Advances in Computer, Signals and Systems (2025)
Clausius Scientific Press, Canada

DOI: 10.23977/acss.2025.090115
ISSN 2371-8838 Vol. 9 Num. 1

High-Precision and Fast Inference for Infrared Small Target Detection through Semantic Gap Reduction

Shen Deng[1,2,a], Yang Yang[1,2,b,*]

[1]*School of Information Science and Technology, Yunnan Normal University, Kunming, China*
[2]*Laboratory of Pattern Recognition and Artificial Intelligence, Yunnan Normal University, Kunming, China*
[a]*dengshen_0423@163.com*, [b]*yyang_ynu@163.com*
**Corresponding author*

Keywords: Infrared Small Target Detection, Deep Learning, Semantic Gap

Abstract: In recent years, significant advancements have been made in the field of infrared small target detection (IRSTD), largely driven by developments in deep learning and computer vision. Deep learning-based methods have demonstrated substantial improvements in both accuracy and inference speed compared to traditional approaches, enabling their integration into real-time embedded systems. However, many data-driven techniques rely on complex network architectures to process large volumes of intricate data, resulting in additional computational overhead. To enhance the efficiency of IRSTD, we propose an improvement based on the classical segmentation framework, introducing a semantic gap elimination module (SGEM) to reduce the level-to-level semantic gap. This enhancement improves the stability and performance of IRSTD. Notably, our method does not rely on complex network architectures, allowing it to outperform other deep learning-based methods in terms of computational efficiency. It also exceeds the performance of the fastest methods, achieving more than a threefold increase in the frames per second (FPS). Furthermore, comparative experiments demonstrate the effectiveness of our approach, showing superior performance over recent methods in both segmentation and localization accuracy.

1. Introduction

Infrared small target detection (IRSTD) is a fundamental research area in computer vision with broad applications in various fields, such as detecting fighter jets and ships in military contexts or identifying intruders in critical areas. In the era before the explosion of convolutional neural networks (CNNs) and deep learning, researchers typically applied image decomposition and matrix theory to IRSTD tasks, such as local contrast filtering and matrix rank decomposition. In recent years, deep learning methods have achieved breakthrough progress across various computer vision domains. In IRSTD, Dai et al. [1] redefined IRSTD as a segmentation task, which mitigates the issue of incomplete convergence of the anchor box during model training caused by the small size of the target in the data.

Building on this foundation, many researchers have continued to innovate and proposed several

algorithms based on the segmentation framework. These methods significantly outperform traditional algorithms in both accuracy and inference speed. However, they still struggle to detect extremely small targets in complex scenes. Li et al. [2] identified the issue of semantic gaps between layers in segmentation frameworks and designed a dense nested structure to enhance feature interactions between layers, attempting to reduce the semantic gaps. However, this approach imposes a substantial computational burden and often requires more time for inference.

To address this challenge, we have improved the classic U-Net segmentation network by designing a semantic gap elimination module (SGEM) to reduce semantic gaps between layers. Our method does not require adding considerable network layers to compute semantic correlations between layers, as in dense nested structures. Instead, it directly calculates the semantic correlation between intermediate and deep layers, thereby eliminating the largest semantic gap in deep networks. Compared to existing deep learning methods, our approach not only achieves superior segmentation and localization accuracy but also offers significantly faster inference speeds, boasting 3 times the frames per second (FPS) compared to the fastest current deep learning methods.

2. Related Work

2.1. Segmentation Framework

The segmentation framework for IRSTD was first proposed by Dai et al [1]. They also attempted to integrate local contrast operators from traditional algorithms as prior knowledge for neural networks to assist in model training [3]. Kou et al. [4] combined detection and segmentation networks, employing coarse detection as a prior for the segmentation network within a multi-task framework, which further enhanced the localization capability of the segmentation network. Wu et al. [5] improved the classic segmentation network, U-Net, by embedding a U-Net structure in each network layer, strengthening the model's ability to retain small targets. Zhang et al. [6] and Li et al. [7] both introduced edge supervision methods. By designing additional network modules to process the target edges obtained from label images using the Sobel operator, they assisted in the training of the segmentation network. These models are highly sensitive to targets with significant edge features but still face challenges when dealing with extremely small targets with blurred edges.

2.2. Semantic Gap

Li et al. [2] argued that the semantic gap between layers is the primary reason for suboptimal segmentation performance in IRSTD tasks. They added numerous network layers to the classic U-Net architecture and continuously guided the recovery of previous layer features with deeper features throughout the decoding process, thereby constructing a densely nested structure within the network framework. This structure achieved groundbreaking progress in segmentation accuracy. However, due to the large number of network layers requiring additional computation, its spatial and temporal consumption is more than twice that of other models, which poses a significant burden for certain IRSTD applications, such as military and real-time monitoring. Later, other researchers continued to improve the dense nested framework, achieving some optimization in terms of accuracy and computational efficiency, but still could not avoid the inherent flaws of this structure.

3. Method

3.1. Segmentation Framework

We make improvements to the classic U-Net segmentation network, and the framework of our

model is shown in Figure 1. As can be seen, our model follows an encoder-decoder architecture, where the encoder extracts feature of the original image at different scales, and the decoder progressively fuses these features to recover the spatial resolution. Notably, we have eliminated the skip connection between the third layer of the encoder and the second layer of the decoder, replacing it with the SGEM. Specifically, the features from the third layer of the encoder and the first layer of the decoder are both fed into the SGEM. By calculating the linear correlation between these features, the module produces a semantic probability distribution matrix. This output is then employed as the result for the second layer of the decoder, contributing to the subsequent feature fusion and decoding operations. This approach replaces the dense nested structure, which relies on considerable network layers to bridge the semantic gaps between layers, leading to improvements in both accuracy and inference speed compared to the dense nested structure. The computational process of the SGEM will be described later.

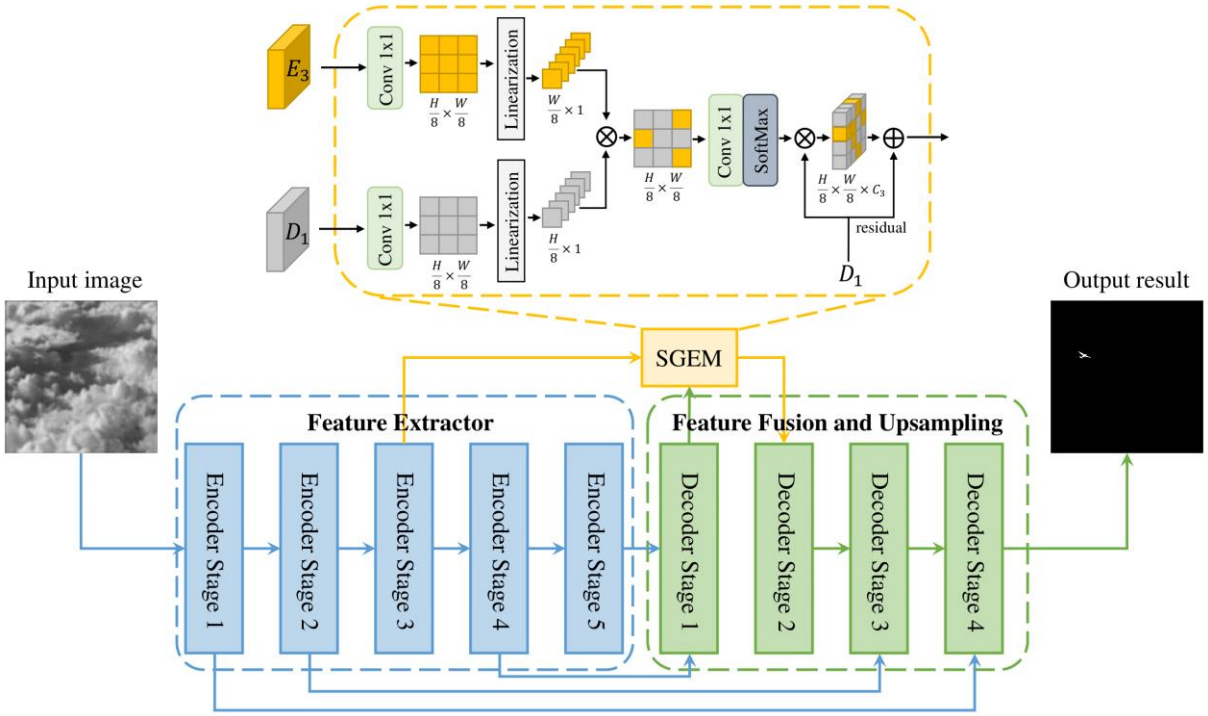

Figure 1: The overall architecture of our proposed method.

3.2. Semantic Gap Eliminate Module

In this section, we provide a detailed description of the implementation process of the SGEM, as shown in Figure 1. We simultaneously input the output of the third encoder layer and the output of the first decoder layer into the SGEM to compute the semantic probability distribution matrix. Specifically, the input features are first transformed into linear vectors through different initial convolutional and linear layers. These vectors are then processed through matrix multiplication and per-channel convolutions to restore the original three-dimensional features. Finally, the probability distribution is normalized to the range of 0 to 1 using a softmax activation function. Residual connections are employed to ensure the effective transmission of deep features, which are then output to the second layer of the decoder. The detailed implementation of the semantic probability distribution matrix are as follows:

$$Q = Linearization(Conv_{1\times1}(E_3)) \tag{1}$$

$$K = Linearization(Conv_{1\times1}(D_1)) \tag{2}$$

$$Matrix = Softmax(Conv_{1\times1}(Q \otimes K)) \tag{3}$$

Our design is inspired by linear cross-attention, a method that adapts the kernel self-attention mechanism from Transformers to compute correlations between different network layers. In this approach, input features from various sources are modeled as Q, K, and V, and linearized attention operations are performed. The linearization process serves two main purposes: it reduces the dimensionality of matrix operations, thereby decreasing computational complexity, and it eliminates redundant information through model training, ultimately enhancing the efficiency of IRSTD.

3.3. Loss Function

We employ the SoftIoU loss function to train our IRSTD network model. The SoftIoU loss function is particularly effective in handling class-imbalanced datasets, as it provides smoother gradient updates, which help mitigate oscillations during the training process. This makes it well-suited for datasets with imbalanced categories, ensuring more stable and effective training.

In segmentation models, predictions are typically represented as binary masks. To evaluate the similarity between the predicted and ground truth masks, we compute the intersection and union of the two masks, followed by the calculation of the intersection over union (IoU) ratio. The SoftIoU loss function first scales the model's predictions using a sigmoid function, mapping the outputs to a range between 0 and 1. This scaling enables the use of derivatives in the loss calculation. It then computes the intersection and union sets of the predicted and true masks, ultimately calculating the IoU ratio. The formal definitions are as follows:

$$loss = -\frac{1}{|C|}\sum_{C}\frac{\sum_{pixels} y_{true}y_{pred}}{\sum_{pixels}\left(y_{true} + y_{pred} - y_{true}y_{pred}\right)} \tag{4}$$

Where C represents the total number of categories, y_{true} denotes the true target pixel points, and y_{pred} represents the predicted pixel points.

4. Experiment

4.1. Datasets

We select the NUDT-SIRST dataset as the experimental dataset. This dataset, developed by Li et al. [2], is a synthetic collection consisting of 1,327 images with a resolution of 256x256 pixels. The backgrounds include various scenes such as cities, fields, oceans, and skies, while the targets primarily consist of small objects such as drones, airplanes, and ships. For model training, we adopt the original data split proposed by the authors, with the dataset divided into training and testing sets in a 1:1 ratio, resulting in 663 training images and 664 testing images.

4.2. Implement details

All traditional methods are implemented on MATLAB R2023b, all data-driven methods are implemented by PyTorch on a computer equipped with an Intel(R) Core(TM) i7-11700K @ 3.60GHz CPU and an NVIDIA GeForce RTX 3090 GPU. The images input into models are randomly cropped to 256 × 256. The training epoch, batch size and learning rate are set to 1500, 8 and 0.0001.

4.3. Evaluation Metrics

We use Intersection over Union (IoU) to evaluate the model's ability to segment target boundaries, F1 score to provide a comprehensive assessment of the model's precision and recall, floating point operations (FLOPs) to evaluate the computational complexity of the algorithm, Params to assess the model size, and FPS to measure the inference speed. These metrics offer a thorough evaluation of each model's overall performance in terms of accuracy and computational efficiency.

4.4. Quantitative Results

Table 1: Comparison to the data-driven methods in terms of IoU ($\times 10^2$), F1($\times 10^2$), FLOPs(G), Params(M) and FPS.

	IoU	F1	FLOPs(G)	Params(M)	FPS
ACMNet[1]	60.75	75.58	1.31	1.54	103
ALCNet[3]	81.60	89.87	4.34	0.37	20
IAANet[4]	84.03	91.32	436.79	14.05	14
UIUNet[5]	85.83	92.37	54.50	50.54	48
ISNet[6]	66.77	80.08	30.63	1.09	70
DNANet[2]	86.57	92.80	14.28	4.70	27
ABCNet[8]	83.28	90.87	5.31	9.09	95
EGPNet[7]	71.52	83.39	19.55	3.54	92
Ours	87.64	93.41	15.92	8.69	342

As shown in Table 1, our model achieves the best results in both segmentation accuracy and localization precision. Although our approach does not demonstrate a significant advantage over other models in terms of algorithmic complexity and parameter count, it excels in inference speed. Specifically, our model is more than three times faster than the next fastest deep learning method, ACMNet.

Our approach employs computationally intensive linear relational models to address inter-layer semantic gaps, with minimal impact on inference speed. Additionally, we expand the initial number of channels in the model to enhance its learning and representational capabilities during the early encoding phase. This results in a larger parameter count compared to the lightweight model, ALCNet. However, in general, appropriately scaling the model is essential for achieving higher accuracy.

5. Conclusion

This paper improved upon the classic U-Net segmentation network by designing the SGEM to reduce the semantic gap between network layers. Unlike dense nested structures that require the addition of numerous layers to compute level-to-level semantic correlations, our model directly calculated the semantic correlations between intermediate and deeper layers, thereby eliminating the largest semantic gaps in the deeper layers of the network. Compared to existing deep learning methods, our model not only leaded in segmentation accuracy and localization precision but also boasted an exceptionally fast inference speed, achieving three times the FPS of the other fastest deep learning methods.

Although our approach demonstrates significant advantages in both inference speed and model accuracy compared to recent deep learning methods, challenges remain when confronted with complex scenes characterized by strong background clutter and very weak target signals. In such scenarios, our method may still experience issues such as miss detection and false alarm. This presents a significant challenge for applications requiring extremely high detection accuracy. Therefore, in the

future, we plan to further optimize the algorithm and develop datasets for algorithm validation to enable real-world deployment in practical applications.

References

[1] Dai Y, Wu Y, Zhou F, et al. Asymmetric contextual modulation for infrared small target detection[C]. Proceedings of the IEEE/CVF winter conference on applications of computer vision. 2021: 950-959.
[2] Li B, Xiao C, Wang L, et al. Dense nested attention network for infrared small target detection[J]. IEEE Transactions on Image Processing, 2022, 32: 1745-1758.
[3] Dai Y, Wu Y, Zhou F, et al. Attentional local contrast networks for infrared small target detection[J]. IEEE transactions on geoscience and remote sensing, 2021, 59(11): 9813-9824.
[4] Wang K, Du S, Liu C, et al. Interior attention-aware network for infrared small target detection[J]. IEEE Transactions on Geoscience and Remote Sensing, 2022, 60: 1-13.
[5] Wu X, Hong D, Chanussot J. UIU-Net: U-Net in U-Net for infrared small object detection[J]. IEEE Transactions on Image Processing, 2022, 32: 364-376.
[6] Zhang M, Zhang R, Yang Y, et al. ISNet: Shape matters for infrared small target detection[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022: 877-886.
[7] Li Q, Zhang M, Yang Z, et al. Edge-guided perceptual network for infrared small target detection[J]. IEEE Transactions on Geoscience and Remote Sensing, 2024.
[8] Pan P, Wang H, Wang C, et al. ABC: Attention with bilinear correlation for infrared small target detection[C]. 2023 IEEE International Conference on Multimedia and Expo (ICME). IEEE, 2023: 2381-2386.