Onet: Twin U-Net Architecture for Unsupervised Binary Semantic Segmentation in Radar and Remote Sensing Images

Yi Zhou, Hang Su, Tian Wang, Qing Hu

***Abstract*—Segmenting objects from cluttered backgrounds in single-channel images, such as marine radar echoes, medical images, and remote sensing images, poses significant challenges due to limited texture, color information, and diverse target types. This paper proposes a novel solution: the Onet, an O-shaped assembly of twin U-Net deep neural networks, designed for unsupervised binary semantic segmentation. The Onet, trained with an intensity-complementary image pair and without the need for annotated labels, maximizes the Jensen-Shannon divergence (JSD) between the densely localized features and the class probability maps. By leveraging the symmetry of U-Net, Onet subtly strengthens the dependence between dense local features, global features, and class probability maps during the training process. The design of the complementary input pair aligns with the theoretical requirement that optimizing JSD needs the class probability of negative samples to accurately estimate the marginal distribution. Compared to the current leading unsupervised segmentation methods, the Onet demonstrates superior performance in target segmentation in marine radar frames and cloud segmentation in remote sensing images. Notably, we found that Onet's foreground prediction significantly enhances the signal-to-noise ratio (SNR) of targets amidst marine radar clutter. Onet's source code is publicly accessible at https://github.com/joeyee/Onet.**

***Index Terms*—unsupervised semantic segmentation, binary semantic segmentation, marine radar object detection, sea clutter segmentation, cell segmentation, cloud segmentation, maximize mutual information, Onet.**

I. Introduction

Semantic segmentation is the process of assigning meaningful labels (such as 'sky', 'land', or 'road') to each pixel in an image. A particular subtype of this task is binary semantic segmentation, which involves categorizing each pixel into one of two classes: the foreground (target of interest) or the background (noise or clutter) [1]. Serving as a fundamental component for other advanced computer vision applications, binary semantic segmentation enables tasks such as cell tracking in medical images [2], object detection and tracking in radar echoes [3], and cloud detection in remote sensing images [4], among others. With the rise of deep neural networks (DNNs), supervised methods [5, 6] have become the popular approaches for binary semantic segmentation. However, these supervised approaches are often hampered by the need for extensive, domain-specific pixel-level annotation, a process that can be costly and time-consuming [7]. Therefore, unsupervised semantic segmentation [8, 9, 10, 11, 12, 13, 14], which eliminates the need for labeling, has become an attractive alternative in binary semantic segmentation. The core concept of unsupervised semantic segmentation with DNNs is to learn the common features within the same category while ensuring they remain discriminative for different categories. Specific loss functions are employed in the form of maximizing mutual information [10, 12] or contrastive learning [11]. These approaches typically train models on the curated image sets by applying various augmentations to the same category [8, 9] or organizing samples from different categories within a batch [13, 10].

However, in the context of binary semantic segmentation, especially when dealing with highly variable foreground targets (e.g., diverse nuclei shapes in microscopy images or diffused point targets in marine radar), this strategy becomes less practical. The visual diversity and potential low resolution of foreground objects make it difficult to emphasize common features across augmented versions. Moreover, compared to the foreground, the background exhibits a more stable amplitude distribution with relatively low intensity. Examples include terrains in remote sensing, dark backgrounds in microscopy, and thermal noise in radar. These distinctions between foreground and background suggest that binary segmentation algorithms should shift their focus from emphasizing similarities among foreground targets through augmentation to distilling discriminating features between the foreground and the background.

Since the primary distinction between them is the intensity, a normalized image X highlights the foreground, while its reverse, $1 - X$, emphasizes the background. By feeding the network with the complementary input pair $(X, 1 - X)$, the features of the foreground and background can be contrastively learned within the same feature space. Considering that shallow convolutional networks capture local intensity variations [15], while deeper layers extract global features [16], it is beneficial to connect the shallow and deep layers directly to emphasize the influence of intensity on classification. This idea aligns well with the U-Net architecture [17], which uses skip connections to integrate features between the encoding and decoding layers. Motivated by these insights, we utilize the initial layer for local feature expression and the final layer for global feature extraction in the U-Net network (see Fig.1). The class probability maps are then generated by projecting

Yi Zhou, and Qing Hu are with the Department of Electronic Information Engineering, Dalian Maritime University, Dalian, 116026, China. Email: ({yi.zhou, hq0518}@dlmu.edu.cn). Hang Su is with the Department of Computer Science and Technology, Tsinghua University, Beijing, 100084, China. Email:(suhangss@mail.tsinghua.edu.cn). Tian Wang is with the School of Artificial Intelligence, Beihang University and is also with Zhongguancun Laboratory, Beijing, 100083, China. Email:(wangtian@buaa.edu.cn).

the global features onto the local features at the corresponding locations. The segmentation probability is expected to heavily depend on the numerical distributions of the local features. To strengthen this dependence, we propose to maximize the mutual information between local features and segmentation scores by raising its lower bound, the Jensen-Shannon Divergence (JSD) [18, 10]. To effectively handle the complementary input pairs, we propose a dual U-Net architecture which is called Onet in this paper due to its O-shaped structure.

The key contributions of our work are summarized as follows:

1) We have developed an O-shaped unsupervised framework for binary semantic segmentation. Through experiments with multi-modal images, we demonstrate that the traditionally supervised learning-oriented U-Net structure is also effective within our unsupervised framework.
2) Onet leverages numerically complementary pairs at the pixel level to naturally create negative samples for estimating the marginal distribution in the computation of the Jensen-Shannon Divergence (JSD). This approach utilizes the adversarial power of the input to guide the network in learning distinguishable features effectively.
3) In experiments involving object segmentation of both simulated and real radar echoes, we observed that Onet's foreground prediction enhances the signal-to-noise ratio (SNR) of targets amidst clutter. This demonstrates Onet's effective image denoising capability in low SNR scenarios.

II. Related works

A. U-Net

The U-Net was proposed to address semantic segmentation in medical images with a U-shaped convolutional neural network [17, 19]. The U-shaped architecture consists of two symmetric contracting (encoding) and expansive (decoding) paths with skip connections. One of the outstanding features of U-Net is that it can generate highly detailed segmentation maps with limited training samples. Following this elegant structure, U-Net and its variations [20, 21, 22, 23] have achieved great success in wide applications of semantic segmentation. Recently, with the growth of graphic memories, the ensemble of multiple U-Nets has shown advances in the precision of segmentation. One approach is cascading two or more U-Nets together [24]. It uses U-Nets to detect objects in different sizes step by step, e.g., the first U-Net segments a particular organ and the second U-Net detects the tumors within that organ. Another approach ensembles multiple U-Nets in parallel [25, 26]. In [26, 25], siamese U-Net adopts two contracting paths with shared weights and one expansive path to measure the dissimilarity of the individual medical images and the normal template. It encourages the network to learn the features of annotated regions that are different from the normal template in the supervised manner. Our architecture is close to the Siamese U-Net in the spirit of using paralleled branches of U-Net. In contrast, we use U-Net backbone to learn the features of adversarial intensities in an unsupervised paradigm.

B. Unsupervised Semantic Segmentation via Mutual Information Maximization

Unsupervised DNNs for semantic segmentation are commonly trained by maximizing mutual information or contrastive learning. The sources of the information can be extracted in several ways: between the two class-probability vectors of paired samples [8, 9], between feature maps of different layers [10, 12], or between queries and keys [11]. Despite these methods employing different objective functions with varied inputs, three crucial elements impact their performance of semantic segmentation: (1) the use of augmented inputs [8], (2) local feature representation [8, 10], and (3) the inclusion of negative samples in a training batch [13, 12, 27].

IIC [8] is the landmark paper to address unsupervised semantic segmentation by maximizing the mutual information between two class assignment outputs of DNN, which uses the source image and its augmented form as the input pair. IIC encourages the network to distill the common features of the pair and discard the instance-specific features of each sample. To numerically compute the joint distribution of two classes, IIC averages the probability tensor over local patches and perturbation space within each image. Based on the framework of IIC, [9] proposed to use the diverse masked convolution to improve the quality of patch-level transformation.

Later Harb and Knöbelreiter [10] pointed out the limitation of the IIC: segmentation depends only on local features. It affects the sharpness of the segmentation since similar local features would be shared among objects with different types. Their approach named Infoseg represents multiple semantically meaningful global features in high-dimensional vectors and utilizes the class-probability tensor to weight and expand the vectorized features. The objective of the training is to maximize the mutual information between the local features and the weighted global features. With the introduction of the global features, Infoseg gets superior performance than those approaches which only use the local feature [8, 9] in the dataset of Potsdam and COCO-Persons [10].

Our work is motivated by [10], but there are two major differences.

(1) Infoseg enhances global features, which are represented as high-dimensional vectors (1024 dimensions by default) using a customized ResNet [28]. To align with the global features, it projects dense local features and score maps to the same high-dimensional space, which consumes substantial memory. Infoseg has to represent local features in a down-sampled size to save memory in practice, but this compromises segmentation granularity, as the interpolation causes blur when recovering the output mask back to the original input size. In contrast, Onet extracts dense local features L and global features H directly from the front and end layers of U-Net, respectively. These feature maps are tensors with the same dimensionality (default 64 channels) at full image size, maintaining the granularity of semantic segmentation.

(2) Unlike Infoseg, where the marginal distribution of negative samples is computed by random sampling in a batch, Onet guarantees a negative sample by taking the complementary pair $(X, 1-X)$ as input. This strategy is crucial for improv-

ing segmentation performance. More details are discussed in ablation study section V-F1.

III. Preliminary: Background of Mutual Information Neural Estimator (MINE)

Mutual information measures the non-linear statistical dependencies between random variables. It is equal to the Kullback-Leibler (KL) divergence between the joint distribution $\mathbb{P}_{\mathrm{xy}}$ and the product of the marginals $\mathbb{P}_{\mathrm{x}}$ and $\mathbb{P}_{\mathrm{y}}$ in the form [29],

$$\begin{aligned} I\left(\mathrm{X};\mathrm{Y}\right) &= D_{\mathrm{KL}}\left(\mathbb{P}_{\mathrm{xy}} \parallel \mathbb{P}_{\mathrm{x}} \otimes \mathbb{P}_{\mathrm{y}}\right) \\ &= \int_{\mathrm{xy}} p\left(x,y\right) \log \frac{p\left(x,y\right)}{p\left(x\right)p\left(y\right)} dxdy, \end{aligned} \tag{1}$$

where x and y are samples from random variables X and Y, respectively. In [30] KL divergence D_{KL} is defined as one member of f-divergence family:

$$D_{\mathrm{KL}}\left(\mathbb{P}_{\mathrm{xy}} \parallel \mathbb{P}_{\mathrm{x}} \otimes \mathbb{P}_{\mathrm{y}}\right) = \int_{\mathrm{xy}} p\left(x\right)p\left(y\right) f_1\left(u\right) dxdy, \tag{2}$$

where $f_1\left(u\right) = u\log(u)$ and $u = p(x,y)/p(x)p(y)$. Now, change the function form to $f_2(u) = -(u+1)\log\frac{1+u}{2} + u\log u$, we have Jensen-Shannon divergence D_{JS} and its associated mutual information I_{JS} as follows:

$$\begin{aligned} I_{\mathrm{JS}}\left(\mathrm{X};\mathrm{Y}\right) &= D_{\mathrm{JS}}\left(\mathbb{P}_{\mathrm{xy}} \parallel \mathbb{P}_{\mathrm{x}} \otimes \mathbb{P}_{\mathrm{y}}\right) \\ &= \int_{\mathrm{xy}} p\left(x\right)p\left(y\right) f_2\left(\frac{p\left(x,y\right)}{p\left(x\right)p\left(y\right)}\right) dxdy \\ &= \int_{\mathrm{xy}} p(x,y)\log\frac{2p(x,y)}{p(x,y)+p(x)p(y)} \\ &\quad + p(x)p(y)\log\frac{2p(x)p(y)}{p(x,y)+p(x)p(y)} dxdy, \end{aligned} \tag{3}$$

It has been proven that $I \geq 2I_{\mathrm{JS}}$ [31] and I_{JS} has a variational lower bound [30] that can be approximated by a deep neural network [18] in the form:

$$\begin{aligned} I_{\mathrm{JS}} \geq \hat{I}_\theta =& \mathbb{E}_{(x,y)\sim\mathbb{P}_{\mathrm{xy}}}\left[-\log\left(1+e^{-T_\theta(x,y)}\right)\right] \\ & - \mathbb{E}_{(x,y)\sim\mathbb{P}_{\mathrm{x}}\otimes\mathbb{P}_{\mathrm{y}}}\left[\log\left(1+e^{T_\theta(x,y)}\right)\right]. \end{aligned} \tag{4}$$

Here, a deep neural network T_θ with parameters θ approximates the underlying function $T : \mathrm{X}\times\mathrm{Y} \to \mathrm{R}$. Therefore maximizing mutual information I can be solved by training the deep neural networks to maximize the lower bound $\hat{I}_\theta$ of I_{JS}, which is called the Mutual Information Neural Estimator (MINE) in [18].

In [12], MINE is applied to image classification, where the objective is to maximize the Jensen-Shannon mutual information I_{JS} between the local and the global feature distribution as follows:

$$\begin{aligned} I_{\mathrm{JS}} \geq \hat{I}_{\theta,\phi} =& \mathbb{E}_{\mathbb{P}_{\mathrm{LH}}}\left[-\log\left(1+e^{-T_\theta(L_\phi(X),H_\phi(X))}\right)\right] \\ & - \mathbb{E}_{\mathbb{P}_{\mathrm{L}}\otimes\mathbb{P}_{\mathrm{H}}}\left[\log\left(1+e^{T_\theta\left(L_\phi(X),H_\phi(X')\right)}\right)\right]. \end{aligned} \tag{5}$$

Here local spatial feature $L_\phi(X)$ and global feature $H_\phi(X)$ take the places of x, y in (4). Here $L_\phi(X)$ and $H_\phi(X)$ are the local spatial feature and global feature, respectively, which are derived from different branches of a deep neural network with parameters ϕ. Note that to generate a marginal distribution $\mathbb{P}_{\mathrm{H}}$ in the right side of (5), another negative input image X' (which is preferred to be different from X) is needed for computing $H_\phi(X')$ [12].

Inspired by the idea of representative global feature learning [12], Infoseg [10] represents the high-dimensional global feature H as a vector (default setup is 1024 dimensions) via a full-connect layer of a customized Resnet backbone. To fuse the information from the global and local features through the inner product, Infoseg needs to generate a local feature map L with very deep channels. The class probability map is generated by projecting each global feature onto every local dense feature spatially. To maintain the spirit of maximizing the I_{JS} between local and global features, Infoseg transforms the vectorized global feature into a dense global feature (named soft-global-feature assignment A) by:

$$A_{i,j}\left(X\right) = \sum_k P_{i,j,k} \cdot H_k\left(X\right), \tag{6}$$

where i, j denote the spatial location of the probability map of segmentation, and k is the index of the K classes. Since Infoseg represents high-dimensional global features as a vector, it requires the use of deep local and soft-assigned global features (1024 channels by default). To save graphical memory, Infoseg represents the local feature and score maps in a downsampled size (one-quarter of the full image size).

During training, Infoseg maximizes I_{JS} between the local feature L and A by (5), but uses a simple inner product function to replace the neural network T_θ as follows:

$$\begin{aligned} &\hat{I}_\psi\left(L_{i,j}\left(X\right), A_{i,j}(X)\right) = \\ &\quad \mathbb{E}_{\mathbb{P}_{\mathrm{LA}}}\left[-\log\left(1+e^{-L_{i,j}(X)\,\cdot\,A_{i,j}(X)}\right)\right] \\ &\quad - \mathbb{E}_{\mathbb{P}_{\mathrm{L}}\otimes\mathbb{P}_{\mathrm{A}}}\left[\log\left(1+e^{L_{i,j}(X)\,\cdot\,A_{i,j}(X')}\right)\right]. \end{aligned} \tag{7}$$

To estimate the marginal distribution $\mathbb{P}_{\mathrm{A}}$, Infoseg randomly selects one sample X' from a training batch as the negative sample. In fact, this approach may not always produce a true negative sample in the semantic sense.

When understanding the unsupervised framework of MINE, Three key questions arise for binary semantic segmentation: (1) Do we need such deep local features (e.g., more than 1000 channels) to align with the global feature vector? (2) Is it possible to directly maximize the mutual information between the dense local features and the class probability? (3)Can we find a better way to generate negative samples X' and estimate the marginal distribution in a more unbiased manner? To answer these questions, we propose the architecture of Onet in the following section.

IV. Architecture

Onet consists of dual U-Nets (see Fig.1). The bottom U-Net and reversed top U-Net form a symmetric structure. Both them follow the same structure as the standard U-Net [17], consisting of a contracting path (which applies sequential convolution and max-pooling to shrink the spatial dimensions and deepen the channels) and an expansive path (which applies sequential up-convolution and concatenation to expand

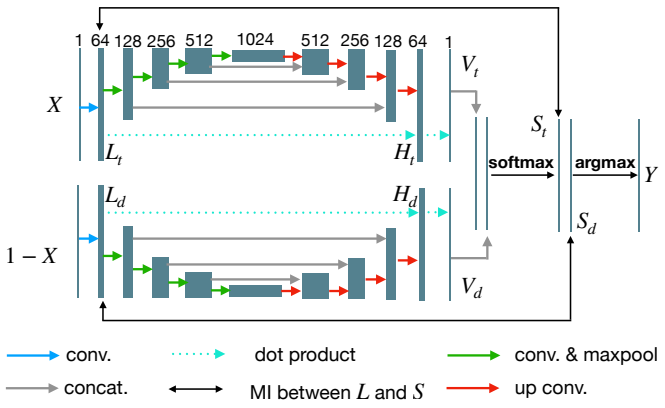

Fig. 1: The architecture of Onet. Two symmetric U-Nets (top and down) are assembled to form the 'O' shape ring. Complementary one-channel images X and $1-X$ are fed into the respective U-Nets. The numbers at the top indicate the tensor flows, with increasing channels from 1 to 1024 in the contracting path and decreasing from 1024 to 1 in the expansive path of each U-Net. The dot product between the front local feature L_t and the correspondent global feature H_t generates a prediction score V_t for one class. In the same manner, a prediction score V_d for another class is generated by the bottom U-Net. After concatenating the two V maps and applying softmax, we obtain the probability maps S_t and S_d for the two classes. The argmax operation on the class probabilities generates the binary segmentation mask Y. Onet is trained in an unsupervised manner by maximizing the mutual information $\hat{I}(L_t, S_t)$ and $\hat{I}(L_d, S_d)$.

the spatial dimensions and reduce the channels). In Fig. 1, arrows of different types and colors represent various tensor operations. The input X to top U-Net's is normalized to the range $[0, 1]$. Its complementary $1-X$ is fed into the bottom U-Net. In Fig.1 the subscript d and t represent the items from the bottom and top U-Net respectively. It is worth noting that the top and bottom U-Nets can be implemented either as twin U-Nets (denoted as Onet-TW) or as a single U-Net in weight-sharing mode (denoted as Onet-WS). The performance and effects of these two variants are discussed in Subsection V-F2.

The key difference between Onet and U-Net lies in the operation at the output end. Instead of the standard concatenation and convolution in U-Net, Onet projects the local feature L onto the global feature H using a dot product operation at each spatial location (i.e. $V = \langle L, H \rangle$). The two projection maps, V_t and V_d represent the prediction scores for the two classes, respectively. Onet randomly determine whether V_t or V_d corresponds to the foreground score based on the network's initialization. The two score maps are then concatenated and passed through a softmax function:

$$S_t = \frac{\exp\left(\langle L_t, H_t \rangle\right)}{\exp\left(\langle L_t, H_t \rangle\right) + \exp\left(\langle L_d, H_d \rangle\right)}. \tag{8}$$

Here, S_t denotes the probability map for each pixel in one class. By changing the subscript to d in the numerator of (8), we obtain S_d, where $S_d = 1 - S_t$. The final binary mask Y is obtained by applying the argmax operation on the concatenated S_t and S_d. Since Y has the same size as the input X, the detection threshold for the foreground is set to greater than 0.5 (i.e. the foreground probability exceeds the corresponding background probability). Finally, the top U-Net is trained by maximizing $\hat{I}(L_t, S_t)$ as in (7) which is formulated as:

$$\begin{aligned}\hat{I}\left(L_t(X), S_t(X)\right) = {} & \mathbb{E}_{\mathbb{P}_{\mathrm{L_t S_t}}}\left[-\log\left(1 + e^{-L_t(X)\cdot S_t(X)}\right)\right] \\ & - \mathbb{E}_{\mathbb{P}_{\mathrm{L_t}}\otimes\mathbb{P}_{\mathrm{S_t}}}\left[\log\left(1 + e^{L_t(X)\cdot S_t(X')}\right)\right].\end{aligned} \tag{9}$$

In (9) the spatial position index i, j of L and S are omitted for simplicity. To compute the top U-Net's $\hat{I}$, we need the prediction map $S_t(X')$ of the negative sample (X') to form the unbiased distribution $\mathbb{P}_{\mathrm{S_t}}$. In the context of binary segmentation, $1-X$ is a natural negative sample for X. By feeding them into Onet, the symmetric structure would directly generate the negative sample's class probability map. Therefore, we can use $S_d(1-X)$ from the bottom U-Net to replace $S_t(X')$. Now, (9) can be rewritten as

$$\begin{aligned}\hat{I}\left(L_t(X), S_t(X)\right) = {} & \mathbb{E}_{\mathbb{P}_{\mathrm{L_t S_t}}}\left[-\log\left(1 + e^{-L_t(X)\cdot S_t(X)}\right)\right] \\ & - \mathbb{E}_{\mathbb{P}_{\mathrm{L_t}}\otimes\mathbb{P}_{\mathrm{S_t}}}\left[\log\left(1 + e^{L_t(X)\cdot S_d(1-X)}\right)\right].\end{aligned} \tag{10}$$

Similarly, approximation of the down U-Net's $\hat{I}(L_d, S_d)$ is facilitated by the score map from the top U-Net. Assuming Onet's parameter set is denoted as ψ, the JSD of Onet is represented as:

$$\begin{aligned}\hat{I}_\psi(L, S) = {} & \hat{I}\left(L_t(X), S_t(X)\right) \\ & + \hat{I}\left(L_d(1-X), S_d(1-X)\right) \\ = {} & \mathbb{E}_{\mathbb{P}_{\mathrm{L_t S_t}}}\left[-\log\left(1 + e^{-L_t(X)\cdot S_t(X)}\right)\right] \\ & - \mathbb{E}_{\mathbb{P}_{\mathrm{L_t}}\otimes\mathbb{P}_{\mathrm{S_t}}}\left[\log\left(1 + e^{L_t(X)\cdot S_d(1-X)}\right)\right] \\ & + \mathbb{E}_{\mathbb{P}_{\mathrm{L_d S_d}}}\left[-\log\left(1 + e^{-L_d(1-X)\cdot S_d(1-X)}\right)\right] \\ & - \mathbb{E}_{\mathbb{P}_{\mathrm{L_d}}\otimes\mathbb{P}_{\mathrm{S_d}}}\left[\log\left(1 + e^{L_d(1-X)\cdot S_t(X)}\right)\right].\end{aligned} \tag{11}$$

The training objective is to find the parameter ψ^* that maximizes $\hat{I}_{\psi^*}$ across the parameter space.

V. Experiment

A. Implementation Details

We use CNN-based U-Net as the baseline to evaluate Onet's performance. The single U-Net has 4 down-convolution layers and 4 up-convolution layers. Each layer applies two consecutive convolutions with a kernel size of 3 (padding=1), followed by batch normalization. Onet is tested in three experiments: (1) extended target segmentation in simulated clutter, (2) target segmentation in real clutter from marine radar echoes, and (3) cloud detection in remote sensing images. For training Onet on the simulated clutter datasets, we set the batch size to 10 and use the Adam optimizer with a fixed learning rate of 3e-6, along with $\beta_1 = 0.9$ (exponential decay rate for

the first moment) and $\beta_2 = 0.999$ (for the second moment). For the cloud segmentation experiment, we use a similar Adam optimizer. However, we incorporate a cosine annealing schedule over 300 epochs, which allows the learning rate to decay from an initial value of 1e-4 to 1e-6. Here, the batch size is set to be 5. Data augmentation techniques such as flipping, rotation, distortion, and random brightness adjustments are applied to the training set of the cloud dataset. Training for both experiments is conducted over 300 epochs, with a total of 13k and 15k iterations for parameter updates, respectively. Since there are very few rain clutter images, we reuse the model of Onet trained on the simulated datasets and test it directly on the radar images with rain echoes. Before diving into the details of the experiments and ablation tests, we first discuss the evaluation metrics.

B. Evaluation Metrics

In our experiments, we utilize Mean Intersection over Union (mIoU) and Overall Accuracy (OA) to evaluate the performance of binary semantic segmentation models, as proposed in [10, 8, 32]. For extended target segmentation in marine radar, the model's ability to segment the foreground target from the noisy background is of primary interest. Therefore, we also measure the detection rate P_d and false alarm rate P_{fa}, which are discussed in the experiments of extended target segmentation in both simulated and real marine radar echoes. These four metrics are expressed as follows:

$$\text{mIoU} = \sum_{i=0}^{1} \text{IoU}_i/2, \tag{12}$$

where IoU denotes the Intersection over Union, and the index i represents 0 (background) or 1 (foreground target). Given the predicted foreground mask Y and ground truth mask M, both with logic values (false for background and true for target), the IoU_i is defined as:

$$\text{IoU}_1 = \frac{|\text{Y} \cap \text{M}|}{|\text{Y} \cup \text{M}|} \quad \text{or} \quad \text{IoU}_0 = \frac{|(1-\text{Y}) \cap (1-\text{M})|}{|(1-\text{Y}) \cup (1-\text{M})|}. \tag{13}$$

Here $|\cdot|$ denotes the cardinality (the number of 'true' value) in the referred set. Let TP, TN, FP, and FN represent True Positive, True Negative, False Positive, and False Negative, respectively. They are computed as follows:

$$\begin{aligned} \text{TP} &= |\text{Y} \cap \text{M}|, \\ \text{TN} &= |(1-\text{Y}) \cap (1-\text{M})|, \\ \text{FP} &= |\text{Y} \cap (1-\text{M})|, \\ \text{FN} &= |(1-\text{Y}) \cap \text{M}|. \end{aligned} \tag{14}$$

From these, the OA, Detection Rate P_d and False Alarm Rate P_{fa} are computed as:

$$\begin{aligned} \text{OA} &= \frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}}, \\ P_d &= \frac{\text{TP}}{|\text{M}|}, \\ P_{fa} &= \frac{\text{FP}}{|1-\text{M}|}. \end{aligned} \tag{15}$$

When true labels are evenly distributed in the images, the higher OA and mIoU genreally indicate better semantic segmentation performance. However, if one label dominates the majority of pixels, high OA does not necessarily indicate good segmentation performance. For example, if clutter occupies 98% of the frame (i.e., TN =0.98), a model that fails to detect the target (i.e., FP $= 0$, TP $= 0$ and FN $= 0.02$) can still achieve a high OA (98%). In radar target segmentation experiments, the model that achieves a higher P_d with a reasonable P_{fa} (i.e., within the same order of magnitude) is preferred.

C. Extended Object Segmentation in Rayleigh-Distributed Clutter

In the field of marine radar, object segmentation aims to mark target pixels in a cluttered background. This goal is similar to that of binary semantic segmentation, but with a different strategy. Traditionally, it is assumed that the radar operates in a homogeneous water environment, where the intensity of the background clutter follows a stable random distribution. Once the statistical model of the clutter is known, it is easy to filter out targets with a Constant False Alarm Rate (CFAR) by setting an appropriate intensity threshold [33, 34]. Specifically, if the clutter follows a Rayleigh distribution, the mean μ and variance of the clutter can be directly measured in a local area. By multiplying the mean μ by a constant coefficient k, the threshold value for clutter filtering is defined as $\nu = k\mu$. Pixels with intensities greater than ν are marked as targets. Otherwise, they are labeled as clutter. CFAR is popular for its simplicity in numerical calculations. In most cases, it works well when the target of interest (e.g., ship, island, buoy) reflects strong or reinforced radio waves. However, CFAR struggles when the background clutter intensity is comparable to the target.

In this subsection, we compare the Onet model with CFAR for extended target segmentation in the Rayleigh-distributed clutter. We simulate Rayleigh-distributed clutter in frames and embed multiple extended targets (see Fig.2) with varying Peak-Signal-to-Noise Ratio (PSNR). Target template are generated using 2d elliptical Gaussian function to imitate the energy diffusion of the ship's echoes in marine radar signal, as discussed in [35, 36]. For the simulation, we consider a total of 11 PSNR levels ranging from 0 to 10. For each PSNR, we randomly samples 150 frames. In each frame, 20 targets are scattered at random positions, with target widths ranging from 8 to 14 pixels and heights between14 and 22 pixels. This setup results in a dataset of 1650 frames and 33000 labeled targets, spanning from the very low PSNR=0 to the high PSNR=10. The training dataset consists of 450 frames with PSNR levels of 0, 1, 2 (90% of them for training and 10% frames for model validation). For each PSNR level, the test dataset is composed of 10% of the corresponding 150 frames. A low PSNR indicates that the background clutter's intensity is comparable to that of the target, providing a greater challenge for the model to distinguish the target from the clutter. The ablation experiment regarding the selection train datasets is discussed in the supplementary materials of the author's Onet repository on Github.

TABLE I: Performance of target segmentation in the simulated Rayleigh clutter under different PSNRs.

PSNR	Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
0	IIC	0.8420	0.4313	0.1623	0.1434
	Infoseg	0.6671	0.3438	0.3747	0.3266
	CFAR	0.5378	0.2810	0.7121	0.4659
	Onet-TW	0.5850	0.3060	**0.8604**	0.4196
	Onet-WS	**0.9512**	**0.4863**	0.0649	**0.0339**
2	IIC	0.8366	0.4259	0.1266	0.1486
	Infoseg	0.5452	0.2760	0.1844	0.4470
	CFAR	0.5431	0.2854	0.7975	0.4624
	Onet-TW	0.5940	0.3128	**0.9653**	0.4123
	Onet-WS	**0.9547**	**0.5129**	0.2091	**0.0327**
5	IIC	0.8285	0.4206	0.1123	0.1566
	Infoseg	0.5555	0.2853	0.4171	0.4416
	CFAR	0.5519	0.2912	0.8779	0.4550
	Onet-TW	0.6351	0.3361	**0.9996**	0.3711
	Onet-WS	**0.9688**	**0.6024**	0.5838	**0.0247**
10	IIC	0.8211	0.4142	0.0663	0.1632
	Infoseg	0.9712	0.6885	0.9560	0.0285
	CFAR	0.5647	0.2996	0.9530	0.4435
	Onet-TW	0.9777	0.7029	**0.9997**	0.0227
	Onet-WS	**0.9936**	**0.8052**	0.6222	**0.0001**
0-10 ave	IIC	0.8297	0.4214	0.1105	0.1551
	Infoseg	0.7423	0.4388	0.6306	0.2553
	CFAR	0.5516	0.2909	0.8642	0.4550
	Onet-TW	0.7282	0.4192	**0.9763**	0.2760
	Onet-WS	**0.9720**	**0.6384**	0.4489	**0.0192**

The best metrics in each PSNR setting are marked in bold blue. Onet-WS denotes that Onet works in weight-share mode via a single U-Net. Onet-TW means the Onet contains the twins U-Nets. For target segmentation, the method with the higher OA, mIoU, P_d (up arrow) and the lower P_{fa} (down arrow) is more favorable.

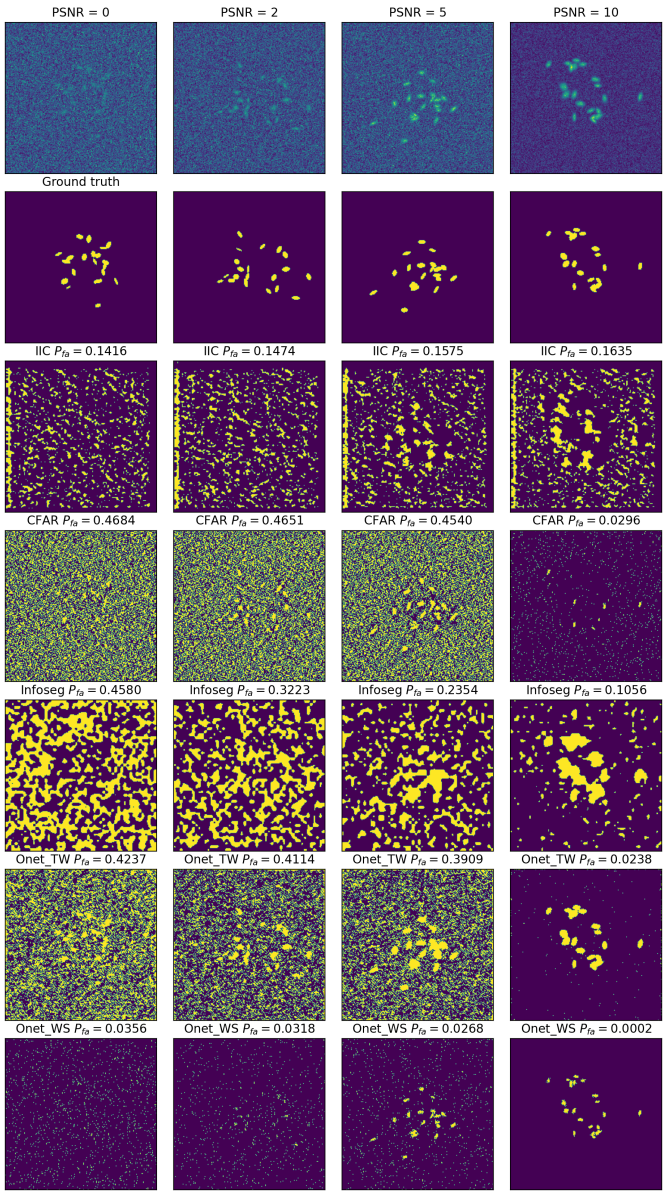

Fig. 2: Comparison of target segmentation in Rayleigh-distributed clutter with varied PSNRs for IIC, CFAR, Infoseg, Onet-TW, and Onet-WS. The first row illustrates the targets, ranging from weak (left) to strong (right). The second row shows the ground truth masks. In the third row, IIC appears sensitive to clutter and produces poor target shapes. The fourth row displays the CFAR segmentation with a comparable P_{fa} to Infoseg and Onet. The last three rows display the results for Infoseg and the two Onet modes, respectively. Compared to Infoseg, Onet-TW achieves a higher detection rate and better segmentation granularity. Onet-WS generates fewer false alarms and predicts favorable target shape when the PSNR is high (≥ 5), but loses sensitivity in detecting weak targets at low PSNR levels (≤ 2).

In Table I, we list the performance results under PSNR levels of 0, 2, 5, 10, and the overall average. In the radar target segmentation, the method that achieves the highest detection rate P_d while maintaining the same level (magnitude order) of P_{fa} is considered the best. For a fair comparison, we tune the P_{fa} of CFAR to approximately 0.45. In this experiment, the clutter background occupies more than 95% of the frame area. Methods with low P_{fa} tend to achieve higher OA regardless of their P_d. Notably, when PSNR=0, Infoseg, IIC and Onet-WS all achieve high OA with relatively lower P_{fa}. However, their P_d is much lower than that of Onet-TW. Onet-TW achieves a relatively high P_d (bigger than 0.85 even when PSNR=0). This demonstrates that Onet in twin mode can learn target feature in highly cluttered backgrounds. As the PSNR increases, both Infoseg and Onet improve their P_d and reduce P_{fa}. CFAR comes in second place for P_d. There is no significant change in P_d for IIC even as the PSNR increases to 10, indicating that IIC struggles to segment small target regions within the cluttered background.

Fig. 2 shows the target segmentation results for the five methods. For better visual comparison, CFAR uses two different P_{fa} values: one around 0.46 for PSNR≤ 5 and another around 0.03 for PNSR=10. Infoseg's segmentation tends to produce larger segmented particles compared to CFAR and Onet. It indicates that Infoseg is not good at clustering small extended targets. For PSNR=10, Infoseg's target region and spiky noise both exhibit high dilation. This is mainly caused by its strategy of feature representation and segmentation in down-sampled space. The subsequent interpolation back to the original input size impacts the granularity of segmentation.

In contrast, Onet segments the target regions with clearer boundaries.

It is noted that when PSNR ≤ 5, Onet-TW segments more target regions than Onet-WS, but at the cost of generating many more false alarms on spiky clutter. In the twin-Unet structure of Onet-TW, the input image X and its complement 1-X are processed separately by symmetrical U-Nets. This allows the power of weak targets to accumulate and influence network updates within the isolated networks, enabling the network to capture features associated with weak targets. In contrast, Onet-WS uses a single U-Net to process the complementary input pair simultaneously. The power of weak targets is easily canceled out by the reversed operation. For example, if a weak target has a strength of 0.5, its complementary would also be 1-0.5=0.5. When fed into the network, these complementary values generate opposite effects within the same network path, leading to a counteracting influence when computing the JSD loss. This counteracting behavior weakens Onet-WS's ability to detect weak targets, as the network effectively "cancels out" the relevant features. However, when the PSNR is high (≥ 5), Onet-WS's weight-sharing approach allows it to generate significantly fewer false alarms in mid-level noise, making it more effective in distinguishing strong targets from the background. Therefore, Onet-WS is better equipped to learn the feature differences between strong targets and dim backgrounds when the PSNR exceeds 5.

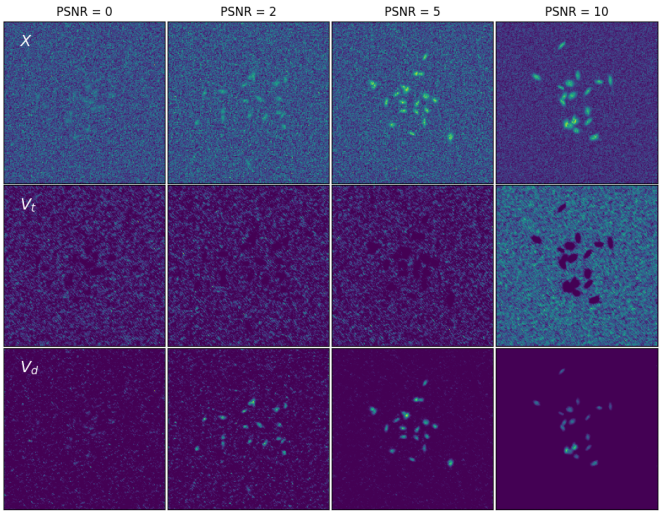

Fig. 3: Feature maps of V_t and V_d in Onet-TW for 4 frames with PSNR levels (0, 2, 5, 10). In each column, from up to down, the sub-figures display the input X, the feature map of the background V_t, and the foreground V_d, respectively, for a given PSNR. V_t and V_d show complementary energy distribution. As the PSNR increases, V_t reveals a more continuous background region, while V_d highlights clearer foreground targets. Notably, the foreground prediction V_d in Onet-TW visually corresponds to the denoising of the input X.

To further investigate Onet-TW's ability to detect weak targets, in Fig. 3 we observed the feature map of V_t and V_d before the softmax operation. When PSNR is low, the background becomes sparse in V_t, while the foreground gets spiky noises which are the source of the false alarms. This type of noise simulates sea clutter caused by high waves in the marine radar domain. Interestingly, in V_d targets are more easily detected than in the original input X. This suggests that Onet's prediction of the foreground effectively denoises the input X. To quantify this, we calculated the real average power σ_t of the target intensities in the ground truth masks and recorded its ratio to the clutter's average power σ_c in decibel (dB) form as the signal-to-noise ratio (SNR):

$$\text{SNR} = 10 \log_{10} \frac{\sigma_t}{\sigma_c}. \tag{16}$$

In table II we present the measured SNR of X and V_d for various PSNRs. The significant numerical improvement in SNR explains the cleaner background observed in V_d in Fig. 3.

TABLE II: Comparison of SNR between the source image X and Onet's foreground score map V_d at different PSNR levels. The results show that V_d consistently achieves higher SNR than X. The improvement in SNR becomes more pronounced as the PSNR increases.

PSNR	**SNR of X**	**SNR of V_d**	**Improvement** (dB)
0	1.88	5.58	+ 3.70
2	3.30	12.52	+ 9.22
5	5.90	23.46	+17.56
10	10.08	35.12	+25.04

D. Target Segmentation in The Real Rain Clutter of The Marine Radar Echoes

In this subsection, we compare the performance of various target segmentation methods applied to real rain clutter in marine radar echoes. The radar site was located at the 1st Sea Bathing Beach of Yantai, China. During data acquisition, a thunderstorm occurred in the view of the radar. The echoes of the cloud and rain masses were recorded with a scanning speed of 6 rotations per minute [37]. We used the 10th frame of the radar data (time stamp was 20200819144753). Fig. 4 provided an overview of the scene and rain clutter in the marine radar image. Eleven sub-regions in the rain clutter (PSNR=18.3 in average) were selected for target segmentation. We named them simply by alphabets from 'A' to 'K'. Each sub-region has a resolution of 200×200 pixels. The targets within each region were manually labeled for testing purposes.

Given the limited number of rain clutter images, we reused the Infoseg and Onet (in weight-sharing mode) models, which were trained on simulated clutter and targets as described in Section V-C, and directly tested them on the 11 sub-images.

TABLE III: Performance of target segmentation in rain clutter.

Method	**OA** $\uparrow$	**mIoU** $\uparrow$	$\mathbf{P_d}$ $\uparrow$	$\mathbf{P_{fa}}$ $\downarrow$
CFAR	0.9648	0.5105	0.4866	0.0322
Infoseg	0.9921	0.7089	**0.8740**	0.0072
Onet	**0.9976**	**0.7978**	0.6388	**0.0005**

While Infoseg exhibits higher detection rates compared to Onet, its false alarm rates are an order of magnitude higher.

Figure 5 illustrates that Onet's predictions demonstrate the best contour and the least clutter interference. This explains why Onet achieves the highest OA and mIoU in Table III. It is worth noting that Onet effectively learns how to represent extended targets in clutter, which enables it to suppress background clutter in the segmented foreground. However, when applied to ground sub-regions, Onet's segmentation may unintentionally suppress the ground echoes as well (see Fig.6). In practical maritime navigation, where radar overlays electronic charts to define target acquisition areas, Onet can serve as an effective tool for clutter suppression and target segmentation.

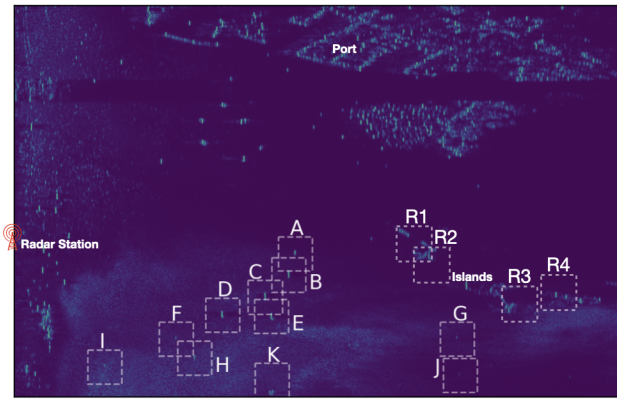

Fig. 4: Overview of the rain masses in the marine radar image (in azimuth-range coordinates). The radar data was collected at No.1 Bathing Beach, Yantai, China on 08/19/2020 [37]. The echoes of the rain masses appeared like a silk scarf at the bottom of the frame. Eleven sub-regions (named from 'A' to 'K' in alphabetical order) in the clutter contained the targets of interest. Four sub-regions of the islands were labeled from 'R1' to 'R4'. These sub-images (200×200) are marked with white-dash rectangles and used for model testing.

E. Cloud Segmentation in Remote Sensing Image Thumbnails

The ZY3 thumbnail dataset serves as a cloud segmentation benchmark for testing cloud detection performance across various terrains [32, 38]. The dataset consists of 250 training images without annotated labels and 50 test thumbnails with annotations. The resolutions of the thumbnails range from 1k $\times$ 1k to 3k $\times$ 3k. Detecting cloud masks in these thumbnails is more challenging due to their use of only RGB channels or a single grayscale channel, and their relatively small resolution compared to other multispectral remote sensing images [38].

In [32], the unsupervised domain adaption (UDA) model trains the segmentation backbone Deeplabv2 [39] using two other large annotated cloud datasets (containing over 50k samples) to learn representative features. It then further trains the additional unsupervised domain-invariant-feature-alignment network on the augmented ZY3 thumbnails.

In contrast, we train Onet, Infoseg, and IIC from scratch without any annotations. Since the cloud images have an average PSNR of 13.5, we adopt Onet in weight-sharing mode for comparison in this subsection. To reduce computational

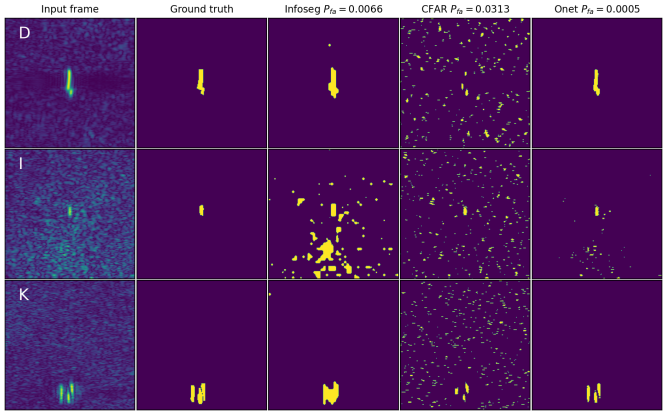

Fig. 5: The comparison of extended target segmentation in rain clutter for marine radar is presented for Infoseg, CFAR, and Onet in the 3rd to 5th columns, respectively. The first column displays the input sub-images 'D', 'I', and 'K', while the second column provides the true labels of the targets. Infoseg's coarse granularity expands both the targets and clutter, resulting in a higher probability of detection but also generating significantly more false alarms. CFAR misses more pixels of the true targets, even though its P_{fa} is two orders of magnitude higher than Onet's. Onet demonstrates the best target boundary accuracy.

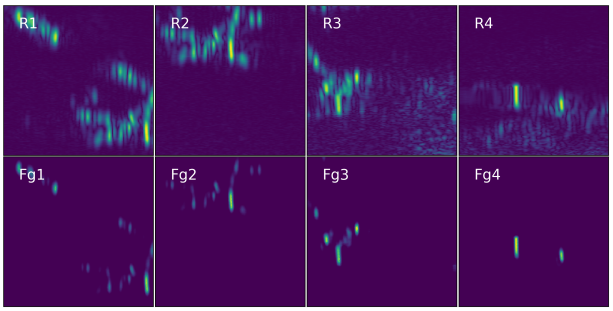

Fig. 6: The top row presents radar echoes from 4 islands, while the bottom row displays the corresponding foreground predictions of Onet, which has the side effect of suppressing weak ground echoes in the non-clutter environment.

load, we resize the thumbnails and crop them centrally to form square images of size 224×224. Onet successfully segments clouds in barren, wetland and urban terrains, as shown in Fig.7. Compared to the expert-annotated labels, Onet predicts much clearer cloud boundaries than Infoseg, which struggles to simultaneously segment clouds into both small clusters and large, uniform regions. This issue likely arises from Infoseg's reliance on a single global representation, which can lead to conflicts in clustering different clouds across the entire training set. In the last two rows of Fig.7, we display the background B_g and foreground F_g feature maps of Onet. Additional positive samples are provided in the supplementary material.

We note that Onet fails in snow/ice-covered mountains and barren lands (Fig. 8). These regions exhibit oversaturated intensity (pure white color), making it difficult for Onet to

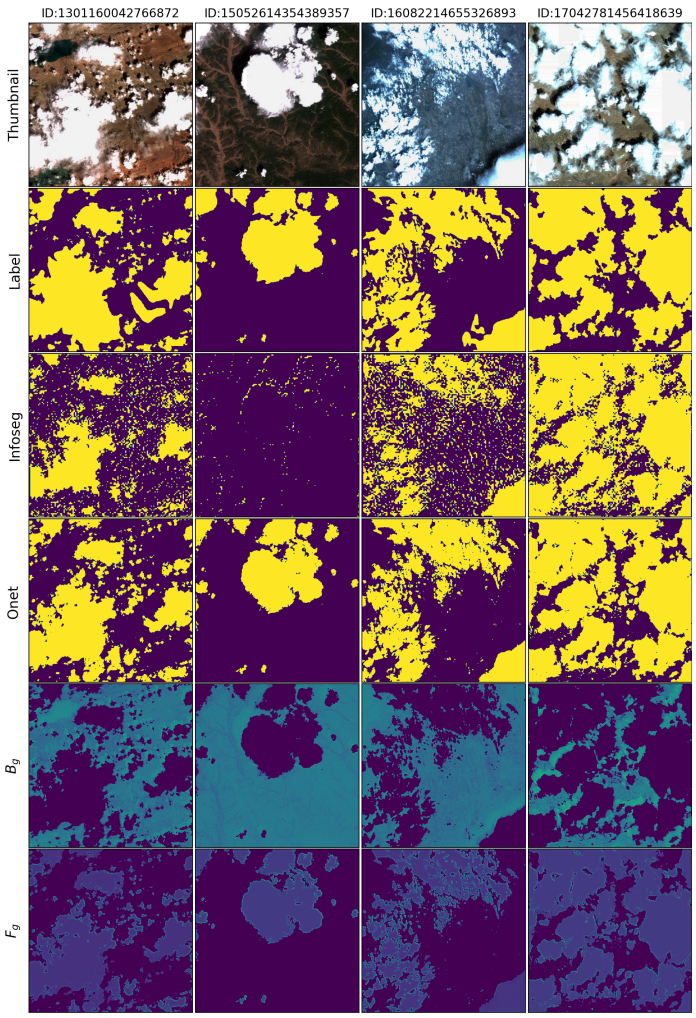

Fig. 7: Cloud segmentation in ZY3 thumbnails (positive examples). The thumbnails are displayed in pseudo-RGB format for better visual effect. Their unique IDs are written at the top of each subgraph. The domain experts spent 1 to 2 hours labeling the clouds at the pixel level for one image in the 2nd row [32]. The predictions of Infoseg and Onet are shown in the third and fourth rows, respectively. The last two rows show the feature maps of the background B_g and foreground F_g of Onet. Zooming in on the F_g subplot reveals the fine boundary of the cloud.

distinguish ice/snow from the clouds. The 2nd and 4th columns of Fig. 8 show that Onet also fails to segment thin and translucent clouds, which can easily be mistaken for background due to their low intensity. To further improve the segmentation performance of Onet, we apply data augmentation techniques, including snow removal and cloud enhancement, to the image set. Details of these techniques are discussed in the supplementary material.

We list the performance on all 50 test images in Table IV. IIC achieves the lowest scores both in mIoU and OA. The DeepLabv2 model was pre-trained on two annotated cloud benchmarks using a supervised approach and then tested on the ZY3 dataset. Compared to the UDA approach and its DeepLabv2 backbone, Onet achieves the best overall accuracy

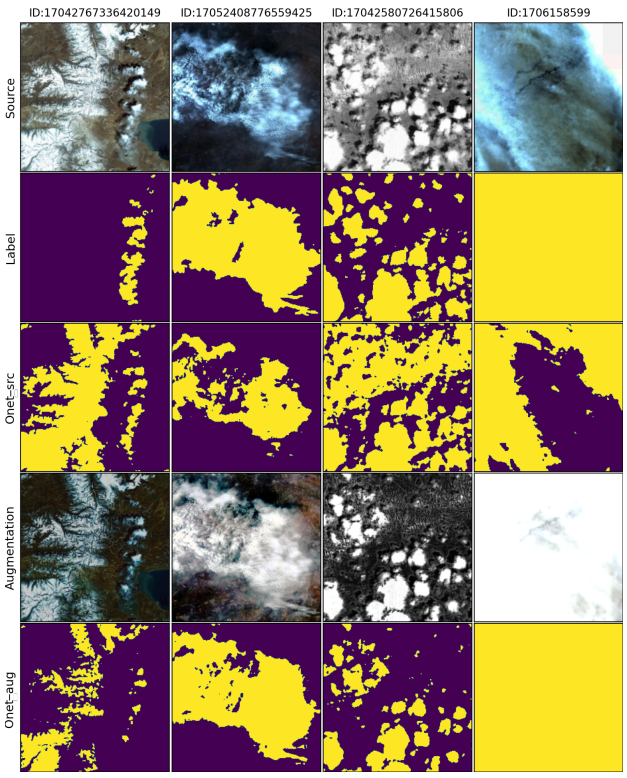

Fig. 8: Negative results for the ZY3 thumbnails are shown in the third row (Onet-src, Onet's prediction on the source image). In the first and third thumbnails, the terrains are covered with ice and snow. Onet-src fails to distinguish these from clouds due to their similar intensity distribution. In the other two thumbnails, Onet-src fails to identify the thin or semi-transparent clouds. However, with the augmentation of snow removal and cloud enhancement in the fourth row, Onet-aug produces better predictions, as shown in the last row.

(OA) score. The annotated cloud samples from snow/ice images enabled the UDA network to learn features that distinguish clouds from snow/ice. As a result, UDA achieves a better IoU for the background segmentation in the snow/ice terrains, leading to a superior mIoU. In contrast, Onet lacks the ability to differentiate between object types with similar intensity characteristics. However, its true positive (TP) and true negative (TN) rates are strong enough to maintain a relatively higher OA.

The key advance of Onet is that it can be trained from scratch on thumbnails without any annotations. This demonstrates the potential as an effective tool for fine labeling and as an attention probe for more advanced tasks.

TABLE IV: Performance of cloud segmentation. Mark '*' indicates the method uses supervised training or pre-training.

Model	**OA** ↑	**mIoU** ↑
IIC[8]	0.6417	0.4465
Infoseg[10]	0.7260	0.5542
Deeplabv2*[39]	0.8399	0.6730
UDA*[32]	0.9127	**0.8216**
Onet	**0.9208**	0.7842

F. Ablation Studies

In this subsection, we evaluate different design choices for Onet, including input configurations, weight-sharing mechanisms, segmentation heads, and backbone architectures.

1) Random Selected Samples Versus Complementary Input Pair: In this ablation study, we test two schemes for computing the marginal distributions $\mathbb{P}_{\mathrm{S_t}}$, $\mathbb{P}_{\mathrm{S_d}}$ used in (9) and (11). First, we use the conventional approach of Randomly Selected samples as Negatives (RSN) [12, 10] to estimate the marginal distribution in (9), and denote this model as Onet-RSN. Then, Onet utilizes the same architecture as Onet-RSN in weight-sharing mode, but computes the marginal distribution using the feature map generated by the complementary input pair, as described in (11). As shown in Table V, Onet significantly outperforms Onet-RSN across all evaluation metrics. This indicates that the use of complementary inputs is a critical factor that enhances the performance of the Onet model.

TABLE V: Performance comparison between Onet-RSN and Onet.

Dataset: Simulated clutter with PSNR 0-2

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-RSN	0.9007	0.4510	0.0081	0.0842
Onet	0.9522	0.4988	0.1359	0.0338

Dataset: Simulated clutter with PSNR 5-10

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-RSN	0.8126	0.4080	0.0371	0.1740
Onet	0.9867	0.7389	0.6198	0.0072

Dataset: Rain clutter

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-RSN	0.9435	0.5034	0.5688	0.4230
Onet	0.9976	0.7978	0.6388	0.0005

Dataset: ZY3

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-RSN	0.8158	0.6024	0.6392	0.1461
Onet	0.9208	0.7842	0.7852	0.0635

2) Twins U-Nets versus Weight-Sharing U-Net for the Backbone: In Section V-C, we discussed that the Onet-TW (twins U-Nets mode) has higher P_d and P_{fa} than Onet-WS (weight-sharing mode) in the simulated clutter environment. In high PSNR ($\geq$ 10) cases, we find that the Onet-WS alone can segment the target with good accuracy and relatively low P_{fa}. Since the datasets for rain clutter and cloud segmentation have high PSNR ($\geq$ 10), we suggest to use the Onet-WS as the default backbone. Table VI provides a detailed comparison between Onet-TW and Onet-WS.

3) Segmentation Head: Dot Product Versus Convolution: Binary segmentation networks with U-Net structure typically use 2D convolution layers to classify the dense global features H into two categories. In our proposal, we use the dot product between dense local features L and global features H to generate the score maps. We refer to the former as 'OPC' (Output via Convolution) and the latter as 'LHD' (L and H Dot-product) for comparison in Table VII. Since the Onet model adopts the weight-sharing mode in all the datasets except for the clutter with PSNR 0-2, we omit the 'WS' (Weight-Sharing) suffix for simplicity. We found that the LHD

TABLE VI: Performance comparison between twins (Onet-TW) and weight-sharing (Onet-WS) mode.

Dataset: Simulated clutter with PSNR 0-2

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-WS	0.9522	0.4988	0.1359	0.0338
Onet-TW	0.5490	0.2871	0.9018	0.4570

Dataset: Simulated clutter with PSNR 5-10

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-WS	0.9867	0.7389	0.6198	0.0072
Onet-TW	0.7679	0.4256	0.9994	0.2359

Dataset: Rain clutter

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-WS	0.9976	0.7978	0.6388	0.0005
Onet-TW	0.9874	0.7051	0.9175	0.0122

Dataset: ZY3

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-WS	0.9208	0.7842	0.7852	0.0635
Onet-TW	0.9222	0.7862	0.7752	0.0529

scores slightly higher P_d than OPC in the object segmentation of radar clutter. The reason is that LHD enhances more on the strength of local intensity, as the dot product between L and H forces the Onet to learn the global features in regions with strong local intensity. This strategy helps improve the target detection rate when the clutter is strong. In contrast, OPC enhances the global features, such as shape and boundary. When the target is immersed in strong clutter and the external contour is corrupted (e.g. in the low PSNR case), 'OPC' may struggle to focus on the target.

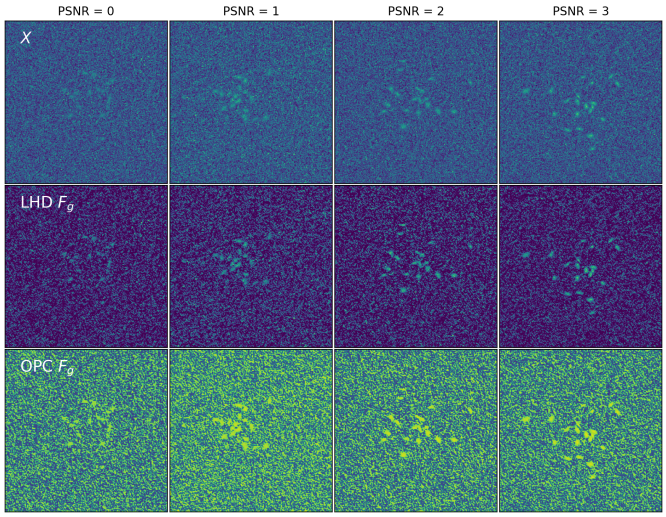

Fig. 9: Comparison of foreground score maps in Rayleigh-distributed clutter with low PSNR values for Onet-LHD-TW and Onet-OPC-TW. Segmentation head with LHD effectively denoises the input in low PSNR cases.

In Fig. 9, LHD model effectively reduces the noise and achieves better P_d. In the rain clutter and ZY3 cases, the performance of OPC and LHD is comparable in accuracy of segmentation (OA and mIoU), as shown in Table VII.

4) Choices of Backbones for the Modern U-Net: It has been observed that many new variants of the U-Net architecture

TABLE VII: Performance comparison of different segmentation heads.

Dataset: Simulated clutter with PSNR 0-2

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-LHD-TW	0.5409	0.2829	0.9097	0.4654
Onet-OPC-TW	0.5677	0.2973	0.8908	0.4378

Dataset: Simulated clutter with PSNR 5-10

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-LHD	0.9873	0.7168	0.6075	0.0063
Onet-OPC	0.9920	0.7727	0.5994	0.0014

Dataset: Rain clutter

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-LHD	0.9969	0.7864	0.6228	0.0005
Onet-OPC	0.9968	0.7810	0.6030	0.0004

Dataset: ZY3

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-LHD	0.9239	0.7947	0.8022	0.0623
Onet-OPC	0.9208	0.7842	0.7852	0.0635

have been proposed, incorporating modern backbones (e.g., ConvNeXt [40], Transformer [41], and Swin Transformer [42]) for supervised learning. In this ablation study, we explore the utilization of these network structures within the Onet's unsupervised learning framework. All the dense local features are extracted from the first encoding layer of the respective models, and all the models are evaluated in a weight-sharing mode.

Since the new U-Net variants use the OPC segmentation head, we replace the LHD with OPC head for the CNN based U-Net in Fig. 1 as the baseline for comparison (named Onet-CNN in Table VIII). In this experiment, Onet-ConvNeXt, Onet-TransUnet and Onet-SwinUnet utilize the same U-Net structures with 4 downscaling layers and 4 expanding layers [43, 44], but with ConvNeXt, Transformer and Swin-Transformer blocks in each layer, respectively.

In low PSNR (≤ 2) scenarios, Onet-CNN exhibits the best detection rate P_d, showcasing its ability to denoise the clutter. In contrast, the other modern backbones demonstrate more power in global semantic representation. Therefore, in low PSNR conditions with strong clutter, these models tend to be biased towards the texture variation caused by the spiky clutter and lack the ability to effectively detect the targets. For instance, Onet-ConvNeXt and Onet-SwinUnet have a zero P_d in simulations with PSNR 0-2. While their high OA and mIoU scores may appear impressive, they are essentially meaningless (which are marked in gray background color in Table VIII), as these models omit all the targets in such low PSNR dataset. Conversely, in the high PSNR cases (including the rain clutter and ZY3 datasets), the modern U-Net variants outperform Onet-CNN in terms of P_d, OA and most mIoU scores, showcasing their enhanced ability to learn the shape and contour of strong targets.

VI. Conclusion

In this work, we leverage the dense local and global features of U-Net to generate the segmentation score map. By using complementary input pairs, we maximize the mutual information between the dense feature and score map by increasing their Jensen-Shannon divergence. This unsupervised approach helps to effectively segment extended targets in cluttered environment. However, Onet struggles to learn distinguishable information between targets when they have similar intensity characteristics. For instance, in remote sensing imagery where clouds and snow/ice exhibit equally strong intensities, Onet fails to reliably differentiate clouds from snow/ice. This indicates that further modeling efforts are required to address this challenge.

TABLE VIII: Performance comparison of different backbones.

Dataset: Simulated clutter with PSNR 0-2

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-CNN	**0.9522**	**0.4988**	**0.1359**	0.0338
Onet-ConvNeXt	0.9833	0.4917	0.0000	0.0000
Onet-TransUnet	0.8594	0.4348	0.0894	**0.0128**
Onet-SwinUnet	0.9833	0.4917	0.0000	0.0000

The rows with gray background color are unreliable for its zero P_d

Dataset: Simulated clutter with PSNR 5-10

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-CNN	0.9916	0.7726	0.5985	0.0017
Onet-ConvNeXt	0.9929	0.7839	0.5750	**3.8E-6**
Onet-TransUnet	**0.9934**	**0.8123**	**0.6900**	0.0016
Onet-SwinUnet	0.9929	0.7811	0.5696	4.5E-6

Dataset: Rain clutter

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-CNN	0.9969	**0.7912**	0.6017	**0.0002**
Onet-ConvNeXt	**0.9976**	0.7796	**0.6607**	0.0006
Onet-TransUnet	0.9954	0.7356	0.6472	0.0024
Onet-SwinUnet	**0.9976**	0.7764	0.6570	0.0007

Dataset: ZY3

Model	OA ↑	mIoU ↑	$\mathbf{P_d}$ ↑	$\mathbf{P_{fa}}$ ↓
Onet-CNN	0.9208	0.7842	0.7852	0.0635
Onet-ConvNeXt	**0.9250**	0.8010	0.8184	**0.0628**
Onet-TransUnet	0.9243	0.7981	0.8144	0.0641
Onet-SwinUnet	0.9247	**0.8011**	**0.8212**	0.0650

VII. Acknowledgment

The authors would like to thank Mr. Jianhua Guo for providing the ZY3 Cloud datasets. They also wish to express their gratitude to the anonymous reviewers for their valuable time and insightful comments, which have helped improve the quality of this paper. This work is supported by the NSFC Projects (92370124, 62350080, 92467108, 62141604, 62032016), Beijing Natural Science Foundation L247011, and Beijing Nova Program (20220484106, 20230484451).

References

[1] S. Aich, W. van der Kamp, and I. Stavness, "Semantic binary segmentation using convolutional networks without decoders," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 182–186. 1

[2] M. Maška et al., "The cell tracking challenge: 10 years of objective benchmarking," *Nature Methods*, 2023. 1

[3] G. Vivone and P. Braca, "Joint probabilistic data association tracker for extended target tracking applied to x-band marine radar data," *IEEE Journal of Oceanic*

Engineering, vol. 41, no. 4, pp. 1007–1019, Oct. 2016. 1

[4] S. Mahajan and B. Fataniya, "Cloud detection methodologies: variants and development—a review," *Complex & Intelligent Systems*, vol. 6, no. 2, pp. 251–261, 2020. 1

[5] S. Mohajerani, T. A. Krammer, and P. Saeedi, "A cloud detection algorithm for remote sensing images using fully convolutional neural networks," in *IEEE 20th International Workshop on Multimedia Signal Processing*, 2018, pp. 1–5. 1

[6] H. Kim, D. Kim, and S.-M. Lee, "Marine object segmentation and tracking by learning marine radar images for autonomous surface vehicles," *IEEE Sensors Journal*, vol. 23, no. 9, pp. 10 062–10 070, 2023. 1

[7] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 4037–4058, 2021. 1

[8] X. Ji, A. Vedaldi, and J. Henriques, "Invariant information clustering for unsupervised image classification and segmentation," in *International Conference on Computer Vision*, 2019, pp. 9864–9873. 1, 2, 5, 9

[9] Y. Ouali, C. Hudelot, and M. Tami, "Autoregressive unsupervised image segmentation," in *European Conference on Computer Vision*, 2020, pp. 142–158. 1, 2

[10] R. Harb and P. Knöbelreiter, "Infoseg: Unsupervised semantic image segmentation with mutual information maximization," in *German Conference on Pattern Recognition 2021*. Springer, Oct. 2021. 1, 2, 3, 5, 9, 10

[11] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9726–9735. 1, 2

[12] D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio, "Learning deep representations by mutual information estimation and maximization," in *International Conference on Learning Representations*, 2019, pp. 9153–9176. 1, 2, 3, 10

[13] M. Hamilton, Z. Zhang, B. Hariharan, N. Snavely, and W. T. Freeman, "Unsupervised semantic segmentation by distilling feature correspondences," in *International Conference on Learning Representations*, 2022, pp. 14 716–14 741. 1, 2

[14] D. Niu, X. Wang, X. Han, L. Lian, R. Herzig, and T. Darrell, "Unsupervised universal image segmentation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 22 744–22 754. 1

[15] L. A. Gatys, A. S. Ecker, and M. Bethge, "Texture synthesis using convolutional neural networks," in *International Conference on Neural Information Processing Systems*, vol. 1, 2015, pp. 262–270. 1

[16] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015. 1

[17] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention*, 2015, pp. 234–241. 1, 2, 3

[18] M. I. Belghazi, A. Baratin, S. Rajeshwar, S. Ozair, Y. Bengio, A. Courville, and D. Hjelm, "Mutual information neural estimation," in *Proceedings of the International Conference on Machine Learning*, vol. 80, 2018, pp. 531–540. 2, 3

[19] T. Falk et al., "U-net: deep learning for cell counting, detection, and morphometry," *Nature Methods*, vol. 16, no. 1, pp. 67–70, 2019. 2

[20] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3d u-net: Learning dense volumetric segmentation from sparse annotation," in *Medical Image Computing and Computer-Assisted Intervention*, 2016, pp. 424–432. 2

[21] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: A nested u-net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, 2018, pp. 3–11. 2

[22] N. Siddique, S. Paheding, C. P. Elkin, and V. Devabhaktuni, "U-net and its variants for medical image segmentation: A review of theory and applications," *IEEE Access*, vol. 9, pp. 82 031–82 057, 2021. 2

[23] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, and B. Xiao, "Deep high-resolution representation learning for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 10, pp. 3349–3364, 2021. 2

[24] X. Feng, C. Wang, S. Cheng, and L. Guo, "Automatic liver and tumor segmentation of ct based on cascaded u-net," in *Proceedings of 2018 Chinese Intelligent Systems Conference*, 2019, pp. 155–164. 2

[25] M. K. Abd-Ellah, A. A. M. Khalaf, A. I. Awad, and H. F. A. Hamed, "Tpuar-net: Two parallel u-net with asymmetric residual-based deep convolutional neural network for brain tumor segmentation," in *Image Analysis and Recognition*, 2019, pp. 106–116. 2

[26] D. Kwon, J. Ahn, J. Kim, I. Choi, S. Jeong, Y.-S. Lee, J. Park, and M. Lee, "Siamese u-net with healthy template for accurate segmentation of intracranial hemorrhage," in *Medical Image Computing and Computer Assisted Intervention*, 2019, pp. 848–855. 2

[27] O. J. Hénaff, A. Srinivas, J. De Fauw, A. Razavi, C. Doersch, S. M. A. Eslami, and A. Van Den Oord, "Data-efficient image recognition with contrastive predictive coding," in *Proceedings of International Conference on Machine Learning*, 2020, pp. 4182–4192. 2

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778. 2

[29] S. Kullback, *Information Theory and Statistics*. Courier Corporation, 1997. 3

[30] S. Nowozin, B. Cseke, and R. Tomioka, "F-gan: Training generative neural samplers using variational divergence minimization," in *Proceedings of International Conference on Neural Information Processing Systems*, 2016, pp. 271–279. 3

[31] J. Lin, "Divergence measures based on the shannon entropy," *IEEE Transactions on Information Theory*, vol. 37, no. 1, pp. 145–151, 1991. 3

[32] J. Guo, J. Yang, H. Yue, X. Liu, and K. Li, "Unsupervised domain-invariant feature learning for cloud detection of remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022. 5, 8, 9

[33] H. M. Finn and R. S. Johnson, "Adaptive detection mode with threshold control as a function of spatially sampled clutter level estimates," *RCA Review*, vol. 29, pp. 414–464, 1968. 5

[34] Y. He, J. Guan, and X. Meng, *Radar target detection and CFAR processing*. Tsinghua University Press, 2011. 5

[35] Y. Zhou, H. Su, S. Tian, X. Liu, and J. Suo, "Multiple-kernelized-correlation-filter-based track-before-detect algorithm for tracking weak and extended target in marine radar systems," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 58, no. 4, pp. 3411–3426, 2022. 5

[36] H. Jiang, W. Yi, T. Kirubarajan, L. Kong, and X. Yang, "Multiframe radar detection of fluctuating targets using phase information," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 2, pp. 736 – 749, 2017. 5

[37] N. Liu, H. Ding, Y. Huang, Y. Dong, G. Wang, and K. Dong, "Annual progress of sea-detecting x-band radar and data acquisition program," *Journal of Radars*, vol. 10, no. 1, 2021. 7, 8

[38] J. Yang, J. Guo, H. Yue, Z. Liu, H. Hu, and K. Li, "Cdnet: Cnn-based cloud detection for remote sensing imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 8, pp. 6195–6211, 2019. 8

[39] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "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, pp. 834–848, 2018. 8, 9

[40] Z. Liu, H. Mao, C. Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," in *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 976–11 986. 11

[41] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *International Conference on Neural Information Processing Systems*, 2017, pp. 6000–6010. 11

[42] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," *International Conference on Computer Vision (ICCV)*, 2021. 11

[43] J. Chen, Y. Lu, Q. Yu, X. Luo, E. Adeli, Y. Wang, L. Lu, A. L. Yuille, and Y. Zhou, "Transunet:transformers make strong encoders for medical image segmentation," *arXiv:2102.04306*, 2021. 11

[44] H. Cao, Y. Wang, J. Chen, D. Jiang, X. Zhang, Q. Tian, and M. Wang, "Swin-unet: Unet-like pure transformer for medical image segmentation," in *Proceedings of the European Conference on Computer Vision Workshops(ECCVW)*, 2022, pp. 205–218. 11

Yi Zhou received the B.S. and M.S. degrees of Electronic Engineering from Dalian Maritime University, Dalian, China, in 2003 and 2006 respectively. He obtained a joint educated Ph.D. degree in Signal Processing from the Institute of Image Communication and Information Processing at Shanghai Jiao Tong University, Shanghai, China, and ICD/LM2S, University of Technology of Troyes, France in 2012. Since 2013, he has been a lecturer in the Department of Electronic Information Engineering at Dalian Maritime University, Dalian, China. He briefly visited the TSAIL group of Tsinghua University for three months in Nov. 2021. His research interests include signal processing and computer vision.

Hang Su (IEEE member), is an associate professor in the Department of Computer Science and Technology at Tsinghua University, Beijing, China. He received his Ph.D. degree from Shanghai Jiao Tong University in 2014 and worked as a visiting scholar at Carnegie Mellon University from 2011 to 2013. His research interests include the adversarial machine learning and robust computer vision, areas in which he has published more than 50 papers including CVPR, ECCV, TMI, etc. He has served as area chair for NeurIPS and as a workshop cochair in AAAI 2022. He received "Young Investigator Award" from MICCAI2012, the "Best Paper Award" in AVSS2012, and "Platinum Best Paper Award" in ICME2018.

Tian Wang received the M.S. degree from Xi'an Jiaotong University, China, in 2010, and the Ph.D. degree from the University of Technology of Troyes, France, in 2014. He is an associate professor at the School of Artificial Intelligence, Beihang University. His research interests include computer vision and pattern recognition.

Qing Hu received the Ph.D. degree in Communication and Information Systems from Dalian Maritime University in 2011. He is currently a professor in the School of Information Science and Technology of Dalian Maritime University and the director of the National Engineering Research Center of Maritime Navigation System. His main research interests include marine intelligent communication and navigation technology, intelligent ship networking technology, maritime intelligent traffic management technology, and e-navigation strategy.