Enhanced Small Object Detection with YOLOv8 in Garbage Detection

Hao Sun Hongchen Song

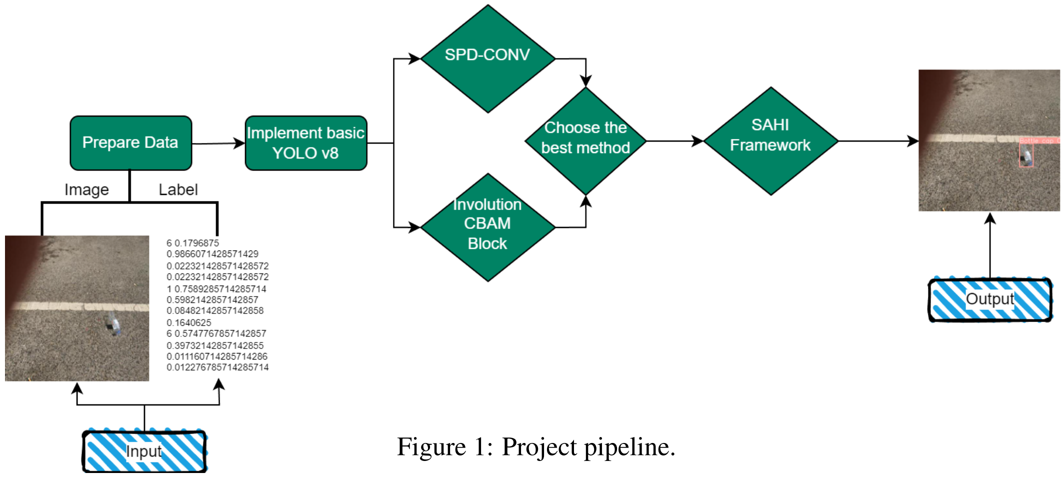

Figure 1: Project pipeline.

Abstract

With the dramatic increase in the amount of garbage worldwide, garbage classification and recycling have become a key part of environmental protection and resource recycling. The aim of this project is to develop a machine learning-based garbage recognition system for the automatic classification of daily garbage. We adopt deep learning methods, especially Convolution Neural Networks to recognize and classify various kinds of garbage. By collecting and processing a large amount of garbage image data, we trained a high-precision model and performed several rounds of optimization on it. Preliminary test results show that the system can effectively recognize and classify different kinds of garbage and provide a decision basis for its automatic sorting. The successful implementation of this system will not only improve the efficiency of garbage sorting but also encourage society to make more progress in garbage management and resource recovery.

1. Introduction

Object detection is a key task in computer vision, vital for applications ranging from autonomous driving to medical imaging and automated surveillance. As the challenge of global waste management intensifies, the application of object detection in automated garbage classification becomes increasingly crucial. However, a significant hurdle is the effective detection of small objects, a task where existing methods like the YOLO series often fall short. Our project aligns with technological advancements in AI and robotics, aiming to contribute to efficient, eco-friendly waste management in smart cities. To this end, we introduce an augmented YOLO model, specifically tailored to improve small object detection, particularly in the context of garbage detection tasks

2. Solutions

Based on our project's pipeline(Figure 1), we initially utilized the original YOLO model as a control group. We proceeded to enhance this model by incorporating SPD [6] and CBAM+Involution [7] methods, coupled with an added detection head to improve our object identification and classification efficiency. Subsequently, we conducted a series of ablation experiments to meticulously evaluate the individual contributions of each modification to the model's overall performance. This methodical approach enabled us to determine the most effective model configuration. Building upon this optimized foundation, we applied the SAHI framework [1], achieving further enhancements and attaining the best possible results.

SPD-Conv [6] includes a Space-to-Depth (SPD) layer and a non-strided convolution layer. It can replace strided convolutions and pooling layers in CNN architectures. The SPD layer(Figure 2) adeptly redistributes each spatial dimension of the input feature map to the channel dimension, effectively conserving the informational content within the channels. This is accomplished by transposing each input feature map pixel to a channel, thereby diminishing the spatial dimensions but augmenting the channel dimensions. The subsequent non-strided convolution layer, adhering to standard convolution operations, processes every pixel mapping, thus mitigating the down-sampling impact of the SPD layer and preserving detailed information. In our project, we have adapted the SPD-Conv block by replacing seven stride-2 convolutions in YOLOv8's backbone and neck with the SPD-Conv blockThis strategic replacement considerably enhances the CNN's efficacy in detecting low-resolution images and small objects, reducing dependency on high-quality input.

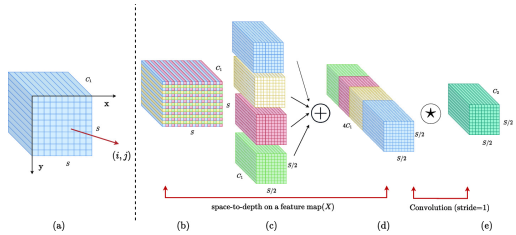

Figure 2: SPD-Conv Block

CBAM and Involution block [7] enhance detection efficacy for small objects. **CBAM**, positioned at the terminal end of the network's backbone, introduces a sophisticated attention mechanism that operates through two stages(Figure 3): channel and spatial attention. This mechanism adeptly prioritizes salient features across channels ('what' to focus on) and then hones in on relevant spatial regions within these channels ('where' to focus). The result is a dual-attention process that amplifies key features while suppressing the less relevant ones, enhancing the overall feature representation. Additionally, the CBAM's integration contributes to reducing computational load, simultaneously boosting the accuracy of the model in complex spatial scenarios.

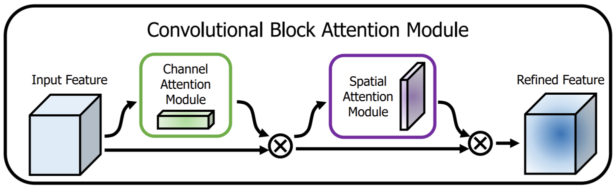

Figure 3: CBAM Block

Involution block(Figure 4), complementing CBAM, strategically placed between the backbone and the neck of the network, heralds a paradigm shift from traditional convolution operations to a more dynamic spatial feature processing approach. Unlike the uniform application of kernels in conventional convolution, the involution operation generates spatially specific kernels, allowing for the dynamic adjustment of pixel transformations based on contextual spatial information. This innovative approach enables more nuanced and context-aware processing of spatial details, proving especially beneficial in varied and complex scenes where spatial characteristics differ significantly. The operational principle of the Involution kernel is delineated

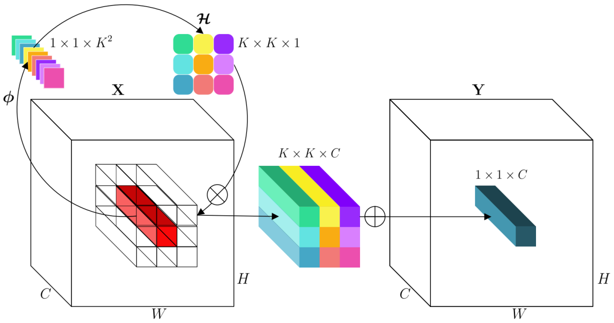

Figure 4: Involution Block

as follows: let $\mathbf{x}_{i,j}^{g} \in \mathbb{R}^{K \times K}$ represent the involution kernel for the g^{th} group, where K specifies the spatial dimension and the feature map is partitioned into G groups. The output feature map $\mathbf{Y}$ is computed as:

$$Y_{i,j,k} = \sum_{(u,v) \in \mathcal{K}} \mathbf{x}_{\lfloor i/K \rfloor, \lfloor j/K \rfloor, k}^{g} \cdot X_{i+u, j+v, k} \tag{1}$$

where $\mathcal{K}$ denotes the kernel size, and the involution operation dynamically adjusts the transformation for each pixel based on the surrounding spatial context. This results in a dispersion of channel-specific information across the spatial domain, thereby enriching the receptive field and enhancing the model's detection capabilities. We integrate these two methods thoughtfully within our network: CBAM slots into the Backbone, fine-tuning the feature extraction process right at the core, and involution takes its place within the Neck, ensuring that the feature pyramid generation is just as dynamic. This setup allows us to highlight essential features early on in the backbone phase, which is more efficient, given that the feature maps are smaller here than in the neck phase. The final structure like Figure 5. This placement keeps the computing costs in check while significantly boosting the network's ability to detect and interpret complex images.

SAHI(Slicing Aided Hyper Inference) [1] is a lightweight vision library designed for large-scale object detection and instance segmentation. It addresses the challenges in detecting small objects and performing inference on large images, which are critical issues in practical computer vision applications. SAHI provides a framework-agnostic approach for sliced or tiled inference, incorporating interactive user interfaces and tools for error analysis. Its utilities aim to assist developers in overcoming real-world problems in computer vision, particularly in enhancing the accuracy and efficiency of object detection and segmentation tasks. Its capabilities include both standard and sliced prediction methods, enabling more efficient processing of large images by dividing them into smaller, manage-

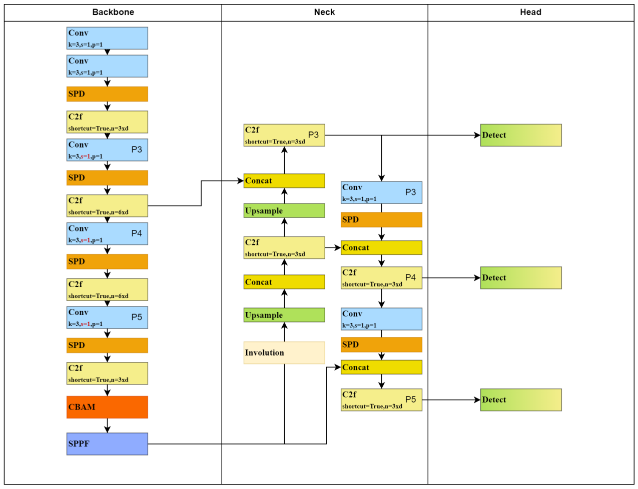

Figure 5: Our YOLOv8 Architecture

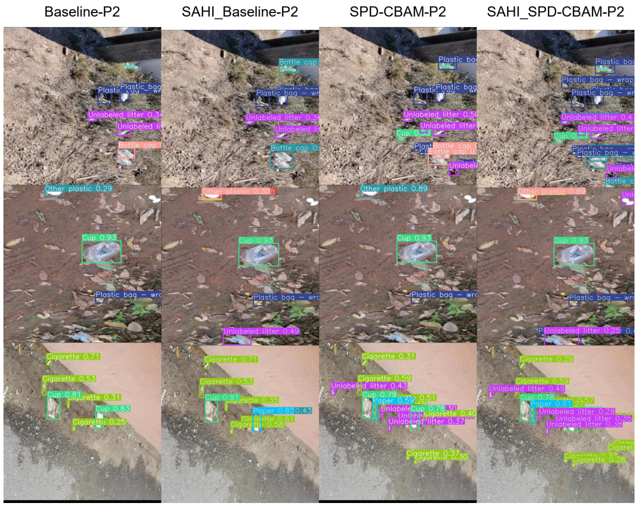

Figure 6: Best Model Visual Results

able slices. This slicing approach helps in maintaining high accuracy while dealing with large-scale data. Additionally, SAHI offers interactive visualization and inspection tools, which are crucial for error analysis and improving model performance. These tools provide users with a more intuitive understanding of the model's behavior and its decision-making process.

After implementing these modifications, our model demonstrated significant improvements in the experiments, which are detailed in our findings below.

3. Result

Training details In all experiments, we used the YOLOv8 model as a baseline and the TACO database as our datasets[4], focusing on enhancing the performance of small object detection. We utilize the automated setup provided by YOLO v8 [2]. For optimization, the Adam solver [3] is employed, with a configured batch size of 16. The training process has1000 epochs, during which we use the SGD optimizer [5]. This optimizer is set with a learning rate of 0.01 and a momentum of 0.9.

Visual Results The visual comparison across models suggests that the SPD-CBAM-P2 model outperforms others in detecting and identifying litter with greater accuracy and confidence. For instance, it correctly identifies and labels various objects like bottle caps, plastic bags, and cigarettes with high confidence scores (e.g., bottle cap at 1.00 and cigarette at 0.59), showcasing its robustness in recognizing a diverse range of litter against complex backgrounds.

Incorporating the SAHI methodology further refines the SPD-CBAM-P2 model's performance, as indicated by even higher confidence scores and precise bounding boxes around the identified objects. The SAHI-SPD-CBAM-P2 variant consistently demonstrates superior detection capabilities, as evidenced by its detailed recognition across all types of litter, making it the most effective model among those evaluated.

In a formal assessment, the baseline exhibited basic detection with lower confidence, while advanced methods showed notable improvements in both detection capabilities and confidence scores, with the hybrid model achieving the highest, reflecting enhanced recognition precision.

Ablation Study The SPD-CBAM-P2 model's structural complexity, characterized by its extensive layering and parameterization, underpins these performance gains. Comprising 309 layers and over 41 million parameters, the model manifests a computational complexity of approximately 90 GFLOPs. This complexity is indicative of the model's advanced processing and analytical capabilities, where the increased number of layers allows for deeper learning and data representation. Similarly, the expanded parameter count enhances the model's capacity to capture and retain a wide array of features. Although this results in heightened computational demands, as reflected in the significant GFLOPs, the model's intricate design, when adeptly trained, is apt to deliver more precise and detailed predictions, leveraging its sophisticated architecture for optimal performance. In evaluating the performance of the SPD-CBAM-P2 model, key metrics such as mean average precision (mAP) and recall are pivotal. These metrics provide insights into the model's accuracy and recognition capabilities, critical for assessing advancements in object detection algorithms. The SPD-CBAM-P2 model demonstrates its efficacy through these parameters, achieving an mAP of 34.30 at a 50% Intersection over Union (IOU) threshold and 25.63 at a more rigorous 50%-95%

Method	Layers	Parameters
SPD-CBAM-P2	309	4151122
SPD-CBAM	256	3948034
SPD	231	3541200
SPD-P2	284	3744288
CBAM	250	5914946
CBAM-P2	302	3781458
Baseline	225	3157200
Baseline-P2	277	3354144

Table 1: Object Detection Model Configurations: Layer and Parameter Counts

Method	mAP50 (%)	mAP50-95 (%)
SPD-CBAM-P2	34.30	25.63
SPD-CBAM	32.24	23.10
SPD	32.01	24.24
SPD-P2	35.03	26.10
CBAM	26.01	17.93
CBAM-P2	25.64	18.61
Baseline	23.77	17.15
Baseline-P2	25.31	18.33

Table 2: Performance of Object Detection Models at Different mAP Thresholds

Method	Precision (%)	Recall (%)
SPD-CBAM-P2	61.58	32.95
SPD-CBAM	61.44	30.09
SPD	61.16	29.85
SPD-P2	60.16	33.13
CBAM	53.39	23.60
CBAM-P2	50.40	24.88
Baseline	49.43	22.76
Baseline-P2	49.13	24.65

Table 3: Performance Evaluation of Object Detection Models: Precision and Recall

IOU threshold. These figures notably surpass those of competing models, illustrating the model's superior detection precision. Additionally, the model registers an average precision of 61.58 and a recall of 32.95, further confirming its high accuracy and efficiency in recognizing diverse object classes. When compared with its predecessors, the SPD-CBAM-P2 model exhibits a general improvement across all metrics, suggesting that its enhanced architecture significantly bolsters performance. The table below shows the configurations sorted by these performance metrics:

These results not only validate the effectiveness of our approach but also lay a foundation for an in-depth discussion on the applications and potential of our model.

Discussion In this work, SPD-Conv reshapes feature maps to enrich the input for subsequent processing, enhancing the network's channel information for improved feature selection by CBAM's attention mechanism, especially after SPD's dimensionality adjustment. CBAM then refines the model's capacity to capture details, focusing on the most informative aspects post-SPD conversion. Involution complements this by dynamically adapting to spatial contexts, allowing for a more granular feature discernment. The integration of these techniques enhances the detection of small and low-resolution targets while maintaining computational efficiency, a notable challenge in traditional convolutional networks. Collectively, SPD-Conv, CBAM, and Involution form a robust, efficient approach to handling multi-scale features and complex detection tasks, significantly boosting the model's adaptability and recognition accuracy across various object complexities and scales. Furthermore, our model integrates the SAHI (Sliced Approach for High-resolution Inference) framework, revolutionizing the detection of smaller objects. SAHI achieves this by dividing images into smaller segments, facilitating focused detection on each slice, and subsequently amalgamating the results. This sliced inference methodology sharpens the model's focus on smaller objects, potentially enhancing accuracy and performance in challenging tasks such as detecting minute pieces of garbage in diverse environments.

For applications like small garbage detection, incorporating SAHI yields substantial improvements. By implementing sliced inference, the system gains enhanced capabilities in recognizing and classifying small, inconspicuous items of litter. The precision in smaller areas, as indicated by increased confidence scores, is particularly evident in the detection of objects like bottle caps and cigarette butts when employing the SAHI methodology. Moreover, SAHI's compatibility with various detection models, coupled with its suite of visualization and analysis tools, significantly streamlines the development and evaluation of small object detection systems.

In summary, the integration of the SPD-Conv, CBAM, Involution, and the SAHI framework constitutes a robust and efficient approach to handling multi-scale features and complex detection tasks. This hybrid model demonstrates remarkable adaptability and recognition accuracy across a wide range of object complexities and scales.

References

[1] F. C. Akyon, S. Onur Altinuc, and A. Temizel. Slicing aided hyper inference and fine-tuning for small object detection. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 966–970, 2022. 1, 2

[2] G. Jocher, A. Chaurasia, and J. Qiu. Ultralytics yolov8, 2023. 3

[3] D. Kingma and J. Ba. Adam: A method for stochastic optimization. 2015. 3

[4] P. F. Proença and P. Simões. Taco: Trash annotations in context for litter detection. *arXiv preprint arXiv:2003.06975*. 3

[5] H. Robbins and S. Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951. 3

[6] R. Sunkara and T. Luo. No more strided convolutions or pooling: A new cnn building block for low-resolution images and small objects, 2022. 1

[7] S. Tang, Y. Fang, and S. Zhang. Hic-yolov5: Improved yolov5 for small object detection, 2023. 1, 2