Research Article

Automatic Garbage Scattered Area Detection with Data Augmentation and Transfer Learning in SUAV Low-Altitude Remote Sensing Images

Tengfei You,[1,2] **Weiyang Chen** [1,2] **Haifeng Wang,**[1,2] **Yang Yang** [1,2] **and Xinang Liu** [2,3]

[1]*School of Information Science and Technology, Yunnan Normal University, Kunming 650500, China*
[2]*The Laboratory of Pattern Recognition and Artificial Intelligence, Yunnan Normal University, Kunming 650500, China*
[3]*Asset Management, First Affiliated Hospital of Kunming Medical University, Kunming 650032, China*

Correspondence should be addressed to Weiyang Chen; cwy_work@126.com, Yang Yang; yyang_ynu@163.com, and Xinang Liu; dianyunshanfang@sina.com

Cleaning up the garbage timely plays an important role in protecting the ecological environment of nature reserves. The traditional approach adopts manual patrol and centralized cleaning to clean up garbage, which is inefficient. In order to protect the ecological environment of nature reserves, this paper proposes an automatic garbage scattered area detection (GSAD) model based on the state-of-the-art deep learning EfficientDet method, transfer learning, data augmentation, and image blocking. The main contributions of this paper are (1) we build a garbage sample dataset based on small unmanned aerial vehicle (SUAV) low-altitude remote sensing and (2) we propose a novel data augmentation approach based on garbage scattered area detection and (3) this paper establishes a model (GSAD) for garbage scattered area detection based on data augmentation, transfer learning, and image blocking and gives future research directions. Experimental results show that the GSAD model can achieve the F1-score of 95.11% and average detection time of 1.096 s.

1. Introduction

Many nature reserves and scenic spots have been shut down due to the destruction of the ecological environment. Take 2018 as an example: the Everest administration of the Tibet Autonomous Region issued an announcement prohibiting any individuals and units from entering the core area above the Rongbu Temple, and Boracay in the Philippines was shut down for half a year. This year, Nianbaoyuze National Geopark, Zhaling-Eling Lake, and Xingxinghai reserves in China were also shut down. Among the many shutdown factors, secondary pollution caused by the inability to clean up garbage in time is considered to be the primary factor, which destroys water sources, vegetation, soil, glaciers, etc. Therefore, it is necessary to propose an automatic garbage scattered area detection approach to improve the cleaning efficiency and reduce secondary pollution.

Object detection is a long-term hot issue in the field of computer vision. Currently, object detection algorithms can be divided into two classes: based on traditional computer vision approaches and based on deep learning. Traditional computer vision approaches have difficulty in discriminating colors, textures, edges, shapes, and sizes, such as Histogram of Gradient (HOG) [1], Scale-invariant Feature Transform (SIFT) [2], Gabor Wavelets [3], Gabor filters [4], Fisher Kernels [5]. Object detection approaches based on deep learning have become popular in the field of computer vision because of the massive labeled data in natural images and some state-of-art deep learning models. Currently, these deep learning models can be divided into two classes: based on region proposals and based on regression. Deep learning models based on region proposals have higher accuracy but the speed is slower, such as R-CNN [6], Faster R-CNN [7], and Mask R-CNN [8]. Deep learning models based on

regression are faster but the accuracy is poor, such as YOLO [9] and SSD [10]. In addition, these deep learning models have requirements for the input image size. Most importantly, there is no publicly available dataset for garbage detection.

The above reasons brought great challenges to the garbage scattered area detection, resulting in few related studies found in the published literature. In the existing literature, Mittal et al. [11] built a dataset containing the garbage scattered area. Finally, they achieved the detection accuracy of 87.7% in urban images by using AlexNet [12]. Wei and Cheng [13] collected 372 urban images containing the garbage scattered area and expanded the number of images by 9 times through traditional data augmentation operations. Finally, they achieved the detection accuracy of 89.7% by using Faster R-CNN. However, the above research is based on the garbage scattered area detection in the urban areas and relies heavily on cameras. These approaches cannot be implemented in nature reserves.

In order to solve the above issues, we propose an automatic garbage scattered area detection (GSAD) model based on the state-of-the-art deep learning EfficientDet [14] method, which can achieve the best trade-off between accuracy and speed. As shown in Figure 1, a garbage scattered area sample is detected by our model. The main contributions in this paper are (1) we build a garbage sample dataset based on SUAV low-altitude remote sensing; (2) we propose a novel data augmentation approach based on garbage scattered area detection; (3) this paper establishes an model (GSAD) for garbage scattered area detection based on EfficientDet, data augmentation, transfer learning, and image blocking and gives future research directions.

2. Materials and Methods

In this section, we describe the dataset in detail and introduce a series of approaches used to optimize the GSAD model. The workflow of the GSAD is shown in Figure 2. It is composed of seven modules: (1) data collection; (2) data preprocessing; (3) data augmentation; (4) transfer learning with EfficientDet; (5) image blocking; (6) test; and (7) model evaluation.

2.1. Dataset. We collected a total of 630 remote sensing images that were taken by DJI Mavic 2 Pro with a flying height of 30 meters. The training set contains 480 images that were taken at Yunnan Normal University. And the test set contains 150 images that were taken in Dali Cangshan Erhai Nature Reserve. The image sizes range from $4,000 \times 3,000$ pixels to $5,472 \times 3,648$ pixels and the background of the garbage scattered area in the image is complex and diverse, including barren mountains, grassland, wetlands, and rivers. The size of the garbage scattered area in the image ranges from 200×200 pixels to 600×600 pixels. The garbage scattered area in the image is mainly presented in the form of stack and scatter. In addition, we fully consider the impact of brightness during the process of collecting images. The details of the dataset are shown in Table 1. Some training

samples taken from the dataset are shown in Figure 3 and some test samples are shown in Figure 4.

2.2. Data Preprocessing. We cannot directly use the training set to train deep learning models because the current deep learning models have requirements for the input image size. For example, YOLOv3 has three structures, including YOLOv3-320 structure, YOLOv3-416 structure and YOLOv3-608 structure. And the YOLOv3-416 structure is widely used. If the resolution of the image does not match the input size of the model, the image will be automatically scaled by the model. In this paper, we use SSD-512 structure, YOLOv3-416 structure, YOLOv4-512 [15] structure, and EfficientDet-512 structure as the base model.

For high-resolution remote sensing images, this direct scaling operation will seriously damage performance of the deep learning model for small objects detection because a lot of image information will be lost. Instead of converting low-resolution images into high-resolution images [16], we crop remote sensing images into subimages that contain the garbage scattered area. Considering that the size of the garbage scattered area in the image ranges from 200×200 pixels to 600×600 pixels, these subimages sizes range from $1,400 \times 1,400$ pixels to $1,800 \times 1,800$ pixels. Finally, we obtain 500 subimages cropped from the training set. We use these subimages as the new training set. However, these images will still be scaled. Due to the robustness of the deep learning model, a certain degree of scaling will not have a large impact on training the deep learning model with high prediction accuracy.

2.3. Data Augmentation. Training an automatic garbage scattered area detection model with high prediction accuracy requires massive training samples for support. However, the number of original training samples is limited. Fewer training samples will lead to poor performance of the deep learning model. In order to overcome the problem caused by insufficient samples, we adopted two data augmentation approaches to increase training samples. They are named DA1 and DA2, respectively.

2.3.1. DA1. The first is to use traditional data augmentation operations. The training sample is increased by brightness change, translation, flip, rotation, and zoom. The brightness change refers to the brightening and darkening of the image. The brightness change in remote sensing images has a significant impact on the detection performance of deep learning models. The rotation operation refers to rotating the image at any angle. In addition, we apply translation, flip, zoom in, and zoom out to increase training samples. These operations are shown in Figure 5.

2.3.2. DA2. The second is to use our proposed data augmentation approach. The pseudo code is in Algorithm 1. We use a camera to shoot a single garbage. Then we use simple image processing techniques to extract single garbage. In this way, we have extracted about 100 kinds of single garbage.

FIGURE 1: A garbage scattered area sample is detected in the remote sensing image. This sample is collected in Dali Cangshan Erhai Nature Reserve. The area of the green bounding box is enlarged and displayed.
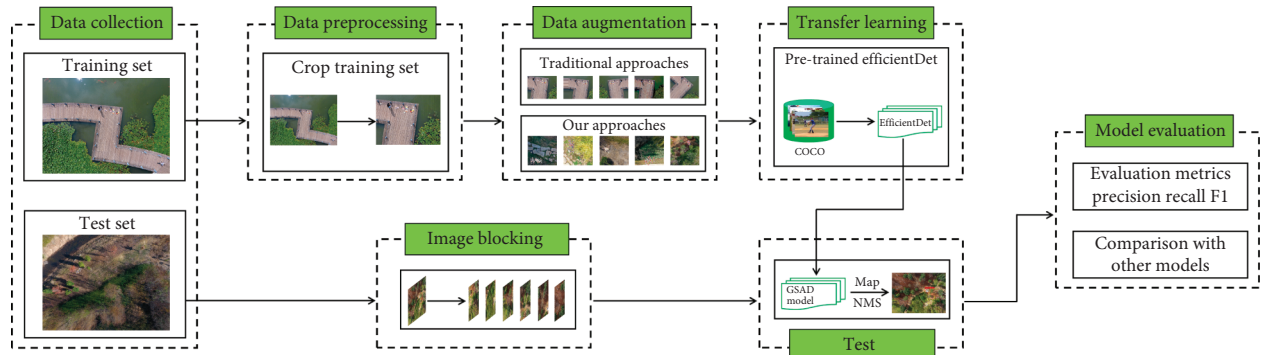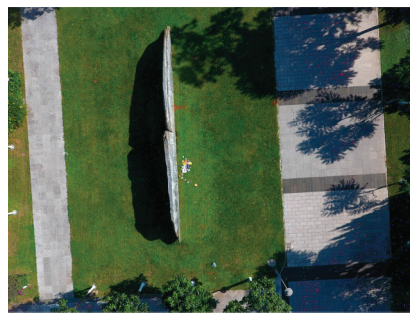

FIGURE 2: Workflow of GSAD model.

TABLE 1: The composition and details of the dataset.

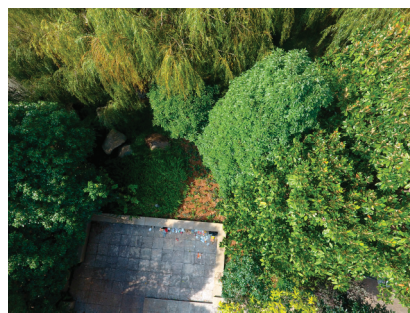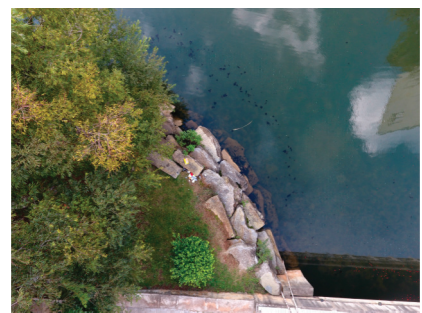| Dataset	Image number	Object number	Image size (pixels)	Object (pixels)	Height (m)
Training set	480	500	$4{,}000 \times 3{,}000 \sim 5{,}472 \times 3{,}648$	$200 \times 200 \sim 600 \times 600$	30
Test set	150	159	$4{,}000 \times 3{,}000 \sim 5{,}472 \times 3{,}648$	$200 \times 200 \sim 600 \times 600$	30

(a)

(b)

(c)

(d)
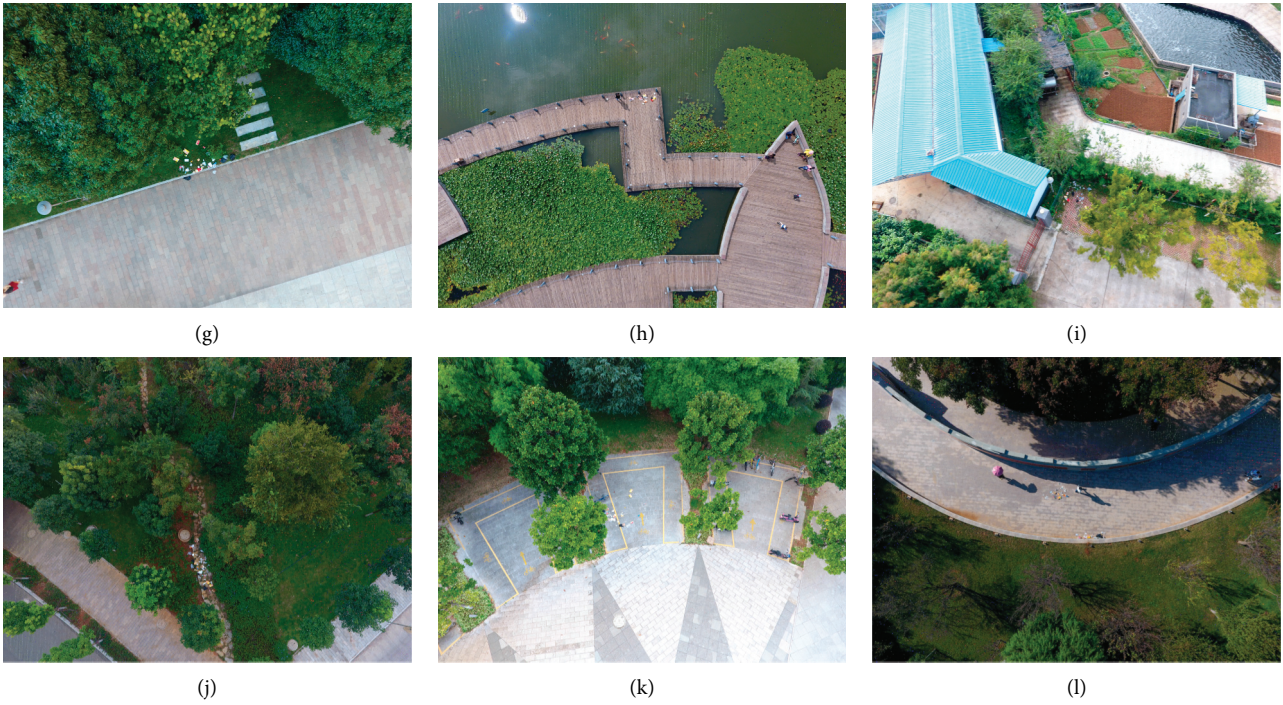

(e)

(f)

FIGURE 3: Continued.

(g)

(h)

(i)

(j)

(k)

(l)

FIGURE 3: Some training examples taken at Yunnan Normal University.

(a)

(b)

(c)

(d)

(e)

(f)
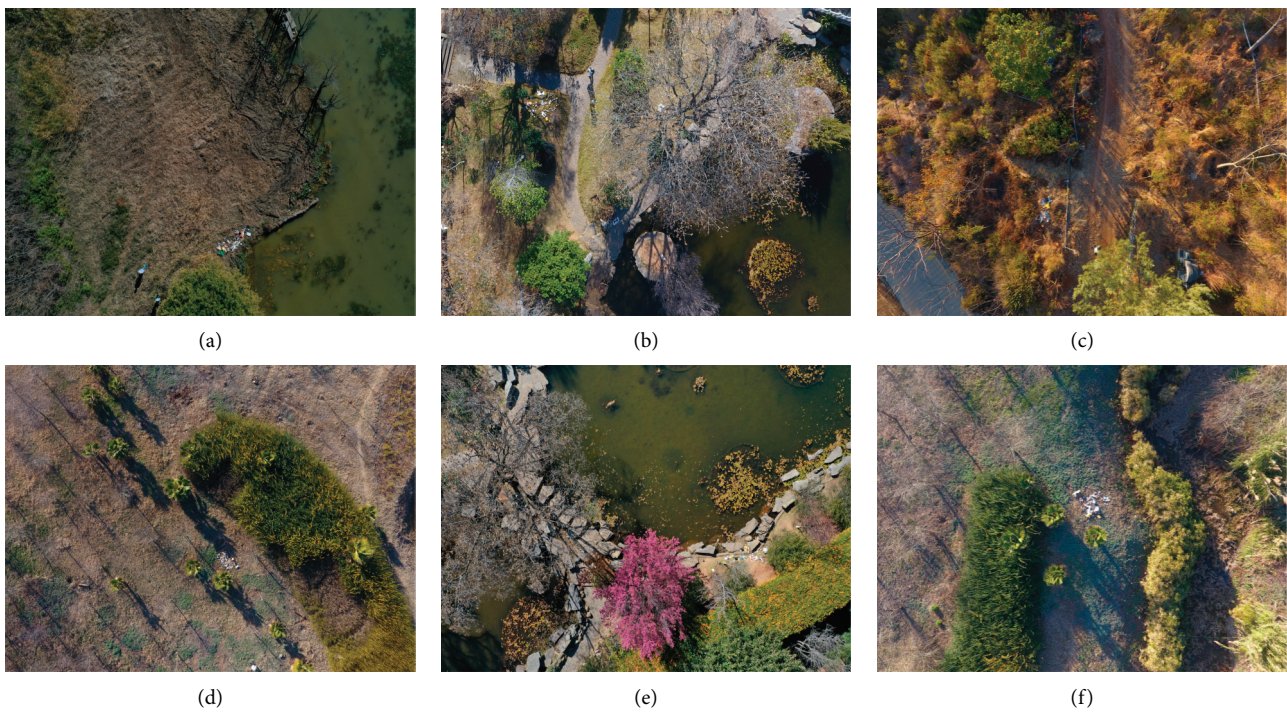
FIGURE 4: Continued.

(g)   (h)   (i)

(j)   (k)   (l)
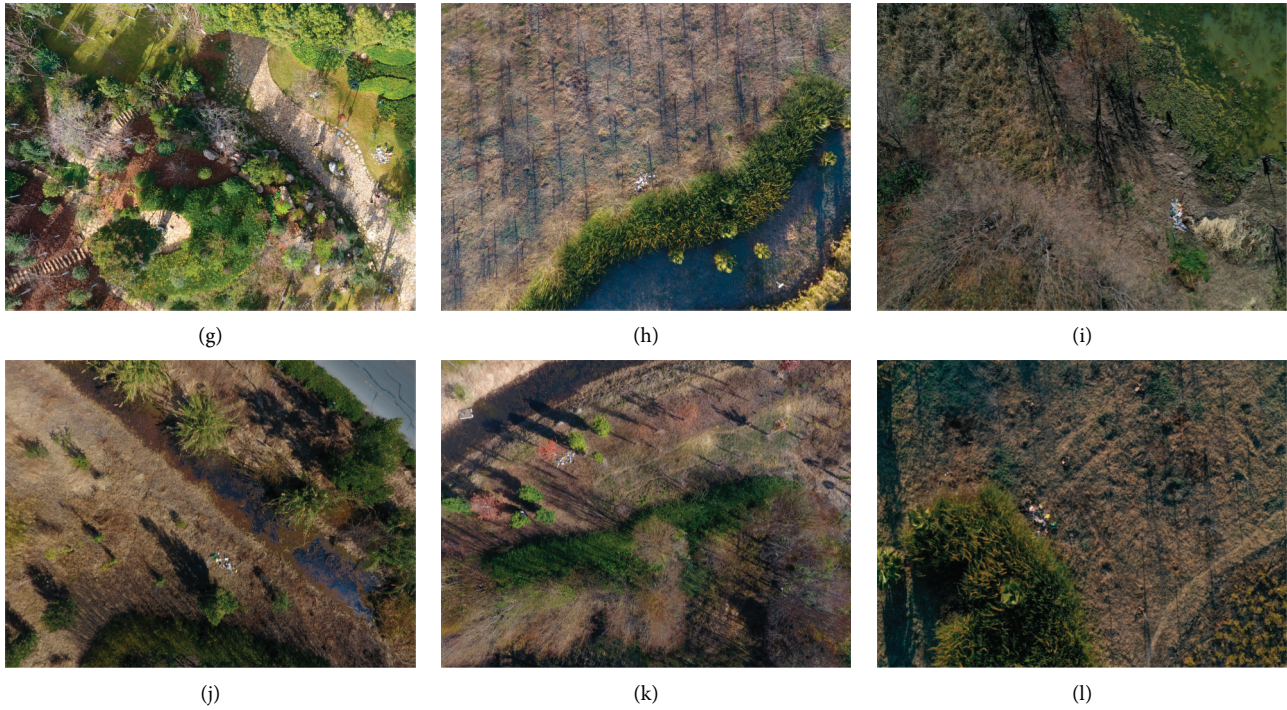
FIGURE 4: Some test examples taken in Dali Cangshan Erhai Nature Reserve.

(a)   (b)   (c)   (d)   (e)

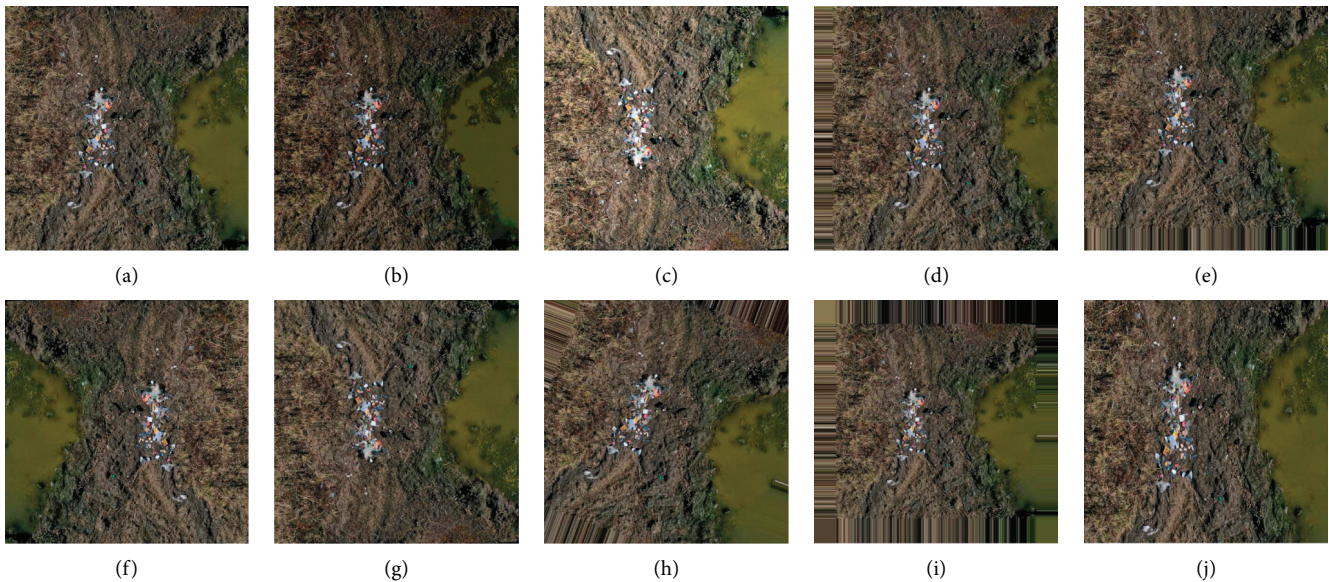(f)   (g)   (h)   (i)   (j)

FIGURE 5: Images that are generated by DA1: (a) original image, (b-c) brightness change, (d-e) horizontal translation and vertical translation, (f-g) horizontal flip and vertical flip, (h) rotation, and (i-j) zoom in and zoom out.

They have different shapes and appearances. Some samples are shown in Figure 6. We randomly select a certain number of single garbage and adjust the size of selected single garbage according to the proportion of single garbage in remote sensing images.

After we select a certain number of single garbage, we need to simulate the shape of selected garbage because the texture feature is one of the main features extracted by the deep learning model. The shape of the garbage scattered area has irregular characteristics. So we use irregular polygons to simulate the shape of the garbage scattered area. First, we randomly generate some key point sets. Then, we generate irregular polygons according to the rule that two points are closest to each other and do not cross. Some simulated shapes of the garbage scattered area are shown in Figure 7.

The distribution of the garbage scattered area is another major feature extracted by the deep learning model. Therefore, in the simulated shape, it is of great significance to
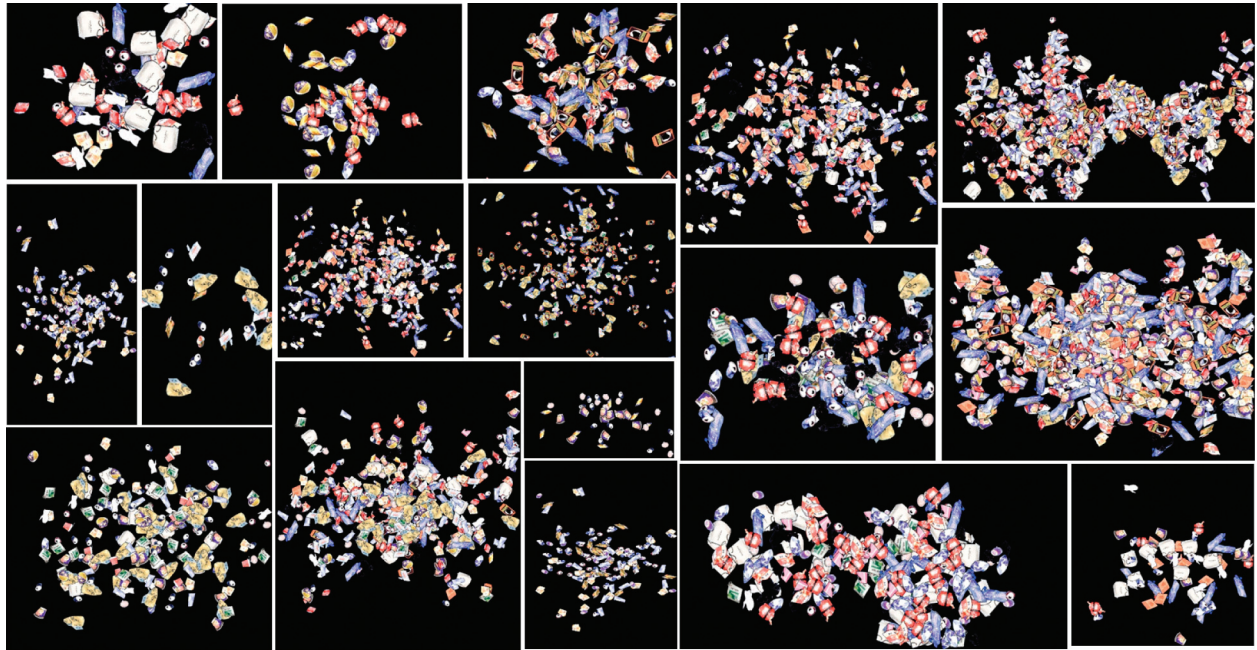
FIGURE 6: Some single garbage examples that have been extracted.

FIGURE 7: The white area is the simulated shape of the garbage scattered area.

simulate the distribution of the garbage scattered area as much as possible. We combine 8 distributions such as Normal distribution, Uniform distribution, and Exponential distribution. Then, we randomly generate scale factors for each distribution. The final distribution is determined by the following formula:

$$L = \sum_{i}^{N} Y_i B_i. \qquad (1)$$

Y_i is a scale factor for each distribution. B_i is a type of distribution function. N is the number of distributions. In this paper, N is 8. Some simulated distributions of the garbage scattered area are shown in Figure 8.

Finally, we choose the original dataset as the background image. The selected garbage will be automatically placed in a random location of the background image according to the simulated shape and simulated distribution. We crop these generated images into subimages that contain the simulated

Figure 8: Some simulated distributions of the garbage scattered area.

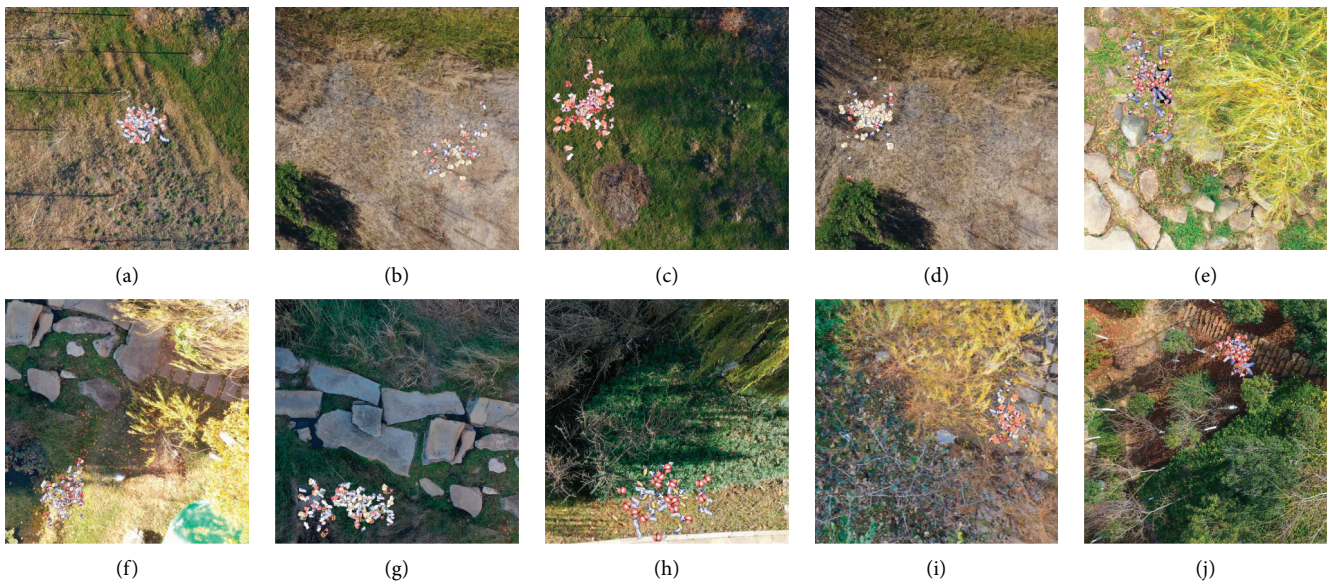| (a) | (b) | (c) | (d) | (e) |

| (f) | (g) | (h) | (i) | (j) |

Figure 9: Some generated training samples by DA2.

garbage scattered area. These subimages sizes also range from $1{,}400 \times 1{,}400$ pixels to $1{,}800 \times 1{,}800$ pixels. These subimages become a new training set. Some generated training samples are shown in Figure 9.

Through the above two data augmentation approaches, 4500 images and 1000 images are generated, respectively. The details of the final training set are shown in Table 2.

2.4. Transfer Learning. Through two data augmentation approaches, we have increased a certain number of training samples. However, the number of images in the training set is still not enough to train the deep learning model. We use transfer learning to solve the problem of garbage scattered area detection with a small training set. The transfer learning process can be divided into two main steps. First, we use the deep learning model trained on the COCO dataset [17] as the pretrained model. Then, we use the training set to fine-tune the pretrained model.

2.5. Image Blocking. In the test phase, we use image blocking to solve small objects detection in low-altitude high-resolution images. We will divide the input image into $a \times b$

Input: Single garbage aggregate A, Background image I.
Output: Image I containing the garbage scattered area.
(1) Randomly select a certain number of single garbage from A.
(2) Adjust the size of each selected single garbage according to the proportion of single garbage in remote sensing images.
(3) Simulate the shape B of the garbage scattered area according to the irregular polygon.
(4) In the simulated shape B, simulate the distribution L of the garbage scattered area.
(5) Automatically place the selected garbage in a random location of the image I according to the simulated shape B and simulated distribution L.
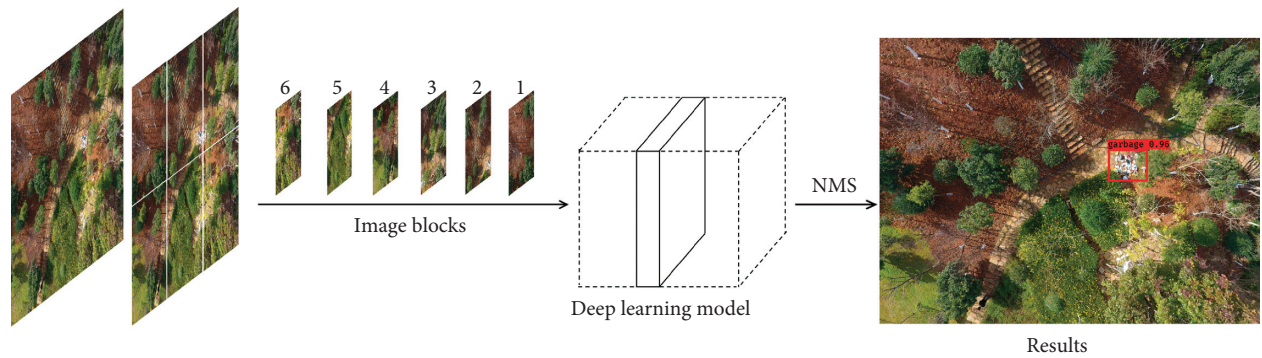

ALGORITHM 1: DA2

FIGURE 10: The process of detecting garbage scattered area in low-altitude remote sensing images by using our proposed approach.

TABLE 2: The details of the final training set.

Final training set	Original training set	DA1	DA2
Image number	500	4,500	1,000
Object number	500	4,500	1,126
Image size (pixels)	$1,400 \times 1,400 \sim 1,800 \times 1,800$	$1,400 \times 1,400 \sim 1,800 \times 1,800$	$1,400 \times 1,400 \sim 1,800 \times 1,800$

TABLE 3: Environment configurations.

Environment	Version
Operating system	Windows 10 (64 bit)
Framework	Pytorch
Memory	32 G
GPU	NVIDIA GTX-2080 Ti
CPU	Intel(R) core (TM) i7-9700 k
Library	CUDA 10.1, CUDNN 7.6.5

subimages before the image is input to the deep learning model. After we set a to 3 and b to 2, the width and height of each subimage can be calculated through the following formula:

$$a = \frac{\text{OriginalWidth} - \text{Overlap}}{\text{BlockWidth} - \text{Overlap}}, \qquad (2)$$

$$b = \frac{\text{OriginalHeight} - \text{Overlap}}{\text{BlockHeight} - \text{Overlap}}, \qquad (3)$$

where OriginalWidth and OriginalHeight are the width and height of the image in test set, respectively. Overlap is the overlap distance between two subimages, which is set to 200

pixels in this paper. BlockWidth and BlockHeight are the width and length of each subimage, respectively. The subimage sizes are approximately from $1,400 \times 1,400$ pixels to $1,800 \times 1,800$ pixels, which correspond to the size of the training set. Then, each subimage is detected by the deep learning model and the detection results of six subimages are mapped to the original image. Finally, we use the non-maximum suppression (NMS) algorithm in the original image to remove redundant bounding boxes. The detailed process of prediction is shown in Figure 10.

3. Results and Discussion

In this section, we introduce the experimental environment and evaluation criteria. In addition, a series of optimization experiments and comparative experiments are organized and implemented.

3.1. Experimental Environment. All experiments are conducted on a desktop with an Intel single Core i7 CPU, NVIDIA GTX-2080Ti GPU (11 GB video memory). Other experimental environments are shown in Table 3. In the training phase, the deep learning model is trained for a total

TABLE 4: Test results among EfficientDet models. ADT stands for average detection time.

Models	Precision (%)	Recall (%)	F1-score (%)	ADT (s)
EfficientDet	86.15	78.79	82.31	0.013
EfficientDet-DA1	88.12	82.14	85.02	0.013
EfficientDet-DA2	89.72	81.98	85.68	0.013
EfficientDet-DA1 & DA2	90.39	83.48	86.80	0.013
EfficientDet-DA1 & DA2 & TL	91.89	84.83	88.22	0.013
GSAD	95.68	94.55	95.11	1.096

TABLE 5: Quantitative comparison between different models. ADT stands for average detection time.

Models	Faster R-CNN (%)	SSD (%)	YOLOv3 (%)	YOLOv4 (%)	GSAD (%)
Precision	93.10	75.61	83.25	84.92	95.68
Recall	90.26	93.48	94.75	95.27	94.55
F1-score	91.66	83.60	88.63	89.80	95.11
ADT	2.012	1.281	1.312	1.072	1.096

(a)   (b)   (c)

(d)   (e)   (f)

(g)   (h)   (i)

FIGURE 11: Continued.

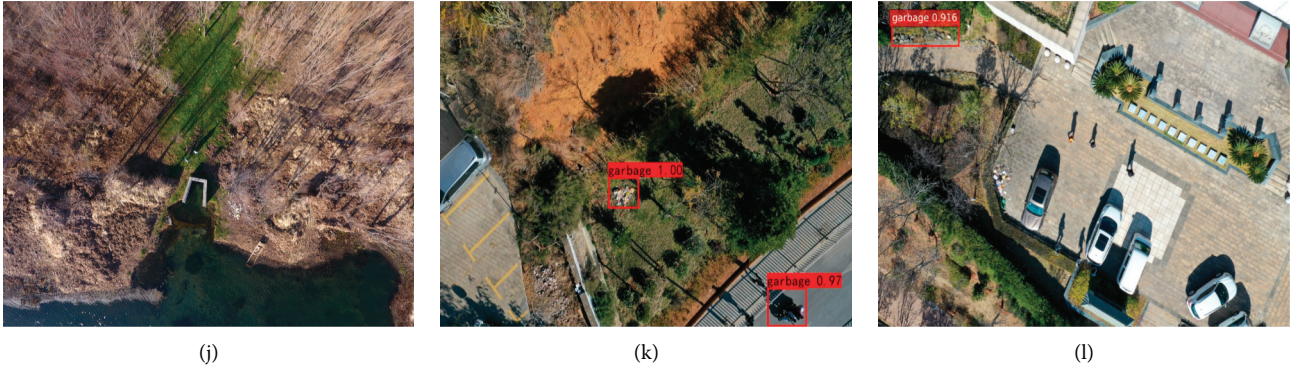(j)   (k)   (l)

Figure 11: False and missed detection samples detected by other deep learning models.

(a)   (b)   (c)

(d)   (e)   (f)

(g)   (h)   (i)
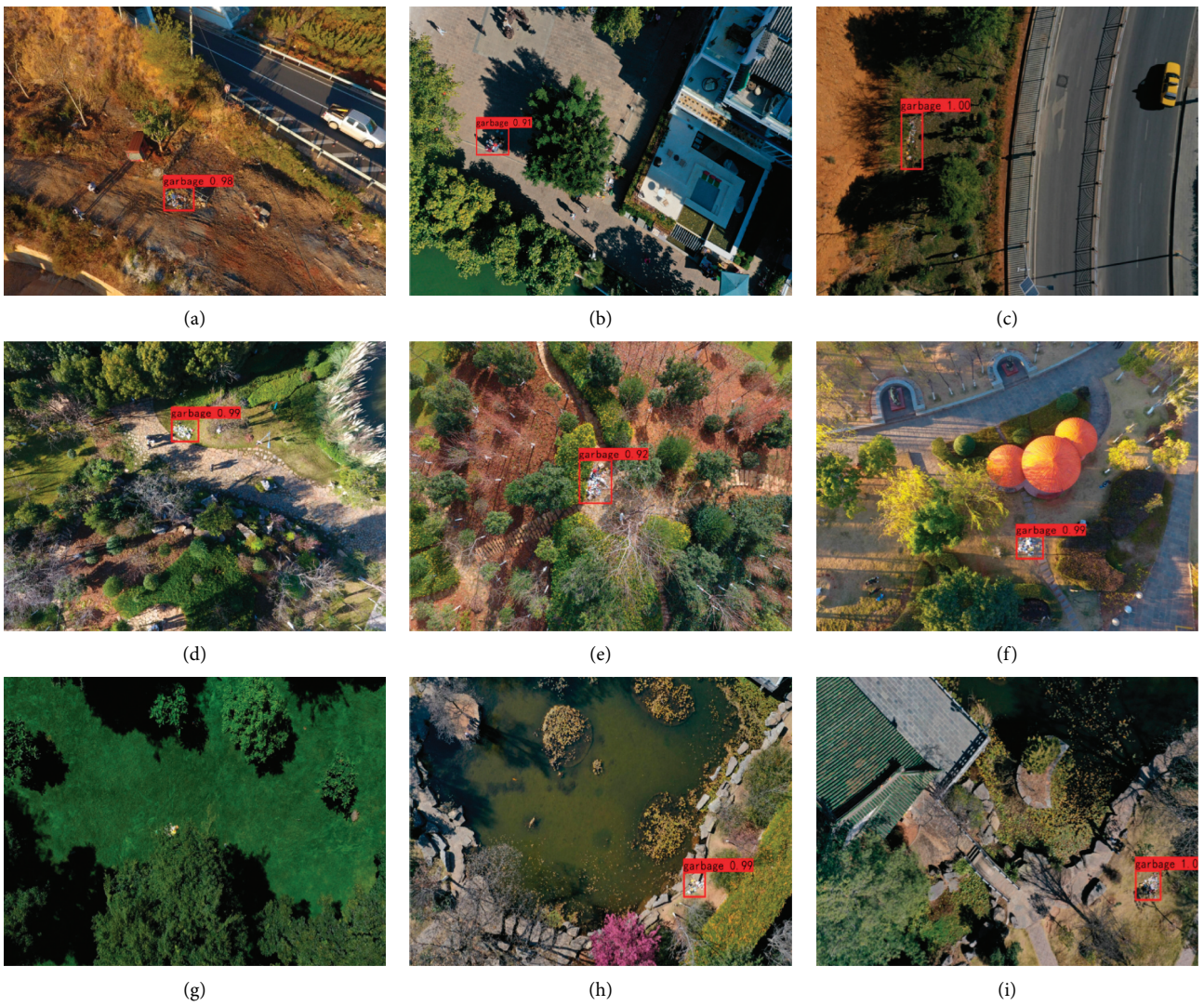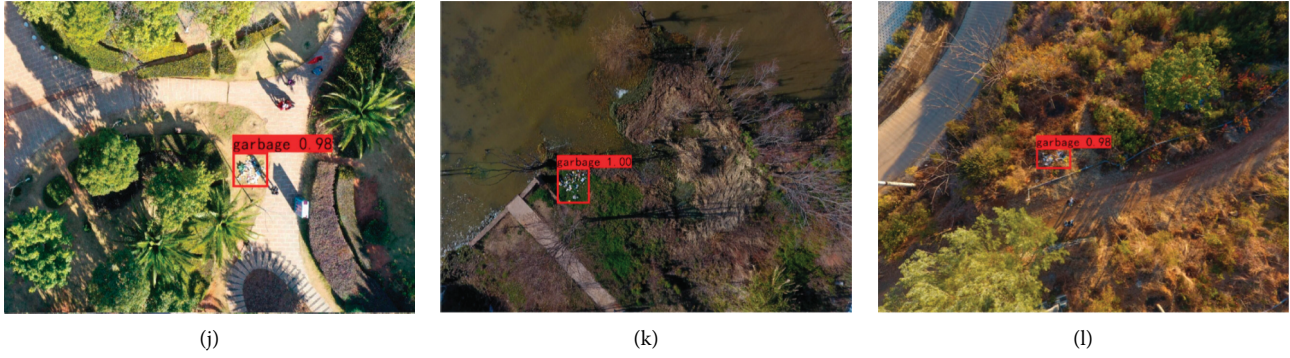
Figure 12: Continued.

FIGURE 12: Some test samples detected by the GSAD model.

of 40,000 iterations. The initial learning rate is set to 0.001. Then, the learning rate is divided by 10 on the 30,000th iteration and the 35,000th iteration. The batch size is set to 12. The weight decay is 0.0005 and the momentum is 0.9.

3.2. Evaluation Criteria.

In this paper, we use precision, recall, and F1-score as the evaluation criteria to evaluate the deep learning model for garbage scattered area detection in SUAV low-altitude remote sensing images. They are defined below:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{4}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{5}$$

$$\text{F1} - \text{score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}. \tag{6}$$

Precision represents the ratio of the true garbage scattered areas to the detected garbage scattered areas. Recall represents the ratio of the true garbage scattered areas to the total garbage scattered areas. F1-score is the harmonic mean between precision and recall.

3.3. Experimental Results

3.3.1. Comparison among EfficientDet Models.

In the first series of experiments, we test the effectiveness of the DA2, transfer learning, and image blocking. Table 4 shows the quantitative comparison results among six different EfficientDet models. In Table 4, EfficientDet denotes the EfficientDet without data augmentation; EfficientDet-DA1 denotes the EfficientDet with traditional data augmentation operations; EfficientDet-DA2 denotes the EfficientDet with our proposed data augmentation approach; EfficientDet-DA1 & DA2 denotes the EfficientDet with two data augmentation approaches; EfficientDet-DA1 & DA2 & TL denotes the EfficientDet with two data augmentation approaches and transfer learning. The GSAD model is based on EfficientDet, two data augmentation approaches, transfer learning, and image blocking.

As shown in Table 4, the precision of EfficientDet-DA2 is higher than EfficientDet-DA1, which demonstrates that DA2 can more effectively improve the precision of the GSAD model because DA2 produces new image information. The detection performance of using two data augmentation approaches is better than using DA1 and DA2 separately, which demonstrates that the two data augmentation approaches complement each other. When we use the transfer learning approach, the performance of the model will be slightly improved, which reflects the positive influence of transfer learning. The image blocking technology will improve the F1-score significantly because the deep learning model can extract more object information. However, this operation will destroy the real-time detection performance of the GSAD model. Its average detection time of 1.096 s is still acceptable.

3.3.2. Results on Other Deep Learning Models.

In the second series of experiments, we compare with other deep learning models. Four deep learning models based on transfer learning, data augmentation and image blocking are compared with the GSAD model. The results of the quantitative comparison are shown in Table 5. It is obvious that the Faster R-CNN model cannot recognize some garbage scattered areas, thus resulting in the effect that the recall is low. But the precision is high, which means the Faster-CNN is more capable of not falsely extracting nongarbage scattered areas. In addition, the speed of the Faster R-CNN is slow because the region proposals step and image blocking. When compared with the SSD, YOLOv3, and YOLOv4, those models will recognize people, trees, rocks, and other brightly colored things with similar characteristics as garbage, thus resulting in the effect that the precision is very low. And the recall of those deep learning models is high. This means that those models are more capable of extracting all the garbage scattered areas. Some false and missed samples detected by other models are shown in Figure 11. The precision and recall of the GSAD model are relatively high. This means that the GSAD model is more capable to extract all the garbage scattered areas and to not falsely extract nongarbage scattered areas. The average detection time of the GSAD model is still acceptable. However, when the garbage scattered area is partially obscured by vegetation, the model cannot effectively detect the garbage scattered area in remote sensing

images. In addition, the crowd gathered will also be detected as a garbage scattered area by the GSAD model with a certain probability. This is because the characteristics of the crowd gathered and the garbage scattered area are very similar. Some test samples detected by the GSAD model are shown in Figure 12.

4. Conclusions

In order to improve the cleaning efficiency and reduce secondary pollution of nature reserves, this paper proposes an automatic garbage scattered area detection (GSAD) model based on the state-of-the-art deep learning EfficientDet method, data augmentation, transfer learning, and image blocking. First, a garbage sample dataset based on remote sensing images is built. Then, we use two data augmentation approaches to increase training samples and use transfer learning to solve the problem of insufficient training samples. Finally, we apply image blocking technology to improve the detection performance of small objects in remote sensing images. Experimental results demonstrate that the performance of our proposed GSAD model is far superior to other deep learning models. GSAD can achieve the F1-score of 95.11% and average detection time of 1.096 s. According to the detection results, we obtain the location information of all garbage scattered areas in remote sensing images, and then formulate a reasonable cleaning route in nature reserves.

In future, we will study the SUAV aerial photography planning method. And we will propose an range positioning approach based on ground resolution estimation. Finally, we will develop a software platform that can automatically monitor garbage scattered areas and formulate a reasonable cleaning route in nature reserves.

Data Availability

The data used in the study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Authors' Contributions

Tengfei You and Weiyang Chen contributed equally to this work.

Acknowledgments

References

[1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of the IEEE computer society conference on computer vision & pattern recognition*, pp. 886–893, San Diego, CA, USA, June 2005.

[2] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[3] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837–842, 1996.

[4] A. K. Jain, N. K. Ratha, and S. Lakshmanan, "Object detection using Gabor filters," *Pattern Recognition*, vol. 30, no. 2, pp. 295–309, 1997.

[5] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the Fisher Kernel for large-scale image classification," in *Proceedings of the European Conference on Computer Vision*, pp. 143–156, Heraklion, Greece, September 2010.

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, Columbus, OH, USA, June 2014.

[7] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 91–99, Vancouver, Canada, December 2015.

[8] K. He, G. Georgia, D. Piotr, and R. Girshick, "Mask R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2961–2969, Venice, Italy, March 2017.

[9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, Las Vegas, NV, USA, June 2016.

[10] W. Liu, D. Anguelov, D. Erhan et al., "SSD: single shot multibox detector," in *Proceedings of the European Conference on Computer Vision*, pp. 21–37, Amsterdam, Netherlands, December 2016.

[11] G. Mittal, K. B. Yagnik, M. Garg, and N. C. Krishnan, "Spotgarbage: smartphone app to detect garbage using deep learning," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 940–945, Heidelberg, Germany, September 2016.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 1097–1105, Lake Tahoe, NV, USA, December 2012.

[13] S. Wei and Z. Cheng, "Image-based automatic detection of urban scene garbage," *Integration Technology*, vol. 6, no. 1, pp. 39–52, 2017.

[14] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: scalable and efficient object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10781–10790, Seattle, WA, USA, June 2019.

[15] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "YOLOv4: optimal speed and accuracy of object detection," 2020, http://arxiv.org/abs/2004.10934.

[16] P. Yi, Z. Wang, K. Jiang, Z. Shao, and J. Ma, "Multi-temporal ultra dense memory network for video super-resolution," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 8, pp. 2503–2516, 2020.

[17] T. Y. Lin, M. Maire, S. Belongie et al., "Microsoft COCO: common objects in context," in *Proceedings of the European Conference on Computer Vision*, pp. 740–755, Zurich, Switzerland, September 2014.