# CTS: A Unified Deep Learning Approach for Ship Detection in Complex Maritime Environments

Wenxiu Wang, Danbei Wang

*Abstract*—**This study presents an innovative high-precision ship detection framework, named CTS, which combines the Transformer architecture with CNN in an ingenious manner. To address the challenges posed by intricate backgrounds in ship images, our methodology adopts a dual-path feature extraction mechanism, integrating CNN and Transformer features via a novel fusion module, and subsequently employs a semantic segmentation network for meticulous pixel-level detection. Experimental outcomes underscore the superiority of our approach over conventional methods, attaining a Pixel Accuracy of 95.26%. The framework introduced in this study marks a notable progression in leveraging remote sensing techniques for ship detection, providing superior feature characterization and improved detection precision in challenging maritime environments.**

*Index Terms*—**Convolutional Neural Network, Transformer, Semantic segmentation, Feature fusion.**

## I. Introduction

REMOTE sensing-based ship detection has garnered increasing attention within the intelligent shipping community due to its notable advantages, including a wide field of view and minimal susceptibility to ground interference. These characteristics render it an ideal and highly efficient approach for ship detection across diverse maritime environments [1]. Concurrently, advancements in space technology have led to an exponential growth in the volume of remote sensing data, providing abundant technical resources and data support for ship target detection. However, identifying ship targets in remote sensing images has become progressively more challenging, primarily due to the significant variations in ship size [2], a wide array of categories, and intricate background features.

As computer vision technology rapidly advances, the accuracy limitations of conventional object detection techniques have become increasingly apparent [3]. Within this domain, object detection techniques powered by deep learning have become a focal point of research, with frameworks like Fast R-CNN, SSD, and YOLO demonstrating notable advancements. Research shows that Zhou et al. [4] innovatively integrated the ECA mechanism and the BiFPN into the YOLO V5 framework, achieving a remarkable improvement in monitoring accuracy. Zhang et al. [5], on the other hand, enhanced the SSD algorithm by introducing the Dense RBF and LSTM networks, which effectively improving detection accuracy and significantly strengthening the algorithm's adaptability to complex scenarios such as blurred images, target occlusion, and partial cropping. However, existing methods primarily rely on bounding box-annotated training samples for feature learning, whitch may lead to misidentification of background pixels as ship features, thereby affecting detection performance.

To bridge this gap, semantic segmentation models have become an important research direction for suppressing object interference [6]. FCN-based architectures [7] established the standard framework for semantic segmentation algorithms. Subsequent classic algorithms, such as UNet [8], SegNet [9], and PSPNet [10], effectively addressed the challenges in semantic segmentation. Additionally, the DeepLab series (including v1 [11], v2 [12], v3 [13], and v3+ [14]) advanced the field by improving feature representation and segmentation accuracy. These methods laid the foundation for pixel-level ship target detection by providing robust frameworks for feature extraction and pixel-wise classification.

Chen et al. [15] proposed an FCN-based method, termed SNFCN and SDFCN, which incorporates a densely connected block structure to enhance feature propagation and reuse across layers. In the SDFCN framework, three additional mapping connections are introduced between the encoder and decoder to facilitate effective gradient backpropagation. Tian [16] integrated the strengths of FCN and GAN to expand the model's receptive field, significantly improving semantic segmentation performance. Furthermore, Chen et al. [17] proposed an enhanced DeepLab method, which employs Conditional Random Fields (CRF) combined with Recurrent Neural Networks (RNN) are employed to fine-tune segmentation details, which considerably enhances the mean Intersection over Union (mIoU). Moreover, attention mechanisms [18] [19] have emerged as a critical research direction, enabling high-precision object detection and segmentation by dynamically focusing on salient features.

Despite significant progress in deep learning for remote sensing ship detection, limitations persist. CNNs face limited receptive fields, requiring multiple layers to capture global information. However deeper networks risk gradient vanishing or information loss, reducing model expressiveness. Additionally, CNNs struggle with feature fusion, particularly integrating multi-scale or contextual information. Transformers, successful in sequence modeling tasks like speech processing [20], have gained traction in computer vision for their ability to model long-range dependencies [21], [22]. Their self-attention mechanism efficiently captures global context, enhancing model expressiveness [23]. However, Transformers lack the translation invariance and localized feature extraction of

Manuscript received October 20, 2024; revised May 13, 2025.

This work was supported by the Youth Foundation of Jinling Institute of Technology (040521200106; 040521400151).

Wenxiu Wang is an Lecturer of School of Electronic Information Engineering, Jinling Institute of Technology, Nanjing, China ( corresponding author phone: 181-6809-2519; e-mail: wangwenxiu@jit.edu.cn).

Danbei Wang is an Lecturer of School of Electronic Information Engineering, Jinling Institute of Technology, Nanjing, China ( e-mail: 00000005182@jit.edu.cn).

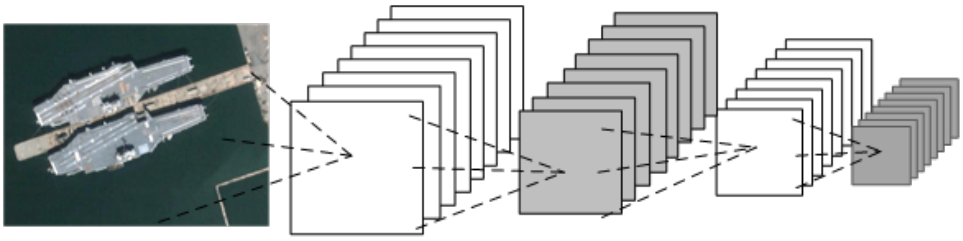

Fig. 1: Feature extraction process of convolutional neural network

CNNs, making them less effective in data-limited scenarios due to higher parameter complexity and weaker inductive biases.

In summary, this paper presents a novel framework, CNN-Trans-Segmentation (CTS), which combines Transformer with CNNs to enable dual-path feature extraction and encoding. During the decoding and segmentation stages, features derived from both pathways are combined to form a unified, end-to-end semantic segmentation framework that delivers highly precise pixel-level ship detection. The main contributions of this study are detailed below:

1) Global Context Modeling: Utilizing the Transformer architecture, the self-attention mechanism efficiently captures long-range relationships between every position in the input sequence. This capability enhances the model's comprehension of the overall context and improves its ability to extract distinctive features from ship targets.

2) Local Feature Enhancement: By leveraging the local feature extraction capabilities of CNNs, the network's receptive field is expanded, which in turn strengthens its ability to detect ship targets in remote sensing imagery while enhancing the overall effectiveness of feature representation.

The remainder of this paper is organized as follows. Section II reviews previous research on both conventional CNN architectures and Transformer models. In Section III, we describe the proposed network model, and Section IV outlines the experimental setup, optimization strategy, and performance evaluation. Lastly, Section V provides the conclusions of the study.

## II. RELATED WORK

In this study, we present an innovative method that merges the benefits of both CNNs and Transformers, aiming to deliver detection with enhanced accuracy and robustness. We first conduct a comprehensive analysis of traditional CNNs, with a focus on the VGG architecture. Subsequently, we delve into the Transformer architecture, examining its self-attention mechanism and global context modeling capabilities, which are critical for enhancing detection performance.

### *A. Transformer-based feature extraction*

In the Transformer-based feature extraction process, the input image $X$ with height $H$ and width $W$ is first divided into $N = \frac{H}{S} \times \frac{W}{S}$ parts. The values of $C = 3$ and $S = 16$ are typical. The three-dimensional image is transformed into a two-dimensional input$(N, D)$, where $D$ is the dimension corresponding to each image patch. This involves performing a linear transformation on each $S^2 \times C$ image patch to form a sequence of $N \times (S^2 \times C)$ [24].

Next, the segmented image sequence is fed into the multi-head self-attention (MSA) module of the Transformer encoder. This approach allows the model to concentrate on various parts of the input patches and learn relationships among them [27]. The input sequence is transformed using a linear matrix, generating $q, k, v$. Attention operations are then performed on $q, k$ to obtain the weights $\alpha$, and a weighted sum is computed between $\alpha$ and $v$.

$$SA(Z_i) = softmax(\frac{q_i k^T}{\sqrt{D}}) \tag{1}$$

The output is passed through a multi-layer perceptron (MLP) to simulate complex nonlinear functions in a fully connected manner. Finally, the output undergoes normalization to obtain the final encoded sequence, completing the Transformer process. Figure 1 illustrates this process, with L set to 8.

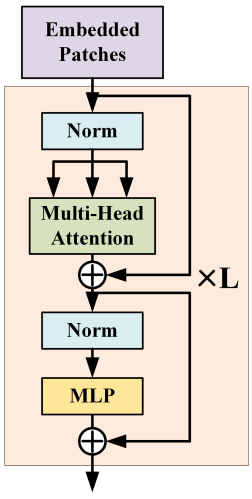


Fig. 2: Feature extraction process based on Transformer

To achieve fusion of features from the Transformer and CNN, the output sequence from the Transformer is transformed into parameters of size $N \times \frac{H}{S} \times \frac{W}{S}$. An upsampling process is then designed to generate featuremaps of size $\frac{H}{8} \times \frac{W}{8}$ and $\frac{H}{4} \times \frac{W}{4}$, which are subsequently fused with the CNN feature layers for comprehensive feature integration.

TABLE I: Feature extraction layer of Transformer

| Layer | Input Size | Output Size |
|---|---|---|
| rashape | $H \times W$ | $\frac{H}{16} \times \frac{W}{16}$ |
| upsampling | $\frac{H}{16} \times \frac{W}{16}$ | $\frac{H}{8} \times \frac{W}{8}$ |
| upsampling | $\frac{H}{8} \times \frac{W}{8}$ | $\frac{H}{4} \times \frac{W}{4}$ |

### B. CNN-based feature extraction

CNN-based feature extraction leverages convolutional neural networks to automatically learn spatial and hierarchical features from images. By employing convolutional layers, pooling operations, and nonlinear activation functions, CNNs incrementally derive increasingly complex feature representations. The early layers are responsible for detecting edges and textures, whereas the deeper layers capture intricate patterns such as shapes and parts of objects. This layered strategy yields advanced representations that make CNNs particularly effective for applications such as object recognition, segmentation, and various other image processing applications. The process of extracting features from ship targets is depicted in Figure 1.

In the CNN extraction process, $X \in \mathbb{R}^{H \times W \times C}$ is the original image, where $H$ is the height of image , $W$ represents the image's width, $C$ represents the image's depth. The pooling layer (typically max pooling or average pooling) can reduce the feature map size, lowering computational complexity while retaining important feature information. With a pooling window of size 2, each $2 \times 2$ region is reduced to a single value, Thus, the dimensions of the feature mapłspecifically are defined as $\frac{H}{2} \times \frac{W}{2}$. After two pooling layers, the size is $\frac{H}{4} \times \frac{W}{4}$. After three pooling layers, the size is $\frac{H}{8} \times \frac{W}{8}$. The LeNet architecture, initially introduced in [25], laid foundational groundwork for convolutional neural network. This foundational framework was subsequently developed and refined. Notably, the AlexNet and VGG architectures marked significant advancements.

### C. VGG19

VGG19 [26] was specifically designed to handle large-scale image recognition tasks. It features an architecture of 19 layers that uniformly utilizes convolution filters with size $3 \times 3$ and pooling layers $2 \times 2$ to effectively extract features at various depths. Its depth is a key strength, enabling accurate object recognition and localization through the capture of complex, hierarchical features. Moreover, VGG19 is often selected for transfer learning because its pre-trained weights, derived from extensive datasets like ImageNet, enable effective fine-tuning for new tasks, thereby making it a versatile asset in computer vision applications. In this study, the feature extraction process leverages the outputs from the first four pooling layers of VGG19, effectively capturing low- to mid-level features essential for ship target detection.

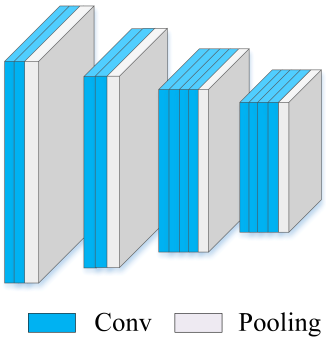


Fig. 3: Feature extraction process of convolutional neural network

TABLE II: Feature extraction layer of CNN

| | Kernel | Filters | Stride | Output Size |
|---|---|---|---|---|
| Conv1_1 | 3×3 | 64 | 1 | $H \times W$ |
| Conv1_2 | 3×3 | 64 | 1 | $H \times W$ |
| MaxPool1 | 2×2 | - | 2 | $\frac{H}{2} \times \frac{W}{2}$ |
| Conv2_1 | 3×3 | 128 | 1 | $\frac{H}{2} \times \frac{W}{2}$ |
| Conv2_2 | 3×3 | 128 | 1 | $\frac{H}{2} \times \frac{W}{2}$ |
| MaxPool2 | 2×2 | - | 2 | $\frac{H}{4} \times \frac{W}{4}$ |
| Conv3_1 | 3×3 | 256 | 1 | $\frac{H}{4} \times \frac{W}{4}$ |
| Conv3_2 | 3×3 | 256 | 1 | $\frac{H}{4} \times \frac{W}{4}$ |
| Conv3_3 | 3×3 | 256 | 1 | $\frac{H}{4} \times \frac{W}{4}$ |
| MaxPool3 | 2×2 | - | 2 | $\frac{H}{8} \times \frac{W}{8}$ |
| Conv4_1 | 3×3 | 512 | 1 | $\frac{H}{8} \times \frac{W}{8}$ |
| Conv4_2 | 3×3 | 512 | 1 | $\frac{H}{8} \times \frac{W}{8}$ |
| Conv4_3 | 3×3 | 512 | 1 | $\frac{H}{8} \times \frac{W}{8}$ |
| MaxPool4 | 2×2 | - | 2 | $\frac{H}{16} \times \frac{W}{16}$ |

## III. Our Method

### A. Overall Network Architecture of CTS

To accurately identify ship pixels in remote sensing images, we propose a novel framework CTS that integrates the strengths of both CNN and Transformer. As shown in Figure 4, the CTS framework is structured around four principal components: (1) Transformer-based feature extraction and upsampling, which captures global context through self-attention; (2) CNN-based feature extraction, which extracts local spatial features; (3) fusion of Transformer and CNN features through concatenation, enabling the integration of global and local information; and (4) upsampling of the fused features to restore spatial resolution for pixel-level prediction.

The Transformer feature extraction and upsampling section includes patch embedding, the Transformer encoder, and an upsampling module (leftward arrow in Figure 4). The CNN feature extraction module comprises convolutional layers and downsampling operations (downward arrow in Figure 4). Transformer and CNN features are fused through element-wise multiplication and addition, integrating global context and local details. The fused features are then concatenated with CNN layer outputs and upsampled to generate the final pixel-level detection results.

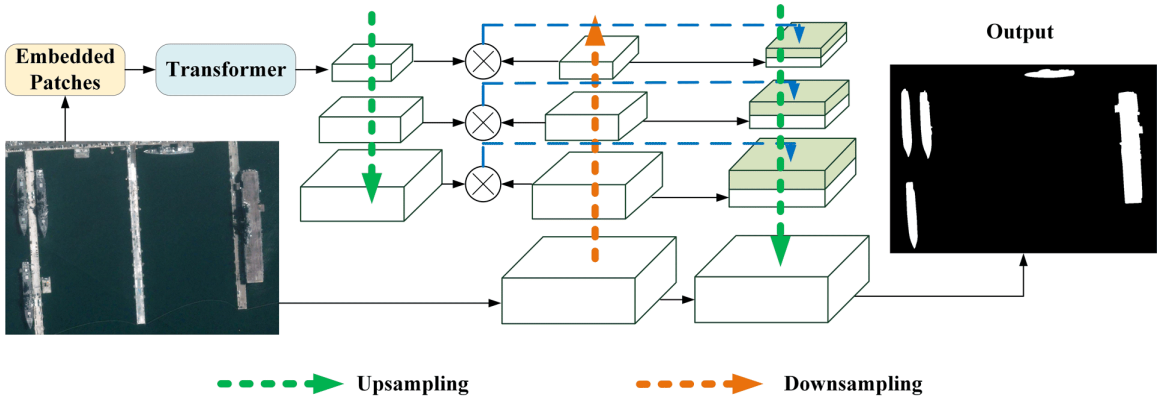


Fig. 4: Overview for the Proposed Ship Target Detection Method CTS

### B. Downsampling and Upsampling

The downsampling process uses Max Pooling, which not only reduces spatial dimensions but also improves feature robustness by preserving the most salient information. The mathematical representation of Max Pooling is defined by the expression:

$$O(i,j) = \max_{m,n}(Input_{(i+m),(j+n)}) \tag{2}$$

where $O(i,j)$ is the value at position $(i,j)$ in the output feature map, and $Input_{(i+m),(j+n)}$ represents the values in the local window of the input feature map, with $m$ and $n$corresponding to the dimensions of the pooling window.

Upsampling is a critical step to restore the spatial resolution of feature maps, ensuring the output matches the input image dimensions for pixel-level classification. In our model, we employ Transposed Convolution for upsampling, which not only reconstructs spatial dimensions but also leverages learnable parameters to enhance feature representation. This approach improves segmentation performance by effectively capturing spatial dependencies and fine-grained details in the data.

$$O(i^{'},j^{'}) = \sum_{m=0}^{k-1}\sum_{n=0}^{k-1} Input(i^{'}+m\cdot S, j^{'}+n\cdot S)W(m,n) \tag{3}$$

Where, $O(i^{'},j^{'})$ refers to the value at the $(i^{'},j^{'})$ position of the output feature map, while $Input$ indicates the feature map undergoing upsampling. The transposed convolution kernel is characterized by the weights $W(m,n)$, with $k$ representing its size and $s$ denoting the stride that governs the intervals at which the kernel is applied.

Transposed convolution is a widely used upsampling technique in CNNs. It uses learnable weights to upscale low-resolution feature maps. During training, these weights are optimized to capture complex patterns like textures and edges, allowing the network to better distinguish image regions and enhancing segmentation performance.

### C. Fusion Module

The parallel fusion strategy operates both CNN and Transformer branches simultaneously, merging their features in a way that maximizes the strengths of each. This approach enables a complementary integration of local features from CNN with global information from Transformer, effectively leveraging the unique advantages of both architectures.

We implement feature fusion using feature multiplication, enabling a complementary combination of CNN and Transformer features by modeling fine-grained interactions between the features from both branches. The multiplication effectively integrates the strengths of each approach, enabling the model to recognize intricate patterns and dependencies through robust feature representations. This fusion strategy is particularly beneficial for tasks requiring detailed feature integration.

## IV. Experiment Results and Analysis

### A. Dataset and Evaluation

The experiment utilizes the instance segmentation portion of the HRSC2016 dataset, which encompasses 444 segmented instances of targets such as aircraft carriers and military ships. The segmentation data employs different colors to distinguish individual instances within the same category, which are converted to grayscale for simplified processing. Prior to the experiment, the dataset is preprocessed and cleaned to enhance data quality.

To begin with, elements such as coastlines that do not represent ships are classified as background, while ship targets depicted in color are identified as foreground objects, as illustrated in Figure 5. In the first row, the unprocessed images are displayed, whereas the second row presents the initial segmentation outcomes layered on top of these images, and the third row displays the refined semantic segmentation outcomes designed to improve ship detection accuracy. During experimentation, three images with inaccurate pixel-level labels from the original dataset were removed to ensure the remaining labels were more accurate and reliable.

Following initial enhancements, a set of 441 high-quality remote sensing images featuring ships was retained, representing over 1,200 vessels and averaging 2.7 ships per image. The dataset is augmented by applying horizontal and vertical flips, as well as random affine transformations (translation, rotation, and scaling). This results in a new ship detection dataset, providing a more diverse set of training samples for developing robust detection models.

The training and testing processes utilize a 64-bit Ubuntu 18.04 system, RTX 3060 GPU, CUDA 11.4, and PyTorch 1.10.2, along with various graphics processing and display

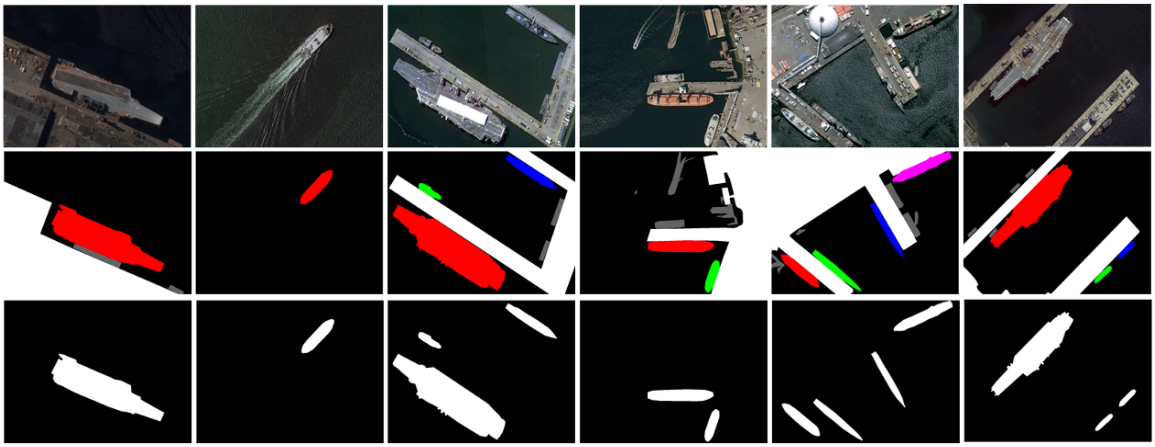

Fig. 5: Adjustment of ship detection data set

libraries, enabling effective training process analysis and observation. Around 70% of the dataset was utilized for training purposes, with the remaining 30% allocated for testing. The model was trained and tested using the PyTorch.

To further confirm the efficacy of our approach, We conducted a quantitative analysis utilizing metrics like Pixel Accuracy (PA), Dice Ratio (DR), and Intersection over Union (IoU) [29]. The definitions of PA and IoU are provided below:

$$PA = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k}\sum_{j=0}^{k} p_{ij}} \tag{4}$$

$$DR = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{5}$$

$$IoU = \frac{area(C) \bigcap area(G)}{area(C) \bigcup area(G)} \tag{6}$$

In Equation (4), $P_{ij}$ represents the count of pixels that truly belong to class $i$ but have been classified as class $j$. In Equation (5), TP indicates the number of genuine positive identifications, FP denotes the instances incorrectly marked as positive, and FN refers to the positive instances that were erroneously classified as negative. In Equation (6), $P_{ii}$ denotes the count of pixels that have been correctly identified. Meanwhile, $P_{ji}$ represents the number of background pixels that were incorrectly classified as targets, and $P_{jj}$ is the number of background pixels correctly recognized.

### *B. Ablation Experiments*

In remote sensing image processing, integrating CNN and Transformer into feature fusion modules is essential for enhanced performance. CNNs capture local details such as edges and textures, while Transformers model long-range dependencies via self-attention. Combining these strengths yields efficient and robust feature representations, which are crucial for accurate remote sensing models.

There is a significant difference between the features extracted by CNN and Transformer, yet both can be aligned in terms of spatial dimensions after appropriate processing. Based on this, this paper proposes strategies for feature fusion, including pointwise multiplication, addition, and elementwise multiplication followed by elementwise addition, and validates their effectiveness through comparative experiments. Specifically, elementwise multiplication is applied to highlight the most relevant features from both models. This method integrates the CNNs capability of extracting local features with the Transformers proficiency in global modeling, thereby improving both the precision and breadth of feature representation.

During the Transformer feature extraction process, smaller patch sizes ($8 \times 8$) preserve more local detail, but increase the number of patches and computational cost. Conversely, larger patch sizes ($32 \times 32$) capture more global information but may lose some finer details. To balance preserving local details with capturing global information, this paper sets the patch size to $16 \times 16$. Based on this, the input image size is configured as $256 \times 192$.

In comparative experiments, we evaluated several fusion methods and analyzed their impact on model performance. Based on three key metrics-Dice ratio, IOU, and Pixel accuracy-we conducted an analysis. The metrics curves for different epochs are shown in Figures 6, 7, 8.

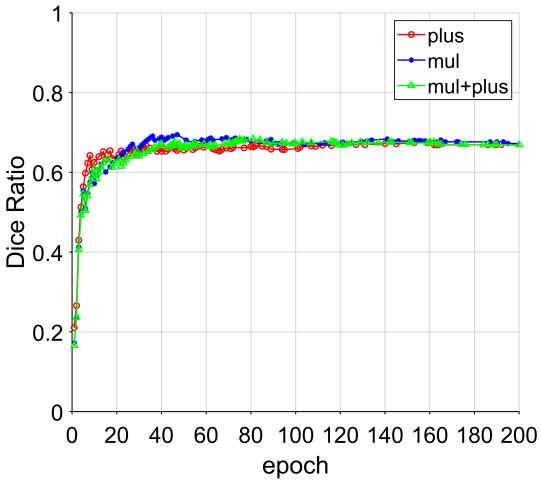


Fig. 6: Dice Ratio curves of trans

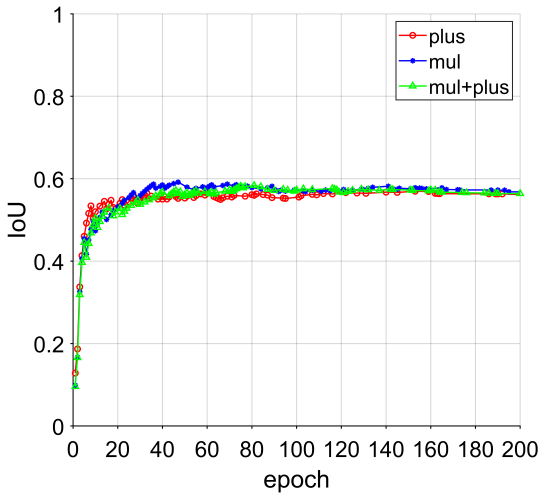


Fig. 7: IoU curves of trans

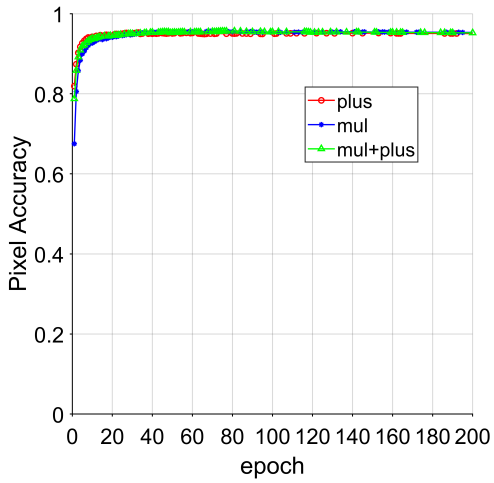


Fig. 8: Pixel Accuracy curves of trans

Table 2 presents the average accuracy and average loss values for different feature fusion modules, including pointwise multiplication, addition, and elementwise multiplication followed by addition, on the test dataset.

TABLE III: The segmentation results of different transformer and CNN feature fusion

| | Dice Ratio | IoU | Pixel Accuracy |
|---|---|---|---|
| mul | 0.6659 | 0.5569 | 94.99% |
| plus | **0.6726** | **0.5652** | **95.27%** |
| mul+plus | 0.6644 | 0.5561 | 95.10% |

In general, the performance differences among the three fusion methods are relatively small with respect to Dice Ratio, IoU, and Pixel Accuracy. However, multiplication-based feature fusion achieves better performance in all three evaluated criteria compared to the other two approaches. In our method, the use of multiplication for feature integration accentuates common significant attributes between the two branches while diminishing the influence of irrelevant information. This fusion technique improves the model's ability to focus on significant areas, resulting in higher accuracy and better generalization.

To further preserve unique information from both branches, CTS concatenates CNN and Transformer features after multiplication. This retains local details from CNN and global context from Transformer, enabling the model to exploit complementary information. This dual strategy enhances performance on complex tasks such as ship target segmentation in remote sensing images.

### C. Comparative Experiments

This study evaluates the effectiveness of the proposed method by comparing it with two widely recognized semantic segmentation techniques, SegNet and UNet, which are noted for their broad utilization and proven performance. Qualitative segmentation results are shown in Figure 9.

Based on the segmentation results compared to the ground truth labels (GT), both SegNet and UNet fail to accurately capture the complete edges of targets. Additionally, individual targets are frequently split into multiple instances, indicating a need for better handling of target density and clustering. Furthermore, both methods exhibit a high number of false positives and missed detections, particularly in dense target regions.

This study compares CTS with four high-performance CNN-based segmentation networks, including PSPNet, DenseASPP, DeepLabV3, and DeepLabV3+. These networks have consistently excelled in semantic segmentation while also achieving outstanding results in object detection, instance segmentation, and image classification. Their robust performance stands as a credible benchmark that attests to the efficacy and benefits of CTS.

In our experiments, we evaluated network performance using three key metrics: Dice Ratio (DR), Intersection over Union (IoU), and Pixel Accuracy (PA). The corresponding performance curves are shown in Figures 10, 11, and 12.

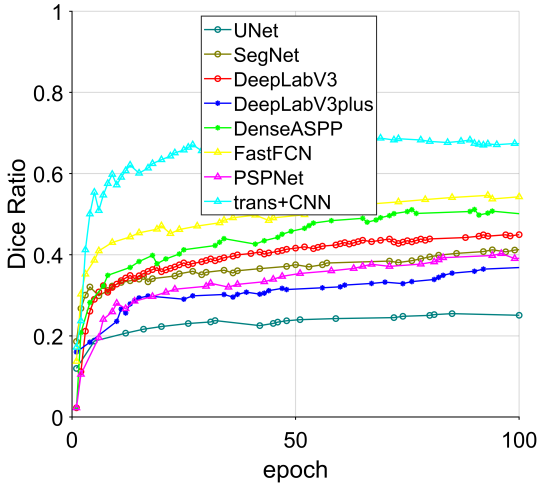


Fig. 10: Dice Ratio curves of different methods

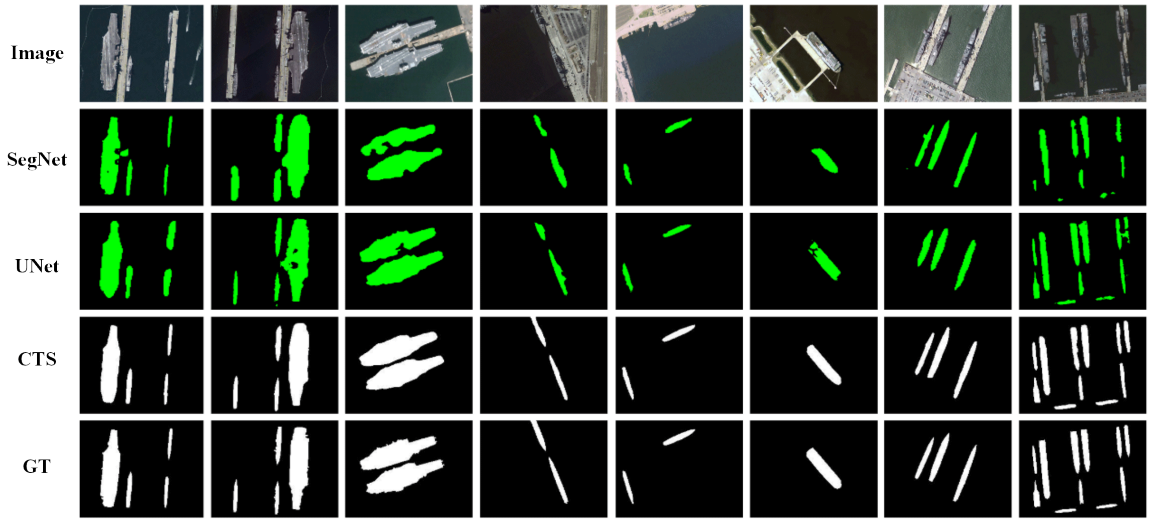


Fig. 9: Ship semantic segmentation performance comparison under different models

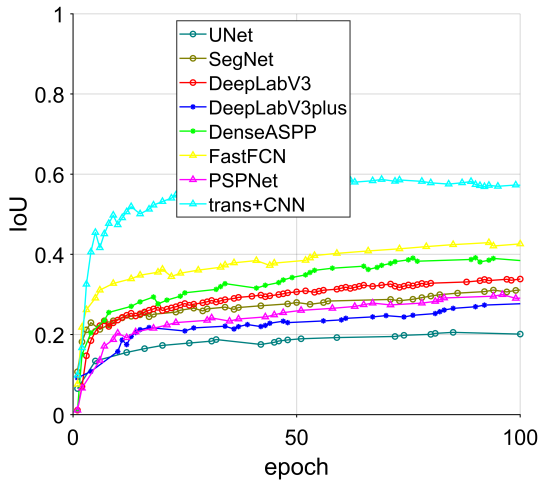


Fig. 11: IoU curves of different methods

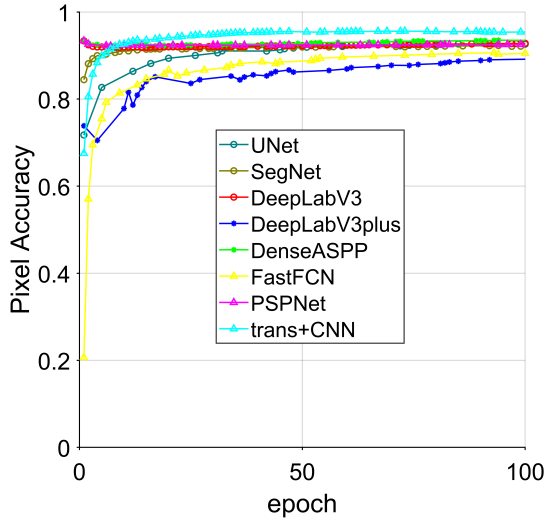


Fig. 12: Pixel Accuracy curves of different method

Compared to traditional CNN-based segmentation methods, CTS demonstrates superior performance in key aspects, including target completeness and edge accuracy. This is evidenced by higher Dice Ratio and Intersection over Union (IoU) values, as well as more accurate edge detection and target representation.

In remote sensing ship images, ship targets often appear partially occluded or incomplete, complicating accurate identification and analysis. To address this, advanced segmentation techniques can break down targets into manageable segments, improving tracking accuracy and core position determination.

Figure 13 demonstrates that the proposed method outperforms the original ground truth labels in recognizing cropped targets (as shown in subfigures (a), (b), and (c)) and ensuring target completeness (as shown in subfigures (d), (e), and (f)).

During the training process, the training accuracy curve continued to increase after 100 epochs. To further optimize the model's performance, we increased the number of epochs to 200 to allow for more thorough convergence. The experimental results are shown in Tables IV and V.

TABLE IV: Performance of segmentation results when epoch is 100

| Method | Dice Ratio | IoU | Pixel Accuracy |
|---|---|---|---|
| UNet | 0.2524 | 0.2031 | 92.72% |
| SegNet | 0.4159 | 0.3139 | 90.20% |
| PSPNet | 0.3957 | 0.2944 | 92.59% |
| DenseASPP | 0.5001 | 0.3836 | 93.48% |
| DeepLabV3 | 0.4528 | 0.3415 | 92.84% |
| DeepLabV3plus | 0.3694 | 0.2776 | 89.19% |
| CTS | **0.6722** | **0.5707** | **95.38%** |

After extending the training period to 200 epochs, the model's loss gradually stabilized, and performance metrics such as Dice Ratio and IoU showed improvement.

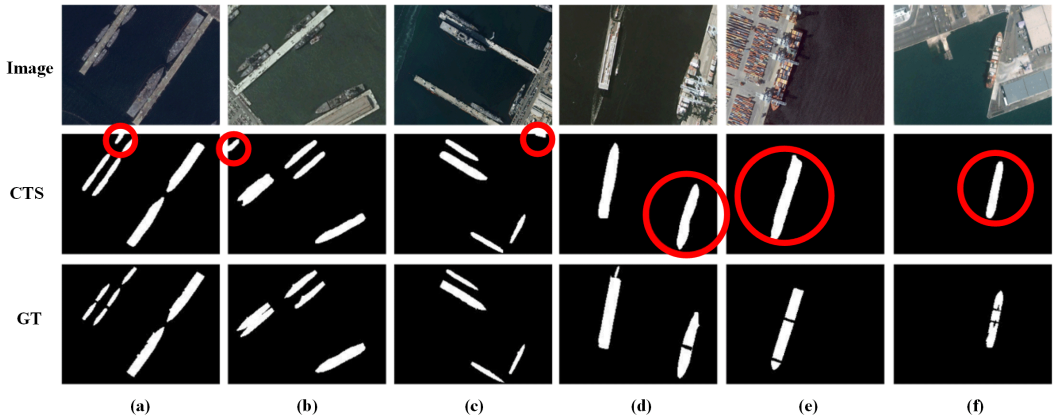


Fig. 13: Ship semantic segmentation performance in details

TABLE V: Performance of segmentation results when epoch is 200

| Method | Dice Ratio | IoU | Pixel Accuracy |
|---|---|---|---|
| UNet | 0.3190 | 0.2661 | 94.15% |
| SegNet | 0.5311 | 0.4292 | 93.84% |
| PSPNet | 0.4785 | 0.2728 | 93.43% |
| DenseASPP | 0.5103 | 0.4008 | 93.38% |
| DeepLabV3 | 0.4747 | 0.3681 | 92.84% |
| DeepLabV3plus | 0.5153 | 0.4143 | 92.28% |
| CTS | **0.6725** | **0.5652** | **95.26%** |

Overall, CTS outperforms other cutting-edge techniques. Specifically, our method achieves 1.1% higher Pixel Accuracy than the previous best method, along with substantial gains of approximately 0.14 in both Dice Ratio and IoU. These quantitative improvements indicate that our method is more effective and robust in complex ship segmentation tasks.

## V. Conclusion

To efficiently identify remote sensing ship targets in complex environments, this paper combines the advantages of CNNs and Transformers to propose a novel detection method calls CTS. The CNN channel feature extraction is based on specific layers of VGG19. Experimental results indicate that this method achieves high detection performance and segmentation accuracy, as measured by Dice Ratio and IoU, on ship datasets. Although the algorithm performs well, there is still potential for enhancement due to the intricate ship backgrounds, a wide range of classes and angles, as well as difficulties in accurately detecting ships located near shores or on land. Future research will focus on developing more precise detection methods, incorporating advanced data augmentation, multi-scale feature fusion, and optimized real-time processing to handle diverse maritime environments.

## References

[1] J. Zhou, P. Jiang, A. Zou, et al, "Ship Target Detection Algorithm Based on Improved YOLOv5," *Journal of Marine Science and Engineering*, vol.9, no.8, p.908, 2021.

[2] T. Liu, Z. Zhang, Z. Lei, et al, "An Approach to Ship Target Detection Based on Combined Optimization Model of Dehazing and Detection," *Engineering Applications of Artificial Intelligence*, vol. 127, p.107332, 2024.

[3] Y. X. Geng, L. Wang, Y. G. Wang, "Large Kernel Disassembling Attention Mechanism for Remote Sensing Object Detection," *IAENG International Journal of Computer Science*, vol. 51, no. 9, pp1367-1373, 2024.

[4] Q. K. Zhou, W. Zhang, J. D. Li, et al, "Ship Classification Detection Method of Optical Remote Sensing Image Based on Improved YOLOv5s," *Laser & Optoelectronics Progress*, vol. 59, no. 16, pp476-483, 2022.

[5] T. Zhang, X. G. Yang, X. Q. Lu, et al, "Ship Target Detection in Dense RFB and LSTM Remote Sensing Images," *National Remote Sensing Bulletin*, vol. 26, no.9, pp1859-1871, 2022.

[6] X. Chen, X. Wu, D. K. Prasad, et al, "Pixel-wise Ship Identification from Maritime Images via a Semantic Segmentation Model," *IEEE Sensors Journal*, vol. 22, no. 18, pp18180-18191, 2022.

[7] J. Long, E. Shelhamer, T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp3431-3440, 2015.

[8] O. Ronneberger, P. Fischer, T. Brox, "U-net: Convolutional Networks for Biomedical Image Segmentation," *Springer International Publishing*, pp234-241, 2015.

[9] V. Badrinarayanan, A. Kendall, R. Cipolla, "Segnet: a Deep Convolutional Encoder-decoder Architecture for Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no.12, pp2481-2495, 2017.

[10] H. Zhao, J. Shi, X. Qi, et al, "Pyramid Scene Parsing Network," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp2881-2890, 2017.

[11] L. C. Chen, G. Papandreou, I. Kokkinos, et al, "Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected Crfs," *International Conference on Learning Representations*, p.1412.7062, 2015.

[12] L. C. Chen, G. Papandreou, I. Kokkinos, et al, "Deeplab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected Crfs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp834-848, 2017.

[13] L. C. Chen, G. Papandreou, F. Schroff, et al, "Rethinking Atrous Convolution for Semantic Image Segmentation," *ArXiv Preprint ArXiv*, no. 5, p.1706.05587, 2017.

[14] J. Xiao, Z. Wang, "DeepLabv3+: Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp1-5, 2019.

[15] G. Chen, X. Zhang, Q. Wang, et al, "Symmetrical Dense-shortcut Deep Fully Convolutional Networks for Semantic Segmentation of Very-high-resolution Remote Sensing Images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 5, pp1633-1644, 2018.

[16] L. Tian, X. Zhong, M. Chen, et al, "Semantic Segmentation of Remote Sensing Image Based on Gan and Fcn Network Model," *Scientific Programming*, no. 1, p.9491376, 2021.

[17] M. Yasir, L. Zhan, S. Liu, et al. "Instance Segmentation Ship Detection Based on Improved Yolov7 using Complex Background SAR Images," *Frontiers in Marine Science*, vol. 10, p.1113669, 2023.

[18] W. Wang, Y. Fu, F. Dong, et al, "Semantic Segmentation of Remote Sensing Ship Image via a Convolutional Neural Networks Model," *IET Image Processing*, vol. 13, no. 6, pp1016-1022, 2019.

[19] Z. Fan, J. Hou, Q. Zang, et al, "River Segmentation of Remote

Sensing Images Based on Composite Attention Network," *Complexity*, p.7750281, 2022.
[20] A. Vaswani, N. Shazeer, N. Parmar, et al, "Attention Is All You Need," *Corr*, p.1706.03762, 2017.
[21] X. Yu, J. Tao, X. Yunjiong, et al, "Transformer Image Recognition System Based on Deep Learning," *International Conference on Systems*, pp595-599, 2019.
[22] D. Chen, H. Hsieh, T. Liu, "Adaptive Image Transformer for One-shot Object Detection," *Computer Vision and Pattern Recognition*, pp12247-12256, 2021.
[23] Q. Sun, N. Fang, Z. Liu, et al, "Hybridctrm: Bridging CNN and Transformer for Multimodal Brain Image Segmentation," *Journal of Healthcare Engineering*, p.7467261, 2021.
[24] X. Xu, G. Li, G. Xie, et al, "Weakly Supervised Deep Semantic Segmentation Using Cnn and Elm with Semantic Candidate Regions," *Complexity*, p.9180391, 2019.
[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp770C778, 2016.
[26] S. Karen, "Very Deep Convolutional Networks for Large-scale Image Recognition," *ArXiv Preprint ArXiv*, p.1409.1556, 2014.
[27] Z. Hu, H. Wu, L. He, "A Neural Network for EEG Emotion Recognition that Combines CNN and Transformer for Multi-scale Spatial-temporal Feature Extraction," *IAENG International Journal of Computer Science*, vol. 51, no. 8, pp1094-1104, 2024.
[28] Y. Wang, T. Ni, "Remote Sensing Image Semantic Segmentation Algorithm Based on Improved Enet Network," *Scientific Programming*, p.5078731, 2021.
[29] H. Zhuang, W. Liu, " Underwater Biological Target Detection Algorithm and Research Based on YOLOv7 Algorithm," *IAENG International Journal of Computer Science*, vol. 51, no. 6, pp594-601, 2024.