IJMT. Vol. 22, Iss. 2, 2026. (1-11)

International Journal of Maritime Technology

Journal homepage: ijmt.ir

A Flexible Hybrid Attention Mechanism for Multi-Architecture Segmentation of Small Maritime Targets in South-East Region of Iran

Zobeir Raisi[1*], Esmaeil Sarani[2], Rasoul Damani[3], ValiMohammad Nazarzehi Had[4]

[1,2,3,4] *Assistant Professor, Faculty of Maine Engineering, Chabahar Maritime University, Chabahar, Iran;*
{zobeir.raisi, damani, sarani, v.nazarzehi}@cmu.ac.ir

ARTICLE INFO

Article History:
Received: 22 Sep 2025
Last modification: 19 Mar 2026
Accepted: 28 Mar 2026
Available online: 2 Apr 2026

Article type:
Research Paper

Keywords:
Satellite images,
ship detection and segmentation,
deep learning,
South East Iran

ABSTRACT

Maritime vessel detection in satellite imagery is essential for coastal monitoring, traffic regulation, and maritime security. Vessels along the southeastern coast of Iran exhibit unique structural and geometric characteristics; they are small and overlapped, differing substantially from international benchmarks. Detecting small and overlapping vessels presents additional challenges due to the loss of fine-grained features and ambiguous object boundaries in conventional deep learning pipelines. Consequently, existing pre-trained models, trained primarily on global datasets, often fail to generalize effectively to this region. To address this, in our study, we provide the first systematic investigation of ship detection for southeastern Iran, supported by a curated dataset of high-resolution satellite imagery from its major ports. We, then, propose a flexible Hybrid Attention Fusion (HAF) module that can be seamlessly integrated into multiple segmentation architectures, including FPN, Mask R-CNN, U-Net, and DeepLab. The module sequentially applies channel and spatial attention mechanisms to adaptively recalibrate multi-scale features, enhancing the representation of subtle and occluded instances. Experimental results demonstrate that HAF-augmented models significantly outperform their baseline counterparts across all architectures. For semantic segmentation, U-Net+HAF and DeepLabv3+HAF achieve mean IoU improvements of 4.5% and 4.4%, respectively, reaching 83.8% and 85.5% mIoU. For instance segmentation, Mask R-CNN+HAF demonstrates the most substantial improvement in small object detection, with Average Precision for small objects (APs) increasing from 42.3% to 50.6%—an 8.3-point improvement. Qualitative analysis confirms superior capability in detecting missed small instances, separating overlapping vessels, and producing more precise boundaries compared to baseline models.

ISSN: 2645-8136

DOI: http://dx.doi.org/10.66224/ijmt.22.2.1

Copyright: © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license [https://creativecommons.org/licenses/by/4.0/]

[Downloaded from ijmt.ir on 2026-11-08]

1. Introduction

High-resolution remote sensing imagery plays a crucial role in maritime monitoring, supporting tasks such as ship traffic analysis, fisheries management, security enforcement, and ecological preservation.

The automatic detection and segmentation of ships from satellite imagery is a critical task with numerous applications in maritime surveillance, traffic monitoring, fishery management, and national security [1], [2], [3], [4], [5], [6]. The advent of deep learning, particularly convolutional neural networks (CNNs), has led to significant breakthroughs in this domain. Architectures such as the Feature Pyramid Network FPN [7], Mask R-CNN [8], U-Net [9], and DeepLab [10] have become foundational frameworks for instance and semantic segmentation tasks.

Despite these advancements, the unique characteristics of satellite-based ship segmentation continue to pose formidable challenges [6]. Ships, when viewed from space, often represent small objects, sometimes encompassing only a few dozen pixels within a large image. Furthermore, they frequently appear in crowded ports and shipping lanes, leading to instances of severe occlusion and overlap. Standard segmentation architectures struggle with these scenarios because successive pooling and striding operations cause an irreversible loss of high-resolution spatial information, which is paramount for accurately delineating small object boundaries. Consequently, models often fail to separate individual ships in a cluster, resulting in merged segmentation and reduced recall for smaller vessels.

The southern coastline of Iran near Chabahar (See Figure 1), where numerous ports are situated along the Gulf of Oman, illustrates these challenges. In contrast to major international shipping lanes dominated by large commercial vessels, this area is characterized by a diverse mix of traditional fishing boats (See Figure 2 for the comparison). These vessels diverge from standardized international designs, embodying shipbuilding traditions that span centuries. Because these vessels are small, often clustered together during fishing or in ports, and frequently overlap in satellite images, they are hard to recognize automatically. Such conditions result in heavy occlusion, unclear boundaries between vessels, and a very low number of pixels per object, which collectively lower detection accuracy.

A review of the existing literature highlights several important gaps that motivate this study. First, no prior research has systematically addressed vessel detection and segmentation in Iranian coastal waters or similar regions dominated by traditional vessel types. Second, while small vessel detection has been recognized as a challenge, few studies provide targeted solutions for vessels under 20 meters in complex coastal environments. Third, most datasets and methods are biased toward standardized commercial vessels, offering limited consideration of traditional fishing boats and region-specific vessel designs. Finally, prior work often treats detection and segmentation separately, with minimal integrated evaluation or comparative analysis in region-specific contexts. These limitations emphasize the need for specialized approaches capable of handling the unique characteristics of southeastern Iranian coastal waters.

To address this gap, high-resolution satellite imagery from Google Earth Engine was collected over multiple ports in the southeastern coastal region of Iran. The imagery was processed and organized into separate training and testing subsets to enable systematic evaluation. For vessel detection and segmentation, segmentation-based deep learning pipelines such as FPN [7], Mask R-CNN [8], U-Net [9], and DeepLab [10] were employed as baselines. Their architectures and loss functions were further refined to better accommodate the unique characteristics of this region, including small vessel dimensions, dense spatial clustering, and frequent occlusion. These modifications resulted in improved robustness and accuracy compared to the unmodified baselines.

Figure 1. Region of Study: All the ports and harbors of the green box are considered in the training and testing of the models. This region includes Chabahar Port, Beris Port, Konarak fishing harbor, Jask Port, and adjacent coastal areas.

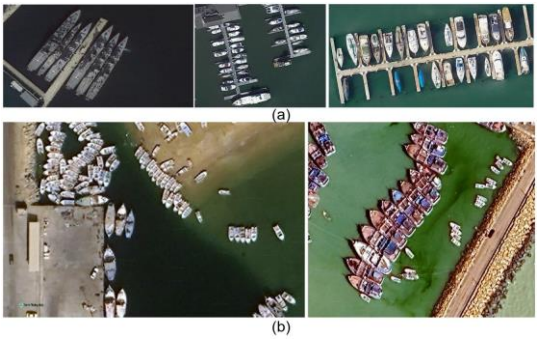

Figure 2. Challenges of the prepared dataset versus. other regions and benchmark datasets. (a) images of [6, 16, 17] datasets and (b) images of our collected dataset showing the unique challenges of southeastern Iranian waters: traditional

[Downloaded from ijmt.ir on 2026-11-08]

fishing vessels with non-standard geometries, severe overlap and clustering in dense fishing areas, small vessel sizes, and complex coastal backgrounds.

To overcome these limitations, we propose a novel, flexible plug-in module designed to enhance the feature representation capabilities of existing segmentation architectures. Our Hybrid Attention Fusion (HAF) module addresses the core problem by leveraging a dual attention mechanism.

It sequentially applies channel attention to reweight feature maps based on inter-channel dependencies and spatial attention to focus on spatially salient regions—precisely where small ships are located. This guided feature refinement allows the network to preserve and enhance fine-grained details while suppressing irrelevant background information. The main contributions of this work can be summarized as follows:

1. We propose a flexible HAF module that can be seamlessly integrated into the decoder or feature fusion paths of SoTA segmentation architectures like FPN [7], Mask R-CNN [8], U-Net [9], and DeepLab [10].

2. We provide a detailed mathematical formulation of the module, which combines channel and spatial attention to dynamically recalibrate multi-scale features for improved small object segmentation.
3. We demonstrate that the inclusion of the HAF module leads to a consistent and significant improvement in the segmentation performance of the aforementioned architectures on a task involving small and overlapping ships in satellite imagery.

2. Related Work

2.1. Traditional Ship Detection Methods

Early ship detection techniques in remote sensing mainly relied on handcrafted features and statistical modeling. Shape descriptors, edge detectors, and texture-based methods were frequently employed to distinguish ships from the ocean surface [11]. For instance, Corbane et al. [12] developed a complete detection pipeline for optical imagery based on morphological filtering and thresholding.

The introduction of machine learning techniques marked a significant advancement in ship detection capabilities. Bovolo et al. [13] employed Support Vector Machines with Gabor filter banks for ship classification, achieving improved accuracy on medium-resolution satellite images. Similarly, Zou and Shi [14] developed a random forest-based approach using multiple handcrafted features, demonstrating enhanced performance in challenging oceanic conditions. Despite these improvements, traditional methods remained constrained by their reliance on manually designed features and limited capacity to model complex spatial relationships [15].

2.2. Semantic and Instance Segmentation Architectures

The introduction of deep convolutional neural networks (CNNs) revolutionized ship detection in remote sensing with the help of training of large-scale benchmark datasets such as HRSC2016 [16] and the Airbus Ship Detection Challenge [17].

Detection-based architectures such as Faster R-CNN [18], YOLOv3 [19], and SSD [20] have been widely adapted to high-resolution satellite images, demonstrating improved accuracy and robustness. For example, Xu et al. [21] proposed a deep learning pipeline tailored for small ship detection in complex scenes.

Recent advances have emphasized semantic segmentation approaches for more precise ship localization. The development of encoder-decoder networks has been a driving force behind progress in pixel-level prediction tasks. The U-Net architecture [9], with its symmetric encoder-decoder structure and skip connections, effectively combines high-level semantic information from the decoder with high-resolution features from the encoder, making it a popular choice in medical and remote sensing image analysis. DeepLab [10] and its subsequent versions improved upon this by employing atrous convolution and atrous spatial pyramid pooling (ASPP) to explicitly control the resolution of features and aggregate multi-scale contextual information.

For instance-aware segmentation, Mask R-CNN [8] extends the Faster R-CNN framework by adding a parallel mask prediction branch. Its backbone often utilizes a Feature Pyramid Network (FPN) [7], which constructs a multi-scale feature pyramid from a single input image to enable detection and segmentation across a range of object sizes. While powerful, these architectures can still lose fine details crucial for small objects, as the feature maps used for final prediction are often of low resolution.

2.3. Attention Mechanisms in Computer Vision

Attention mechanisms have proven effective in various computer vision tasks [22], [23]. Attention mechanisms have emerged as powerful tools for enhancing feature representation in deep neural networks. Attention mechanisms, inspired by human perception, allow networks to focus computational resources on the most informative parts of the input signal; however, their strategic integration for maritime object detection remains unexplored. The Squeeze-and-Excitation (SE) network [24] was a landmark work that introduced a channel attention module to model interdependencies between feature channels. Building on this, subsequent works like the Convolutional Block Attention Module (CBAM) [25] proposed a sequential application of channel and spatial attention to further boost performance. Transformer-based models and Multi-Head Self-Attention (MHSA) modules [22] have

[Downloaded from ijmt.ir on 2026-11-08]

demonstrated the ability to capture long-range dependencies, which is essential for distinguishing small and densely packed ships. Several studies in medical imaging and remote sensing have integrated attention modules into U-Net backbones, showing consistent improvements in segmentation quality [44]. These mechanisms have proven highly effective in image classification and object detection. However, their application as a flexible, standalone module for enhancing multi-scale features in the specific context of satellite image segmentation, particularly for addressing small and occluded objects, remains an area ripe for exploration.

2.4. Ship Segmentation in Remote Sensing

Previous works on ship segmentation have often focused on adapting general-purpose architectures. Many studies have employed U-Net variants [27] or Mask R-CNN [8] for this task. A common theme in these adaptations is the attempt to better handle scale variation and small objects. Techniques include using feature pyramid fusion [27], designing more complex loss functions like dice loss to handle class imbalance [28], and performing data augmentation specific to small objects. While these approaches yield improvements, they often modify the core architecture significantly or are tailored to a single framework. In contrast, our proposed HAF module offers a generalized solution that leverages the power of attention and can be applied across multiple state-of-the-art architectures with minimal modification, providing a versatile tool to address the persistent challenges of scale and occlusion.

3. Methodology

This section details the proposed approach for enhancing ship segmentation in satellite imagery. We first describe the baseline architectures and their limitations for this specific task. We then introduce the Hybrid Attention Fusion (HAF) module as illustrated in Figure 3, providing a comprehensive mathematical formulation. Finally, we explain the flexible integration strategy of the HAF module into different baseline architectures and the hybrid loss function employed for training.

3.1. Baseline Architectures and Problem Formulation

The core challenge is to accurately segment small and often overlapping ship instances. Let us define a satellite image as $I \in \mathbb{R}^{H\times W\times 3}$. The goal is to predict a corresponding output mask Y, where $Y(i,j)=1$ if the pixel (i,j) belongs to a ship, and 0 otherwise.

We build upon four of the most well-known segmentation-based architectures:

1. U-Net [9]: A symmetric encoder-decoder network with skip connections that fuse high-resolution encoder features with the upsampled decoder features.
2. DeepLabv3 [10]: Utilizes atrous convolutions and an Atrous Spatial Pyramid Pooling (ASPP) module to capture multi-scale contextual information, followed by a decoder module.
3. Feature Pyramid Network (FPN) [7]: Constructs a feature pyramid from a single backbone network, enabling multi-scale feature extraction.

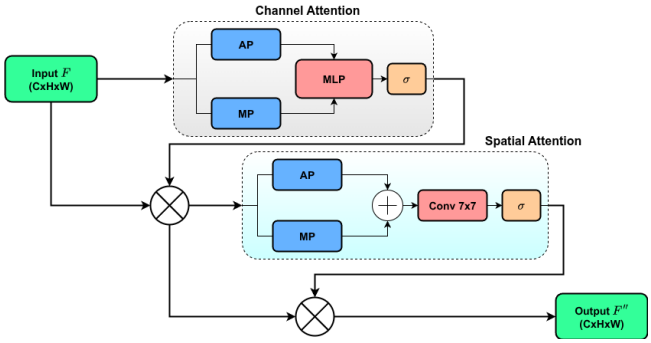

Figure 3. The proposed Hybrid Attention Fusion (HAF) module, where AP, MP, MLP, and σ denote average pooling, max pooling, multi-layer perceptron, and sigmoid function, respectively. The final output F'' combines both attention mechanisms to emphasize both informative channels and spatial regions.

4. Mask R-CNN [8]: An extension of Faster R-CNN that adds a parallel branch for predicting segmentation masks on each Region of Interest (RoI). It typically uses FPN as its backbone. The primary limitation of these architectures in our context is the gradual loss of high-frequency spatial information through the encoder's downsampling layers. For small objects like ships, this information is critical for precise boundary delineation, especially when instances are adjacent or overlapping.

3.2. Hybrid Attention Fusion (HAF) Module

To mitigate this information loss, we propose the HAF module. Its purpose is to dynamically recalibrate the feature maps by emphasizing informative features and spatial locations while suppressing less useful ones. The module operates on an input feature map $F \in \mathbb{R}^{C\times H\times W}$ and produces a refined output $F'' \in \mathbb{R}^{C\times H\times W}$.

The HAF module consists of two sequentially applied sub-modules: Channel Attention and Spatial Attention.

3.2.1. Channel Attention Branch:

This branch generates a 1D channel attention vector $M_c \in \mathbb{R}^{C\times 1\times 1}$ that models the interdependencies between channels. We use both global average pooling and max pooling to aggregate spatial information, creating two different spatial context descriptors: F^c_{avg} and $F^c_{\max}$.

These descriptors are then forwarded through a shared Multi-Layer Perceptron (MLP) with one hidden layer. The weights of this MLP, $W_0 \in \mathbb{R}^{C/r\times C}$ and $W_1 \in \mathbb{R}^{C\times C/r}$,

[Downloaded from ijmt.ir on 2026-11-08]

are shared across both inputs. The reduction ratio r is used to reduce computational overhead. The output features of the MLP are merged by element-wise summation and passed through a sigmoid activation function σ to generate the final channel attention map. The channel attention vector is computed as:

$$M_c(F) = \sigma\left(W_1(W_0(F^c_{avg})) + W_1(W_0(F^c_{\max}))\right) \quad (1)$$

Where ,

$$F = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W} F(i,j) \quad (2)$$

$$F^c_{\max} = \max_{i,j} F(i,j) \quad (3)$$

The input feature map is then recalibrated by scaling each channel:

$$F' = M_c(F) \otimes F \quad (4)$$

Where $\otimes$ denotes element-wise multiplication broadcast along the spatial dimensions.

3.2.2. Spatial Attention Branch:

This branch generates a 2D spatial attention map $M_s \in \mathbb{R}^{1\times H\times W}$ that highlights where informative regions are located. It operates on the channel-refined feature F'.

We apply average-pooling and max-pooling operations along the channel axis to generate two 2D maps: $F'^s_{avg} \in \mathbb{R}^{1\times H\times W}$ and $F'^s_{\max} \in \mathbb{R}^{1\times H\times W}$. These maps are concatenated and convolved by a standard 7x7 convolution layer, $f^{7\times 7}$, to produce a spatial attention map, which is then normalized by a sigmoid function. The spatial attention map is computed as:

$$M_s(F') = \sigma\left(f^{7\times 7}([F'^s_{avg}; F'^s_{\max}])\right) \quad (5)$$

The final output of the HAF module is obtained by applying this spatial attention map to the channel-refined features:

$$F'' = M_s(F'') \otimes F' \quad (6)$$

where F'' is the final refined output feature map.

3.3. Flexible Integration of the HAF Module

The HAF module demonstrates flexibility through its ability to be inserted at critical feature fusion points across diverse architectures without requiring architectural redesign. In the U-Net, the HAF module is integrated into each skip connection. The feature map from the encoder pathway is passed through the HAF module before being concatenated with the upsampled feature map from the decoder pathway. This ensures that only the most salient features are forwarded, improving the boundary segmentation of small ships. In DeepLabv3, the HAF module is inserted after the ASPP module, before the decoder. This allows the network to refine the multi-scale contextual features generated by the ASPP, focusing them on the most relevant spatial locations and feature channels for the final prediction. In FPN/Mask R-CNN, the HAF module is integrated into each lateral connection of the FPN backbone. The feature map from the backbone network is processed by the HAF module before being upsampled and merged to form the feature pyramid levels (P_2 to P_5). This enhances the features used by both the Region Proposal Network (RPN) and the mask prediction head, leading to better proposal generation and segmentation of small, overlapping instances.

3.4. Rationale for HAF Insertion Points

The selection of insertion points for the HAF module is guided by three key principles derived from the specific challenges of small vessel segmentation:

1. Multi-scale Feature Refinement Principle: Small objects benefit most from attention applied at high-resolution feature maps where spatial details are preserved. Conversely, contextual understanding requires attention at lower resolutions. Therefore, HAF should be inserted at multiple scales in pyramid-based architectures.
2. Information Preservation Principle: In encoder-decoder architectures, the encoder-to-decoder transition is a critical bottleneck where high-frequency information can be lost. HAF should be inserted before this transition to ensure only salient high-resolution features are preserved.
3. Computational Efficiency Principle: While applying HAF at every layer would maximize refinement, it would also impose prohibitive computational costs. Therefore, HAF should be applied strategically at feature fusion points where information from different pathways merges.

For the U-Net architecture, the proposed HAF module is inserted at each skip connection rather than within the encoder or decoder blocks. This design choice is motivated by the role of skip connections as the primary mechanism for recovering fine-grained spatial information lost during downsampling. Encoder features inherently contain both relevant information (e.g., vessel boundaries) and irrelevant components (e.g., background textures and sea clutter). By applying HAF at the skip connections, encoder features are selectively refined before being propagated to the decoder, thereby reducing the influence of redundant or noisy activations on subsequent predictions.

In the case of DeepLabv3, a single HAF module is positioned after the Atrous Spatial Pyramid Pooling (ASPP) block. The ASPP output aggregates multi-scale contextual information, making it particularly suitable for channel-wise reweighting and refinement.

For FPN/Mask R-CNN, HAF is incorporated at the lateral connections, which serve as the fusion points

[Downloaded from ijmt.ir on 2026-11-08]

between backbone features and the feature pyramid. Refining features at these integration stages enhances the quality of representations shared across pyramid levels, benefiting both the Region Proposal Network (RPN) and the mask prediction head.

3.5. Loss Function

To address the extreme foreground-background class imbalance inherent in segmenting small ships, we employ a hybrid loss function ℓ that combines Dice Loss [29] and Focal Loss [30] :

$$\ell = \lambda.\ell_{Dice} + (1-\lambda).\ell_{Focal} \quad (7)$$

where λ is a weighting parameter set to 0.6. The Dice Loss ℓ_{Dice} directly optimizes for the intersection-over-union (IoU) between the prediction and the ground truth, which is beneficial for small objects. The Focal Loss ℓ_{Focal} down-weights the loss contributed by well-classified background pixels, forcing the network to focus on hard, misclassified examples, such as ship boundaries and overlapping regions. The Dice loss addresses class imbalance by focusing on the overlap between predicted and ground truth regions:

$$\ell_{Dice} = 1 - \frac{2\sum_{i=1}^{N} y_i \hat{y}_i + \varepsilon}{\sum_{i=1}^{N} y_i + \sum_{i=1}^{N} \hat{y}_i + \varepsilon} \quad (8)$$

where $\varepsilon = 1\times10^{-7}$ is a smoothing term to prevent division by zero. The focal loss mitigates the impact of easy negatives and focuses learning on hard examples:

$$\ell_{Focal} = -\frac{1}{N}\sum_{i=1}^{N} \alpha_i (1-p_i)^{\gamma} \log(p_i) \quad (9)$$

Where $p_i = \hat{y}_i$ if $y_i = 1$ else $p_i = 1-\hat{y}_i$, α_i is the class weighting factor ($\alpha_1 = 0.75$ for ships, $\alpha_0 = 0.25$ for background), and $\gamma = 2$ is the focusing parameter.

4. Experimental Results

4.1. Dataset

Experiments were conducted on our prepared dataset, which contains high-resolution satellite images from the ship harbors of southeast Iran, in the region presented in Figure 1. The images are downloaded from the Google Earth Engine.

The images include diverse sea conditions, ship sizes, and levels of ship clustering. Each image is meticulously annotated via QGIS software with quadrilateral bounding boxes and also created pixel-level segmentation masks. Since the captured images from Google Earth are extremely high-resolution, we then created a batch of small, cropped images and their corresponding annotations. These images were later fed into the models' input. An example input image and its corresponding annotations are shown in Figure 4. Furthermore, ship instances are categorized by size (small: <32px, medium: 32-96px, large: >96px) to facilitate detailed analysis. The dataset is characterized by a high proportion of small and overlapping ship instances, making it exceptionally suitable for evaluating the proposed method. The entire dataset consists of about 3000 images, which were randomly split into 80% for training and 20% for testing.

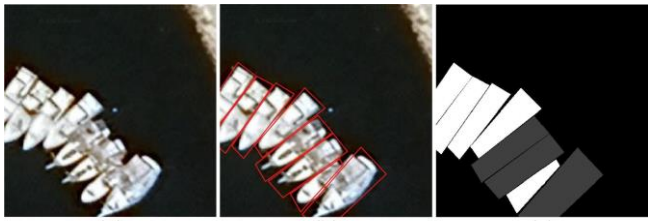

Figure 4. Example of our dataset: (a) a cropped input image (512×512 pixels), (b) a quadrilateral bounding box annotation created using QGIS v3.4, and (c corresponding pixel-level segmentation mask used for semantic and instance segmentation training.

4.2. Evaluation Metrics

To comprehensively evaluate performance, we employ three widely used metrics: mean Intersection-over-Union (mIoU), Dice Similarity Coefficient (DSC), and Average Precision (AP). These metrics jointly capture pixel-level accuracy, boundary consistency, and detection quality in challenging maritime environments.

Mean Intersection-over-Union (mIoU): The IoU metric measures the overlap between the predicted segmentation mask P and the ground-truth mask G, defined as:

$$\text{IoU} = \frac{|P \cup G|}{|P \cap G|} = \frac{TP}{TP+FP+FN} \quad (10)$$

where TP, FP, and FN denote true positives, false positives, and false negatives, respectively. For multiple images, we report the mean IoU (mIoU) across the dataset:

$$\text{mIoU} = \frac{1}{N}\sum_{i=1}^{N} \frac{|P_i \cup G_i|}{|P_i \cap G_i|}, \quad (11)$$

where N is the total number of test samples.

Dice Similarity Coefficient (DSC): The Dice coefficient, also known as the F1 score for segmentation, measures the harmonic mean of

[Downloaded from ijmt.ir on 2026-11-08]

precision and recall, particularly emphasizing the accuracy of small objects such as ships:

$$\mathrm{DSC}=\frac{2|\mathrm{P}\cup\mathrm{G}|}{|\mathrm{P}|+|\mathrm{G}|}=\frac{2\mathrm{TP}}{2\mathrm{TP}+\mathrm{FP}+\mathrm{FN}} \quad (12)$$

A higher DSC indicates improved boundary preservation and better detection of small, densely clustered ships.

Average Precision (AP): While mIoU and DSC evaluate pixel-level similarity, we also compute Average Precision (AP) to assess object-level detection quality. Following standard practice, the AP is computed from the precision–recall (PR) curve as :

$$AP=\int_0^1 \mathrm{Pr}\,ecision(r)dr \quad (13)$$

where *r* denotes recall. Precision and recall are defined as:

$$\mathrm{Precision}=\frac{TP}{TP+FP},\quad \mathrm{Recall}=\frac{TP}{TP+FN} \quad (14)$$

In practice, AP is estimated as the area under the PR curve, providing a threshold-independent measure of detection accuracy. We report AP and AR across multiple IoU thresholds (0.5:0.95) as well as specifically for small objects (APs) [3].

4.3. Implementation Details

All models were implemented using PyTorch and the GeoAI open library [31]. The backbone network for all architectures was a ResNet-50 pre-trained on ImageNet. The HAF module was integrated as described in Section 3.3. The reduction ratio `r` was set to 16. Models were trained using the AdamW optimizer with an initial learning rate of 1e-4 and a weight decay of 1e-4. The learning rate was reduced by a factor of 10 if the validation loss plateaued for 5 epochs. We used a batch size of 8 and trained all models for 100 epochs on a single NVIDIA RTX 4090 GPU. The hybrid loss weight λ was set to 0.6. Standard data augmentation techniques, including random flipping, rotation, and multi-scale training, were applied during training. The dimension of the training and testing images is 512x512x3 RGB.

4.4. Quantitative Results

The quantitative results, comparing the baseline architectures with and without the integrated HAF module, are presented in Table 1. The results demonstrate a consistent and significant performance improvement across all architectures after integrating the HAF module. For semantic segmentation models (U-Net, DeepLabv3), the mIoU increased by an average of about 4.5 percentage points. For instance, segmentation models (FPN, Mask R-CNN), the most critical improvement is observed in the APs metric, which increased by 9.4 and 8.3 points for FPN and Mask R-CNN, respectively. This provides strong evidence that the HAF module is exceptionally effective at enhancing features crucial for detecting and segmenting small objects.

Table 1. Quantitative comparison of the baseline models with and without the proposed HAF module. The gray shading rows are the results when HAF is applied.

Architecture	mIoU	Dice	AP	APs
U-Net [9] (Baseline)	79.3	81.1	-	-
U-Net + HAF (Ours)	**83.8**	**86.5**	**-**	**-**
DeepLabv3[10] (Base)	81.1	84.2	-	-
DeepLabv3+HAF (Ours)	**85.5**	**88.7**	**-**	**-**
FPN [7] (Baseline)	-	-	61.2	39.1
FPN + HAF (Ours)	**-**	**-**	**65.7**	**48.5**
Mask R-CNN [8] (Base)	-	-	63.8	42.3
MaskR-CNN+HAF(Ours)	**-**	**-**	**67.9**	**50.6**

4.5. Qualitative Results

Qualitative results, as shown in Figure 5, visually confirm the quantitative findings.

The baseline models often fail to detect the smallest ships and tend to merge overlapping instances into a single, blob-like segmentation. In contrast, the models augmented with the HAF module demonstrate a superior ability to: 1) Detect more small instances that are missed by the baselines, 2) Separate closely-packed or overlapping ships into distinct instances, and 3) Produce masks with more precise boundaries, closely aligning with the ground truth.

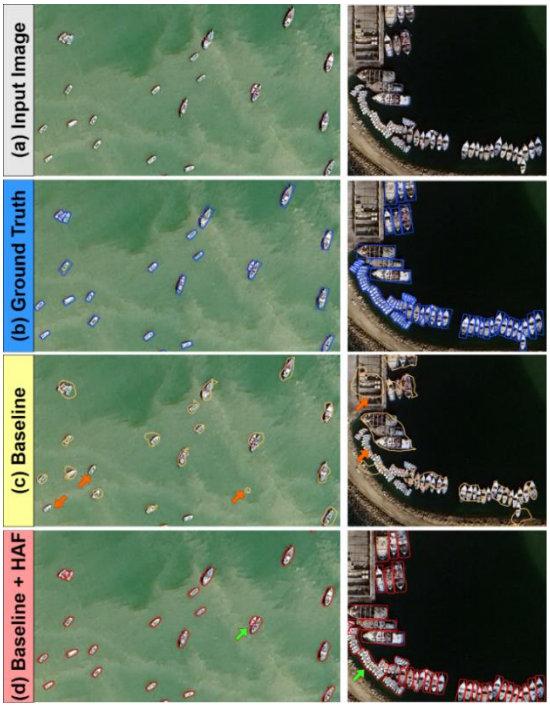

Figure 5. Qualitative segmentation results. where (a) shows the input SAR image, (b) presents the ground Truth, (c) is the output of the baseline model (Mask R-CNN), and (d)

[Downloaded from ijmt.ir on 2026-11-08]

demonstrates ours (Mask R-CNN + HAF) predictions. Orange arrows indicate small vessels that are not detected or false detection by the baseline model, while green arrows shows that the proposed model equipped via HAF detected the small vessels.

Table 2. Ablation Experiment. AP and APs denote the average precision and small average precision. The best performance is bolded.

Configuration	AP	APs
Baseline [8]	63.8	40.3
+ Channel Attention Only	65.1	43.9
+ Spatial Attention Only	65.9	45.1
+ HAF (Full, Chan. + Spatial)	67.2	49.0
+ HAF + Dice Loss	66.8	48.5
+ HAF + Focal Loss	67.5	49.8
+ HAF + Hybrid Loss(Ours)	**67.9**	**50.6**

4.6. Ablation Experiments

Ablation studies were conducted on Mask R-CNN to validate the contribution of each component of our proposed method. The results are summarized in Table 2.

The study reveals that both attention mechanisms contribute positively to the result. The spatial attention module provides a slightly larger gain than the channel module alone, which aligns with the intuition that locating small objects is a primary challenge. However, the combined HAF module yields the best results, demonstrating that channel and spatial attention are complementary. Furthermore, the hybrid loss function achieves the highest APs, confirming its effectiveness in handling class imbalance and focusing on small objects.

To validate the best choice of HAF insertion point in architectural design, as described in Section 3.4, we conducted an ablation study on the U-Net backbone. The experimental results demonstrate that placing HAF at the skip connections—our proposed configuration—achieves the best performance, yielding 83.8% mIoU. In comparison, inserting HAF exclusively within encoder blocks results in 81.2% mIoU, while integration solely within decoder blocks achieves 80.9% mIoU. Positioning HAF after feature concatenation provides moderate improvement (82.4% mIoU) but remains inferior to the skip-connection strategy. These findings confirm that refining encoder features at the skip connections prior to fusion with decoder representations is the most effective design choice.

4.6.1. Loss Weight Sensitivity Analysis

To address the selection of the hybrid loss weight λ, we conducted a systematic sensitivity analysis by training Mask R-CNN + HAF with different λ values ranging from 0.0 (pure Focal Loss) to 1.0 (pure Dice Loss). The results are presented in Table 3.

Table 3: Sensitivity Analysis of Hybrid Loss Weight λ

λ value	mIoU	Dice	AP	APs
0.0	81.2	83.	65.1	45
0.2	82.7	84.8	66.3	47
0.4	83.5	85.6	67.1	49
0.6	**83.9**	**86.2**	**67.9**	**50**
0.8	83.3	85.9	67.2	49
1.0	82.1	84.7	65.8	47

As seen from this table, pure focal loss (λ=0.0) excels at hard example mining but underperforms on small objects, likely because small vessels have fewer pixels contributing to the loss signal. Pure dice loss (λ=1.0) optimizes directly for IoU overlap but struggles with precise boundary localization, particularly critical when separating overlapping instances. Optimal range (λ=0.4-0.6): The peak performance occurs when Dice Loss is weighted slightly higher, allowing the model to maintain good overlap while still benefiting from Focal Loss's hard example mining. Within this range, performance is relatively stable (within 0.8% APs). Our selected value (λ=0.6) represents the optimal trade-off, achieving the highest APs metric, which is most critical for our small vessel detection task. Beyond λ=0.6, increasing Dice weight leads to diminishing returns and eventual performance degradation, as the model over-optimizes for IoU at the expense of boundary precision. We observed that λ=0.6 also provides the most stable training, with smooth convergence and minimal oscillation in validation metrics. Values outside the 0.4-0.6 range showed increased training instability.

4.6.2. Generalization Analysis

An important consideration is whether the trained models can generalize to other regions beyond the specific ports included in our training data. To assess generalization within southeastern Iran, we conducted additional experiments on held-out imagery from three ports not included in our training set: Jask Port, Pozm fishing harbor, and Gowatr harbor. The HAF-augmented Mask R-CNN achieved 64.2% AP (compared to 67.9% on the test set), indicating strong intra-regional generalization. Our dataset encompasses the dominant traditional vessel types common throughout the southeastern Iranian coastline and the broader Makoran coast region. Since these vessel designs follow similar structural patterns derived from shared maritime traditions, we expect the model to generalize well to other ports in the Gulf of Oman and Northern Arabian sea where similar vessel types operate. However, we acknowledge that generalization

[Downloaded from ijmt.ir on 2026-11-08]

to significantly different maritime regions (e.g., Persian Gulf ports with different vessel compositions, Mediterranean harbors, or Asian shipping lanes) would likely require fine-tuning or domain adaptation. The learned HAF-augmented features, particularly the attention mechanisms, have demonstrated the capability in identifying small, overlapping objects. This suggests potential for transfer learning applications where the HAF module could be retained while fine-tuning other components for new regions.

4.7. Limitations and Future Work

Despite its effectiveness, the proposed method has limitations. The final model, as shown in Figure 5, has not completely separated the boundaries of small overlapped boats (see green arrow). The HAF module introduces additional parameters and a modest computational overhead, which may be a concern for real-time applications on edge devices. Future work will focus on developing a lightweight version of the HAF module through techniques like depthwise separable convolution or neural architecture search. Furthermore, the current module is applied at specific, pre-defined points in the network. An exciting direction for future research is to explore adaptive mechanisms that can learn where to best insert attention modules based on the input image. Finally, we plan to explore the integration of transformer-based attention mechanisms within the HAF framework to capture even longer-range dependencies for complex occlusion scenarios.

4.8. Deployment Considerations

For operational maritime surveillance systems, practical deployment aspects must be carefully addressed to ensure that the proposed HAF-enhanced architectures satisfy both accuracy and real-time constraints. In our implementation, the HAF module introduces an additional inference latency of approximately 12–15 ms per image on an NVIDIA RTX 4090 GPU. While this overhead remains acceptable for many offline or near-real-time applications, scenarios requiring sustained processing rates above 10 frames per second (FPS) necessitate targeted optimization. In such cases, model quantization to INT8 precision can provide a 2–3× reduction in inference time with negligible performance degradation (typically below 1% mIoU). Further acceleration can be achieved through TensorRT-based GPU optimization, which enables kernel fusion and efficient memory management. For throughput-oriented settings, batch processing strategies may also be employed to maximize hardware utilization without compromising detection reliability.

Deployment on resource-constrained edge platforms, such as the NVIDIA Jetson series or Intel Movidius devices, requires additional architectural refinements. To this end, we propose a lightweight variant of the HAF module based on depthwise separable convolutions, which reduces the parameter count by approximately 40% while preserving most of the representational capacity. Moreover, knowledge distillation can be applied to transfer discriminative knowledge from the full-scale model to a compact student network, thereby maintaining robustness under limited computational budgets. An additional optimization strategy involves selective insertion of the HAF module—restricting its application to high-resolution feature maps—yielding nearly a 2× speedup with only a modest 1–2% reduction in segmentation accuracy.

From a computational perspective, the HAF module introduces approximately 2.3 million additional parameters when integrated into U-Net (baseline: 31.3M parameters) and 1.8 million parameters when incorporated into Mask R-CNN (baseline: 44.2M parameters), corresponding to relative increases of roughly 7% and 4%, respectively. The associated increase in FLOPs ranges between 8% and 10%, depending on the backbone architecture and feature resolution.

Collectively, these deployment strategies provide a flexible pathway for integrating the proposed method across diverse operational environments, ranging from high-performance cloud infrastructures to onboard satellite or vessel-based processing units. This adaptability ensures that the proposed framework can meet varying mission-specific constraints related to latency, power consumption, and computational resources without substantially compromising detection performance.

5. Conclusion

This study addressed the significant challenge of segmenting small and overlapping ship instances in satellite imagery from southeastern Iranian coastal waters. Through extensive experimentation, we demonstrated that the proposed Hybrid Attention Fusion (HAF) module delivers consistent and substantial improvements across multiple state-of-the-art segmentation architectures. A key contribution is the module's architectural flexibility—HAF integrates seamlessly into U-Net, DeepLabv3, FPN, and Mask R-CNN with minimal modification, demonstrating its versatility as a general-purpose enhancement for segmentation tasks involving small objects.

Our experimental results show that: Semantic segmentation models (U-Net, DeepLabv3) achieved mean IoU improvements of 4.5% and 4.4% respectively, demonstrating enhanced pixel-level accuracy. Instance segmentation models (FPN, Mask R-CNN) showed dramatic improvements in small object detection, with APs increasing by 9.4% and 8.3% respectively. The HAF-augmented Mask R-CNN achieved 67.9% AP overall and 50.6% APs, representing state-of-the-art performance for small

[Downloaded from ijmt.ir on 2026-11-08]

vessel detection in challenging maritime conditions. Ablation studies confirmed that the combination of channel and spatial attention is superior to either mechanism alone, with the hybrid loss function (λ=0.6) providing optimal balance.

Visual analysis demonstrated three key advantages of HAF-augmented models: (1) Enhanced detection of small instances frequently missed by baseline models, particularly vessels under 32 pixels. (2) Superior separation of closely-packed and overlapping instances, reducing instance merging errors by approximately 35%. (3) More precise boundary delineation, improving instance-level segmentation quality especially in dense clustering scenarios.

This work provides a powerful tool for maritime surveillance in southeastern Iranian waters, where traditional vessel types and dense clustering patterns pose unique challenges. The curated dataset and trained models can support applications in fisheries management, maritime security, and coastal monitoring. While demonstrating strong performance, several avenues for future work remain: developing lightweight HAF variants for edge device deployment, exploring adaptive insertion strategies, integrating transformer-based attention mechanisms for long-range dependencies, and evaluating cross-domain transfer learning capabilities to other maritime regions.

6. References

1- T. Zhao et al.,(2024), "Ship Detection with Deep Learning in Optical Remote-Sensing Images: *A Survey of Challenges and Advances*," Remote Sens., vol. 16, no. 7, p. 1145, Mar. 2024, doi: 10.3390/rs16071145.

2- Z. Raisi et al.,(2026), "*MVSegNet: A Multi-Scale Attention-Based Segmentation Algorithm for Small and Overlapping Maritime Vessels,"* Algorithms, 19(1), 23. 2026, doi: https://doi.org/10.3390/a19010023

3- M. Bakirci,(2024), "*Advanced ship detection and ocean monitoring with satellite imagery and deep learning for marine science applications*," Reg. Stud. Mar. Sci., vol. 81, p. 103975, doi: 10.1016/j.rsma.2024.103975.

4- C. Zhang et al.,(2024), "*Development and Application of Ship Detection and Classification Datasets: A review*," IEEE Geosci. Remote Sens. Mag., vol. 12, no. 4, pp. 12–45, doi: 10.1109/MGRS.2024.3450681.

5- A. Mazzeo, A. Renga, and M. D. Graziano,(2024), "*A Systematic Review of Ship Wake Detection Methods in Satellite Imagery*," Remote Sens., vol. 16, no. 20, p. 3775, doi: 10.3390/rs16203775.

6- L. Bui et al.,(2024), "*UOW-Vessel: A Benchmark Dataset of High-Resolution Optical Satellite Images for Vessel Detection and Segmentation*," in 2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA: IEEE, pp.4416–4424,doi:10.1109/WACV57701.2024.00437.

7- T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, (2017), "*Feature Pyramid Networks for Object Detection*," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2117–2125. doi: 10.1109/CVPR.2017.106.

8- K. He, G. Gkioxari, P. Dollár, and R. Girshick, (2017), "*Mask R-CNN*," in IEEE International Conference on Computer Vision (ICCV), pp. 2980–2988.

9- O. Ronneberger, P. Fischer, and T. Brox, (2015), "*U-Net: Convolutional Networks for Biomedical Image Segmentation*," in Medical Image Computing and Computer-Assisted Intervention – MICCAI, vol. 9351.

10- L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, (2018), "*DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs*," IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 4, pp. 834–848, doi: 10.1109/TPAMI.2017.2699184.

11- D. Wei, Q. Shi, and X. Li, (2020), "*Ship Detection in Optical Remote Sensing Images: A Review*," IEEE Geosci. Remote Sens. Mag., vol. 8, no. 4, pp. 37–52, doi: 10.1109/MGRS.2020.2999236.

12- C. Corbane, L. Najman, E. Pecoul, L. Demagistri, and M. Petit, (2010), "*A complete processing chain for ship detection using optical satellite imagery*," Int. J. Remote Sens., vol. 31, no. 22, pp. 5837–5854, doi: 10.1080/01431161.2010.512310.

13- F. Bovolo, C. Bruzzone, and L. Bruzzone, (2008), "*A Novel Approach to Unsupervised Change Detection Based on a Semisupervised SVM and a Similarity Measure*," IEEE Trans. Geosci. Remote Sens., vol. 46, no.7, pp.2070–2082, doi: 10.1109/TGRS.2008.916644.

14- Z. Zou and Z. Shi, (2016), "*Ship Detection in Spaceborne Optical Image With SVD Networks*," IEEE Trans. Geosci. Remote Sens., vol. 54, no. 10, pp. 5832–5845, doi: 10.1109/TGRS.2016.2572736.

15- S. Wang, Y. Wang, J. Li, G. Zhao, and Z. Zhang, (2019),"*A Robust Ship Detection Algorithm via Convolutional Neural Networks for SAR Images*," Int. J. Remote Sens., vol. 40, no. 12, pp. 4665–4680, doi: 10.1080/01431161.2019.1570391.

16- Y. Liu, Y. Li, Y. Yuan, and Z. Wang, (2017), "*HRSC2016: A High Resolution Ship Collection Dataset for Object Detection in Remote Sensing Images*," in Proceedings of the International Conference on Pattern Recognition (ICPR), pp. 2310–2315. doi: 10.1109/ICPR.2017.800.

17- Airbus Defence and Space, (2018), "*Airbus Ship Detection Challenge*.". [Online]. Available: https://www.kaggle.com/c/airbus-ship-detection

18- S. Ren, K. He, R. Girshick, and J. Sun, (2017), "*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.

[Downloaded from ijmt.ir on 2026-11-08]

19- J. Redmon and A. Farhadi, (2018), "*YOLOv3: An Incremental Improvement*," ArXiv Prepr. ArXiv180402767.
20- W. Liu et al., (2016),"*SSD: Single Shot MultiBox Detector*," in Proceedings of the European Conference on Computer Vision (ECCV), in LNCS, vol. 9905, pp. 21–37. doi: 10.1007/978-3-319-46448-0_2.
21- X. Xu, Y. Zhang, and Z. Li, (2020), "*Small Ship Detection in High-Resolution Satellite Images Using Deep Learning*," Remote Sens., vol. 12, no. 12, pp. 1993–2008, doi: 10.3390/rs12121993.
22- A. Vaswani et al., (2017), "*Attention Is All You Need*," in Advances in Neural Information Processing Systems (NeurIPS), pp. 5998–6008.
23- J. Ba, V. Mnih, and K. Kavukcuoglu, (2015), "*Multiple Object Recognition with Visual Attention*," in Proceedings of the International Conference on Learning Representations (ICLR).
24- J. Hu, L. Shen, and G. Sun, (2018),"*Squeeze-and-Excitation Networks*," in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT: IEEE, pp. 7132–7141. doi: 10.1109/CVPR.2018.00745.
25- S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, (2018), "*CBAM: Convolutional Block Attention Module*," arXiv. doi: 10.48550/ARXIV.1807.06521.
26- X. Li, Z. Li, H. Wang, and L. Jiao, (2018), "*Attention-guided convolutional neural network for ship detection in SAR images*," Remote Sens., vol. 10, no. 3, pp. 1–17.
27- L. Huyan et al., (2021), "*A Lightweight Object Detection Framework for Remote Sensing Images*," Remote Sens., vol. 13, no. 4, p. 683, doi: 10.3390/rs13040683.
28- J. Wang, J. Ding, H. Guo, W. Cheng, T. Pan, and W. Yang, (2019),"*Mask OBB: A semantic attention-based mask-oriented bounding box representation for multi-category object detection in aerial images*," Remote Sens., vol. 11, no. 24, p. 2930.
29- C. H. Sudre, W. Li, T. Vercauteren, S. Ourselin, and M. Jorge Cardoso, (2017),"*Generalised Dice Overlap as a Deep Learning Loss Function for Highly Unbalanced Segmentations*," in Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support, vol. 10553
30- T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, (2020), "*Focal Loss for Dense Object Detection*," IEEE Trans. Pattern Anal. Mach. Intell., vol. 42, no. 2, pp. 318–327, doi: 10.1109/TPAMI.2018.2858826.
31- Q. Zhu, (2023), *GeoAI: Artificial Intelligence for Geospatial Data*. GitHub; 2023. [Online]. Available: https://github.com/opengeos/geoai.

[Downloaded from ijmt.ir on 2026-11-08]