrones ocultos en las características visuales. En los últimos años, las redes neuronales convolucionales se han utilizado cada vez más en diversos sistemas inteligentes de detección y reciclaje de basura. [18]. Dentro de la arquitectura de redes neuronales más conocidas en el campo de la detección y localización de objetos dentro de una imagen, esta YOLOv8, dicha modelo está conformada por capas convolucionales seguida de capas completamente conectadas para realizar la predicción de las distintas clases y delimitar los cuadros delimitadores [19]. Además, YOLOv8 es el modelo más avanzado de la serie YOLO, siendo un algoritmo de detección de objetos muy eficiente, en tiempo real y de una sola etapa que equilibra la precisión de reconocimiento y la velocidad de detección de objetos [20]. En la detección de objetos mediante imágenes, la técnica consiste en dividir la imagen en una cuadrícula de celdas de tamaño uniforme (k*k). Cada celda se encarga de detectar objetos cuyos centros caigan dentro de su área. Estas celdas pueden predecir varios recuadros delimitadores que incluyen las coordenadas del centro (x, y), las dimensiones del recuadro (ancho y alto) y un valor de confianza que indica la probabilidad de presencia del objeto dentro del recuadro. YOLO utiliza la Intersección sobre Unión (IOU) para seleccionar el mejor cuadro delimitador representativo de un objeto dentro de una celda de la cuadrícula, después de predecir un cuadro delimitador, además, se utiliza la supresión no máxima para eliminar los cuadros delimitadores redundantes. [21]. La arquitectura de YOLOv8 puede separarse en dos componentes principales: la columna vertebral y la cabeza.

YOLOv8 procesa de manera homogénea las 4 categorías de residuos que se están analizando, como lo son la madera, el papel, el plástico y el cartón. Por lo tanto, se procede a detallar una explicación para los cuatro tipos de residuos mencionados anteriormente. El modelo YOLOv8 utiliza CSPDarknet53 como estructura subyacente, una sofisticada red neuronal que extrae características a varias escalas. [22].

La parte inicial (backbone) es responsable de la extracción de características de la imagen de entrada. Está compuesto por múltiples capas convolucionales, cada una de las cuales incluye una capa de convolución seguida de una capa de normalización por lotes y una función de activación de SiLU. Dentro de cada capa de convolución se aplica un kernel a la imagen de entrada. Los kernels vienen siendo las longitudes de los filtros aplicados en las capas convulsionales, que se encarga de la detección de las características locales en la imagen, como bordes y texturas.

Cada capa convolucional puede tener múltiples kernels para detectar diferentes características. Un kernel es una pequeña matriz de pesos que se desplaza sobre la imagen, realizando una operación de producto punto en cada posición, suelen ser de tamaño pequeño 3x3 o 5x5, que son los más comunes en arquitecturas modernas, proporcionando equilibrio dentro del campo receptivo y costos computacionales. Un kernel de 3x3, por ejemplo, permite capturar detalles finos y preservar una alta resolución en las capas iniciales de la red, mientras que uno de 5x5 puede capturar características más complejas a medida que se avanza en las capas más profundas. En esta investigación, se han utilizado específicamente kernels de tamaños 1x1 y 3x3. El kernel de 1x1, aunque pequeño, es poderoso y versátil, ya que permite realizar transformaciones de las características sin cambiar la dimensión espacial de la imagen. Se usa a menudo para reducir la dimensionalidad, combinar características o aplicar no linealidades adicionales sin aumentar significativamente el costo computacional. Por otro lado, el kernel de 3x3 es utilizado para capturar relaciones espaciales y textuales de mayor complejidad, manteniendo un bajo costo computacional y proporcionando un equilibrio óptimo en términos de rendimiento y eficiencia.

En cuanto a la función de activación aplicadas a las activaciones normalizadas, se emplea la función SiLU que se utiliza en las capas de convolución para mejorar la capacidad del modelo para detectar objetos. Esta función combina propiedades de funciones lineales y no lineales permitiendo a la red que aprenda representaciones más complejas de datos. La función SiLU se define matemáticamente tal como se muestra a continuación:

$$\text{SiLU}(x) = x \cdot \sigma(x)$$

Donde $\sigma(x)$ es la función sigmoide.

$$\sigma(x) = \frac{1}{(1+e^{-x})}$$

En la (Figura 1), se observa que la curva de SiLU muestra cómo la función combina linealidad para valores altos y saturación sigmoide para valores bajos. Esto permite una activación más suave y continua, lo que puede mejorar la capacidad del modelo para aprender patrones complejos. Por el contrario, ReLU es lineal para valores positivos y cero para valores negativos. Es simple y efectiva, pero puede llevar a que las neuronas dejen de activarse durante el entrenamiento.

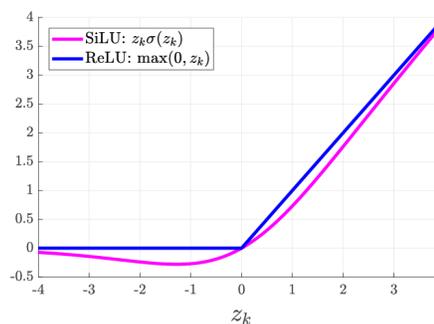


Figura 1. Función de activación SiLU con la función ReLU

Al final de la columna vertebral, YOLOv8 incorpora una capa de agrupación rápida de pirámide espacial (SPPF), cuyo objetivo principal de esta capa es acelerar el cálculo agrupando características de varias escalas en un mapa de características de tamaño uniforme. Una capa SPPF se compone de una capa convolucional seguida de tres capas de agrupamiento máximo.

La salida de cada capa de agrupamiento se agrega a una salida unificada. Además, se añade una capa convolucional adicional. Por el contrario, la cabeza del modelo YOLOv8 se ocupa por separado del proceso de clasificación y de la identificación de objetos. Además, cabe destacar que la cabeza del modelo no utiliza anclajes, lo que simplifica el proceso de entrenamiento en varios conjuntos de datos. Así mismo, en la última capa de la cabeza, se utiliza la función softmax para determinar la probabilidad de clasificación de cada clase, que se encarga de convertir las predicciones de clase en probabilidades, permitiendo al modelo determinar con qué confianza cree que una región contiene un objeto de una clase específica. Por último una ventaja que posee YOLOv8 es que incorpora la supresión suave no máxima (NMS) en lugar de la NMS convencional. Esto implica que cuando los cuadros delimitadores se solapan, no se eliminan por completo, sino que conserva la información del objeto.

Para la dataset, una vez completado la recolección de las imágenes necesarias para empezar con la programación del algoritmo, se hizo uso de Roboflow que es una plataforma de visión por ordenador que permite a los clientes entrenar y desplegar modelos de detección de objetos en tiempo real [23]. Esa plataforma web permitirá para identificar visualmente la ubicación de los distintos tipos de basura en las fotografías y así poder crear etiquetas para cada categoría de residuos (Madera, cartón, papel y plástico). Técnicamente esta actividad consistió en identificar determinadas regiones dentro de la imagen que corresponden a distintas categorías de basura y asignarles la etiqueta correspondiente. Este proceso de anotación es esencial para entrenar un modelo de red neuronal convolucional, ya que la precisión de dichas anotaciones repercute directamente en la capacidad del modelo para comprender con exactitud los rasgos distintivos de cada categoría de basura. Tras el proceso de anotación y mejora de la imagen se obtuvo cuatro carpetas que integraron el conjunto de datos final para nuestro proyecto, siendo 943 de papel, 583 plástico, 253 de cartón y 1426 de madera. El proceso de etiquetado en Roboflow se puede observar mejor en la (Figura 2).



Figura 2. Etiquetado de imágenes en Roboflow

Codificación y entrenamiento son las etapas más importantes en todo el proceso de desarrollo del algoritmo, y para ello se hizo uso del lenguaje de programación de Python. Primeramente, se creó un ambiente nuevo en Anaconda, que viene a ser una distribución libre y abierta del lenguaje de programación, luego se realizó la importación de librerías de YOLOv8 de ultralytics en la script de Python, de igual manera se importó la librería (OS) que sirve para cargar archivos al ambiente y (cv2) sirve para cargar las imágenes, convertirlas de color a escalas gris, detectar contornos y utilizar estos puntos de contorno para llevar a cabo

la transformación en perspectiva de la imagen proporcionada por el usuario [24]. Como se visualiza en la (Tabla 2).

Tabla 2. Códigos de importaciones de librerías

Librerías	Código de importación
Ultralytics	from ultralytics import YOLO
(OS)	import os HOME = os.getwd () Print ("HOME :", HOME)

Luego se procedió a organizar nuestra data set estructurada para el entrenamiento y validación, asegurándose de que cada imagen tenga un archivo de etiqueta correspondiente en formato YOLO. Posteriormente se define las rutas del modelo, donde se procede a cargar el modelo con los pesos base para entrenar, esto son descargados de yolov8; también se realiza la ruta de la configuración de la Dataset, donde se crea un archivo YAML que será el encargado de especificar la ruta de los conjuntos de datos y las clases, siendo esta el núcleo de codificación, como se muestra en la (Tabla 3).

Tabla 3. Definición de rutas

Rutas	Código
Rutas del modelo y la ruta de la configuración de la DATASET	model_path = os.path.join(HOME, "PESOS", "yolov8m.pt") data_path = os.path.join(HOME, "DATASET", "data.yaml")

A partir de la ruta se crea el modelo, y por último procedimiento se configura el entrenamiento, donde se configura el número de épocas, el número de workers, el tamaño de la imagen y el optimizador, que en este caso se colocó en automático para que el mismo programa encuentre los parámetros más apropiados para entrenar, como por ejemplo el learning rate y el momentum. Mostrado en la (Tabla 4).

Tabla 4. Configuración de entrenamiento del modelo

Configuración del modelo	model train(data=data_path, epochs=60, patience=10, batch=4, workers=2, imgsz=640, save period=2, optimize=True, erasing=0.4)
--------------------------	---

Por último, se inicia con el entrenamiento del modelo YOLOv8, aquí se hizo una muy detallada observación de los registros y gráficos que se generaron durante el entrenamiento, esto con el fin de poder monitorear la pérdida y la precisión en cada imagen.

Una vez completado el proceso de entrenamiento, se procede a evaluar el modelo, mediante el uso de métricas de rendimiento, esto se mide ejecutando el modelo en el conjunto de datos de prueba. Se utilizó la precisión, sensibilidad y precisión promedio media (mAP) como métricas. Estas métricas ayudan a evaluar la efectividad de un modelo de detección de objetos en términos de cuán precisas y completas son sus predicciones [25]. En la Ec. (1), se calcula la precisión que mide que tan exactas son las predicciones del modelo, es decir, todas las detecciones que el modelo identificó como positivas (objetos) cuantas son correctas. Las épocas representadas muestran valores de 0,187 y 0,26, que representan la precisión del modelo a la hora de determinar con exactitud la ubicación de los objetos. Esta precisión es de suma

importancia para la eficacia global del proyecto de clasificación de basura.

$$\text{Precision (P)} = \frac{TP}{TP+FP} \dots\dots\dots (1)$$

En la Ec. (2), se calcula la sensibilidad del modelo para encontrar todos los objetos reales en las imágenes. Los valores de 0,26 y 0,33 indican que el modelo es capaz de identificar con precisión una fracción sustancial de los artículos, pero aún hay potencial para seguir mejorando.

$$\text{Recall (R)} = \frac{TP}{TP+FN} \dots\dots\dots (2)$$

En la Ec. (3), se calcula la precisión promedio media (mAP) se determina calculando la media de los valores de precisión media (AP) de todas las clases. Los valores de mAP50, concretamente 0,183 y 0,173, sugieren que el modelo tiene una capacidad moderada para identificar y localizar con precisión los restos sólidos. Los valores 0,0835 y 0,0711 representan el rendimiento del modelo a la hora de reconocer y encontrar con exactitud los residuos dentro de un rango de precisión más riguroso y preciso. El análisis se ve en la figura 3.

$$mAP = \frac{1}{n} \sum_{i=1}^n [AP_i] \dots\dots\dots (3)$$

3. RESULTADOS

Tras hacer varias pruebas al algoritmo, se recolectaron 95 muestras, las cuales se desarrollaron en una matriz de confusión por cada residuo sólido, es decir, se realizó una tabla donde se puede evaluar el rendimiento y funcionamiento del sistema de clasificación desarrollado a través de la codificación en Python. Como se aprecia en la (Figura 3).



Figura 3. Detección de las 4 categorías

La matriz de confusión permite ver qué tipos de aciertos y errores está teniendo el modelo a la hora de pasar por el proceso de aprendizaje y clasificación, es así como, los tipos de aciertos o errores que puede tener el modelo realizados son:

Verdadero Positivo (TP): Este viene a ser el resultado en el que el modelo detecta el residuo y detecta correctamente el tipo de residuo. Verdadero Negativo (TN): Este viene a ser el resultado en el que el modelo indica de manera correcta que un residuo no pertenece a su determinado tipo de residuo. Falso Positivo (FP): Este viene a ser el resultado en el que el modelo detecta incorrectamente el tipo de residuo.

Falso Negativo (FN): Este tipo de error, indica el resultado en el que el modelo no detecta ningún tipo de residuo cuando en realidad si hay un tipo de residuo.

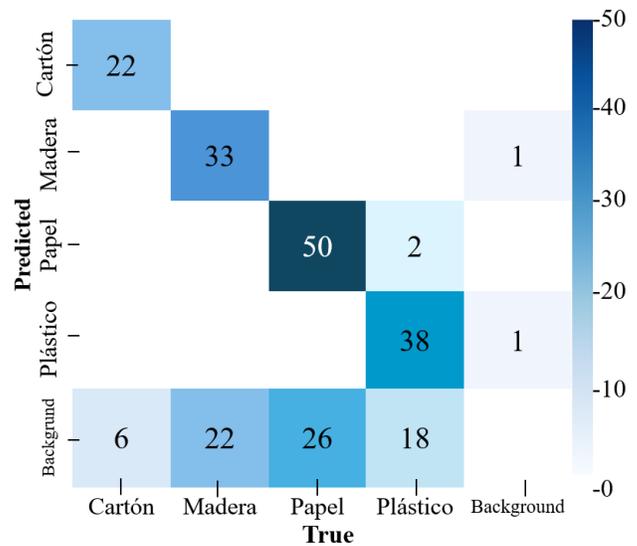


Figura 4. Matriz de confusión

En la (Figura 4), se puede analizar que de los resultados arrojados se distribuyeron de manera que hubo una cantidad de 38, 33, 50 y 22 muestras verdaderamente positivas referente a la detección de plástico, madera, papel y cartón respectivamente. En cuanto a los falsos negativos hubo una cantidad de 6, 1, 2 y 4 muestras referentes a la detección de plástico, madera, papel y cartón respectivamente. Por último, hubo una cantidad de 18, 22, 26 y 6 muestras falsos positivos referentes a la detección de plástico, madera, papel y cartón respectivamente.

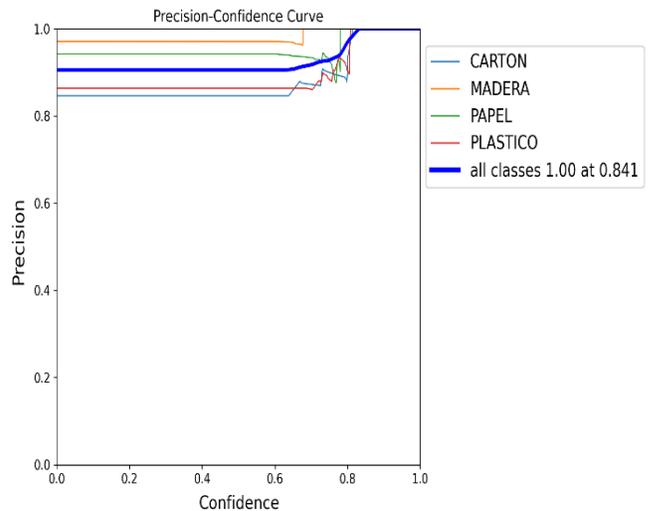


Figura 5. Curva Precision-Confidence

Después de los resultados de las pruebas, el programa nos muestra las siguientes curvas. En la (Figura 5), se presenta la curva Precision-Confidence que se relaciona con la proporción de verdaderos positivos entre el total de positivos. Cada línea de diferente color representa cada una de las 4 categorías de residuos sólidos, la línea de curva roja que representa el plástico está un poco más baja a comparación de las demás, en especial a niveles de confianzas menos. La línea azul indica que todas las clases alcanzan una precisión de 1.00 a un nivel de confianza de 0.841.

Eso sugiere que el modelo más exacto debido a que medida que la confianza aumente, aumentará la precisión al captar más verdaderos positivos que falsos positivos.

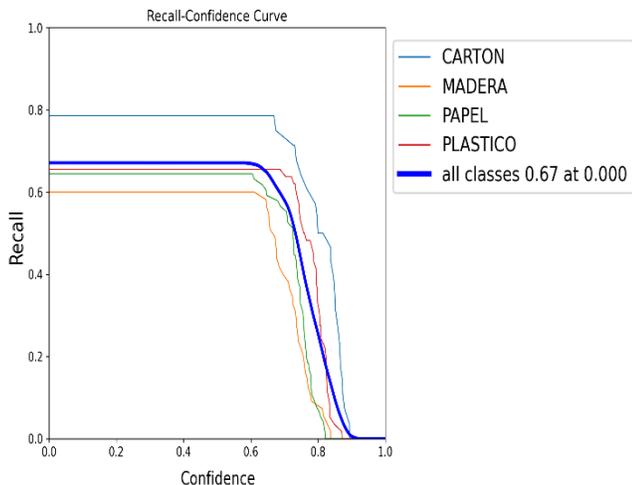


Figura 6. Curva Recall-Confidence

En la (Figura 6), se puede observar la curva Recall – Confidence, donde se representa la relación del nivel de confianza con los verdaderos positivos. Cada curva corresponde a una clase específica de residuos (cartón, madera, papel y plástico), a medida que aumenta el nivel de confianza, el recall disminuye; además, cada clase está asignada por un color diferente. La línea azul representa el rendimiento general de todas las clases, indicando que el recall es de 0.67 cuando la confianza es 0.0. En las líneas bajas de confianza, el recall es más alto debido a que se hace muchas predicciones positivas que a su vez incluye verdaderas y falsas. Pero si el nivel de confianza aumenta el modelo se vuelve más conservador haciendo menos predicciones positivas. En esta figura, se puede observar que la curva de cartón presenta un mejor rendimiento, manteniendo un recall alto a través de líneas de confianza altas.

Adicionalmente, para tener plena confianza en que el modelo del algoritmo de clasificación es fiable y confiable se hizo uso de la métrica de Precisión/Precision (P) y Sensibilidad/Recall (R) como se muestra en la (Tabla 5).

Tabla 5. Resultados de Precision y Recall

Class	Images	Instances	Box (P)	R	mAP50	mAP50-95
ALL	95	217	0.906	0.671	0.798	0.52
CARTON	19	28	0.846	0.786	0.848	0.605
MADERA	23	55	0.971	0.6	0.789	0.5
PAPEL	49	76	0.942	0.645	0.782	0.463
PLASTICO	41	58	0.864	0.655	0.772	0.513

De la tabla 5, se muestra que hay 95 imágenes y 217 instancias en total. La madera tiene el porcentaje más alto de precisión (P) el 97.1% que calcula cual exacto son las predicciones en el modelo y el cartón tiene el valor más alto con el 78.6% de sensibilidad (R) que mide la proporción de instancias positivas reales que el modelo identificó correctamente [26].

4. DISCUSIÓN

Este proyecto propone el desarrollo de un algoritmo que emplea redes neuronales convolucionales, utilizando específicamente el modelo YOLOv8, para clasificar distintos tipos de residuos sólidos en áreas universitarias. La finalidad es mejorar la clasificación de desechos frecuentes como el cartón, la madera, el papel y el plástico, lo cual facilitará una gestión de residuos más eficaz en los entornos educativos superiores.

Para la ejecución de este proyecto, se compiló y procesó una base de datos que incluye 1203 imágenes de variados residuos. Se entrenó el modelo YOLOv8 con estas imágenes, capacitando al algoritmo para reconocer y categorizar eficazmente los diferentes materiales. La precisión del modelo, que evalúa la capacidad para clasificar acertadamente cada tipo de material, obtuvo los siguientes valores: 84.6% en cartón, 97.1% en madera, 94.2% en papel y 86.4% en plástico; logrando una precisión global de 90.6%.

Al realizar un análisis comparativo con estudios anteriores, se destaca que la investigación [1] abordó la clasificación de residuos sólidos en La Universidad Sudafricana de Tecnología (SAUoT). Dicho estudio empleó un total de 3171 imágenes para el entrenamiento y aplicó el modelo YOLOv3, obteniendo una precisión del 80%. En nuestro estudio, a pesar de haber utilizado un volumen menor de imágenes, específicamente 1203 para el entrenamiento del algoritmo, se obtuvo una precisión global del 90.6%, superando el rendimiento del estudio anterior. Esta variación se puede atribuir al uso de un modelo más actual de YOLO, en su versión YOLOv8, que presenta una arquitectura de red mejorada con bloques más eficientes y capas adicionales, lo que permite una mayor capacidad de aprendizaje y una mejor representación de características, caso contrario a lo sucedido con YOLOv3 que usa una arquitectura más antigua basada en Darknet-53.

Por otra parte, el trabajo [2] se enfocó en la clasificación de residuos sólidos dentro del contexto universitario, donde se definieron diversas categorías como plástico, vidrio, metal, papel, cartón, orgánico y "no reciclable". Para llevar a cabo el estudio, se optó por utilizar el modelo de YOLOv5 y se entrenó con un total de 800 imágenes específicamente etiquetadas en el programa de etiquetado Makesense.ai. El entrenamiento del modelo se extendió a lo largo de 60 épocas, un proceso crucial para optimizar los resultados finales. A pesar de estos esfuerzos, el estudio reportó una precisión global del 54%. La investigación atribuye este rendimiento inferior a las condiciones no controladas del entorno donde se obtuvieron las fotografías, puesto que fueron tomadas directamente en las instalaciones universitarias, lo cual generó dificultades en la identificación precisa de los residuos. El presente trabajo de investigación también se desarrolló en un entorno no controlado obteniendo métricas mayores debido a un mejor entrenamiento, una mayor base de datos almacenada en el programa de etiquetado Roboflow y un total de 100 épocas que nos dio una precisión del 90.6%.

En suma, la implementación exitosa del modelo a YOLOv8 ha permitido clasificar con alta precisión diversos materiales debido a que representa una tecnología más avanzada lo que permitió mayor precisión en los resultados arrojados.

5. CONCLUSIÓN

En el presente estudio se logró desarrollar un algoritmo mediante redes neuronales convolucionales para la clasificación de residuos sólidos en universidades. Ya que al aplicar eficazmente un algoritmo de red neuronal convolucional utilizando YOLOv8, en Python, se obtuvo una impresionante precisión global del 90,6% al identificar con exactitud cuatro tipos distintos de residuos: cartón, madera, papel y plástico. Por lo tanto, la ejecución de este algoritmo no solo mejora la gestión de la basura en las universidades, sino que sienta un precedente para futuros estudios en el ámbito de la inteligencia artificial aplicada a la sostenibilidad medioambiental.

Se consiguió recopilar 1203 imágenes representativas de los diferentes tipos de residuos sólidos, (plástico, papel, cartón y madera) que terminaron formando la data set, la cual se usó para realizar el proceso de entrenamiento del modelo YOLOv8 para detectar y clasificar con precisión la basura; todas las imágenes fueron tomadas en distintos ambientes de la universidad, ya que eran enfocadas a un ambiente no controlado.

Así mismo, se consiguió realizar una correcta codificación y posterior entrenamiento de la red neuronal YOLOv8, para lograr detectar e identificar el tipo de residuo presente en la imagen. En esta investigación, se llevó a cabo la codificación y entrenamiento exitoso de la red neuronal YOLOv8 para la detección e identificación de tipos de residuos en imágenes. Los resultados destacaron las ventajas de emplear modelos avanzados como YOLOv8, evidenciando mejoras en precisión y capacidad de detección según la literatura y documentación técnica. Sin embargo, no se realizaron pruebas comparativas con versiones anteriores en este estudio, por lo que se sugiere la realización de estudios futuros que evalúen su desempeño frente a versiones anteriores para confirmar estas mejoras de manera más concluyente.

Por último, los resultados obtenidos de la precisión y eficacia alcanzadas por el modelo desarrollado ponen de relieve la importancia de seguir explorando y perfeccionando estas herramientas tecnológicas para hacer frente a los retos medioambientales contemporáneos, es así, que este proyecto no sólo valida la viabilidad del uso de redes neuronales convolucionales para la clasificación de residuos sólidos en contextos universitarios, sino que también proporciona una base sólida para la integración de tecnologías avanzadas en la gestión medioambiental. Y así en un futuro, el detector de objetos YOLO personalizado podrá desplegarse en el prototipo de diseño propuesto para el contenedor de basura y la recolección de objetos de desecho.

6. REFERENCIAS

- [1] I. E. Agbehadji, A. Abayomi, K.-H. N. Bui, R. C. Millham y E. Freeman, “Nature-Inspired Search Method and Custom Waste Object Detection and Classification Model for Smart Waste Bin”, *Sensors*, vol. 22, n.º 16, p. 6176, agosto de 2022. Disponible: <https://doi.org/10.3390/s22166176>
- [2] J. Puentes y S. Jiménez, “PROPUESTA DE IMPLEMENTACIÓN DE CLASIFICACIÓN DE RESIDUOS SÓLIDOS POR MEDIO DE INTELIGENCIA ARTIFICIAL EN SITIOS DE DISPOSICIÓN EN LA UNIVERSIDAD EAN”, UNIV. EAN, Bogotá, 2023. Disponible: <http://hdl.handle.net/10882/13267>
- [3] V. A. Bondarenko y D. I. Popov, “Research and development of algorithms for the formation of an effective ensemble of convolutional neural networks for image classification”, *Программные системы и вычислительные методы*, n.º 1, pp. 48–67, enero de 2024. Disponible: <https://doi.org/10.7256/2454-0714.2024.1.69919>
- [4] S. Wen, Y. Yuan y J. Chen, “A Vision Detection Scheme Based on Deep Learning in a Waste Plastics Sorting System”, *Appl. Sci.*, vol. 13, n.º 7, p. 4634, abril de 2023. Disponible: <https://doi.org/10.3390/app13074634>
- [5] Mao, W.-L., Chen, W.-C., Wang, C.-T., Lin, Y.-H. Recycling waste classification using optimized convolutional neural network. (2021) *Resources, Conservation and Recycling*, 164, art. no. 105132. Disponible: <https://doi.org/10.1016/j.resconrec.2020.105132>
- [6] J. Máquez et al, “Prototipo de sistema automatizado con visión artificial para la selección de empaques de plástico, vidrio y lata en el proceso de reciclaje”, Título de Ingeniero de Sistemas, UNIV. CUNDINAMARCA, Cundinamarca, 2019. Disponible: <http://hdl.handle.net/20.500.12558/2497>
- [7] M. Chaparro y H. Vargas, “PROTOTIPO DE SISTEMA DE CLASIFICACIÓN DE MATERIALES RECICLABLES MEDIANTE EL RECONOCIMIENTO DE IMÁGENES, HACIENDO USO DE HARDWARE PARA LA AUTOMATIZACIÓN DE LA RECOLECCIÓN”, Título de Ingeniero de Sistemas, UNIV. ECCI, Bogotá, 2022. Accedido el 29 de abril de 2024. <https://repositorio.ecci.edu.co/bitstream/handle/001/3344/Trabajo%20de%20grado.pdf?sequence=1&isAllowed=y>
- [8] M. Martin, “Desarrollo de un sistema inteligente para la clasificación de residuos sólidos”, Grado de Ingeniería Telemática, Univ. Politec. Madr., Madrid, 2022. Disponible: https://oa.upm.es/72295/1/TFG_MARINA_MARTIN_MORENO.pdf
- [9] Z. Moreno, “Desarrollo de un Algoritmo Computacional basado en Deep Learning para la Clasificación de Residuos Orgánicos”, Título de Ingeniero de Sistemas, Univ. Nac. Abierta Distancia UNAD, Bogotá, 2023. Disponible: <https://repositorio.unad.edu.co/bitstream/handle/10596/60395/zvmorenog.pdf?sequence=1>
- [10] L. Cabellos et al, “Diseño de un algoritmo de clasificación de imágenes usando redes neuronales para la correcta clasificación de residuos en la I.E. 15122 El Puente Quiroz”, Ingeniería Industrial y de Sistemas, Univ. Piura, Piura, 2022. Disponible: <https://pirhua.udep.edu.pe/backend/api/core/bitstreams/8f21144c-5173-477e-bb83-e5a6785f7258/content>
- [11] J. Beltran y J. Quinatana, “Diseño y desarrollo de un sistema automatizado enfocado en el reconocimiento y clasificación de residuos sólidos para una mejor gestión”, Título Profesional de Ingeniero Mecatrónico, Univ. Cont., Huancayo, 2023. Disponible: https://repositorio.continental.edu.pe/bitstream/20.500.12394/12572/2/IV_FIN_112_TE_Beltran_Quinatana_2023.pdf
- [12] H. Abu-Qdais, N. Shatnawi y E. AL-Alamie, “Intelligent system for solid waste classification using a combination of image processing and machine learning models”, *J. Exp. & Theor. Artif. Intell.*, pp. 1–12, febrero de 2024. Accedido el 26 de abril de 2024. [En línea]. Disponible: <https://doi.org/10.1080/0952813x.2024.2323043>
- [13] L. M. Espinoza Tinoco, A. E. Congacha Aushay y J. C. Díaz Ordóñez, “Calidad de datos con Python: Un enfoque práctico”, *Esprint Investig.*, vol. 2, n.º 2, pp. 26–34, septiembre de 2023. Accedido el 27 de abril de 2024. [En línea]. Disponible: <https://doi.org/10.61347/ei.v2i2.55>

- [14] O. I. Funch, R. Marhaug, S. Kohtala y M. Steinert, “Detecting glass and metal in consumer trash bags during waste collection using convolutional neural networks”, *Waste Manage.*, vol. 119, pp. 30–38, Enero de 2021. Accedido el 26 de abril de 2024. [En línea]. Disponible: <https://doi.org/10.1016/j.wasman.2020.09.032>
- [15] A. Paszke. et al, “PyTorch: An Imperative Style, High-Performance Deep Learning Library”, *Advances neural inf. process. syst.*, vol. 32, p. 12, 2019. Accedido el 26 de abril de 2024. [En línea]. Disponible: <https://doi.org/10.48550/arXiv.1912.01703>
- [16] D. Ziouzos, N. Baras, V. Balafas, M. Dasygenis y A. Stimoniaris, “Intelligent and Real-Time Detection and Classification Algorithm for Recycled Materials Using Convolutional Neural Networks”, *Recycling*, vol. 7, n.º 1, p. 9, febrero de 2022. Disponible: <https://doi.org/10.3390/recycling7010009>
- [17] M. Malik et al., “Waste Classification for Sustainable Development Using Image Recognition with Deep Learning Neural Network Models”, *Sustainability*, vol. 14, n.º 12, p. 7222, junio de 2022. Disponible: <https://doi.org/10.3390/su14127222>
- [18] T.-W. Wu, H. Zhang, W. Peng, F. Lü y P.-J. He, “Applications of convolutional neural networks for intelligent waste identification and recycling: A review”, *Resour., Conservation Recycling*, vol. 190, p. 106813, marzo de 2023. Accedido el 27 de mayo de 2024. [En línea]. Disponible: <https://doi.org/10.1016/j.resconrec.2022.106813>
- [19] S. Moreno, “Sistema de clasificación de residuos mediante redes convolucionales”, *Grado en Ingeniería de Sistemas de Telecomunicaciones*, Univ. Politec. Catalunya, Barcelona, 2023. Accedido el 27 de mayo de 2024. [En línea]. Disponible: <https://upcommons.upc.edu/bitstream/handle/2117/392008/memoria.pdf?sequence=2&isAllowed=y>
- [20] Z. Wang, Z. Hua, Y. Wen, S. Zhang, X. Xu y H. Song, “E-YOLO: Recognition of Estrus Cow Based on Improved YOLOv8n Model”, *Expert Syst. with Appl.*, p. 122212, octubre de 2023. Accedido el 23 de mayo de 2024. [En línea]. Disponible: <https://doi.org/10.1016/j.eswa.2023.122212>
- [21] M. Sportelli et al., “Evaluation of YOLO Object Detectors for Weed Detection in Different Turfgrass Scenarios”, *Appl. Sci.*, vol. 13, n.º 14, p. 8502, julio de 2023. Accedido el 23 de mayo de 2024. [En línea]. Disponible: <https://doi.org/10.3390/app13148502>
- [22] D. Reis, J. Kupec, J. Hong y A. Daoudi, “Real-Time Flying Object Detection with YOLOv8”, *Preimpresion arXiv*, p. 10, 2023. Accedido el 22 de mayo de 2024. [En línea]. Disponible: <https://doi.org/10.48550/arXiv.2305.09972>
- [23] G. R. Matuck, A. J. Abreu Castro, L. E. Da Silva y E. Gomes Carvalho, “RECONHECIMENTO FACIAL COM INTELIGÊNCIA ARTIFICIAL UTILIZANDO A PLATAFORMA ROBOFLOW”, *Rev. Prociencias*, vol. 6, n.º 2, pp. 114–131, enero de 2024. Accedido el 27 de mayo de 2024. [En línea]. Disponible: <https://doi.org/10.15210/prociencias.v6i2.25948>
- [24] A. Gangal, P. Kumar y S. Kumari, “Complete Scanning Application Using OpenCv”, *arXiv Prepr.*, p. 10, 2021. Accedido el 19 de mayo de 2024. [En línea]. Disponible: <https://doi.org/10.48550/arXiv.2107.03700>
- [25] M. Butt, N. Glas, J. Monsuur, R. Stoop y A. de Keijzer, “Application of YOLOv8 and Detectron2 for Bullet Hole Detection and Score Calculation from Shooting Cards”, *AI*, vol. 5, n.º 1, pp. 72–90, diciembre de 2023. Accedido el 27 de mayo de 2024. [En línea]. Disponible: <https://doi.org/10.3390/ai5010005>
- [26] D. Gomez et al., “Advancing common bean (*Phaseolus vulgaris* L.) disease detection with YOLO driven deep learning to enhance agricultural AI”, *Scientific Rep.*, vol. 14, n.º 1, julio de 2024. Accedido el 10 de agosto de 2024. [En línea]. Disponible: <https://doi.org/10.1038/s41598-024-66281-w>