Small Visual Object Detection in Smart Waste Classification Using Transformers with Deep Learning

Jianchun Qi, Minh Nguyen, Wei Qi Yan
Auckland University of Technology, New Zealand

Abstract. Smart object waste classification is relatively essential for protecting the environment and saving resources. This is considered a vital pathway towards sustainability. In waste classification, we see that it is challenging to detect waste of small visual objects with low resolutions that directly affect the overall performance of waste classification. While current visual object detection algorithms focus on the exploration of larger objects, the development of small object detection is being expanded relatively slowly due to the inability to acquire more visual information. In this paper, we propose a novel method combining contextual information and multiscale learning to improve small object detection performance in waste classification by enabling small object detection to obtain more feature information at high resolution. Furthermore, based on the advantages of parallel computing in Transformers, we utilize the DETR model to explore our method. The experimental results show that our method achieves high accuracy in the detection of a small object in waste.

Keywords: Small object detection · Transformers · Waste detection · Waste classification

1 Introduction

Waste classification generally refers to the conversion of waste into a public resource by classifying wastes into storage and transportation according to classification standards. The purpose is to increase the economic and resource value of waste, promote the recycling of resources, reduce the cost of waste disposal and the consumption of land resources so as to protect our environment. Besides, conventional waste disposal, such as landfilling and stacking, may produce harmful chemicals, which contaminate soil and groundwater resources and lead to reduced crop yields [4]. Therefore, it is necessary to develop efficient and accurate waste classification methods.

The development of computer vision has made pattern classification and visual object detection easy. Visual object detection occupies a vital position in the research field of computer vision [10, 20, 27, 33]. It can solve problems such as pedestrian tracking, visual object segmentation, and smart driving, etc. By applying deep learning [22, 23, 29, 32] to waste classification, exploring automated and efficient waste classification methods has ecological, social, and economic significance. Meanwhile,

we find that the accuracy of detecting small waste objects could be improved, such as broken nut shells and button batteries. These objects are small in size compared to plastic bottles and carton boxes. If they are detected in an image, a fewer of pixels are occupied in the image than other objects. This makes the waste classification task as a challenging problem.

Similarly, small visual object detection is also abundant and broadly applied to ordinary life, such as traffic sign detection in automated driving, etc. Small object detection has always been a challenging task in visual object detection, because the visual features of small objects need to be accurately detected. For example, a small object may have less than 32×32 pixels while a standard image resolution is 1024×1024.

In recent years, the performance of small object detection has also gradually improved [11, 13], but the performance is still inferior to that of large objects. The feature maps of small objects do not have high resolution, resulting in less visual information to be detected by deep neural networks. Currently, too many downsampling operations and too big receptive field are all the factors that could affect small object detection. Furthermore, solving the problem of small object detection also requires both shallow representational information and deep semantic information.

To sum up, we make use of both context learning [19] and multiscale learning [34] to improve the small object detection performance in waste classification. Also, owing to the advantage that Transformer models can be computed in parallel, there are fewer studies on small object detection based on Transformer, we choose to study the detection of small objects in waste using Transformer. In this paper, we choose the DETR model. The main contributions of this paper are as follows:

(1) Transformers for detecting small objects of waste are trained, the results are high in accuracy.

(2) Multiscale learning and context learning are combined together for the improvement of small object detection for waste classification.

(3) A dataset including the small waste object is created.

In this paper, our related work is presented in Section 2, while Section 3 shows the methods, after the results is stated in Section 4. Finally, Section 5 contains our conclusions.

2 Related Work

2.1 Small Object Detection

Currently, the detection of small visual objects is significantly different from that of large objects, in many cases, it has only half size of large objects. However, small object detection has important research significance. For example, in autonomous vehicles, it

is important to accurately detect small visual objects that can trigger traffic accidents to preserve the road safety. There are a slew of solutions for the shortcomings of small object detection as follows.

2.1.1 Data Augmentation

Data augmentation is a simple and effective method to improve the performance of small object detection. It enhances the generalization ability and robustness of the model by expanding the size of small object samples. In recent years, a plenty of data enhancement methods for regular object detection have been broadly employed, such as random cropping [12], translation [31], adjusting image saturation [17, 21], and mosaic enhancement.

Similarly, data augmentation methods for small object detection have also emerged. For example, the number of small objects is increased by repeatedly copying and pasting the small objects in the image to improve the model performance [11]. There is also an adaptive learning method proposed to enhance the performance of the small object detection. The data augmentation has solved the problems of small number of samples and lack of features in small object detection which improved the generalization ability of the model.

2.1.2 Contextual Information

Contextual information can improve the performance of small object detection because there is a group of informational correlations between the object and the background. For example, while a small object is flying in sky, we may not be able to see exactly what the object is, but with the background of the sky and the size of the object, we will associate a bird flying over our heads. Using this informational correlation will assist us to improve the detection of small visual objects.

Currently, a spate of studies explored and exploited this research issue. A method based on contextual feature enhancement is proposed [14], which firstly generates image proposal regions, and then produces multiscale windows around the targets for object feature enhancement. There are also recurrent neural networks proposed to encode and concatenate contextual information. However, though these methods improved the performance of small object detection, they are still affected by the size of the receptive field, resulting in partial loss of contextual information.

2.1.3 Multiscale Learning

Multiscale learning allows small visual object detection to take into account in both the need for representing shallow information and deep semantic information, avoiding the

loss of location and feature information of small objects as the depth of the network increments. There are various ideas for using multiscale detection. For example, using dilated convolution to obtain various receptive field sizes, image pyramids [6], multiscale object detection [15], deconvolution layers [2], and feature pyramids [18]. These methods improve the resolution of small object feature maps, but some of them also have the problems with too much computational cost.

Overall, multiscale learning can effectively betterment the performance of small object detection, but the huge computational costs and unstable feature fusion process are also the reasons that hinder the further development of multiscale learning.

2.2 Visual Object Detection

2.2.1 Convolutional Neural Network

Convolutional neural networks (CNN) can be trained using the corresponding feature maps from a large number of visual object samples and reduce the complexity of the model by using downsampling, weight sharing, and local receptive fields [17]. At present, the existing CNN models applied to object detection can be classified into two categories: One-stage network and two-stage network.

One-stage network. It directly returns the class and position information of visual object through backbone network without using Region Proposal Network (RPN), which is fast in visual object detection but low in accuracy [26]. At present, the classical one-stage object detection networks are YOLO [24, 25], YOLOv4 [1], and Single Shot MultiBox Detector (SSD) [15].

Two-stage network. It mainly extracts features through convolutional neural networks, trains the RPN network, then conducts fine-tuning the network with the proposal regions, which has high accuracy but lower detection speed than one-stage models. Currently, classical algorithms include Region-CNN (R-CNN) [8], Faster R-CNN [18], and Mask R-CNN [9].

2.2.2 Transformer Models

In recent years, Transformer models [28] have become popular. Compared with CNNs, Transformers have better computational complexity and solve the problem of time consumption. With the advancement of Transformers in the field of Natural Language Processing (NLP), Transformers applied in the computer vision field are also emerging.

Vision Transformer [7], which was applied to pattern classification, cuts the 3D data of an image into patches, arranges them in sequence, converts them into serialized data, and uses the Transformer model for processing. Similarly, there is also a Transformer model applied to visual object detection, namely DETR [3]. It firstly extracts feature

maps by using CNN to form a patch sequence, then Robject queries and a new loss function are formed. The model is very succinct and concise.

3 Our Method

In this paper, the proposed model structure is shown in Fig. 1. The detection of small visual objects is often difficult, visual features extracted from the proposed regions have weak discriminative ability. After considering multiple methods for small object detection [12, 14, 16], inspired by R-CNN [8] and context-based small object detection methods [5, 34], based on the DETR model [3], we introduce a new combination of neural networks, which can provide important feature maps for small object detection. We input the target image and its context into different neural networks through the target channel and context channel, respectively; we aggregate the contextual information for fusion, and input the obtained visual features into the Transformer [28] encoder and decoder. In this case, the context channel is modified according to the CAB model [5]. The model is mainly split into three stages, its details are represented in the following sections. It is worth noting that our proposed method can also be applied in other detectors such as Faster R-CNN.

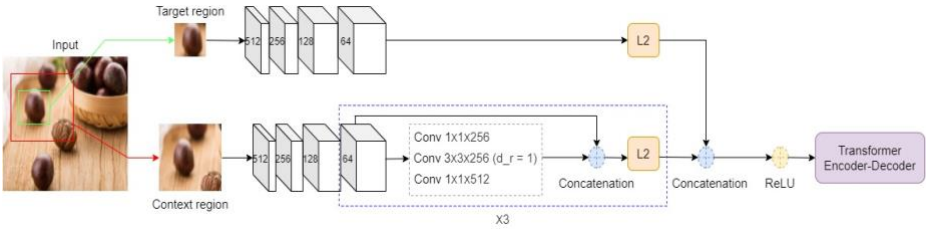

Fig. 1. The proposed model structure.

Backbone network

In our experiments, we are use of ResNet-50 as the backbone network to extract feature information from the images. Firstly, we adjust the input image to 512×512. After that, we extract the visual features by using downsampling. Following this step, we assign the model settings of ResNet-50 for the experiments. The target channel and contextual channel parameters are consistent that have the same structure. In this stage, we only kept the four layers: Conv1, Conv2, Conv3, and Conv4, in the ResNet-50 net as shown in Fig.1. Retaining the shallow feature map not only reduces the loss of small object features but also preserves the receptive field of small object detection and strengthens the accuracy of border regression [30].

Target channel. We firstly crop the proposed region in the image as the input of the target channel. As known from the backbone network part, after input the target image

region, it is convolved by Conv1, Conv2, Conv3, and Conv4. Hence, we add an L_2 normalization layer.

Contextual channel. The structure of the context channel is different from the target channel as shown in Fig. 1. At the first step, we crop the contextual region with the proposed region in the image as the input of the context channel. Again, as shown in the backbone network section, we keep the first four convolutional layers. After that, in this channel, we take use of multiple stacked dilated convolution layers, which is consistent with CAB model [5], we expand the convolution kernel by adding 0 to the convolution kernel to achieve the goal of expanding the receptive field and obtain multiscale contextual information without losing resolution [34].

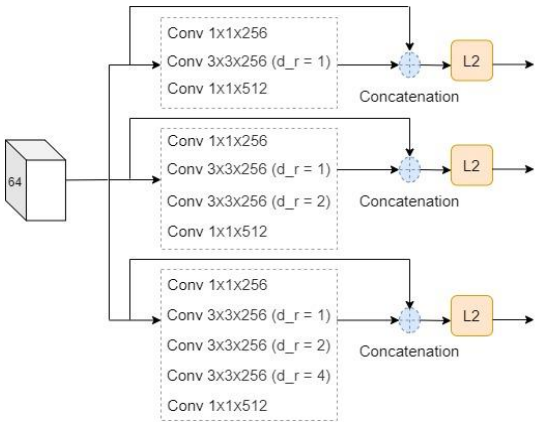

Fig. 2. The architecture of context channel.

Specifically, a 1×1 convolution layer is added, after multiple stacked dilated convolution layers are employed to obtain more contextual information. This reduces the computational effort by not introducing additional parameters. Besides, regarding the dilated convolution, we choose three parallel stacked dilated convolutions, each stacked dilated convolution has an increasing dilation rate as shown in Fig. 2. Thus, more contextual information from different angles can be obtained. Then, we concatenate each input layer with its corresponding dilated convolution features for optimization. Finally, after each output is added to an L2 normalization, the three outputs are concatenated. In this model, the dilated convolution is calculated as Eq. (1).

$$f_k = \mathrm{dr} \times (\mathrm{k} - 1) + 1 \tag{1}$$

where f_k is the convolution kernel size after expansion, *dr* is the expansion coefficient, and *k* is the convolution kernel size. In this model, *n* is the number of convolutions, we set the selection rule of *dr* as 1, 2, and 4, that means, the computational complexity is

1+n(n-1)/2. The size of receptive field is determined by the kernel size and stride size. Therefore, the computational method of the receptive field is shown in Eq. (2).

$$RF_{n+1} = RF_n + (f_k - 1) \times \prod_{i=1}^{n} s_i \quad (2)$$

where RF_n is the size of the receptive field corresponding to the n-th convolutional layer, s_i is the stride size of layer i. Because the stride length of the convolution kernel represents the extraction accuracy, we set the stride size as 1.0 to avoid losing the information of the original image [5].

If dr=1, the convolution kernel size is 3×3. If dr=2, the size of the convolution kernel after adding the hole is 5×5, the receptive field is 7×7. If dr=4, the convolution kernel size is 9×9, the receptive field increases to 15×15. Although the output dimensions of all dilated convolutions are the same, we see that the receptive fields are distinct. It is also worth noting that the parameter quantities do not change after dilated convolution. Increasing the receptive field does not group the size of the convolution kernel, even in multiple stacked dilated convolutions [34].

Finally, the output of the target channel after L_2 normalization is concatenated with the three outputs of the contextual channel, then a layer of ReLU is added to feed the result into the encoder-decoder structure of the Transformer model, so that the visual feature can show a better balance between semantic and spatial aspects, and achieve the ideal combination of multiscale learning and contextual learning.

Transformer Detection

In DETR [3], Transformer and feedforward network (FFN) are combined to form the net architecture for visual object detection. Regarding the Transformer, its structure is almost identical to the original one of encoder-decoder architecture. The encoder consists of a multi-head self-attention and an FFN with the addition of positional encoding to obtain the attention results of each target. While the decoder retains the original multi-head self-attention, multi-head attention, and FFN, we decode the targets in parallel and queries these targets together. The results are fed into a fixed number of FFNs in the form of embedding [28]. Finally, the predicted classification and bounding box corresponding to each target are obtained by parallel calculation. In our model, finding the optimal bipartite matching [3] is also employed to determine the bounding box of each object, as shown in Eq. (3).

$$\hat{\sigma} = arg \min_{\sigma \epsilon \Subset N} \sum i^N \pounds \text{match}\big(yi, \hat{y}\sigma(i)\big) \quad (3)$$

where N is the number of predictions, y is the ground truth set, and $\hat{y}$ is the set of predictions, $\hat{y}$ contains the predicted category and bounding box. The loss between each

y and $\hat{y}$ is L_i. Therefore, Eq. (3) finds a permutation that can map the predicted indices to the indices of the ground truth, avoiding getting the same loss in different ranking predictions. Besides, regarding the calculation of L_i, we adopt the same method as DETR, which is a linear combination of L_1 loss and GIOU loss [29] for ground truth and predicted values.

4 Result Analysis

The model takes use of AdamW optimizer with an initial learning rate of 10^{-4}. The backbone network has a learning rate of 10^{-5} and a weight decay of 10^{-4}. Additionally, a dropout of 0.1 was adopted, 300 epochs were selected for model training.

4.1 Our Dataset

In this paper, we collected the waste dataset with 1, 053 images of small objects, including batteries, fruit cores, nut shells, egg shells, and bottle caps. Furthermore, we selected small waste objects with an area of less than 32×32 pixels in the image.

Table 1. The number of samples of each class

Classes	Numbers of samples
Battery	202
Egg shell	221
Bottle cap	220
Nut shell	207
Fruit core	203
Total	1,053

Fig. 3. The samples in the waste dataset. The images, (a), (b), (c), (d), and (e) show battery, egg shell, bottle cap, nutshell, and fruit core, respectively.

We merge these five types of wastes into four classes according to the waste classification criteria, i.e., batteries belong to the class “Hazardous”, fruit cores and egg shells are “Wet” class, the nut shells are classified into “Dry” class, and the bottle caps

are the “Recyclable” class. Fig. 3 shows the small waste images in the dataset and Table 1 illustrates the number of samples of each class.

4.2 Evaluation Methods

In order to verify the performance of the model, we take use of a series of evaluation metrics: Average Precision (AP) and Mean Average Precision (mAP). The range of thresholds is [0.5: 0.05: 0.95]. In addition, we also adopted the Precision-Recall curve (PR curve) for the performance evaluations.

4.3 Result Analysis

Fig. 4 shows us an example of small waste object detection. We see that there are various classes of waste samples in the image, each class has a color bounding box and label.

Fig. 4. Visual object detection results (a) the results of classifying batteries, which belong to “Hazardous” class, (b) the classification results of bottle cap and battery, which are from “Recyclable” class and “Hazardous” class (c) the classification results of fruit core, which belong to “Wet” class.

In Fig. 5, we see the PR curves of three small object waste classifications by using multiple models. AP values are calculated by calculating the area under the curve. In Fig.5(a), AP values of the battery of our proposed model, DETR, and Faster R-CNN are 12%, 9%, and 8%, respectively. For bottle cap, its AP value is the highest, reaching about 28% in our proposed model, 23% and 21% in DETR and Faster R-CNN,

respectively. Finally, in Fig.5(c), the AP values of fruit core of our model, DETR and Faster R-CNN are 11%, 10%, and 7%, respectively.

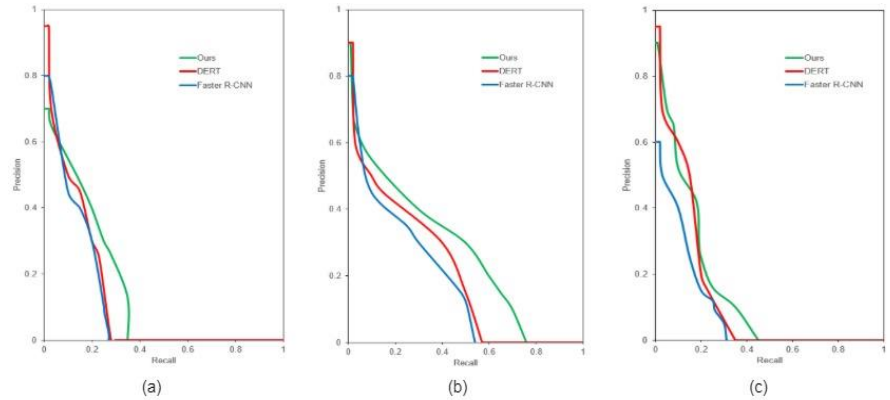

Fig. 5. PR curves of three small object classifications, comparing our model to other two advanced models (a) the PR curve of battery (b) the PR curve of bottle cap (c) the PR curve of fruit core.

We quantitatively compare our model with other models by using our own dataset. Table 2 shows the comparison results. The mAP of DETR is 28.8%, which is slightly lower than that of our model by 0.9%. Then, the mAP of Faster R-CNN (ResNet-50) and Mask R-CNN is 20.5% and 26.2%, respectively. Finally, SSD has 23.7% mAP values, which is 5.1% higher than Faster R-CNN (VGG16). It is evident that our model is more vibrant for small waste objects.

Table 2. Mean average precision results between five models

Models	Backbone	mAP (%) (Small)
Faster R-CNN	VGG16	18.6
Faster R-CNN	ResNet-50	20.5
Mask R-CNN	ResNet-101	26.2
SSD	VGG16	23.7
Mask R-CNN	Swin Transformer	27.8
DETR	ResNet-50	28.3
Ours	ResNet-50	29.2

Afterwards, Table 3 shows the comparisons for all waste classes. The AP of egg shell and nut shell is higher than that of other classes. Meanwhile, the fruit core has almost the lowest AP value among the five classes.

4.4 Ablation Experiments

Pertaining to the overall performance of these models, the components were employed to explore the model and facilitate a better understanding of the model. According to the characteristics of our proposed model, we choose to conduct comprehensive ablation experiments on the model through four aspects.

Table 3. Average precision results between five models for each class

Models	Backbones	Battery (%)	Bottle Cap (%)	Fruit Core (%)	Egg Shell (%)	Nut Shell (%)
Faster R-CNN	VGG16	6.9	19.6	6.3	30.6	29.7
Faster R-CNN	ResNet-50	8.1	21.4	7.2	31.6	34.7
Mask R-CNN	ResNet-101	11.2	24.9	9.6	43.3	42.2
SSD	VGG16	9.8	22.5	9.7	37.6	39.1
Mask R-CNN	Swin Transformer	11.5	25.4	9.8	47.6	44.9
DETR	ResNet-50	9.6	23.7	10.3	49.4	48.8
Ours	ResNet-50	12.1	28.0	11.6	45.1	49.6

4.4.1 Number of Channels

Our model has a target channel and a contextual channel. Firstly, we cut off the target channel and make use of only one input image for visual object detection, as shown in Table 4. If only the target channel is kept, it is impossible to perform detection better. After keeping the context channel, though the method of context information cannot be used, the mAP of the multiscale learning of the receptive field reaches 23.4%, which is 5.8% lower than the way of retaining both. The speed also decreased from 3.1 to 1.6 FPS, indicating that the effect of the target channel on the model is also present. Furthermore, this brings us to a future direction on how to make the model improve FPS with guaranteed accuracy.

Table 4. Influence of target channel on mAP and FPS

Target Channel	Context Channel	mAP % (Small)	Speed (FPS)
√		—	—
	√	23.4	1.6
√	√	29.2	3.1

4.4.2 Number of Convolutional Layers

In the paper, Conv1, Conv2, Conv3, and Conv4 layers are preserved. Therefore, in our ablation experiments, we keep Conv2, Conv3, and Conv4 layers, respectively, the results are shown in Table 5. By retaining the feature map of Conv3 for detection, the FPS is only 1.0 FPS, while using Conv4 normally, the model will get a speed of 3.1

FPS. Regarding mAP, the results are also 7.3% lower (from 29.2% to 21.9%), with only the convolutional layers retained to Conv3 than with Conv4.

Table 5. Influence of convolutional layers on mAP and FPS

Layer	mAP % (Small)	Speed (FPS)
Conv2	—	—
Conv3	21.9	1.0
Conv4	29.2	3.1
Conv5	28.6	2.7
Conv6	26.3	2.4

4.4.3 Application of Dilated Convolutions

In our model, we employ the dilated convolution in the context channel. Therefore, we also applied the dilated convolution in the target channel as well by conducting ablation experiments. The experimental results are shown in Table 6. We see that the mAP value of using the dilated convolution in both channels is only 0.2% lower than that of the original model. This shows the importance of the dilated convolution to the model. But on the contrary, the speed is only 0.8 FPS.

Table 6. Influence of the application of dilated convolutions on mAP and FPS

Dilated Convolution	mAP % (Small)	Speed (FPS)
In Context Channel	29.2	3.1
In Both Channels	29.0	0.8

4.4.4 Number of Dilated Convolutions

In the model, we applied three dilated convolutions. Therefore, we increase the number of dilated convolutions to evaluate the model performance. The number of dilated convolutions is denoted as d_c, the experimental results are shown in Table 7.

Table 7. Influence of the number of dilated convolutions on mAP values

Num(d_c)	mAP % (Small)	Speed (FPS)
2	22.7	3.8
3	29.2	3.1
4	30.0	1.9
5	30.2	1.8

We see that the mAP value increases with the increase of Num(d), from 22.7% to 30.0%. However, if four dilated convolutions are employed, the FPS value is only 1.9.

Since the mAP at Num(d) of 4 is only 0.8% more than that at Num(d) of 3, on balance, we choose to set Num(d) as 4.

5 Conclusion

In this paper, we improve the performance of Transformer models for small visual object detection in waste classification with the DETR model. One is to expand the receptive field to obtain more feature information for small waste object detection whilst ensuring high resolution. Secondly, the contextual information of small objects is enhanced by extracting target regions and contextual regions. The experimental results show that the proposed model achieves small object classification for the wastes. In future, we will improve the model in three directions: Improving FPS, simplifying the model and reducing the computation, and expanding the small waste object dataset.

References

1. Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M.: YOLOv4: Optimal speed and accuracy of object detection. arXiv, (2020).
2. Cai, Z., Fan, Q., Feris, R. S., Vasconcelos, N.: A unified multiscale deep convolutional neural network for fast object detection. ECCV, pp. 354-370 (2016).
3. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. ECCV, pp. 213-229 (2020).
4. Chen, S.S., Huang, J.L., Xiao, T.T., Gao, J., Bai, J.F., Luo, W., Dong, B.: Carbon emissions under different domestic waste treatment modes induced by garbage classification: Case study in pilot communities in Shanghai, China. Science of the Total Environment. 717, 137193 (2020).
5. Cui, L., Lv, P., Jiang, X., Gao, Z., Zhou, B., Zhang, L., Shao, L., Xu, M.: Context-aware block net for small object detection. IEEE Transactions on Cybernetics, pp. 2300-2313 (2020).
6. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. IEEE CVPR, pp. 886-893 (2005).
7. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X.H., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv (2020).
8. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. IEEE CVPR, pp. 580-587 (2014).
9. He, K.M., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. IEEE ICCV, pp. 2961-2969 (2017).
10. He, K.M., Zhang, X.Y., Ren, S.Q., Sun, J.: Deep residual learning for image recognition. IEEE CVPR, pp. 770-778 (2016).
11. Kisantal, M., Wojna, Z., Murawski, J., Naruniec, J., Cho, K.: Augmentation for small object detection. arXiv, (2019).
12. Krizhevsky, A., Sutskever, I., Hinton, G. E.: ImageNet classification with deep convolutional neural networks. NIPS, pp. 1-9 (2012).

13.Li, J., Wei, Y., Liang, X., Dong, J., Xu, T., Feng, J., Yan, S.: Attentive contexts for object detection. IEEE Transactions on Multimedia, pp. 944-954 (2016). https://doi.org/10.1109/TMM.2016.2642789
14.Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., Berg, A. C. SSD: Single shot multibox detector. ECCV, pp. 21-37 (2016).
15.Liu, Z., Mao, H.Z., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie. S.N.: A ConvNet for the 2020s. arXiv, (2022).
16.Li, Z., Zhou, F.: FSSD: Feature fusion single shot multibox detector. arXiv:1712.00960 (2017).
17.Luo, Z., Nguyen, M., Yan, W.: Sailboat detection based on automated search attention mechanism and deep learning models. IEEE IVCNZ (2021).
18.Nie, Z.F., Duan, W.J., Li, X.D.: Domestic garbage recognition and detection based on Faster R-CNN. Journal of Physics: Conference Series (2021).
19.Oliva, A., Torralba, A.: The role of context in object recognition. Trends in Cognitive Sciences, pp. 520-527 (2017).
20.Pan, C. Yan, W.: A learning-based positive feedback in salient object detection. IEEE IVCNZ (2018)
21.Pan, C. Yan, W.: Object detection based on saturation of visual perception. Multimedia Tools and Applications 79 (27-28), 19925-19944 (2020).
22.Pan, C., Liu, J., Yan, W., Zhou, Y. Salient object detection based on visual perceptual saturation and two-stream hybrid networks. IEEE Transactions on Image Processing (2021).
23.Qi, J., Nguyen, M., Yan, W.: Waste classification from digital images using ConvNeXt. PSIVT (2022).
24.Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You Only Look Once: Unified, real-time object detection. IEEE CVPR, pp. 779-788 (2016).
25.Redmon, J., Farhadi, A.: YOLO9000: Better, faster, stronger. IEEE CVPR, pp. 7263-7271 (2017).
26.Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S.: Generalized intersection over union: A metric and a loss for bounding box regression. IEEE CVPR, pp. 658-666 (2019).
27.Shen, D., Xin, C., Nguyen, M., Yan, W.: Flame detection using deep learning. ICCAR (2018).
28.Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. NIPS, (2019).
29.Wan, L., Zeiler, M., Zhang, S., Le Cun, Y., Fergus, R.: Regularization of neural networks using DropConnect. ICML, pp. 1058-1066 (2013).
30.Yin, X., Goudriaan, J. A. N., Lantinga, E. A., Vos, J. A. N., & Spiertz, H. J.: A flexible sigmoid function of determinate growth. Annals of Botany, 91, 361-371 (2002).
31.Xiao, B.J., Minh N., Yan, W.Q.: Apple ripeness identification using deep learning. International Symposium on Geometry and Vision, 1386, 53 (2021).
32.Yan, W.Q.: Computational Methods for Deep Learning - Theoretic. Practice and Applications. Springer, Heidelberg (2021).
33.Yan, W.Q.: Introduction to Intelligent Surveillance - Surveillance Data Capture, Transmission, and Analytics, 3rd edn. Springer, Heidelberg (2019).
34.Yu, F., Koltun, V. Multiscale context aggregation by dilated convolutions. ICLR, (2016).