electronics

MDPI

Article

Object-to-Pixel Visual Understanding for Automated Packaging Waste Sorting in Sustainable Educational Recycling Ecosystems

Fei Yu [1], Ruifeng Liu [2], Gufeng Gong [2,*] and Liujun Li [3]

[1] School of Education, Teesside University (In Partnership with Amity Global Institute), 228 Orchard Road, Singapore 238853, Singapore; e4445979@live.tees.ac.uk
[2] School of Computer Science and Mathematics, Central South University of Forestry and Technology, Changsha 410004, China; t20061093@csuft.edu.cn
[3] Department of Soil and Water Systems, University of Idaho, Moscow, ID 83844, USA; liujunl@uidaho.edu
* Correspondence: t20090523@csuft.edu.cn

Abstract

Automated packaging waste sorting requires rapid object localization and accurate boundary delineation under irregular shapes, deformation, reflection, dense stacking, and partial occlusion. This study presents an edge AI-enabled object-to-pixel framework in which PSSM-YOLO performs object detection and CAA-DeepLabv3+ independently performs semantic segmentation on complete RGB images. EPWaste contains 4528 RGB images and 11,384 annotated object instances from ten representative categories, with bounding-box and pixel-level annotations for detection and segmentation. PSSM-YOLO combines multi-head self-attention, an SRG multi-scale feature fusion module, Shuffle Attention, and a progressively refined detection head. CAA-DeepLabv3+ combines channel attention aggregation with CARAFE content-aware upsampling to reconstruct weak and irregular boundaries. On EPWaste, PSSM-YOLO achieved 87.2% mAP50, 74.1% mAP50:95, 93.6% precision, and 85.8% recall. CAA-DeepLabv3+ achieved 82.4% *mIoU*, 83.6% *mPA*, 83.3% *mPrecision*, and 83.6% *mRecall* on complete images. For 640 x 640 input, PSSM-YOLO and CAA-DeepLabv3+ required 12.8 M and 6.7 M parameters and 32.6 and 19.8 GFLOPs, respectively. On an NVIDIA Jetson Orin Nano, INT8 quantization with structured pruning reduced PSSM-YOLO latency to 11.1 ms (90.1 FPS) and power to 5.1 W. In a laboratory-scale conveyor test, the prototype correctly sorted 938 of 974 objects (96.30%), with a throughput of 58.3 objects per minute and an end-to-end latency of 312 ms. These controlled results support the feasibility of the framework while motivating broader cross-environment validation.

Keywords: packaging waste sorting; object-to-pixel learning; edge artificial intelligence; object detection; semantic segmentation; content-aware upsampling; model quantization; automated sorting

Check for updates

Academic Editor: Xian-Bo Wang

Received: 30 July 2026
Revised: 24 September 2026
Accepted: 27 September 2026
Published: 30 October 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

The increasing global focus on ecological issues, including environmental pollution and climate change, has made the advancement of sustainable and effective urban waste management systems a critical concern in international environmental governance [1,2]. The swift growth of e-commerce and courier services has markedly exacerbated the production of packaging waste, rendering it a critical global challenge for sustainable development [3]. By 2025, it is projected that Germany and the United States will generate 5 billion and 25 billion courier shipments yearly, respectively, resulting in substantial packaging waste [4]. In 2023, China recorded 132.07 billion express deliveries, reflecting a 58.3%

rise from 2020 [5], hence highlighting the concern about courier package waste. Packaging materials come in many different types, from corrugated cardboard boxes to plastic bags, paper envelopes and foam containers, which all present significant challenges for recycling, resulting in low recycling efficiencies. The recycling rate of corrugated cardboard boxes is about 50% [3], but most other courier packaging is mixed with municipal solid waste, among which 63% is landfilled, 30% is burnt and 7% is illegally disposed [6]. These methods can lead to a loss of resources and can pose a threat to human wellbeing [7]. University campuses are typical high-density consumption spaces where online shopping, food delivery, and parcel collection generate continuous packaging waste streams. In educational settings, the sorting system can also serve as an authentic learning infrastructure: students can observe material flows, interpret model outputs, and connect recycling decisions with environmental citizenship, data literacy, and systems thinking. These educational uses are proposed applications rather than outcomes measured in the present study.

The existing processes for recycling courier package waste are mainly based on manual sorting. These materials will require secondary sorting, cleansing and reprocessing after amalgamation with domestic refuse before they can be reused [8]. This traditional recycling approach is time-consuming and impractical and also generates additional carbon emissions from shipping and processing [9]. More recent research has involved the application of traditional machine vision techniques in recognition to facilitate the management of solid waste, with initial automated sorting being achieved using picture feature extraction and pattern matching [10]. However, these approaches still face great limitations in terms of low feature representation efficiency, poor adaptability to complex backgrounds, and low cross-material recognition accuracy [11]. Most of the existing studies focus on the classification and identification of common household waste (e.g., kitchen waste, metal, glass and plastic waste), but few systematic studies have been conducted on courier package waste, which is a waste type with high diversity [12]. Courier packing materials vary widely in the materials used (paper, plastic, foam, etc.), the shape of the material (boxes, bags, sleeves, etc.), and stacking and occlusion. These variables greatly reduce the effectiveness of existing general-purpose detection models and often result in lower recognition accuracy, poor classification granularity, and higher false detection rates in challenging situations. Thus, attaining effective and precise recognition and categorization of courier packaging trash continues to pose a substantial technical problem. In educational settings, improving packaging waste sorting also supports campus sustainability practices and strengthens the operational basis for environmental education. From the perspective of educational sustainability, these limitations also reduce the value of intelligent sorting systems as campus learning infrastructures. A system that cannot reliably distinguish materials, visualize object boundaries, or explain sorting outcomes provides limited support for student observation and inquiry. Accurate object-level recognition and pixel-level material delineation therefore have a dual role: they enable automated recycling operations and provide interpretable visual information that can be reused in environmental education, project-based learning, and AI-literacy activities.

With the rapid development of Industry 4.0 and Intelligent Manufacturing, it is very important to detect objects in an image and segment them using Computer Vision technologies, especially deep learning technologies. They have the capability to learn extensive features, which helps them to recognize objects and locate their positions in complex backgrounds, providing new ideas for intelligent waste sorting. YOLO convolutional neural network architectures are widely used in solid waste identification systems because of their real-time performance and high detection accuracy [13], while detection–segmentation fusion and boundary-aware models can provide finer object contours and structural information [14,15]. However, there are inherent limitations of individual ap-

proaches to detection and segmentation for express packaging waste. Detection models often have trouble correctly defining boundaries between objects that are touching, while segmentation methods often have problems with mis-segmentation in complex stacking situations. In this paper, a Detection–Segmentation Dual-Driven Network to recognize express packaging waste is proposed. The proposed framework ensures detection efficiency and simultaneously achieves higher boundary recognition accuracy and completeness in cases where an object is occluded or overlapping; it also deals with material diversity by using feature sharing and cross-layer fusion. This study has the following major findings:

(1) EPWaste was organized as a task-oriented benchmark for object-to-pixel packaging waste analysis. It contains 4528 RGB images, 11,384 annotated instances, and ten inner- and outer-packaging categories. Bounding boxes and complete-image semantic masks support controlled evaluation under deformation, reflection, stacking, occlusion, and complex backgrounds.
(2) PSSM-YOLO is presented as a task-oriented integration of established attention, lightweight convolution, multi-scale fusion, and refined head components. Its value is evaluated through unified comparisons and ablations for ten-category packaging waste detection rather than through a claim that the individual components are newly invented.
(3) CAA-DeepLabv3+ adapts channel attention aggregation and CARAFE content-aware upsampling to complete-image semantic segmentation of packaging waste. The model targets weak, irregular, reflective, and partially occluded boundaries; no detector-derived or ground-truth regions of interest are used as segmentation inputs.
(4) A laboratory-scale edge AI sorting prototype was evaluated to connect visual recognition with ten-category output and actuator control. Complexity, accuracy, latency, memory, and power are reported under a stated Jetson protocol, and the practical claims are limited to the tested laboratory conditions.

2. Related Work

2.1. Object Detection Algorithms in Waste Detection

In the last few years, the detection of waste and refuse has been accompanied by algorithms based on deep learning. Among them, two-stage detectors, such as R-CNN [16], Faster R-CNN [17] and Mask R-CNN (instance segmentation) [18], achieve very high detection accuracy. However, they have unique region proposal and classification methods, which means they are very complex and cannot achieve good real-time performance when processing a large amount of small-sized targets.

On the other hand, one-stage methods such as SSD [19] and YOLO [20] directly predict the box locations and labels of items, which has unique advantages for real-time detection. Various studies have improved YOLO-based models for different applications, achieving better accuracy. Ma et al. [21] developed DSYOLO-trash by integrating attention mechanisms and object tracking for real-time solid-waste detection. To quickly recognize multiple categories of solid waste in a multi-label dataset, Zhang et al. [22] proposed a YOLO-WASTE model developed using a transfer learning approach. To reduce the parameters of the model, Chen et al. [23] reconstructed YOLOv5 using ShuffleNetv2 and depthwise separable convolutions, providing new potential for mobile and embedded applications.

Despite these advances, existing detection technologies still have some challenges in dealing with express package waste. Because of the multi-material composition, high deformation, semi-transparency, and occlusion of this packaging type, resistance arises in the accurate identification of objects during boundary extraction and the localization accuracy of the pure object identification model. In educational recycling environments, object detection also provides a direct and interpretable representation of waste categories

and spatial locations. Such outputs can support campus recycling demonstrations and project-based learning by allowing students to observe how intelligent systems identify different packaging materials and make sorting decisions. Therefore, accurate object-level recognition is relevant not only to automated sorting efficiency but also to the development of visible and interactive recycling infrastructures for sustainability education.

2.2. Semantic Segmentation Algorithms in Waste Detection

Semantic segmentation can classify each pixel in an image, precisely delineate the target regions, and give fine-grained information concerning shape, boundaries, and inner structures. Some of the well-known architectures are FCN [24], U-Net [25], SegNet [26], PSPNet [27], and DeepLab [28]. These methods have succeeded in performing well in complex visual backgrounds by using multi-layer convolutional feature extraction and upsampling/decoder operations.

Lu et al. [29] built a high-quality dataset of 5366 construction waste images and optimized an improved DeepLabv3+ model through orthogonal experiments to efficiently segment nine kinds of construction waste. Yong et al. [30] fused remote sensing images with DeepLabv3+ to detect large-scale construction waste landfill sites, offering a new technical tool for environmental supervision. Li et al. [31] also proved the usefulness of RGB-D fusion in construction and demolition waste recognition.

These studies have yielded good results in their respective application scenarios, but waste-oriented intelligent-vision studies remain concentrated on construction and demolition waste, including complex-scene detection and segmentation [32]. There is still a lack of specialized research on express packaging waste. The significance of this gap is that express packaging has a larger variety of morphologies, materials, illumination, stacking, transparency, and deformation, and it is difficult to recover.

Accordingly, this study evaluates a coordinated object-to-pixel framework with two complementary outputs. PSSM-YOLO localizes and classifies packaging objects, whereas CAA-DeepLabv3+ predicts ten-category semantic masks directly from the complete RGB image. The outputs can be associated during system postprocessing for positioning and control, but detection boxes are not used to crop or constrain the segmentation input. From an educational perspective, pixel-level segmentation provides more detailed visual information than category labels alone because it explicitly shows the shape, boundary, and spatial distribution of each waste material. These visual outputs can be used to support the interpretation of material composition, recycling processes, and AI-based decision making in campus sustainability activities. Accordingly, semantic segmentation can serve both operational sorting requirements and the visual communication needs of educational recycling ecosystems.

3. Method

3.1. Experimental Design

The experimental workflow contains complementary detection and segmentation branches (Figure 1; Algorithm 1). PSSM-YOLO predicts bounding boxes, confidence scores, and ten-category labels from a complete RGB image. Independently, CAA-DeepLabv3+ receives the same complete image and predicts a ten-category pixel mask. Detection and segmentation are trained and evaluated on their corresponding annotations under the same image partitions. During prototype operation, their outputs are associated by category and spatial overlap for coordinate calculation and actuator scheduling. This formulation preserves complete-image context; neither ground-truth ROIs nor detector-derived ROIs are used as segmentation inputs. For potential educational deployment, the same workflow can support project-based activities in which learners compare detection boxes with pixel

masks, examine failure cases, and discuss the trade-offs between recognition accuracy, energy use, and responsible automation. No learner intervention or learning outcome was evaluated in this study.

Algorithm 1. Complete-image object-to-pixel inference and sorting workflow.

Input: complete RGB image I; trained detector D; trained segmenter S
1. Re-size and normalize I using the fixed evaluation preprocessing.
2. Compute object detections B = D(I): boxes, classes, and confidence scores.
3. Compute semantic mask M = S(I) on the complete image; no ROI crop is applied.
4. Associate B and M by predicted class and spatial overlap; estimate mask centroids.
5. Transform image coordinates to conveyor coordinates and issue one of ten category outputs.
Output: detections B, complete-image mask M, coordinates, and sorting commands.

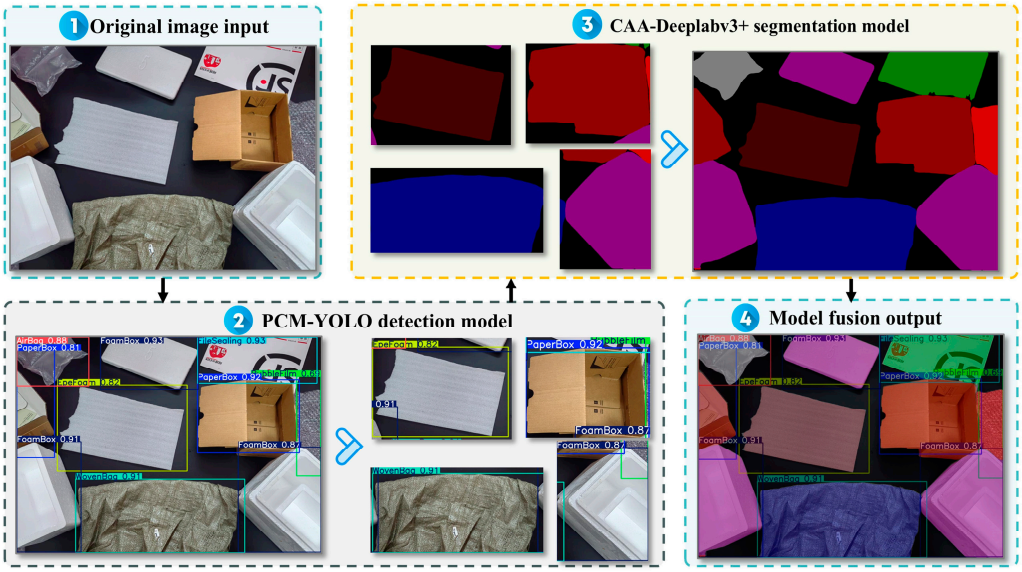


Figure 1. Research design.

3.2. Dataset Construction

EPWaste supports object detection and semantic segmentation of express packaging waste (Figure 2). It contains 4528 RGB images and provides two annotation subsets for different tasks. EPWaste-Detection contains bounding-box labels, whereas EPWaste-Segmentation contains complete-image semantic masks. The two tasks use the same category definitions and image partitions.

The dataset contains 11,384 annotated object instances in ten categories: PaperBox (1736), FileSealing (1024), PlasticBag (1295), WovenBag (812), FoamBox (1584), GasColumnBag (975), AirBag (1098), EPEFoam (742), BubbleFilm (1670), and IceBag (448). PaperBox, FileSealing, PlasticBag, WovenBag, and FoamBox represent outer packaging; GasColumnBag, AirBag, EPEFoam, BubbleFilm, and IceBag represent inner protective packaging (Figure 2). These counts reveal a naturally imbalanced distribution, with IceBag as the least represented class.

The 4528 images were divided into 3170 training, 453 validation, and 905 test images. Samples from the same acquisition sequence or stacking layout were assigned to the same subset to reduce near-duplicate leakage, and category-aware stratification was applied to preserve representation of less frequent classes. Bounding boxes were checked for class and spatial consistency. Pixel masks were traced along visible object boundaries on the complete images and reviewed in a second quality control pass; ambiguous boundary regions were rechecked by an annotator before release into the experimental split.

Images were acquired with a Canon EOS 600D using multiple viewpoints and camera heights under indoor illumination and conveyor simulation conditions. The scenes included stacking, occlusion, deformation, reflective surfaces, and heterogeneous backgrounds. Because the associated project and final quality control of the pixel annotations are still ongoing, the enhanced dataset is not yet deposited in a public repository. The data and annotation protocol are available from the corresponding author upon reasonable request, subject to project restrictions.

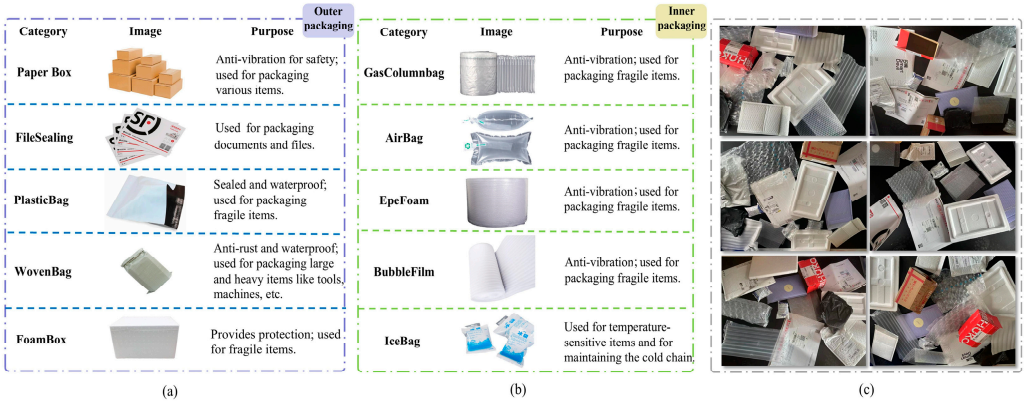


Figure 2. The EPWaste datasets: (**a**) express outer packaging; (**b**) express inner packaging; (**c**) sample images.

3.3. The PSSM-YOLO Model

Figure 3 summarizes the adapted detector components. PSSM-YOLO is a task-oriented detector assembled to address small objects, background interference, morphological diversity, and latency constraints in packaging waste scenes. Multi-head self-attention (MHSA) is placed after the SPPF layer to capture long-range contextual dependencies. The module complements convolutional features when objects are partially occluded or visually similar. Figure 3 summarizes the adapted components and their positions in the detector.

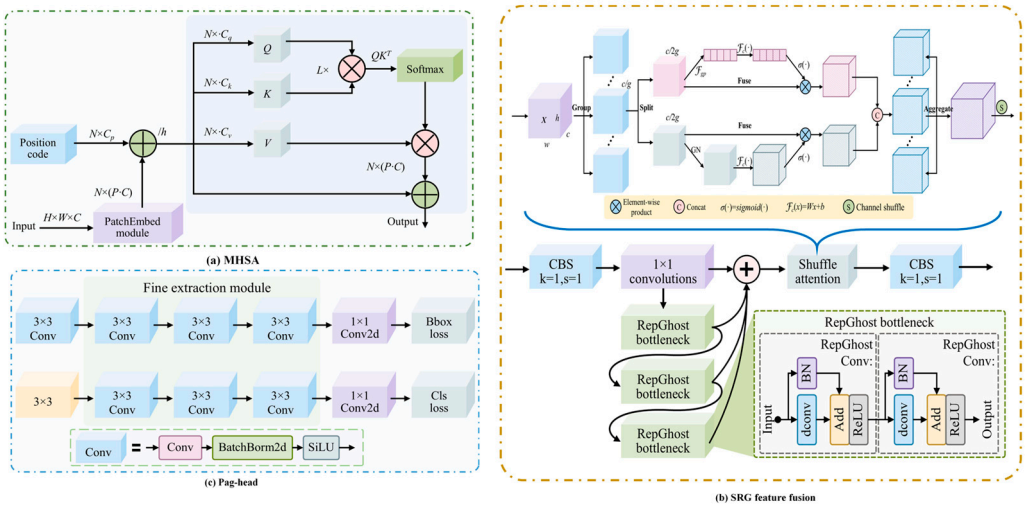


Figure 3. Schematic illustration of improvements provided by the PSSM-YOLO model: (**a**) the MHSA; (**b**) the SRG feature fusion module; (**c**) the Pag-head module.

The SRG feature fusion module combines Ghost convolution, re-parameterization, and multi-scale aggregation to reduce redundant computation while retaining features at different resolutions. Shuffle Attention recalibrates and exchanges channel information after fusion. The Pag-head adds progressive convolutional refinement in the classification

and regression branches. These components are established mechanisms adapted and jointly evaluated for the present ten-category task; the contribution lies in the integration, task-specific evaluation, and accuracy–efficiency analysis.

3.3.1. The Multi-Head Self-Attention Module

In express packaging waste detection, the background setting is often complex, different types of waste are mixed, and many objects are small, which often prevents traditional detection methods from achieving a 100% detection rate and leads to false positives. To enhance the accuracy and robustness of express packaging waste identification in challenging environments, the authors introduce a multi-head self-attention (MHSA) module after the SPPF layer in this study. The linkage between the backbone and neck is very important for feature extraction and fusion, as it directly affects the ability of the model to process multi-scale features. To enable global context modeling and capture long-range dependencies, the MHSA module is introduced to improve the representational ability of this link. The Patch Embedding (PatchEmbed) procedure consists of dividing the two-dimensional feature map (e.g., 80 × 80 resolution) into 2 × 2 patches to get a sequence of length $N = HW / P^2$. The positional encoding is used to preserve the spatial relationships between patches. The three matrices, namely, question (Q), key (K), and value (V), are then constructed and fed to eight parallel attention heads, each with a channel dimension of 256. This can be properly stated as:

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{C_k}}\right)V \quad (1)$$

where Q, K, and V are the query, key, and value matrices, respectively, which are computed using linear transformation of the input feature representations. *Softmax* normalization is applied to the similarity scores to obtain the corresponding attention weights, and K^T is the key matrix transposed to calculate the similarity between the query and key vectors. C_k is the dimensionality of each key vector; that is, the number of feature components in each vector of the key matrix.

3.3.2. The SRG Feature Fusion Module

The SRG feature fusion module replaces selected conventional C2f blocks to combine lightweight representation and multi-scale feature aggregation. Its RepGhost blocks use Ghost convolution and re-parameterization to generate additional feature maps with inexpensive operations and to fuse training-time branches for inference. Shuffle Attention is then used to exchange information between channel groups and recalibrate the fused responses. The RepGhost operation is expressed as follows:

$$F_{rg}(x) = ReParam(GhostConv_1(x), GhostConv_2(x)) \quad (2)$$

where *GhostConv* is a ghost convolution operation, *ReParam* is a re-parameterization fusion process, x is the input feature map, and $F_{rg}(x)$ is the feature output after the RepGhost block.

The SRG module contains a RepGhost Bottleneck submodule that allows the merging of feature data on multiple scales. RepGhost adds attributes captured at different levels of resolution as shortcut links, thus strengthening the model's ability to identify packaging waste for express deliveries in complex background scenarios. Concatenation and fusion can be precisely outlined as:

$$F_{fusion}(x_s, x_l) = Concat\left(F_{rg}(x_s), F_{rg}(x_l)\right) \quad (3)$$

where (x_s and x_l) are the feature maps at different scales and (*Concat*) is the feature concatenation operation.

To boost the model's sensitivity to key features after incorporating multi-scale feature information, the Shuffle Attention (*SA*) technique is proposed. *SA* facilitates the interaction between channels by rearranging the channels, which increases the discriminative capacity and representation depth of the retrieved features. The working of *SA* can be written as:

$$F_{SA}(x) = SA\left(F_{fusion}(x_s, x_l)\right) \tag{4}$$

Use of the *SA* mechanism allows the network to adaptively adjust the weight of feature channels, making it possible to focus on more discriminative features and optimize the representation of features and the classification performance. The pseudo-code for the feature fusion process of the SRG Feature Fusion module for the complete feature is provided below.

3.3.3. The Pag-Head Model

Packaging trash is highly polymorphous, multi-sized and includes a wide range of packaging materials, as well as taxonomic characteristics of packaging kinds. This heterogeneity makes it difficult for a conventional detection head to extract adequately discriminative characteristics in a complex logistics context, which results in low item recognition and localization accuracy.

The Pag-head adapts a progressively refined detection head structure using additional convolutional transformations and cross-layer feature use. Its purpose in this study is to increase the representation capacity of the classification and regression branches for heterogeneous packaging textures and shapes.

Two additional 3×3 convolutional layers are applied in the classification and regression branches relative to the baseline head. The deeper task-specific transformations are intended to refine localization and class evidence before the final predictions. The ablation study in Section 5.2.1 evaluates the effect of this adapted head separately from SRG, MHSA, and Shuffle Attention.

$$cv(x) = Concat(Conv_{1\times1}^{C_2\to 4\cdot reg_{max}}((\prod_{i=1}^{4} Conv_{3\times3}^{C_2\to C_2})(x)) \tag{5}$$

$$Conv_{1\times1}^{C_2\to n_c}((\prod_{i=1}^{4} Conv_{3\times3}^{C_3\to C_3})(x)) \tag{6}$$

where $\prod_{i=1}^{4} Conv_{3\times3}^{C_2\to C_2}$ denotes four consecutive 3×3 convolutional layers stacked sequentially to progressively refine and enhance the feature representations. The extracted features are projected into task-specific dimensions, using $Conv_{1\times1}^{C_2\to}$, where the output channel number of the regression branch is $4{\cdot}reg_{max}$ and the output channel number of the classification branch is n_c (the number of categories). Finally, the outputs of both branches are concatenated together along the channel dimension by Concat, and the output is the comprehensive detection output. This structural design significantly enhances the capacity of the model to capture fine-grained properties of express packaging waste in complex backgrounds and the discriminative power of feature representations by means of deeper nonlinear transformations. Therefore, the Pag-head architecture greatly improves localization and classification accuracy, providing a feasible and stable solution to waste identification in complex logistics scenes.

3.4. The CAA-DeepLabv3+ Model

CAA-DeepLabv3+ uses CARAFE in the decoder of the DeepLabv3+ encoder–decoder architecture for complete-image packaging waste segmentation. CARAFE predicts content-aware reassembly kernels to recover spatial detail. Channel attention aggregation (CAA) recalibrates channel responses to strengthen object-related features and suppress background interference. Their combination targets weak boundaries, folds, reflections, de-

formation, and partial occlusion. Figure 4 shows the locations of the two adapted modules. CAA-DeepLabv3+ always receives the complete RGB image; no detector-derived or ground-truth ROI is used.

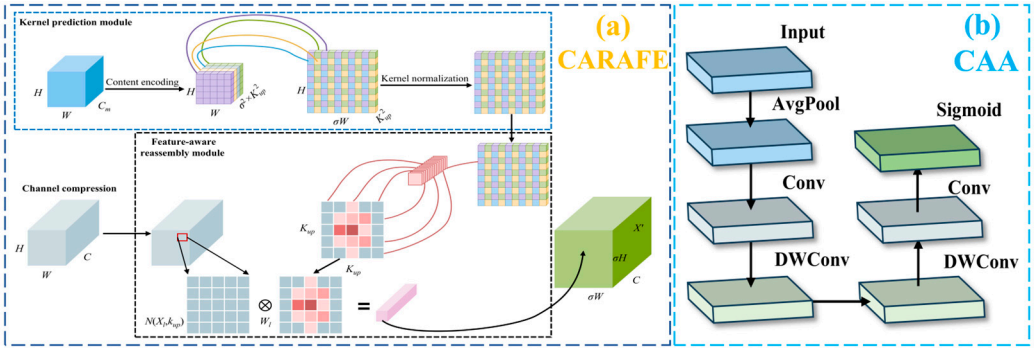


Figure 4. Adapted components of CAA-DeepLabv3+: (**a**) CARAFE content-aware upsampling; (**b**) channel attention aggregation (CAA).

3.4.1. The CARAFE Upsampling Module

When applied to real-world express package waste detection tasks, either the objects themselves or their complete visibility is often hampered by partial occlusion, distortion, and reflected surfaces, creating complex visual situations. Essential traits in these situations are frequently not well detected by conventional detection techniques, leading to reduced identification accuracy. In this paper, a better framework is proposed by combining DeepLabv3+ with the CARAFE upsampling operator to get the most reassembly of features. This enables the model to better recover and utilize locally available contextual information to tackle packaging waste with partially occluded and diverse morphologies. In Figure 4a, it can be seen that the CARAFE operator can be basically divided into two parts: an upsampling kernel prediction module and a feature reassembly module. The basic idea is to dynamically predict upsampling kernels to guide the content-aware reconstruction of features. Given an input feature map x with dimensions $C \times H \times W$, the operator generates an output feature map x' of dimensions $C \times \sigma H \times \sigma W$. For any target position $l' = (i', j')$ corresponding to the input position $l = (i, j)$, the spatial relation satisfies $i = i'/\sigma$ and $j = j'/\sigma$, where σ denotes the upsampling ratio.

The upsampling kernel prediction module consists of three parts: channel compression, content encoding and kernel prediction, and upsampling normalization. To reduce the computational burden, the input feature map x is first fed into a 1×1 convolution layer to reduce the number of channels to C_m, so that the model maintains high computational efficiency even in complex scene settings. Since express packaging waste often shows the phenomenon of detail degradation in a local area caused by occlusion and deformation, the convolution kernel with a size of $K_{encoder} \times K_{encoder}$ is used in the compressed feature map for content encoding and upsampling kernel prediction. This approach allows the network to learn the correlational structure between neighboring regions and create adaptive upsampling kernels to better preserve the structure of the reconstructed features. The shaped expected upsampling kernel is $K_{up}^2 \times \sigma H \times \sigma W$, where K_{up} is the kernel size for upsampling, σH is the kernel size for height and σW is the kernel size for width. This operation will effectively increase the local receptive field; thus, the model will be able to capture intricate features and extract discriminative information even if there is local occlusion or blurring.

In order to make the predicted upsampling kernels for each target position follow a valid probability distribution, a *Softmax* function is applied for normalization. For each tar-

get position l', the corresponding upsampling kernel $w_{(l')}$ is predicted by a function φ based on the local neighborhood $N(x_l, k_{encoder})$ of the input feature map, as formulated below:

$$w_{(l')} = \varphi(N(x_l, k_{encoder})) \tag{7}$$

In order to solve the problem of feature recovery of express packaging waste in the case of occlusion or partial feature information loss, the feature reassembly module is used to make place-to-place correlations between the features of x' in all places and the corresponding positions of the features of x. A local neighborhood of size $K_{up} \times K_{up}$ is first extracted around each target point, and then a dot product is performed between this local neighborhood and the upsampling kernel projected on the target point to generate the output feature. The reassembly procedure can be described as:

$$x'_{(l')} = \varphi(N(x_l, k_{up}), w_{(l')}) \tag{8}$$

where φ is the content-aware feature reassembly function, including local neighborhood $N(x_l, k_{up})$ and the predicted upsampling kernel $w_{(l')}$ to reconstruct fine-grained spatial details. The multi-scale feature fusion strategy is specially designed in this module, which is very effective at alleviating the problem of incomplete expression of packaging waste when occlusion and deformation occur.

3.4.2. The CAA Attention Module

In real-life scenarios, packaging waste is often piled up and/or partially covered in a manner that makes it hard to identify different waste types from purely on-site visual cues. The channel attention aggregation (CAA) paradigm has the ability to collect long-range contextual relationships, which can dramatically improve the representation of waste objects.

First, average pooling (AvgPool) and ordinary convolution (*Conv*) methods are used to obtain preliminary local information, thus eliminating the effect of background noise and redundant information. This can be expressed as the following formulation:

$$F^{INPUT} = Conv(P_{avg}(input)) \tag{9}$$

where $INPUT$ is the input image or feature map, $P_{avg}(input)$ is the feature representation generated by the average pooling (AvgPool) operation on the input image, *Conv* is the regular convolution operation, and F^{INPUT} is the feature representation after the convolution operation.

In turn, depthwise separable convolution (*DWConv*) is used to effectively model long-range contextual relationships to counteract the strong local blurring of features caused by occlusion between waste objects. The formulation is as follows:

$$F^w = DWConv\left(F^{INPUT}\right) \tag{10}$$

$$F^h = DWConv(F^w) \tag{11}$$

where *DWConv* is the depthwise separable convolution operation, F^w is the feature map after the first *DWConv*, and F^h is the feature map after the second *DWConv* (which further incorporates long-range contextual information).

The resulting attention weight map (after *Sigmoid* activation of the feature map) is used to adaptively improve the feature expression of the target region and give full play to the idea of information focus, highlighting the packaging waste object that may be covered

or hidden in a complex background, thereby greatly improving the segmentation accuracy of the model. The exact composition is as follows:

$$A = Sigmoid(Conv(F^h)) \tag{12}$$

where A denotes the final generated attention weight map and $Sigmoid$ represents the activation function, which normalizes the weights within the range of [0, 1] to enhance the feature representation of target regions.

4. Experimental Environment and Evaluation Metrics

Table 1 summarizes the computing environment and common settings. Within each task, all compared models used the same EPWaste train/validation/test partitions, 640 × 640 input resolution, pretrained initialization policy, augmentation policy, training budget, early stopping rule, confidence threshold, non-maximum suppression threshold, and evaluation code. The detection models were optimized with SGD using an initial learning rate of 0.001 and cosine decay; the segmentation models were optimized with AdamW using the same partitions and task-specific loss. The results are from controlled single runs with random seed 0. No claim of statistical significance or run-to-run stability is made; repeated-run evaluation remains a limitation.

Table 1. Experimental environment and parameters.

Hardware environment	CPU	15 vCPU AMD EPYC 7642 48-Core Processor
	RAM	32 GB
	Video memory	24 GB
Software environment	Operating system	Ubuntu 20.04.4 LTS
	GPU	NVIDIA GeForce RTX 3090
	Python	3.8.10
	Pytorch	Pytorch 2.0.1
	CUDA	12
	cuDNN	8.7
Parameters	Size of input images	640 × 640
	Learning rate	0.001
	Epochs	100
	Batch size	16
	Decay	0.0005

In this work, Intersection over Union (IoU) is used as the main metric to quantify the effectiveness of the proposed model in expressing packaging waste detection and segmentation. IoU is the overlap between the predicted region and the ground-truth annotation, and IoU can be defined as follows:

$$IoU = \frac{Area(predicted\ region\ \cap\ ground\ truth\ region)}{Area(predicted\ region\ \cup\ ground\ truth\ region)} \tag{13}$$

The basic idea of IoU is the same for object detection and semantic segmentation, but the way it is implemented is very different. For object detection, IoU is the measure of the spatial intersection of two boxes, the predicted one and the actual box in the ground truth, and it is the key to calculating Mean Average Precision (mAP). To measure the detection efficacy, mAP is used, defined as the area under the precision–recall (PR) curve described below.

$$mAP = \frac{1}{C}\sum_{c=1}^{c} AP_c,\ AP_c = \int_0^1 P_c(R_c)dR_c \tag{14}$$

where C denotes the number of classes and AP_c is the average precision for class c. Detection performance is reported using precision, recall, mAP50 at IoU = 0.50, and mAP50:95 averaged over IoU thresholds from 0.50 to 0.95 in increments of 0.05. The corresponding expressions for precision and recall are defined as follows:

$$Precision = \frac{TP}{TP + FP}, \; Recall = \frac{TP}{TP + FN} \tag{15}$$

where TP, FP, and FN are the number of true positive, false positive, and false negative results, respectively.

IoU measures how accurately the pixels are classified in semantic segmentation by calculating the spatial overlap between the predicted and ground-truth regions. For this purpose, comprehensive measurements ($mIoU$, mPA, $mPrecision$ and $mRecall$) are used to quantify overall segmentation performance, which is defined as follows:

$$mIoU = \frac{1}{N}\sum_{i=1}^{N} \frac{TP_i}{TP_i + FP_i + FN_i}, \; mPA = \frac{1}{N}\sum_{i=1}^{N} \frac{TP_i}{TP_i + FN_i} \tag{16}$$

$$mPrecision = \frac{1}{N}\sum_{i=1}^{N} \frac{TP_i}{TP_i + FP_i}, \; mRecall = \frac{1}{N}\sum_{i=1}^{N} \frac{TP_i}{TP_i + FN_i} \tag{17}$$

The first corresponds to the number of classes, the second to the number of pixels correctly classified in each class, the third to the number of pixels misclassified in each class and the last to the number of pixels to be classified that were missed or misclassified by the model.

5. Experimental Results and Analysis

5.1. Comparative Experiments

5.1.1. Comparative Experiments with the PSSM-YOLO Model

PSSM-YOLO was compared with Faster R-CNN, SSD, and YOLO variants from v3s to v11s under the unified protocol described in Section 4. As shown in Table 2, PSSM-YOLO obtained 87.2% mAP50, 74.1% mAP50:95, 93.6% precision, and 85.8% recall. Relative to YOLOv8s in this controlled single run, the changes were +3.4 percentage points in mAP50, +0.7 points in mAP50:95, −0.2 points in precision, and +2.7 points in recall. The result indicates a favorable balance for this dataset, without implying statistical significance across repeated training runs.

Table 2. Values of the *mAP* and recall rate of different models on the EPWaste dataset.

Method	Backbone	mAP50 (%)	mAP50:95 (%)	Precision (%)	Recall (%)
Faster-RCNN	ResNet	70.1	57.2	81.4	66.4
SSD	VGG16	72.2	59.7	84.8	69.5
YOLOv3s	Darknet-53	83.8	71.9	95.4	81.6
YOLOv5s	CSPDarknet	83.7	72.3	96.9	80.1
YOLOv6s	CSPDarknet	83.4	72.6	90.1	82.7
YOLOv7	CSPDarknet	82.8	71.7	91.0	79.5
YOLOv8s	CSPDarknet	83.8	73.4	93.8	83.1
YOLOv9s	-	84.0	73.4	94.9	81.6
YOLOv10s	-	83.1	73.3	96.3	77.6
YOLOv11s	-	83.7	73.1	96.4	81.6
Ours (PSSM-YOLO)	-	87.2	74.1	93.6	85.8

The models were thoroughly evaluated under various behavioral conditions, revealing the limitations of traditional YOLO variants (v3–v7) in complex visual scenarios. YOLOv3s and YOLOv5s demonstrated low recognition ability for clear or reflective materials (like bubble wrap and plastic bags) and a high miss-detection rate for smaller fragments of trash objects. Despite the enhanced overall accuracy of YOLOv8s and YOLOv9s, they

often generated superfluous bounding boxes in high-density stacking situations, indicating shortcomings in spatial feature filtering.

PSSM-YOLO has many modules, which synergistically make the performance of PSSM-YOLO excellent. The global self-attention mechanism of the MHSA module enhances the learning of complex features of translucent and small-scale objects. The Pag-head structure improves spatial localization awareness and significantly improves robustness in environments with densely stacked debris, blockage, etc. Furthermore, even in cases where edges are blurred and partly covered and the packing cases are not standard, the visual results in Figure 5 confirm that PSSM-YOLO achieves better localization accuracy and fewer missed detections.

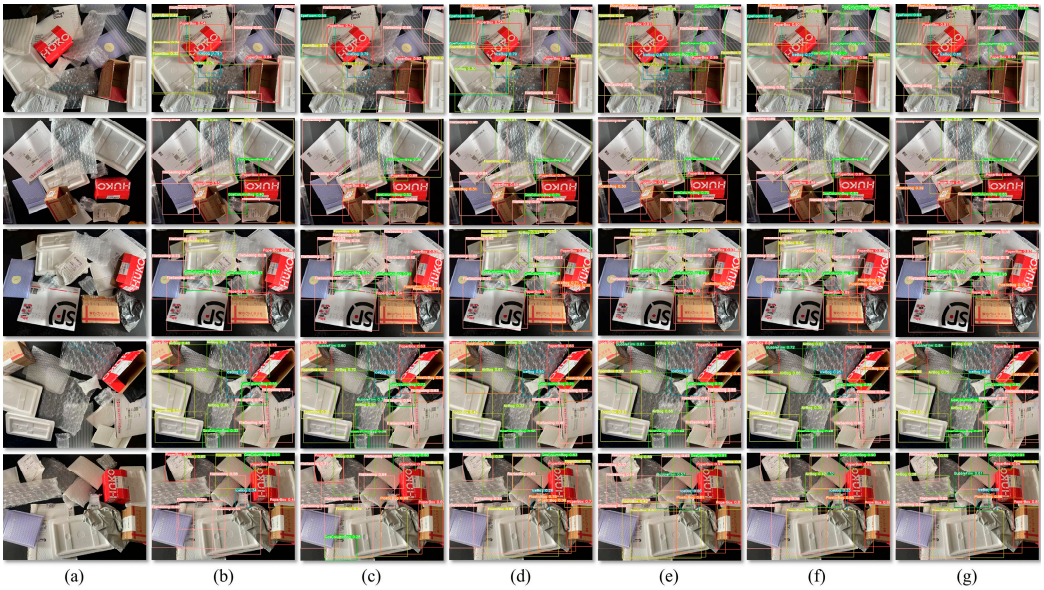


Figure 5. Object detection performance comparison. Visualization results from left to right: (**a**) original image; (**b**) YOLOv3s; (**c**) YOLOv5s; (**d**) YOLOv8s; (**e**) YOLOv9s; (**f**) YOLOv11s; (**g**) PSSM-YOLO.

5.1.2. Comparative Experiments with the CAA-DeepLabv3+ Model

CAA-DeepLabv3+ combines channel attention aggregation with CARAFE content-aware upsampling in a MobileNetV2-based DeepLabv3+ architecture. All segmentation models receive complete RGB images and are evaluated using the same split and preprocessing. The comparison therefore measures full-image semantic segmentation rather than segmentation of ground-truth or detector-derived crops.

Table 3 compares CAA-DeepLabv3+ with FCN, PSPNet, U-Net, and the DeepLabv3+ baseline. CAA-DeepLabv3+ obtained 82.4% *mIoU*, 83.6% *mPA*, 83.3% *mPrecision*, and 83.6% *mRecall*. In this controlled single run, its *mIoU* exceeded FCN, PSPNet, U-Net, and DeepLabv3+ by 7.4, 3.6, 1.9, and 1.7 percentage points, respectively. These results support the selected adaptation on EPWaste, while repeated runs and cross-scene tests are needed to quantify stability and generalization.

The following mechanisms can be attributed to the remarkable performance of CAA-DeepLabv3+. The interdependence among channels in the CAA module makes it possible to extract distinctive features and suppress redundant features in the image. The CARAFE operator, which is based on adaptive kernel prediction, can preserve the complicated structure and boundary information in the process of upsampling and is especially effective in reconstructing very small structures, such as gaps between parcels and creases in plastic films. At the same time, the application of a lightweight backbone ensures the model's real application in the sorting system. In the complex operation environment, CAA-DeepLabv3+ has excellent pixel-level measurement performance and good segmen-

tation ability, providing a stable technical solution for the high-accuracy automated sorting of express packaging waste. The class-level results show that FoamBox and PaperBox have the highest *IoU*, whereas IceBag is the most difficult class. IceBag is also the least represented category and often occupies small, deformable regions with weak boundaries. The mean Boundary F-score of 78.6% complements *mIoU* by measuring contour agreement. All values in Table 4 were computed from complete-image predictions.

Table 3. Complete-image semantic segmentation results on the EPWaste test set.

Dataset	Method	Backbone	*mIoU* (%)	*mPA* (%)	*mPrecision* (%)	*mRecall* (%)
EPWaste	FCN	VGG16	75	79.2	79.8	79.2
	PSPNet	ResNet	78.8	81.5	81.6	81.5
	UNet	ResNet	80.5	82.6	82.4	82.6
	DeepLabv3+	MobileNetv2	80.7	82.8	82.4	82.8
	CAA-DeepLabv3+ (Ours)	MobileNetv2	82.4	83.6	83.3	83.6

Table 4. Per-class complete-image segmentation performance of CAA-DeepLabv3+.

Category	*IoU* (%)	Boundary F-Score (%)
PaperBox	85.8	82.1
FileSealing	81.3	77.5
PlasticBag	82.7	78.9
WovenBag	80.5	76.8
FoamBox	86.6	83.4
GasColumnBag	83.2	79.2
AirBag	82.0	78.0
EPEFoam	83.7	80.5
BubbleFilm	84.4	80.8
IceBag	73.8	69.1
Mean	82.4	78.6

5.2. Ablation Experiments

5.2.1. Ablation Experiments with the PSSM-YOLO Model

Table 5 separates the effects of Pag-head, SRG, MHSA, and Shuffle Attention. The configuration before Shuffle Attention achieved 86.8% mAP50, 75.6% mAP50:95, 87.2% precision, and 85.4% recall. Adding Shuffle Attention changed these values to 87.2%, 74.1%, 93.6%, and 85.8%, respectively. Thus, the final model does not maximize every metric: it gains 0.4 points in mAP50, 6.4 points in precision, and 0.4 points in recall, while mAP50:95 decreases by 1.5 points. The final configuration was selected for its balance of detection confidence, recall, and deployment requirements. The MHSA and Shuffle Attention effects are therefore reported separately rather than described as uniformly complementary gains.

Table 5. Ablation results for PSSM-YOLO on the EPWaste test set.

Method	Pag-Head	SRG	MHSA	SA	mAP50 (%)	mAP50:95 (%)	Precision (%)	Recall (%)
YOLOv8s					83.8	73.4	93.8	83.1
YOLOv8s + Pag-head	Yes				83.8	73.4	96.3	80.7
YOLOv8s + Pag-head + SRG	Yes	Yes			87.1	74.1	93.6	85.7
YOLOv8s + Pag-head + SRG + MHSA	Yes	Yes	Yes		86.8	75.6	87.2	85.4
PSSM-YOLO	Yes	Yes	Yes	Yes	87.2	74.1	93.6	85.8

5.2.2. Ablation Experiments with the CAA-DeepLabv3+ Model

To examine the impact of the CAA channel attention and CARAFE upsampling modules on the segmentation efficacy of DeepLabv3+ for express packaging waste,

four sets of ablation experiments were performed on the EPWaste dataset: (A) the baseline DeepLabv3+, (A + B) incorporating CAA, (A + C) substituting the upsampling operator with CARAFE, and (A + B + C) combining both CAA and CARAFE, referred to as the proposed CAA-DeepLabv3+.

The baseline model A attained *mIoU*, *mPA*, *mPrecision*, and *mRecall* values of 80.7%, 82.8%, 82.4%, and 82.8%, respectively. Consequently, A + B elevated *mIoU* to 81.4%, *mPA* to 83.1%, *mPrecision* to 82.8%, and *mRecall* to 83.1%, demonstrating that the CAA module significantly augmented the network's depiction of package edges and intricate details via channel recalibration. Correspondingly, A + C attained *mIoU*, *mPA*, *mPrecision*, and *mRecall* scores of 81.3%, 83.0%, 82.7%, and 83.0%, respectively, substantiating the efficacy of CARAFE in reconstructing multi-scale feature details.

When both modules operated in conjunction, CAA-DeepLabv3+ attained optimal overall performance (Table 6), with *mIoU*, *mPA*, *mPrecision*, and *mRecall* reaching 82.4%, 83.6%, 83.3%, and 83.6%, respectively, indicating enhancements of 1.7, 0.8, 0.9, and 0.8 percentage points compared to the baseline. The results unequivocally illustrate the synergistic advantage of integrating CAA and CARAFE, which markedly improves model resilience in the presence of morphological diversity, blurred edges, and stacking occlusion. This synergy establishes a robust basis for high-precision automated sorting and recycling of express packaging trash in intricate real-world settings.

Table 6. Ablation results for CAA-DeepLabv3+ on the EPWaste test set.

Dataset	Number Method	A DeepLabv3+	B A + CAA	C A + CARAFE	A + B + C CAA-DeepLabv3+ (Ours)
EPWaste	*mIoU* (%)	80.7	81.4	81.3	82.4
	mPA (%)	82.8	83.1	83	83.6
	mPrecision (%)	82.4	82.8	82.7	83.3
	mRecall (%)	82.8	83.1	83	83.6

5.3. Hardware-Oriented Edge AI Deployment and System-Level Evaluation

To examine laboratory-scale feasibility, the two outputs were integrated into a conveyor prototype comprising image acquisition, edge inference, object positioning, conveyor control, sorting actuation, and monitoring units. The Jetson measurements reported below use JetPack 6.0, CUDA 12.2, TensorRT 8.6.2, 15 W power mode, fixed clocks, batch size 1, 100 warm-up iterations, and 500 timed iterations. Latency was measured with device synchronization; peak memory and board power were recorded using tegrastats.

The image acquisition unit consists of an MVS-CA050-10GM industrial camera and an E3Z-LS61 photoelectric sensor. The industrial camera provides 5-megapixel global shutter image acquisition, while the photoelectric sensor has a nominal response time of 1 ms. When an object enters the observation area, the sensor generates a triggering signal and the camera captures an image of the packaging object. The image is then transmitted to an NVIDIA Jetson Orin Nano edge computing device equipped with 8 GB of memory and a theoretical computing capacity of 40 TOPS.

For each triggered RGB frame, PSSM-YOLO predicts boxes and category labels, while CAA-DeepLabv3+ independently predicts a semantic mask from the complete frame. The segmentation input is neither a ground-truth crop nor a detector crop. During postprocessing, detections and mask components are associated by class and spatial overlap, after which mask centroids are transformed into conveyor coordinates. The controller then issues one of the ten EPWaste category outputs.

The conveyor control unit is implemented using a Siemens S7-1200 programmable logic controller with a DC/DC/DC configuration. The conveyor platform has a width

of 400 mm and supports an adjustable operating speed of 0–60 m/min. Servo motors or pneumatic cylinders are used as sorting actuators, and the system is powered by a 24 V/10 A direct current supply. The controller receives the category and position information generated by the Jetson device and activates the corresponding actuator when the object reaches the assigned sorting point.

5.3.1. Edge Inference Performance

PSSM-YOLO was benchmarked on the Jetson Orin Nano under FP32, FP16, INT8, and INT8 with structured pruning (Table 7 summarizes model complexity; Table 8 reports deployment accuracy and efficiency). Throughput increased from 23.4 FPS in FP32 to 40.7 FPS in FP16, 63.3 FPS in INT8, and 90.1 FPS after INT8 quantization and pruning. These figures describe detector inference only; they are not presented as latency for the combined detection–segmentation control pipeline.

Latency decreased from 42.8 ms in FP32 to 24.6 ms in FP16, 15.8 ms in INT8, and 11.1 ms after INT8 quantization and pruning. Relative to FP32, the final configuration reduced mAP50 by 1.3 percentage points and mAP50:95 by 1.5 points while reducing latency by 74.1% and peak memory from 1460 MB to 760 MB. This comparison makes the accuracy–efficiency trade-off explicit.

The Jetson results show that quantization and pruning can reduce the detector's resource demand. CAA-DeepLabv3+ complexity is reported in Table 8, but a separate TensorRT precision sweep for the segmenter was not conducted. Accordingly, the PSSM-YOLO latency values are not attributed to the complete two-model pipeline.

Table 7. Computational complexity of the proposed models at 640 × 640 input and batch size 1.

Model	Parameters	MACs	FLOPs	FP16 Size
PSSM-YOLO	12.8 M	16.3 G	32.6 G	25.8 MB
CAA-DeepLabv3+	6.7 M	9.9 G	19.8 G	13.5 MB

Table 8. PSSM-YOLO accuracy and efficiency on the NVIDIA Jetson Orin Nano.

Configuration	Precision	Pruning	mAP50 (%)	mAP50:95 (%)	Latency	FPS	Peak Memory
FP32	FP32	No	87.2	74.1	42.8 ms	23.4	1460 MB
FP16	FP16	No	87.0	73.9	24.6 ms	40.7	1180 MB
INT8	INT8	No	86.5	73.3	15.8 ms	63.3	910 MB
INT8 + pruning	INT8	Yes	85.9	72.6	11.1 ms	90.1	760 MB

5.3.2. Power Consumption

Board power was 15.2 W in FP32, 10.6 W in FP16, 7.6 W in INT8, and 5.1 W with INT8 quantization and structured pruning (Table 9). Power was sampled with tegrastats over the same timed inference interval used for latency measurement.

Relative to FP32, FP16 reduced power by 30.3%, INT8 by 50.0%, and INT8 with pruning by 66.4%. These savings accompany a modest loss in detection accuracy and should be interpreted as an accuracy–efficiency trade-off rather than an unqualified improvement.

Table 9. PSSM-YOLO board power under different Jetson deployment configurations.

Configuration	Power	Reduction vs. FP32
FP32	15.2 W	-
FP16	10.6 W	30.3%
INT8	7.6 W	50.0%
INT8 + pruning	5.1 W	66.4%

At the measured throughput of 58.3 objects per minute, the 5.1 W edge inference configuration corresponds to an estimated edge-device energy consumption of approximately 5.25 J per processed object. This value reflects the power demand of the edge inference unit only and does not include the energy consumed by the conveyor belt, controller, camera, or pneumatic actuator.

5.3.3. Sorting Accuracy

A total of 974 packaging objects were used in the laboratory-scale prototype test. All objects were assigned to the same ten categories used by EPWaste: PaperBox, FileSealing, PlasticBag, WovenBag, FoamBox, GasColumnBag, AirBag, EPEFoam, BubbleFilm, and IceBag. In aggregate, 938 objects were assigned to the correct category output, giving an overall accuracy of 96.30%.

Overall sorting accuracy was calculated as follows:

$$\text{Overall accuracy} = 938/974 \times 100\% = 96.30\%.$$

The prototype used ten logical category outputs. For the laboratory apparatus, these outputs were mapped to actuator commands and collection positions through the controller configuration shown in Table 10.

The aggregate result demonstrates feasibility under the tested indoor illumination, conveyor speed, object spacing, and camera geometry. It does not establish industrial performance or class-balanced accuracy across uncontrolled environments.

Verified per-class prototype counts were not available; therefore, class-level precision, recall, and F1-score are not reported for the physical sorting test. The aggregate result is interpreted together with the ten-category output taxonomy rather than as evidence of class-balanced performance.

Future system tests will record a ten-by-ten confusion matrix and class-level throughput and error rates under varying speeds, illumination, contamination, and object density.

Table 10. Ten-category output taxonomy used by the laboratory prototype.

Group	EPWaste Category	Prototype Output
Outer packaging	PaperBox	PaperBox
Outer packaging	FileSealing	FileSealing
Outer packaging	PlasticBag	PlasticBag
Outer packaging	WovenBag	WovenBag
Outer packaging	FoamBox	FoamBox
Inner packaging	GasColumnBag	GasColumnBag
Inner packaging	AirBag	AirBag
Inner packaging	EPEFoam	EPEFoam
Inner packaging	BubbleFilm	BubbleFilm
Inner packaging	IceBag	IceBag

The 36 aggregate errors were associated qualitatively with surface reflection, folding, partial occlusion, deformation, and overlap. A verified class-level error allocation was not available, so no numerical category attribution is made.

5.3.4. End-to-End System Performance

The complete system achieved an average throughput of 58.3 objects per minute (Table 11), corresponding to an average interval of approximately 1.03 s per object. End-to-end latency was 312 ms from photoelectric triggering to actuator execution and included image acquisition, transfer, object detection, complete-image segmentation, coordinate calculation, controller communication, conveyor synchronization, and actuator response.

The end-to-end latency is higher than the inference latency of the Jetson device because the complete sorting cycle contains several hardware and communication processes in addition to neural network inference. Nevertheless, the measured latency remained below the average object-processing interval, allowing the system to complete visual recognition and actuator scheduling before the next object reached the sorting position.

The prototype completed an eight-hour laboratory run without a recorded software interruption or system crash. This observation is limited to the tested configuration and duration and is reported as preliminary operational evidence rather than proof of long-term industrial reliability.

Table 11. Aggregate end-to-end performance of the laboratory-scale sorting system.

Evaluation Indicator	Result
Total number of test objects	974
Correctly sorted objects	938
Overall sorting accuracy	96.30%
Average throughput	58.3 objects/min
Average object interval	Approximately 1.03 s/object
End-to-end latency	312 ms
Continuous operating time	8 h
Software interruption	0
System crash	0
Operating status	Stable

5.3.5. System-Level Analysis

The laboratory results show that object detection, complete-image semantic segmentation, coordinate estimation, and actuator control can be connected in one workflow. PSSM-YOLO supplies object-level hypotheses, and CAA-DeepLabv3+ supplies pixel-level class regions from the full frame. Their outputs are associated only during postprocessing; the segmenter is not conditioned on detector boxes. In a campus teaching context, this integrated workflow could provide a visible demonstration platform for computer vision, embedded AI, and circular-economy education. The reported engineering metrics should not be interpreted as evidence of improved student learning because educational effectiveness was not measured.

For detector deployment, INT8 quantization with structured pruning achieved 90.1 FPS, 11.1 ms latency, 760 MB peak memory, and 5.1 W board power, with mAP50 and mAP50:95 of 85.9% and 72.6%. At the system level, the prototype achieved 96.30% aggregate accuracy, 58.3 objects per minute, and 312 ms end-to-end latency under controlled laboratory conditions.

Observed failures were associated with visually similar materials, reflection, deformation, partial covering, and overlap. The current evidence is limited by one dataset, controlled acquisition conditions, single-run model comparisons, one hardware platform, and a laboratory conveyor test. Independent cross-environment validation, repeated training, longer operation, and a verified ten-class system confusion matrix are required before broader practical claims can be made.

6. Conclusions, Result(s), Limitations, Future Studies

This study developed a detection–segmentation dual-driven vision framework for packaging waste recognition and sorting in sustainable educational recycling ecosystems. Using the self-constructed EPWaste dataset, the study evaluated both object-level detection robustness and pixel-level segmentation accuracy for ten representative categories of inner and outer express packaging waste. The main conclusions are as follows.

(1) EPWaste provides 4528 RGB images, 11,384 object instances, ten packaging categories, bounding-box annotations, and complete-image semantic masks. The fixed split contains 3170 training, 453 validation, and 905 test images.
(2) PSSM-YOLO integrates established attention, lightweight convolution, multi-scale fusion, and refined head components for the target task. In a controlled single run, it achieved 87.2% mAP50, 74.1% mAP50:95, 93.6% precision, and 85.8% recall. The ablation results reveal a metric trade-off: Shuffle Attention improved mAP50, precision, and recall but reduced mAP50:95 relative to the preceding configuration.
(3) CAA-DeepLabv3+ combines channel attention aggregation with CARAFE content-aware upsampling and operates on complete RGB images. It achieved 82.4% *mIoU*, 83.6% *mPA*, 83.3% *mPrecision*, 83.6% *mRecall*, and a mean Boundary F-score of 78.6%. No ground-truth or detector-derived ROI was used in training or evaluation.
(4) At 640 × 640 input, PSSM-YOLO and CAA-DeepLabv3+ require 12.8 M and 6.7 M parameters and 32.6 and 19.8 GFLOPs, respectively. On the Jetson Orin Nano, INT8 quantization with pruning reduced PSSM-YOLO latency to 11.1 ms and power to 5.1 W, with the stated accuracy trade-off. The laboratory prototype correctly sorted 938 of 974 objects across the ten-category output scheme, with 58.3 objects per minute and 312 ms end-to-end latency. These results establish laboratory feasibility only.

The study is limited by controlled acquisition, class imbalance, one dataset, single-run comparisons, incomplete public release of the enhanced annotations, and a laboratory-scale prototype. Future work will include repeated experiments, independent cross-environment testing, verified ten-class system confusion matrices, uncontrolled illumination, contamination, denser streams, higher conveyor speeds, longer operation, and release of the enhanced annotations after project completion and quality control. A further limitation is that the study did not evaluate educational outcomes, teacher workload, student accessibility, or the effects of automated feedback on recycling behavior. Future work should therefore use co-designed classroom or campus studies with teachers and students, alongside technical validation, to assess whether the system improves conceptual understanding and sustained sorting practices.

Author Contributions: Conceptualization, F.Y. and R.L.; methodology, F.Y. and R.L.; software, F.Y. and L.L.; validation, F.Y. and G.G.; formal analysis, R.L. and G.G.; investigation, F.Y. and G.G.; resources, F.Y. and R.L.; data curation, F.Y. and R.L.; writing—original draft preparation, F.Y.; writing—review and editing, F.Y.; visualization, F.Y. and G.G.; supervision, F.Y. and G.G.; project administration, F.Y. and G.G.; funding acquisition, F.Y. and G.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data underlying this article are available from the corresponding author on reasonable request, subject to project restrictions.

Acknowledgments: The authors greatly appreciate the constructive comments and suggestions from the editors and anonymous reviewers.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Cheng, X.; Yang, J.; Zhang, L. Express packaging waste recycling: Stakeholders' dynamic behavioral changes based on a computational evolutionary game theoretic approach. *Comput. Ind. Eng.* **2023**, *183*, 109519. [CrossRef]
2. Huang, H.; Zeng, X.; Ge, L.; Sun, K. Social media engagement in waste sorting: The role of sentiment in shaping public awareness. *Humanit. Soc. Sci. Commun.* **2025**, *12*, 1763. [CrossRef]

3. Cheng, X.; Wu, F.; Zhang, L.; Yang, J. Willingness to pay and its determinants: Comparing the deposit-refund and increased-price systems among Chinese citizens for express packaging waste management. *Environ. Impact Assess. Rev.* **2023**, *101*, 107148. [CrossRef]
4. Yang, J.; Long, R.; Chen, H.; Yang, M. Revealing the determinants of residents' recycling behavior of express delivery packaging: Insights from the network embeddedness. *Environ. Impact Assess. Rev.* **2024**, *105*, 107361. [CrossRef]
5. State Post Bureau of China. Chinese Statistic Bulletin of Post Business in 2023. Available online: https://www.spb.gov.cn/gjyzj/c100015/c100016/202401/59eeb6e8b0e7404f8127aa2c7aebded6.shtml (accessed on 26 September 2026).
6. Duan, H.; Song, G.; Qu, S.; Dong, X.; Xu, M. Post-consumer packaging waste from express delivery in China. *Resour. Conserv. Recycl.* **2019**, *144*, 137–143. [CrossRef]
7. Huang, H.; Long, R.; Chen, H.; Sun, K.; Sun, Q.; Li, Q. Examining public attitudes and perceptions of waste sorting in China through an urban heterogeneity lens: A social media analysis. *Resour. Conserv. Recycl.* **2023**, *199*, 107233. [CrossRef]
8. Su, Y.; Duan, H.; Wang, Z.; Song, G.; Kang, P.; Chen, D. Characterizing the environmental impact of packaging materials for express delivery via life cycle assessment. *J. Clean. Prod.* **2020**, *274*, 122961. [CrossRef]
9. Zhan, Y.; Sun, Y.; Xu, J. A study on the recycling classification behavior of express packaging based on UTAUT under "Dual Carbon" targets. *Sustainability* **2023**, *15*, 11622. [CrossRef]
10. Lu, W.; Chen, J. Computer vision for solid waste sorting: A critical review of academic research. *Waste Manag.* **2022**, *142*, 29–43. [CrossRef] [PubMed]
11. Zhang, Q.; Zhang, X.; Mu, X.; Wang, Z.; Tian, R.; Wang, X.; Liu, X. Recyclable waste image recognition based on deep learning. *Resour. Conserv. Recycl.* **2021**, *171*, 105636. [CrossRef]
12. Ma, B.; Zheng, R.; Lu, J.; Kong, L.; Zheng, Y.; Wan, T.; Ma, W. GVCG-YOLO: A fast and lightweight twin-fusion fiber-core algorithm for efficient kitchen waste detection. *Results Eng.* **2026**, *30*, 110115. [CrossRef]
13. Liu, H.; Jiang, Y.; Zhang, W.; Li, Y.; Ma, W. Intelligent electronic components waste detection in complex occlusion environments based on the focusing dynamic channel-you only look once model. *J. Clean. Prod.* **2025**, *486*, 144425. [CrossRef]
14. Ma, W.; Chen, H.; Long, R. An integrated detection-semantic fusion and near-infrared system for food-delivery packaging waste. *Waste Manag.* **2025**, *207*, 115125. [CrossRef] [PubMed]
15. Dong, Z.; Chen, J.; Lu, W. Computer vision to recognize construction waste compositions: A novel boundary-aware transformer (BAT) model. *J. Environ. Manag.* **2022**, *305*, 114405. [CrossRef] [PubMed]
16. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
17. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 91–99.
18. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
19. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single shot MultiBox detector. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
20. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
21. Ma, W.; Chen, H.; Zhang, W.; Huang, H.; Wu, J.; Peng, X.; Sun, Q. DSYOLO-trash: An attention mechanism-integrated and object tracking algorithm for solid waste detection. *Waste Manag.* **2024**, *178*, 46–56. [CrossRef] [PubMed]
22. Zhang, Q.; Yang, Q.; Zhang, X.; Wei, W.; Bao, Q.; Su, J.; Liu, X. A multi-label waste detection model based on transfer learning. *Resour. Conserv. Recycl.* **2022**, *181*, 106235. [CrossRef]
23. Chen, Y.; Luo, A.; Cheng, M.; Wu, Y.; Zhu, J.; Meng, Y.; Tan, W. Classification and recycling of recyclable garbage based on deep learning. *J. Clean. Prod.* **2023**, *414*, 137558. [CrossRef]
24. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
25. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241.
26. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef] [PubMed]
27. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
28. Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Semantic image segmentation with deep convolutional nets and fully connected CRFs. *arXiv* **2014**, arXiv:1412.7062.

29. Lu, W.; Chen, J.; Xue, F. Using computer vision to recognize composition of construction waste mixtures: A semantic segmentation approach. *Resour. Conserv. Recycl.* **2022**, *178*, 106022. [CrossRef]
30. Yong, Q.; Wu, H.; Wang, J.; Chen, R.; Yu, B.; Zuo, J.; Du, L. Automatic identification of illegal construction and demolition waste landfills: A computer vision approach. *Waste Manag.* **2023**, *172*, 267–277. [CrossRef] [PubMed]
31. Li, J.; Fang, H.; Fan, L.; Yang, J.; Ji, T.; Chen, Q. RGB-D fusion models for construction and demolition waste detection. *Waste Manag.* **2022**, *139*, 96–104. [CrossRef] [PubMed]
32. Wan, T.; Jiang, Y.; Hu, X.; Zheng, Y.; Ma, W. Intelligent construction and demolition waste detection in complex environments using an improved CLRD-YOLO model with knowledge distillation. *Case Stud. Constr. Mater.* **2026**, *25*, e06230. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.