LW-IRSTNet: Lightweight Infrared Small Target Segmentation Network and Application Deployment

Renke Kou, Chunping Wang, Ying Yu, Zhenming Peng, *Member, IEEE*, Mingbo Yang, Fuyu Huang, and Qiang Fu

***Abstract*— Efficiently and accurately separating infrared (IR) small targets from complex backgrounds presents a significant challenge. Numerous studies in the literature have proposed various feature fusion modules designed specifically to enhance the extraction of IR small target features. While these designs offer some incremental improvement to the accuracy of IR small target detection, they come at a steep cost of significantly increasing network parameters and floating-point operations per second (FLOPs). Striving for a balance between computational efficiency and model accuracy, we decided to forgo these complex feature fusion modules. Instead, we developed a new lightweight encoding and decoding structure known as the lightweight IR small target segmentation network (LW-IRSTNet). This structure integrates regular convolutions, depthwise separable convolutions, atrous convolutions, and asymmetric convolutions modules. In addition, we devised postprocessing modules, including an eight-neighborhood clustering algorithm and an online target feature adjustment strategy. Experimental results indicate that: 1) the segmentation accuracy metrics of LW-IRSTNet match the best results of 14 state-of-the-art comparative baselines; 2) the parameters and FLOPs of LW-IRSTNet at only 0.16M and 303M, respectively, are significantly smaller in comparison to these baselines; and 3) the postprocessing modules enhance both user-friendliness and the robustness of algorithm deployment. Moreover, LW-IRSTNet has been successfully implemented on both embedded platforms and websites, expanding its range of applications. Utilizing the open neural network exchange (ONNX) framework, neural network processing unit (NPU) acceleration, and CPU multithreaded resource allocation, we have been able to achieve high-performance inference capabilities, as well as online dynamic threshold adjustment with the LW-IRSTNet. The source codes for this project can be accessed at https://github.com/kourenke/LW-IRSTNet.**

***Index Terms*— Embedded deployment, infrared (IR) small target segmentation, lightweight design, real-time inference.**

Manuscript received 18 June 2023; revised 18 July 2023; accepted 9 September 2023. Date of publication 12 September 2023; date of current version 3 October 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62171467 and in part by the Natural Science Foundation of Hebei Province under Grant F2021506004. *(Corresponding authors: Qiang Fu; Renke Kou.)*

Renke Kou, Chunping Wang, Ying Yu, Fuyu Huang, and Qiang Fu are with the Shijiazhuang Campus, Army Engineering University, Shijiazhuang 050003, China (e-mail: krk_Optics@aeu.edu.cn; Wang_ChunPing@aeu.edu.cn; yingyu@sanyau.edu.cn; hfyoptics@163.com; Fu_Qiang@aeu.edu.cn).

Zhenming Peng is with the Laboratory of Imaging Detection and Intelligent Perception, School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China (e-mail: zmpeng@uestc.edu.cn).

Mingbo Yang is with the School of Mechanical and Materials Engineering, North China University of Technology, Beijing 100144, China (e-mail: yangmingbo@ncut.edu.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/TGRS.2023.3314586, provided by the authors.

Digital Object Identifier 10.1109/TGRS.2023.3314586

I. Introduction

INFRARED(IR) small target detection technology plays a vital role in early warning and reconnaissance tasks [1]. Given the rigorous demands for precision and speed in remote target detection within these tasks, many researchers have invested notable efforts into refining IR small target detection algorithms over the years.

IR small target detection algorithms primarily fall into two categories: track before detection (TBD) and detect before tracking (DBT). Since TBD algorithms require the association of multiframe images and do not perform optimally in real-time scenarios, a significant portion of research has focused on the development of DBT algorithms, which rely on single-frame detection [2]. Over the recent decades, single-frame IR small target detection has primarily utilized model-driven algorithms. These algorithms leverage prior knowledge (specifically, the physical and imaging characteristics of the target) to form reasonable assumptions. They typically encompass background estimation methods [3], [4], morphological methods [5], [6], local contrast methods [7], [8], [9], [10], directional derivative/gradient methods [11], [12], frequency-domain methods [13], [14], and low-rank sparse methods [15], [16], [17], among others. However, these model-driven algorithms require extensive parameter setting. Consequently, they can underperform when there are significant changes to the target's size, shape, signal-to-noise ratio, and background clutter, leading to lower detection performance in practical implementation.

With the development of deep learning, particularly since the release of the IR small target dataset by Wang et al. [18] in 2019, a plethora of algorithms for IR small target segmentation tasks have emerged [19]. Dai et al. [20] proposed an asymmetric context modulation module (ACM) with either feature pyramid network (FPN) [21] or UNet [22] as the backbone and introduced the high-quality SIRST dataset. Zhang et al. [23] designed an attention-guided pyramid context network (AGPCNet) with residual network (ResNet) [24] as the backbone and expanded the SIRST dataset. Li et al. [25] devised a dense nested attention network (DNANet) with UNet as the backbone, which facilitated progressive interaction between high- and low-level features and published the NUDT-SIRST dataset. Huang et al. [26] utilized Visual Geometry Group (VGG) [27] as the backbone and designed a local similarity pyramid module (LSPM) to effectively capture multiscale features of IR small targets. Zuo et al. [28] developed a multiscale feature fusion pyramid module (AFFPN)

TABLE I
HYPERPARAMETERS SETTING IN DAAA MODULES

DAAA modules		Kernel size	Atrous ratio	Number of channels in convolutional branches		
DAAA1	Dw-Conv.	3	-	64	128	64
	At-Conv.	3	2	64	16	64
	As-Conv.	7	-	64	16	64
	At-Conv.	3	4	64	16	64
DAAA2	Dw-Conv.	3	-	Repeat DAAA1		
	At-Conv.	3	8			
	As-Conv.	7	-			
	At-Conv.	3	16			
DAAA 3 and 4: Repeat DAAA 1 and 2						

with ResNet as the backbone, aiming to address target loss in deep networks. Wang et al. [29] utilized ResNet as the backbone and introduced a method that involved using a region proposal network (RPN) to extract candidate targets, an fully convolutional network (FCN) [30] network to generate feature maps, and a transformer [31] to determine candidate targets. In summary, the algorithms discussed above follow a general pattern: 1) they employ classical networks such as VGG, UNet, and ResNet for encoding; 2) the multiscale feature fusion modules (MFFM) are designed on the high-level feature map; and 3) the channel and spatial attention mechanism fusion modules are designed on the same scale feature map corresponding to encoding and decoding.

By reproducing the codes from the aforementioned literature, the experimental results indicate that adding MFFM and channel–spatial attention fusion modules can slightly enhance the detection accuracy. However, it also leads to a significant increase in network parameters and floating-point operations per second (FLOPs). In light of this, we propose a lightweight IR small target segmentation network (LW-IRSTNet) that can be deployed on embedded platforms. The contributions of our study are given as follows.

1) LW-IRSTNet achieves excellent segmentation results on multiple IR small target datasets without relying on feature fusion and context modules. It has only 0.16 M parameters and 303 M FLOPs, striking a balance between segmentation accuracy and speed.
2) In LW-IRSTNet, we propose a bottleneck structure known as the depthwise separable-atrous-asymmetric-atrous module (DAAA). This structure not only minimizes computational complexity but also effectively learns multiscale features of IR small targets.
3) A postprocessing module is introduced. This not only enhances the algorithm's user-friendly and robustness in application deployment but also broadens its range of applications. It can meet the requirements of real-time, high-precision, and online dynamic target feature adjustment. We also deployed LW-IRSTNet on the embedded platform and website to realize engineering applications.

II. RELATED WORK

A. Design Strategy for IRSTNet

To effectively extract IR small target features, existing IRSTNets employ various design strategies. The following strategies have been summarized.

1) *Asymmetric Context Feature Fusion Strategy:*Both ACM [20] and AGPCNet [23] utilize the top-down channel attention mechanism and the bottom-up spatial attention mechanism to extract high-level semantic information and low-level feature information, respectively.

2) *Densely Nested Interactive Feature Fusion Strategy:* DNANet [25] achieves progressive interaction between high- and low-level features through dense nested interaction networks. By repeatedly fusing and enhancing contextual information, DNANet effectively combines and utilizes the information of small targets.

3) *Multiscale Feature Fusion Strategy:* Since IR small targets exhibit changes across different scales, LSPM [26] and AFFPN [28] construct multiscale feature maps on the high-level feature maps using atrous convolution and adaptive global average pooling. This enables the network to learn the characteristics of IR small targets at different scales.

4) *Generative Adversarial Strategy:* Zhao et al. [32] proposed a generative adversarial network that can predict IR small targets as a special type of noise based on the learned data distribution and hierarchical characteristics. This strategy leverages the generative power of the adversarial network to enhance target segmentation.

5) *Based on Transformer + CNN Structure:* Liu et al. [33] were the first to use the transformer architecture for IR small target segmentation to capture large-range dependencies. Wu et al. [34] introduced the multilevel TransUNet (MTUNet), which combines a hybrid vision transformer (ViT) encoder and convolutional neural networks (CNN) to extract multilevel features.

While the aforementioned designs aim to extract low-level feature information and high-level semantic information of IR small targets, they often come at the cost of increased computational complexity. Given that the features of IR small targets are not very distinctive, this study focuses on minimizing network computational complexity while ensuring accurate segmentation of IR small targets.

B. Lightweight Network Design Strategy

A complex network model typically achieves higher accuracy compared to a simpler one. However, due to its large storage requirements and consumption of computing resources, effectively applying this network to embedded platforms is challenging. As a result, there has been significant research focused on developing lightweight networks. Let us review the key design strategies for lightweight networks, which can be categorized into manually designed lightweight network structures, model compression, and automated neural network architectures.

1) Manually Designed Lightweight Network Structures: Several manually designed lightweight network structures, such as Xception [35], SqueezeNet [36], MobileNet V1-V3 [37], [38], [39], ShffleNet V1-V2 [40], [41], MixNet [42], GhostNet [43], and short-term dense concatenate network (STDC) [44], have successfully reduced computational complexity while maintaining accuracy. These design strategies typically include: 1) reducing backbone depth, kernel size, the number of channels, and layers; 2) designing efficient

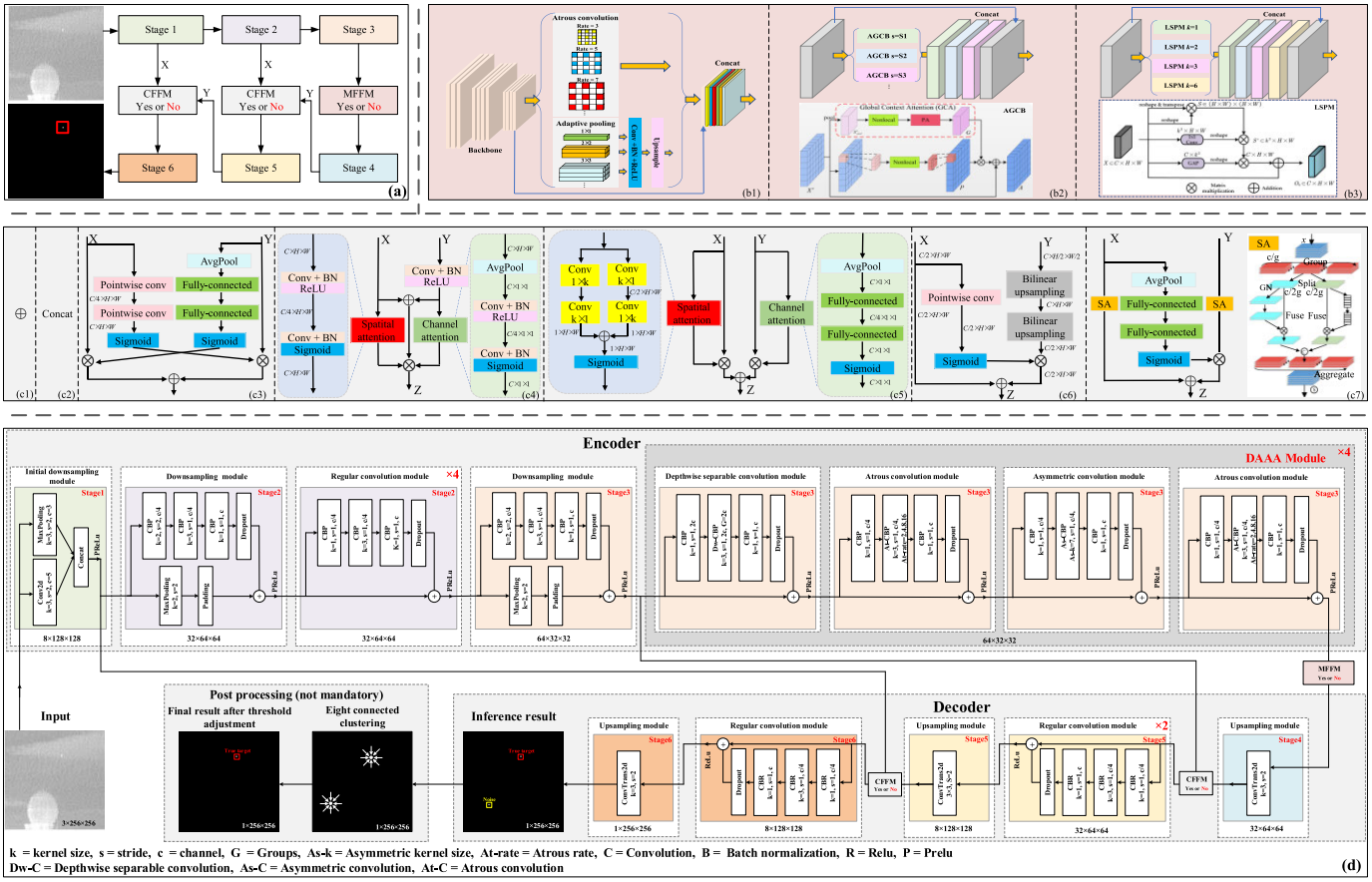


Fig. 1. (a) and (d) Architecture and detailed flowchart of LW-IRSTNet, respectively. (b1)–(b3) Different MFFM, which are derived from [23], [26], and [63]. (c1)–(c7) Different contextual feature fusion modules, where (c3)–(c7) are derived from [20], [23], [28], [64], and [65], respectively.

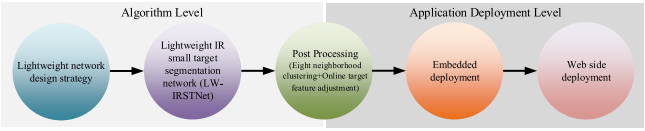


Fig. 2. Research roadmap of this study.

convolution operations, such as using group convolution, depthwise separable convolution, or mixed depthwise separable convolution instead of traditional convolutional layers; and 3) introducing channel shuffling and attention mechanism strategies to enhance the accuracy.

2) Model Compression: Model compression techniques can be categorized into knowledge distillation, pruning, quantization, and low-rank decomposition.

1) Knowledge distillation [45], [46], [47], [48] involves creating a network of "student" models from larger "teacher" models, enabling the small model to benefit from the feature extraction capabilities of the larger model.

2) Pruning [49], [50], [51] refers to removing weights in the network structure to compress the network.

3) Quantization [52], [53], [54], [55] involves converting floating-point model parameters into low-bit-width fixed-point or binary numbers to reduce storage space and computational requirements.

4) Low-rank decomposition [56], [57] focuses on decomposing weight matrices into low-rank matrices, reducing the computational complexity of the network.

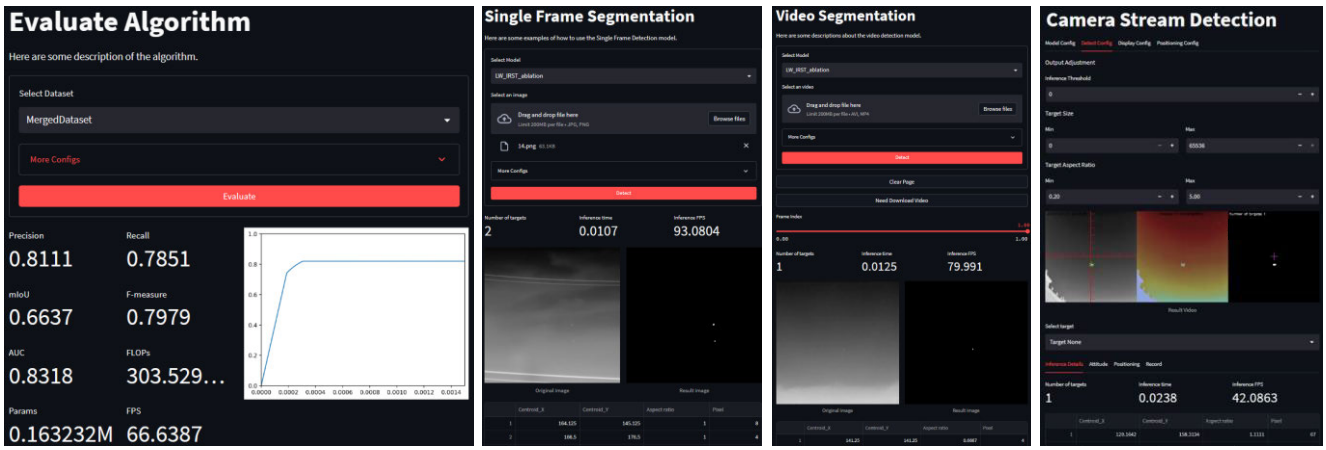
3) Automated Neural Network Architectures: The development of automated neural architecture search (NAS) has automated the network design process to some extent. NAS methods [58], [59], [60], [61] have achieved remarkable results in designing lightweight networks. These techniques search for the optimal network architecture from a predefined search space based on performance constraints.

III. Network Architecture of LW-IRSTNet

In this section, we present the design of LW-IRSTNet and its postprocessing module based on the design principles of lightweight networks in previous research and practical engineering considerations. We begin by describing the overall structure of LW-IRSTNet and then provide the design details and motivations for each module within the network. Finally, we analyze the postprocessing module from an engineering application perspective.

A. Network Architecture

The architecture of LW-IRSTNet is shown in Fig. 1(a) and consists of six stages. To further reduce the computational complexity, we have omitted the MFFM [Fig. 1(b1)–(b3)] and context feature fusion modules [CFFM, Fig. 1(c2)–(c7)], focusing instead on carefully designing the encoding and decoding structure. Only the sum connection [Fig. 1(c1)] is used for feature fusion between encoding feature maps and decoding feature maps at the same scale.



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

Fig. 3. Web-based deployment of HCI systems. (a) Offline single-frame segmentation system. (b) Offline video segmentation system. (c) Online real-time segmentation system, which can dynamically adjust target feature thresholds. (d) Algorithm performance evaluation system.

TABLE II

COMPARATIVE ANALYSIS OF SEGMENTATION ACCURACY OF DIFFERENT ALGORITHMS

Year	Algorithm	FLOPs (M)	Params (M)	FPS	Fusion datasets (MDFA [18] (Black hot) + SIRST [20] (White hot)+ SIRST Aug [23] (White hot)+ NUDT-SIRST [25] (White hot))			
					Precision	Recall	mIoU	F1
Classic/Real-time/Significance object segmentation network								
2015	UNet [22]	65475.4447	34.5251	2.2231	0.7462	0.8298	0.6471	0.7858
2016	FusionNet [68]	123722.0065	81.6691	1.4374	0.7778	0.7214	0.5981	0.7485
2016	ENet [66]	**535.1505**	**0.3491**	12.2438	0.7114	0.7779	0.5913	0.7431
2018	BiSeNet [69]	10176.4396	23.0635	**20.5268**	0.7074	0.6846	0.5335	0.6958
2018	DeeplabV3+ [63]	20755.5758	54.6075	2.0119	0.7740	0.6716	0.5615	0.7192
2019	DABNet [70]	1321.1126	0.7520	10.8797	0.7236	0.7304	0.5710	0.7270
2019	DFANet [71]	541.7969	2.3662	3.6255	0.7109	0.6644	0.5230	0.6868
2020	CGNet [72]	888.4796	0.4917	11.8747	0.7274	0.7154	0.5641	0.7213
Segmentation networks designed for IR small targets								
2021	ACM-UNet [20]	**503.7925**	**0.5198**	**36.0439**	0.6756	0.7171	0.5334	0.6957
2021	LSPMNet [26]	61706.9384	31.1408	0.0728	0.6712	0.7220	0.5334	0.6957
2022	DNANet [25]	14279.0852	4.6968	2.7881	0.8275	0.7574	**0.6542**	0.7909
2023	AGPCNet [23]	43180.4202	12.3541	2.6315	0.8039	0.8047	**0.6727**	**0.8043**
2023	ABCNet [73]	83132.9812	73.5092	1.6007	0.7810	0.7772	0.6382	0.7791
2023	MTUNet [34]	6216.8125	8.2212	9.4076	0.8073	0.8014	**0.6727**	**0.8044**
2023	Ours (LW-IRSTNet)	**303.5300**	**0.1632**	**30.0332**	0.8113	0.7850	**0.6638**	**0.7979**

Note: The best three scores are shown in red, green, and blue colors, respectively.

We are pleased to report that the simplified network structure still achieves excellent segmentation results on IR small target datasets. The six stages of LW-IRSTNet include an initial downsampling module, downsampling modules, regular convolution modules, depthwise separable convolution (Dw-Conv.) modules, atrous convolution (At-Conv.) modules, asymmetric convolution (As-Conv.) modules, and upsampling modules, as shown in Fig. 1(d). For detailed network specifications, please refer to Section III-B.

B. Network Details

1) Initial Downsampling Module: Considering the high computational resource requirements of processing high-resolution input images, we adopt the design concept of Inception V2 [62] in the initial downsampling module to achieve parameter compression and accelerate inference time. Thus, the feature map of Stage 1 (F_{map1}) can be represented as

$$F_{\text{map1}} = P(B(\text{Concat}(\text{Conv2}d(x), \text{MaxPooling}(x)))) \quad (1)$$

where the kernel size, stride, and channel number of the convolution branch (Conv2d) are 3, 2, and 5, respectively. The kernel size, stride, and channel number of MaxPooling branch are 3, 2, and 3, respectively. x is the input image ($3 \times 256 \times 256$). B is batch normalization. P is the activation function PReLu.

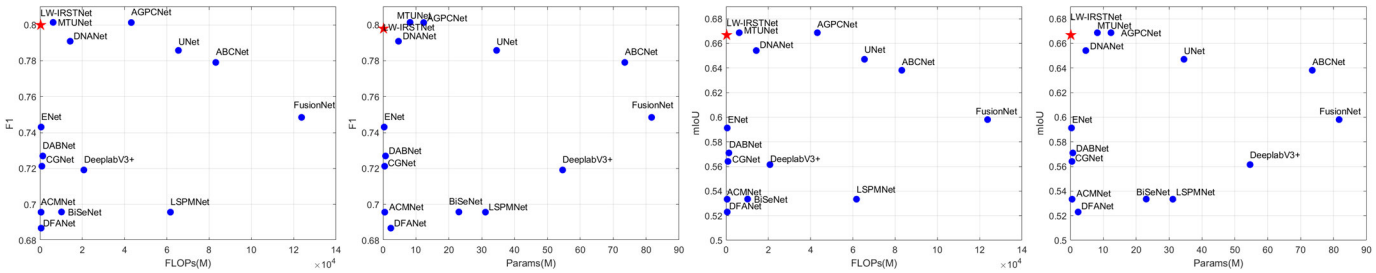


Fig. 4. Comparative analysis of accuracy and computational complexity of different algorithms. (a) $F1$ versus FLOPs. (b) $F1$ versus Params. (c) mIoU versus FLOPs. (d) mIoU versus Params.

2) Regular, Depthwise Separable, Atrous, and Asymmetric Convolution Modules: In Fig. 1(d), the structure of these four modules is similar, comprising a convolutional branch and a shortcut connection branch. The convolutional branch consists of three convolutional layers: the first 1×1 convolution is employed to reduce or expand channel dimensions, the main convolution (regular/depthwise separate/atrous) is used to extract target features, and the second 1×1 convolution is utilized to restore the original channel dimension.

To fully extract the shallow feature information from the target, we employ four regular convolution modules in Stage 2. To capture deep semantic information of multiscale IR small targets without increasing the parameter count, we incorporate four consecutive DAAA modules in Stage 3. The specific parameters are provided in Table I, representing the optimal hyperparameters determined through extensive ablation experiments.

3) Downsampling Module: The structure of the downsampling module is similar to that of a regular convolution module but with two key differences. First, in the convolutional branches, the first 1×1 convolution is replaced with a 2×2 convolution with a stride of 2. Second, in the shortcut connection branches, we initially reduce the feature map resolution by applying a 2×2 MaxPooling operation with a stride of 2. Subsequently, we zero-pad the activations to match the number of feature maps. Finally, the feature maps from the two branches are added and activated.

4) Upsampling Module: The ablation experiment shows that deconvolution is more helpful in improving segmentation accuracy compared to other upsampling methods (e.g., bilinear, bilinear + deconvolution).

5) Other Network Details: 1) Inspired by the design ideas of efficient neural network (ENet) [66] networks, we integrate the dropout strategy into the downsampling, regular, depthwise separable, atrous, and asymmetric convolution modules to further reduce parameters and mitigate overfitting. Specifically, we set the dropout rate to 0.01 in Stage 2 and 0.1 in Stages 3, 5, and 6.

2) Considering the shallow network depth, the activation function of PReLU is adopted in the encoding stage (Stages 2 and 3) and ReLU is adopted in the decoding stage (Stages 5 and 6).

3) After each convolutional layer, we perform batch normalization and activation.

4) To reduce the parameters and overall memory operations, we do not use bias terms throughout the entire network.

C. Postprocessing

1) Eight Neighborhood Clustering Algorithms: The IR small target segmentation task belongs to a binary classification task with extremely imbalanced positive and negative samples. After inference and threshold segmentation by LW-IRSTNet, the segmentation result can be obtained, defined as

$$\text{Output}_{\text{mask}} = \text{Infer}_{\text{saliencymap}} > \text{Th} \tag{2}$$

where $\text{Output}_{\text{mask}}$ is the segmentation result (0 is the background and 255 is the target). $\text{Infer}_{\text{saliencymap}}$ is the saliency map. Th is the threshold (in the training stage, and learn with 0 as the threshold to obtain the network weight parameters. Therefore, the threshold is set to 0 in the inference stage).

Next, we perform eight connected clusterings [25] on the segmentation result to obtain the number of targets, centroids, aspect ratio, pixel size, and coordinates in the image field of view, providing data support for subsequent tracking tasks.

The eight connected clustering algorithms can be represented as

$$\begin{cases} N_8(x_i, y_i) \cap N_8(x_j, y_j) \neq \emptyset \\ p_{(x_i,y_i)} = p_{(x_j,y_j)}, \quad \forall p_{(x_i,y_i)}, p_{(x_j,y_j)} \in \text{Output}_{\text{mask}} \end{cases} \tag{3}$$

where $p_{(x_i,y_i)}$ and $p_{(x_j,y_j)}$ are any two pixels on the segmentation result ($\text{Output}_{\text{mask}}$). If they have intersecting regions in eight neighborhoods and have the same value (0 or 255), then they belong to the same category (target or background).

2) Online Target Feature Adjustment Strategy: Under complex backgrounds and similar noise interference, any algorithm has the possibility of false alarms. Since optimizing algorithms is very difficult, why do not we change our perspective and improve the robustness of the algorithm from an engineering perspective?

Following this viewpoint, we propose an online target feature threshold adjustment strategy, which is defined as:

$$p_{T_i} = \begin{cases} 0, & T_{\text{size}} \leq m, \text{or } T_{\text{size}} \geq k, \\ & \text{or} T_{\text{ratio}} \leq l, \text{ or } T_{\text{ratio}} \geq n \\ 255, & else \end{cases} \tag{4}$$

where T_{size} is the pixel size of the target, m and k are the lower and upper threshold values of the target pixel size, respectively, T_{ratio} is the aspect ratio of the target, and l and n are the lower and upper thresholds for the target aspect ratio, respectively.

This strategy is particularly practical in the engineering application of LW-IRSTNet. Note that the setting of threshold

parameters needs to be dynamically adjusted based on prior knowledge or scene changes, which can further improve the segmentation accuracy of IR small targets, but it is not a necessary operation in a clean background.

IV. Application Deployment of LW-IRSTNet

From an academic standpoint, there has been a predominant focus on improving the accuracy of detection algorithms, often overlooking the importance of lightweight design and practical deployment. However, in military applications, the deployment of mobile devices is crucial for IR small target detection tasks. Therefore, Section III concentrated on designing LW-IRSTNet and proposing a postprocessing module suitable for engineering applications. This section now shifts the focus to the application deployment of LW-IRSTNet, with the overall research roadmap shown in Fig. 2.

A. Embedded Deployment

Within the academic community, the Pytorch framework is widely utilized for designing network models and conducting training and testing. To further optimize the inference speed of LW-IRSTNet and deploy it on embedded platforms, we convert the model (.pkl format) trained using the Pytorch framework into the open neural network exchange (ONNX) format. It is worth noting that major companies have also optimized their own inference frameworks for embedded devices, such as NVIDIA cuda NCNNs, Tencent neural network (TNN), Alibaba's mobile neural network (MNN), and NVIDIA's TensorRT. For the trained LW-IRSTNet network model, we have published four inference frameworks on Github: ONNX, NCNN, TNN, and MNN.

B. Design of the HCI System and Website Deployment

To display real-time information including the number of targets, centroid coordinates, aspect ratio, and pixel size, as well as dynamically adjust target feature thresholds (as a postprocessing module in Section III-C), we designed a human–computer interaction (HCI) system. This system can perform four tasks: offline single-frame segmentation, video segmentation, online real-time segmentation, and algorithmic performance evaluation. Considering the varying compatibility of algorithms with different embedded platforms, we have deployed this HCI system on the website, effectively resolving the compatibility problem, as shown in Fig. 3. Fig. 3(a) shows the algorithm evaluation system, which assesses various segmentation algorithms on different datasets based on metrics such as precision, recall, mean intersection over union (mIoU), $F1$, area under the curve (AUC), FLOPs, Params, and frames per second (FPS). Fig. 3(b) shows the offline single-frame segmentation system, capable of loading different single-frame IR images and segmenting small targets within them. Fig. 3(c) shows the offline video segmentation system, which can load different videos and perform small target segmentation within them. Fig. 3(d) shows the online IR small target real-time segmentation system, featuring the ability to dynamically adjust the inference threshold, target aspect ratio, and pixel size. It should be noted that this system also allows for the further deployment of the IR small target tracking algorithm [67].

V. Experimental Analysis

A. Basic Parameters

1) Training Details:

a) Software and hardware configuration: The system employs Ubuntu 18.04 as the operating system, with 32 GB of memory. The CPU is Intel[1] Xeon[1] E5-2630 v3 at 2.40 GHz × 32, and the GPU is NVIDIA GeForce RTX 2080 Ti.

b) Training hyperparameters: The batch size is set to 64, and the epoch is 100. The optimizer used is stochastic gradient descent (SGD), with a momentum of 0.9 and weight decay of $1e^{-4}$. The initial learning rate is 0.05, and the learning rate strategy is poly. The loss function is SoftLoULoss [23].

c) Datasets: Considering the lack of clear definitions for the shape and size of IR small targets within public datasets and the limited size of the data samples available, we combined four existing public datasets (MDFA [18], SIRST [20], SIRST Aug [23], and NUDT-SIRST [25]) for both training and testing purposes. This fusion allowed us to evaluate the robustness and multiscale detection capabilities of various algorithms. Also, we consider the black heat mode in IR imaging, so the background and target within the dataset MDFA are inverted in grayscale.

B. Compare Baselines

They are given as follows: U-Net [22], FusionNet [68], ENet [66], BiSeNet [69], DeeplabV3+ [63], DABNet [70], DFANet [71], CGNet [72], ACM-UNet [20], LSPMNet [26], DNANet [25], AGPCNet [23], ABCNet [73], MTUNet [34], and ours (LW-IRSTNet).

C. Evaluation Metrics

To compare the computational complexity and accuracy of 15 different algorithms, we use FLOPs, Params, FPS, mIoU, and $F1$ for analysis, as shown in Table II.

D. Comparative Experiment

1) Qualitative Comparative Analysis:

a) Computational complexity analysis: In Table II, LW-IRSTNet stands out with the lowest FLOPs and Params compared to the other 14 algorithms, making it suitable for deployment on embedded platforms. Furthermore, the inference speed of LW-IRSTNet is faster and ranks second among the other algorithms with an FPS of approximately 30.

To further enhance its speed, we adopted a two-step approach. First, we trained the model using the PyTorch framework and then converted it to the ONNX framework. Subsequently, we loaded the ONNX framework into OpenCV for inference. This optimization resulted in a substantial improvement in the inference speed of LW-IRSTNet, increasing it by 2.6 times from 30 to 79 FPS.

[1]Registered trademark.

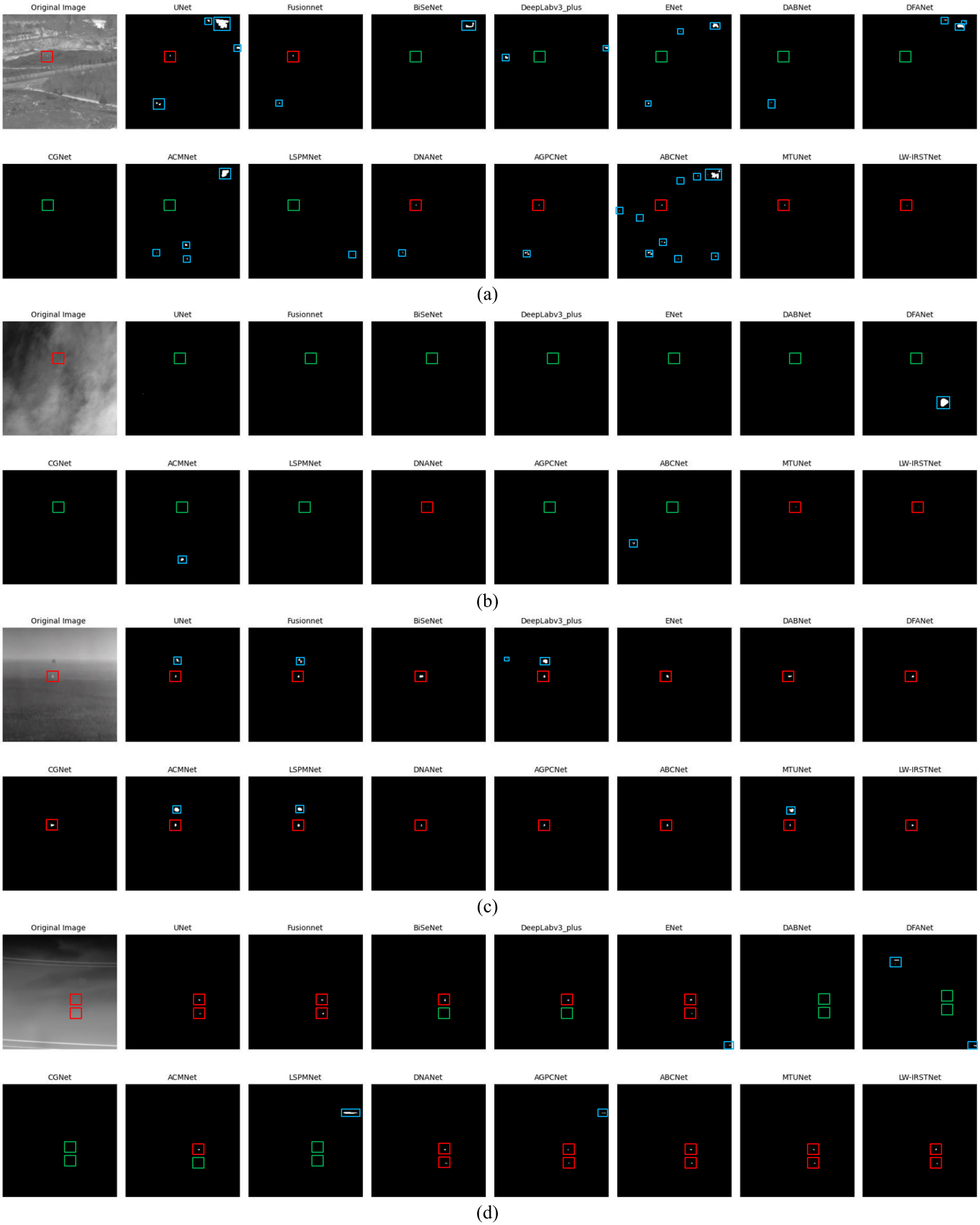


Fig. 5. Segmentation results of 15 algorithms for IR small targets of different scales and types in complex backgrounds. (a) Point target (drone) in the complex ground background. (b) Point target (aircraft) in the complex dense cumulus background. (c) Point target (reentry capsule) in the background of the sky-ground line. (d) Two point targets (birds) in the sky background. (e) Two point targets (drones) in the sky, ground, and architectural background. (f) Regional target (drone) in the sky, clouds, ground, and jungle background. (g) Regional target (drone) in the clean sky background.

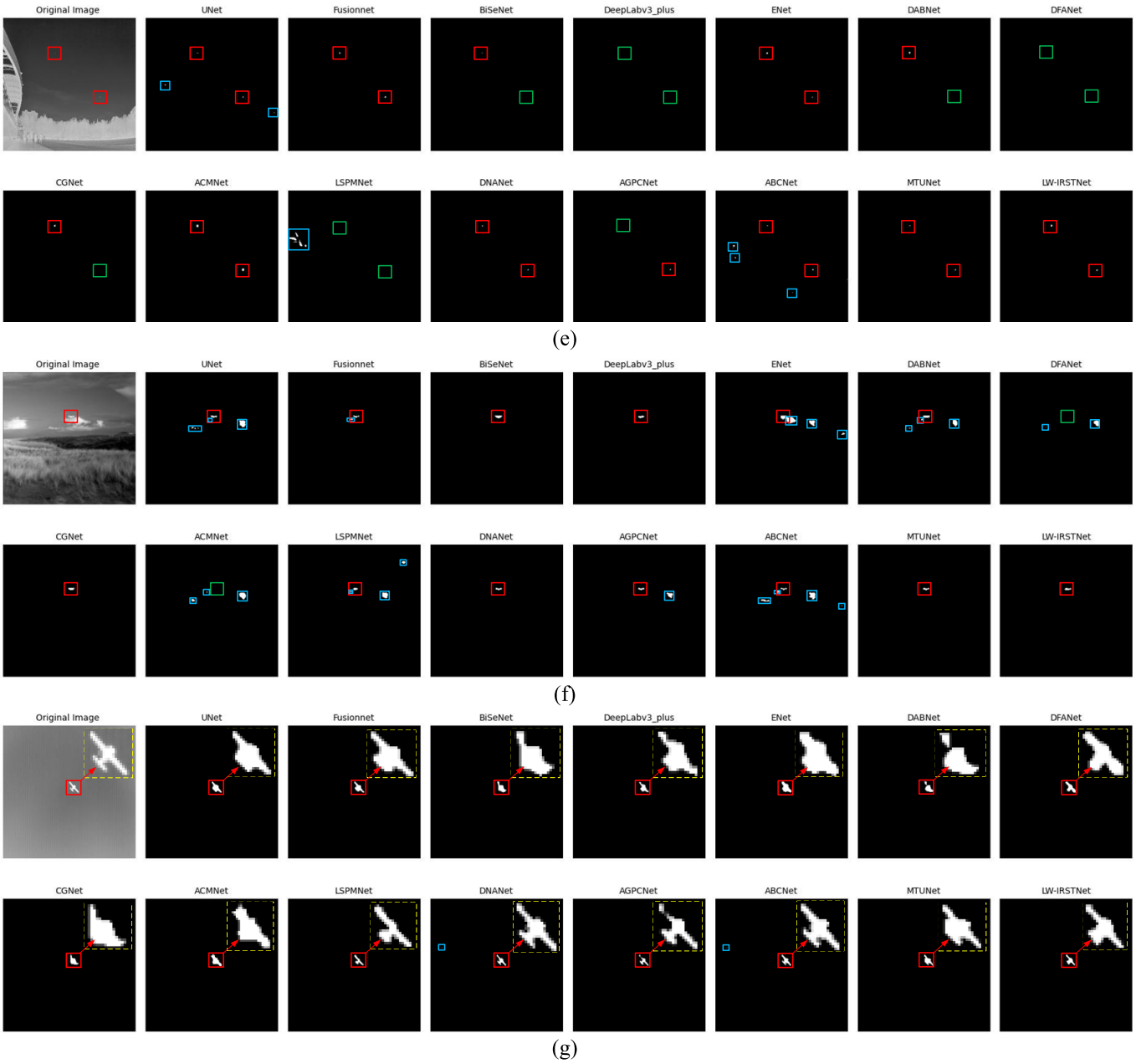


Fig. 5. *(Continued.)* Segmentation results of 15 algorithms for IR small targets of different scales and types in complex backgrounds. (a) Point target (drone) in the complex ground background. (b) Point target (aircraft) in the complex dense cumulus background. (c) Point target (reentry capsule) in the background of the sky-ground line. (d) Two point targets (birds) in the sky background. (e) Two point targets (drones) in the sky, ground, and architectural background. (f) Regional target (drone) in the sky, clouds, ground, and jungle background. (g) Regional target (drone) in the clean sky background.

b) Accuracy and robustness analysis: It is important to note that the primary aim of this study was to enable embedded deployment by minimizing computational complexity while ensuring segmentation accuracy. In Table II, although the LW-IRSTNet algorithm ranks second in mIoU and third in $F1$, the reduction in mIoU is only 1.32% and the reduction in $F1$ is merely 0.81% compared to the algorithm with the best accuracy, MTUNet. Meanwhile, the FLOPs and Params of LW-IRSTNet are significantly reduced by 95.12% and 98.01%, respectively, compared to MTUNet.

Furthermore, MTUNet, which achieves the highest accuracy, is based on the transformer framework, which is not particularly suitable for embedded deployments at this stage. Similarly, although the top three AGPCNet algorithms in terms of accuracy utilize the CNN framework, their computational complexity is much higher than that of LW-IRSTNet, which is not favorable for embedded deployment.

c) Comprehensive analysis: To provide a visualization of the relationship between accuracy and computational complexity among the different algorithms, we have plotted a scatter diagram in Fig. 4. It is evident that our proposed LW-IRSTNet algorithm effectively balances the segmentation accuracy and computational complexity, fulfilling the initial objective of the study.

2) Qualitative Comparative Analysis: To visually compare the segmentation effects of different algorithms, we carefully

selected seven representative single-frame IR small target images with diverse scenes, target scales, and target types for testing, as shown in Fig. 5. Note: In Fig. 5, the red, green, and blue boxes represent correct target detection, missed detection, and false alarms, respectively.

In Fig. 5(a), a long-range point drone is almost submerged in the jungle background. Only the LW-IRSTNet and MTUNet algorithms successfully detect the target without any false alarms or missed detections. In Fig. 5(b), an aircraft is barely visible due to its weak local contrast and long distance. Only the LW-IRSTNet, MTUNet, and DNANet algorithms manage to detect it amidst the dense cumulus background. In contrast, the other 12 algorithms exhibit missed detections. Fig. 5(c) shows the detection of a return capsule about to land, extracted from a public video. While all 15 algorithms successfully detect the return capsule, six of them mistakenly identify the parachute as a target. However, the LW-IRSTNet algorithm adeptly detects the return capsule without any false alarms or missed detections. In Fig. 5(d) and (e), there are two small birds and two drones flying in the sky, respectively, with weak local contrasts. Among the 15 algorithms, only LW-IRSTNet, MTUNet, DNANet, ENet, and Fusionnet accurately detect multiple targets without false alarms and missed detections. In Fig. 5(f), although the drone edge contour is clearer and it has stronger local contrast, it is closer to the highlighted clouds, thus giving false alarms to the nine algorithms. Conversely, the LW-IRSTNet algorithm not only captures the drone's shape accurately but also avoids any false alarms or missed detections. Finally, in Fig. 5(g), we observe a large and well-defined drone against a relatively clean background. Although all 15 algorithms correctly segment the drone, the LW-IRSTNet algorithm exhibits superior accuracy in terms of pixel-level segmentation.

In summary, the LW-IRSTNet has good segmentation results for IR small targets at different scales with different complex backgrounds, while other algorithms all have different degrees of false alarms or missed detections.

E. Ablation Experiment

1) DAAA Modules: To demonstrate the efficacy of the DAAA module, all convolution modes in the DAAA module are successively changed to regular, atrous, depthwise separable convolution or remove DAAA 3 and 4. The results of these experiments are summarized in Table III. The experimental results show that making any changes in the DAAA module will reduce the mIoU and $F1$.

2) Channel Expansion or Compression Ratio in the DAAA Modules: In this experiment, we systematically altered the expansion ratio of the depthwise separable convolution channel and the compression ratio of the atrous/asymmetric convolution channel within the DAAA module, as shown in Table IV. The experimental results show that the expansion or compression channel rate has a small impact on the results of mIoU and $F1$, but a large impact on FLOPs and Params. To strike a balance between performance and computational complexity, we ultimately selected the design scheme of (2, 4).

3) Atrous Ratio Setting in the DAAA Modules: The atrous ratio is changed successively, as shown in Table V.

TABLE III
CONVOLUTION METHODS OF DAAA MODULES

Convolution methods	FLOPs (M)	Params (M)	Fusion datasets	
			mIoU	F1
Reguler Conv.	239.89	0.10	0.6187	0.7645
At-Conv.			0.6503	0.7881
Dw- Conv.	472.15	0.32	0.6303	0.7732
As-Conv.	262.18	0.12	0.6485	0.7868
Remove DAAA 3 and 4	231.60	0.10	0.6153	0.7618
Ours (DAAA 1-4)	303.53	0.16	**0.6638**	**0.7979**

TABLE IV
CHANNEL EXPANSION OR COMPRESSION RATIO

Expansion rate (Dw-Conv.)	Compression ratio (At-Conv. As-Conv.)	FLOPs (M)	Params (M)	Fusion datasets	
				mIoU	F1
2	2	491.09	0.30	0.6646	0.7985
4	4	380.08	0.24	0.6588	0.7896
2	4	**303.53**	**0.16**	0.6638	0.7979

TABLE V
ATROUS RATIO

Atrous ratio			FLOPs (M)	Params (M)	Fusion datasets	
DAAA1	DAAA2	DAAA 3-4			mIoU	F1
(2,2)	(2,2)	Repeat DAAA 1-2	303.53	0.16	0.6396	0.7802
(2,2)	(2,4)				0.6443	0.7837
(2,2)	(4,8)				0.6541	0.7909
(2,4)	(8,16)				**0.6638**	**0.7979**
(4,8)	(16,16)				0.6344	0.7763
(8,16)	(16,16)				0.6310	0.7738

We observed that changing the atrous rate did not affect the FLOPs and Params of LW-IRSTNet. However, the segmentation accuracy, as measured by mIoU and $F1$ scores, consistently yielded the highest results when the atrous ratio was set as (2, 4, 8, 16).

4) Kernel Sizes of Asymmetric Convolution in the DAAA Modules: The kernel size of asymmetric convolution in the DAAA module is changed sequentially, as shown in Table VI. Notably, we found that setting the kernel size to 7 resulted in relatively higher mIoU and $F1$ scores compared to other kernel sizes.

5) Number of Channels in Different Stages: The number of channels in different stages is changed in turn, as shown in Table VII. While the combination of (8, 16, 32) significantly reduced FLOPs and Params, we also observed a decline of 2%–3% points in mIoU and $F1$. Conversely, the combination of (16, 64, 128) substantially increased FLOPs and Params by approximately 3–4 times. Ultimately, we determined that a combination of (8, 32, 64) struck a balance between performance and computational complexity.

6) Upsampling Methods: In Table VIII, we discovered that using bilinear interpolation in the upsampling stage resulted in reduced model parameters and FLOPs. However, there was a tradeoff of 3%–4% loss in mIoU and $F1$ scores. Considering

TABLE VI
KERNEL SIZES OF ASYMMETRIC CONVOLUTION

Kernel sizes of As-Conv. in the DAAA 1- 4				FLOPs (M)	Params (M)	Fusion datasets	
						mIoU	F1
3	3	3	3	295.14	0.15	0.6440	0.7835
3	5	3	5	297.24	0.16	0.6174	0.7635
5	5	5	5	299.34	0.16	0.6526	0.7898
5	7	5	7	301.43	0.16	0.6527	0.7899
5	9	5	9	303.53	0.16	0.6330	0.7752
7	7	7	7	303.53	0.16	**0.6638**	**0.7979**
7	9	7	9	305.63	0.17	0.6400	0.7805
9	9	9	9	307.72	0.17	0.6603	0.7954
3	5	7	9	301.43	0.16	0.6544	0.7911

TABLE VII
NUMBER OF CHANNELS IN DIFFERENT STAGES

Number of channels			FLOPs (M)	Params (M)	Fusion datasets	
Stage1	Stage2	Stage3			mIoU	F1
8	16	32	99.20	0.04	0.6085	0.7566
16	64	128	1147.91	0.63	0.6612	0.7961
8	32	64	303.53	0.16	**0.6638**	**0.7979**

TABLE VIII
UPSAMPLING METHODS

Upsampling	FLOPs (M)	Params (M)	Fusion datasets	
			mIoU	F1
Bilinear	198.28	0.14	0.6067	0.7552
Bilinear (Stage4、5) Deconvolution (Stage6)	202.87	0.14	0.6223	0.7672
Deconvolution	303.53	0.16	**0.6638**	**0.7979**

this balance, we opted for deconvolution as the upsampling method for LW-IRSTNet.

7) Feature Fusion Methods: Given that existing IRSTNets typically employ complex feature fusion methods [20], [23], [26], [28], [63], [64], [65], we conducted ablation experiments to ascertain if incorporating these methods into LW-IRSTNet would improve segmentation accuracy. Table IX presents the results of these experiments, revealing that our designed LW-IRSTNet algorithm achieves the highest segmentation accuracy while maintaining smaller parameter size and FLOPs, even without relying on complex feature fusion methods.

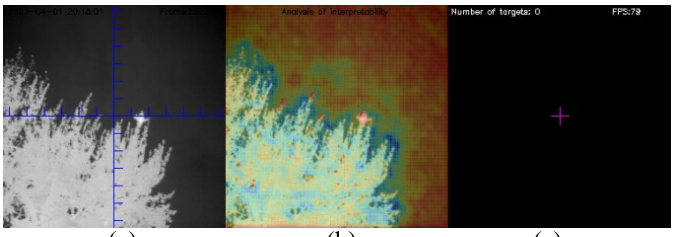
In summary, the comprehensive comparative experiments and ablation analysis provide substantial evidence that the LW-IRSTNet algorithm successfully balances lightweight design considerations with high segmentation accuracy.

VI. DISCUSSION

Although the LW-IRSTNet algorithm demonstrates robustness in segmenting small multiscale IR targets in complex backgrounds, it may still encounter challenges in certain scenarios. For instance, if a drone is completely submerged in a wooded background, it becomes difficult to identify it, as shown in Fig. 6. However, the corresponding thermal map in Fig. 6(b) reveals that the drone's thermal value is significantly higher than the background, even though it does not exceed the inference threshold. Therefore, future research will focus on addressing the segmentation of targets in cases involving background occlusion.

TABLE IX
FEATURE FUSION METHODS

Fusion methods	FLOPs (M)	Params (M)	Fusion datasets	
			mIoU	F1
LW-IRSTNet + ASPP [63]	677.45	0.20	0.5947	0.7519
LW-IRSTNet + CPM [23]	620.90	0.17	0.6326	0.7855
LW-IRSTNet + AFM [23]	724.50	0.18	0.5483	0.7406
LW-IRSTNet + ACM [20]	724.50	0.18	0.6003	0.7742
LW-IRSTNet + BiLocal [20]	731.84	0.18	0.6398	0.7889
LW-IRSTNet + BiGlobal [20]	717.16	0.18	0.5956	0.7844
LW-IRSTNet + CPM + AFM	738.34	0.18	0.5869	0.7709
Concat	315.06	0.17	0.6087	0.7568
No fusion	303.53	0.16	0.5946	0.7313
Sum			**0.6638**	**0.7979**



(a) (b) (c)

Fig. 6. (a) Original image containing a small drone, which is submerged in the woods. (b) Heat map after inference by LW-IRSTNet algorithm. (c) Real-time segmentation results.

VII. CONCLUSION

In conclusion, considering the initial research objective, we aimed to design an IRSTNet that effectively reduces computational complexity while maintaining high segmentation accuracy. The experimental results confirm that LW-IRSTNet successfully meets these design requirements. Moving forward, our research will focus on two key areas based on LW-IRSTNet: 1) exploring methods to enhance the detection capability of IR small targets under complex ground jungle background occlusion and 2) investigating real-time IR small target tracking algorithms.

REFERENCES

[1] R. Kou, H. Wang, Z. Zhao, and F. Wang, "Optimum selection of detection point and threshold noise ratio of airborne infrared search and track systems," *Appl. Opt.*, vol. 56, no. 18, p. 5268, Jun. 2017, doi: 10.1364/AO.56.005268.

[2] M. Zhao, W. Li, L. Li, J. Hu, P. Ma, and R. Tao, "Single-frame infrared small-target detection: A survey," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 2, pp. 87–119, Jun. 2022, doi: 10.1109/MGRS.2022.3145502.

[3] H. Ding and H. Zhao, "Adaptive method for the detection of infrared small target," *Opt. Eng.*, vol. 54, no. 11, Nov. 2015, Art. no. 113107, doi: 10.1117/1.OE.54.11.113107.

[4] P.-Y. Lv, S.-L. Sun, C.-Q. Lin, and G.-R. Liu, "Space moving target detection and tracking method in complex background," *Infr. Phys. Technol.*, vol. 91, pp. 107–118, Jun. 2018, doi: 10.1016/j.infrared.2018.03.007.

[5] X. Bai and F. Zhou, "Analysis of new top-hat transformation and the application for infrared dim small target detection," *Pattern Recognit.*, vol. 43, no. 6, pp. 2145–2156, Jun. 2010, doi: 10.1016/j.patcog.2009.12.023.

[6] L. Deng, J. Zhang, G. Xu, and H. Zhu, "Infrared small target detection via adaptive M-estimator ring top-hat transformation," *Pattern Recognit.*, vol. 112, Apr. 2021, Art. no. 107729, doi: 10.1016/j.patcog.2020.107729.

[7] Y. Chen, B. Song, D. Wang, and L. Guo, "An effective infrared small target detection method based on the human visual attention," *Infr. Phys. Technol.*, vol. 95, pp. 128–135, Dec. 2018, doi: 10.1016/j.infrared.2018.10.033.

[8] P. Du and A. Hamdulla, "Infrared small target detection using homogeneity-weighted local contrast measure," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 3, pp. 514–518, Mar. 2020, doi: 10.1109/LGRS.2019.2922347.

[9] J. Han, C. Liu, Y. Liu, Z. Luo, X. Zhang, and Q. Niu, "Infrared small target detection utilizing the enhanced closest-mean background estimation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 645–662, 2021, doi: 10.1109/JSTARS.2020.3038442.

[10] R. Kou, C. Wang, Q. Fu, Y. Yu, and D. Zhang, "Infrared small target detection based on the improved density peak global search and human visual local contrast mechanism," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 6144–6157, 2022, doi: 10.1109/JSTARS.2022.3193884.

[11] H. Zhang, L. Zhang, D. Yuan, and H. Chen, "Infrared small target detection based on local intensity and gradient properties," *Infr. Phys. Technol.*, vol. 89, pp. 88–96, Mar. 2018, doi: 10.1016/j.infrared.2017.12.018.

[12] Y. Bi, J. Chen, H. Sun, and X. Bai, "Fast detection of distant, infrared targets in a single image using multiorder directional derivatives," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 3, pp. 2422–2436, Jun. 2020, doi: 10.1109/TAES.2019.2946678.

[13] X. Kong, L. Liu, Y. Qian, and M. Cui, "Automatic detection of sea-sky horizon line and small targets in maritime infrared imagery," *Infr. Phys. Technol.*, vol. 76, pp. 185–199, May 2016, doi: 10.1016/j.infrared.2016.01.016.

[14] Y. Shi, Y. Wei, H. Yao, D. Pan, and G. Xiao, "High-boost-based multiscale local contrast measure for infrared small target detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 1, pp. 33–37, Jan. 2018, doi: 10.1109/LGRS.2017.2772030.

[15] L. Zhang and Z. Peng, "Infrared small target detection based on partial sum of the tensor nuclear norm," *Remote Sens.*, vol. 11, no. 4, p. 382, Feb. 2019, doi: 10.3390/rs11040382.

[16] F. Zhou, Y. Wu, Y. Dai, and P. Wang, "Detection of small target using Schatten 1/2 quasi-norm regularization with reweighted sparse enhancement in complex infrared scenes," *Remote Sens.*, vol. 11, no. 17, p. 2058, Sep. 2019, doi: 10.3390/rs11172058.

[17] H. Zhu, H. Ni, S. Liu, G. Xu, and L. Deng, "TNLRS: Target-aware non-local low-rank modeling with saliency filtering regularization for infrared small target detection," *IEEE Trans. Image Process.*, vol. 29, pp. 9546–9558, 2020, doi: 10.1109/TIP.2020.3028457.

[18] H. Wang, L. Zhou, and L. Wang, "Miss detection vs. false alarm: Adversarial learning for small object segmentation in infrared images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8508–8517, doi: 10.1109/ICCV.2019.00860.

[19] R. Kou et al., "Infrared small target segmentation networks: A survey," *Pattern Recognit.*, vol. 143, Nov. 2023, Art. no. 109788, doi: 10.1016/j.patcog.2023.109788.

[20] Y. Dai, Y. Wu, F. Zhou, and K. Barnard, "Asymmetric contextual modulation for infrared small target detection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 949–958, doi: 10.1109/WACV48630.2021.00099.

[21] 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. 936–944, doi: 10.1109/CVPR.2017.106.

[22] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent. (MICCAI)*, 2015, pp. 234–241, doi: 10.1007/978-3-319-24574-4_28.

[23] T. Zhang, L. Li, S. Cao, T. Pu, and Z. Peng, "Attention-guided pyramid context networks for detecting infrared small target under complex background," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 4, pp. 4250–4261, Aug. 2023, doi: 10.1109/TAES.2023.3238703.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[25] B. Li et al., "Dense nested attention network for infrared small target detection," *IEEE Trans. Image Process.*, vol. 32, pp. 1745–1758, 2023, doi: 10.1109/TIP.2022.3199107.

[26] L. Huang, S. Dai, T. Huang, X. Huang, and H. Wang, "Infrared small target segmentation with multiscale feature representation," *Infr. Phys. Technol.*, vol. 116, Aug. 2021, Art. no. 103755, doi: 10.1016/j.infrared.2021.103755.

[27] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[28] Z. Zuo et al., "AFFPN: Attention fusion feature pyramid network for small infrared target detection," *Remote Sens.*, vol. 14, no. 14, p. 3412, Jul. 2022, doi: 10.3390/rs14143412.

[29] K. Wang, S. Du, C. Liu, and Z. Cao, "Interior attention-aware network for infrared small target detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5002013, doi: 10.1109/TGRS.2022.3163410.

[30] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017, doi: 10.1109/TPAMI.2016.2572683.

[31] A. Dosovitskiy et al., "An image is worth 16 × 16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[32] B. Zhao, C. Wang, Q. Fu, and Z. Han, "A novel pattern for infrared small target detection with generative adversarial network," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4481–4492, May 2021, doi: 10.1109/TGRS.2020.3012981.

[33] F. Liu, C. Gao, F. Chen, D. Meng, W. Zuo, and X. Gao, "Infrared small-dim target detection with transformer under complex backgrounds," 2021, *arXiv:2109.14379*.

[34] T. Wu et al., "MTU-Net: Multilevel TransUnet for space-based infrared tiny ship detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5601015, doi: 10.1109/TGRS.2023.3235002.

[35] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1800–1807, doi: 10.1109/CVPR.2017.195.

[36] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size," 2016, *arXiv:1602.07360*.

[37] A. G. Howard et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[38] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520, doi: 10.1109/CVPR.2018.00474.

[39] A. Howard et al., "Searching for MobileNetV3," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1314–1324, doi: 10.1109/ICCV.2019.00140.

[40] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6848–6856, doi: 10.1109/CVPR.2018.00716.

[41] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "ShuffleNet V2: Practical guidelines for efficient CNN architecture design," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 122–138, doi: 10.1007/978-3-030-01264-9_8.

[42] M. Tan and Q. V. Le, "MixConv: Mixed depthwise convolutional kernels," 2019, *arXiv:1907.09595*.

[43] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "GhostNet: More features from cheap operations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1577–1586, doi: 10.1109/CVPR42600.2020.00165.

[44] M. Fan et al., "Rethinking BiSeNet for real-time semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9711–9720, doi: 10.1109/CVPR46437.2021.00959.

[45] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2014, *arXiv.1503.02531*.

[46] X. Yang, J. Ye, and X. Wang, "Factorizing knowledge in neural networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2022, pp. 73–91, doi: 10.1007/978-3-031-19830-4_5.

[47] S. Liu, K. Wang, X. Yang, J. Ye, and X. Wang, "Dataset distillation via factorization," 2022, *arXiv:2210.16774*.

[48] X. Yang, D. Zhou, J. Feng, and X. Wang, "Diffusion probabilistic model made slim," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 22552–22562.

[49] Z. Liu, J. Li, Z. Shen, G. Huang, S. Yan, and C. Zhang, "Learning efficient convolutional networks through network slimming," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2755–2763, doi: 10.1109/ICCV.2017.298.

[50] R. Yu et al., "NISP: Pruning networks using neuron importance score propagation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9194–9203, doi: 10.1109/CVPR.2018.00958.

[51] J.-H. Luo, J. Wu, and W. Lin, "ThiNet: A filter level pruning method for deep neural network compression," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5068–5076, doi: 10.1109/ICCV.2017.541.

[52] F. Tung and G. Mori, "CLIP-Q: Deep network compression learning by in-parallel pruning-quantization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7873–7882, doi: 10.1109/CVPR.2018.00821.

[53] C. Leng, Z. Dou, H. Li, S. Zhu, and R. Jin, "Extremely low bit neural network: Squeeze the last bit out with ADMM," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2018, vol. 32, no. 1, pp. 3466–3473, doi: 10.1609/aaai.v32i1.11713.

[54] Y. Jing, Y. Yang, X. Wang, M. Song, and D. Tao, "Meta-aggregator: Learning to aggregate for 1-bit graph neural networks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 5281–5290, doi: 10.1109/ICCV48922.2021.00525.

[55] P. Wang, Q. Hu, Y. Zhang, C. Zhang, Y. Liu, and J. Cheng, "Two-step quantization for low-bit neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4376–4384, doi: 10.1109/CVPR.2018.00460.

[56] X. Zhang, J. Zou, K. He, and J. Sun, "Accelerating very deep convolutional networks for classification and detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 1943–1955, Oct. 2016, doi: 10.1109/TPAMI.2015.2502579.

[57] Y.-D. Kim, E. Park, S. Yoo, T. Choi, L. Yang, and D. Shin, "Compression of deep convolutional neural networks for fast and low power mobile applications," 2015, *arXiv:1511.06530*.

[58] J.-D. Dong, A.-C. Cheng, D.-C. Juan, W. Wei, and M. Sun, "DPP-Net: Device-aware progressive search for pareto-optimal neural architectures," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 540–555, doi: 10.1007/978-3-030-01252-6_32.

[59] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8697–8710, doi: 10.1109/CVPR.2018.00907.

[60] M. Tan et al., "MnasNet: Platform-aware neural architecture search for mobile," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2815–2823, doi: 10.1109/CVPR.2019.00293.

[61] C. Liu et al., "Progressive neural architecture search," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 1–16, doi: 10.1007/978-3-030-01246-5_2.

[62] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," 2015, *arXiv:1512.00567*.

[63] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder–decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 833–851, doi: 10.1007/978-3-030-01234-2_49.

[64] C. Yu et al., "Pay attention to local contrast learning networks for infrared small target detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022, doi: 10.1109/LGRS.2022.3178984.

[65] X. Tong, B. Sun, J. Wei, Z. Zuo, and S. Su, "EAAU-Net: Enhanced asymmetric attention U-net for infrared small target detection," *Remote Sens.*, vol. 13, no. 16, p. 3200, Aug. 2021, doi: 10.3390/rs13163200.

[66] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A deep neural network architecture for real-time semantic segmentation," 2016, *arXiv:1606.02147*.

[67] R. Kou, C. Wang, Y. Yu, Z. Peng, F. Huang, and Q. Fu, "Infrared small target tracking algorithm via segmentation network and multi-strategy fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5612912, doi: 10.1109/TGRS.2023.3286836.

[68] T. M. Quan, D. G. C. Hildebrand, and W.-K. Jeong, "FusionNet: A deep fully residual convolutional neural network for image segmentation in connectomics," *Frontiers Comput. Sci.*, vol. 3, May 2021, Art. no. 613981, doi: 10.3389/fcomp.2021.613981.

[69] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "BiSeNet: Bilateral segmentation network for real-time semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 1–17, doi: 10.1007/978-3-030-01261-8_20.

[70] G. Li, I. Yun, J. Kim, and J. Kim, "DABNet: Depth-wise asymmetric bottleneck for real-time semantic segmentation," 2019, *arXiv:1907.11357*.

[71] H. Li, P. Xiong, H. Fan, and J. Sun, "DFANet: Deep feature aggregation for real-time semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9514–9523, doi: 10.1109/CVPR.2019.00975.

[72] T. Wu, S. Tang, R. Zhang, and Y. Zhang, "CGNet: A light-weight context guided network for semantic segmentation," 2018, *arXiv:1811.08201*.

[73] P. Pan, H. Wang, C. Wang, and C. Nie, "ABC: Attention with bilinear correlation for infrared small target detection," 2023, *arXiv:2303.10321*.

Renke Kou received the B.E. degree from the Changchun University of Technology, Changchun, China, in 2015, and the M.E. degree from Air Force Engineering University, Xi'an, China, in 2017. He is currently pursuing the Ph.D. degree with Army Engineering University, Shijiazhuang, China.

His research interests include image process and pattern recognition.

Mr. Kou is a reviewer of the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING.

Chunping Wang received the Ph.D. degree from the Shijiazhuang Mechanical Engineering College, Shijiazhuang, Hebei, China, in 1999.

He is currently a Professor with Army Engineering University, Shijiazhaung. He has coauthored 200 articles. He obtained four national defense invention patents. His research interests include control theory and application, and information processing.

Dr. Wang is the Winner of the Military Science and Technology Progress Award. At the same time, he is an Editorial Board Member of the journal of *Fire Control and Command Control* and *Aerodynamic Missile*.

Ying Yu received the B.E. degree from Hebei Normal University, Shijiazhuang, China, in 2011, and the M.S. degree from Yunnan Normal University, Kunming, China, in 2014. She is currently pursuing the Ph.D. degree with Army Engineering University of PLA, Shijiazhuang.

From 2014 to 2022, she was a Faculty at the School of Information and Intelligence Engineering, University of Sanya, Sanya, China, where she has been an Associate Professor since 2021. Her research interests include computer vision and pattern recognition.

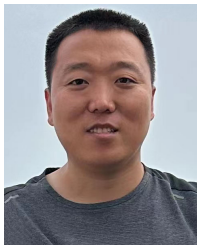
Zhenming Peng (Member, IEEE) received the Ph.D. degree in geodetection and information technology from the Chengdu University of Technology, Chengdu, China, in 2001.

From 2001 to 2003, he was a Post-Doctoral Researcher with the Institute of Optics and Electronics, Chinese Academy of Sciences, Beijing, China. He is currently a Professor with the University of Electronic Science and Technology of China, Chengdu. His research interests include image processing, signal processing, and target recognition and tracking.

Prof. Peng is a member of the Optical Society of America (OSA), the China Optical Engineering Society (COES), and the Chinese Society of Astronautics (CSA).

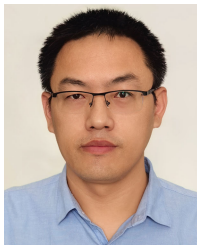
Mingbo Yang received the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2013.

From 2013 to 2015, he was a Post-Doctoral Fellow of control theory and control engineering at Tsinghua University, Beijing. Since 2015, he has been a Lecturer at the School of Mechanical and Materials Engineering, North China University of Technology, Beijing. His research interests include artificial intelligence, wireless energy transmission, unified power quality conditioner (UPQC), and embedded system development.



Fuyu Huang received the B.M. degrees from the Department of Optical and Electronic Engineering, Mechanical Engineering College, Shijiazhuang, China, in 2008, 2010, and 2014, respectively.

He currently presides over two National Natural Science Foundation projects and one Natural Science Foundation Project of Hebei Province. His research interests include optical signal processing, infrared detection, and optical design.



Qiang Fu received the B.S. degree in electrical engineering from the Shijiazhuang Mechanical Engineering College, Shijiazhuang, Hebei, China, in 1999, and the Ph.D. degree in computer science and technology from Tsinghua University, Beijing, China, in 2017.

After the Ph.D. degree, he has been a Lecturer at Army Engineering University, Shijiazhuang. He has authored over 50 technical articles. His teaching and research interests include automatic control and image engineering.