electronics

Article

Small Foreign Object Detection in Automated Sugar Dispensing Processes Based on Lightweight Deep Learning Networks

Jiaqi Lu, Soo-Hong Lee *, In-Woo Kim, Won-Joong Kim and Min-Soo Lee

School of Mechanical Engineering, Yonsei University, Seoul 03722, Republic of Korea; jiaqi@yonsei.ac.kr (J.L.)
* Correspondence: shlee@yonsei.ac.kr

Abstract: This study addresses the challenges that conventional network models face in detecting small foreign objects on industrial production lines, exemplified by scenarios where a single piece of iron filing occupies approximately 0.002% of the image area. To tackle this, we introduce an enhanced YOLOv8-MeY model for detecting foreign objects on the surface of sugar bags. Firstly, the introduction of a 160 × 160-scale small object detection layer and integration of the Global Attention Mechanism (GAM) attention module into the feature fusion network (Neck) increased the network's focus on small objects. This enhancement improved the network's feature extraction and fusion capabilities, which ultimately increased the accuracy of small object detection. Secondly, the model employs the lightweight network GhostNet, replacing YOLOv8's principal feature extraction network, DarkNet53. This adaptation not only diminishes the quantity of network parameters but also augments feature extraction capabilities. Furthermore, we substituted the Bottleneck in the C2f of the YOLOv8 model with the Spatial and Channel Reconstruction Convolution (SCConv) module, which, by mitigating the spatial and channel redundancy inherent in standard convolutions, reduced computational demands while elevating the performance of the convolutional network model. The model has been effectively applied to the automated sugar dispensing process in food factories, exhibiting exemplary performance. In detecting diverse foreign objects like 2 mm iron filings, 7 mm wires, staples, and cockroaches, the YOLOv8-MeY model surpasses the Faster R-CNN model and the contemporaneous YoloV8n model of equivalent parameter scale across six metrics: precision, recall, mAP@0.5, parameters, GFLOPs, and model size. Through 400 manual placement tests involving four types of foreign objects, our statistical results reveal that the model achieves a recognition rate of up to 92.25%. Ultimately, we have successfully deployed this model in automated sugar bag dispensing scenarios.

Keywords: automatic material dispensing; deep learning; lightweight network; machine vision; small object detection

Citation: Lu, J.; Lee, S.-H.; Kim, I.-W.; Kim, W.-J.; Lee, M.-S. Small Foreign Object Detection in Automated Sugar Dispensing Processes Based on Lightweight Deep Learning Networks. *Electronics* **2023**, *12*, 4621. https://doi.org/10.3390/electronics12224621

Academic Editors: Haihua Chen, Yunhe Feng, Tozammel Hossain and Junhua Ding

Received: 27 September 2023
Revised: 7 November 2023
Accepted: 10 November 2023
Published: 12 November 2023

1. Introduction

Recent advances in deep learning have had a significant impact across a range of domains, from spam detection in product reviews to precision inspections in industrial applications. The development of hybrid spam detection models utilizing PU learning has been demonstrated to effectively identify deceptive practices within user-generated content [1]. Moreover, the need for robust security in smart contracts has led to innovative approaches in fuzz testing, revisiting the importance of call sequences and significant branches to uncover vulnerabilities [2]. In the realm of industrial inspections, techniques leveraging improved 3D point cloud instance segmentation have been applied to enhance the guidance accuracy of container ship units [3]. Furthermore, multimodal fusion in visual question answering systems, particularly the use of attention mechanisms, has been scrutinized to improve the interpretability and performance of these systems [4]. Lastly,

the sentiment classification of short texts remains a challenging task due to the sparsity of expressive features, but recent methods have shown promise in addressing this through improved multi-label approaches [5]. These diverse yet interconnected strands of research highlight the ongoing pursuit of more sophisticated and effective deep learning models.

One particular food factory, located in Zhejiang, China, specializes in producing chocolate and candy products, which have garnered appreciation from children across the world. Food production safety is a vital focus of this factory. On average, this factory requires tens of tons of sugar raw materials daily. In the past, the factory relied on manual labor for sugar dispensing, where workers used specialized tools to cut open sugar bags and visually inspected the surface of the bags for foreign objects like metal or insects. Long-term visual inspections can lead to eye fatigue, which can cause workers to miss or misjudge metallic objects, such as iron wires or staples, in sugar bags, posing a significant food safety risk; especially if these foreign objects are accidentally consumed by children, wherein the risks are even greater. Relying solely on visual inspection is not only inefficient, but also ties up considerable human resources, so that product safety cannot be consistent.

Traditional object detection algorithms that can replace human eye inspection mainly use a sliding window approach for input images to select candidate regions, extract features from these candidate regions, and then classify them. Consequently, these conventional object detection algorithms suffer from long execution times, complex feature extraction modules, high computational costs, low robustness of extracted features, and low recognition accuracy. In contrast, deep learning-based convolutional neural networks extract image features incrementally and learn feature information independently, resulting in a higher accuracy and smaller models [6,7]. Therefore, in recent years, deep learning-based convolutional neural networks have been widely used in the fields of foreign object and defect detection.

There are two categories within the number of detection stages. One includes two-stage detection algorithms, where candidate regions must first be generated before detection can be performed for those regions, resulting in longer detection times and a failure to meet real-time requirements. Examples of this include R-CNN [8], proposed by Girshick R; Fast-RCNN [9]; and Faster-RCNN [10]. The other category consists of one-stage detection algorithms, including the YOLO [11–14] series, the SSD [15–17] series, and OverFeat [18].

YOLOv8 [19], the latest model in the YOLO series, offers improved feature extraction and accuracy functions compared to previous versions. However, the model parameters and complexity of the structure have also increased. In practical real-time industrial inspection scenarios, efficient model inference is required in addition to high accuracy. For example, due to production capacity requirements in automatic sugar bag dispensing, the entire inspection process for small foreign objects must be completed within a time frame of 0.6 s. Therefore, this paper focuses on the improvement and optimization of YOLOv8n for the detection of superficial foreign objects detection on sugar bags.

In typical cases, common methods to slim down a network using mobile devices usually involve model compression, such as model pruning [20,21], model quantization [22], etc. However, the matrix structure of fully connected layers obtained by these compression methods is irregular and sparse, which does not suit modern hardware acceleration platforms such as the graphics processing units (GPUs). As a result, the computation time remains the same despite reduced mesh parameters, and thus the requirements for a lightweight mesh cannot be met.

For these reasons, lightweight networks have been proposed, which aim to increase feature extraction capabilities while pursuing network lightweightness. A new convolution module, Spatial and Channel Reconstruction Convolution (SCConv) [23], introduces a spatial reconstruction unit (SRU) [23] and channel reconstruction unit (CRU) [23] to reduce feature redundancy, thereby not only reducing parameters but also enhancing performance. GhostNet [24] uses a simple linear operation to generate similar feature maps and capture network redundancy. Based on these lightweight networks, researchers have started to modify detection networks. Li et al. [25] used GhostNet to modify the detection of the

COCO dataset. Zhang et al. [26] applied MobileNetv3 to modify YOLOv3 for pedestrian detection under drone conditions. Although the above methods achieved good detection results, challenges with foreign objects and defect detection in actual industrial scenarios still need to be addressed. In industrial scenes, the ratio of the target detection instance area to the image area is often between 0.08% and 0.5%, belonging to the small target detection problem [27]. The detection of foreign objects on the surface of sugar bags in this study used three industrial cameras to inspect a 90 cm × 50 cm area on the sugar bag surface through continuous stitching. In accordance with the food safety standards of the manufacturing facility, raw materials should be devoid of iron contaminants exceeding 2 mm in length. Even according to the maximum area of iron filings, 2 mm × 2 mm, a single iron filing object would only occupy about 0.002% of the image area, hence it is part of a high-demand small target detection problem, akin to finding an ant on a monitor.

In order to maintain the same lightweight level and fast inference capability as YOLOv8n while enhancing accuracy, particularly in small object detection, the primary goal of this research was to optimize and design a lightweight model suitable for industrial small object detection.

This study selected the YOLOv8 model as the basic framework. It made a series of modifications to address the challenges of small foreign object detection in industrial environments and the limited computing capabilities of computers deployed in factories. The modified model was successfully deployed for foreign object inspection on the surface of sugar bags in automated sugar bag dispensing equipment:

1. The introduction of a 160 × 160-scale small object detection layer and the simultaneous integration of the Global Attention Mechanism (GAM) [28] attention module into the feature fusion network (Neck) increased the network's interest in small objects, improved the network's feature extraction and fusion capabilities, and ultimately increased the accuracy of its small object detection.
2. The lightweight network GhostNet was employed to replace DarkNet53, which is YOLOv8's leading feature extraction network, which reduced the number of network parameters and strengthening the feature extraction capability.
3. In the YOLOv8 model's C2f segment, the traditional Bottleneck was substituted with the Spatial and Channel Reconstruction Convolution (SCConv) module. This modification effectively minimized spatial and channel redundancies, leading to a reduction in computational overhead and model storage requirements. Simultaneously, this substitution bolstered the overall performance of the convolutional network.

2. Introduction to YOLOv8 Network

YOLOv8, as the latest YOLO model in 2023, is divided into YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x based on different network depths and widths. Considering the size of the model, this study opted for the YOLOv8n network, which is smaller and more precise. The YOLOv8n detection network primarily consists of four parts: Input, Backbone, Neck, and Head, as seen in Figure 1.

The Input uses Mosaic for data augmentation, but this is turned off for the last 10 epochs. It employs an anchor-free mechanism to directly predict the object's center, instead of the offset of the known anchor box. This approach reduces the number of anchor box predictions and accelerates non-maximum suppression (NMS) [29]. The Backbone component primarily serves the purpose of feature extraction and encompasses modules such as Conv, C2f, and SPPF: the Conv module primarily performs convolution, BN, and SiLU activation functions on the input image; the C2f in YOLOv8n has a newly designed structure, which is the main module for learning residual features, allowing YOLOv8n to have rich gradient flow of information while ensuring lightweightness; and the SPPF, also known as spatial pyramid pooling, can convert feature maps of any size into feature vectors of a fixed size.

The Neck network primarily fuses multi-scale features and generates a feature pyramid. The Neck adopts the PANet structure, its core composed of a feature pyramid network

(FPN) [30] and path aggregation network (PAN) [31]. FPN first extracts feature maps from the convolutional neural network to construct a feature pyramid and then uses up-sampling and coarser-grain feature map fusion to achieve the integration of features at different levels, but FPN alone would lack object position information; PAN therefore supplements FPN, using a bottom-up structure, fusing feature maps from different levels through a convolutional layer and precisely retaining spatial information. This combination of FPN and PAN fully realizes the fusion of the network's up-and-down information flow, enhancing the network's detection performance. As the final prediction part, the Head output obtains the category and location information of different-sized target objects according to feature maps of different sizes.

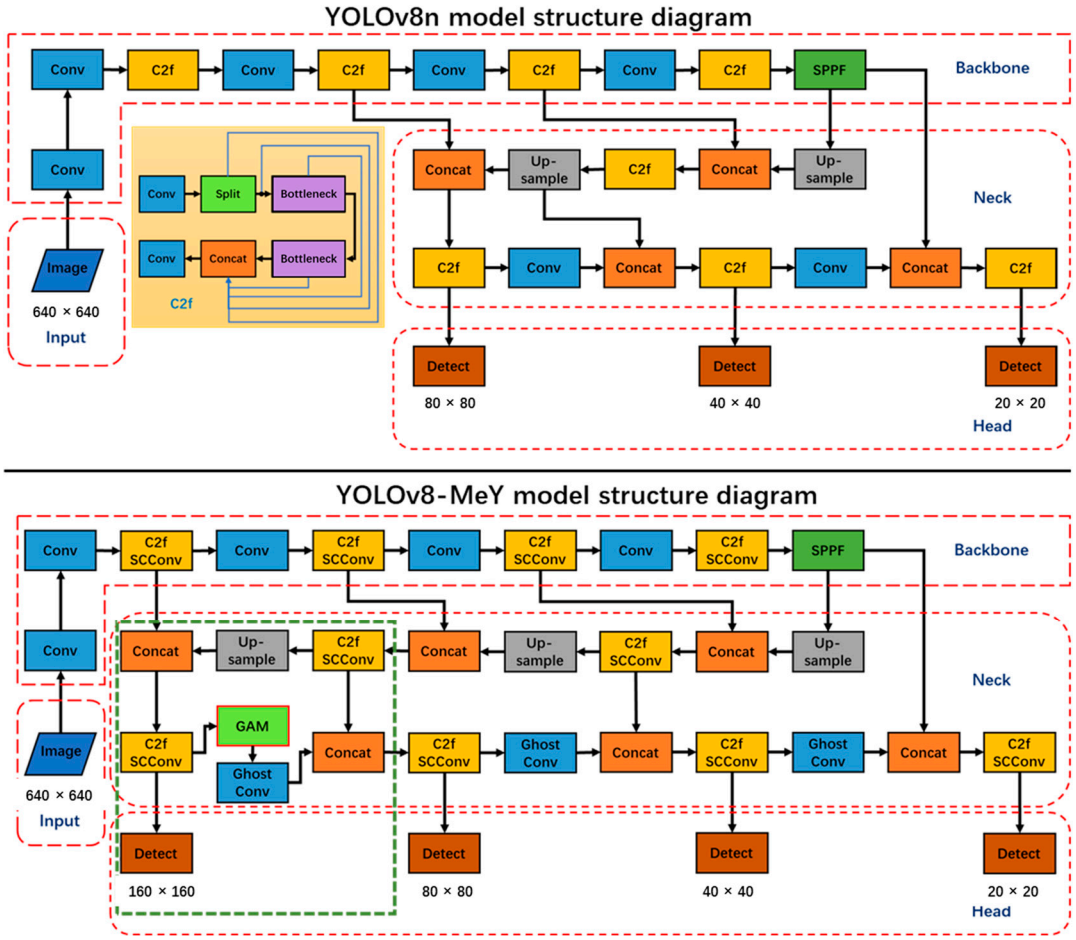

Figure 1. YOLOv8n model and YOLOv8-MeY model structure diagrams. Our model introduced a small object detection layer at the 160 × 160 scale, and the lightweight network GhostNet was employed to replace DarkNet53. Also, the Spatial and Channel Reconstruction Convolution (SCConv) module was used to replace the Bottleneck in C2f.

3. Improvement Methods for YOLOv8 Algorithm

To address issues such as inaccuracy in detecting minor object anomalies and the excessive size and parameter quantity of traditional networks, this paper introduces a model for detecting foreign objects on the surface of sugar bags based on YOLOv8n. The network structure of the proposed algorithm model, YOLOv8-MeY, as seen in Figure 1, mainly consists of three parts. Since the images based on foreign object detection primarily involve small objects, we have added a small object detection layer, as shown by the green box in Figure 1. The Spatial and Channel Reconstruction Convolution (SCConv) module replaces the Bottleneck in the C2f of the original YOLOv8 model, reducing computation by minimizing the widespread spatial and channel redundancy in standard convolution, and

simultaneously enhancing the performance of the convolutional network model. Additionally, GhostConv from the GhostNet network has been introduced to reduce the model's quantity of parameters.

3.1. Small Object Detection Layer

Due to the small size of the target objects and the relatively large down-sampling factor of YOLOv8, deeper feature maps have difficulty capturing the features of these small objects, resulting in a limited ability of the original YOLOv8 model to detect small objects. The input image size of the original model is 640 × 640 pixels, and the smallest detection scale is 80 × 80 pixels, with a receptive field of 8 × 8 pixels for each grid. However, in this paper, the minimum size for foreign object detection is 4–5 pixels. To improve its capability of detecting small foreign objects, the YOLOv8-MeY improved model in this paper introduces a layer for detecting small objects at the 160 × 160 scale and integrates the attention module GAM into the feature fusion network (Neck). This not only increases the network's interest in small objects, but also strengthens its feature extraction and fusion capabilities, ultimately leading to improved accuracy when detecting small objects.

3.1.1. GAM Attention Module

In the context of foreign object detection, the dataset used in this scenario usually consists of images where the background occupies a significant portion of the image. There is also the need to detect small objects within these foreign objects. To improve the model's ability to represent features of small foreign objects, this algorithm introduces the Global Attention Mechanism (GAM). The GAM is a novel attention module that follows CBAM [32], as shown in Figure 2. It is a mechanism that combines both channel attention and spatial attention, enabling an enhancement of the interactions between features in the global dimension while reducing scattered information. A feature map (F1) passes through two separate attentional submodules. First, it is corrected using the channel attention mechanism to obtain an intermediate state (F2). Then, it is further refined using the spatial attention mechanism, resulting in the final feature map (F3).

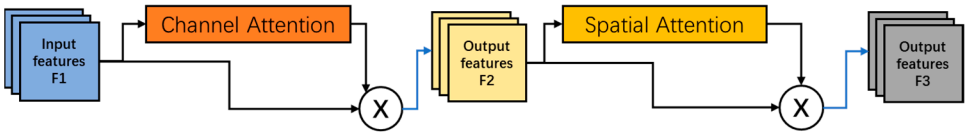

Figure 2. GAM attention module.

The channel attention submodule, shown in Figure 3, uses a three-dimensional arrangement to obtain three-dimensional information. It employs a two-layer multilayer perceptron (MLP) to enhance cross-channel spatial dependencies.

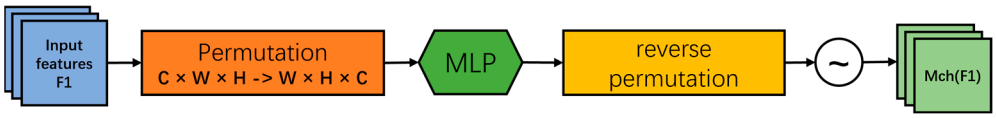

Figure 3. Channel attention submodule.

The definition of the intermediate state is provided in Formula (1) (where Mch represents the channel attention map, and ⊗ denotes element-wise multiplication):

$$F_2 = Mch(F_1) \otimes F_1 \quad (1)$$

The definition of the intermediate state is found in Formula (2) (where Msp represents the spatial attention map):

$$F_2 = Msp(F_2) \otimes F_2 \quad (2)$$

The spatial attention submodule, shown in Figure 4, operates on an input feature map (F2) with dimensions C × H × W. It uses two 3 × 3 convolutions to model the nonlinear relationships between pixels in a 3 × 3 block, so that the parameters can better capture the spatial dependencies among pixels.

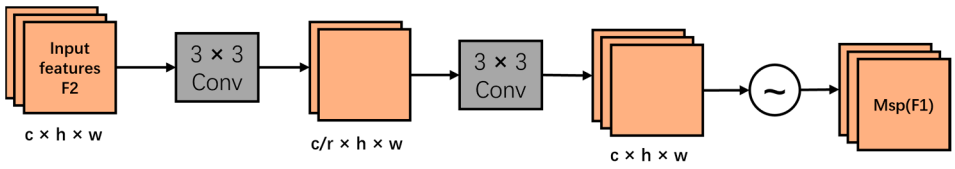

Figure 4. Spatial attention submodule.

3.1.2. 160 × 160 Small Object Detection Layer

Due to the small size of the target objects and the relatively large down-sampling factor of YOLOv8, deeper feature maps have difficulty capturing the features of these small objects, resulting in the original YOLOv8 model's limited ability to detect small objects. The original model has an input image size of 640 × 640 pixels, and the smallest detection scale is 80 × 80 pixels, with a receptive field of 8 × 8 pixels for each grid. If the width and height of the targets in the original image are both smaller than 8 pixels, the original network finds it challenging to recognize the target features within the grid. In this paper, the requirement was to detect very small foreign objects, i.e., 2 mm iron debris. In a scenario with a field of view of 52.4 cm × 37.5 cm, the obtained image size is 3840 × 2748 pixels, with the length of the 2 mm iron debris occupying only 15 pixels. When resized to the 640 × 640 base format, the minimum length of the target after resizing is only 4 to 5 pixels. However, the current minimum detection feature map size is 80 × 80, which is used to detect targets larger than 8 × 8. To ensure that targets 4–5 pixels long or more can be detected, we decided to add a small object detection layer that can detect targets of at least 4 × 4 pixels.

As shown in the green box in Figure 1, adding a small object detection layer of a 160 × 160 scale into the original network, which includes supplementary fusion feature layers, introduces additional detection heads to enhance the semantic information and feature expression capability of small objects. Firstly, the 80 × 80-scale feature layer in the fifth layer of the Backbone and the up-sampling feature layer in the Neck adds to the stack of layers. After C2f and up-sampling processing, a deep semantic feature layer containing the small object feature information is obtained, which continues to stack onto the shallow position feature layer in the third layer of the Backbone, supplementing and improving the expression ability of the 160 × 160-scale fusion feature layer to obtain the semantic features and position information of small objects. Finally, the data is sent to an additional Decoupled Head in the Head through the C2f. This supplementary Head part allows the feature information of small objects to continue to be transmitted to the feature layers of the other three scales along the down-sampling path through the Head structure, thereby enhancing the network's feature fusion ability and improving its accuracy in the detection of small objects. Adding a Decoupled Head can expand the detection range for foreign objects.

3.2. SCConv Module

The SCConv module, standing for Spatial and Channel Reconstruction Convolution, depicted in Figure 5, represents a novel compression method for CNNs, aiming to jointly reduce the spatial and channel redundancy present in convolutional layers. It achieves considerable performance enhancement while significantly reducing the computational load through two unique modules: the spatial reconstruction unit (SRU) [23] and the channel reconstruction unit (CRU) [23], designed to minimize redundancy in the feature maps. The SRU and CRU are two core modules within SCConv, designed to address redundancy challenges in convolutional neural networks.

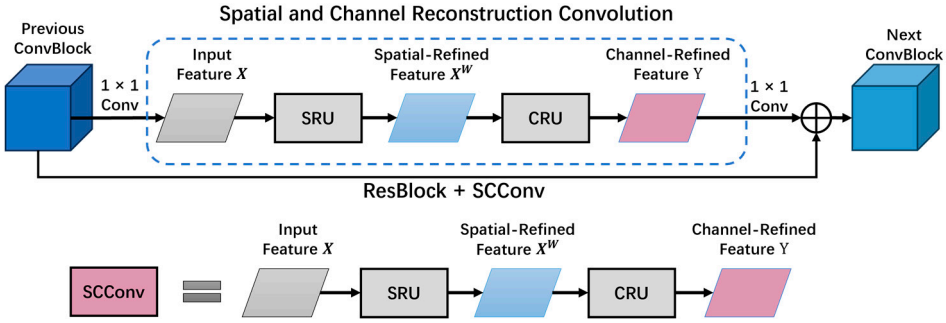

Figure 5. The architecture of SCConv is integrated with the spatial reconstruction unit (SRU) and channel reconstruction unit (CRU). This figure shows the exact position of our SCConv module within a ResBlock.

The SRU primarily focuses on the spatial dimension of feature maps; its goal is to mitigate spatial redundancy. To achieve this, the SRU employs a mechanism that breaks down the input feature map into multiple spatial blocks, applying distinct convolution kernels to each. This approach not only captures feature information more precisely within each block but also significantly reduces the overall spatial redundancy. As a result, feature extraction becomes more efficient and accurate while also lessening the computational complexity.

On the other hand, while the CRU is similarly geared toward optimizing redundancy, its focus is on the channel dimension of the feature maps. To process channel information more flexibly and efficiently, the CRU introduces a lightweight fully connected layer. This allows the model to amalgamate information more effectively from different channels, thereby decreasing channel redundancy. This method not only enhances the discriminative power of the features but further reduces computational demands.

Moreover, SCConv serves as a plug-and-play module, offering versatility and compatibility without necessitating any alterations to the existing model structure, hence quickly replacing standard convolution. Extensive experiments on various SOTA methods in image classification and object detection have demonstrated that models embedded with SCConv achieve a better balance between their performance and efficiency.

In the design of YOLOv8-MeY, as detailed in this article, SCConv substitutes the Bottleneck in the C2f of the original YOLOv8 model. The C2f is transformed into the C2f–SCConv structure, as shown in Figure 6, thereby reducing the prevalent spatial and channel redundancy in standard convolution and consequently enhancing the performance of the convolutional network model, all while lowering computational demands.

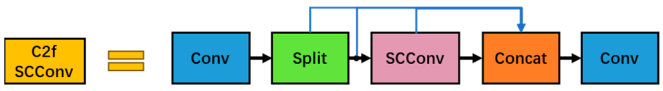

Figure 6. C2fSCConv structure.

3.3. GhostNet Module

GhostNet is a lightweight network, and GhostConv is a convolutional module within the GhostNet network that can replace regular convolutions. The core idea of GhostConv is to decompose the conventional convolution. As shown in Figure 7, GhostConv first extracts features from a small number of non-linear convolutions and then uses linear convolution operations to generate Ghost feature maps. Then, feature maps from both convolutional stages are combined to obtain feature maps with more channels. This process eliminates redundant features and reduces the computational load of the model.

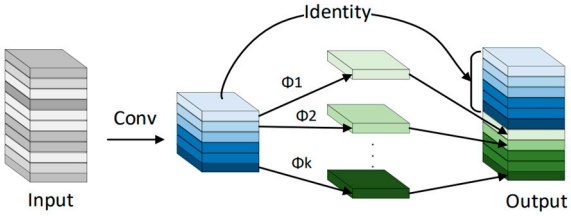

Figure 7. The GhostConv module. An illustration of the convolutional layer and the proposed Ghost module for outputting the same number of feature maps. Φ represents the Cheap operation.

The lightweight GhostNet network model maintains the size of the original convolutional output feature maps while reducing the computational cost and number of parameters. It achieves this by first using a small number of regular convolution kernels to extract feature information from the input feature maps. Then, it applies a less expensive linear transformation operation to the feature maps compared to regular convolution. Lastly, the final feature map is generated via concatenation. That is, the first step uses a 1×1 convolution kernel to produce a condensed feature of the input feature layer. In the second step, a computationally efficient depth-wise convolution is applied to the generated feature maps to obtain another half of the feature maps. Finally, the two sets of feature maps are concatenated together, resulting in a feature map with a similar expressive capability to a traditional convolutional layer. The comparison of computational parameters between Ghost convolution and traditional convolution is calculated as follows in Formula (3):

$$\begin{aligned} Comp &= \frac{Ghost_{parameters}}{Conv_{parameters}} = \frac{\frac{n}{s} \times h' \times w' \times m \times k \times k + (s-1)\ \frac{n}{s} \times h' \times w' \times k \times k}{h' \times w' \times n \times k \times k \times m} \\ &= \frac{\frac{1}{s} \times h' \times w' \times k \times k \times m + \frac{(s-1)}{s} \times k \times k}{k \times k \times m} \\ &\approx \frac{s+m-1}{s \times m} \\ &\approx \frac{1}{s} \end{aligned} \tag{3}$$

where:

m represents the number of input channels;
h represents the height of the input feature map;
w represents the width of the input feature map;
n represents the number of output channels;
h' represents the height of the output feature map;
w' represents the width of the output feature map;
$k \times k$ represents the size of the convolution kernel;
s represents the number of Ghost feature maps; and s is much smaller than m.

Thus, the theoretical compression ratio is approximately equal to the number of Ghost feature maps (s). Therefore, Ghost convolution greatly reduces the computational load of network parameters. The numerator in Equation (3), $\text{Ghost}_{\text{Parameters}}$, represents the parameter quantity of Ghost convolution, while the denominator, $\text{Conv}_{\text{Parameters}}$, represents the parameter quantity of traditional convolution.

As shown in Figure 8, the "Cheap operation" is a low-cost linear operation. In our research, GhostConv first applies a convolution with the same size as the original 3×3 convolution to generate half of the feature maps. Subsequently, another 3×3 convolution with a step size of one is performed in the cost-effective computation (Cheap operation) to obtain the other half of the feature maps. Finally, these two sets of feature maps are joined together to form a complete feature map.

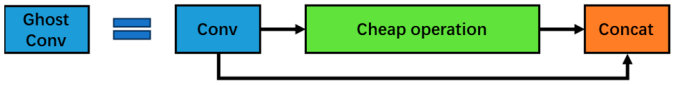

Figure 8. GhostConv structure.

The introduction of GhostConv into the YOLOv8-MeY design significantly reduces the size of the model and its number of parameters and computational requirements. This allows the model to be easily deployed in low-end computers and embedded devices in factories, facilitating online inspection in the context of industrial digital transformation.

4. Experimental Design and Results

During this research process, an automated system for dispensing sugar ingredients based on visual detection of foreign objects was used and designed in Figure 9. The cyclic workflow of the system is as follows:

1. The operator transports each stack of sugar bags to a designated track via a forklift, with each stack consisting of 12 bags arranged in four layers, each bag measuring 90 cm × 54 cm and weighing 50 kg. The forklift transports ten stacks at a time.
2. Upon sensing the arrival of the stack, the conveyor belt below moves it to the area directly beneath the 3D camera and within the robotic arm's range, where it remains stationary.
3. Using the positional information of each layer of sugar bag, provided by the 3D camera, the ABB robotic arm grabs the bags and moves them over to the dispensing table. The aim is for the fixed blades on the table to pierce the sugar bags.
4. The robotic arm continues to tear the sugar bags with a straight horizontal glide, shaking the torn bags to dispense all the sugar onto the table. The entire process achieves automated dispensing through the cooperation of the robotic arm and 3D camera.

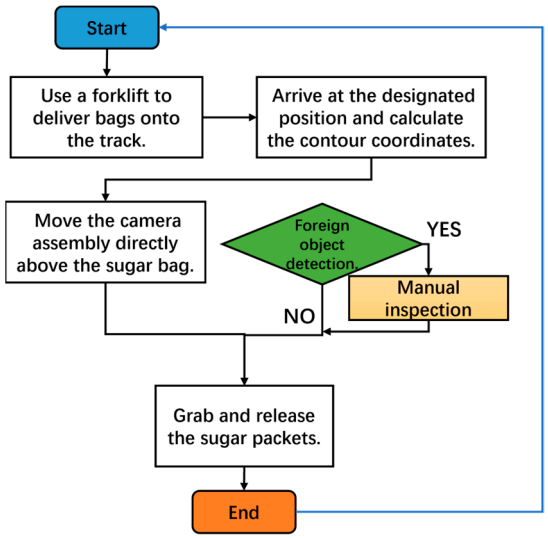

Figure 9. The cyclic workflow of the automated system for dispensing sugar ingredients based on visual detection of foreign objects.

The YOLOv8-MeY model proposed in this paper will be integrated into step 3. Furthermore, to meet production capacity requirements, images of the sugar bags that are going to be grabbed must be captured and inspected for the presence of foreign objects within 2 s before the robotic arm's grabbing action. Running the YOLOv8n model in a Pytorch environment. Once a foreign object is detected, it will generate a signal or warning. We have preset this signal as the character "0". This signal is then transmitted to the PLC via the industrial communication protocol, Modbus. Upon receiving this signal, the PLC

processes it. Based on predetermined logic, the PLC will generate an emergency stop output signal to the robotic arm, instructing it to halt all movements.

4.1. Experimental Design

In this study, three sets of Basler aAC3800-14uc industrial cameras with a resolution of 3840 × 2748 pixels, i.e., 10.55 million pixels; Keyence CA-LH8 lenses controlling the field of view at 52.4 cm × 37.5 cm; and three CSS LDR2-90SW white light sources are combined to form a set of equipment for image acquisition and detection, as shown in Figure 10. Simultaneously, to protect the camera lens and other equipment from direct impact damage and reduce the direct contact of sugar dust particles in the factory with the camera and lens, we designed protective covers for the camera and light sources to prevent dust particles from contacting the camera and other equipment. The camera and light sources are fixed through right-angle and circular fixing brackets. The center distance between the three cameras is 30 cm. These devices are connected to an industrial control computer carrying the YOLOv8-MeY model via data cables for the foreign object discrimination detection program. Additionally, the protective cover has dimensions of 80 cm × 40 cm × 40 cm and is made from 2 mm-thick stainless steel. The casing, equipped with visual detection devices, is then securely locked in a position parallel to the front of the robotic arm's vacuum-gripping jaws. The purpose of this is to allow the visual detection system to inspect for foreign objects on the surface of individual sugar packets before each pick-up with the robotic arm. The industrial control computer has a CPU: i5 12400f, GPU: GTX 1060 6 G, and 16 G of RAM memory. The software environment version is Python: 3.9, Pytorch: 2.0.0, and CuDNN: 8.4.1.

Figure 10. Vision detection system mounted at the front end of the robotic arm's gripper. Three sets of industrial cameras (Basler aAC3800-14uc) and lenses (Keyence CA-LH8), along with white ring-light illuminators (CSS LDR2-90SW), are parallelly mounted at equal intervals of 30 cm within a stainless steel protective enclosure measuring 80 cm × 40 cm × 40 cm. They are also secured parallel to the front of the mechanical arm's vacuum-gripping claw. The purpose of this arrangement is to enable the vision inspection system to detect any foreign objects on the surface of the individual sugar packets before each one is picked by the robotic arm.

During actual operation, with the cooperation between the 3D camera and the robot, the gripper can accurately reach above the sugar bag and maintain a height of 55 cm relative to the center point of the sugar bag. It is important to note that each bag of sugar has a maximum length of 90 cm and a maximum width of 50 cm when placed horizontally. In this project, the field of view for a single camera is 52.4 cm × 37.5 cm, with a depth of 55 cm.

Since the three cameras are placed side by side with a horizontal spacing of 30 cm after adjustment, the resulting field of view is 97.5 cm × 52.4 cm, covering the entire sugar bag.

When the robot reaches the predetermined photo-taking position, it will send a photo-taking signal to the industrial control computer equipped with the YOLOv8-MeY model to initiate the foreign object detection program. The foreign object detection program will automatically trigger the camera to take pictures, and the results will be returned to the program through a data cable. The program will check whether the images returned by the camera are standard, and then resize the three images to 800 × 800 before feeding them into the YOLOv8-MeY model for computation and returning the identified information. The results computed by the model are finally converted into the final signal for foreign object detection and sent to the robot via a network connection. If a foreign object is detected and manual intervention is needed, the images from the detection will be saved. The entire foreign object detection program will continue to monitor while operating normally. If an anomaly occurs with the camera or other programs, leading to computation failure, an error signal will be immediately sent to the robot. This will halt the following grabbing action and trigger an alarm to summon the staff for manual confirmation.

4.2. Image Collection and Data Processing

Under the premise of ensuring the consistency of the laboratory environment with the on-site equipment-operating environment, we set up three light sources parallel to and above the sugar bags in the laboratory space. During the experiment, we aimed to find an appropriate and reproducible light source brightness. We placed various foreign objects at different positions on the sugar bag and captured images while adjusting the brightness with a light source controller. The light source controller has a total of 10 levels, and can be adjusted in increments of 0.5 levels each time. After that, we visually inspected the images under each brightness level and selected three sets of data with the most effective brightness values. The average brightness value from these three sets was set as the final brightness value. Regarding the creation of the dataset, there are four major types of foreign objects: (1) 2 mm iron shavings; (2) 7 mm wires; (3) staples; and (4) cockroach insects, as shown in Figure 11. For each session, we randomly and uniformly threw 8–12 samples of the same type of foreign objects onto the surface of the sugar bags, ensuring that the camera can generally photograph each sample. After collecting the images, we screened them to determine whether there were issues such as occlusion, incomplete shooting, or repeated shooting. If any of the above problems occurred, the picture was deemed invalid and deleted. For the valid pictures, a LabelImg tool was used for manual annotation. In total, we collected 3500 images of 2 mm iron shavings (35,462 labels), 950 images of 7 mm wire (12,120 labels), 1080 images of staples (12,410 labels), and 960 images of cockroach samples (2594 labels) in Table 1. Ultimately, we wrote script programs to divide the pictures automatically and randomly, with 90% as the training set and 10% as the validation set. In the subsequent training process, we used data augmentation methods such as Mosaic, Mixup [33], and random probability methods like rotation and flipping, enhancing the model's generalization ability for foreign object detection at different angles and brightness levels.

Table 1. Number of dataset sample images and labels.

Category	**Single Foreign Object Category**			
Foreign Object Classification	2 mm Iron Shavings	7 mm Iron Wires	Staples	Cockroaches
Number of Sample Images	3500	950	1080	960
Number of Sample Labels	35,462	12,120	12,410	2594

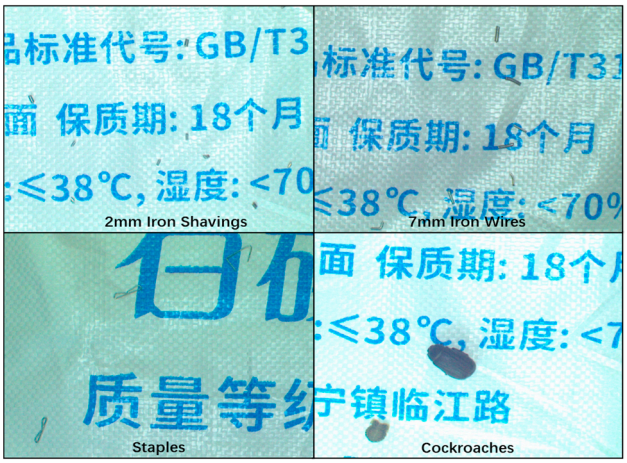

Figure 11. The four types of samples. Samples of four different foreign objects were randomly scattered onto the surface of the sugar bag, including 2 mm iron shavings and 7 mm iron wires, each having two different levels of reflectivity and thicknesses of 1.5 mm and 2 mm, respectively; various randomly bent paper clips; and cockroach insects of different body sizes. (The Chinese text on the sugar bags is a product description printed on the bags before they leave the factory. The content of the Chinese characters is not related to this study and can be regarded as background).

4.3. Actual Deployment and Verification Comparison

To verify the practicality of the YOLOv8-MeY foreign object detection algorithm proposed in this paper, we used precision, recall, mAP@0.5, parameters, and model size as evaluation indicators. We compared our algorithm with Faster R-CNN, YOLOv5, and YOLOv8 under the same conditions, configurations and dataset. From Table 2, it can be observed that YOLOv8n, with fewer parameters and a smaller model size compared to the other five networks, outperforms Faster R-CNN and YOLOv5n in terms of precision, recall, and mAP@0.5. Although YOLOv5s is slightly better than YOLOv8n, the parameter count and model size of YOLOv5s are almost double that of YOLOv8n. The improved algorithm proposed in this paper, YOLOv8-MeY, has a smaller parameter count and model size compared to the original YOLOv8n, and it also surpasses the original YOLOv8n algorithm in terms of precision, recall, mAP@0.5, parameters, GFLOPs, and model size. In the actual validation dataset, when compared with the YOLOv8-MeY model, YOLOv8n experienced many missed detections when detecting 2 mm and 7 mm iron wires, as seen in Figure 12. The optimized YOLOv8-MeY shows a significant improvement in the detection of these small objects.

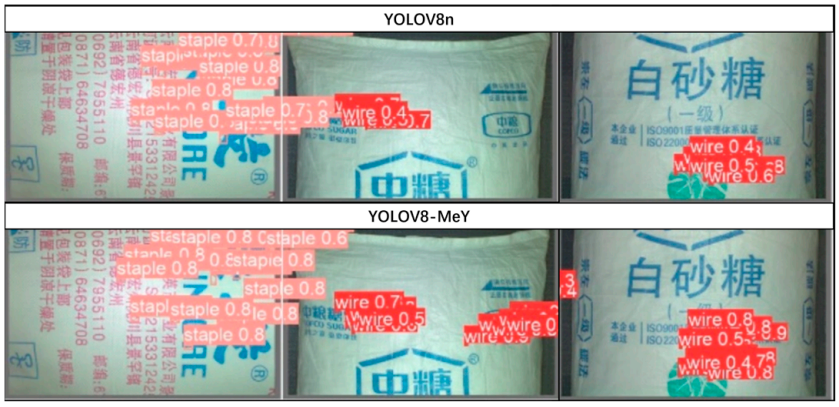

Figure 12. Comparison of detection performance between YOLOv8n and YOLOv8-MeY.

Table 2. Contrasting experiment results.

Models	Precision	Recall	mAP@0.5	mAP@0.5–0.95	FPS	Parameters/M	Model Size/MB
Faster R-CNN	0.935	0.929	0.941	0.682	26	28.39	106.89
YOLOv5n	0.916	0.918	0.927	0.568	105	1.9	7.2
YOLOv5s	0.940	0.939	0.945	0.663	49	7.13	14.68
YOLOv8n	0.928	0.926	0.935	0.576	87	3.019	6.27
Ours YOLOv8-MeY	0.939	0.938	0.944	0.562	75	2.46	6.02

4.4. Ablation Experiments

To evaluate the effectiveness of the algorithmic improvements in this paper, the original YOLOv8n model was used as the baseline model. Metrics such as precision, recall, mAP@0.5, mAP@0.5–0.95, FPS, parameters, and model size were used for evaluation. Ablation experiments were performed with different combinations of improvement modules, as shown in Table 3:

1. Table 3 shows that compared to the original YOLOv8n model, the model with the additional small object detection layer showed an increase of 0.002 in precision, 0.003 in recall, 0.003 in mAP@0.5, and 0.007 in mAP@0.5–0.95. However, the model's inference speed was reduced by 15 fps. After adding the GAM attention mechanism, there was an increase of 0.005 in precision, 0.005 in recall, 0.002 in mAP@0.5, and 0.009 in mAP@0.5–0.95, but the model's inference speed was then reduced by 14 fps. The ablation experiments in this stage verified that the introduction of the 160 × 160-scale small object detection layer, along with the concurrent integration of the Global Attention Mechanism (GAM) attention module into the feature fusion network (Neck), heightened the network's focus on small objects, enhanced the network's feature extraction and fusion abilities, and ultimately elevated the accuracy of its small object detection.
2. The introduction of the lightweight GhostConv module resulted in a 0.03 increase in mAP@0.5, as well as a 2.82 G reduction in parameters, a reduction of 71 FPS, and a 6.41 MB reduction in model size; but the mAP@0.5–0.95 was decreased by 0.007. The ablation experiments in this stage verified that employing the lightweight network GhostNet to replace DarkNet53 can reduce the number of network parameters and strengthen the feature extraction capability.
3. The addition of SCConv led to a 0.001 increase in recall and mAP@0.5, at least a 2.46 G reduction in parameters, a reduction of 75 FPS and a 6.02 MB reduction in model size; but the mAP@0.5–0.95 decreased by 0.025. The ablation experiments in this stage verified that the traditional Bottleneck was successfully replaced with the Spatial and Channel Reconstruction Convolution (SCConv) module. This modification effectively reduced spatial and channel redundancies, leading to a decrease in computational overhead and model storage requirements. Concurrently, this substitution enhanced the overall performance of the convolutional network.

Table 3. Ablation experiments. To evaluate the effectiveness of the algorithmic improvements in this paper, the original YOLOv8n model was used as the baseline model. Ablation experiments were performed with different combinations of improvement modules.

Small Layer	GAM	GhostConv	SCConv	Precision	Recall	mAP@0.5	mAP@0.5–0.95	FPS	Parameters/M	Model Size/MB
X	X	X	X	0.928	0.926	0.935	0.576	87	3.02	6.28
O	X	X	X	0.930	0.929	0.938	0.583	72	3.06	6.55
O	O	X	X	0.935	0.934	0.940	0.594	58	3.37	7.19
O	O	O	X	0.938	0.937	0.943	0.587	71	2.82	6.41
O	O	O	O	0.938	0.938	0.944	0.562	75	2.46	6.02

The experimental results in this table demonstrate that the improved YOLOv8-MeY model in this paper outperforms the original YOLOv8n network model. Despite a slight decrease in mAP@0.5–0.95, the improved model shows significant increases in precision (0.010), recall (0.012), mAP@0.5 (0.009), and FPS (12). Moreover, the parameters and the model size decreased by 0.56 G and 0.26 MB, respectively, confirming the effectiveness of the algorithmic improvements proposed in this paper. However, the frequent use of feature compression and the 1×1 and 3×3 convolutions led to a loss of detail in the feature maps, resulting in unsatisfactory performance in mAP@0.5–0.95.

4.5. Model Experimental Results and Analysis

Based on the results in Table 4, we decided to select the YOLOv8-MeY model for actual simulation testing. We placed four types of foreign objects at the bag's top, middle, and bottom. These objects were placed against complex backgrounds such as red and blue fonts, shown in Figure 13. Subsequently, we randomly placed the same type of foreign objects at each location for 400 tests and calculated the accuracy based on the model's identification results. If the model successfully identified the foreign object, we marked it as *T* (True); if it misidentified the type of foreign object, we marked it as *F* (False). For any cases of excessive sensitivity caused by the model, we marked them as *E* (Error). Ultimately, we calculated the *Pr* (precision) and the false–positive *Ac* rate (accuracy) using Formulas (1) and (2).

$$Pr = \frac{T}{T + F + E} \times 100\% \tag{4}$$

$$Ac = \frac{E}{T + F + E} \times 100\% \tag{5}$$

Table 4. Results of testing with manual placement of foreign object samples between YOLOv8n and YOLOv8-MeY.

Model	2 mm Iron Shavings%	7 mm Iron Wires%	Staples%	Cockroaches%
YOLOv8n (Pr)	75.00	91.75	96.75	98.00
YOLOv8_MeY (Pr)	79.50	93.50	97.50	98.50
YOLOv8n (Ac)	7.75	2.25	1.00	0.50
YOLOv8_MeY (Ac)	6.75	2.00	1.00	0.25

Figure 13. The sugar bag is divided into four areas. Blue zone: area with a complex background; yellow zone: area with a simple background; green zone: wrinkle and seam area; and orange zone: four right-angle corner areas. For the YOLOv8n and YOLOv8-MeY models, we manually placed various types of foreign objects in each area and tested them 100 times, resulting in a total of 400 single-type foreign object recognition tests, and then compiled the results into a confusion matrix.

Based on the confusion matrix, illustrated in Figure 14, it was found that the YOLOv8-MeY model performs better than the YOLOv8n model regarding the four detection rates. Especially when detecting 2 mm iron shavings, the YOLOv8-MeY model had a 4.5% higher *Pr* (precision) than the YOLOv8n model and a 0.89% improvement in *Ac* (accuracy).

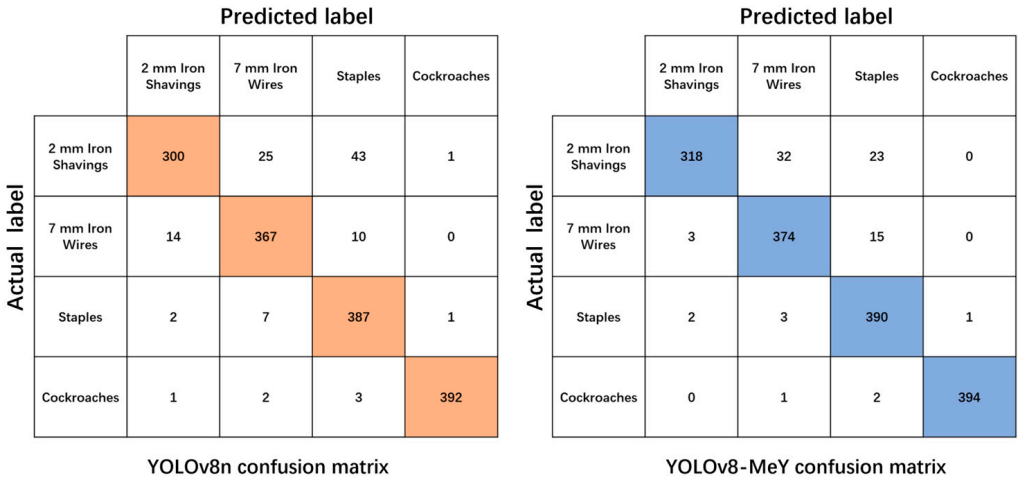

Figure 14. Confusion matrix results. The performance of the YOLOv8-MeY model is superior to that of the YOLOv8n model, especially in detecting 2 mm iron shavings as foreign objects.

The average *Pr* (precision) and Ac of the YOLOv8-MeY model were 92.25% and 2.25%, respectively, as seen in Table 3. However, with specific backgrounds, such as at the edges of the small red and blue fonts and the blue patterned part at the bottom of the bag, the model had a poor detection performance for the 2 mm targets. However, on-site usage scenarios are more concerned about whether foreign objects are detected than misjudgments between categories. Therefore, these verification results are acceptable.

5. Discussion

The use of Basler aAC3800-14uc industrial cameras in our experimental setup has proven to be highly effective for high-resolution image capture in an industrial environment. The high pixel density of these cameras, combined with the Keyence CA-LH8 lenses, provided the wide and detailed field of view necessary for accurate foreign object detection on the surface of sugar packets. The spatial arrangement of the cameras, ensuring full coverage of the sugar bags' surface area, exemplifies a thoughtful consideration of physical constraints within the experimental design. The adaptation of the YOLOv8-MeY model to an industrial control computer highlights the growing trend of integrating advanced machine learning algorithms with traditional industrial machinery. The robustness of the YOLOv8-MeY model, which outperforms its predecessors and competitors in most metrics, is still limited by the physical realities surrounding its deployment, such as lighting conditions and the subtlety of foreign object characteristics against complex backgrounds.

In the ablation experiments, the incorporation of a 160 $\times$ 160-scale small object detection layer and the simultaneous integration of the Global Attention Mechanism (GAM) attention module into the feature fusion network (Neck) heightened the network's sensitivity to small objects. While the GhostConv and SCConv modules reduced computational demands and improved processing speeds, the slight decrease in mAP@0.5–0.95 suggests that there is still room for improvement in the model's generalization ability across different scenarios, particularly in the detection of finer details. Additionally, as we push towards operational deployment, the model's lower detection performance concerning certain image backgrounds remains a concern. It emphasizes the difference between controlled environments and complex real-world conditions.

6. Conclusions

This study's addition of the 160 × 160-scale small object detection layer and simultaneous integration of the GAM attention module into the feature fusion network (Neck) increased the network's interest in small objects, improved the network's feature extraction and fusion capabilities, and ultimately increased the accuracy of small object detection. Additionally, the lightweight network GhostNet was used to replace DarkNet53, the main feature extraction network of YOLOv8, which reduced the number of network parameters while improving the feature extraction ability. At the same time, the spatial and channel reconstruction module SCConv was used to replace the Bottleneck in the C2f of the YOLOv8 model, reducing the computational cost and model storage size by cutting down the spatial and channel redundancy widely existing in standard convolutions.

The improved YOLOv8-MeY network, when applied to actual cases of foreign object detection on the surface of sugar bags, shows a reduction in parameters and model size, and performs better in six aspects: precision, recall, mAP@0.5, parameters, FPS, and model size, compared to the original YOLOv8n algorithm. Through actual manual placement defect testing, this model achieved a Pr recognition rate of 92.25% in detecting foreign objects on the surface of sugar bags. Particularly in detecting the 2 mm iron shavings, the YOLOv8-MeY model showed a 4.5% improvement in Pr recognition rate compared to the YOLOv8n model. The false–positive Ac rate increased by 0.89%. Currently, it has been deployed and is in use in automated food factories.

While maintaining the same lightweight level as YOLOv8n, YOLOv8-MeY has improved accuracy in small object detection, especially in detecting tiny 2 mm foreign objects, where it shows a significant improvement. However, its performance in the mAP@0.5–0.95 metric is not as good as that of YOLOv8n. The frequent use of feature compression along with 1 × 1 and 3 × 3 convolutions leads to a loss of detail in the feature maps, resulting in an unsatisfactory performance in the mAP@0.5–0.95 metric. Fortunately, in industrial applications, users are more concerned about the model's ability to accurately detect and recognize the presence of foreign objects than its capability to infer their precise location in the target area.

Regarding future research objectives and directions, while these improvements in detecting small objects, particularly 2 mm foreign objects, are promising, there are areas in our current approach that warrant further exploration. One significant area needing further investigation is the optimization of the model's performance in the mAP@0.5–0.95 metric. The evident loss of detail in the feature maps due to its frequent use of feature compression and 1 × 1 and 3 × 3 convolutions indicates a potential avenue for improvement. We will look into refining our convolution strategies and feature extraction methodologies to better capture the nuanced details essential for achieving higher accuracy across broader evaluation metrics.

Author Contributions: Conceptualization and methodology, J.L.; writing—original draft preparation, J.L., I.-W.K., W.-J.K. and M.-S.L.; formal analysis, S.-H.L.; writing—review and editing, S.-H.L. and J.L. 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 in this study is unavailable due to privacy reasons.

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

References

1. Wu, Z.; Cao, J.; Wang, Y.; Wang, Y.; Zhang, L.; Wu, J. hPSD: A hybrid PU-learning-based spammer detection model for product reviews. *IEEE Trans. Cybern.* **2018**, *50*, 1595–1606. [CrossRef] [PubMed]
2. Liu, Z.; Qian, P.; Yang, J.; Liu, L.; Xu, X.; He, Q.; Zhang, X. Rethinking smart contract fuzzing: Fuzzing with invocation ordering and important branch revisiting. *IEEE Trans. Inf. Forensics Secur.* **2023**, *18*, 1237–1251. [CrossRef]
3. Zong, C.; Wan, Z. Container ship cell guide accuracy check technology based on improved 3D point cloud instance segmentation. *Brodogr. Teor. Praksa Brodogr. Pomor. Teh.* **2022**, *73*, 23–35. [CrossRef]

4. Lu, S.; Liu, M.; Yin, L.; Yin, Z.; Liu, X.; Zheng, W. The multi-modal fusion in visual question answering: A review of attention mechanisms. *PeerJ Comput. Sci.* **2023**, *9*, e1400. [CrossRef] [PubMed]
5. Liu, X.; Shi, T.; Zhou, G.; Liu, M.; Yin, Z.; Yin, L.; Zheng, W. Emotion classification for short texts: An improved multi-label method. *Humanit. Soc. Sci. Commun.* **2023**, *10*, 306. [CrossRef]
6. Ahmad, M.; Ding, Y.; Qadri, S.F.; Yang, J. Convolutional-neural-network-based feature extraction for liver segmentation from CT images. In Proceedings of the Eleventh International Conference on Digital Image Processing (ICDIP 2019), Guangzhou, China, 10–13 May 2019; SPIE: Bellingham, WA, USA, 2019; Volume 11179, pp. 829–835.
7. Qadri, S.F.; Shen, L.; Ahmad, M.; Qadri, S.; Zareen, S.S.; Khan, S. OP-convNet: A patch classification-based framework for CT vertebrae segmentation. *IEEE Access* **2021**, *9*, 158227–158240. [CrossRef]
8. Agrawal, P.; Girshick, R.; Malik, J. Analyzing the performance of multilayer neural networks for object recognition. In Proceedings of the Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014; Proceedings, Part VII 13. Springer International Publishing: New York, NY, USA, 2014; pp. 329–344.
9. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
10. 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*. [CrossRef]
11. 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, 27–30 June 2016; pp. 779–788.
12. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
13. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
14. Bochkovskiy, A.; Wang, C.Y.; Liao HY, M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
15. 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; Proceedings, Part I 14. Springer International Publishing: New York, NY, USA, 2016; pp. 21–37.
16. Fu, C.Y.; Liu, W.; Ranga, A.; Tyagi, A.; Berg, A.C. Dssd: Deconvo-lutional single shot detector. *arXiv* **2017**, arXiv:1701.06659.
17. Li, Z.; Zhou, F. FSSD: Feature fusion single shot multibox detector. *arXiv* **2017**, arXiv:1712.00960.
18. Sermanet, P.; Eigen, D.; Zhang, X.; Mathieu, M.; Fergus, R.; LeCun, Y. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv* **2013**, arXiv:1312.6229.
19. Terven, J.; Cordova-Esparza, D. A comprehensive review of YOLO: From YOLOv1 to YOLOv8 and beyond. *arXiv* **2023**, arXiv:2304.00501.
20. Han, S.; Mao, H.Z.; Dally, W.J. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv* **2015**, arXiv:1510.00149.
21. Frankle, J.; Carbin, M. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv* **2018**, arXiv:1803.03635.
22. Zhu, F.; Gong, R.; Yu, F.; Liu, X.; Wang, Y.; Li, Z.; Yang, X.; Yan, J. Towards unified INT8 training for convolutional neural network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 1966–1976.
23. Li, J.; Wen, Y.; He, L. SCConv: Spatial and Channel Reconstruction Convolution for Feature Redundancy. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 6153–6162.
24. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. Ghostnet: More features from cheap operations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 1580–1589.
25. Li, S.; Sultonov, F.; Tursunboev, J.; Park, J.H.; Yun, S.; Kang, J.M. Ghostformer: A GhostNet-based two-stage transformer for small object detection. *Sensors* **2022**, *22*, 6939. [CrossRef] [PubMed]
26. Zhang, X.X.; Li, N.; Zhang, R.X. An improved lightweight network MobileNetv3 based YOLOv3 for pedestrian detection. In Proceedings of the IEEE International Conference on Consumer Electronics and Computer Engineering, Guangzhou, China, 15–17 January 2021; pp. 114–118.
27. Chen, C.; Liu, M.Y.; Tuzel, O.; Xiao, J. RCNN for small object detection. In *Proceeding of Asian Conference on Computer Vision, Taipei, Taiwan, 20–24 November 2016*; Springer: Cham, Switzerland, 2016; p. 214230.
28. Liu, Y.; Shao, Z.; Hoffmann, N. Global attention mechanism: Retain information to enhance channelspatial interactions. *arXiv* **2021**, arXiv:2112.05561.
29. Bodla, N.; Singh, B.; Chellappa, R.; Davis, L.S. Soft-NMS—Improving object detection with one line of code. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5561–5569.
30. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
31. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.

32. Mi, N.; Zhang, X.; He, X.; Xiong, J.; Xiao, M.; Li, X.Y.; Yang, P. CBMA: Coded-backscatter multiple access. In Proceedings of the 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), Dallas, TX, USA, 7–9 July 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 799–809.
33. Zhang, L.; Deng, Z.; Kawaguchi, K.; Ghorbani, A.; Zou, J. How does mixup help with robustness and generalization? *arXiv* **2020**, arXiv:2010.04819.