Dense and Small Object Detection in UAV-Vision Based on a Global-Local Feature Enhanced Network

Tao Ye, Wenyang Qin, Yunwang Li, Shouan Wang, Jun Zhang, and Zongyang Zhao

***Abstract*— Unmanned aerial vehicles (UAVs) have been widely used in postdisaster search and rescue operations, object tracking, and other tasks. Therefore, the autonomous perception of UAVs based on computer vision has become a research hotspot in recent years. However, UAV images include dense objects, small objects, and arbitrary object directions, which bring about significant challenges to existing object detection methods. To alleviate these issues, we propose a global-local feature enhanced network (GLF-Net). Considering the difficulty of processing UAV images with complex scenes and dense objects, we designed a backbone based on an involution and self-attention that can extract effective features from complex objects. A multiscale feature fusion module is also proposed to address the presence of numerous small objects in UAV images through multiscale object detection and feature fusion. To accurately detect rotated objects, a rotated regional proposal network was designed based on the midpoint offset representation, which can apply a rotated box to determine the real direction and contour of an object. GLF-Net achieves a state-of-the-art detection accuracy [86.52% mean average precision (mAP)] on our created rotated object detection UAV (RO-UAV) dataset, while achieving 96.95% and 97% mAP on the public datasets high resolution ship collections 2016 (HRSC2016) and the University of Chinese Academy of Sciences High Resolution Aerial Object Detection Dataset (UCAS-AOD), respectively. The experimental results demonstrate that our method achieves a high detection accuracy and generalization, which can meet the practical requirements of UAVs under various complex scenarios.**

***Index Terms*— Autonomous perception, computer vision, global-local feature enhanced, rotated object detector, unmanned aerial vehicles (UAVs).**

I. Introduction

IN RECENT years, the rapid development of unmanned aerial vehicle (UAV) technology has enabled its application in various fields. Because the flight of a UAV is not limited by terrain, it can easily complete tasks that cannot be accomplished by ground equipment, thus greatly improving the work efficiency and reducing labor demand and the consumption of resources [1]. In addition to the traditional transport of goods and pesticide spraying, new UAVs equipped with cameras and communication modules have been widely applied in landscape photography, search and rescue operations, safety monitoring, disaster prediction, criminal tracking, smart cities, and other areas [2], [3], [4]. However, most UAV applications are based on real-time remote control applied by ground personnel; that is, UAVs need to transmit captured images to the server through a network and then conduct the next operation after an analysis by the staff. Therefore, a reliable data link between a UAV and its server is crucial. This link is impossible to achieve under harsh working conditions, such as in postdisaster rescue or criminal tracking, which severely limits the working range and usage scenarios of such vehicles [5], [6].

The rapid development of computer vision technology makes it possible for UAVs to autonomously perceive objects within their field of view [7]. However, compared with images of natural scenes, images captured under the field of view of a UAV are more complex, and the requirements for an object detector are more stringent. This is mainly reflected through the following.

1) The image captured by a UAV based on its field of view is changeable and complex, and the distribution of objects may be dense or sparse, resulting in a low robustness of the existing object detector.
2) When a UAV conducts object detection, it is usually far from the ground, and the objects in the collected images show fine-grained characteristics; however, fine-grained objects face a risk of disappearing features when they are downsampled under a deep neural network model.
3) The objects in an image based on the vision system of the UAV are mostly disorganized, which makes it difficult to extract the real contour of an object.

The above factors bring about significant challenges to high-precision object detection within the field of view of the UAV.

In this study, a global-local feature enhanced network (GLF-Net) is proposed according to the actual characteristics of UAV images, which alleviates the problem in which current methods are unable to properly detect objects in UAV images. First, we designed a novel backbone that can effectively extract features from complex scenes. This backbone mainly consists of three modules: a feature recombination module (FRM), a local feature extraction (LFE) module based on an involution operator, and a global feature extraction (GFE) module based on a multihead self-attention mechanism. The FRM is used

Manuscript received 3 April 2022; revised 16 June 2022; accepted 23 July 2022. Date of publication 4 August 2022; date of current version 16 August 2022. This work was supported in part by the State Key Laboratory of Coal Mining and Clean Utilization, China, under Grant 2021-CMCU-KF012; in part by the National Science Foundation of China under Grant 52121003; and in part by the Fundamental Research Funds for the Central Universities under Grant 2022YJSJD01 and Grant 2022YQJD04. The Associate Editor coordinating the review process was Lei Zhang. *(Corresponding author: Tao Ye.)*

Tao Ye, Wenyang Qin, Shouan Wang, Jun Zhang, and Zongyang Zhao are with the School of Mechanical Electronic and Information Engineering, China University of Mining and Technology, Beijing 100083, P. R. China, also with the State Key Laboratory of Coal Mining and Clean Utilization, Beijing 100083, China, and also with the Key Laboratory of Intelligent Mining and Robotics, Ministry of Emergency Management, Beijing 100083, China (e-mail: ayetao198715@163.com; zqt2000401020@student.cumtb.edu.cn; shuttworth@foxmail.com; z18811768267@163.com; zzy303616426@126.com).

Yunwang Li is with the School of Mechanical Electronic and Information Engineering, China University of Mining and Technology, Beijing 100083, P. R. China, and also with the Key Laboratory of Intelligent Mining and Robotics, Ministry of Emergency Management, Beijing 100083, China (e-mail: yunwangli@cumtb.edu.cn).

Digital Object Identifier 10.1109/TIM.2022.3196319

for a stable downsampling of the input images with little loss of low-level feature information. By using an involution instead of a convolution, LFE can effectively extract local feature information of the feature maps with few computations. GFE extracts global information of images through a multihead self-attention mechanism to realize the interaction of information at a long range. Second, we propose the use of a multiscale feature fusion (MFF) module to solve the problem of small object detection. The MFF module fuses the global features obtained by GFE and local features at other levels to enhance the expression capability of the features, thus improving the accuracy of small object detection by the network. In addition, the model can supplement detailed information for high-level feature maps through a bottom-up branch to improve the network's anti-interference ability and multiscale object detection ability. Finally, a rotated regional proposal network (RPN) based on a midpoint offset representation generates rotated proposals, using a rotated box to accurately determine the real contour of an object from among other disorganized objects. Combining these three technologies, our method achieves promising results in the object detection of UAV images.

To evaluate the effectiveness of GLF-Net, we created a large rotated object detection dataset of UAV images under rich scenes, called RO-UAV, which contains 6534 images with a pixel resolution of 4056×3040 and 69 873 annotated instances for three categories, i.e., car, truck, and special. Compared with previous methods, our approach achieves the highest detection accuracy of 86.52% mean average precision (mAP) on the RO-UAV dataset. In addition, it achieves 96.95% and 97% mAP on the public datasets high resolution ship collections 2016 (HRSC2016) and the University of Chinese Academy of Sciences High Resolution Aerial Object Detection Dataset (UCAS-AOD), respectively.

Our main contributions are summarized as follows.

1) Aiming at the above problems with the field of view of images captured by UAVs, we propose a deep convolutional neural network (CNN) model called GLF-Net. The LFE and GFE modules are used to extract the local and global features of the image, respectively, complete the robust feature extraction in complex and dense scenes, and achieve a highly precise detection of dense objects. The MFF module further enhance their expression ability by fusing these global and local features, thus improving the detection accuracy of multiscale objects.
2) To address the challenge of disorganized object detection in the field of view of a UAV, we designed a rotated RPN based on the midpoint offset representation method. Our proposed method boxes objects more accurately by generating high-quality rotated proposals, allowing the real contour of an object to be extracted.
3) Our method achieved the highest detection accuracy of 86.52% mAP on the RO-UAV dataset. In addition, 96.95% and 97% mAP were achieved on the public datasets HRSC2016 and UCAS-AOD, respectively. The experimental results show that GLF-Net can meet the requirements of high-precision UAV object detection under various complex scenarios.

The remainder of this article is organized as follows. Section II summarizes previous relevant studies. Section III introduces the details of GLF-Net. Section IV presents extensive experimental results. Finally, Section V provides some concluding remarks.

II. Related Work

UAVs can quickly reach a scene regardless of the presence of obstacles, whether on a damaged road after a disaster or during an urban traffic jam, for human search and rescue operations or object tracking. However, the dependence on network signals during manual operation seriously affects the operational ability of a UAV; therefore, UAVs require to automatically and accurately detect objects under various complex environments.

A. UAV Vision

With the development of machine learning technology, many researchers have begun to focus on applying vision systems to UAVs. Kendoul *et al.* [8] used a low-resolution onboard camera and a low-cost inertial measurement unit (IMU) to estimate the optical flow, aircraft self-motion, and depth map to realize autonomous UAV flights. Saripalli *et al.* [9] used cameras instead of radar and GPS to assist a UAV in accurately identifying the location of a runway and ensuring a safe landing. In addition, Meier *et al.* [10] designed a micro-air vehicle for autonomous flight and obstacle detection using an IMU and computer vision. Scaramuzza *et al.* [11] also developed a micro-flying robot, which requires only a single airborne camera and an IMU to navigate autonomously, with no GPS signal or 3-D mapping of the ground. However, these methods simply cooperate with other sensors for auxiliary driving and cannot achieve object detection through a UAV.

Subsequently, with the rapid development of computer vision technology, some UAV object detection methods based on image processing were introduced. Moranduzzo and Melgani [12] extracted the key points on a scale-invariant feature transform and used a support vector machine (SVM) to classify the feature points as either "car" or "car free." In [13], a method was proposed to extract histogram of gradient (HOG) features through a filtering operation and detect objects by comparing the similarity with a reference. In [14], an object detection method was proposed based on different orders and Gaussian process modeling. Xu *et al.* [15] proposed a scheme combining Viola–Jones (V–J) and a linear SVM classifier with HOG features (HOG + SVM) to realize UAV object detection. These methods use image processing or an artificial feature design to realize UAV object detection, which requires sufficient prior knowledge or experience. However, the traditional manual features and classifier design cannot mine additional information with high-level semantics for complex and changeable scenes. Therefore, it is difficult to design a unified detection method to adapt the object detection problem for various complex scenarios.

To overcome the limitations of traditional manual feature extraction, some researchers have begun to use CNNs for adaptive feature extraction. Nassim *et al.* [16] aimed to address the high resolution of UAV images by segmenting the images into small homogeneous regions, extracting features through a deep CNN, and classifying them as "car" and "no-car" using a linear SVM. In addition, Bazi and Melgani [17] proposed a convolutional SVM (CSVM) to solve the challenge in UAV images caused by changes in scale. CSVM

includes several alternating convolution layers and reduction layers, as well as the last linear SVM classification layer. Bejiga *et al.* [18] used a pretrained CNN to extract discriminative features and then integrated an SVM at the top of the CNN to detect the object of interest. Although these methods have made considerable progress, they only use CNNs for feature extraction followed by an SVM for classification, which does not form a complete detection network. The detection speed and accuracy are unable to meet the actual needs of UAVs in complex environments.

B. Object Detection Using CNN

1) Horizontal Object Detection: With the development of deep learning technology, vision-based object detection algorithms have achieved incredible results. Over the past few years, object detection algorithms based on deep learning have been divided into two categories, i.e., one- and two-stage algorithms. Two-stage methods, as represented by Regions with CNN features (R-CNN) [19], Fast R-CNN [20], Faster R-CNN [21], and feature pyramid network (FPN) [22], use region proposals to generate candidates for object detection, which has the advantages of a high precision but requires heavy calculations. One-stage methods, such as Single Shot MultiBox Detector (SSD) [23], You Only Look Once (YOLO) [24], and RetinaNet [25], treat the detection problem as a regression problem and directly predict the location and category of the objects. Although an advantage with such an approach is the high speed attained, the detection accuracy is relatively low. In recent years, to avoid the limitation of anchor boxes on the further improvement of the detection effect, anchor-free methods, such as Fully Convolutional One-Stage Object Detection (FCOS) [26], CenterNet [27], and You Only Look Once X (YOLOX) [28], have also become popular and have achieved significant results. The recent successful application of transformers, particularly vision transformers [29], in the area of vision has propelled object detection methods into a new era. However, although these methods have achieved satisfactory results in natural scene images, they are based on horizontal boxes. As shown in Fig. 1, a horizontal box cannot accurately determine the real direction, aspect ratio, or contour of an object. In particular, dense objects often cause serious interference between horizontal boxes and affect the correct judgment of the UAV.

2) Rotated Object Detection: Rotated object detection refers to building detectors using a rotated bounding-box representation. The objects in aerial images and scene text are usually dense and distributed in a disorderly manner, appearing in any direction. When using horizontal detection boxes, they cannot surround the object well. To alleviate this problem, Zhang *et al.* [30] used a rotating region of interest (RROI) pooling layer to extract rotated object features and applied a rotated box to detect ships in remote sensing images. In addition, Ding *et al.* [31] transformed a horizontal ROI into an RROI and extracted rotation-invariant features to promote subsequent classification and regression. Xu *et al.* [32] accurately described multidirectional objects by gliding the vertex of the horizontal bounding box on each corresponding side. Li and Zhu [33] also employed a set of adaptive points to capture the geometric and spatial information of arbitrarily oriented objects and mapped these point sets into an oriented bounding box using the oriented conversion function.

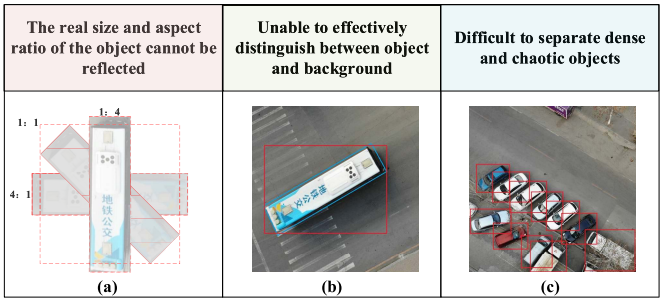


Fig. 1. Limitations of horizontal detection box. (a) Horizontal detection box does not reflect the real size and aspect ratio of the object. (b) Horizontal detection box does not effectively distinguish between object and background. (c) Horizontal detection box is difficult to separate dense and chaotic objects.

Moreover, Xie *et al.* [34] found that an oriented R-CNN generates oriented proposals using an RPN in the first step and refines and recognizes them in the second stage through the oriented R-CNN head. Ma *et al.* [35] proposed using an RRPN to add a direction variable to a traditional anchor, realizing a text detection in a natural scene by generating 54 proposals with different directions, sizes, and scales at each point. Han *et al.* [36] incorporated rotation-equivariant networks into the detector to extract rotation-equivariant features, which can accurately predict the orientation, leading to a significant reduction in the model size. In addition, Yang and Yan [37] proposed a circular smooth label (CSL) technique to handle the periodicity of an angle. In [38], two densely coded labels were designed to replace the sparsely coded label and improve the training speed. In addition, Yang *et al.* also proposed several excellent rotated object detectors, including SCRDet [39], R2CNN++ [40], and R3Det [41].

These rotated object detection methods achieve good results in remote sensing images. However, UAV images present more serious challenges than traditional remote sensing images. For instance, the shooting conditions of a traditional remote sensing image (e.g., shooting height and angle) are almost unchanged. Hence, the features of objects of the same type in the image are essentially the same. By contrast, UAV images are affected by the shooting mode, weather, and UAV flight speed, which makes their features more complex. These challenges have led to rotated object detection algorithms remaining unable to meet the requirements of UAV object detection under complex working scenes.

III. Proposed Network Framework

We propose a rotated object detection network called GLF-Net to meet the object detection requirements of UAVs under various complex scenarios, such as postdisaster search and rescue operations and object tracking. GLF-Net mainly contains three parts: a backbone containing FRM, LFE, and GFE; an MFF module; and a rotated RPN. Among them, the backbone can stably extract effective information from complex images, and MFF improves the detection ability of small objects through MFF. The rotated RPN accurately determines the true direction and contour of an object using a rotated box as an object frame.

A. Backbone

1) FRM: When image features are extracted using a deep neural network, shallow features are often destroyed as the size of the feature map is significantly reduced. The FRM is

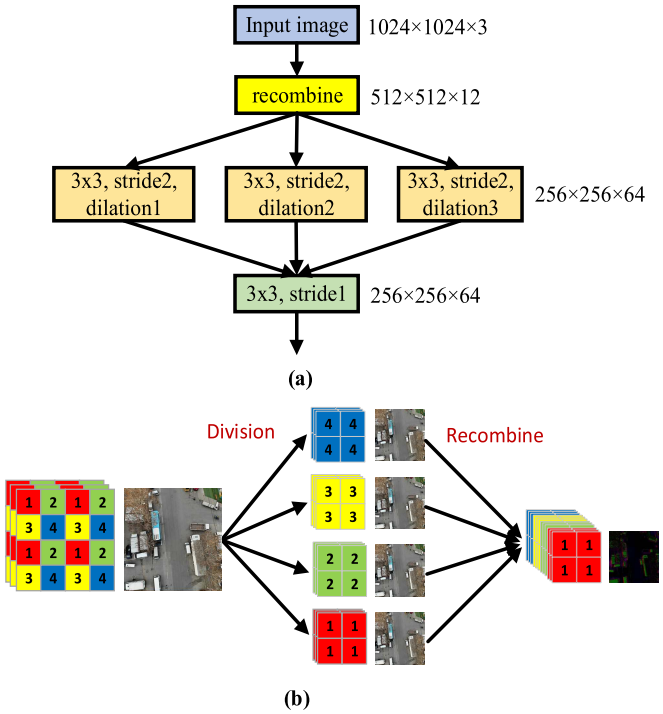


Fig. 2. Illustration of the FRM: (a) overall structure of the FRM and (b) schematic and an example of recombination.

designed to retain more information contained in the shallow features of the image during downsampling, as shown in Fig. 2. Specifically, a value for every other pixel in the input image is first applied by the FRM to obtain four similar and complementary images containing all information of the input image, and the four images are then recombined according to the channel dimension, allowing a centralization of the spatial dimension information into the channel dimension without a loss of information. The next multibranch channel obtains a variety of receptive fields of different sizes using a dilated convolution with different dilatation rates, thus obtaining information of different scales. Finally, a 3×3 convolution is used to combine the information. Compared with traditional methods, this structure can achieve a stable downsampling while retaining more underlying feature information.

Mainstream object detection algorithms often use small images as inputs. However, UAV images have the characteristics of a large field of view and high resolution, and there are many small objects cluttered throughout the image. Therefore, using low-resolution images as the input will significantly reduce the detection accuracy of the network. In the model, the input image is resized to a pixel resolution of 1024×1024 after considering various factors, such as the number of calculations, accuracy, and small objects. Then, fourfold downsampling can be achieved by the FRM with little loss of information, which effectively improves the feature expression capability of the model.

2) LFE: As the main construction method used in deep neural networks, a convolution has two remarkable properties: spatial-agnostic and channel-specific operation. In the spatial dimension, the convolution kernel ensures efficiency through two mechanisms: parameter sharing and translation invariance. In the channel dimension, the convolution kernel

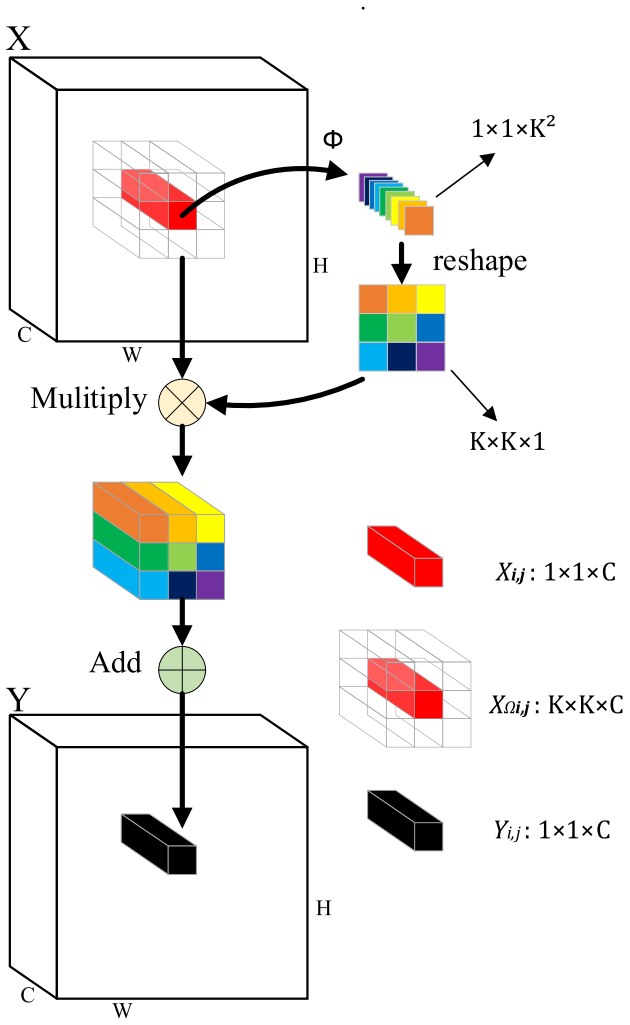


Fig. 3. Illustration of an involution. The involution kernel is generated from the feature vector of a single point (i, j) by the function Φ and a reshape operation. The multiply–add is divided into two steps, with multiply indicating the multiplication broadcast across C channels, and add indicating the summation aggregated within the $K \times K$ spatial neighborhood.

uses a high-dimensional matrix to realize the information exchange of different features between channels. However, on the one hand, the convolution kernel cannot flexibly adjust the parameters to extract different spatial location features according to the input because of the spatial-agnostic property of the convolution kernel. On the other hand, the channel-specific convolution kernel leads to serious redundancy in the internal channels of the convolution filters and affects the work efficiency [42].

By contrast, an involution operator is spatial-specific and channel-agnostic. Specifically, it uses different involution kernels at different spatial locations and shares parameters among the channel dimensions. To ensure that it can accommodate different inputs, an involution kernel is generated according to the corresponding input feature map, as shown in Fig. 3. For the feature vector of a point in the spatial input feature map, we first map it to $1 \times 1 \times K^2$ using the kernel generation function Φ, where K is the size of the involution kernel, and the kernel generation function Φ for the same involution operation is shared. It is then expanded into a kernel shape through a reshape operation to obtain the corresponding involution

kernel for that point. Finally, a multiply–add operation with the feature matrix of this point within the spatial neighborhood is applied to the input feature map to obtain the output feature map. The specific process is as follows:

$$\begin{aligned} &X_{i,j}: 1 \times 1 \times C \xrightarrow{\Phi} 1 \times 1 \times K^2 \xrightarrow{\text{reshape}} K \times K \times 1 \\ &X_{\Omega_{i,j}}: K \times K \times C \\ &Y_{i,j}: X_{i,j} \xrightarrow{\text{M Add}} X_{\Omega_{i,j}} \longrightarrow 1 \times 1 \times C \end{aligned} \tag{1}$$

where Ω is the spatial neighborhood of $K \times K$ near point (i, j) of the input feature map.

Assuming that the size of the input feature map is $W \times H \times C$, where W, H, and C are the width, height, and number of channels of the feature map, respectively, and that the output feature map size is also $W \times H \times C$, the parameters for using an ordinary convolution are as follows:

$$\mathcal{M}_{\mathrm{c}} = C \times C K_{W\mathrm{o}} K_{H0} = C^2 K_{W\mathrm{c}} K_{H\mathrm{c}} \tag{2}$$

where $K_{W\mathrm{c}}$ and $K_{H\mathrm{c}}$ are the width and height of the convolution kernel, respectively. The number of calculations of a convolution operation is

$$\mathcal{H}_{\mathrm{c}} = C \times WH \times C K_{W\mathrm{c}} K_{H\mathrm{c}} = WHC^2 K_{W\mathrm{c}} K_{H\mathrm{c}}. \tag{3}$$

Under the same conditions, the number of involution operation parameters is

$$\mathcal{M}_{\mathrm{I}} = 1 \times 1 \times C \times 1 \times 1 \times K_{W\mathrm{I}} K_{H\mathrm{I}} = C K_{W\mathrm{I}} K_{H\mathrm{I}} \tag{4}$$

where $K_{W\mathrm{I}}$ and $K_{H\mathrm{I}}$ are the width and height of the involution kernel, respectively. The number of calculations of the involution operation is

$$\begin{aligned} \mathcal{H}_{\mathrm{I}} &= \mathcal{H}_{\mathrm{I}}^{\Phi} + \mathcal{H}_{\mathrm{I}}^{\text{M Add}} \\ &= WH \times C K_{W\mathrm{I}} K_{H\mathrm{I}} + WH \times C K_{W\mathrm{I}} K_{H\mathrm{I}} \\ &= 2WHC K_{W\mathrm{I}} K_{H\mathrm{I}} \end{aligned} \tag{5}$$

where $\mathcal{H}_{\mathrm{I}}^{\Phi}$ is the number of calculations of the kernel generation function Φ, and $\mathcal{H}_{\mathrm{I}}^{\text{M Add}}$ represents the number of calculations of a multiply–add operation.

The number of parameters of the involution operation is only $1/C$ (where C is usually a large number) of the convolution operation, and the number of calculations is only $2/C$ of the convolution operation under the same conditions. Therefore, an involution allows us to use a large kernel for a spatial feature extraction without a computational explosion. In addition, although the filter parameters are not shared between different spatial locations, the kernel generation function Φ of each point is shared when the kernel is generated, and thus, it realizes an information exchange and migration between different spatial locations at a higher level than a convolution.

The LFE module shown in Fig. 4 was designed to combine different characteristics of an involution. Because an involution currently has difficulty in changing the number of channels of the feature map, it is inevitable to use a 1×1 convolution filter for control. Compared with a common 3×3 convolution, a 7×7 involution has a larger receptive field, which can efficiently extract local features. In the case of UAV images with a high resolution and complex content, the LFE module can effectively extract local detailed information with fewer calculations and improve the detection accuracy.

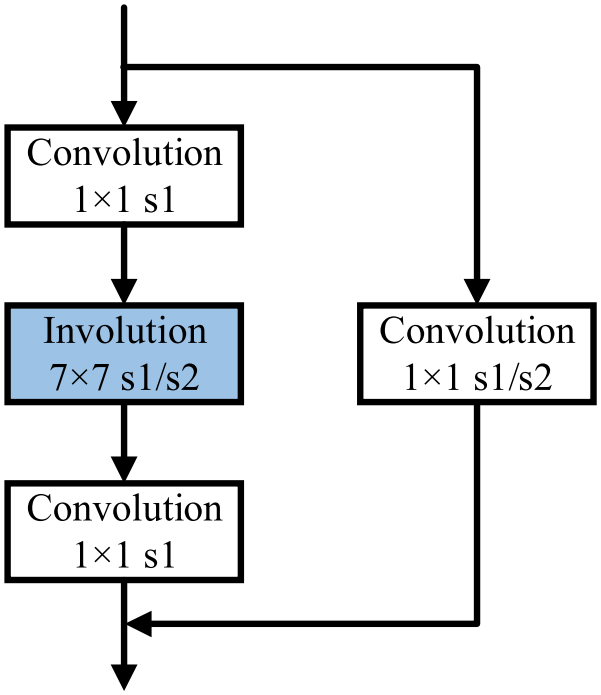


Fig. 4. Structure of the LFE module. The residual structure adjusts the size using a 1×1 convolution with a stride of 2 during downsampling and a stride of 1 at other times.

3) GFE: Although the LFE module can capture detailed information more effectively than a traditional deep CNN, vision tasks require the modeling of long-range dependencies. In particular, when the same objects in UAV images show completely different features under different circumstances, it is extremely useful to collect and associate scene information from a large neighborhood to learn the relationship between objects. Convolution-based networks tend to aggregate information within a wide range of images by stacking multiple layers, which makes these methods unable to effectively obtain global information.

Self-attention [43] was first applied to natural language processing tasks that also need to model long-range dependencies. It realizes a pairwise entity interaction through a content-based addressing mechanism, thus learning a rich hierarchy of associative features across long sequences. Vision transformers [29] have recently applied a self-attention mechanism to the field of computer vision, the results of which have proven that aggregating the global information of images can effectively improve the detection performance of the network. Therefore, we designed the GFE module shown in Fig. 5 according to the UAV image characteristics.

The GFE module has a hybrid structure that includes a transformer module based on multiheaded self-attention (MHSA) and a convolution. Some study found that the MHSA in a transformer block requires the memory and the number of calculations to be squared with the spatial and channel dimensions, thus causing a vast overhead for training and inference. The channel of the feature map contains a large amount of redundant information, and therefore, we used a 1×1 convolution in the GFE module to integrate the information in the channel dimension and to control the number of channels of the input feature map, thus reducing the number of network calculations and increasing the detection

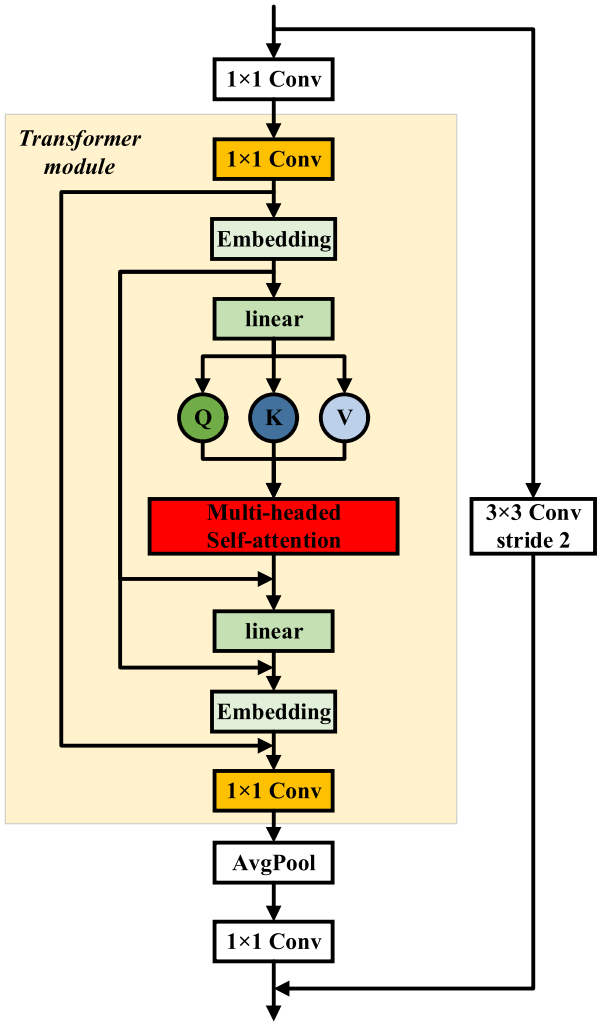


Fig. 5. Overall structure of GFE. The stride of the residual structure is 2 when downsampling and is 1 at other times.

speed. In addition, we skillfully designed the residual structure in the transformer module to realize an information exchange at different locations with few calculations. Using GFE, the network can infer the category of an object by modeling long-range dependencies, thus solving the problem in which the features of the same object in a UAV image vary greatly under different shooting conditions.

Although the transformer module can extract the global information of an image, it is difficult to extract the low-level features of some basic structures (e.g., corners and edges) in the image, and the number of calculations also increases due to the large size of the input image. Therefore, we used the GFE module at the back end of the network to model long-range dependencies and applied LFE, which is highly suitable for large-size feature maps, to extract the detailed features.

B. MFF Module

Initially, Faster-RCNN was proposed to detect objects with different scales within an entire map by setting anchors with different scales and ratios at each point on the final extracted feature map, and has achieved significant success. However, whereas the high-level semantic features in an image are easily extracted as the depth of the neural network increases, most of the detailed information is destroyed as the size of the feature map decreases, which is detrimental to detecting small objects that contain only simple semantic information. Therefore, some methods realize multiscale object detection by predicting objects of a different scale on different layer feature maps. However, poor positioning results are achieved for small objects due to the lack of location information of high-level semantic features in a shallow feature map. In view of the large changes in object scale in a UAV image, particularly for the majority of small objects, the MFF shown in Fig. 7 was designed to fuse high- and low-level semantic information of different layer feature maps and improve the accuracy of small object detection. In particular, we extracted different levels corresponding to feature maps from four appropriate locations of the backbone to predict objects of different scales. Among them, the feature map of the highest level contains the global information extracted by the GFE module, while the feature maps of other levels contain rich local information. MFF can enrich the feature content of different feature maps by fusing them, thus improving the detection accuracy of multiscale object detection. It can be seen from Table V that MFF can effectively improve the detection accuracy of the network for various scales.

Specifically, MFF first used 1×1 convolution to adjust the channels of these feature maps to a consistent level, and then, interpolate–add operations were conducted from top to bottom. The high-level semantic information of a deep network was introduced into a shallow feature map to improve the localization of small objects, and a 7×7 involution kernel was used to further integrate and communicate this knowledge. In addition, MFF also introduced a bottom-up supply channel to supplement the missing detailed information in the deep feature maps, improving the network's anti-interference ability and detection ability for large-scale objects. Finally, five different scales of features {P2, P3, P4, P5, P6} can be produced through an MFF.

C. Rotated RPN

We designed a rotated RPN based on a midpoint offset representation to generate high-quality rotated proposals. Specifically, we assigned three horizontal anchors with three aspect ratios of 1:2, 1:1, and 2:1 at each spatial position of these features. The pixel areas occupied by these anchors on {P2, P3, P4, P5, P6} are $\{32^2, 64^2, 128^2, 256^2, 512^2\}$, respectively. Each anchor a can be represented by a 4-D vector (a_x, a_y, a_w, a_h), where (a_x, a_y) are the coordinates of the anchor center point, and a_w and a_h are its width and height, respectively. This is followed by two sibling branches: one is used to estimate the object score for each rotated proposal, and the other is used to output the offset $\delta = (\delta_x, \delta_y, \delta_w, \delta_h, \delta_\alpha, \delta_\beta)$ of the proposals relative to the anchors. The rotated proposal can be obtained through the following formula to decode the regression structure:

$$\begin{cases} x = \delta_x \cdot a_w + a_x, & y = \delta_y \cdot a_h + a_y \\ w = a_w \cdot e^{\delta_w}, & h = a_h \cdot e^{\delta_h} \\ \Delta\alpha = \delta_\alpha \cdot w, & \Delta\beta = \delta_\beta \cdot h \end{cases} \tag{6}$$

where (x, y) are the central coordinates of the predicted rotated proposal; w and h represent the width and height of

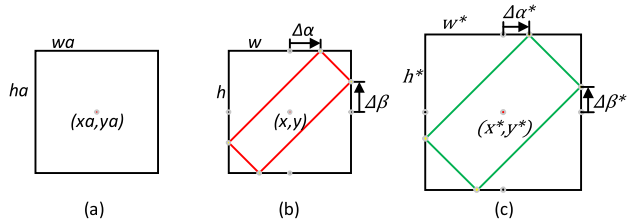


Fig. 6. Illustration of the RPN regression branch. The yellow dot indicates the vertex of the rotated bounding box, and the black dot is the midpoint of the edge: (a) anchor box; (b) predicted box; and (c) ground-truth box.

the external rectangle box of the predicted rotated proposal, respectively; and $\Delta\alpha$ and $\Delta\beta$ are the offsets of the proposal vertex relative to the top and right sides of the external rectangle, respectively. Finally, we obtain the rotation proposals $(x, y, w, h, \Delta\alpha, \Delta\beta)$, as shown in Fig. 6.

Without any restrictions, we would have obtained more than 100 000 proposals through the rotated RPN. However, most of them would be redundant, which is obviously not conducive to network training. Therefore, we assigned anchors that meet the following two conditions as positive samples: 1) an anchor has the highest intersection over union (IoU) with a ground-truth box or 2) the IoU between an anchor and any ground-truth box is higher than 0.7. In general, the second condition is sufficient to determine positive samples; however, considering that the second condition cannot find a positive sample in some special cases, we still adopt the first condition. An anchor is assigned as a negative sample when its IoU with all ground-truth boxes is less than 0.3, and all other anchors are discarded.

Based on these definitions, our loss function is defined as

$$L_1(\{p_i\},\{\delta_i\}) = \frac{1}{N}\sum_i F_{\text{cls}}(p_i, p_i^*) + \frac{1}{N} p_i^* \sum_i F_{\text{reg}}(\delta_i, \delta_i^*). \tag{7}$$

Here, i and N are the indexes of the anchors and the total number of samples in a mini-batch, respectively. In addition, p_i^* is the ground-truth label of the ith anchor, and it is 1 if the anchor is positive and zero if the anchor is negative. Moreover, δ_i^* is a 6-D vector $\delta_i^* = (\delta_x^*, \delta_y^*, \delta_w^*, \delta_h^*, \delta_\alpha^*, \delta_\beta^*)$ of the ground-truth bounding box associated with a positive sample. Finally, F_{cls} is the cross-entropy loss, and F_{reg} is the smooth $L1$ loss.

For a regression of the bounding box (see Fig. 6), we adopted the parameterizations of the six coordinates as follows:

$$\begin{cases} \delta_x = (x - x_a)/w_a, & \delta_y = (y - y_a)/h_a \\ \delta_w = \log(w/w_a), & \delta_h = \log(h/h_a) \\ \delta_\alpha = \Delta\alpha/w, & \delta_\beta = \Delta\beta/h \\ \delta_x^* = (x^* - x_a)/w_a, & \delta_y^* = (y^* - y_a)/h_a \\ \delta_w^* = \log(w^*/w_a), & \delta_h^* = \log(h^*/h_a) \\ \delta_\alpha^* = \Delta\alpha^*/w^*, & \delta_\beta^* = \Delta\beta^*/w^* \end{cases} \tag{8}$$

where (x, y), w, and h represent the center coordinates of the external rectangular box and its width and height, respectively. Variables x, x_a, and x^* denote the predicted box, anchor box, and ground-truth box, respectively (similar to y, w, and h). In addition, $\Delta\alpha$ and $\Delta\beta$ are the offsets of the top and right vertexes of the predicted box relative to the top and left sides of the external rectangle box, respectively. This can be considered a regression from the anchor box to a nearby ground-truth box.

When training the rotated RPN, since the number of negative samples is much higher than positive samples, the prediction results of the network may be biased toward negative samples. Therefore, we randomly sample 256 anchors in an image to compute the loss function of a mini-batch, where the sampled positive and negative anchors have a ratio of up to 1:1. If there are fewer than 128 positive samples in an image, we pad the mini-batch with negative ones.

TABLE I
GLF-NET OF DIFFERENT SIZES

	output size	GLF-Net S	GLF-Net	GLF-Net L
Input size		1024×1024		
FRM	256×256	×1		
LFE1_x	256×256	×1	×3	×3
LFE2_x	128×128	×2	×4	×4
LFE3_x	64×64	×4	×6	×23
SPP	64×64	×1		
GFE	32×32	×1	×3	×3
GFLOPs		215.1	235.87	319.02
Params(M)		48.52	55.82	74.26

D. Overview

We designed a novel high-precision UAV image rotated object detection network, as shown in Fig. 7, which is a two-stage detector of which the first stage generates a series of rotated proposals using the rotated RPN, and the second stage classifies and regresses these proposals. GLF-Net is trained in an end-to-end manner by jointly optimizing rotated RPN and detection head. During inference, some rotated RPN proposals highly overlap with each other. To reduce redundancy, we adopted nonmaximum suppression (NMS) on the proposals based on their classification scores. Considering the inference speed, we adopted the IoU threshold of 0.8 for NMS and finally remained 2000 proposals per level in the first stage. After NMS, we merged the remaining proposals from all levels and chose top 1000 ones based on their classification scores as the input of the second stage.

GLF-Net resizes an image of any size to a pixel resolution of 1024×1024 as the input and applies various types of image enhancement processing. The image first passes through the FRM four times for downsampling, retaining most of the underlying information. Next, the LFE, Spatial Pyramid Pooling (SPP), and GFE modules are used to construct the feature extraction network to effectively extract the local and global features of the feature map. In order to accelerate the convergence speed of the model training and avoid the problem of gradient disappearance or gradient explosion, we used batch normalization layer after all convolutional layers in the network. We obtained the three networks listed in Table I according to different stacking methods applied to these modules, and it can be seen that the number of parameters of our method is still extremely small. The feature maps of four

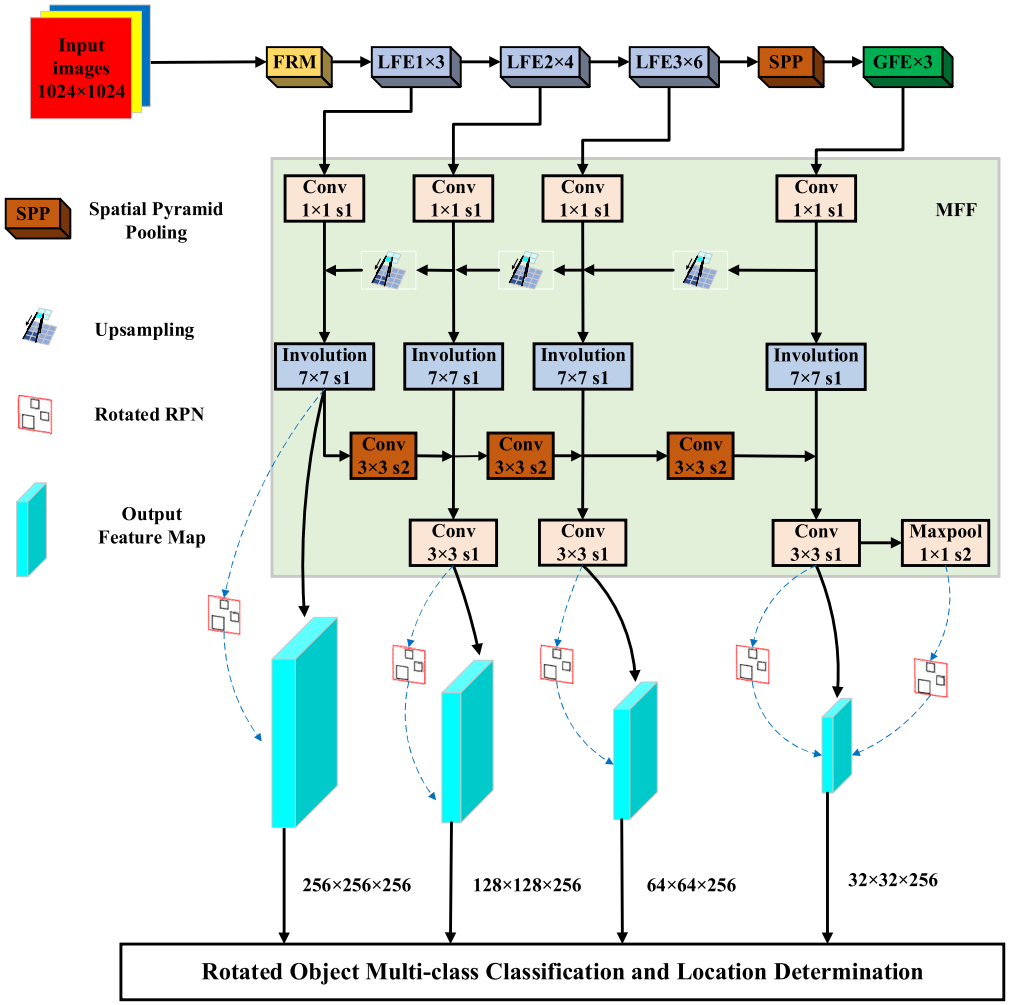


Fig. 7. Overall framework of GLF-Net. The five rotated RPN structures in the image are the same.

specific locations are then taken as the input of the MFF for feature interaction and multiscale prediction. It is helpful to classify and locate objects at different scales by fusing low-level feature maps with rich details and high-level feature maps with rich location information. The proposal generated by the rotated RPN is mapped to the corresponding feature map using the method shown in Fig. 7, and finally, the GLF-Net head is applied for fine detection.

IV. Experiments

Experiments were conducted on a server using a Ubuntu16.04 LTS, which was equipped with an Intel[1] Core I7-6950X CPU and four NVIDIA GeForce GTX 1080Ti graphics cards with 11 GB of memory. Our specific experimental environment is python3.7.11, pytorch1.6.0 torchvision0.7.0, opencv4.5.3, and cuda10.2. We used average precision (AP) and mAP to evaluate the accuracy of object detection and frames per second (FPS) to evaluate detection speed.

A. Datasets

The public datasets currently used for rotated object detection (e.g., DOTA, HRSC2016, and UCAS-AOD) are made up of satellite remote sensing images, their object features are different from those of UAV images, and thus, they are not completely suitable for UAV scenes. To evaluate the real detection performance of UAV images, we used a UAV to collect 6534 real images with a pixel resolution of 4056 × 3040 in dozens of locations in cities and suburbs. The RO-UAV was obtained by annotating it with a rotated annotation box. Some samples of the dataset are shown in Fig. 8. The dataset contains approximately 69 873 instances with various directions, proportions, and shapes, as well as three categories, i.e., car, truck, and special. Table II lists the numbers of category labels in the dataset. We used 50% of the data for training, 20% for validation, and the remaining 30% for testing.

TABLE II
Number of Images and Labels by Category in the Dataset

Class	Car	Truck	Special	Total
Images	6441	2346	1497	6534
Labels	64389	3603	1881	69873

B. Results on RO-UAV

The stochastic gradient descent (SGD) method was used to train the network. The initial learning rate was set to 0.005,

[1]Trademarked.

TABLE III
COMPARISON WITH THE STATE-OF-THE-ART METHODS ON THE RO-UAV DATASET

Method	Backbone	Car	Truck	Special	mAP	FPS
S^2ARN [44]	Resnet50	90.6	76.5	76.6	81.2	11.2
	Resnet101	90.6	76.0	73.2	79.9	9.2
R^3Det	Resnet50	89.7	73.5	44.6	69.3	2.3
	Resnet101	89.7	72.3	50.4	70.8	1.0
Faster-RCNN	Resnet50	90.18	78.02	74.25	80.82	9.1
	Resnet101	90.18	78.65	72.49	80.44	7.8
RetinaNet	Resnet50	90.40	62.11	53.95	68.82	11.7
	Resnet101	90.45	60.94	52.13	67.84	9.6
YOLO_DOTA_OBB (CSL)	Yolov5l	89.04	70.83	63.45	74.44	7.5
	Yolov5x	89.33	72.15	71.86	77.78	5.8
Roi-transformer	Resnet50	90.62	79.08	77.43	82.37	7.7
	Resnet101	90.59	80.89	77.53	83.00	6.2
Oriented R-CNN	Resnet50	90.39	74.85	66.84	77.36	10.6
	Resnet101	90.43	75.77	70.37	78.86	8.1
ReDet	ReResnet50	90.33	**84.39**	75.87	83.53	2.9
Gliding vertex	Resnet50	90.12	72.06	64.00	75.39	9.4
	Resnet101	90.08	72.02	63.24	75.11	6.9
GLF-Net S		90.62	82.53	83.90	85.68	8.7
GLF-Net		90.60	83.53	85.43	86.52	7.8
GLF-Net L		**91.59**	83.91	**86.58**	**87.36**	6.1

Fig. 8. Examples of the RO-UAV dataset.

the momentum was 0.9, and the weight decay was 0.0001. Unless otherwise noted, we trained all models for 72 epochs without using pretraining weights. To ensure the stability of the training process, the learning rate was decayed by 0.1 at {48, 66} epochs. We used two 1080Ti GPUs with a total batch size of 4 for train and a single 1080Ti GPU for inference. Further, various data enhancement methods such as random flip and color change were used to enhance and expand the image in the dataset.

1) Comparison With Other Methods: We compared our GLF-Net with a current mainstream rotated object detection method on the RO-UAV dataset. Table III lists the detailed results. YOLO_DOTA_OBB is implemented based on a CSL, and we trained 200 epochs using the default parameters. All other methods are implemented based on mmdetection2.10.0. For Faster-RCNN and RetinaNet, we added the angle regression branch of GLF-Net to their detection head, allowing them to be used in the detection of rotated objects.

The results show that our method achieved the highest results for most of the categories. The normal version of GLF-Net and a deeper version of GLF-Net L achieved a mAP of 86.52% and 87.36%, respectively. Even the lightweight version of GLF-Net S with an extremely shallow depth achieved better results than all other methods using ResNet101. In addition, it can be seen from the results of detection speed that our method achieves a good balance between detection speed and detection accuracy.

2) Visualization Results on the RO-UAV Dataset: Fig. 9 shows the results for the RO-UAV dataset. The detection results show that the proposed method can accurately determine the real direction and contour of an object among dense objects and achieves a high multiscale detection capability.

3) Ablation Studies: We conducted a series of ablation experiments on RO-UAV to evaluate the effectiveness of the proposed modules. Resnet50 + FPN is the baseline structure without the addition of any other modules. Table IV lists the performance of different modules used.

To ensure that the network calculation does not explode, we substituted the FRM with a 7×7 convolution and 3×3 maximum pooling layer with a stride of 2 to adjust the input features to the desired size. We found that the FRM improved the average accuracy of GLF-Net by 2% mAP and 4.56% mAP over the baseline. Then, ablation experiments were conducted using a bottleneck block in ResNet instead of an LFE module.

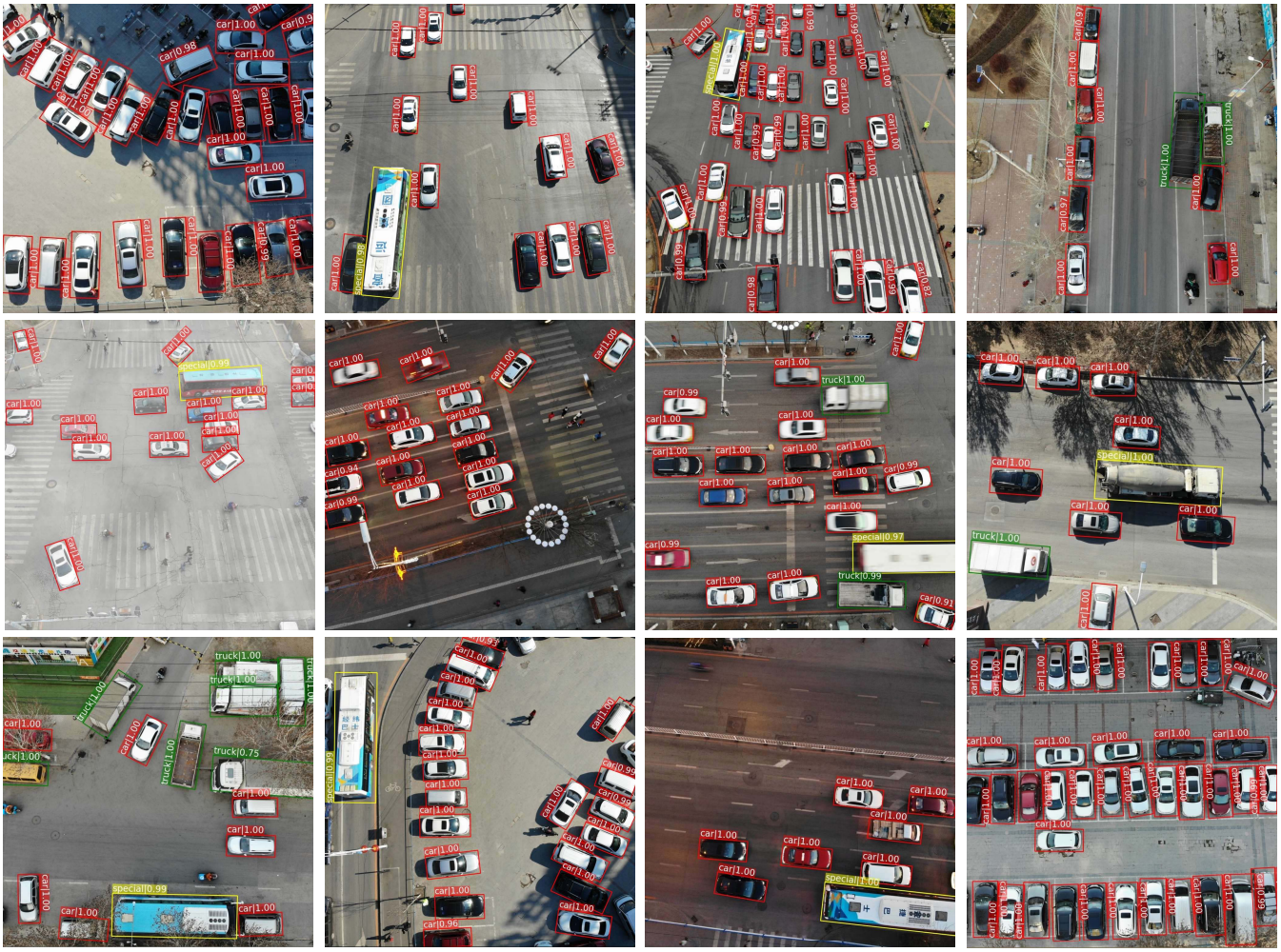


Fig. 9. Examples of detection results on the RO-UAV dataset.

TABLE IV
EFFECTS OF DIFFERENT NETWORK MODULES

Methods	FRM	LFE	SPP	GFE	MFF	Car	Truck	Special	mAP
baseline	-	-	-	-	-	90.39	74.85	66.84	77.36
FRM	√	-	-	-	-	90.47	80.52	74.77	81.92
LFE	-	√	-	-	-	90.52	80.10	77.49	82.70
GFE	-	-	-	√	-	90.50	79.74	78.33	82.85
MFF	-	-	-	-	√	90.52	80.04	79.99	83.52
NO-FRM	-	√	√	√	√	90.59	81.08	81.90	84.52
NO-LFE	√	-	√	√	√	90.67	81.04	82.24	84.65
NO-SPP	√	√	-	√	√	**90.68**	81.87	81.37	84.64
NO-GFE	√	√	√	-	√	90.52	83.15	83.42	85.70
NO-MFF	√	√	√	√	-	90.53	81.30	75.90	82.58
GLF-Net	√	√	√	√	√	90.60	**83.53**	**85.43**	**86.52**

LFE improved the accuracy of GLF-Net by 1.87% mAP and 5.34% mAP over the baseline. With the addition of the SPP module, the detection accuracy of the GLF-Net was improved by 1.88% mAP. Then, we conducted ablation experiments on the GFE module using LFE instead of GFE to ensure the integrity of the network structure. Although the results showed

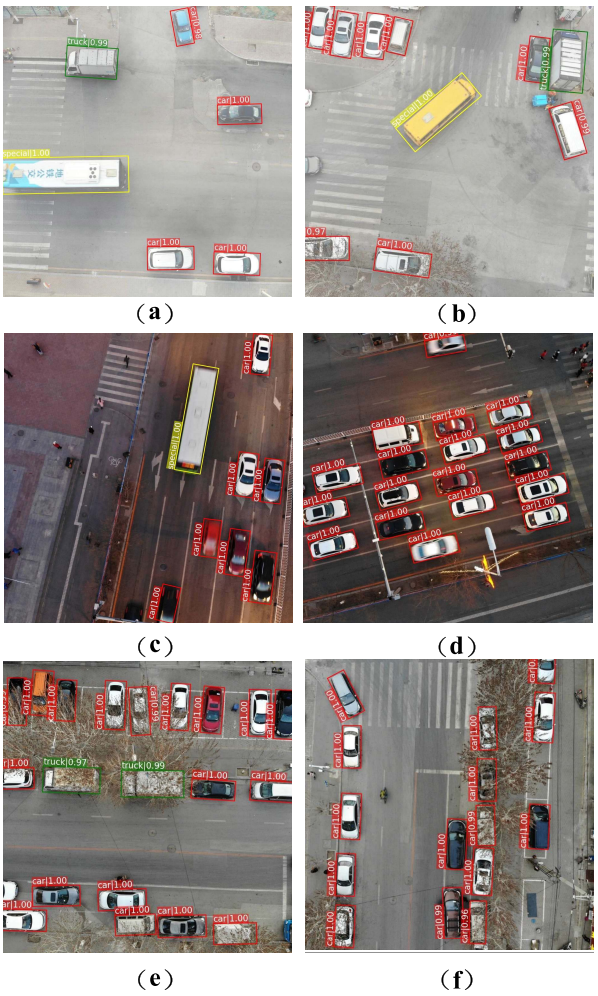


Fig. 10. Anti-interference experiments for different scenes. (a) and (b) Experimental results in haze weather. (c) and (d) Experimental results at night. (e) and (f) Experimental results of objects being occluded.

that GFE improved the detection accuracy by 0.82% mAP, its addition to the baseline improved the detection accuracy by a remarkable 6.09% mAP. Finally, we replaced MFF with an FPN structure and found that MFF improved the detection accuracy of GLF-Net by 3.94% mAP; in addition, adding it to the baseline could improve the accuracy by 6.16% mAP.

The rotated RPN is an important structure of GLF-Net that generates rotated proposals, and we set up experiments to verify its effectiveness, as shown in Table V. We used an RPN in a Faster-RCNN to replace the rotated RPN for comparison, the results of which show that the AP of all categories and the mAP of the method using the rotated RPN structure are higher, proving that the rotated RPN can effectively improve the detection accuracy of the rotated objects.

4) Anti-Interference Capability Experiment: We tested the anti-interference capability of GLF-Net under different environments. The experimental results are presented in Fig. 10. Although the imaging quality of a camera deteriorates in poor weather and insufficient lighting conditions, the results in Fig. 10(a) and (b) show that the network can still accurately detect objects in hazy weather. In addition, Fig. 10(c) and (d) indicates that GLF-Net can ensure a high detection accuracy at night. Serious occlusions have been a major challenge for object detection in remote sensing images,

TABLE V
COMPARISON OF ROTATED RPN WITH ORDINARY RPN

Method	RPN	Rotated RPN
Car	90.54	**90.60**
Truck	79.23	**83.53**
Special	81.62	**85.43**
mAP	83.79	**86.52**

TABLE VI
COMPARISON WITH THE STATE-OF-THE-ART METHODS ON HRSC2016 DATASET

Method	Backbone	mAP(07)	mAP(12)
S^2ANet	Resnet-101	90.17	95.01
R^3Det	Resnet-101	89.26	96.01
Faster-RCNN	Resnet-101	77.30	81.40
RetinaNet	Resnet-101	83.06	84.79
CSL	Resnet-50	89.62	96.10
RoI Transformer	Resnet-101	86.20	-
Oriented R-CNN	Resnet-50	90.40	96.50
ReDet	ReResnet50	90.46	**97.63**
Gliding Vertex	Resnet-101	88.20	-
ours		**90.62**	96.95

particularly in UAV images. However, based on the results in Fig. 10(e) and (f), our method has a strong antiocclusion capability. Anti-interference experiments show that GLF-Net has a high detection accuracy under various harsh environments, which ensures that the UAV can meet the requirements of object tracking, postdisaster search and rescue operations, smart cities, and other practical scenarios.

C. Comparison Results on HRSC2016 and UCAS-AOD

To evaluate the generalization of GLF-Net under other scenarios, we conducted comparative experiments on the public datasets HRSC2016 and UCAS-AOD. HRSC2016 and UCAS-AOD are common datasets in remote sensing image rotated object detection, and their data feature distribution is similar to UAV images. To ensure reliability, all data are from relevant studies [36], [37], [38], [39], [40] or tested in the same experimental environment as in the RO-UAV dataset.

1) Results on HRSC2016: HRSC2016 is a ship dataset annotated with a rotated box, which contains 1061 images ranging in pixel resolutions of 300 × 300 to 1500 × 900. We used 621 of the images for training and the remaining 440 for testing. All images were resized to 1024 × 1024 without changing the aspect ratio. We used the mAP of PASCAL2007 and 2012 metrics to evaluate the detection accuracy, the experimental results of which are shown in Table VI. Our method achieved the optimal accuracy.

2) Results on UCAS-AOD: UCAS-AOD contains 1510 remote sensing images with a size of approximately 659 × 1280, including a total of 14 596 planes and cars. We used 80% of the images for training and the remaining 20% for testing. As the experimental results presented in

TABLE VII
COMPARISON WITH THE STATE-OF-THE-ART METHODS ON UCAS-AOD DATASET

Method	mAP	Plane	Car
S^2ARN	94.90	97.60	92.20
R^3Det	96.17	98.20	94.14
Faster-RCNN	92.04	97.23	86.84
RetinaNet	94.16	97.52	90.79
CSL	95.07	97.22	92.93
RoI Transformer	93.50	97.01	89.99
Oriented R-CNN	90.78	96.78	84.95
ReDet	90.30	90.80	89.80
Gliding Vertex	92.97	97.87	88.07
ours	**97.00**	**98.97**	**95.02**

Table VII show, our method achieved the best detection results for each category.

Based on the experimental results on the public datasets, our method achieves a good generalization and can be used for the high-precision detection of rotated objects under various scenarios.

V. CONCLUSION

In this article, a high-precision rotated object detector, called GLF-Net, was proposed for UAV images with complex scenes, dense objects, small objects, and rotated objects. Aiming at overcoming a shortcoming in which the current object detector cannot effectively extract features from dense objects, we designed a new backbone that includes FRM, LFE, and GFE. Among them, FRM reduces the size of the input feature map through an information recombination. LFE and GFE use an involution and self-attention to extract local and global information of the feature map, respectively, thus achieving an effective feature extraction of dense objects and complex scenes. To overcome the difficulty of detecting many small objects, we designed an MFF module to further enhance the expression of the features by fusing different feature levels, thus improving the detection accuracy of small objects. To determine the real direction and contour of an object from among other disorganized objects, the rotated RPN is proposed to generate the rotated proposals. The experimental results show that GLF-Net achieved an 86% mAP on the RO-UAV dataset we created, which is much higher than that of other current object detection algorithms. Moreover, GLF-Net achieved 96.95% and 97% mAP on the public datasets HRSC2016 and UCAS-AOD, respectively, which proves that the method has a high generalization and can meet the detection requirements of UAVs under various complex environments. In the future, we will evaluate more challenging datasets, such as postdisaster areas for search and rescue operations or urban object tracking using UAVs.

REFERENCES

[1] N. Tijtgat, W. Van Ranst, B. Volckaert, T. Goedeme, and F. De Turck, "Embedded real-time object detection for a UAV warning system," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 2110–2118.

[2] Y. Zhou, T. Rui, Y. Li, and X. Zuo, "A UAV patrol system using panoramic stitching and object detection," *Comput. Electr. Eng.*, vol. 80, Dec. 2019, Art. no. 106473.

[3] H. Zhang, M. Sun, Q. Li, L. Liu, M. Liu, and Y. Ji, "An empirical study of multi-scale object detection in high resolution UAV images," *Neurocomputing*, vol. 421, pp. 173–182, Jan. 2021.

[4] P. Mittal, R. Singh, and A. Sharma, "Deep learning-based object detection in low-altitude UAV datasets: A survey," *Image Vis. Comput.*, vol. 104, Dec. 2020, Art. no. 104046.

[5] P. Chen, Y. Dang, R. Liang, W. Zhu, and X. He, "Real-time object tracking on a drone with multi-inertial sensing data," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 1, pp. 131–139, Jan. 2018.

[6] C. Huang, P. Chen, X. Yang, and K.-T.-T. Cheng, "REDBEE: A visual-inertial drone system for real-time moving object detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 1725–1731.

[7] X. Wang *et al.*, "Fast and accurate, convolutional neural network based approach for object detection from UAV," in *Proc. 44th Annu. Conf. IEEE Ind. Electron. Soc.*, Oct. 2018, pp. 3171–3175.

[8] F. Kendoul, I. Fantoni, and K. Nonami, "Optic flow-based vision system for autonomous 3D localization and control of small aerial vehicles," *Robot. Auto. Syst.*, vol. 57, nos. 6–7, pp. 591–602, Jun. 2009.

[9] S. Saripalli, J. F. Montgomery, and G. S. Sukhatme, "Vision-based autonomous landing of an unmanned aerial vehicle," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 3, May 2002, pp. 2799–2804.

[10] L. Meier, P. Tanskanen, L. Heng, G. H. Lee, F. Fraundorfer, and M. Pollefeys, "PIXHAWK: A micro aerial vehicle design for autonomous flight using onboard computer vision," *Auto. Robots*, vol. 33, nos. 1–2, pp. 21–39, Aug. 2012.

[11] D. Scaramuzza *et al.*, "Vision-controlled micro flying robots: From system design to autonomous navigation and mapping in GPS-denied environments," *IEEE Robot. Autom. Mag.*, vol. 21, no. 3, pp. 26–40, Sep. 2014.

[12] T. Moranduzzo and F. Melgani, "A SIFT-SVM method for detecting cars in UAV images," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2012, pp. 6868–6871.

[13] T. Moranduzzo and F. Melgani, "Detecting cars in UAV images with a catalog-based approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 10, pp. 6356–6367, Oct. 2014.

[14] T. Moranduzzo, F. Melgani, Y. Bazi, and N. Alajlan, "A fast object detector based on high-order gradients and Gaussian process regression for UAV images," *Int. J. Remote Sens.*, vol. 36, no. 10, pp. 2713–2733, May 2016.

[15] Y. Xu, G. Yu, Y. Wang, X. Wu, and Y. Ma, "A hybrid vehicle detection method based on viola-jones and HOG + SVM from UAV images," *Sensors*, vol. 16, no. 8, p. 1325, Aug. 2016.

[16] N. Ammour, H. Alhichri, Y. Bazi, B. Benjdira, N. Alajlan, and M. Zuair, "Deep learning approach for car detection in UAV imagery," *Remote Sens.*, vol. 9, no. 4, p. 312, 2017.

[17] Y. Bazi and F. Melgani, "Convolutional SVM networks for object detection in UAV imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3107–3118, Jun. 2018.

[18] M. B. Bejiga, A. Zeggada, and F. Melgani, "Convolutional neural networks for near real-time object detection from UAV imagery in avalanche search and rescue operations," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2016, pp. 693–696.

[19] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[20] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[21] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[22] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.

[23] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 21–37.

[24] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[25] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[26] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9627–9636.
[27] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," 2019, *arXiv:1904.07850*.
[28] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO series in 2021," 2021, *arXiv:2107.08430*.
[29] A. Dosovitskiy *et al.*, "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.
[30] Z. Zhang, W. Guo, S. Zhu, and W. Yu, "Toward arbitrary-oriented ship detection with rotated region proposal and discrimination networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 11, pp. 1745–1749, Aug. 2018.
[31] J. Ding, N. Xue, Y. Long, G.-S. Xia, and Q. Lu, "Learning RoI transformer for detecting oriented objects in aerial images," 2018, *arXiv:1812.00155*.
[32] Y. Xu *et al.*, "Gliding vertex on the horizontal bounding box for multi-oriented object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 4, pp. 1452–1459, Apr. 2021.
[33] W. Li, Y. Chen, K. Hu, and J. Zhu, "Oriented RepPoints for aerial object detection," 2021, *arXiv:2105.11111*.
[34] X. Xie, G. Cheng, J. Wang, X. Yao, and J. Han, "Oriented R-CNN for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 3520–3529.
[35] J. Ma *et al.*, "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Trans. Multimedia*, vol. 20, no. 11, pp. 3111–3122, Mar. 2018.
[36] J. Han, J. Ding, N. Xue, and G.-S. Xia, "ReDet: A rotation-equivariant detector for aerial object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 2786–2795.
[37] X. Yang and J. Yan, "Arbitrary-oriented object detection with circular smooth label," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 677–694.
[38] X. Yang, L. Hou, Y. Zhou, W. Wang, and J. Yan, "Dense label encoding for boundary discontinuity free rotation detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 15819–15829.
[39] X. Yang *et al.*, "SCRDet: Towards more robust detection for small, cluttered and rotated objects," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8232–8241.
[40] X. Yang *et al.*, "R2CNN++: Multi-dimensional attention based rotation invariant detector with robust anchor strategy," 2018, *arXiv:1811.071262*.
[41] X. Yang, J. Yan, Z. Feng, and T. He, "R3Det: Refined single-stage detector with feature refinement for rotating object," 2019, *arXiv:1908.05612*.
[42] D. Li *et al.*, "Involution: Inverting the inherence of convolution for visual recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 12321–12330.
[43] A. Vaswani *et al.*, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.
[44] S. Bao, X. Zhong, R. Zhu, X. Zhang, Z. Li, and M. Li, "Single shot anchor refinement network for oriented object detection in optical remote sensing imagery," *IEEE Access*, vol. 7, pp. 87150–87161, 2019.

Tao Ye received the B.S. degree in measurement and control technology and instrumentation from the China University of Mining and Technology (CUMT), Xuzhou, China, in 2009, the M.S. degree in mechanical and electronic engineering from CUMT, Beijing, China, in 2012, and the Ph.D. degree in measurement technology and instruments from the Key Laboratory of Precision Opto-Mechatronics Technology, Ministry of Education, Beihang University, Beijing, in 2016.

From March 2016 to March 2019, he worked as an Engineer with the Beijing Institute of Remote Sensing and Equipment, Beijing. He is currently a Senior Engineer with the School of Mechanical Electronic and Information Engineering, CUMT, Beijing. His current research interests include deep learning and traffic detection.

Wenyang Qin received the B.S. degree in mechanical engineering from Henan Polytechnic University, Jiaozuo, China, in 2020. He is currently pursuing the M.S. degree in mechanical engineering with the China University of Mining and Technology, Beijing, China.

His research interests include artificial intelligence, deep learning, and unmanned aerial vehicle remote sensing.

Yunwang Li received the B.S. and Ph.D. degrees from the China University of Mining and Technology (CUMT), Beijing, China, in 2003 and 2010, respectively.

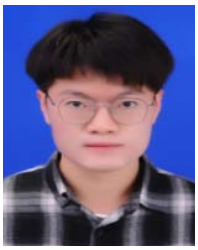
From 2010 to 2020, he taught and engaged in research in the fields of intelligent mine equipment and mine robot technology with the School of Mechatronic Engineering, CUMT. Since 2021, he has been engaged in scientific research with the Key Laboratory of Intelligent Mining and Robotics, Ministry of Emergency Management, Beijing. His current research interests include intelligent mining and mine robotics.

Shouan Wang received the B.S. degree in materials science and engineering from the China University of Mining and Technology, Beijing, China, in 2019, where he is currently pursuing the M.S. degree in mechanical engineering with the College of Mechanical and Electrical Engineering.

His current research interests include object detection and multisensor fusion.

Jun Zhang received the B.S. degree in mechanical engineering from the China University of Mining and Technology, Beijing, China, in 2020, where he is currently pursuing the M.S. degree in mechanical engineering.

His research interests include artificial intelligence, deep learning, and multiobject tracking.



Zongyang Zhao received the B.S. degree in mechanical engineering from the China University of Mining and Technology, Beijing, China, in 2020, where he is currently pursuing the M.S. degree in mechanical engineering.

His current research interests include deep learning, computer vision, and railway object detection.