An Efficient Multiband Infrared Small Objects Detection Approach for Low-Altitude Artificial Intelligence of Things

Yuhuai Peng, Jing Wang, Wenqian Wang, Lei Liu, *Member, IEEE*, Mohammed Atiquzzaman, *Life Senior Member, IEEE*, Mohsen Guizani, *Fellow, IEEE*, and Schahram Dustdar, *Fellow, IEEE*

***Abstract*—As a cutting-edge technology of low-altitude Artificial Intelligence of Things (AIoT), autonomous aerial vehicle object detection significantly enhances the surveillance services capabilities of low-altitude AIoT. However, the difficulty of object detection is exacerbated by the high proportion of small and obscure objects in the captured images. To address the mentioned challenges, we present an efficient multiband infrared small object detection approach for low-altitude intelligent surveillance services. First, we propose the multiband infrared image fusion algorithm based on cascade-GAN (MIF-CGAN), which produces fused images with high information entropy and high contrast. Then, the Transformer-based multiscale dense small object detection (MsDSOD) algorithm is proposed. The algorithm consists of the global–local object detection (G-LOD) network, the object dense area extraction (O-DAE) module, and the weighted boxes fusion (WBF) module. It extracts small objects features at different scales from infrared images and fuses the global and local detection results to accurately identify small objects in dense scenes. Furthermore, compared to the traditional algorithms, the mean average precision (mAP) of MsDSOD is improved by 0.80% and the average precision in small object detection (AP_s) is improved by 0.72%. The proposed algorithm is optimally suited to deal with complex scenes with dense small objects and background occlusion.**

***Index Terms*—Artificial Intelligence of Things (AIoT), low-altitude surveillance services, multiband infrared images, object detection, transformer network.**

Received 18 November 2024; revised 8 February 2025; accepted 18 February 2025. Date of publication 20 February 2025; date of current version 9 June 2025. This work was supported in part by the Joint fund of Equipment Preresearch and Ministry of Education under Grant 8091B032131; in part by the Aeronautical Science Foundation of China under Grant 2020Z066050001; in part by the China Postdoctoral Science Foundation under Grant 2024T170695; and in part by the National Natural Science Foundation of China under Grant 62271391. *(Corresponding author: Yuhuai Peng.)*

Yuhuai Peng is with the School of Computer Science and Engineering, Northeastern University, Shenyang 110819, and also with the Strategic Research Department, Zhiyuan Research Institute, Hangzhou 310024, China (e-mail: pengyuhuai@mail.neu.edu.cn).

Jing Wang and Wenqian Wang are with the School of Computer Science and Engineering, Northeastern University, Shenyang 110819, China (e-mail: 2110651@stu.neu.edu.cn; m18778995803@163.com).

Lei Liu is with the Guangzhou Institute of Technology, Xidian University, Guangzhou 510555, China, and also with Shanxi Key Laboratory of Information Communication Network and Security, Xi'an University of Posts and Telecommunications, Xi'an 710121, Shanxi, China (e-mail: tianjiaoliulei@163.com).

Mohammed Atiquzzaman is with the School of Computer Science, University of Oklahoma, Norman, OK 73019 USA (e-mail: atiq@ou.edu).

Mohsen Guizani is with the Department of Machine Learning, Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE (e-mail: mguizani@ieee.org).

Schahram Dustdar is with the Distributed Systems Group, Technische Universität Wien, 1040 Vienna, Austria, and also with ICREA, Universitat Pompeu Fabra Barcelona, 08002 Barcelona, Spain (e-mail: dustdar@dsg.tuwien.ac.at).

Digital Object Identifier 10.1109/JIOT.2025.3544258

I. Introduction

TYPICAL of emerging productivity, the low-altitude Artificial Intelligence of Things (AIoT) seamlessly integrates Internet of Thing (IoT), artificial intelligence (AI), cloud computing, and other cutting-edge information technologies [1]. It has been widely used in a range of applications, from agricultural monitoring and urban planning to traffic management, disaster rescue, and environmental protection [2]. As the core of low-altitude AIoT, autonomous aerial vehicle (AAV) is steadily advancing toward the peak of fully autonomous awareness and control. Through the leveraging of object detection, AAVs have acquired the ability to independently sense and understand surrounding environment. Object detection is essential to enhance the autonomous of AAVs in situational awareness, obstacle avoidance, and object tracking. Nevertheless, the complexity of object detection tasks is combined to the constraints of AAV computational capabilities, the limited bandwidth of air-to-ground communication networks, and the inherent characteristics of AAV-captured images [3]. Efficient dense small object detection techniques improve the speed and precision of AAV perception, which is becoming prerequisite for the autonomy and intelligence of the low-altitude AIoT.

AAV-captured images are characterized by nonuniform spatial distributions and high proportion of small objects, with occlusion and overlap among dense objects. In contrast to ground-based images, AAV-captured image boasts a wide field of view providing a wealth of contextual information. Increasing scene complexity and object diversity leads to more noise interference in object detection task. Then, images captured at wide ranges often appear as punctiform features with a small percentage of effective information, and are particularly difficult to detect accurately due to factors, such as mutual overlap of densely objects, background occlusion, or the light changes [4]. In addition, AAVs have limited computing capability and energy to support high-resolution object detection algorithms. The above makes dense small object detection of low-altitude AIoT a challenging mission.

As a common sensing method, AAV-borne infrared thermography has a wide imaging range and all-weather

detection [5]. However, the nature of infrared imaging typically produces images without discernible shape, color, and texture cues. To make matters worse, the relatively large pixel sizes and low spatial resolutions inherent in infrared imaging sensors result in blurred boundaries between objects and their surroundings. In addition, complex background clutter and ubiquitous random noise increase the difficulty of AAV infrared small object detection.

To address the above challenges, conventional infrared small object detection methods rely on suppressing extraneous background information, eliminating clutter and noise [6], or enhancing object information through contrast enhancement [7]. While these approaches perform well in relatively homogenous detection scenarios, they fall short when faced with the real-world environments characterized by complex backgrounds. In contrast, deep learning techniques avoid the need for manually generated features, instead relying on neural networks to automatically extract relevant features directly from original images [8]. The method improves model precision by using large amounts of training data, and accelerates the model training and detection process by using an end-to-end learning approach. However, small objects in dense scenes are unevenly distributed and lack obvious features, such as shape and texture. This makes it difficult for existing methods to accurately detect dense small objects in infrared images.

Therefore, to improve the object detection accuracy of AAVs, we propose a multiscale dense small object detection (MsDSOD) approach based on AAV-borne multiband infrared sensing. The main contributions of this article can be summarized as follows.

1) This article presents a fusion-based dense small object detection framework, including multiband infrared image fusion algorithm based on cascade-GAN (MIF-CGAN) and MsDSOD. The MIF-CGAN on the AAV generates composite infrared images with enriched feature representations, which are transmitted to the ground computing center. Then, the MsDSOD detects small objects in the fused images. This framework significantly improves the accuracy of AAV-based small object detection.
2) To enhance the quality of AAV-borne infrared images, we present the MIF-CGAN. This approach combines a denoising generative adversarial network (DnGAN) and a fusion generative adversarial network (FuGAN), exploiting the complementary information provided by each band to significantly improve the feature extraction performance.
3) To improve the precision of densely small object detection, the transformer-based MsDSOD algorithm is proposed. The algorithm consists of global–local object detection (G-LOD) network, object dense area extraction (O-DAE) module, and weighted boxes fusion (WBF) module. The integration of global contextual information with local detail improves the accuracy of dense small object identification.
4) Compared with GAN-FM, the average gradient (AG) of MIF-CGAN is improved by 4.78%. And compared to SCSDet, the mean average precision (mAP) of MsDSOD is improved by 0.80%, and the average precision in small object detection (AP_s) is improved by 0.72%. The proposed algorithm is optimally suited to deal with complex scenes for dense small objects and background occlusion.

The remainder of this article is organized as follows. In Section II, related works are introduced. In Section III, the system model is given. In Section IV, multiband infrared image fusion network based on Cascade-GAN is given. In Section V, the Transformer-based multiscale dense small target detection algorithm is given. In Section VI, the performance metrics are analyzed by experiments. Finally, Section VII concludes this article.

II. Related Works

In recent years, scholars from both domestic and international communities have conducted in-depth explorations on related technologies, such as image fusion and object detection. Ma et al. [9] introduced generative adversarial networks (GANs) into the image fusion field for the first time, and utilized the adversarial game between generator and discriminator to generate high-quality fused images. Wang et al. [10] proposed a self-supervised fusion model based on comparison learning self-supervised fusion model, which guides the backbone network to generate the fused image by estimating the feature compensation map of the infrared image. Yang et al. [11] designed a dual-stream bootstrap filtering network, which extracts the image features in the way of two independent data streams, preserving more background and detail information. Zhao et al. [12] proposed a multimodal image fusion network, the correlation-driven feature decomposition fusion, which optimizes the extraction and fusion of cross-modal features through specific techniques, significantly improves the quality of the fused images. In order to make the generator capture comprehensive spatial information, Li et al. [13] integrated a multiscale attention mechanism in the generator and discriminator of GAN, so that the fusion network pays more attention to the typical regions of the source image to reconstruct the fusion map. However, fused multiband infrared images still have different degrees of defects, such as missing texture detail information, low contrast, and poor signal-to-noise ratio.

Deep learning techniques have become a mainstream method in the field of object detection. The backbone is a feature extractor for the object detection task and outputs a feature map of the image. Li et al. [14] proposed lightweight large selective kernel network (LSKNet) to fully utilize the a priori knowledge in small object scenarios and dynamically adjust the spatial sensing field, which improves the detection precision and at the same time reduces the number of parameters and computation of the model. Du et al. [15] proposed a global context-enhanced adaptive sparse convolutional network for efficient and low-latency object detection on computationally resource-constrained AAV platforms. Neck [16] mixes and combines image features to pass image features from the backbone to the prediction layer. However, in areas of the infrared image where objects are small and densely distributed, the object information is limited and there is mutual occlusion, which still increases the difficulty of detection.

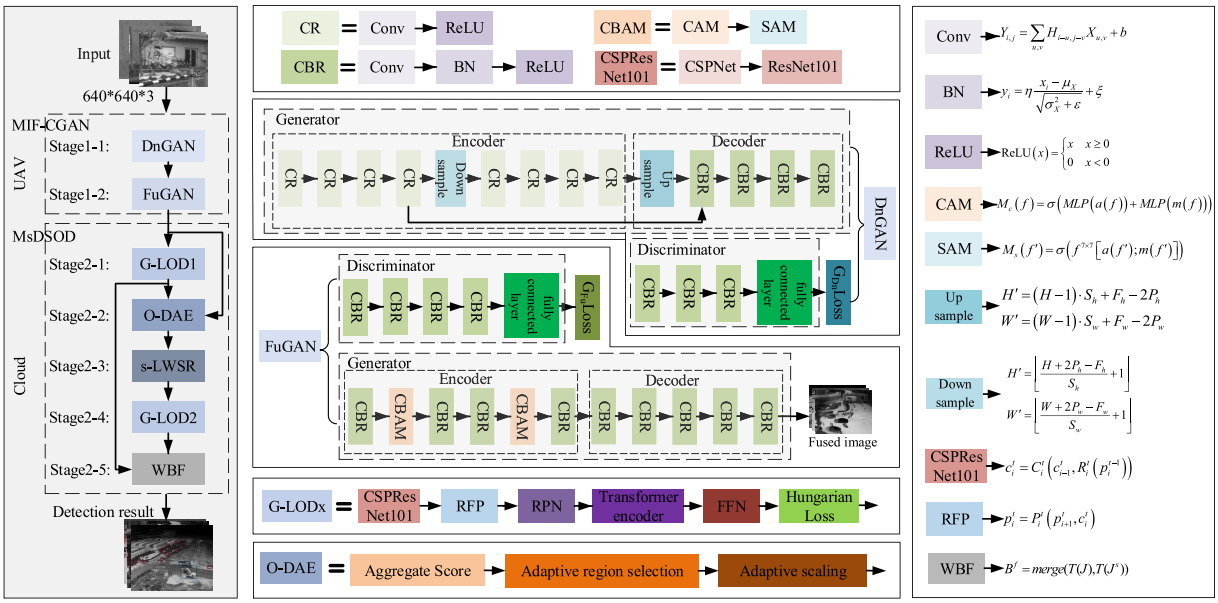


Fig. 1. Fusion-based dense small object detection architecture.

Currently, object detection frameworks are mainly categorized into single-stage detection and two-stage detection. The most representative single-stage detectors are the YOLO series [17], [18], [19]. Detection transformer (DETR) [20] based on Transformer achieves end-to-end detection through Hungarian bisection matching, eliminating manual operations, such as nonmaximal suppression. Ye et al. [21] proposed a Cascade-DETR approach to object detection, which improves the localization precision and calibration confidence of generic object detection. Xu et al. [22] proposed a dynamic prior along with the coarse-to-fine assigner, which effectively solves the label allocation problem in directional tiny object detection and improves the detection precision of tiny objects. Tian et al. [23] implemented a pixel-by-pixel object detection algorithm based on a fully convolutional network, which utilizes the idea of centrality to suppress low-quality prediction frames, and also achieves better detection results. The two-stage detector first detects each prediction frame by generating region suggestions and then refining them. The most representative of two-stage detection is the R-CNN family, including Fast R-CNN [24], Faster R-CNN [25], Cascade R-CNN [26], and Cascade Mask R-CNN [27].

III. System Model

Due to the limited computational capabilities of AAVs, high-precision object recognition algorithms cannot be deployed directly on board. This manuscript presents a fusion-based framework for dense detection of small objects, consisting of MIF-CGAN and MsDSOD. The MIF-CGAN component operates onboard the AAV, denoising and fusing multiple infrared images [near-infrared (NIR), mid-infrared (MIR), and long-infrared (LIR)] acquired from the same perspective to produce composite infrared images with enriched feature representations, which are then transmitted to a ground-based computing center. This fusion-based approach minimizes the need to transmit large amounts of raw infrared data, significantly reducing the communication burden between the AAV and the ground. The MsDSOD component runs in the cloud computing center, detecting densely packed small objects in the fused images and either presenting the results to the user or transmitting them back to the AAV. This two-component approach significantly improves the accuracy of AAV-based small object detection.

Since multiband infrared images all have their own characteristics, we use the AAV-borne sensor to acquire NIR, MIR and LIR images of the same scene. As shown in Fig. 1, to enhance the image quality, we propose the MIF-CGAN, which comprises DnGAN and FuGAN. The DnGAN is used to filter out noise from multiband infrared images of the same scene. The denoised image is inputted into FuGAN, which produce a fused image with distinguished features, thereby enhancing the image signal-to-noise ratio. This holistic approach exploits the complementary information provided by each band, increasing the effectiveness and detail of subsequent target detection analysis.

To improve the precision of infrared object detection, we propose an MsDSOD method. The method comprises G-LOD, O-DAE, super-lightweight super-resolution (s-LWSR), and WBF. G-LOD is used to detect the global objects of the input images. Subsequently, O-DAE extracts the dense regions of the objects. s-LWSR is used to super-resolve cropped local images. G-LOD is then used once more to obtain local detection results, which are combined with the global detection results in WBF to achieve the final detection results and the desired level of precision.

IV. Multiband Infrared Image Fusion Method Based on Cascade-GAN

In order to improve the quality of infrared images, the MIF CGAN network is proposed, as shown in Fig. 2. The denoised

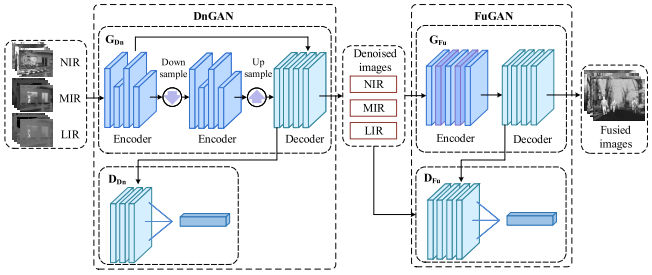


Fig. 2. MIIF-CGAN network.

images generated by DnGAN are used to guide the game-adversarial training of FuGAN to obtain high-quality fused images.

A. Network Structure of DnGAN

Stage 1-1: NIR, MIR, and LIR images are taken as inputs, and the denoised multiband infrared images are generated by the game confrontation between the generator and the discriminator of DnGAN.

To express more clearly, we propose a formalized representation of the fusion process. Given a pair of aligned multi-infrared image I_{NIR}, I_{MIR}, and I_{LIR}, the goal is to synthesize a fused image I_{fused}

$$f_e(\text{NIR}) = \{\phi_N^1, \ldots, \phi_N^m, \ldots, \phi_N^M\} \tag{1}$$

$$f_e(\text{MIR}) = \{\phi_M^1, \ldots, \phi_M^m, \ldots, \phi_M^M\} \tag{2}$$

$$f_e(\text{LIR}) = \{\phi_L^1, \ldots, \phi_L^m, \ldots, \phi_L^M\} \tag{3}$$

where $f_e(\cdot)$ denotes the extraction function learned by the encoder. ϕ_N, ϕ_M, and ϕ_L represent the feature maps extracted from multi-infrared image I_{NIR}, I_{MIR}, and I_{LIR}, respectively. M is the number of feature maps. The corresponding extracted features of the corresponding images are fused

$$\{\phi_f^1, \ldots, \phi_f^M\} = \{f_\phi\left(\phi_N^1, \phi_M^1, \phi_L^1\right), \ldots, f_\phi\left(\phi_N^M, \phi_M^M, \phi_L^M\right)\} \tag{4}$$

where ϕ_f represents the fused feature maps. f_ϕ denotes the fusion process. As reconstruction is the inverse process of extraction, we employ a decoder to learn the inverse transformation of $f_e(\cdot)$. $f_d(\cdot)$ is the reconstruction process. The fused image is generated as

$$I_{\text{fused}} = f_d\left(\phi_f^1, \ldots, \phi_f^M\right). \tag{5}$$

DnGAN is used to denoise the input image. The generator G_{Dn} consists of an encoder and a decoder. In the encoder, the receptive field of the convolutional kernel is expanded by upsampling and downsampling, so that the model can make full use of the context information to realize the secondary extraction of image features. Denoised images is reconstructed in the decoder. The network structure parameters specific to the generator G_{Dn} are shown in Table I.

As shown in Fig. 3, the encoder of the G_{Dn} consists of four convolutional layers. Each layer of the network comprises a 3×3 convolutional kernel and a rectified linear unit (ReLU). The stacked 3×3 convolutional kernels require a few parameters, which can reduce the complexity of the model

TABLE I
NETWORK STRUCTURE PARAMETERS OF G_{Dn}

Network structure	layer	Kernel size	Output	Next layer
Input	raw image	/	$640 \times 640 \times 3$	Conv1
Encode 1	Conv1	$3 \times 3, S = 1$	$640 \times 640 \times 32$	Conv2
	Conv2	$3 \times 3, S = 1$	$640 \times 640 \times 64$	Conv3
	Conv3	$3 \times 3, S = 1$	$640 \times 640 \times 128$	Conv4
	Conv4	$3 \times 3, S = 1$	$640 \times 640 \times 256$	Conv5, Conv11
DownSampling	Conv5	$1 \times 1, S = 2$	$320 \times 320 \times 256$	Conv6
Encode 2	Conv6	$3 \times 3, S = 1$	$320 \times 320 \times 32$	Conv7
	Conv7	$3 \times 3, S = 1$	$320 \times 320 \times 64$	Conv8
	Conv8	$3 \times 3, S = 1$	$320 \times 320 \times 128$	Conv9
	Conv9	$3 \times 3, S = 1$	$320 \times 320 \times 256$	Conv10
UpSampling	conv10	$1 \times 1, S = 2$	$640 \times 640 \times 256$	Conv11
Decode	Conv11	$3 \times 3, S = 1$	$640 \times 640 \times 128$	Conv12
	Conv12	$3 \times 3, S = 1$	$640 \times 640 \times 64$	Conv13
	Conv13	$3 \times 3, S = 1$	$640 \times 640 \times 32$	Conv14
	Conv14	$3 \times 3, S = 1$	$640 \times 640 \times 1$	Output

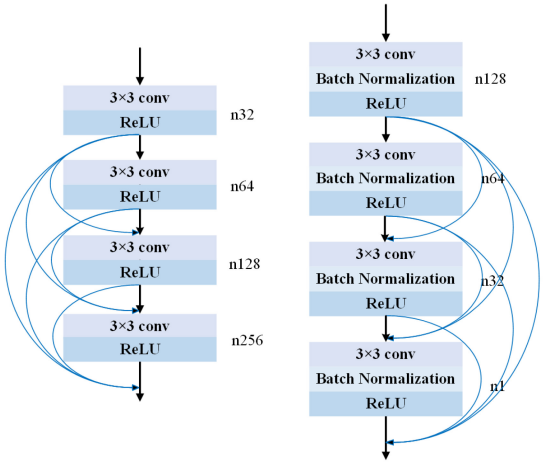


Fig. 3. Encoder and decoder of the generator G_{Dn} in DnGAN.

while ensuring speed of training. Moreover, DenseNet [28] is directly connected between the convolutional layers to reuse the original features. This can reduce the loss of features and avoid the disappearance of gradients. The decoder is also composed of four convolutional layers

$$Y_{i,j} = \sum_{u,v} H_{i-u,j-v} X_{u,v} + b. \tag{6}$$

To avoid gradient explosion or disappearance, batch normalization (BN) is applied to each convolutional layer. BN increases the robustness of the system by normalizing the input data to achieve constraints on the search space of the system parameters. The input batch data $X = \{x_1, \ldots, x_i, \ldots, x_K\}$ are taken to be the mean value μ. Find the variance σ^2 of the batch data from the resulting μ. The representation of the network is changed due to the transformation of the normalized x_i from the original data distribution to the normal distribution. To achieve better scale transformation and bias, BN introduces two new parameters that can be obtained by learning through model training: 1) the translation factor ξ and 2) the scale factor η

$$y_i = \eta \frac{x_i - \mu}{\sqrt{\sigma^2 + \varepsilon}} + \xi. \tag{7}$$

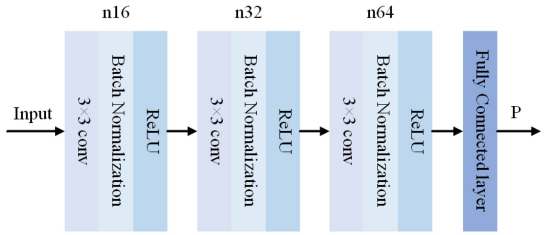


Fig. 4. Discriminator D_{Dn} model of DnGAN.

The decoder also employs a ReLU activation function in each neural network to speed up convergence

$$\mathrm{ReLU}(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0. \end{cases} \tag{8}$$

As shown in the above equation, the function outputs zero for negative input values, effectively inhibiting the activation of the corresponding neuron. This selective activation of neurons contributes to a simplified network architecture, leading to substantial computational efficiency. The convolutional kernel stride for all encoders and decoders is set to 1. As shown in Fig. 4, the discriminator D_{Dn} consists of three convolutional layers and a fully connected layer. Each convolutional layer is set with 3 × 3 convolutional kernels, BN, and ReLU.

B. Loss Functions for DnGAN

1) Loss Function of G_{Dn}: The loss function L_{Gd} of G_{Dn} consists of the reconstruction loss and the perceptual loss. The reconstruction loss L_{Gdmse} is defined as the mean-squared error (MSE) between the denoised image generated by DnGAN and the noiseless image

$$L_{Gd} = L_{\mathrm{Gdmse}} + L_{\mathrm{Gdper}} \tag{9}$$

$$L_{\mathrm{Gdmse}}(x, \tilde{x}) = \frac{1}{hw} \sum_{i=1}^{h} \sum_{j=1}^{w} \left[G_{Dn}(x)_{i,j} - \tilde{x}_{i,j} \right]^2 \tag{10}$$

where x represents the input noisy image, and $\tilde{x}$ represents the noiseless image. $G_{Dn}(x)$ represents the denoised image generated by G_{Dn}, where i and j denote the rows and columns of features, respectively. h and w represent the height and width of the image.

The clarity of the denoised image is improved by the perceptual loss. The infrared images I_{NIR}, I_{MIR}, and I_{LIR} in three bands are convolved with a single channel 1 × 1 to obtain the source image features F. Similarly, the denoised result I_1 is obtained after convolution of the denoised images I_{denosied}. The final loss L_{Gdper} is calculated using the $L2$ paradigm

$$F = \mathrm{conv}[\mathrm{concat}(I_{\mathrm{NIR}}, I_{\mathrm{MIR}}, I_{\mathrm{LIR}})] \tag{11}$$

$$I_1 = \mathrm{conv}(I_{\mathrm{denosied}}) \tag{12}$$

$$L_{\mathrm{Gdper}}(F, I_1) = \frac{1}{H_p W_p} \left\| \varphi_p(F) - \varphi_p(I_1) \right\|_2^2 \tag{13}$$

where p denotes the pth layer of the network. H_p and W_p represent the height and width of the input features, respectively. $\varphi_p(F)$ and $\varphi_p(I_1)$ indicate the output features obtained through the pth layer of the network. $\| \cdot \|_2^2$ denotes the $L2$ paradigm.

TABLE II
NETWORK STRUCTURE PARAMETERS OF G_{Fn}

Network structure	layer	Kernel size	Output	Next layer
Input	Denoising image		640 × 640 × 1	Conv1
Encode	Conv1	3 × 3, S = 1	640 × 640 × 32	CBAM1
	CBAM1	/	640 × 640 × 32	Conv2
	Conv2	3 × 3, S = 1	640 × 640 × 64	Conv3
	Conv3	3 × 3, S = 1	640 × 640 × 128	CBAM2
	CBAM2	/	640 × 640 × 128	Conv4
	Conv4	3 × 3, S = 1	640 × 640 × 256	Conv5
Decode	Conv5	3 × 3, S = 1	640 × 640 × 128	Conv6
	Conv6	3 × 3, S = 1	640 × 640 × 64	Conv7
	Conv7	3 × 3, S = 1	640 × 640 × 32	Conv8
	Conv8	3 × 3, S = 1	640 × 640 × 16	Conv9
	Conv9	3 × 3, S = 1	640 × 640 × 1	Output

2) Loss Function of D_{Dn}: The loss function L_{Dd} of D_{Dn} comprises decision losses of the denoised image and noiseless image, denoted by L_{Ddx} and $L_{Dd\tilde{x}}$, respectively

$$L_{Ddx} = \frac{1}{2N} \sum_{i=1}^{N} \left\{ \left[P_x\left(G_{Dn}(x^n)\right) - a_1 \right]^2 + \left[P_{\tilde{x}}\left(G_{Dn}(x^n)\right) - a_2 \right]^2 \right\} \tag{14}$$

$$L_{Dd\tilde{x}} = \frac{1}{2N} \sum_{i=1}^{N} \left\{ \left[P_x(\tilde{x}^n) - a_3 \right]^2 + \left[P_{\tilde{x}}(\tilde{x}^n) - a_4 \right]^2 \right\} \tag{15}$$

where N is the number of input images. P_x represents the probability that D_{Dn} judges the input as a denoised image, and $P_{\tilde{x}}$ represents the probability of being judged as a noiseless image. $G_{Dn}(x^n)$ denotes the nth denoised image generated by G_{Dn}. a_1 and a_2 are probability labels. When the input is a denoised image, $P_x \to 1$ and $P_{\tilde{x}} \to 0$ are expected. Thus, a_1 is set to 1, and a_2 is set to 0. Similarly, a_3 is set to 0, and a_4 is set to 1.

C. Network Structure of FuGAN

Stage 1-2: The denoised multiband image generated by DnGAN is fed into FuGAN through three channels to obtain a high quality multiband fused image. FuGAN achieves high-quality fusion images through the adversarial game between the generator G_{Fu} and the discriminator D_{Fu}. The generator G_{Fu} consists of an encoder and a decoder. The network structure parameters specific to the generator G_{Fn} are shown in Table II.

As shown in Fig. 5, the encoder of G_{Fu} consists of four convolutional layers and two convolutional block attention modules (CBAMs). The flow of information between networks is enhanced by the establishment of jump links. Like decoder of G_{Du}, each convolutional layer is set with 3 × 3 convolutional kernels, BN, and ReLU.

The CBAM is inserted after the first and third convolutional layers to establish skip connections. CBAM enables the network model to focus on information that is more critical to the task at hand and reduces the focus on other information, thereby increasing the efficiency of the entire network. CBAM consists of two parts: 1) channel attention module (CAM) and 2) spatial attention module (SAM). The features output from the convolutional layer are first compressed by CAM in spatial dimension to obtain a 1-D channel attention map. The features output from the convolutional layer are multiplied with the channel attention map and input to the SAM. This

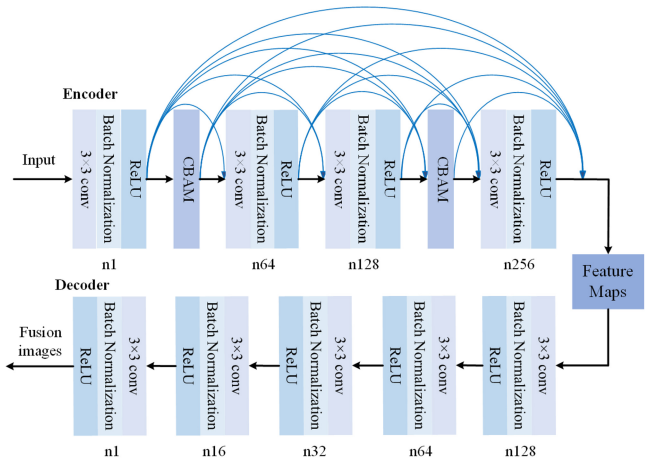


Fig. 5. Generator G_{Fu} model of FuGAN.

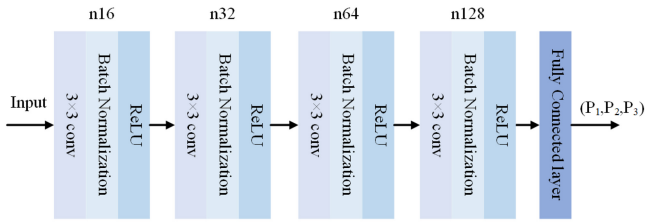


Fig. 6. Discriminator D_{Fu} model of FuGAN.

part compresses the channel to obtain a 2-D spatial attention map, which is multiplied with the input data to obtain the final weighted result

$$f' = M_c(f) \otimes f \tag{16}$$

$$f'' = M_s(f') \otimes f' \tag{17}$$

where f represents the features output from the convolutional layer, f' represents the features output from the CAM layer, and f'' represents the features output from the SAM layer. M_c represents the 1-D channel attention map, and M_s represents the 2-D spatial attention map.

As shown in Fig. 6, the discriminator D_{Fu} is designed as a multiclassifier. It consists of four convolutional layers and one fully connected layer. Based on the image features extracted by the convolutional layers, the fully connected layer discriminates the input to obtain a probability vector for image fusion. Each layer is composed of a 3 × 3 convolutional kernel, ReLU, and BN. The stride of all convolutional layers is set to 1, and the padding method is SAME.

D. Loss Functions for FuGAN

1) Loss Function of G_{Fu}: The loss function L_{Gf} of G_{Fu} consists of adversarial loss L_{Gfadv}, perceptual loss L_{Gfper}, and structural similarity index (SSIM) loss L_{Gfssim}

$$L_{Gf} = L_{\text{Gfadv}} + L_{\text{Gfper}} + L_{\text{Gfssim}} \tag{18}$$

$$L_{\text{Gfadv}} = \frac{1}{3N}\sum_{n=1}^{N}\Big\{\big[D_{Fu}\big(I^n_{\text{fused}}\big)[0]-e\big]^2 + \big[D_{Fu}\big(I^n_{\text{fused}}\big)[1]-e\big]^2 + \big[D_{Fu}\big(I^n_{\text{fused}}\big)[2]-e\big]^2\Big\} \tag{19}$$

where e is the probability that the fused image matches the source image. The goal of G_{Fu} is to make D_{Fu} indistinguishable between the fused image and the source image, hence e is set to 1. I^n_{fused} indicates the nth fused image input to D_{Fu}. As the discriminator is a multiclassifier, the output is a 1 × 3 probability vector. The three terms of this vector represent the probabilities that the fusion image is NIR, MIR, and LIR image, expressed by $D_{Fu}(\cdot)[0]$, $D_{Fu}(\cdot)[1]$, and $D_{Fu}(\cdot)[2]$, respectively.

The perceptual loss L_{Gfper} can motivate F_{Fu} to generate fused images with high information entropy, which is defined in the same way as L_{Gdper} in DnGAN. The SSIM loss L_{Gfssim} can constrain the correlation, luminance distortion, and contrast distortion of the fusion image

$$I_2 = \text{conv}(I_{\text{fused}}) \tag{20}$$

$$L_{\text{Gfper}}(F, I_2) = \frac{1}{H_p W_p}\left\|\varphi_p(F) - \varphi_p(I_2)\right\|_2^2 \tag{21}$$

$$\text{SSIM} = \frac{1}{3}\text{SSIM}(I_{\text{fused}}, I_{\text{NIR}}) + \frac{1}{3}\text{SSIM}(I_{\text{fused}}, I_{\text{MIR}}) + \frac{1}{3}\text{SSIM}(I_{\text{fused}}, I_{\text{LIR}}) \tag{22}$$

$$L_{\text{Gfssim}} = 1 - \text{SSIM}. \tag{23}$$

2) Loss Function of D_{Fu}: D_{Fu} is a multiclassifier which adopts the least squares loss function L_{Df}. And L_{Df} comprises four decision losses for NIR, MIR, LIR images, and fusion images, denoted by L_{DfNIR}, L_{DfMIR}, L_{DfLIR}, and L_{DfFused}, respectively

$$L_{Df} = L_{\text{DfNIR}} + L_{\text{DfMIR}} + L_{\text{DfLIR}} + L_{\text{DfFused}}. \tag{24}$$

Considering the output of the discriminator is a 1 × 3 vector $D_{Fu}(\cdot)$, so that $P_{\text{NIR}} = D_{Fu}(\cdot)[0]$, $P_{\text{MIR}} = D_{Fu}(\cdot)[1]$, $P_{\text{LIR}} = D_{Fu}(\cdot)[2]$. The corresponding L_{DfNIR} loss, L_{DfMIR} loss, and L_{DfLIR} loss are defined as

$$L_{\text{DfNIR}} = \frac{1}{3N}\sum_{n=1}^{N}\Big\{\big[D_{Fu}\big(I^n_{\text{fused}}\big)[0]-b_1\big]^2 + \big[D_{Fu}\big(I^n_{\text{fused}}\big)[1]-b_2\big]^2 + \big[D_{Fu}\big(I^n_{\text{fused}}\big)[2]-b_3\big]^2\Big\} \tag{25}$$

$$L_{\text{DfMIR}} = \frac{1}{3N}\sum_{n=1}^{N}\Big\{\big[D_{Fu}\big(I^n_{\text{fused}}\big)[0]-b_4\big]^2 + \big[D_{Fu}\big(I^n_{\text{fused}}\big)[1]-b_5\big]^2 + \big[D_{Fu}\big(I^n_{\text{fused}}\big)[2]-b_6\big]^2\Big\} \tag{26}$$

$$L_{\text{DfLIR}} = \frac{1}{3N}\sum_{n=1}^{N}\Big\{\big[D_{Fu}\big(I^n_{\text{fused}}\big)[0]-b_7\big]^2 + \big[D_{Fu}\big(I^n_{\text{fused}}\big)[1]-b_8\big]^2 + \big[D_{Fu}\big(I^n_{\text{fused}}\big)[2]-b_9\big]^2\Big\} \tag{27}$$

$$L_{Df\text{Fused}} = \frac{1}{3N}\sum_{n=1}^{N}\Big\{\big[D_{Fu}\big(I^n_{\text{fused}}\big)[0]-b_{10}\big]^2 + \big[D_{Fu}\big(I^n_{\text{fused}}\big)[1]-b_{11}\big]^2 + \big[D_{Fu}\big(I^n_{\text{fused}}\big)[2]-b_{12}\big]^2\Big\} \tag{28}$$

where I^n_{fused} represents the nth NIR image. b_1, b_2, and b_3 are probability labels. The generative network is expected to output images that are independently and identically distributed across the training samples. When the input is an NIR image, it is expected that $P_{\text{NIR}} \to 1$, and P_{MIR}, $P_{\text{LIR}} \to 0$, so, b_1

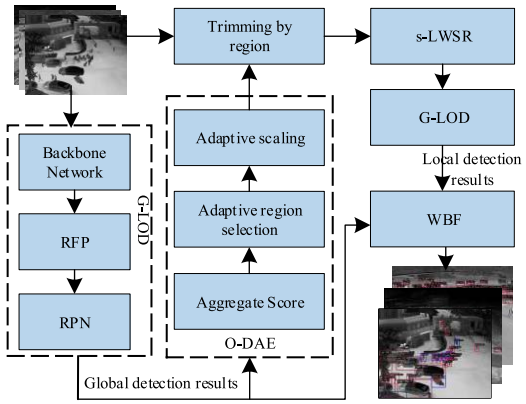


Fig. 7. Multiscale infrared dense small object detection architecture.

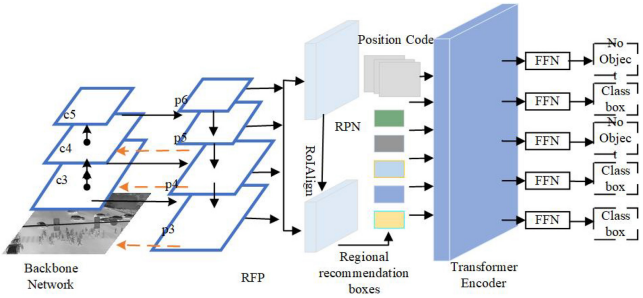


Fig. 8. Global–local object detection network.

is set to 1, and b_2 and b_3 are set to 0. Similarly, when I^n_{fused} represents the nth MIR image, b_5 is set to 1, and b_4 and b_6 are set to 0. If I^n_{fused} represents the nth LIR image. b_9 is set to 1, and b_7 and b_8 are set to 0. And in (28), b_{11}, b_{12}, and b_{13} are all set to 0. In other words, from the viewpoint of the discriminator, the fused images are the same degree of pseudo-NIR image, pseudo-MIR image, and pseudo-LIR image.

V. Multiscale Infrared Dense Small Object Detection Method Based on Transformer

As shown in Fig. 7, the fused image is fed into the G-LOD network to extract global information. The O-DAE algorithm is defined to specify the boundary of the object detection region. The input fused image is cropped according to the boundary. s-LWSR improves image resolution. The output is then analyzed locally by G-LOD for detection. The WBF combines global and local predictions to achieve high-precision detection.

A. G-LOD

Stage 2-1: The fused image is fed into the G-LOD network to get global detection results $T(J)$ (category, bounding box).

The G-LOD network consists of CSPResNet101, recursive feature pyramid (RFP), Transformer encoder, and feedforward networks (FFNs), as shown in Fig. 8. CSPResNet101 was chosen as the backbone network of G-LOD [29]. The RFP was added to the Neck to ensure reasonable full utilization of multiscale features. The prediction frame information is extracted from the multilayer feature map by the RoIAlign,

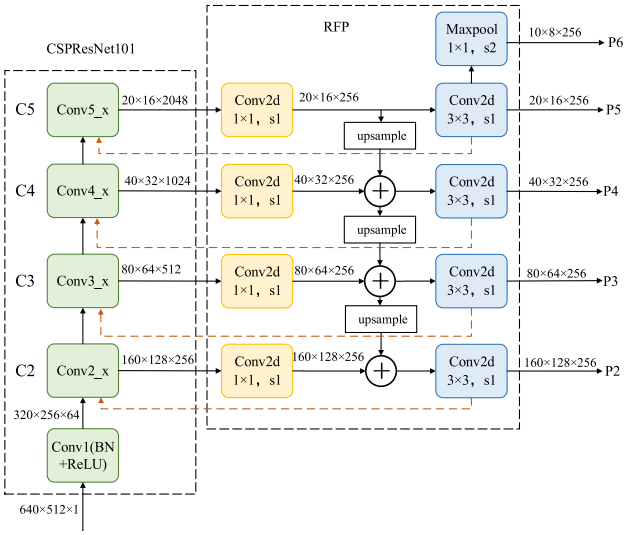


Fig. 9. Structure of CSPResNet101 and RFP.

which is then combined with positional data relating to candidate regions before being fed into the detection head for training. As Transformer is capable of end-to-end detection, it was used to extract the input detection header of the candidate region. To ensure the detection precision, a Bipartite Graph Matchings was used to force constraints on the detection range, effectively avoiding operations, such as nonmaximum suppression (NMS) within the detection model.

To improve the detection accuracy of dense small objects, we introduce the classical backbone network ResNet101, which relies on the residual learning mechanism to ensure the training performance of the deeper network by copying the features of the shallow network to the deeper network for effective feature extraction. To achieve richer gradient combinations, the cross stage partial network (CSPNet) is used to combine with ResNet101 to form CSPResNet101.

The structure of CSPResNet101 and RFP is shown in Fig. 9. The backbone network CSPResNet101 is used to extract features from the input image and map the features to the RFP to produce a multilevel feature map. By setting additional feedback connections in the RFP, the semantically information-rich high-level features are brought back to the lower level feature layers of the backbone network to enhance the feature extraction performance of the backbone network and achieve accelerated training of the network.

Region proposal network (RPN) uses the global feature map to determine whether a object is present in a candidate region. Binary class labels are assigned to each candidate region by generating anchor frame coverage images at different scales. A candidate region is assigned a positive label if its Intersection over Union (IoU) overlap with the real frame is above a certain threshold. This label of 1 means that the candidate region is a object region; 0 means that it is not a object region.

The position of the candidate region is defined as [PE(X):PE(Y):PE(w):PE(h)], where [:] denotes the

connection, $(X, Y) \in [0, 1]^2$ is the coordinates of the upper left corner of the prediction box, and $(w, h) \in [0, 1]^2$ is the width and height of the prediction box

$$\mathrm{PE}(X)_{2i} = \sin\left(X/10000^{2i}\right) \tag{29}$$

$$\mathrm{PE}(X)_{2i+1} = \cos\left(X/10000^{2i+1}\right). \tag{30}$$

After the feature information of the candidate regions is input to the Transformer encoder, it is aggregated by the self-attention mechanism and finally reaches the shared FFN. The FFN consists of a 3-layer perceptron with ReLU activation and a linear layer that predicts the category labels of each candidate region (including the "no object") and bounding box. The bounding box of global detection result is denoted as $B^l = \{c_k^l, b_k^l\}$, where l denotes the original image, k is the prediction frame code, c is the object category, and b is the prediction frame location information.

This section uses the Hungarian loss for training supervision of the detection head. $\bar{y} = \{\bar{y}_i\}_{u=1}^{\mathcal{M}}$ denotes the set of true objects and $\hat{y} = \{\hat{y}_i\}_{u=1}^{\mathcal{N}}$ denotes the set of predictions

$$\mathcal{L}_{\text{Hungarian}}(\bar{y}, \hat{y}) = \sum_{i=1}^{\mathcal{N}} \left[\mathcal{L}_{\text{class}}^{i,\hat{\delta}(i)} + \mathbb{1}_{\{\hat{y}\neq\emptyset\}} \mathcal{L}_{\text{box}}^{i,\hat{\delta}(i)}\right]. \tag{31}$$

In general $\mathcal{M} < \mathcal{N}$ because there are cases where the prediction frame corresponds to no object. $\mathcal{L}_{\text{class}}^{i,\hat{\delta}(i)}$ and $\mathcal{L}_{\text{box}}^{i,\hat{\delta}(i)}$ are the classification loss and bounding box regression loss, respectively, and $\hat{\delta}$ is the matching loss between $\bar{y}$ and $\hat{y}$, denoted as follows:

$$\hat{\delta} = \underset{\delta\in\mathfrak{S}\mathcal{N}}{\operatorname{argmin}} \sum_{i=1}^{\mathcal{N}} \mathcal{L}_{\text{match}}(\bar{y}_i, \hat{y}_{\delta(i)}) \tag{32}$$

where $\delta \in \mathfrak{S}\mathcal{N}$ is the set of $\mathcal{N}$ prediction frames and $\mathcal{L}_{\text{match}}$ is the pairwise matching loss.

B. O-DAE

Stage 2-2: The global detection results of G-LOD network are input to O-DAE module, and the clustering score is used to obtain the region with denser small objects, and then the region is adaptively adjusted to determine the final cropping region.

The O-DAE module can adaptively crop out dense object regions in the image for G-LOD to achieve fine detection. The pseudocode for the O-DAE is shown in Algorithm 1.

The aggregate score model is used to measure the denseness of the area where the bounding box $B^l = \{c_k^l, b_k^l\}$ obtained in the global detection is located

$$G(X, Y) = \begin{cases} \sum_k 1, & \text{if } (X, Y) \ \ in\ b_k^l \\ 0, & \text{otherwise} \end{cases} \tag{33}$$

$$E_q = \{(X, Y) \ \mid \ \text{for } \Phi(G(X, Y)) > \Theta\} \tag{34}$$

where (X, Y) represents the coordinates of the adaptive region. Θ is the score threshold. The aggregate score is used as a screening condition for dense regions, and screening high-scoring coordinates as an input for adaptive region selection not only ensures credible zoomed-in regions, but also speeds up the selection. The set of high-resolution coordinates E_q and the set of coordinates of the boundary of the prediction frame Z are fed into the adaptive region selection based on density-based spatial clustering of applications with noise (DBSCAN).

Algorithm 1 O-DAE Algorithm

Input: The detection result $T(J)$ (category, bounding box $B^l = \{c_k^l, b_k^l\}$);
Output: The final cropped local image set J';

```
Calculate the density of the area where the bounding box B^l = {c_k^l, b_k^l} is located by Eq. (33);
if G(X, Y) > Θ then
    Get the coordinate set E_q;
end if
The source image J is divided into subregion images set {J_1, J_2, ..., J_λ, ..., J_υ} according to the E_q;
for λ = 1:υ do
    Obtain the boundaries of the subregion J_λ;
    Calculate the center coordinates (X_0, Y_0) of the subregion J_λ by Eq. (35);
    Calculate the scale standard s and width to height ratio r by Eq. (36)-(37);
    Crop the subregion images J'_λ according to Eq. (40)-(41);
end for
return The final cropped local image set J' = {J'_1, J'_2, ..., J'_λ, ..., J'_υ};
```

Each coordinate in E_q is assigned to a specific class, the boundaries of the subregion can be easily obtained, and the intercepted subregion contains the global box of all targets, which can avoid object truncation. As a result, each subregion has a different size. The source image J is divided into subregion images set $\{J_1, J_2, \ldots, J_\lambda, \ldots, J_\upsilon\}$.

Due to the different scales of the subregions, they cannot be fed directly to the local detector. In order to maintain the scale and proportion of the subregions within a preset range, the subregion adaptive scaling method is proposed. The bounding box of the subregion images J_λ is (X_1, X_2, Y_1, Y_2). (X_1, Y_1) are the coordinates of the upper left corner of the bounding box, (X_2, Y_2) stands for the coordinates of the lower right corner of the bounding box, the center coordinates are (X_0, Y_0), S indicates the standard size, and r denotes the width to height ratio

$$(X_0, Y_0) = \left(\frac{X_1 + X_2}{2}, \frac{Y_1 + Y_2}{2}\right) \tag{35}$$

$$S = \sqrt{(X_2 - X_1)(Y_2 - Y_1)} \tag{36}$$

$$r = \frac{Y_2 - Y_1}{X_2 - X_1}. \tag{37}$$

When $S \geq S'$, $r \in [r_{\min}, r_{\max}]$

$$h^\lambda = \max\left(S', \frac{X_2 - X_1}{2}, Y_2 - Y_1\right) \tag{38}$$

$$w^\lambda = \max\left(S', \frac{Y_2 - Y_1}{2}, X_2 - X_1\right) \tag{39}$$

$$\left(X_1^\lambda, Y_1^\lambda\right) = \left(X_0 - \frac{w^\lambda}{2}, Y_0 - \frac{h^\lambda}{2}\right) \tag{40}$$

$$\left(X_2^\lambda, Y_2^\lambda\right) = \left(X_0 + \frac{w^\lambda}{2}, Y_0 + \frac{h^\lambda}{2}\right) \tag{41}$$

where $(X_1^\lambda, Y_1^\lambda)$ and $(X_2^\lambda, Y_2^\lambda)$ denote the new coordinates of the upper left corner and the lower right corner of the bounding box of the final cropped subregion image J'_λ, respectively. h^λ and w^λ denote the final cropped height and width of the bounding box. Final cropped local image set $J' = \{J'_1, J'_2, \ldots, J'_\lambda, \ldots, J'_\upsilon\}$ is output.

We did a sampling statistic on all the datasets and its distribution presents are characterized by nonuniform spatial distributions and a high proportion of small objects, with occlusion and overlap between dense objects. Due to the uneven spatial distribution of the dataset, very empty areas do not have much identification value. Therefore, we set the score threshold at 2 ($\Theta = 2$) to ensure that there is an identification object in the region. Of course, there are very limited instances of complete overlapping or obscuration in the dataset. Although the proposed algorithm can distinguish the object to some extent, it is still unable to do anything for very dense cases. So we have to discard the very dense parts, detection area should not be too small. Therefore, we set the standard size at 5 ($S' = 5$) based on statistics and empirical values. To ensure that the image input to the neural network is square or tends to be square, the aspect ratio of the cropped image should not be too large or too small, so it is set between [0.5, 2] ($r_{\min} = 0.5, r_{\max} = 2$).

C. s-LWSR

Stage 2-3: The cropped local images J' are fed into the s-LWSR network to increase the resolution of the image for more detailed detection information and higher detection accuracy.

After the image has been scaled, the size has reached the acceptance standard of the detector. However, due to a series of processing operations, the cropped image inevitably suffers from blurring of image quality, loss of image details, and degradation of resolution. To address the above problems, we employ the highly efficient s-LWSR network [30], which is based on the a priori knowledge of the image, to recover the lost object edge information of the cropped image so as to obtain a high-resolution image and provide more semantic information to the object detector. Since the adaptive selection algorithm intercepts subregions of different sizes, super-resolution processing for large subregions is obviously unnecessary. So

$$J^s = \begin{cases} \text{super}\left(J'_\lambda\right), & \text{if } s^\lambda \le S_{sr} \\ J'_\lambda, & \text{otherwise} \end{cases} \tag{42}$$

where J^s is a recovered high-resolution image. The cropped image of J' is input to s-LWSR. The local images are processed with super-resolution and fed into the G-LOD local fine detection network.

Stage 2-4: The recovered high-resolution images J^s are fed into the G-LOD network for secondary fine detection of the object and local detection results $T(J^s)$ are obtained.

Algorithm 2 MsDSOD Algorithm

Input: The fused image with multi-band infrared features generated by MIF-CGAN;
Output: the final detection result;

```
for l = 1:L do
    Get the global detection result T(J) and the bounding box B^l = {c_k^l, b_k^l} by G-LOD network;
    Calculate the final cropped local image set J' = {J'_1, J'_2, ..., J'_λ, ..., J'_υ} by Algorithm 1;
    for λ = 1:υ do
        if s^λ ≤ S_sr then
            Enhanced image resolution of J'_λ through s-LWSR network;
        end if
        Get the local detection result T(J^s) by G-LOD network;
    end for
    Calculate the final detection result B^f by Eq.(43);
end for
return The final detection result B^f;
```

D. WBF

Stage 2-5: Both local and global detection results are fed into the WBF module for result fusion to output the final object detection results

$$B^f = \text{merge}\left(T(J), T\left(J^s\right)\right) \tag{43}$$

where B^f indicates the final detection result, J denotes the source image, and J^s denotes the local super-resolution image generated by the O-DAE network. $T(\cdot)$ indicates network detection result, and merge$(\cdot)$ represents WBF [31]. The pseudocode for the MsDSOD is shown in Algorithm 2.

VI. Simulation Experiments and Analysis of Results

A. Environment Configuration

The experimental environment is Intel Xeon Gold 5218R CPU, 32G DDR4*8 RAM, NVIDIA GeForce RTX3090 GPU. The operating system is Window 10, the programming language is Python, and the deep learning framework is TensorFlow-gpu 1.14.0. Training is conducted for 300 epochs using the Adam optimizer with 0.0001 learning rate and 2 Batchsize and the average is taken.

1) Multiband Infrared Image Dataset: The Multispectral dataset [32] is used as the training and testing data for the multiband image fusion simulation experiments, as shown in Fig. 10. The dataset includes 7512 sets of images in different scenes. Each set of images is divided into four categories: 1) RGB; 2) NIR; 3) MIR; and 4) LIR. It contains a variety of scenes in university environments, such as cars, bicycles, and pedestrians in road scenes, and buildings in natural scenes. The training set is divided into 3740 sets for day and 3772 sets for night. In this article, the selected image experienced scene alignment and scaling processing.

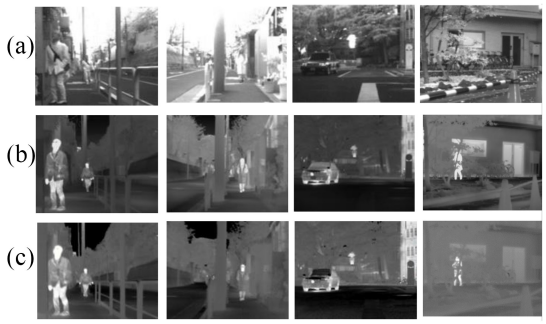


Fig. 10. Multispectral dataset. (a) NIR. (b) MIR. (c) LIR.

Fig. 11. Dataset of MsDSOD.

2) Infrared Small Object Detection Dataset: The MsDSOD simulation experiments utilized the infrared small object detection dataset in aerial photography, as shown in Fig. 11. The dataset collects 11 045 infrared image data under the overhead angle of surveillance, which contains rich infrared small objects: pedestrians, cars, buses, bicycles, cyclists, and trucks. The label files are converted to json format to construct the MS COCO format dataset. We selected 8837 images as training set, 1104 images as validation set, and 1104 images as test set.

B. Evaluation Indicators

To evaluate the performance of the MIF-CGAN algorithm, the fusion algorithm will be quantitatively evaluated using the objective evaluation criteria, which mainly include Entropy (EN), AG, standard deviation (SD), SSIM, peak signal-to-noise ratio (PSNR), and gradient-based fusion performance $Q^{AB/F}$. The metrics for measuring object detection algorithms contain precision (P), recall (R), mAP, and F1 score ($F1$).

EN is the amount of information carried by each image feature in the image grayscale distribution

$$\mathrm{EN} = \sum_{t=0}^{255} p_t \log_2 p_t \tag{44}$$

where p_t denotes the proportion of pixels with gray value t in the image. AG is the clarity of the fused image

$$\mathrm{AG} = \frac{1}{(W-1)(H-1)} \sum_{w=1}^{W-1} \sum_{h=1}^{H-1} \sqrt{\frac{[R(w+1,h)-R(w,h)]^2 + [R(w,h+1)+R(w,h)]^2}{2}} \tag{45}$$

where W and H denote the width and height of the image, respectively, and $R(w, h)$ denotes the pixel value located at $(w,\ h)$.

SD is the degree of dispersion of the gray value of an image pixel with respect to the mean value

$$u = \frac{1}{WH} \sum_{w=1}^{W} \sum_{h=1}^{H} R(w,h) \tag{46}$$

$$\mathrm{SD} = \sqrt{\frac{1}{WH} \sum_{w=1}^{W} \sum_{h=1}^{H} (R(w,h)-u)^2} \tag{47}$$

where u denotes the mean value.

SSIM is the similarity of two images in terms of brightness, contrast, and structure

$$\mathrm{SSIM} = [l(X,R)]^{\alpha}[c(X,R)]^{\beta}[s(X,R)]^{\gamma} \tag{48}$$

where X represents the source image, R represents the fused image, $l(X, R)$, $c(X, R)$, and $s(X, R)$ are the formulas for luminance similarity, contrast similarity, and structural similarity, respectively, and α, β, and γ are the weighting coefficients, which are generally taken to be 1.

PSNR is an objective measure of the difference in noise level between two images

$$\mathrm{MSE} = \frac{1}{WH} \sum_{w=1}^{W-1} \sum_{h=0}^{H-1} [X(w,h)-R(w,h)]^2 \tag{49}$$

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{\mathrm{MAX}_I}{\mathrm{MSE}}\right)^2 = 20\log_{10}\left(\frac{\mathrm{MAX}_I}{\sqrt{\mathrm{MSE}}}\right) \tag{50}$$

where $X(w, h)$ and $R(w, h)$ denote the pixel values of the two images at $(w,\ h)$, MSE denotes the mean-square error of the two images, and MAX_I denotes the maximum value of the image pixels that can be taken.

$Q^{AB/F}$ is the extent to which the salient information of the input is represented in the fused image [33].

C. Results and Analysis of Ablation Experiments

1) MIF-CGAN: In this ablation experiment, we only removed the attention modules from the second and fifth layers of the FuGAN generator. The experiment verifies the effect of the CAM and the SAM. Then, we only removed DnGAN to verify the effectiveness of the cascade structure for image fusion quality improvement. NIR, MIR, and LIR are fed into FuGAN after connecting them along the channels. The experimental results are shown in Table V. MIF-CGAN has significant improvement in EN, AG, SD, and $Q^{AB/F}$ compared to both Ablation1 and Ablation2. This indicates that attention modules in the MIF-CGAN can capture more image features and enhance the resolution of the fused image and DnGAN effectively improves the clarity and information richness of the fused images.

2) MsDSOD: We design five sets of ablation experiments to verify the effectiveness of each module of MsDSOD, and the results are shown in Table III. The network of Group V is the MsDSOD. Compared to Group I, the

TABLE III
RESULTS OF MSDSOD ABLATION EXPERIMENTS

Groups	CSPResNet101	RFP	Transformer Encoder	O-DAE	mAP	AP_{50}	AP_{75}	AP_s
I	×	×	×	×	49.7	86.8	51.1	38.8
II	√	×	×	×	52.9	87.4	59.8	43.1
III	√	√	×	×	59.2	89.3	65.9	51.5
IV	√	√	√	×	60.8	90.4	67.4	52.9
V	√	√	√	√	**63.1**	**93.6**	**70.0**	**55.7**

TABLE IV
COMPARISON EXPERIMENT OF MIF-CGAN

Methods \ Indicators	EN	AG	SD	$SSIM$	$PSNR$	$Q^{AB/F}$	Parameters (MB)	FLOPs (G)	Time (s)
DenseFuse [34]	7.2755	0.0224	69.4440	0.2481	58.4254	0.0805	**0.075**	49.63	0.236
FusionGAN [9]	6.4860	0.0114	34.4700	**0.5030**	**64.7015**	0.1495	0.972	499.37	0.196
DDcGAN [35]	7.5903	0.0247	65.3309	0.3359	59.8904	0.2610	1.121	899.04	0.203
GANMcC [36]	6.7973	0.0133	42.6550	0.4666	61.5798	0.2077	1.901	1010.09	0.200
RFN-Nest [37]	7.1645	0.0133	51.8210	0.4567	59.6889	0.2428	12.033	/	0.216
CSF [38]	7.1734	0.0166	56.2399	0.4488	61.0692	0.2647	0.178	/	3.113
GAN-FM [39]	7.4091	0.0251	68.9384	0.3778	5.97467	**0.3405**	/	/	/
TarDAL [40]	6.9844	0.0183	64.9494	0.4525	60.1620	0.2788	0.317	**15.67**	**0.039**
MIF-CGAN	**7.5982**	**0.0263**	**70.2344**	0.4008	62.2523	0.2990	2.272	63.88	0.259

TABLE V
RESULTS OF MIF-CGAN ABLATION EXPERIMENTS

Methods	EN	AG	SD	$SSIM$	$PSNR$	$Q^{AB/F}$
Ablation1	7.1863	0.0239	65.7710	0.4145	63.1017	0.1840
Ablation2	7.2413	0.0164	59.5917	**0.4541**	**63.5138**	0.1729
ours	**7.5982**	**0.0263**	**70.2344**	0.4008	62.2523	**0.2990**

CSPResNet101 of Group II improves the average detection precision by 6.44%. This shows that the deeper network structure of CSPResNet101 effectively extracts image features. Compared to Group II, the RFP of Group III improves the average precision by 11.91% and the small object detection precision by 19.49%. This is due to the fact that the proposed algorithm introduces an RFP structure RFP, which avoids the loss of small target information through multiscale feature fusion. Compared to Group III, the Transformer Encoder of Group IV effectively improves the average detection precision as well as the small object detection precision. This is due to the fact that the proposed algorithm discards the decoder structure of the Transformer and connects the shared feed-forward network directly after the encoder, which is used to determine the class and location of the target. Compared to Group IV, the average precision of MsDSOD is improved by 3.78%, the small object detection precision is improved by 5.29%, and the large object detection precision is improved by 7.27%. It can be seen that the O-DAE module effectively improves the detection precision.

D. Results and Analysis of Comparative Experiments

1) MIF-CGAN: This algorithm is compared to eight image fusion algorithms, such as DenseFuse, FusionGAN, and DDcGAN. We evaluate the fusion effect of each algorithm in terms of subjective qualitative analysis and objective quantitative analysis. In Table IV and Fig. 12, AG and SD of MIF-CGAN improve 4.78% and 1.13% over the suboptimal algorithms, respectively, and outperform most of the algorithms in EN, PSNR, and $Q^{AB/F}$. This indicates that MIF-CGAN is able to obtain more sufficient visual information from the source image.

The algorithm generates rich fused images with less noise, strong contrast, High clarity. The proposed algorithm achieves suboptimal results in PSNR and $Q^{AB/F}$, after FusionGAN and GAN-FM, indicating that the difference between the fused image and the source image is large. This indicates that Cascade-GAN is able to obtain sufficient visual information from the source image with good visual perception. The SSIM of MIF-CGAN is lower. Because the source images and the fused images are very different in terms of brightness, contrast, and structural information. It is these differences that make the structure of the fused image clearer and solve the problem of low contrast and poor visual effect in infrared images. From the above, it can be seen that the proposed algorithm is able to fully obtain the target and texture detail information from the source image, and produce a fused image with high signal-to-noise ratio and high information entropy, which will be helpful for the subsequent target detection tasks.

Although the proposed algorithm performs well in EN, AG, and SD, it performs poorly in Parameters, FLOPs, and Time. This is due to the fact that the proposed algorithm consists of two GANs, the denoising network and the fusion network. Macroscopically, the training speed is much slower than that of a GAN such as FusionGAN. Although the introduction of the BN module with residual network structure speeds up the training speed of the proposed network to some extent, it is still slower than that of a GAN. This approach improves the network extraction capability and also increases the computational resources.

The effect of the MIF-CGAN algorithm is shown in Fig. 13. Columns (a)–(c) are the source images of NIR, MIR, and LIR, Columns (d)–(k) are the fusion images generated by the

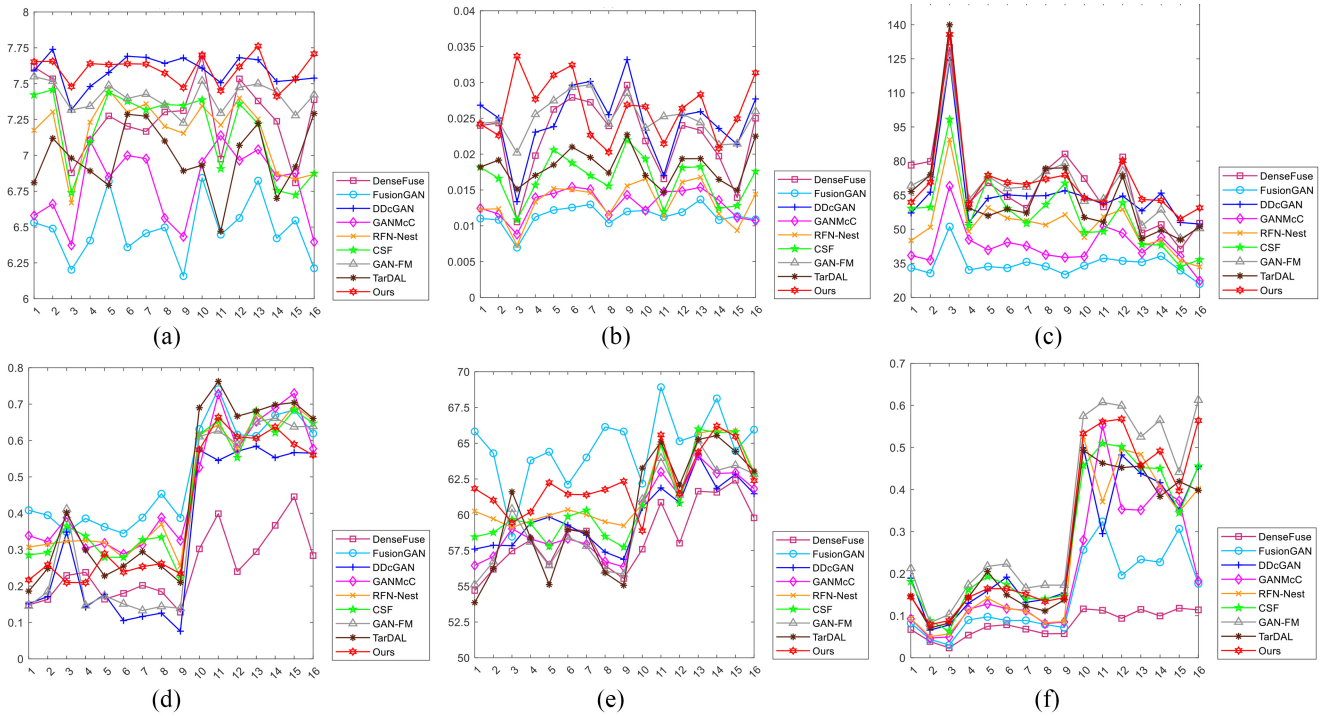


Fig. 12. Quantitative comparisons of the metrics. (a) EN. (b) AG. (c) SD. (d) SSIM. (e) PSNR. (f) $Q^{AB/F}$.

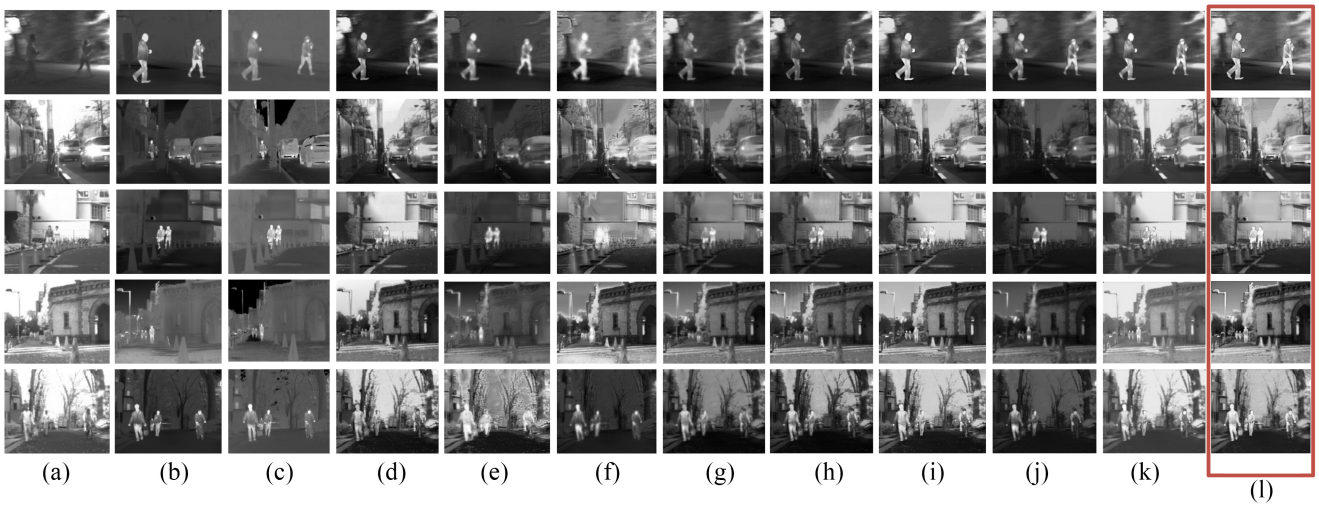


Fig. 13. Comparison of fusion effect. (a) NIR. (b) MIR. (c) LIR. (d) DenseFuse. (e) FusionGAN. (f) DDcGAN. (g) RFN-Nest. (h) CSF. (i) GAN-FM. (j) GANMcC. (k) TarDAL. (l) Ours.

comparison algorithm, and the last column is the fusion image generated by MIF-CGAN. From the overall visual perception, the fusion effects of DDcGAN, FusionGAN, RFN-Next, and GANMcC are all very blurred.Among them, the fused image of FusionGAN is more in favor of MIR and LIR, and the image object boundary is blurred. The background texture of bicycles, leaves, windows, etc. is basically lost, and the noise interference is serious. The DDcGAN fused image produces artifacts. the boundary of the pedestrian target is missing, and the background sharpening is serious, which will cause serious interference in the execution of the subsequent target detection task. RFN-Nest and GANMcC extract the basic texture information of NIR, but the contrast is low. The fused images of DenseFuse, CSF, and GAN-FM are clearer overall, but the boundary is blurred and the object is not bright enough in the pedestrian object area. Comparison reveals that the fused images generated by our algorithm have clearer object boundaries and texture details, more appropriate contrast, and more prominent object features, which are more useful for subsequent object detection.

2) MsDSOD: To verify the improvement of our MsDSOD algorithm for infrared dense small object detection precision, several YOLO variants and their improvements are used as comparison algorithms. The experimental results are shown in Tables VI and VII. Table VI lists the accuracy of various types of algorithms with different labels. In order to show

TABLE VI
AVERAGE PRECISION OF COMPARATIVE EXPERIMENTS

Methods \ Indicators	person	car	bus	cyclist	bike	truck	mAP	AP_{50}	AP_{75}	AP_s	AP_M	AP_L
DETR [20]	30.2	54.3	66.0	40.2	40.7	63.2	49.1	82.1	51.9	37.9	58.7	62.3
FCOS [23]	40.2	63.4	74.2	46.5	52.8	72.4	58.2	90.0	64.5	49.6	66.4	66.3
Cascade R-CNN [26]	40.7	65.1	75.7	47.6	55.9	74.5	59.9	89.9	66.5	51.5	67.3	71.0
Cascade Mask R-CNN [27]	37.1	61.6	70.7	43.2	50.2	70.0	55.5	88.2	60.0	47.4	62.5	65.8
YOLOv3 [17]	43.1	68.6	73.4	46.2	53.2	73.4	59.6	89.7	63.8	47.8	65.0	72.7
YOLOv3-tiny [41]	27.2	56.1	63.1	33.3	35.5	60.1	45.9	80.3	43.4	30.6	53.9	64.7
YOLOv5n [18]	37.8	64.2	69.9	42.4	46.4	67.8	54.7	87.7	56.0	42.1	61.2	70.0
YOLOv8n [19]	41.3	67.7	73.4	45.0	51.1	72.0	58.4	88.6	63.1	45.8	64.8	72.2
YOLOv9s [42]	45.3	**70.2**	77.5	49.0	55.7	75.8	62.2	90.5	66.3	49.4	67.4	**74.8**
AS-YOLOv5 [43]	**46.6**	**70.2**	72.3	50.1	54.6	74.8	61.9	92.9	68.4	48.5	**68.9**	74.3
Light-YOLOv5 [44]	37.6	64.0	70.4	42.8	45.4	68.9	54.8	89.1	62.4	43.4	63.9	74.2
SCSDet [45]	43.1	69.8	77.8	46.7	54.6	**75.5**	62.6	91.2	67.3	55.3	67.9	72.6
MsDSOD	46.2	68.1	**78.3**	**53.7**	**56.9**	75.3	**63.1**	**93.6**	**70.0**	**55.7**	68.8	73.8

TABLE VII
COMPARISON EXPERIMENT OF MSDSOD

Methods \ Indicators	Backbone	Input shape	mAP	AR	F1	Parameters (MB)	FLOPs (G)	FPS	Energy cost (KJ)
DETR [20]	Eficientdet	640 × 640	49.1	59.1	53.6	41.6	75.3	33.1	25.0
FCOS [23]	ResNet-50	640 × 640	58.2	65.7	74.3	25.3	36.8	41.4	17.8
Cascade R-CNN [26]	ResNet-101	640 × 640	59.9	65.9	74.8	88.2	262.1	55.2	14.3
Cascade Mask R-CNN [27]	ResNet-FPN	640 × 640	55.5	62.9	72.7	42.0	157.7	46.1	16.4
YOLOv3 [17]	DarkNet-53	640 × 640	59.6	66.3	73.8	58.7	154.6	59.9	13.8
YOLOv3-tiny [41]	DarkNet-53	640 × 640	45.9	55.1	64.2	8.3	12.9	185.2	4.5
YOLOv5n [18]	CSPDarkNet	640 × 640	54.7	63.3	71.3	**1.7**	**4.2**	**588**	**2.2**
YOLOv8n [19]	CSPDarkNet	640 × 640	58.4	66.0	75.1	2.9	8.1	212.8	3.9
YOLOv9s [42]	CSPDarkNet	640 × 640	62.2	67.9	73.2	6.8	26.7	128.2	6.5
AS-YOLOv5 [43]	CSPDarkNet	640 × 640	61.9	**69.0**	72.3	57.0	171.4	90.1	9.2
Light-YOLOv5 [44]	Shufflenetv2	640 × 640	54.8	66.1	70.9	17.9	24.6	112.4	7.4
SCSDet [45]	SCSDet	640 × 640	62.6	67.7	74.5	9.4	28.9	73.6	11.6
MsDSOD	CSPResNet101	640 × 640	**63.1**	68.8	**75.2**	25.3	36.8	63.1	13.1

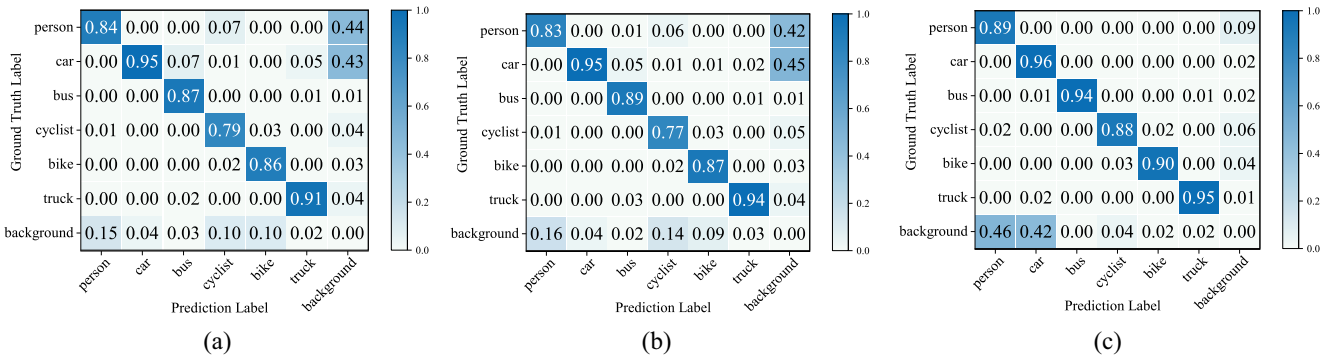
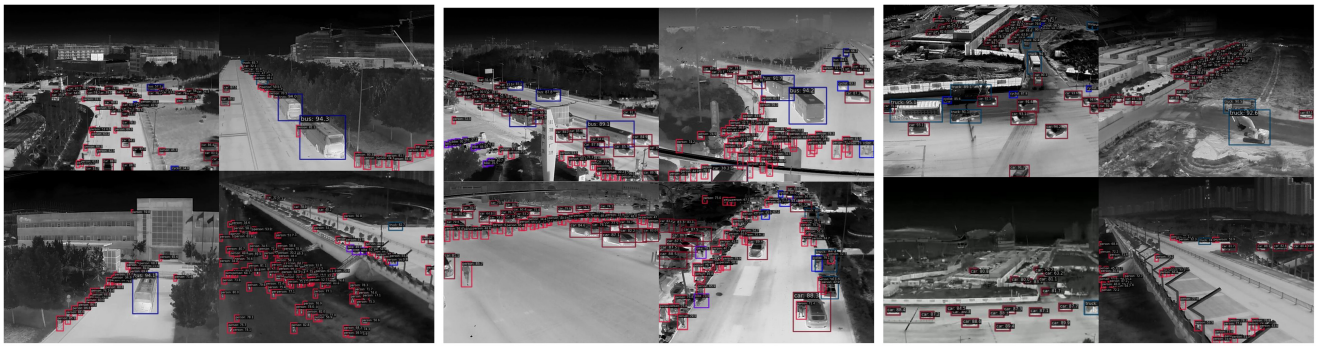


Fig. 14. Confusion matrix. (a) YOLOv5n. (b) YOLOv8n. (c) MsDSOD.

the classification results of our algorithms in more detail, the confusion matrix figure of the MsDSOD algorithms is also attached as shown in Fig. 14. Compared to the confusion matrix of other two algorithms, the proposed algorithm has better detection accuracy and lower probability of confusion between categories.

In Table VI, mAP, AP_{50}, AP_{75}, and AP_s of MsDSOD are the highest. Compared to SCSDet, the mAP improves by 0.80%. Compared to YOLOv9s, the mAP improves by 1.44%. When IoU threshold is 0.5, AP_{50} of MsDSOD reaches 93.6. When IoU threshold is 0.75, AP_{75} of MsDSOD reaches 70.0. AP of MsDSOD is the higher for detecting objects of different sizes. AP_s of MsDSOD improves by 0.72% over SCSDet, and by 8.16% over Cascade R-CNN in small object detection. This is due to the fact that the proposed algorithm employs a two-step detection approach to locally detect dense target regions and identify dense small targets more accurately.

In medium-sized object detection, AP_M of MsDSOD is not far from the highest of AS-YOLOv5, AP_L of MsDSOD reduces by 1.36% compared to YOLOv9s. This is because in convolutional neural networks, deep feature maps usually have a higher degree of abstraction and are suitable for recognizing complex patterns or large objects. Shallow feature maps retain more original visual information in them and are more suitable for detecting fine features, such as edges and textures. The backbone network CSPResNet101 is introduced, which relies



(a) (b) (c)

Fig. 15. Object detection effect in different scenarios. (a) High proportion of small object. (b) Object occlusion. (c) Significant noise interference.

on the residual learning mechanism to ensure the training performance of the deeper network by copying the features of the shallow network to the deeper network for effective feature extraction. The proposed algorithm relies too much on shallow features for prediction and thus performs poorly when dealing with large targets.

Table VII shows the comparison results of various algorithms in terms of metrics, such as accuracy, parameters, and energy cost. Same as mAP, F1 is also the highest and AR is not much different from the highest value of AS-YOLOv5. Our algorithm is not superior in Parameters, FLOPs, FPS, and Energy cost. The best in these aspects is the YOLOv5n algorithm. Since MsDSOD aims at accuracy in detecting small targets and is deployed in a cloud data center, it has not been designed for light weight and recognition speed in order to be mounted on AAV. The detection effect of MsDSOD is shown in Fig. 15. MsDSOD can detect small objects in high proportion of small objects scenes, objects with occlusions scenes, and significant noise interference scenes.

The object detection architecture we have constructed is that the image fusion algorithm is deployed on AAVs, and the object detection algorithm is deployed on the cloud center. Deploying algorithms on AAVs requires consideration of several aspects, including computational resources, energy consumption, and real-time performance. Therefore, we consider using dedicated hardware, such as FPGAs, GPUs, and ASICs to accelerate computation. Meanwhile, we use techniques, such as model pruning, quantization, and knowledge distillation to reduce the number of parameters and the computational complexity of the model. However, the accuracy of supervised learning models tends to decrease rapidly after pruning and compression. That is why our MIF-CGAN uses GANs. Although the proposed methods involve some additional training effort, they can maintain fusion accuracy, making the use of the proposed algorithms on AAVs more efficient and reliable.

VII. Conclusion

To improve the precision of object detection in AAVs, we propose a multiband infrared image fusion and dense small object detection method. An MIF-CGAN network is proposed, which exploits the complementary information provided by each band, increasing the effectiveness and detail of subsequent target detection analysis. The Transformer-based MsDSOD network disentangles tiny objects embedded in densely regions at multiple scales. The performance of the proposed method is significant compared to conventional algorithms. Specifically, the MIF-CGAN has obvious advantages in EN, AG, and SD metrics. MsDSOD consistently increases detection precision across different object dimensions, and excels in scenarios characterized by complex configurations of dense small objects and instances of object occlusion. The simulation experiments of the two proposed algorithms have shown positive results. Our future research focuses on deploying the algorithms on AAVs for real-world validation.

References

[1] X. Dai et al., "Multi-task faster R-CNN for nighttime pedestrian detection and distance estimation," *Infrared Phys. Technol.*, vol. 115, Jun. 2021, Art. no. 103694.

[2] P. Song, L. Weizman, J. F. C. Mota, Y. C. Eldar, and M. R. Rodrigues, "Coupled dictionary learning for multi-contrast MRI reconstruction," *IEEE Trans. Med. Imag.*, vol. 39, no. 3, pp. 621–633, Mar. 2019.

[3] M. Li, R. Pei, T. Zheng, Y. Zhang, and W. Fu, "FusionDiff: Multi-focus image fusion using denoising diffusion probabilistic models," *Exp. Syst. Appl.*, vol. 238, Mar. 2024, Art. no. 121664.

[4] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "NAS-FPN: Learning scalable feature pyramid architecture for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7036–7045.

[5] Z. Lin, Y. Wang, J. Zhang, and X. Chu, "DynamicDet: A unified dynamic architecture for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 6282–6291.

[6] H. Deng, X. Sun, and X. Zhou, "A multiscale fuzzy metric for detecting small infrared targets against chaotic cloudy/sea-sky backgrounds," *IEEE Trans. Cybern.*, vol. 49, no. 5, pp. 1694–1707, May 2019.

[7] Z. Liang, W. Liu, and R. Yao, "Contrast enhancement by nonlinear diffusion filtering," *IEEE Trans. Image Process.*, vol. 25, no. 2, pp. 673–686, Feb. 2016.

[8] C.-Y. Wang et al., "CSPNET: A new backbone that can enhance learning capability of CNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2020, pp. 390–391.

[9] J. Ma, W. Yu, P. Liang, C. Li, and J. Jiang, "FusionGAN: A generative adversarial network for infrared and visible image fusion," *Inf. Fusion*, vol. 48, pp. 11–26, Aug. 2019.

[10] X. Wang, Z. Guan, W. Qian, J. Cao, S. Liang, and J. Yan, "Cs2fusion: Contrastive learning for self-supervised infrared and visible image fusion by estimating feature compensation map," *Inf. Fusion*, vol. 102, Feb. 2024, Art. no. 102039.

[11] X. Yang, H. Huo, J. Li, C. Li, Z. Liu, and X. Chen, "DSG-fusion: Infrared and visible image fusion via generative adversarial networks and guided filter," *Exp. Syst. Appl.*, vol. 200, Aug. 2022, Art. no. 116905.

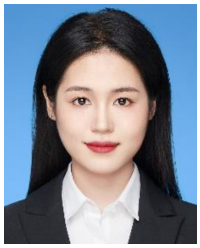
[12] Z. Zhao et al., "CDDFuse: Correlation-driven dual-branch feature decomposition for multi-modality image fusion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 5906–5916.
[13] J. Li, H. Huo, C. Li, R. Wang, and Q. Feng, "AttentionFGAN: Infrared and visible image fusion using attention-based generative adversarial networks," *IEEE Trans. Multimedia*, vol. 23, pp. 1383–1396, 2020.
[14] Y. Li, Q. Hou, Z. Zheng, M.-M. Cheng, J. Yang, and X. Li, "Large selective kernel network for remote sensing object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 16794–16805.
[15] B. Du, Y. Huang, J. Chen, and D. Huang, "Adaptive sparse convolutional networks with global context enhancement for faster object detection on drone images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 13435–13444.
[16] X. Zhu, S. Lyu, X. Wang, and Q. Zhao, "TPH-YOLOV5: Improved YOLOV5 based on transformer prediction head for object detection on drone-captured scenarios," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 2778–2788.
[17] J. Redmon and A. Farhadi. "YOLOV3: An incremental improvement." 2018. [Online]. Available: https://pjreddie.com/media/files/papers/YOLOv3.pdf
[18] G. Jocher. "Ultralytics/YOLOV5: V3.1—Bug fixes and performance improvements." Oct. 2020. [Online]. Available: https://zenodo.org/records/4154370
[19] G. Jocher, A. Chaurasia, and J. Qiu. "Ultralytics YOLO." Jan. 2023. [Online]. Available: https://github.com/ultralytics/ultralytics
[20] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10781–10790.
[21] M. Ye et al., "Cascade-DETR: Delving into high-quality universal object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 6704–6714.
[22] C. Xu et al., "Dynamic coarse-to-fine learning for oriented tiny object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 7318–7328.
[23] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. ICCV*, 2019, pp. 9626–9635.
[24] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.
[25] 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.
[26] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6154–6162.
[27] Z. Cai and N. Vasconcelos, "Cascade R-CNN: High quality object detection and instance segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 5, pp. 1483–1498, May 2021.
[28] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4700–4708.
[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[30] B. Li, B. Wang, J. Liu, Z. Qi, and Y. Shi, "s-LWSR: Super lightweight super-resolution network," *IEEE Trans. Image Process.*, vol. 29, pp. 8368–8380, 2020.
[31] R. Solovyev, W. Wang, and T. Gabruseva, "Weighted boxes fusion: Ensembling boxes from different object detection models," *Image Vis. Comput.*, vol. 107, Mar. 2021, Art. no. 104117.
[32] K. Takumi, K. Watanabe, Q. Ha, A. Tejero-De-Pablos, Y. Ushiku, and T. Harada, "Multispectral object detection for autonomous vehicles," in *Proc. Thematic Workshops ACM Multimedia*, 2017, pp. 35–43.
[33] C. S. Xydeas et al., "Objective image fusion performance measure," *Electron. Lett.*, vol. 36, no. 4, pp. 308–309, 2000.
[34] H. Li and X.-J. Wu, "DenseFuse: A fusion approach to infrared and visible images," *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2614–2623, May 2019.
[35] J. Ma, H. Xu, J. Jiang, X. Mei, and X.-P. Zhang, "DDcGAN: A dual-discriminator conditional generative adversarial network for multi-resolution image fusion," *IEEE Trans. Image Process.*, vol. 29, pp. 4980–4995, 2020.
[36] J. Ma, H. Zhang, Z. Shao, P. Liang, and H. Xu, "GANMcC: A generative adversarial network with multi-classification constraints for infrared and visible image fusion," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–14, 2020.
[37] H. Li, X.-J. Wu, and J. Kittler, "RFN-Nest: An end-to-end residual fusion network for infrared and visible images," *Inf. Fusion*, vol. 73, pp. 72–86, Sep. 2021.
[38] H. Xu, H. Zhang, and J. Ma, "Classification saliency-based rule for visible and infrared image fusion," *IEEE Trans. Comput. Imag.*, vol. 7, pp. 824–836, 2021.
[39] H. Zhang, J. Yuan, X. Tian, and J. Ma, "GAN-FM: Infrared and visible image fusion using GAN with full-scale skip connection and dual Markovian discriminators," *IEEE Trans. Comput. Imag.*, vol. 7, pp. 1134–1147, 2021.
[40] J. Liu et al., "Target-aware dual adversarial learning and a multi-scenario multi-modality benchmark to fuse infrared and visible for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 5802–5811.
[41] P. Adarsh, P. Rathi, and M. Kumar, "YOLO V3-tiny: Object detection and recognition using one stage improved model," in *Proc. 6th Int. Conf. Adv. Comput. Commun. Syst. (ICACCS)*, 2020, pp. 687–694.
[42] C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao, "YOLOv9: Learning what you want to learn using programmable gradient information," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2025, pp. 1–21.
[43] X. Xiong et al., "Adaptive feature fusion and improved attention mechanism-based small object detection for UAV target tracking," *IEEE Internet Things J.*, vol. 11, no. 12, pp. 21239–21249, Jun. 2024.
[44] J. Han, R. Cao, A. Brighente, and M. Conti, "Light-YOLOV5: A lightweight drone detector for resource-constrained cameras," *IEEE Internet Things J.*, vol. 11, no. 6, pp. 11046–11057, Mar. 2024.
[45] G. Mao, H. Liang, Y. Yao, L. Wang, and H. Zhang, "Split-and-shuffle detector for real-time traffic object detection in aerial image," *IEEE Internet Things J.*, vol. 11, no. 8, pp. 13312–13326, Apr. 2024.

Yuhuai Peng received the Ph.D. degree in communication and information systems from Northeastern University, Shenyang, China, in 2013.

He is currently a Professor with the Department of Communications and Electronic Information, Northeastern University. His research interests include Internet of Things, cyber–physical systems, intelligent information networks, industrial communication networks, edge computing, and prognostics and health management.

Jing Wang received the M.E. degree in electronic science and technology from Yanshan University, Qinhuangdao, China, in 2020. She is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering, Northeastern University, Shenyang, China.

Her research interests include digital twin, artificial intelligence, and Internet of Things.



Wenqian Wang received the undergraduate degree in information management and information systems from Guangxi University, Nanning, China, in June 2020, and the M.E. degree in information and communication engineering from Northeastern University, Shenyang, China, in July 2023.

Her research interests are in infrared image processing.

Lei Liu (Member, IEEE) received the B.Eng. degree in communication engineering from Zhengzhou University, Zhengzhou, China, in 2010, and the M.Sc. and Ph.D. degrees in communication engineering from Xidian University, Xi'an, China, in 2013 and 2019, respectively.

From 2013 to 2015, he was employed a subsidiary of China Electronics Corporation, Shenzhen, China. From 2018 to 2019, he was supported by the China Scholarship Council to be a visiting Ph.D. student with the University of Oslo, Oslo, Norway. He is currently a Lecturer with the Department of Electrical Engineering and Computer Science, Xidian University. His research interests include vehicular ad hoc networks, intelligent transportation, mobile-edge computing, and Internet of Things.

Mohammed Atiquzzaman (Life Senior Member, IEEE) received the B.S. degree in electrical and electronic engineering from Bangladesh University of Engineering and Technology, Dhaka, Bangladesh, in 1982, and the M.S. and Ph.D. degrees in electrical engineering and electronics from The University of Manchester, Manchester, U.K., in 1984 and 1987, respectively.

He currently holds the Edith Kinney Gaylord Presidential Professorship with the School of Computer Science, University of Oklahoma, Norman, OK, USA. His research interests are in communications switching, transport protocols, wireless and mobile networks, satellite networks, and optical communications.

Dr. Atiquzzaman is the Editor-in-Chief of *Journal of Networks and Computer Applications* and the Founding Editor-in-Chief of *Vehicular Communications*, and has served/serving on the editorial boards of various IEEE journals and co-chaired numerous IEEE international conferences, including IEEE Globecom.

Mohsen Guizani (Fellow, IEEE) received the B.S. (with Distinction), M.S., and Ph.D. degrees in electrical and computer engineering from Syracuse University, Syracuse, NY, USA, in 1985, 1987, and 1990, respectively.

He is currently a Professor of Machine Learning and the Associate Provost with Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE. Previously, he worked in different institutions in the USA. His research interests include applied machine learning and artificial intelligence, Internet of Things, and application of machine learning in communication.

Prof. Guizani is also a Senior Member of ACM.

Schahram Dustdar (Fellow, IEEE) received the Ph.D. degree in business informatics from the University of Linz, Linz, Austria, in 1992.

He is currently a Professor of Computer Science (Informatics) with a focus on Internet technologies heading the Distributed Systems Group, TU Wien, Vienna, Austria, and the part-time ICREA Research Professor, Universitat Pompeu Fabra Barcelona, Barcelona, Spain.

Prof. Dustdar was a recipient of the ACM Distinguished Scientist Award and the IBM Faculty Award. He has been the Chairman of the Informatics Section of the Academia Europaea, and a member of the IEEE Conference Activities Committee, the Section Committee of Informatics of the Academia Europaea, and the Academia Europaea: The Academy of Europe, Informatics Section.