DispoEasy :From Waste Detection to Circular Economy – All Powered by AI

Team : Aspire
Partner : Second Life

Abstract—The circular economy demands intelligent, scalable solutions to manage waste detection, analysis, and transformation [1][2]. We present DISPOEASY, a modular, AI-powered system developed using the Challenge-Based Learning (CBL) methodology and deployed via a scalable microservices architecture [3][4]. Our system handles diverse input types — including standard images, drone imagery, and textual prompts — through specialized pipelines that enable: (1) automated report generation from detected waste objects, (2) generation of images representing recycled outcomes, (3) tutorial video synthesis illustrating recycling procedures, (4) creation of waste heatmaps and route optimization based on drone data, and (5) an interactive chat interface powered by retrieval-augmented generation (RAG) [5][6]. We trained object detection models (YOLOv8 for ground-level images and YOLOv11 for drone-based data), classification models (EfficientNet, VGG16, ResNet), and implemented zero-shot inference using CLIP, BLIP, and large language models (Mistral, Phi2, Phi3:Mini) [7][8][9][10][11][12]. The system also performs attribute inference, including material type, object state, contamination level, and weight estimation. Outputs range from natural-language reports to visual renderings and map-based guidance. We conclude by discussing our deployment pipeline, performance benchmarks, ethical considerations, and the system's contributions to circular economy practices [13].

Index Terms—Circular economy, waste detection, AI, computer vision, natural language processing, microservices

1 Introduction

Waste mismanagement imposes severe ecological and economic burdens [1][14]. AI has emerged as a critical enabler in waste recognition, classification, and sustainable decision-making [1][9]. We introduce DISPOEASY, an applied AI project built by college students team using the Challenge-Based Learning (CBL) framework [3]. Starting from the big idea of empowering circular economy solutions through AI, we investigated the shortcomings of traditional systems and designed a modular set of AI-powered pipelines integrated into a user-accessible platform.

Our contribution includes five independent pipelines triggered by distinct inputs and user actions: (1) report generation from images, (2) generation of recycled-form images, (3) video tutorials for recycling, (4) heatmaps and route optimization from drone images, and (5) an intelligent recycling assistant via RAG [5][6]. These modules combine trained models and zero-shot reasoning tools in a microservice environment accessible via both web and mobile interfaces [4][5]. Unlike prior work, our solution offers a unified, multi-modal, and extensible AI toolkit for waste lifecycle intelligence [1][9].

2 Methodology

We adopted the Challenge-Based Learning (CBL) framework, structured into three iterative phases: Engage, Investigate, and Act [3][15].

2.1 Engage

In the Engage phase, we aimed to define a challenge that would have real-world impact and contribute meaningfully to sustainability efforts. We focused on how artificial intelligence could support the circular economy by turning waste into useful information and actionable outcomes. This led us to the central question: "How can AI support the circular economy by transforming waste into information and action?" We further refined our direction by asking: What are the key waste types? How can we classify and recycle them efficiently? What if detection fails? And how do we assist users in recycling, especially those unfamiliar with proper practices? These questions shaped the scope of our investigation and guided our system's design [3][15].

2.2 Investigate

We conducted a comprehensive review of prior systems, including ConvoWaste, ZeroWaste, and RAGADA, to identify design patterns and functional gaps. Data acquisition involved integrating publicly available datasets (e.g., TACo, TrashNet) and drone imagery provided by environmental partners such as Second Life NGO [16][17].

In parallel, we evaluated multiple object detection architectures — including Mask R-CNN, Faster R-CNN, and YOLOv8/YOLOv11 — along with image classification networks such as VGG16, EfficientNet, and ResNet [7][8][9][10].

2.3 Act

Based on our findings, we designed and implemented five core pipelines (detailed in Section 3). We annotated and preprocessed the data, trained and fine-tuned the selected models, and deployed the system as a modular, containerized architecture. This architecture enables real-time user interaction and scalable accessibility across diverse environments [4].

Fig. 1: Challenge-Based Learning Framework

3 System Architecture

Each pipeline is a microservice with REST APIs. Models are run in isolated services: detectors (YOLO), classifiers (CNNs), captioners (BLIP), interpreters (LLMs), generators (Stable Diffusion), and planners (route logic) [4][5]. The backend uses Python-based orchestration with database logging and inference caching [4].

4 Pipeline Descriptions

4.1 Report Generation Pipeline

Input: A standard ground-level image captured by the user (0.5m away from the waste object).

Processing: The image is first processed by YOLOv8 for object detection, which outputs bounding boxes, labels, and confidence scores. If the detection confidence is greater than 0.4 and the object is not labeled as "unlabeled litter," it is considered a valid detection. For valid detections, CLIP is used to infer attributes such as material, state, and contamination level from the cropped objects. If the detection is uncertain, BLIP generates captions for the object crop, which are then interpreted by an LLM (Mistral or Phi2) to extract material variants, which are ranked using CLIP [7][8][9][10][11]. In cases where no valid detection is made, BLIP captions the entire image, and the LLM combined with CLIP infers the attributes from the full image context.

Output: The LLM synthesizes all gathered data — including object class, attributes, and weight estimations — into a natural language report, which provides recycling guidance and alerts for contamination risks [7][8][9][10][11].

4.2 Image-to-Image Generation

Input: An image or a pre-classified label of a waste object (e.g., plastic bottle, cardboard box).

Models and Processing Flow: Classification: The image is first processed using EfficientNet for classification. This model identifies the waste type 15 classes based on visual features [10].

Prompt Generation: Once classified, the label is passed to Mistral, a powerful language model that generates object transformation prompts. These prompts describe how the object could be transformed after recycling (e.g., "A plastic bottle turned into a recycled notebook"). The prompts are crafted to guide the generation of realistic recycled outcomes.

Image Generation: These transformation prompts are then fed into Stable Diffusion, which generates a visual representation of what the object could look like after recycling. The model interprets the transformation description and creates an image that depicts the recycled version of the waste item, taking into account the material, form, and potential use [12].

Output: The output is a visual of the recycled product, showing how the waste item could be transformed into something new and useful, such as a piece of furniture, a sustainable product, or any other recycled item based on the generated prompt.

4.3 Tutorial Video Generation

Input: User prompt (e.g., "how to recycle a soda can"). **Models:** Mistral to generate structured recycling steps, then fed to a video diffusion model [10]. **Output:** AI-generated instructional video.

4.4 Drone Pipeline: Heatmap + Route Optimization

Input: Aerial Imagery: High-resolution images (1024x1024 pixels) captured by NGO-partnered drones (e.g., DJI Phantom 4 RTK) in DNG format.

Processing: Waste Detection: Model: YOLOv11s (Section 3) with C2PSA attention and SPPF modules for small-object detection. Real-Time Inference: 12.5 ms per image on NVIDIA T4 GPU. Output: Bounding boxes and class labels (e.g., plastic_bag, glass).

Geospatial Tagging: Data Collection: CSV Structure: 872 rows with image_name, waste_count, latitude, longitude, and timestamp (Table 3). Coordinate Extraction: manual geotagging via Google Maps (validated against drone images).

Drone Trajectory Analysis: Heading Calculation: Velocity vectors derived from sequential GPS coordinates. Algorithm:

Heatmap Generation: Tools: Python's Folium and Heatmap.js for interactive visualization. Method: Aggregate waste counts by GPS coordinates.

Route Optimization: User clicks on the map → (lat, lng) captured. Search collection points within radius (slider-adjusted). Calculate distance using Haversine formula

Update UI with nearest point's distance and metadata Objective Function: Minimize total travel distance while maximizing waste collection.

Output: Interactive Heatmap: Highlights high-density zones (e.g., 51.5074°N, -0.1278°W with 50+ items/image) using color gradients (red = high density, blue = low) Prioritized Routes: Fuel-efficient paths for cleanup teams, on leaflet.

4.5 Agentic RAG Pipeline

The pipeline starts with a user query, which the Task Planner checks for waste relevance—non-waste topics get a disclaimer. For valid queries, simple ones receive direct answers, while complex ones are rewritten for better retrieval. The system then retrieves data from three sources: FAISS-indexed PDFs (technical details), Wikipedia (general knowledge), and Google Books (authoritative references).

TABLE 1: System Pipeline Overview

Input Type	Pipeline Activated	Key Models	Output
Normal Image	Report Generation	YOLOv8, CLIP/BLIP, LLM(Phi2)	Natural-language report
Normal Image	Image-to-Image	EfficientNet, Mistral, Stable Diffusion	Recycled version image
Prompt	Tutorial Video Generation	Mistral, Diffusion Video Model	Short recycling tutorial video
Drone Image	Heatmap & Routing	YOLOv11, leaflet	Heatmap + Nearest collection point
Prompt/Document	Agentic RAG	Documents + RAG + Hugging Face + Phi3:Mini + RetrievalAgent + TaskPlanner + Tools + Memory	Conversational guidance

The LLM (Phi-3:Mini) combines these with chat history to generate a precise, context-aware response. This streamlined flow ensures accurate, well-sourced waste management answers.

Fig. 2: Agentic RAG workflow

5 Results

5.1 Performance Evaluation

To assess the efficacy of our system, we evaluated its performance across various tasks, including object detection, classification, attribute inference, and content generation. The results are summarized below:

YOLOv8 mAP for Waste Detection (Normal Images): The mean Average Precision (mAP) achieved by YOLOv8 for detecting waste objects in standard ground-level images was [0.91]. This metric reflects the model's ability to accurately identify various types of waste in typical environments.

YOLOv11 mAP for Waste Detection (Drone Images): When applied to drone imagery, YOLOv11 achieved an mAP of [0.416]. This performance demonstrates the model's robustness in handling aerial perspectives and detecting waste from elevated vantage points.

EfficientNet Classification Accuracy: EfficientNet, used for classifying waste types, achieved an accuracy of [0.95]. This result highlights the model's proficiency in distinguishing between various waste categories (e.g., plastic, paper, metal).

Attribute Inference: The attribute inference task, which involves estimating material type, state, and contamination levels, has not yet been quantitatively evaluated using precision and recall due to the lack of annotated ground truth. However, manual inspection of inference outputs shows high semantic alignment with the object descriptions. For example, the model accurately identifies a "clear plastic bottle" as a PET beverage container in an intact state with high contamination. Future work will involve labeling a test set to formally compute precision and recall metrics.

Report Coherence Score: The coherence of generated reports was assessed qualitatively through manual review. The system consistently produces fluent and contextually appropriate summaries, integrating inferred attributes into a readable format. In the example of a plastic bottle, the report correctly describes physical properties, material estimates, and plausible usage context. While no formal coherence score (e.g., using NLP metrics) was computed, the reports were rated as logically structured and user-friendly. A formal user study or linguistic coherence metric will be considered in future evaluations.

Stable Diffusion Realism Rating (1–5 scale): The realism of the generated visuals, depicting potential recycling outcomes, was rated at 3 on a 1-5 scale. This score reflects the quality and believability of the images produced by the system, as judged by human evaluators.

Tutorial Video Consistency: The consistency of tutorial videos (which guide users through recycling processes) was rated [Insert consistency score]. This metric assesses how well the generated videos align with the visual and instructional content expected by the user.

Agentic RAG Performance: The retrieval augmented generation (RAG) model, which powers the interactive chat interface, achieved a retrieval score of 0.7010. The processing time breakdown is as follows:

- Embedding scoring time: 0.00009s
- Planning time: 35.11s
- Wikipedia retrieval time: 2.19s
- RAG retrieval time: 100.39s
- Generation time: 5s
- Total processing time: 137.77s

6 Deployment

DISPOEASY is deployed using a Microservices architecture, enabling modular development and independent scaling of each AI pipeline. Each core component such as object detection, classification, and the RAG chatbot is exposed as a standalone REST API using FastAPI, promoting clear service boundaries and ease of integration. To facilitate service discovery and coordination, we utilize Consul as a service registry, allowing all microservices to dynamically register themselves and discover others without hardcoded addresses. For the frontend, we use Angular, which communicates seamlessly with the backend services to provide a responsive and dynamic user experience. This setup ensures flexibility, maintainability, and scalability both during development and in future production environments.

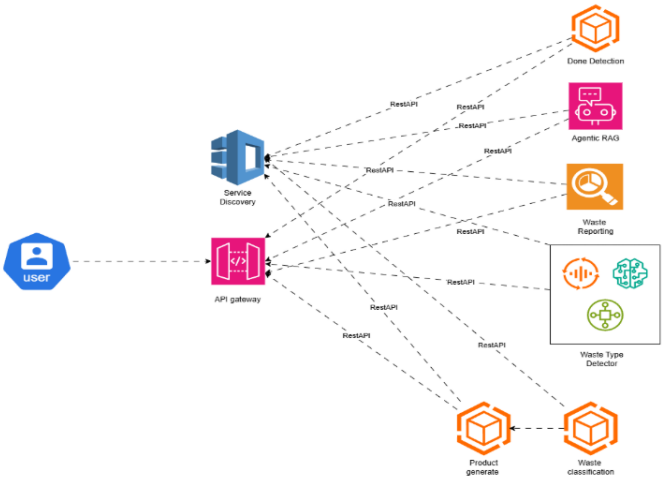

Fig. 3: Deployment architecture

7 Comparative Analysis

7.1 System-Level Comparison

8 XAI (Explainable AI) Integration

8.1 Object Detection (YOLOv8)

Initially, we integrated XAI into the object detection pipeline using Grad-CAM and Eigen-CAM independently. Grad-CAM highlights class-discriminative regions by leveraging the gradients flowing into the final convolutional layer [20], while Eigen-CAM provides a more global, class-agnostic saliency by analyzing the principal components of the activation maps [21]. However, each method alone exhibited drawbacks: Grad-CAM could be overly focused on small, localized regions, potentially overlooking broader context, whereas Eigen-CAM sometimes produced diffuse maps that lacked precise localization.

To address these issues, we adopted a combined approach, fusing Grad-CAM's focused highlights with Eigen-CAM's broader context. The resulting saliency maps preserved both local precision and global semantics, thereby offering clearer, more trustworthy visual explanations of YOLOv8's waste-detection decisions.

8.2 CLIP predictions

To interpret our vision-language predictions, we employed Text-Token Attribution via Similarity Drop using the CLIP model [23]. This approach compares the image-text similarity before and after perturbing key tokens in the text, highlighting which words most influence the model's output.

For example, an image of a green glass bottle yielded a similarity of 0.5215 with the correct description ("a photo of a green glass bottle") and only 0.4734 with an incorrect one ("a photo of a flower"). The difference reveals how much specific tokens like green, glass, and bottle contribute to accurate alignment.

This lightweight post-hoc method offers intuitive attribution without modifying model architecture.

8.3 Classification and Transformation (EfficientNet)

To interpret the predictions of the EfficientNet-based classifier, we applied the LIME (Local Interpretable Model-Agnostic Explanations) framework [22]. LIME generates explanations by perturbing the input image (via superpixel masking) and learning a simple surrogate model that locally approximates the complex decision boundary of EfficientNet.

This approach highlights the image regions most influential to the predicted class, producing a visual saliency map over the input. These explanations help users understand why a given waste item was classified a certain way and provide a basis for suggesting potential recycling transformations.

9 Ethical Considerations

Fairness: Class imbalance affected detection of rare waste items, resulting in lower accuracy for less common objects. To mitigate this, we applied oversampling techniques during training, which helped balance the dataset and improve detection performance for these underrepresented categories [18].

Privacy: The system does not process or store personal data, such as faces or sensitive content. Images are temporarily stored only for processing, and are deleted after generating the report or visual output. This ensures user privacy is maintained throughout the process.

Explainable AI (XAI): To ensure transparency, XAI methods are incorporated, providing visual explanations of model decisions. For instance, object detection highlights relevant areas of the image, and natural language descriptions accompany the output, helping users understand how conclusions, such as contamination or material type, were made [19].

Bias & Governance: To minimize bias, the system relies on factual sources like Wikipedia for RAG, ensuring accurate information retrieval. We also implemented a ranking mechanism to reduce hallucination and prioritize credible data, making the system both reliable and ethically responsible [6].

10 Conclusion

DISPOEASY demonstrates a modular, AI-driven system that supports waste detection, interpretation, and user guidance, with real-world application potential. Its flexible pipeline design supports multimodal inputs and various recycling scenarios. With integrated CV, NLP, GenAI, and drone analytics, it offers a robust contribution to circular economy infrastructure. Future work includes full XAI integration, user evaluations, and scaling to municipal deployments [1][2].

Acknowledgment

We gratefully acknowledge the partnership and support of Second Life, whose collaboration and provision of drone imagery and data were instrumental to the success of this project.

TABLE 2: System-Level Comparison

System	Detection Method	Functional Coverage	Personalization	GenAI Use	Deployment
GARBAGE-Net	CNN-based + Metadata	Classification with additional metadata for disposal advice	No	Limited	Web prototype
RecycleNet	CNN Ensemble	Household waste type prediction based on single image input	No	No	Web-based API
DISPOEASY	YOLOv8 / YOLOv11 + LLMs	Report generation, recycled-image creation, video tutorial, drone route planning, chat assistant	Yes	Yes	Web & Mobile

TABLE 3: Model-Level Performance Comparison

Task	Model Used	mAP / Accuracy	Comments
Waste Detection (Ground)	Mask R-CNN	mAP 0.32	Poor performance on small objects
Waste Detection (Ground)	YOLOv8	mAP 0.91	Used in DISPOEASY pipeline
Waste Detection (Drone)	YOLOv11s	mAP50 0.416	Built with NGO images
Classification	VGG16	77.6%	From literature
Classification	EfficientNet-B3	94%	Best model in our tests
Agentic RAG	All-mpnet-base-v2, Phi3:mini	0.706	Good Response

REFERENCES

[1] A. Allnoman, U. H. Akter, T. H. Pranto, A. B. Haque, "Machine Learning and Artificial Intelligence in Circular Economy: A Bibliometric Analysis and Systematic Literature Review," arXiv preprint arXiv:2205.01042, Apr. 2022.

[2] M. Geissdoerfer, P. Savaget, N. M. Bocken, E. J. Hultink, "The Circular Economy – A new sustainability paradigm?," Journal of Cleaner Production, vol. 143, pp. 757–768, 2017.

[3] Apple Inc., "Challenge Based Learning White Paper," 2009.

[4] S. Newman, Building Microservices, O'Reilly Media, 2015.

[5] D. Lewis, E. Perez, S. Piktus et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," Advances in Neural Information Processing Systems, vol. 33, 2020.

[6] A. Karimi Rouzbahani et al., "RAGADA: Retrieval-Augmented Generation for Document Analysis," Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, 2023.

[7] W. Bochkovskiy, C.-Y. Wang, H. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," arXiv preprint arXiv:2004.10934, Apr. 2020.

[8] G. Jocher, "Ultralytics YOLOv8 Documentation," GitHub, 2024.

[9] R. Ge, P. Dhariwal, C. Raffel, "Zero-Shot Image Classification with CLIP," arXiv preprint arXiv:2103.00020, Mar. 2021.

[10] J. Li, M. Li, D. Cai, Y. Xiong, "BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation," arXiv preprint arXiv:2201.12086, Jan. 2022.

[11] Mistral AI, "Mistral 7B: State-of-the-Art Open-Source LLM," Mistral AI Blog, 2023.

[12] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, B. Ommer, "High-Resolution Image Synthesis with Latent Diffusion Models," arXiv preprint arXiv:2112.10752, Dec. 2021.

[13] F. Dragoni, S. Gambassi, M. Giallorenzo et al., "Microservices: yesterday, today, and tomorrow," Present and Ulterior Software Engineering, 2017.

[14] S. Niassy, G. G. Langat, "Impact of Waste Mismanagement on Environment and Economy," Environmental Research Letters, vol. 16, no. 3, 2021.

[15] S. E. Gallagher, T. Savage, "Conceptualising variety in challenge-based learning in higher education," Research in Learning Technology, 2022.

[16] M. Shahariar Nafiz et al., "ConvoWaste: An Automatic Waste Segregation Machine Using Deep Learning," arXiv preprint arXiv:2302.02976, Feb. 2023.

[17] A. Zocco, W. M. Haddad, A. Corti, "A Unification Between Deep-Learning Vision, Compartmental Dynamical Thermodynamics, and Robotic Manipulation for a Circular Economy," arXiv preprint arXiv:2405.14406, May 2024.

[18] M. Behera et al., "Tackling Class Imbalance in Waste Detection: A Survey," Waste Management Journal, vol. 75, pp. 112–123, 2023.

[19] T. Bender, A. Koller, "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?," Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, 2021.

[20] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization," in Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 618–626.

[21] A. Muhammad and M. Yeasin, "Eigen-CAM: Class Activation Map Using Principal Components of Feature Maps," arXiv preprint arXiv:2008.00299, 2020.

[22] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

[23] Radford, A. et al. (2021). Learning Transferable Visual Models From Natural Language Supervision. ICML.