

YOLOv8 Algorithm Implementation for Detecting and Classifying Waste Types Using CCTV Cameras

N P Dharshinni^{1*}, Muhammad Fikri Akbar Zuhdi², Agung Prabowo³

^{1,3}Informatics Engineering, Faculty of Science and Technology, Universitas Prima Indonesia, Medan 20118, Indonesia

²Visual Communication Design, Faculty of Science and Technology, Universitas Prima Indonesia, Medan 20118, Indonesia

Corresponding Author Email: priyadharshinninaidu@gmail.com

ABSTRACT

Received : 20 June 2025
Revised : 28 June 2025
Accepted : 10 July 2025
Available online : 24 July 2025

Keywords: *YOLOv8, Waste Classification, CCTV, Deep Learning, Computer Vision, Image Processing*

Waste accumulation has emerged as a critical global environmental concern, prompting the need for intelligent and automated waste management solutions. This study presents the development of a real-time waste detection and classification system using closed-circuit television (CCTV) cameras integrated with the You Only Look Once version 8 (YOLOv8) deep learning algorithm. The dataset used in this research was collected from two main sources: direct image captures from various physical locations and publicly available waste image datasets from the Kaggle platform. All images underwent preprocessing steps including resizing to 640×640 pixels, normalization, and annotation to ensure data consistency and quality. To enhance the model's robustness, data augmentation techniques such as horizontal flipping, cropping, rotation, brightness adjustment, and mosaic transformation were applied. The dataset was split into 80% for training and 20% for testing. The YOLOv8 model was trained with optimized hyperparameters suitable for object detection tasks. Performance evaluation was conducted using standard metrics including precision, recall, and mean average precision (mAP). The model achieved a precision of 0.486, recall of 0.361, mAP@50 of 0.368, and mAP@50–95 of 0.273. The detection accuracy varied across different waste categories, with higher performance observed for aluminum cans and cardboard, while lower performance was noted for aerosols and aluminum caps. These results indicate that YOLOv8 shows potential for waste classification but still requires improvements. Future work should focus on expanding dataset diversity, refining data augmentation strategies, and further hyperparameter tuning. This study contributes to the advancement of intelligent waste management through the application of deep learning and computer vision.

1. INTRODUCTION

Waste is defined as material that is considered to have no economic value and is deemed useless, both in general contexts and in specific industrial processes. It includes damaged or defective items produced during manufacturing, as well as excess materials from households [1]. Waste is the residue of routine human activities or natural processes, and it exists in solid or semi-solid form. It may consist of recyclable or disposable materials, biodegradable or non-biodegradable, and is considered unwanted, leading to its disposal into the environment [2]. Waste is also defined as the unwanted and disliked remains left after human use and is a product of non-biological human activities [3]. Waste can originate from various sources, such as residential areas, typically from households living in buildings or dormitories [4]. Waste is generally classified into organic (e.g., food scraps or wet waste) and inorganic (e.g., plastic, cans, glass bottles, etc.) categories [5].

Waste is one of the persistent issues in Indonesia and even globally. Waste production increases in line with population growth and rising levels of public consumption [6]. It remains a critical concern in Indonesia [7]. According to the Ministry of Environment and Forestry,

the total waste generated in Indonesia in 2019 reached approximately 66–67 million tons, an increase from 64 million tons the previous year. Of this amount, plastic waste accounted for 15%, while organic waste made up 60%. Based on information from the National Waste Management Information System (SIPSN), the total waste generated in Indonesia reached 25,200,822.85 tons per year, as reported by 217 regencies/municipalities across the country in 2023 [8]. Of that total, 50.43% or approximately 12,708,249.48 tons per year were managed, with 66.32% or 16,712,597.55 tons properly handled, while 33.68% or 8,488,243.30 tons remained unmanaged. This is closely related to ongoing human activities, which continue to produce waste, both organic and non-organic. When improperly disposed of in the surrounding environment, waste can have a detrimental impact on both ecosystems and human well-being. Waste materials—whether recyclable or single-use—when accumulated in large quantities, can serve as breeding grounds for harmful pests such as rats, cockroaches, stray cats, dogs, and flies. Because such conditions pose a threat to public health and environmental aesthetics, effective and systematic waste management is essential.

Due to its ability to replicate human visual perception, image classification has become one of the most prominent fields in machine learning [9]. Image classification refers to a process in computer vision that enables the categorization of images based on their visual content. By utilizing image classification, computers can rapidly identify types of waste such as cardboard, plastic, paper, metal, and glass from images.

An algorithm capable of extracting image features is required to recognize objects within an image [10]. YOLO (You Only Look Once), which falls under the category of Convolutional Neural Network (CNN) algorithms that mimic the workings of the human brain, is one of the most widely used object detection algorithms [11]. The YOLO deep learning algorithm utilizes a convolutional neural network (CNN) to detect objects. The CNN process consists of three stages: pre-processing, processing, and classifying [12]. Since its initial introduction in 2015, YOLO has rapidly evolved as an object detection framework, with the latest version, YOLOv8, released in January 2023 [13]. The use of deep learning models like YOLO has attracted significant attention in the field of computer vision. YOLO has proven effective in high-speed object detection, which is crucial for real-time applications [14]. In terms of detection time, YOLOv5 excels due to its high inference speed, enabling object detection in milliseconds, especially when supported by a GPU. YOLOv8, with its diverse model variants, offers competitive inference performance, making it well-suited for scenarios with varying computational capabilities [15].

2. LITERATUR REVIEW

Previous studies utilizing the YOLO algorithm have demonstrated its effectiveness across various object detection tasks. For instance, research conducted by Illmawati applied YOLOv5 for vehicle number plate detection and achieved an impressive detection accuracy of 97.2% [16]. Similarly, Sinaga implemented the YOLO algorithm to detect objects on an e-commerce platform, yielding an accuracy rate of 95% [17], highlighting its capability in commercial applications. Nadia's study focused on automatic defect detection on solar panel surfaces using YOLOv8, which resulted in an accuracy of 90.5% [18], showing its potential in industrial quality inspection. Saadeldin's research utilized YOLOv8 for real-time vehicle counting, particularly in resource-constrained edge devices, and achieved an accuracy of 97.5% [19], emphasizing YOLOv8's reliability and efficiency in traffic monitoring systems. Additionally, Sarosa implemented the YOLO algorithm for detecting disaster victims in emergency situations, with an accuracy of 95.3% [20], suggesting its applicability in humanitarian and search-and-rescue operations. Maulidiansyah's research applied YOLO for classifying types of corn based on kernel shape, achieving a high classification accuracy of 98% [21], demonstrating its capability in agricultural applications.

These studies collectively indicate that the YOLO algorithm, including its latest versions like YOLOv8, consistently delivers high accuracy across a wide range of domains—from transportation and e-commerce to renewable energy, agriculture, and disaster management. This versatility and precision make YOLO a strong candidate for applications requiring real-time

object detection and classification, including waste classification systems as proposed in the present study.

3. RESEARCH METHOD

In this study, YOLOv8, the most recent version of the YOLO Deep Learning algorithm, will be utilized. The research methodology applied in this study can be generally outlined as follows:

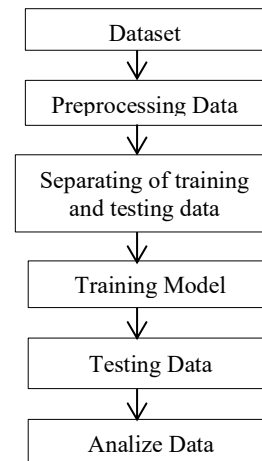


Figure 1. Research Framework

3.1 Dataset

The dataset for this study was collected from the Roboflow website. Roboflow provides a variety of waste images that have been annotated according to standardized labeling practices, which facilitates the data collection process and ensures dataset diversity. Utilizing a dataset from Roboflow enables researchers to access well-organized and properly labeled images suitable for training the YOLOv8 model. The dataset configuration details are as follows:

- Training Set: The dataset used for model training was obtained from `../train/images`.
- Validation Set: The dataset used for model validation was obtained from `../valid/images`.
- Test Set: The dataset used for model testing was obtained from `../test/images`.

The dataset configuration is as follows:

- Number of Classes (nc): 10
- Class Names (names): ['battery', 'can', 'cardboard', 'drink carton', 'glass bottle', 'paper', 'plastic bag', 'plastic bottle', 'plastic bottle cap', 'pop tab'].

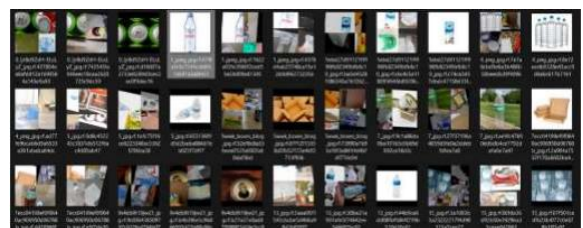


Figure 2. Images of Waste Objects

3.2 Preprocessing

The dataset obtained from Roboflow first undergoes a preprocessing process to ensure that the image quality is adequate and consistent before being used in training the artificial intelligence model. One of the initial stages carried out is Auto-Orient, which is the process of adjusting the image orientation so that all images have a uniform direction, such as portrait or landscape. Non-uniform orientation can cause misinterpretation of objects by the model due to differences in viewing direction. This step is important so that the model can recognize objects more accurately, without being disturbed by varying image positions or directions.

The next stage is Resize, which is adjusting the image size to a fixed dimension of 640x640 pixels. This process uses the fit method, where the image will be adjusted in proportion without distortion, and if necessary, a black edge (padding) will be added to maintain the ratio. This is important because convolutional models require input with a uniform size to work optimally. If the image sizes are different, the feature extraction results can be inconsistent, which ultimately reduces model performance.

The Null Filter process is then applied to filter out images that do not have annotations. Annotations are an important component in model training because they indicate the location and type of objects in the image. Without annotations, the model will not be able to learn effectively because there is no target information to refer to. Therefore, images that do not contain annotations are removed from the dataset so as not to interfere with the training process and to maintain high data quality.

To increase data diversity and prevent the model from overfitting, various augmentation techniques are used. This augmentation creates three different variations of each training image so that the model can see objects in different angles and conditions. One of the augmentation techniques used is horizontal flip, which helps the model recognize objects from different sides. In addition, cropping is done with an enlargement (zoom) of up to 20% of the image area to mimic the variation in viewing distances in real-world shooting.

Additional augmentations include image rotation in the range of -15° to $+15^{\circ}$, which simulates the tilt of the object when the image was taken. Brightness adjustments between -25% to $+25\%$ are also applied so that the model can recognize objects in different lighting. The last technique used is mosaic, which combines multiple images into one frame. This technique allows the model to learn the relationship between objects in a more complex visual context, which will ultimately increase the model's generalization power to new data. With this combination of preprocessing and augmentation, the resulting dataset becomes more robust, varied, and representative of real conditions.

After all the images in the dataset have been processed through the preprocessing and augmentation stages, the next step is to divide the dataset into two main parts: training data (training set) and validation data (validation set). The goal is for the model training process to run effectively and the results obtained can be evaluated more accurately.

Most of the images, namely 5,220 images or around 92% of the total dataset, are allocated as the training set. This data is used to "teach" the YOLOv8 model how to recognize and understand various types of objects in images. During the training process, the model will continue to update its weight based on the patterns found from these images.

Meanwhile, the rest - 435 images or around 8% - are used as the validation set. These images are not included in the training process, but are used to check whether the model is really learning well. So, even though the model has recognized objects from the training data, its performance is still tested on data that has never been seen before. This helps us find out whether the model is just "memorizing" the data or is really able to understand patterns and apply them to new situations.

This division is very important so that the model is not only smart on paper, but also reliable when applied in the real world. By monitoring the validation results periodically, we can find out whether the training is going well or whether it needs to be adjusted, for example by changing the training parameters or improving the dataset. In other words, the training set is like the main teaching material for the model, while the validation set is a test to see how far the model understands the lessons that have been given. The combination of the two allows us to build a system that is not only accurate, but also stable and ready for widespread use.

4. RESULTS AND DISCUSSION

4.1 YOLOv8 Architecture

YOLOv8 is the latest version of the YOLO algorithm released in 2020. Compared to the previous version, YOLOv8 comes with tremendous improvements in terms of speed and accuracy. With this capability, object detection can be done in real-time without any lag, something that was still lacking in the previous version [22]. This makes YOLOv8 a very promising choice in various computer vision applications that require high speed and accuracy.

One of the major improvements in YOLOv8 is the use of a more sophisticated backbone network, namely the Feature Pyramid Network (FPN) technology. This technology allows the model to extract features from images of various sizes, so that small and large objects can be recognized well. This is important because in the real world, objects can appear in different sizes and positions.

YOLOv8 also has a better prediction system with multi-scale capabilities and a new anchor mechanism. This means that this model can make predictions at various levels of resolution and is able to detect objects of various shapes and sizes more accurately. The model structure itself consists of three main parts: backbone, neck, and head. Each part has an important task to process information from the image to produce accurate predictions.

In the neck section, YOLOv8 uses the Cross-Layer Connection (CLC) approach to refine the features that have been taken from the image. Meanwhile, the head section is tasked with producing the final prediction, starting from the object class, object position (bounding box), to its level of confidence. This model consists of 105 layers designed to overcome complex object detection challenges.

Overall, YOLOv8 relies on 24 convolutional layers and two fully connected layers, with an input image size of 448 x 448 x 3 pixels. The combination of 3x3 convolutional layers and 1x1 reduction layers is used to deepen the feature extraction process. All these updates allow YOLOv8 to work faster and more accurately in recognizing objects, making it a very reliable solution for various real-time detection needs [22].

However, when a tighter threshold is used, namely mAP50-95 (with an accuracy level ranging from 50% to 95%), the value decreases to 0.273. This shows that the model's performance is less stable when required to be more precise in the position and size of object detection. This decrease in value is reasonable, but it also shows that there is still room to improve the quality of detection, especially in handling objects with small sizes or shapes that are like each other.

Overall, these results show that the YOLOv8 model is quite good at detecting certain types of trash but still needs to be improved to be more reliable, especially in recognizing more correct objects without losing accuracy.

5. CONCLUSION

This study shows that the YOLOv8 model is capable of detecting and classifying various types of waste, although the results are still varied. In general, the model recorded a precision of 0.486 and a recall of 0.361, which means that almost half of the detections made were correct, but many objects were still missed. The mAP50 value of 0.368 shows quite good performance at loose overlap tolerance limits, while the mAP50-95 of only 0.273 shows that the model's performance decreases when more precision is required. Interestingly, several types of waste such as "Aluminum can" and "Cardboard" were successfully recognized very well, with precision and recall above 0.7, even approaching 0.9. However, there are also classes that are completely undetected, such as "Aerosols" and "Aluminum caps", which shows that there is still room for improvement, especially in terms of the completeness and diversity of the training data used.

6. SUGGESTION

Based on the findings in this study, there are several suggestions that can be applied for further research to improve model performance. First, it is recommended to increase the number and expand the variety of training data, especially for classes that show low performance such as "Aerosols" and "Aluminum caps", so that the model has more examples to learn from and can recognize these objects better. Collecting additional data from different environments, viewpoints, and lighting conditions is also important to improve the model's ability to deal with variations in the real world. In addition, the application of advanced augmentation techniques such as gaussian blur, color jitter, and random erasing can enrich the diversity of training data, thereby helping the model become more adaptive. Optimizing hyperparameters such as learning rate, batch size, and number of epochs also needs to be considered to obtain the most effective training configuration. To evaluate how well the model can perform on new data, testing with other datasets that are similar or come from different sources is highly recommended. Finally, the application of the cross-validation method can be used to ensure that the model not only performs well on one particular subset of data, but is able to maintain its performance consistently, thereby reducing the risk of overfitting.

REFERENCES

- [1] Rima Dias Ramadhani, A. Nur Aziz Thohari, C. Kartiko, A. Junaidi, T. Ginanjar Laksana, and N. Alim Setya Nugraha, "Optimasi Akurasi Metode Convolutional Neural Network untuk Identifikasi Jenis Sampah," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 2, pp. 312–318, Apr. 2021, doi: 10.29207/resti.v5i2.2754.
- [2] Undang-undang (UU) Nomor 18 Tahun 2008 tentang Pengelolaan Sampah. 2008. Accessed: Jun. 26, 2024. [Online]. Available: <https://peraturan.bpk.go.id/Details/39067/uu-no-18-tahun-2008>.
- [3] E. Sutanty, D. K. Astuti, and Maukar, "Penerapan Model Arsitektur VGG16 Untuk Klasifikasi Jenis Sampah," *Decode: Jurnal Pendidikan Teknologi Informasi*, vol. 3, no. 2, pp. 407–419, Sep. 2023, doi: <https://doi.org/10.51454/decode.v3i2.331>.
- [4] Hanna Meilani Damanik, M. L. Purba, H. E. S. Samosir, Nancy Nopeline, and M. B. Lumban Gaol, "Upaya Peningkatan Ekonomi Anggota Bank Sampah 'Satu Hati', melalui Pengelolaan dan Pemanfaatan Limbah Sampah di Kelurahan Sumber Mulyo Rejo, Binjai Timur," *Jurnal Visi Pengabdian Kepada Masyarakat*, vol. 4, no. 1, pp. 12–22, Feb. 2023, doi: 10.51622/pengabdian.v4i1.1141.
- [5] Ambar Tri Ratnaningsih, David Setiawan, and Latifa Siswati, "Pemberdayaan Masyarakat Melalui Pemanfaatan Sampah Anorganik Menjadi Produk Kerajinan yang Bernilai Ekonomis," *Dinamisia: Jurnal Pengabdian Kepada Masyarakat*, vol. 5, no. 6, pp. 1500–1506, Dec. 2021, doi: 10.31849/dinamisia.v5i6.5292.
- [6] K. L. Kohsasih, M. D. A. Rizky, T. Fahriyani, V. Wijaya, and R. Rosnelly, "Analisis Perbandingan Algoritma Convolutional Neural Network dan Algoritma Multi-layer Perceptron Neural dalam Klasifikasi Citra Sampah," *Jurnal TIMES*, vol. 10, no. 2, pp. 22–28, Dec. 2021, doi: <https://ejournal.stmiktime.ac.id/index.php/jurnalTIMES/article/view/655>.
- [7] H. Hendar, T. Rezasyah, and D. S. Sari, "Diplomasi Lingkungan Indonesia Melalui ASEAN dalam Menanggulangi Marine Plastic Debris," *Padjadjaran Journal of International Relations*, vol. 4, no. 2, p.201, Aug. 2022, doi: 10.24198/padmir.v4i2.40721.
- [8] "CAPAIAN KINERJA PENGELOLAAN SAMPAH." Accessed: Jun. 26, 2024. [Online]. Available: <https://sipsn.menlhk.go.id/sipsn/>.
- [9] "Analisis Perbandingan Algoritma SVM, KNN, dan CNN untuk Klasifikasi Citra Cuaca," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 8, no. 2, pp. 311–318, Mar. 2021, doi: 10.25126/jtiik.2021824553.
- [10] F. Indaryanto, A. Nugroho, and A. Suni, "Aplikasi Penghitung Jarak dan Jumlah Orang Berbasis YOLO Sebagai Protokol Kesehatan Covid-19," *Edu Komputika Journal*, vol. 8, no. 1, Jun. 2021, doi: 10.15294/edukomputika.v8i1.47837.
- [11] A. Setiyadi, E. Utami, and D. Ariatmanto, "Analisa Kemampuan Algoritma YOLOv8 Dalam Deteksi Objek Manusia Dengan Metode Modifikasi Arsitektur," *J-SAKTI (JURNAL SAINS KOMPUTER DAN INFORMATIKA)*, vol. 7, no. 2, pp. 891–901, Sep. 2023.
- [12] Y. Yunefri, S. Sutejo, Y. E. Fadrial, K. Anggraini, M. Ramadhani, and R. Hardianto, "Implementation of Object Detection With You Only Look Once Algorithm in Limited Face-To-Face Times in Pandemic," *Journal of Applied Engineering and Technological Science (JAETS)*, vol. 4, no. 1, pp. 400–404, Dec. 2022, doi: 10.37385/jaets.v4i1.1161.
- [13] M. Hussain, "YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection," *Machines*, vol. 11, no. 7, Jul. 2023, doi: 10.3390/machines11070677.
- [14] A. Nugroho et al., "PEMANFAATAN TEKNOLOGI COMPUTER VISION BERBASIS YOLO UNTUK MENDETEKSI KERUMUNAN DI SMKN 4 MALANG," *Jurnal Abdi Insani*, vol. 11, no. 1, pp. 678–689, Mar. 2024, doi: 10.29303/abdiinsani.v11i1.1437.

- [15] Erwin Syahrudin, Ema Utami, and Anggit Dwi Hartanto, "Enhanced Yolov8 with OpenCV for Blind Friendly Object Detection and Distance Estimation," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 8, no. 2, pp. 199–207, Mar. 2024, doi: 10.29207/resti.v8i2.5529.
- [16] Reezky Illmawati and Hustinawati, "YOLO V5 untuk Deteksi Plat Kendaraan di DKI Jakarta," *Jurnal Ilmu Komputer dan Agri-Informatika*, vol. 10, no. 1, pp. 32–43, May 2023, doi: 10.29244/jika.10.1.32-43.
- [17] F. M. Sinaga, Gunawan, S. Winardi, H. Kurniawan, W. S. Lestari, and K. M. Tarigan, "Object Detection in E-Commerce Using YOLO in Real Time," *Teknika*, vol. 13, no. 1, pp. 145–154, Mar. 2024, doi: 10.34148/teknika.v13i1.773.
- [18] D. Nadia and C. Fathia, "Automatic detection of solar cell surface defects in electroluminescence images based on YOLOv8 algorithm," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 32, no. 3, pp. 1392–1404, 2023, doi: 10.11591/IJEECS.V32.I3.PP1392-1404.
- [19] A. Saadeldin, M. M. Rashid, A. A. Shafie, and T. F. Hasan, "Real-time vehicle counting using custom YOLOv8n and DeepSORT for resource-limited edge devices," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 22, no. 1, pp. 104–112, Feb. 2024, doi: 10.12928/TELKOMNIKA.v22i1.25096.
- [20] M. Sarosa and N. Muna, "IMPLEMENTASI ALGORITMA YOU ONLY LOOK ONCE (YOLO) UNTUK DETEKSI KORBAN BENCANA ALAM," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. 8, no. 4, pp. 787–792, Aug. 2021, doi: 10.25126/jtiik.202184407.
- [21] I. Abdillah and Maulidiansyah, "Klasifikasi Jenis Jagung Berdasarkan Bentuk Biji Menggunakan Metode You Only Look Once (YOLO)," *Journal of Electrical Engineering and Computer (JEECOM)*, vol. 5, no. 2, pp. 215–226, Oct. 2023, doi: 10.33650/jeeecom.v4i2.
- [22] J. Terven, D. M. Córdova-Esparza, and J. A. Romero-González, "A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS," *Mach Learn Knowl Extr*, vol. 5, no. 4, pp. 1680–1716, Dec. 2023, doi: 10.3390/make5040083