An Improved Waste Detection and Classification Model Based on YOLOV5

Fan Hu, Pengjiang Qian(✉), Yizhang Jiang, and Jian Yao

School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi 214122, Jiangsu, China
qianpengjiang@jiangnan.edu.cn

Abstract. The improvement in people's lives has resulted in a significant rise in the amount of household garbage created on a daily basis, to the point where waste separation can no longer be disregarded, especially for the series of problems: manual waste classification is time-consuming and laborious, and human waste classification errors are caused by a lack of knowledge reserve related to waste classification. To address these issues, we propose a waste classification network YOLO-CG optimized on the basis of YOLOV5 network structure in campus scene. Firstly, YOLO-CG draws lessons from the optimization idea of Transformer performance improvement by stacking the ConvNeXt Blocks in the ratio of 3:3:9:3 as backbone, adding the big size kernel and other adjustments, upgrading the mean average precision (mAP) of the network model by 5%. Then, to maintain the original accuracy while reducing the number of parameters, a computationally reduced cheap operation is introduced, which employs a simple 3 * 3 convolution to achieve a low-cost acquisition of redundant feature maps, resulting in a reduction of 12% in parameter count while also increasing the mAP. Both theoretical analysis and experiments demonstrate the effectiveness of the improved network model.

Keywords: Trash classification · Object detection · CNN

1 Introduction

China has advocated for the adoption of waste separation and has created related policies to support the sustainable development of society and to do a good job of conserving resources and protecting the environment. However, the implementation results are unsatisfactory due to a lack of environmental awareness and responsibility, or a lack of understanding. Traditional garbage sorting has the drawbacks of poor sorting efficiency, high cost, and high personnel demand, making it difficult to meet efficient and accurate sorting standards, resulting in environmental and resource difficulties. For example, take university campus domestic waste discharge, where a typical area of the campus classroom and dormitory waste has a concentrated and unclassified characteristic, with recyclable waste accounting for 69% of classroom waste and 45% of dormitory waste, respectively. Although most of the waste is correctly sorted, a large proportion is still mixed with perishable waste and other rubbish and then disposed of. This problem is

D.-S. Huang et al. (Eds.): ICIC 2022, LNAI 13395, pp. 741–754, 2022.
https://doi.org/10.1007/978-3-031-13832-4_61

mainly due to the lack of common sense and habit about waste classification, which leads to inefficient initial classification of waste and increases the cost of time and effort for manual secondary classification [1]. With the rapid advancement of technology as a driving force of urban development, smart city construction has become a necessary path for the development of every city, and the organic combination of big data, cloud computing, and artificial intelligence has become effective solution many traditional urban management problems. In Deep learning has yielded rich academic results in the field of computer vision, so various image classification and detection models based on convolutional neural networks have been put to use with remarkable results in urban management such as traffic. Thus, using machine learning for waste classification can effectively compensate for the shortcomings of traditional waste classification and integrate deep learning models with mechanical automation technology for waste classification and disposal work to improve waste disposal automation, save time and space, improve classification accuracy, and partially replace manual work to reduce inefficient work caused by.

The literature [2] proposed a garbage classification algorithm based on deep separable convolutional attention module (DSCAM), which can capture the intrinsic relationship between channels and spatial locations in garbage image features using two attention modules with deep separable convolution, and use ResNet as backbone to improve the recognition ability of the network, which can focus on important classification information in garbage classification scenarios and ignore irrelevant information. The literature [3] presents intelligent public garbage can design based on machine vision and auxiliary sensors that increase the detection accuracy of irregular waste categorization by combining sensors and optimizing vision recognition algorithms before they are used. For recyclable waste, the literature [4] proposes a migration learning-based image classification model, in order to avoid improper handling of recyclable waste in the waste sorting process, which leads to waste of resources. The literature [5], to address the complexity of the marine environment and the lack of hardware for underwater filming, proposes an embedded garbage automatic detection algorithm based on the complex marine environment, which is improved on the basis of the Mask R-CNN algorithm and aims to achieve high accuracy marine garbage detection and instance segmentation. These studies combine software and hardware to present new research paths and ideas for automated trash categorization processing. As a result, the development of a deep learning-based garbage categorization method is both significant for research and practical purposes.

2 Related Research

2.1 Deep Learning

Convolutional neural networks (CNNs) have grown significantly, thanks to the continuous development of optimized computing power conditions, in recent years in the disciplines of image classification, object detection, and other computer vision applications, making deep learning one of the most prominent machine learning approaches. Researchers have presented numerous strong convolutional neural network models with various properties like a data-driven method with a basic structure, few training parameters, and automated feature extraction. In the ImageNet competition in 2012, AlexNet

[6] outperformed all non-deep learning-based models by an absolute margin, demonstrating the importance of deep learning in vision problems. Deep learning based on convolutional neural networks have evolved fast, such as GoogleNet [7], VGG-Net [8], and methods that successfully increase picture categorization accuracy. ResNet [9], proposed by Microsoft team He et al. in 2015, introduced the concept of residual learning to find a solution to vanishing gradient and explosion gradient caused by the excessive depth of the neural network stacked into pure convolutional layers at the time, which resulted in the degradation of the network model rather than increasing effectiveness. The researchers proposed using a Batch Normalization layer in data preprocessing and networks to solve the problem of gradient explosion or disappearance, and a Residual structure to mitigate the degradation problem. The core idea of this structure is to introduce an identify shortcut connection, which is formally a nonlinear mapping of stacked data. The main idea behind this structure is to introduce an identify shortcut connection, which formally entails fitting a new mapping to the stacked nonlinear layers and rewriting the old mapping with the new one, because the new mapping is easier to optimize than the original, and thus can effectively address the problem of training accuracy degradation.

Although convolutional neural networks have advanced rapidly, and various network models have achieved excellent results in various competitions, these network models typically have high computational complexity and large model sizes, making them insufficient to meet application requirements in resource-constrained situations such as computational power, storage space, and power consumption. Researchers have turned their focus to lightweight network models to decrease the re-search threshold for convolutional neural networks, increase the application capability of various algorithms on edge mobile devices, and make research results ready for production usage. In 2017, the Google team proposed Transformer [10] to achieve good achievements in the field of NLP and accomplish the SOTA effect at the moment. The Google team, which primarily applied Transformer to the concept of im-age classification and also led Transformer's following research in the field of computer vision, offered the resultant ViT in 2020.

2.2 Object Detection

The rapid advancement of CNN-based object detection algorithms has paralleled the advancement of convolutional neural networks. Girshick et al. presented the R-CNN [11] method in 2014, which is the first two-stage object detection algorithm that produces object candidate areas first and then extracts features. The Mean Average Precision (mAP), a key validation metric in the field of object detection, was enhanced to 53.3%, which is 30% better than the previous best result, and it's the first time that object detection algorithms based on the CNN framework have been significantly improved. Various R-CNN-based enhancement algorithms, such as Fast R-CNN [12], Faster R-CNN [13], Mask R-CNN [14], and so on, were presented one after the other until the application criteria were met. Researchers studied feature fusion to suggest methods like FPN [15] and M2Det [16] to tackle the unsatisfactory impact of tiny object objection. Since 2016, Redmon J has proposed the One-Stage algorithm YOLO (You Only Look Once) based on the regression method, and improved many optimizations by solving the

inaccurate detection frame, poor detection of small targets, and proposing the Anchor-Free series of algorithms YOLO algorithm [17–20] so that One-Stage object detection algorithm gradually catches up with the Two-Stage. The SSD (Single Shot MultiBox Detector) series [21–23] algorithms were also proposed in the one-stage category, which improved significantly the performance of the object detection algorithm in terms of precision and real-time performance.

3 Algorithm Design

In our network, we not only replace normal Conv with Ghost Conv in the head part of YOLOV5, but we also replace bottleneck from the original C3 module with Ghost bottleneck in the stride of 1. The essence of Ghost bottleneck is a stack of two Ghost modules twice, the first module is used to increase the feature dimension (expansion ration), and the second module is used to decrease the feature dimension to make it consistent with a shortcut. Figure 1 depicts the model diagram presented in this study, with YOLOV5 as the fundamental framework, ConvNeXt as the backbone, and Ghost module as the computational complexity reducer, we named the network YOLO-CG.

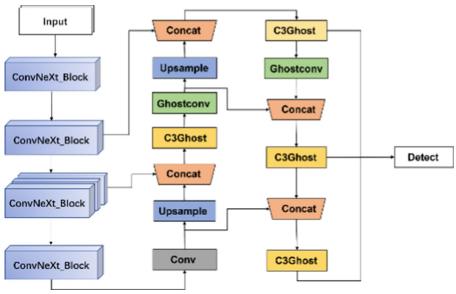

Fig. 1. The Figure is schematic of YOLO-CG with YOLOV5 as the fundamental framework, ConvNeXt as the backbone, and ghost module as the computational complexity reducer.

3.1 YOLOV5

The YOLOV5 [24] framework was introduced in the process of continuous iterative optimization of the YOLO series algorithms, which are incremented as n, s, m, l, and x according to model size, and each model differs in-network depth and width, all of which are made up of four parts: input, backbone, neck, and head. The input largely uses Mosaic data augmentation, adaptive initial anchor frame computation, image scaling, and other techniques to pre-process the image. Backbone originally used the Focus module to down-sample, improved CSP structure, and SPP pooling pyramid structure to extract feature information of images. However, many edge mobile device chips do not support the Focus operator when deploying YOLOV5 training models, and some optimization algorithms with their associated GPU devices found through experiments that the 6 * 6 convolutional layer is more efficient than using the Focus module, so the

Focus module was replaced in later iterations V6.0 and later. Neck mostly uses the FPN + PAN feature pyramid structure to transport feature information of various sizes and address the multi-scale problem. The head to increase the accuracy of network prediction using NMS, the loss functions are employed to compute classification, localization, and confidence loss.

The Conv module is a composite convolution module, and its construction is depicted in Fig. 2(a). The convolution layer, the BN layer, and the activation function layer are all included in the package module. Bottleneck is a simple residual block structure, as shown in Fig. 2(b), that is stacked and embedded in the C3 module for feature learning, using two Conv modules to minimize the number of channels and then expand the alignment to extract feature information, and using the shortcut to control whether to connect residuals. Figure 2(c) shows the construction of the C3 module, which is a modified BottleneckCSP module. The input feature map flows through two branches in the C3 module: the first branch goes through a Conv module, and then the features are learned by the stacking Bottleneck module; the other branch, which is a residual connection, only goes through one Conv module. Finally, the two branches are stitched together by channel and outputted through a single Conv module.

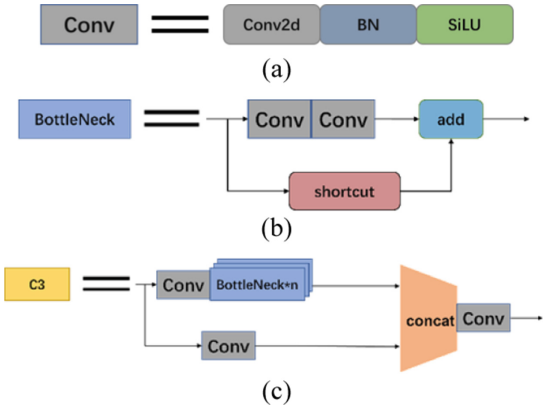

Fig. 2. Key components in the YOLOV5.

Neck's SPP structure is replaced by SPPF. The SPP structure is a spatial pyramid pooling module that can expand the perceptual field, primarily by halving the number of channels through a Conv module, then passing multiple MaxPool of different sizes in parallel, and then splicing the three pooling effects with the input feature map channel by channel, with the number of channels becoming twice the original after merging, in order to maximize the perceptual field at a lower cost. After merging, the maximum number of channels is doubled, which maximizes the perceptual field at a low cost and is utilized to address the multi-scale target problem. The SPPF structure, on the other hand, is serially passed through multiple 5 * 5 MaxPool layers, and while the computational impact and time complexity are the same, the computation speed is accelerated twice under the same situation.

YOLOV5 is a network that caters to a variety of purposes. There are five models to choose from. The fundamental structure of the model content is comparable in YOLOV5n, YOLOV5s, YOLOV5m, YOLOV5l, and YOLOV5x, thanks to the depth multiple and width multiple. Two factors are specified in order to create varied sizes for these five models.

This study provides an object detection neural network for trash classification based on YOLOV5 with optimization and modification, based on the benefits of YOLOV5, such as strong modifiability and quick detection.

3.2 ConvNeXt Block

Since its debut, ViT has swiftly surpassed classic convolutional neural networks as the most sophisticated image classification model. However, typical ViT models have limitations in many computer vision applications, including target identification, semantic segmentation, and object recognition. In March 2021, Swin Transformer [25] developed the notion of stacked Transformer, which included a hierarchical convolutional neural network-like building approach called Hierarchical feature maps, making the backbone helpful for tasks like target identification and instance segmentation. It also makes use of the Shifted Windows Multi-Head Self-Attention idea, which separates the feature map into many discontinuous windows and performs Multi-Head Self-Attention solely in each window, solving the problem of information transmission across nearby windows by employing window offset. On the coco dataset, Swin-Transformer came in No.1 for object identification and instance segmentation at the moment. Why can't conventional CNN accomplish the same impact as ViT by borrowing Transformer model, given that ViT can borrow CNN model-related design and surpass CNN performance? In January 2022 Facebook AI Research and UC Berkeley together presented ConvNeXt [26], a pure convolutional neural network. When the results are compared, ConvNeXt achieves higher accuracy and faster inference than Swin Transformer with the same FLOPs.

The ConvNeXt network model does not provide a significant contribution to innovation, but it does enhance the performance of its own network model by studying ViT's optimization method and the framework it has borrowed from. On ImageNet 1K, ConvNeXt-T accuracy is 0.8% higher than Swin-T at the same FLOPs, while throughput (image/s) is up 47%. To keep the FLOPs stable, the researchers gradually optimize the model design in ConvNeXt in the sequence of macro design, ResNeXt-ify, switching to inverted bottleneck, adopting big kernel size, and micro design.

1). The original ResNet network stacking block number ratio is (3, 4, 6, 3), around 1:1:2:1, in Swin transformer Swin-T block(showed by Fig. 5(a)) stacking ratio is 1:1:3:1, Swin-T stacking ratio is 1:1:9:1. There are two optimization points in macro design, the first modifying stage compute ratio. Because the ratio is 1:1:3:1 and Swin-stacking T's ratio is 1:1:9:1, adjusting the stacking times in ResNet50 to (3, 3, 9, 3) and Swin- T's FLOPs are comparable. The accuracy rate increased from 78.8% to 79.4% following the change. Second, we replace the Stem layer with a convolution size of 7 and a stride of 2 with a convolution operation with a stride of 4 and a size of 4, named "Patchify".

2). The depth-wise convolution used in the lightweight network MobileNet [27] is chosen for convolution selection by borrowing the group convolution grouped convolution from ResNeXt [28] to balance FLOPs and accuracy.
3). The MLP in the Transformer block is similar to the Inverted Bottleneck module of MobileNetV2[29] (thin at both ends and thick in the middle), this structure is because the high-dimensional information will be less lost after passing the ReLU activation function, (a) is the inverted bottleneck module, and (b) is the Inverted Bottleneck module used in ConvNeXt, as shown in Fig. 3 below. The accuracy of smaller models increases from 80.5% to 80.6% with the Inverted Bottleneck module, while bigger models improve from 81.9% to 82.6%.

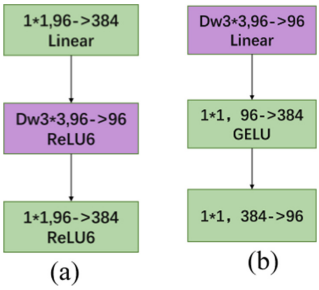

Fig. 3. The schematic of inverted bottleneck designed from MobileNetV2 and ConvNeXt.

4). In general, ViT performs global self-attention, and Swin Transformer also has a 7 * 7 window. As a result, the depth-wise convolution's convolution and size are changed to 7 * 7 and it is moved up, following the example of the MSA module in the transformer, which is put before the MLP module. The precision has increased by 0.7%.
5). Replace ReLU with GELU, use fewer activation functions, use fewer normalization layers, replace BN with LN, and use separate down-sampling layers to improve accuracy by 82%.

In this paper, ConvNext_Block (showed by Fig. 8(b)) is stacked in the ratio of 3:3:9:3 as the backbone of the network model (shown as Fig. 4).

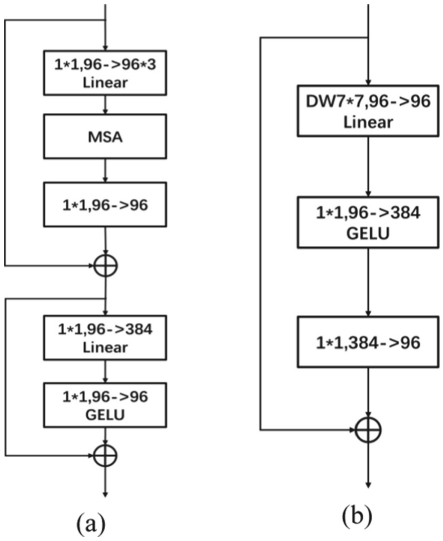

Fig. 4. The figure shows the swin transformer block and ConvNeXt block.

3.3 Ghost Conv

In 2020, Huawei Noah's Ark Lab developed GhostNet [30], a lightweight network model whose major innovation is the suggested Ghost module, which collects redundant information across feature tiers through a low-cost operation. Because the output feature maps of regular convolutional layers typically include a lot of duplicate information and there is no shortage of identical material, there is no reason to create these redundant feature maps with a high number of FLOPs and parameters. As illustrated in Fig. 6(a), some of the output feature maps may be intrinsic feature maps and the other, which is the Ghost feature maps, can be generated by intrinsic feature maps through cheap operations. This is the Ghost module's algorithm design, as seen in Fig. 6. (b). The authors claim that the low-cost operation is "a sequence of linear transformations," which is a linear operation for each channel with a substantially lower computing cost than standard convolution. It can handle a variety of common linear operations including smoothing, blurring, and motion. It can handle a wide range of linear operations, including smoothing, blurring, motion, and so on. However, alternative low-cost linear operations, such as affine transform and wavelet transform, can be studied in the Ghost module to develop the Ghost module.

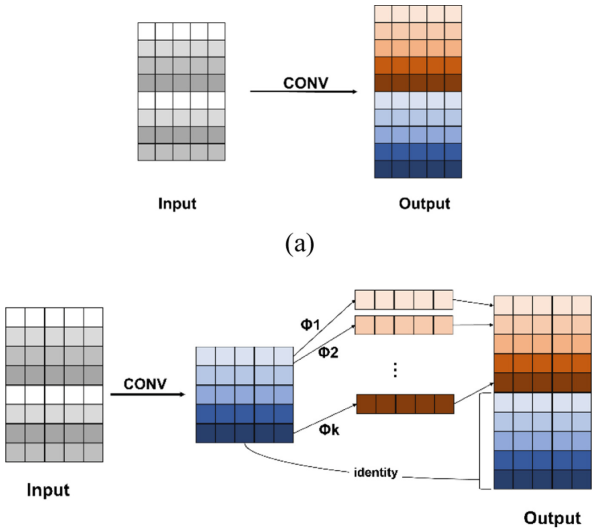

Fig. 5. The figure describes the principle of ghost module implementation.

The number of parameters in typical ordinary convolution is $p_n = k^2 C_{in} C_{out}$, and the Ghost module is $p_g = k^2 C_{in} \frac{C_{out}}{s} + k^2 \frac{c_{out}}{s(s-1)}$, the predicted compression ratio r_c is represented in Eq. (1):

$$r_c = \frac{k^2 \cdot C_{in} \cdot C_{out}}{k^2 \cdot C_{in} \cdot \frac{C_{out}}{s} + k^2 \cdot \frac{C_{out}}{s(s-1)}} \tag{1}$$

since$s << c$, so r_c approximately equal to s. In the case where the input is hwc, the output is $h' \cdot w' \cdot n$, and the size of the convolution kernel is k. Assuming that the linear operation in the Ghost module is deep convolution, the computation of the ordinary convolution is $h' \cdot w' \cdot n \cdot k \cdot k \cdot c$ and the computation of the Ghost module is$h' \cdot w' \cdot \frac{n}{s} \cdot k \cdot k \cdot c + (s-1) \cdot h' \cdot w' \cdot \frac{n}{s} \cdot k \cdot k$,the theoretical speedup ratio is Eq. (2):

$$r_s = \frac{h' \cdot w' \cdot k \cdot k \cdot c}{h' \cdot w' \cdot \frac{n}{s} \cdot k \cdot k \cdot c + (s-1) \cdot h' \cdot w' \cdot \frac{n}{s} \cdot k \cdot k} \tag{2}$$

similarly, r_s is approximate s. Model The parameter compression and operation acceleration ratio are both approximately equal to s (provided that the two convolution kernels k are the same, and k will be slightly less than s if they are not the same). If s = 2, the direct replacement of the Ghost module in the model can theoretically obtain the weight files (weights) and FLOPs of the model directly halved.

4 Experiments and Analysis of Results

4.1 Datasets and Preprocessing

The experimental dataset has six categories, which are bottle, box, can, paper, peel, plastic mainly some common campus garbage, obtained by their own network to find and download, the number of original datasets is 696, and later the dataset is expanded to 3480 by horizontal mirroring, vertical mirroring, brightness, contrast, sharpness, noise, and other data. The distribution of the dataset and the labels are shown in Fig. 7.

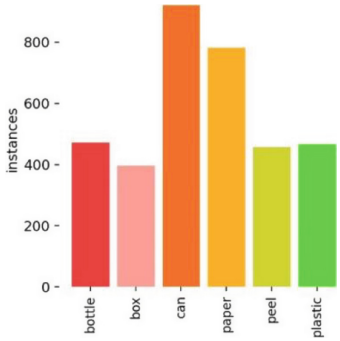

Fig. 6. The figure shows the distribution of the data in our dataset.

Firstly, the dataset label format was transformed by converting PASCAL VOC format to YOLO text format, generating id, x, y, w, h, and normalizing, and then the transformation results were stored in the way of training and validation sets. In the model, data enhancement techniques such as image scrambling, changing brightness, contrast, saturation, hue, adding noise, random scaling, random crop, flip, rotate, random erase, and Mosaic are used. On the other hand, it expands the original data set, prevents overfitting,

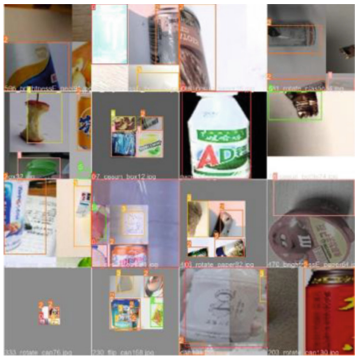

Fig. 7. This is the training image with a batch size of 16. Mosaic data enhancement is performed by selecting four randomly selected images in the sample after stitching and discarding the part that exceeds the input size.

and improves the overall robustness of the model. The effect of the data enhancement is shown in Fig. 8.

4.2 Metrics

In this paper, the mean average precision (mAP) is used as the evaluation index of model detection accuracy, which takes into account the precision (P, the ratio of True Positive in the recognized picture) and recall (R, the ratio of the number of correctly identified positive samples to the number of all positive samples in the test set) of object detection; the number of parameters is used as the evaluation index of model size, and the number of floating point operations (FLOPs) required for one convolution, we adopted the GFLOPs in the paper. In the experiments of this paper, since the mAP between IOU thresholds 0 –0.5 reached 0.995, there was no comparability, so we just use the mAP between IOU thresholds 0.5–0.95 as the evaluation metrics named mAP@0.5:0.95.IOU loss is a response to the detection effect of the prediction box and the real box. As inadequate consideration of the distance between different detection boxes, multiple detection boxes are overlapping but the IOU is the same. GIOU loss solves the problem that the gradient cannot be calculated when IOU is used as a loss function and adds the minimum outsourcing box as a penalty term, the calculation formula is (3):

$$GIOU = \begin{cases} IOU - \frac{C-(A\cup B)}{C}(IOU \neq 0) \\ -1 + \frac{(A\cup B)}{C}(IOU = 0) \end{cases} \tag{3}$$

4.3 Result

In this paper, we use PyTorch, a deep learning framework, to build an experimental environment to study household garbage images, with Intel(R) Xeon(R) Gold 6330 14-core CPU, RTX3090 GPU, PyTorch = 3.8, CUDA = 11.1. The model is trained for 300 epochs.

In the experiment, which use the YOLOV5 as the baseline, the models YOLOV5-nano, YOLOV5-small, YOLOV5_ConvNeXt-Tiny, YOLOV5_ConvNeXt-Small, YOLOV5-transformer and YOLOV5_CG proposed in this paper are learned for six types of garbage, and then all experimental results are compared according to the metrics. Among them, YOLOV5-nano and YOLOV5-small are the relatively small models originally proposed by YOLOV5. YOLOV5_ConvNeXt-Tiny is an optimized YOLOV5 built with the tiny version of ConvNeXt as Backbone, similarly, and YOLOV5_ConvNeXt-Small is an optimized YOLOV5 with the small version of ConvNeXt as the backbone. YOLOV5-transformer is YOLOV5 modified with transformer as backbone. the results of the Comparison experiment are shown in Table 1.

It can be seen from Table 1 that although the precision of YOLOV5_ConvNeXt-S is 0.9872 at the highest and GIOU is 0.1449 at the lowest, the recall is 0.9167 at the lowest mAP@0.5:0.95, the lowest result. The main reason for this is that due to the small amount of data and the large YOLOV5_ConvNeXt-S network, so that the result does not meet expectations. The YOLOV5-CG proposed in this paper is based on YOLOV5s, mAP@0.5:0.95 increased by 3%, recall reached 1, and GIOU decreased by about 0.04.

Table 1. Experimental results of YOLOV5-CG network and related comparison networks

	mAP0:0.95	Precision	Recall	GIoU
YOLOV5n	0.8396	0.9829	0.9293	0.02047
YOLOV5s	0.8553	0.9682	0.9328	0.02112
YOLOV5_ConvNeXt_T	0.8664	0.9617	0.9233	0.01693
YOLOV5_ConvNeXt_S	0.8285	**0.9872**	0.9167	**0.01449**
YOLOV5-transformer	0.83	0.9211	0.9564	0.02271
YOLOV5-CG(**our**)	**0.8813**	0.9703	**1**	0.0167

Also in the ablation experiments in Table 2, it can be seen that the ConvNeXt block has a significant improvement on the average accuracy of the network model, while Ghost Conv also proves that the cheap operation has a significant effect on reducing the model parameters and lowering the computational complexity.

Table 2. Results of ablation experimental.

	ConvNeXt block	Ghost conv	YOLOV5-CG
Param	128824011	5685395	113844555
GLOPs	89.9	13.4	78.7
mAP@0.5:0.95	0.8664	0.8139	0.8813

The YOLOV5-CG network proposed in this paper reduces, compared to ConvNeXt-Tiny and ConvNeXt-Small, the number of parameters by about 12%, FLOPs by 13%, and the average accuracy is improved. Compared to YOLOV5 with Transformer as the

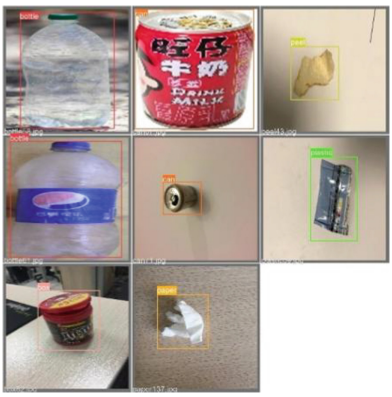

Fig. 8. The figure shows the validation results by our module net.

backbone, the comparison results are not good from all aspects, especially when compared with baseline, the results are reduced, and it verifies that the original transformer proposed in **3.2** is not suitable to be involved in computer vision tasks. YOLOV5-CG was tested on the images in the validation set, and the classification detection graph obtained is shown in Fig. 8.

5 Conclusion

The YOLOV5-CG network proposed in this paper is mainly applied to the object detection task in computer vision to realize garbage classification and detection in the context of school life. Based on the existing object detection network YOLOV5, the classification accuracy is improved through comparison and optimization, and the practicality of the network in realizing garbage classification is verified through experiments.

The design of the network algorithm is based on the development needs of the country and the city and follows the trend of smart city development. In future work, the network generalization capability is improved by acquiring more data sets in order to achieve the demand of garbage classification at hand in more conditions.

References

1. 邓晓妮 et al.大学校园垃圾分类现状调查与研究[J]. 绿色科技**23**(20), 172–174 (2021)
2. Fucong, L., et al.: Depth-wise separable convolution attention module for garbage image classification. Sustainability **14**(5), 3099 (2022)
3. Longyu, G., et al.: A design of intelligent public trash can based on machine vision and auxiliary sensors. J. Robot. Netw. Artif. Life **8**(4), 273–277 (2021)
4. Zhang, H., Song, A.: Research on image classification of recyclable garbage based on transfer learning. Int. Core J. Eng. **7**(6), 153–157 (2021)
5. Hongjie, D., et al.: An embeddable algorithm for automatic garbage detection based on complex marine environment. Sensors **21**(19), 6391 (2021)
6. Krizhevsky, A., Sutskever, I., Hinton, G.: ImageNet classification with deep convolutional neural networks. Adv. Neural Inf. Process. Syst. **25**(2), 1097–1105 (2012)
7. Szegedy, C., et al: Going Deeper with Convolutions. CoRR, abs/1409.4842 (2014)
8. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. CoRR, abs/1409.1556 (2014)
9. He, K., Zhang, X., Ren, S., et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
10. Vaswani, A., Shazeer, N., Parmar, N., et al.: Attention Is All You Need. arXiv: 1706.03762 (2017)
11. Ross, B.G., et al.: Rich feature hierarchies for accurate object detection and semantic segmentation. CoRR, abs/1311.2524 (2013)
12. Ross B. Girshick. Fast R-CNN. CoRR, abs/1504.08083 (2015)
13. Shaoqing, R., et al.: Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1137–1149 (2017)
14. He, K., Gkioxari, G., Dollár, P., et al.: Mask R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2961–2969 (2017)
15. Lin, T.Y., Dollár, P., Girshick, R., et al.: Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2117–2125 (2017)

16. Zhao, Q., Sheng, T., Wang, Y., et al.: M2det: A single-shot object detector based on multi-level feature pyramid network. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, no. 01, pp. 9259–9266 (019)
17. Redmon, J., Divvala, S., Girshick, R., et al.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788 (2016)
18. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7263–7271 (2017)
19. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. arXiv preprint, arXiv:1804.02767 (2018)
20. Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: Yolov4: Optimal speed and accuracy of object detection. arXiv preprint, arXiv:2004.10934 (2020)
21. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C.: SSD: Single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
22. Fu, C.Y., Liu, W., Ranga, A., et al.: Dssd: Deconvolutional single shot detector. arXiv pre-print, arXiv:1701.06659 (2017)
23. Liu, S., Huang, D., Wang, Y.: Receptive field block net for accurate and fast object detection. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11215, pp. 404–419. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01252-6_24
24. Ultralytics/yolov5. https://github.com/ultralytics/yolov5. Accessed 21 Apr 2022
25. Liu, Z., Lin, Y., Cao, Y., et al.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10012–10022 (2021)
26. Liu, Z., Mao, H., et al.: A ConvNet for the 2020s. arXiv preprint, arXiv:2201.03545 (2022)
27. Howard, A.G., Zhu, M., Chen, B., et al.: Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint, arXiv:1704.04861 (2017)
28. Xie, S., Girshick, R., Dollár, P., et al.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1492–1500 (2017)
29. Sandler, M., Howard, A., Zhu, M., et al.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4510–4520 (2018)
30. Han, K., Wang, Y., Tian, Q., et al.: Ghostnet: more features from cheap operations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1580–1589 (2020)