



*Asesorías y Tutorías para la Investigación Científica en la Educación Puig-Salabarría S.C.
José María Pino Suárez 400-2 esq a Lerdo de Tejada, Toluca, Estado de México. 7223898475*

RFC: ATI120618V12

Revista Dilemas Contemporáneos: Educación, Política y Valores.

<http://www.dilemascontemporaneoseduccionpoliticayvalores.com/>

Año: XII

Número: 3

Artículo no.: 71

Período: 1 de mayo al 31 de agosto del 2025

TÍTULO: Dataset de clasificación de Residuos Sólidos Urbanos para Redes Neuronales Convolucionales.

AUTORES:

1. Dra. Mirna Castro Bello.
2. Lic. Víctor Manuel Romero Juárez.
3. Dr. Jorge Fuentes Pacheco.
4. Máster. Sergio Ricardo Zagal Barrera.
5. Máster. Areli Bárcenas Nava.
6. Máster. Diego Esteban Gutiérrez Valencia.

RESUMEN: Los modelos de Redes Neuronales Convolucionales (RNC) requieren datos procesados que aprendan patrones de imágenes para evitar la memorización. Esta investigación presenta la conformación de un *dataset* de 3208 imágenes de clasificación de Residuos Sólidos Urbanos (RSU) en orgánicos e inorgánicos para modelos de RNC. El procesamiento se realizó con una metodología de tres fases: 1. Identificación y selección de *dataset*: Kaggle y Github; 2. Creación del *dataset*: uniformidad de imágenes y ajustes de color; y 3. Conformación del *dataset* de residuos orgánicos e inorgánicos. Los resultados obtenidos fueron los conjuntos de datos de RSU orgánicos conformado por 1574 imágenes y 1634 inorgánicos. Esto permitirá el entrenamiento en modelos de Deep Learning para la clasificación binaria de RSU.

PALABRAS CLAVES: dataset, residuos sólidos urbanos, redes neuronales convolucionales, deep learning, clasificación.

TITLE: Urban Solid Waste Classification Dataset for Convolutional Neural Networks.

AUTHORS:

1. PhD. Mirna Castro Bello.
2. Bach. Víctor Manuel Romero Juárez.
3. PhD. Jorge Fuentes Pacheco.
4. Master. Sergio Ricardo Zagal Barrera.
5. Master. Areli Bárcenas Nava.
6. Master. Diego Esteban Gutiérrez Valencia.

ABSTRACT: Convolutional Neural Network (CNN) models require processed data that learns image patterns to avoid memorization. This research presents the creation of a dataset of 3,208 images for the classification of Urban Solid Waste (MSW) into organic and inorganic forms for CNN models. Processing was carried out using a three-phase methodology: 1. Dataset identification and selection: Kaggle and Github; 2. Dataset creation: image uniformity and color adjustments; and 3. Creation of the organic and inorganic waste dataset. The results obtained were the organic MSW datasets consisting of 1,574 images and 1,634 inorganic images. This will enable the training of Deep Learning models for binary MSW classification.

KEY WORDS: dataset, urban solid waste, convolutional neural networks, deep learning, classification.

INTRODUCCIÓN.

El crecimiento poblacional y la aceleración de la urbanización han provocado un fuerte aumento en la producción de Residuos Sólidos Urbanos (RSU); el Banco Mundial estima que se generan 2.010 millones de toneladas, de las cuales el 33% se gestionan de manera inadecuada, mientras que el promedio per cápita de generación es de 0.74 kg (Kaza et al., 2018); lo que complica su administración a las autoridades gubernamentales, dado que en la mayoría de los países la separación se realiza de forma manual, exponiendo a los pepenadores a enfermedades debido a las sustancias nocivas presentes en la basura (Namoun et al., 2022). A falta de estrategias para mejorar la protección y cuidado del medio ambiente,

diversas investigaciones han aplicado diferentes metodologías para clasificar RSU con procesamiento de imágenes y Redes Neuronales Convolucionales (RNC).

Por otra parte, las tareas de reciclaje generalmente requieren un gran costo de mano de obra, por lo que el reconocimiento y la detección automática de residuos con imágenes sustituye la clasificación manual, debido a los avances en el campo de la Inteligencia Artificial (IA) (Mao et al., 2021). Los algoritmos de Deep Learning (DL) han mejorado la precisión en la clasificación automática, son capaces de abordar problemas complejos en el reconocimiento y clasificación, siendo las Redes Neuronales Convolucionales formadas por una capa de convolución responsable de extracción de características y una capa completamente conectada que actúa como clasificador y contiene una gran cantidad de neuronas y una metodología predominante para la identificación de imágenes (Liu et al., 2019).

En la actualidad es importante disponer de un conjunto de datos previamente procesados con características específicas para su reconocimiento y detección automática en la Inteligencia Artificial (Nnamoko et al., 2022).

Es el caso de Srinilta y Kanharattanachai (2019) plantean el uso de 9200 imágenes de Residuos Sólidos Urbanos para explorar el rendimiento de los clasificadores basados en Redes Neuronales Convolucionales (VGG-16, ResNet-50, MobileNet V2 y DenseNet-121), obteniendo una precisión del 94.86% en el modelo ResNet-50. Adedeji y Wang (2019) proponen un sistema inteligente de clasificación de RSU en diferentes grupos como vidrio, metal, papel y plástico, entre otros, a través de una RNC de 50 capas preentrenada (ResNet-50), probado en la base de datos TrashNet de 2527 imágenes, con una exactitud de 87%. Nnamoko et al. (2022) utilizan un conjunto de datos que contiene 25,077 imágenes categorizadas en 13,966 residuos orgánicos y 11,111 reciclables para evaluar el rendimiento de una Redes Neuronales Convolucionales de cinco capas con dos resoluciones de imágenes diferentes.

Single et al. (2023) crearon un conjunto de datos de residuos en vertederos con un total de 4,808 muestras de una resolución de 524 x 524 para la evaluación de cinco modelos de DL (VGG-16, InceptionResNetV2,

DenseNet121, InceptionV3 y MobileNetV2), logrando 89.19% de precisión. Liang et al. (2021) entrenaron modelos de Redes Neuronales Artificiales mediante el algoritmo de optimización de Arquímedes, algoritmo seno-coseno, la técnica de optimización por enjambre de partículas y los algoritmos genéticos utilizando conjuntos de datos provenientes de siete megaciudades iraníes.

Ruiz et al. (2019) utilizaron datos de TrashNet para entrenar y comparar las arquitecturas de DL (VGG, Inception y ResNet) logrando 88.6% de exactitud con el modelo combinado Inception-ResNet. Zhang et al (2021) construyeron un conjunto de datos de imágenes de residuos denominado NWN-TRASH para la evaluación de un modelo de clasificación de RSU usando DenseNet169 mediante aprendizaje por transferencia, en el que alcanzaron más del 82% de precisión.

Esta investigación presenta la creación de un *dataset* de 3208 imágenes de clasificación de Residuos Sólidos Urbanos para modelos de Redes Neuronales Convolucionales en categorías de orgánico e inorgánico. El procesamiento se realizó a través de una metodología de tres fases:

1. Identificación y selección de *dataset*: Kaggle y Github, dado que son de acceso abierto.
2. Creación del *dataset*: Uniformidad de imágenes con la finalidad de evitar la memorización en los entrenamientos de los modelos DL, y ajustes de color para capturar mejor los patrones en las imágenes de RSU.
3. Conformación del *dataset*.

Como resultados se obtuvieron el conjunto de datos de residuos orgánicos conformado por 1574 imágenes y 1634 inorgánicos para el entrenamiento en modelos de DL con clasificación binaria de RSU.

DESARROLLO.

Materiales.

Para la elaboración del *dataset* se utilizaron seis repositorios públicos; lenguaje de programación Python versión 3.10.14 y la biblioteca Pillow de acceso abierto.

Metodología.

La metodología aplicada constó de tres fases, figura 1.

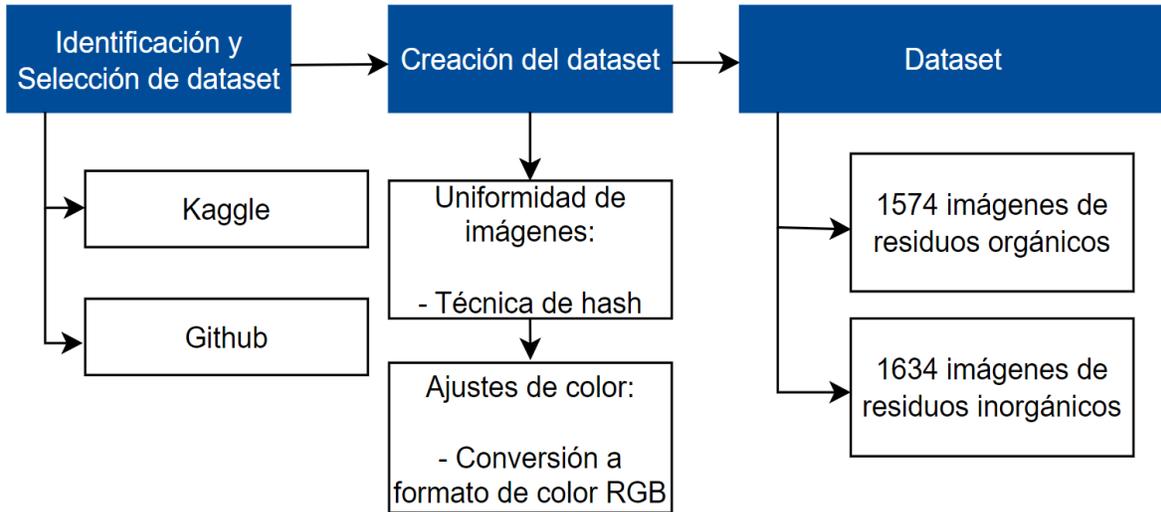


Figura 1. Diagrama metodológico del *dataset*.

Creación del conjunto de imágenes.

Para configurar el conjunto de imágenes, se consultaron distintas bases de datos como TrashNet que considera residuos de plástico, metal, vidrio, papel y cartón (Thung, 2017); Garbage_Huawei que contiene residuos orgánicos, peligrosos e inorgánicos (Xiaohoua, 2023); Waste Classification data con residuos orgánicos e inorgánicos (Sekar, 2018); Garbage Dataset contiene imágenes de metal, vidrio, biológico, papel, baterías, basura, cartón, zapatos, ropa y plástico (Kunwar, 2024); Garbage Classification con ilustraciones de batería, cartón, biológico, ropa, vidrio verde, metal, papel, plástico, zapatos, basura y vidrio blanco (Mohamed, 2020); y Garbage Pictures for Classification, que cuenta con las categorías de cartón, vidrio, metal, papel, plástico y basura (tabla 1).

Tabla 1. Dataset públicos consultados.

Dataset	Enlace	Cantidad
TrashNet.	https://github.com/garythung/trashnet	993
Garbage Huawei.	https://www.kaggle.com/datasets/xiaohoua/garbage-huawei	548

Waste Classification data.	https://www.kaggle.com/datasets/techsash/waste-classification-data	731
Garbage Dataset.	https://www.kaggle.com/datasets/sumn2u/garbage-classification-v2	175
Garbage Classification.	https://www.kaggle.com/datasets/mostafaabla/garbage-classification	275
Garbage Pictures for Classification.	https://www.kaggle.com/datasets/mascot9183/garbage-pictures-for-classification?select=Garbage+classification	339

Se consideraron solo dos repositorios públicos, Kaggle por su variedad y calidad de imágenes en su dataset, y Github debido a que en diferentes estudios aplicados a modelos de DL utilizaron TrashNet por la practicidad en la aplicación a un mayor número de datos.

Para la construcción del conjunto de datos utilizado en este estudio, se definieron criterios específicos que permitieron seleccionar y combinar diversos *dataset* existentes, asegurando la calidad y relevancia del *dataset* final. Los criterios considerados fueron los siguientes:

Diversidad visual.

Se eligieron los conjuntos de datos que ofrecían una alta variabilidad visual, incluyendo diferentes condiciones de iluminación, perspectivas, fondos y resoluciones, con el propósito de mejorar la capacidad de generalización de los modelos.

Pertinencia temática.

Se incluyeron únicamente aquellos cuyo contenido correspondía directamente con el objetivo principal del estudio, garantizando así la relevancia y aplicabilidad de cada imagen seleccionada.

Origen verificable y calidad.

Se priorizaron aquellos provenientes de fuentes confiables y reconocidas, tales como repositorios académicos, bases de datos oficiales, con la finalidad de asegurar la calidad y autenticidad del material visual.

Licencias y permisos de uso.

Se verificó cuidadosamente que todos los seleccionados contaran con licencias abiertas o permisos explícitos de uso que permitiesen su reutilización, modificación y distribución en este proyecto.

Eliminación de duplicados.

Se aplicaron técnicas avanzadas de hashing de detección visual de duplicados, para asegurar que no se incluyeran imágenes repetidas o altamente similares en el conjunto final.

Balance de clases.

Se consideró la inclusión de *dataset* que ofrecieran una distribución equilibrada de imágenes entre las diferentes categorías o clases, minimizando sesgos y mejorando el desempeño predictivo del modelo.

Calidad y resolución de imágenes.

Se optó por *dataset* compuestos por imágenes de resolución y calidad adecuadas, garantizando que las características extraídas fueran óptimas para la tarea propuesta.

Tamaño del dataset.

Se escogieron los que cuentan con un volumen significativo de imágenes, para que tras la aplicación del proceso de filtrado, el conjunto final resultara suficientemente robusto para el entrenamiento efectivo de los modelos.

Estos criterios permitieron obtener un *dataset* final coherente, relevante y libre de redundancias visuales, optimizando así los resultados esperados en los modelos de DL.

Uniformidad de imágenes.

Una característica importante en un conjunto de datos es la uniformidad, que consiste en obtener imágenes auténticas para que los modelos identifiquen las características durante su fase de entrenamiento, y validen a través de la detección de características y conformación de patrones para una correcta clasificación.

Para garantizar la uniformidad en las imágenes obtenidas del *dataset*, cada imagen fue inspeccionada cuidadosamente para identificar y eliminar duplicados, asegurando que no existieran repeticiones ni redundancias entre las distintas fuentes utilizadas. Se implementaron técnicas avanzadas de hash basadas en el contenido, permitiendo detectar similitudes visuales, incluso ante pequeñas variaciones o modificaciones.

Adicionalmente, se realizó un proceso automatizado de etiquetado con un algoritmo especializado que facilitó la clasificación precisa y consistente del conjunto completo de imágenes. Este enfoque combinado permitió aumentar significativamente la calidad y fiabilidad del *dataset* resultante.

Ajuste de color.

Otro aspecto importante en el *dataset* es el ajuste de color, debido a que en los repositorios públicos se encuentran imágenes con distintos formatos; por ejemplo, las imágenes indexadas; para ello, las ilustraciones fueron convertidas al formato de color RGB con la finalidad de asegurar la uniformidad y coherencia en el procesamiento, independientemente del formato de color original en el que fueron adquiridas (escala de grises, CMYK o imágenes indexadas).

El formato RGB se eligió dado a que las arquitecturas modernas de Redes Neuronales Convolucionales están optimizadas para procesar imágenes que constan exactamente de tres canales correspondientes a los colores primarios rojo, verde y azul. Esto permite aprovechar eficazmente los pesos preentrenados en grandes bases de datos, como ImageNet, que comúnmente utilizan imágenes RGB para transferir conocimiento visual previamente adquirido.

Al emplear el estándar RGB, se facilita la extracción y representación consistente de características visuales: colores, texturas y patrones complejos; en consecuencia, la adopción del formato RGB favorece

la estabilidad del entrenamiento, mejora el rendimiento general del modelo, y simplifica el flujo de trabajo para la implementación de técnicas avanzadas como data augmentation, regularización y aprendizaje por transferencia (Bricman y Ionescu, 2018; Ejaz et al., 2025) (figura 2).

```
import os
from PIL import Image  Import "PIL" could not be resolved
ruta_origen = "DATASET_RENOMBRADO/carton"
ruta_destino = "dataset_rgb/carton"
for archivo in os.listdir(ruta_origen):
    # Obtener la ruta completa del archivo
    ruta_imagen = os.path.join(ruta_origen, archivo)
    # Abrir la imagen en modo lectura
    imagen = Image.open(ruta_imagen)
    # Convertir la imagen a RGB
    imagen_rgb = imagen.convert("RGB")
    # Obtener el nombre de la imagen sin la extensión
    nombre_imagen, extension = os.path.splitext(archivo)
    # Generar la ruta de destino para la imagen convertida
    ruta_imagen_destino = os.path.join(ruta_destino, f"{nombre_imagen}{extension}")
    # Guardar la imagen convertida en la ruta de destino
    imagen_rgb.save(ruta_imagen_destino)
```

(a)

```
import os
from PIL import Image  Import "PIL" could not be resolved
ruta_origen = "DATASET_RENOMBRADO/metal"
ruta_destino = "dataset_rgb/metal"
for archivo in os.listdir(ruta_origen):
    # Obtener la ruta completa del archivo
    ruta_imagen = os.path.join(ruta_origen, archivo)
    # Abrir la imagen en modo lectura
    imagen = Image.open(ruta_imagen)
    # Convertir la imagen a RGB
    imagen_rgb = imagen.convert("RGB")
    # Obtener el nombre de la imagen sin la extensión
    nombre_imagen, extension = os.path.splitext(archivo)
    # Generar la ruta de destino para la imagen convertida
    ruta_imagen_destino = os.path.join(ruta_destino, f"{nombre_imagen}{extension}")
    # Guardar la imagen convertida en la ruta de destino
    imagen_rgb.save(ruta_imagen_destino)
```

(b)

```
import os
from PIL import Image  Import "PIL" could not be resolved
ruta_origen = "DATASET_RENOMBRADO/organico"
ruta_destino = "dataset_rgb/organico"
for archivo in os.listdir(ruta_origen):
    # Obtener la ruta completa del archivo
    ruta_imagen = os.path.join(ruta_origen, archivo)
    # Abrir la imagen en modo lectura
    imagen = Image.open(ruta_imagen)
    # Convertir la imagen a RGB
    imagen_rgb = imagen.convert("RGB")
    # Obtener el nombre de la imagen sin la extensión
    nombre_imagen, extension = os.path.splitext(archivo)
    # Generar la ruta de destino para la imagen convertida
    ruta_imagen_destino = os.path.join(ruta_destino, f"{nombre_imagen}{extension}")
    # Guardar la imagen convertida en la ruta de destino
    imagen_rgb.save(ruta_imagen_destino)
```

(c)

```
import os
from PIL import Image  Import "PIL" could not be resolved
ruta_origen = "DATASET_RENOMBRADO/papel"
ruta_destino = "dataset_rgb/papel"
for archivo in os.listdir(ruta_origen):
    # Obtener la ruta completa del archivo
    ruta_imagen = os.path.join(ruta_origen, archivo)
    # Abrir la imagen en modo lectura
    imagen = Image.open(ruta_imagen)
    # Convertir la imagen a RGB
    imagen_rgb = imagen.convert("RGB")
    # Obtener el nombre de la imagen sin la extensión
    nombre_imagen, extension = os.path.splitext(archivo)
    # Generar la ruta de destino para la imagen convertida
    ruta_imagen_destino = os.path.join(ruta_destino, f"{nombre_imagen}{extension}")
    # Guardar la imagen convertida en la ruta de destino
    imagen_rgb.save(ruta_imagen_destino)
```

(d)

<pre>import os from PIL import Image Import "PIL" could not be resolved ruta_origen = "DATASET_RENOMBRADO/plastico" ruta_destino = "dataset_rgb/plastico" for archivo in os.listdir(ruta_origen): # Obtener la ruta completa del archivo ruta_imagen = os.path.join(ruta_origen, archivo) # Abrir la imagen en modo lectura imagen = Image.open(ruta_imagen) # Convertir la imagen a RGB imagen_rgb = imagen.convert("RGB") # Obtener el nombre de la imagen sin la extensión nombre_imagen, extension = os.path.splitext(archivo) # Generar la ruta de destino para la imagen convertida ruta_imagen_destino = os.path.join(ruta_destino, f"{nombre_imagen}{extension}") # Guardar la imagen convertida en la ruta de destino imagen_rgb.save(ruta_imagen_destino)</pre>	<pre>import os from PIL import Image Import "PIL" could not be resolved ruta_origen = "DATASET_RENOMBRADO/vidrio" ruta_destino = "dataset_rgb/vidrio" for archivo in os.listdir(ruta_origen): # Obtener la ruta completa del archivo ruta_imagen = os.path.join(ruta_origen, archivo) # Abrir la imagen en modo lectura imagen = Image.open(ruta_imagen) # Convertir la imagen a RGB imagen_rgb = imagen.convert("RGB") # Obtener el nombre de la imagen sin la extensión nombre_imagen, extension = os.path.splitext(archivo) # Generar la ruta de destino para la imagen convertida ruta_imagen_destino = os.path.join(ruta_destino, f"{nombre_imagen}{extension}") # Guardar la imagen convertida en la ruta de destino imagen_rgb.save(ruta_imagen_destino)</pre>
(e)	(f)

Figura 2. Algoritmos de conversión de color para (a) cartón, (b) metal, (c) orgánico, (d) papel, (e) plástico y (f) vidrio.

Resultados.

El *dataset* está conformado por un total de 3208 imágenes cuidadosamente seleccionadas y clasificadas en dos categorías principales: 1574 imágenes pertenecen a la clase de residuos orgánicos (figura 3) y 1634 a inorgánicos (figura 4), logrando una proporción equilibrada entre ambas clases.

Ese balance es fundamental para asegurar que los modelos de DL puedan generalizar adecuadamente, evitando sesgos hacia alguna categoría específica, y permitiendo que el entrenamiento sea representativo de escenarios reales; además, la distribución equilibrada facilita una evaluación más precisa del desempeño del modelo, al proporcionar condiciones ideales para que se reconozcan y diferencien características específicas relacionadas con cada tipo de residuo. Esta estrategia contribuye significativamente a mejorar tanto la estabilidad del proceso de entrenamiento como la confiabilidad de los resultados obtenidos en la etapa de validación de los modelos.

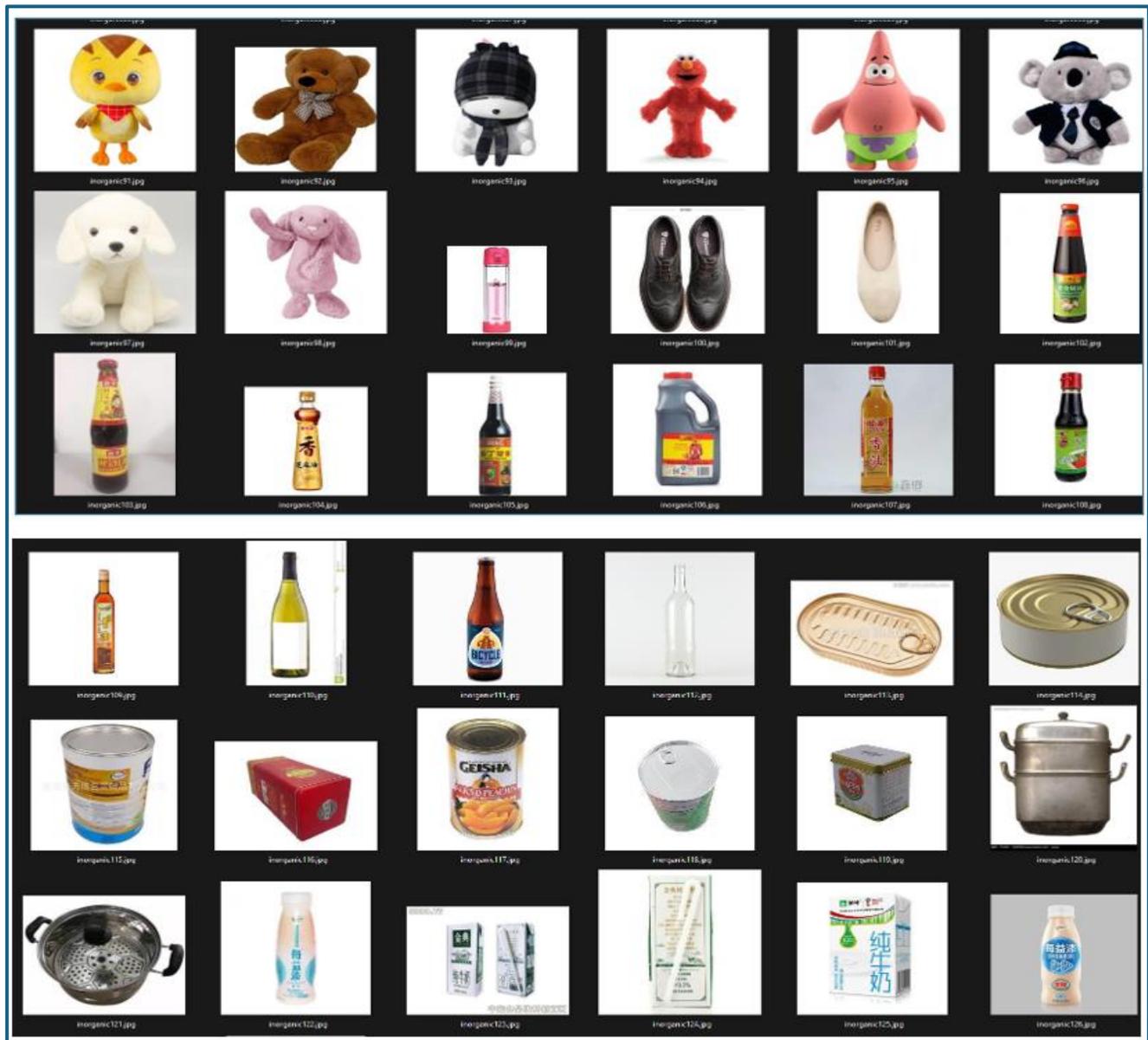


Figura 3. *Dataset* categoría inorgánicos.

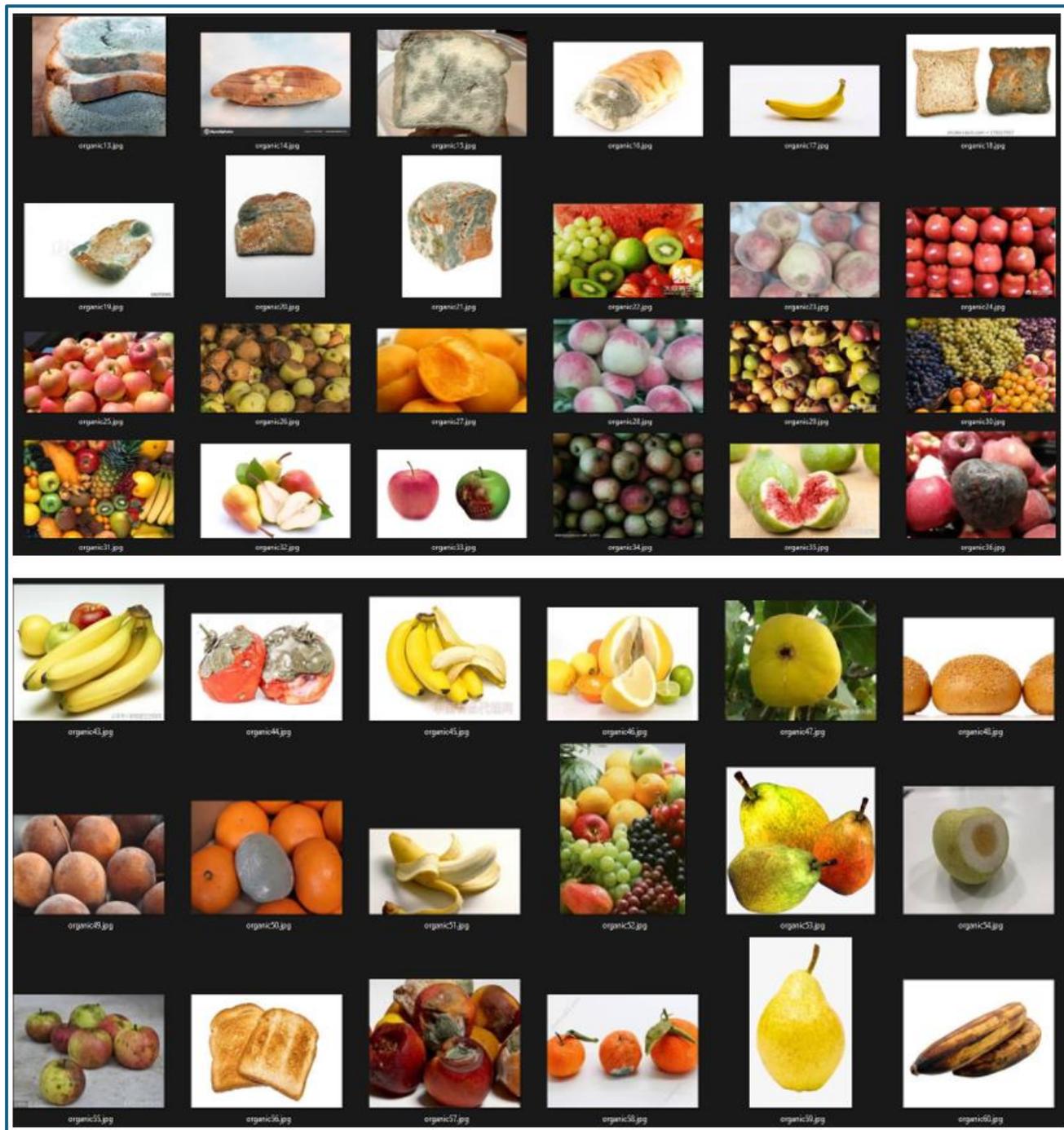


Figura 4. *Dataset* categoría orgánico.

Discusión.

La creación del *dataset* propuesto representa una mejora significativa respecto a conjuntos de datos anteriores, como TrashNet, utilizado en estudios de Adedeji y Wang (2019); Ruiz et al. (2019), quienes muestran una limitación destacable del conjunto de datos en su tamaño relativamente pequeño (2528

imágenes), una diversidad reducida en escenarios, fondos y repetición potencial en imágenes similares. En contraste, el *dataset* propuesto, construido con aplicación de técnicas de hash y conversión de color al formato RGB, garantiza una mayor diversidad y una reducción significativa en la repetición de las imágenes.

La técnica de hashing permitió identificar y eliminar imágenes duplicadas o similares, asegurando que cada imagen del conjunto fuera única y aportara valor específico para los procesos de entrenamientos en modelos de RNC; asimismo, Nnamoko et al. (2022) aunque utilizan un conjunto de datos considerablemente más amplio (25,077 imágenes), también enfrenta la problemática de repetición de imágenes similares o duplicadas, lo cual puede afectar negativamente la precisión y generalización de los modelos de DL.

CONCLUSIONES.

La elaboración de un *dataset* adecuado para la clasificación binaria de Residuos Sólidos Urbanos mediante las Redes Neuronales Convolucionales representa un paso clave en el desarrollo de modelos precisos y robustos. Una selección cuidadosa, inspección exhaustiva, preprocesamiento uniforme (por ejemplo, conversión al formato RGB) y etiquetado automatizado permiten generar un conjunto equilibrado y representativo que facilita la extracción eficiente de características visuales relevantes por parte del modelo, además de aplicar técnicas como la detección de imágenes duplicadas mediante hashing perceptual que garantizan la calidad e integridad de los datos.

Es conclusivo, que un *dataset* construido, bajo estas consideraciones, asegura no solo una mejor generalización del modelo, sino también resultados más confiables, escalables y aplicables en situaciones prácticas para la gestión y tratamiento inteligente de RSU.

Agradecimientos.

Los autores agradecen el apoyo a la Secretaría de Ciencia, Humanidades, Tecnología e Innovación (SECIHTI) y al Tecnológico Nacional de México, Instituto Tecnológico de Chilpancingo para la elaboración de este proyecto.

REFERENCIAS BIBLIOGRÁFICAS.

1. Adedeji, O., y Wang, Z. (2019). Intelligent Waste Classification System Using Deep Learning Convolutional Neural Network. *Procedia Manufacturing*, 35, 607–612. <https://doi.org/10.1016/j.promfg.2019.05.086>
2. Bricman, P. A., y Ionescu, R. T. (2018). CocoNet: A deep neural network for mapping pixel coordinates to color values. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11302 LNCS, 64–76. https://doi.org/10.1007/978-3-030-04179-3_6
3. Ejaz, U., Hamza, M. A., y Kim, H. C. (2025). Channel Attention for Fire and Smoke Detection: Impact of Augmentation, Color Spaces, and Adversarial Attacks. *Sensors* 2025, Vol. 25, Page 1140, 25(4), 1140. <https://doi.org/10.3390/S25041140>
4. Kaza, S., Yao, L. C., Bhada-Tata, P., y Van Woerden, F. (2018). What a Waste 2.0: A Global Snapshot of Solid Waste Management to 2050. *What a Waste 2.0: A Global Snapshot of Solid Waste Management to 2050*. <https://doi.org/10.1596/978-1-4648-1329-0>
5. Kunwar, S. (2024). Garbage Dataset. Kaggle. <https://doi.org/10.34740/KAGGLE/DSV/10182596>
6. Liang, G., Panahi, F., Ahmed, A. N., Ehteram, M., Band, S. S., y Elshafie, A. (2021). Predicting municipal solid waste using a coupled artificial neural network with archimedes optimisation algorithm and socioeconomic components. *Journal of Cleaner Production*, 315, 128039. <https://doi.org/10.1016/j.jclepro.2021.128039>
7. Liu, X., Zhou, Q., Zhao, J., Shen, H., y Xiong, X. (2019). Fault Diagnosis of Rotating Machinery under Noisy Environment Conditions Based on a 1-D Convolutional Autoencoder and 1-D Convolutional Neural Network. *Sensors*, 19(4). <https://doi.org/10.3390/s19040972>

8. Mao, W. L., Chen, W. C., Wang, C. T., y Lin, Y. H. (2021). Recycling waste classification using optimized convolutional neural network. *Resources, Conservation and Recycling*, 164, 105132. <https://doi.org/10.1016/J.RESCONREC.2020.105132>
9. Mohamed, M. (2020). Garbage Classification (12 classes). <https://www.kaggle.com/datasets/mostafaabla/garbage-classification>
10. Namoun, A., Hussein, B. R., Tufail, A., Alrehaili, A., Syed, T. A., y BenRhouma, O. (2022). An Ensemble Learning Based Classification Approach for the Prediction of Household Solid Waste Generation. *Sensors*, 22(9). <https://doi.org/10.3390/s22093506>
11. Nnamoko, N., Barrowclough, J., y Procter, J. (2022). Solid Waste Image Classification Using Deep Convolutional Neural Network. *Infrastructures*, 7(4). <https://doi.org/10.3390/infrastructures7040047>
12. Ruiz, V., Sánchez, Á., Vélez, J. F., y Raducanu, B. (2019). Automatic Image-Based Waste Classification. In J. R. and de la P. L. F. and T. M. J. and A. H. Ferrández Vicente José Manuel and Álvarez-Sánchez (Ed.), *From Bioinspired Systems and Biomedical Applications to Machine Learning* (pp. 422–431). Springer International Publishing. https://doi.org/10.1007/978-3-030-19651-6_41
13. Sekar, S. (2018). Waste Classification data. <https://www.kaggle.com/datasets/techsash/waste-classification-data>
14. Single, S., Iranmanesh, S., y Raad, R. (2023). RealWaste: A Novel Real-Life Data Set for Landfill Waste Classification Using Deep Learning. *Information*, 14(12). <https://doi.org/10.3390/info14120633>
15. Srinilta, C., y Kanharattanachai, S. (2019). Municipal Solid Waste Segregation with CNN. 2019 5th International Conference on Engineering, Applied Sciences and Technology (ICEAST), 1–4. <https://doi.org/10.1109/ICEAST.2019.8802522>

16. Thung, G. (2017). garythung/trashnet: Dataset of images of trash; Torch-based CNN for garbage image classification. <https://github.com/garythung/trashnet?tab=readme-ov-file>
17. Xiaohoua. (2023). Garbage_HuaWei. <https://www.kaggle.com/datasets/xiaohoua/garbage-huawei>
18. Zhang, Q., Yang, Q., Zhang, X., Bao, Q., Su, J., y Liu, X. (2021). Waste image classification based on transfer learning and convolutional neural network. Waste Management, 135, 150–157. <https://doi.org/10.1016/j.wasman.2021.08.038>

DATOS DE LOS AUTORES.

1. **Mirna Castro Bello.** Doctora en Ciencias Ambientales. Docente del Tecnológico Nacional de México, Instituto Tecnológico de Chilpancingo. Email: mirna.cb@chilpancingo.tecnm.mx
2. **Víctor Manuel Romero Juárez.** Estudiante de la Maestría en Ciencias de la Ingeniería en el Tecnológico Nacional de México, Instituto Tecnológico de Chilpancingo. Email: MG16520276@chilpancingo.tecnm.mx
3. **Jorge Fuentes Pacheco.** Doctor y Maestro en Ciencias de la Computación. Docente del Tecnológico Nacional de México, Centro Nacional de Investigación y Desarrollo Tecnológico. Email: jorge.fp@cenidet.tecnm.mx
4. **Sergio Ricardo Zagal Barrera.** Maestro en Educación. Docente del Tecnológico Nacional de México, Instituto Tecnológico de Chilpancingo. Email: sa@chilpancingo.tecnm.mx
5. **Areli Bárcenas Nava.** Maestra en Desarrollo de Ingeniería de Software. Docente del Tecnológico Nacional de México, Instituto Tecnológico de Chilpancingo. Email: arebar29@gmail.com ó dir@chilpancingo.tecnm.mx
6. **Diego Esteban Gutiérrez Valencia.** Maestro en Ciencias, Doctorante en la Facultad de Matemáticas de la Universidad Autónoma de Guerrero. Docente del Tecnológico Nacional de México, Instituto Tecnológico de Chilpancingo. Email: diego.gv@chilpancingo.tecnm.mx

RECIBIDO: 26 de febrero del 2025.

APROBADO: 5 de abril del 2025.